SYS-CON

JMS: MORE THAN ASYNCHRONOUS DATA TRANSFER PG 26

JAN/FEB 2005 VOL:4 ISSUE:1

wldj™

WWW.WLDJ.COM

THE LEADING INDEPENDENT MAGAZINE FOR WEBLOGIC™ PROFESSIONALS

Perimeter Authentication via Identity Assertion

WHERE THE BACK END TRUSTS THE FRONT END ...14

PLUS...

$8.99US $9.99CAN

01

0 09281 03424 7

TRANSACTION: *Measuring the Value of Software Infrastructure* page 10

MANAGEMENT: *Strategies for WebLogic Domain Configuration* page 20

JROCKIT: *Memory Leak Detection with the JRockit JVM* page 46

Businesses spend more than $80 billion annually on integration.

$19 billion of that cost is spent on manually reconciling data between systems.

Automating the Exchange of Data Across an SOA

Over 40% of IT development time is spent on manually reconciling and validating the data shared between systems - now there is a better way.

Pantero's unique Exchange Modeling™ technology captures data reconciliation and validation operations as metadata and generates Shared Data Services where operations can be enforced, reused, and dynamically redeployed without incurring additional integration costs. And, Pantero is seamlessly integrated into the BEA 8.1 Platform.

Contact us today!

781-890-2890

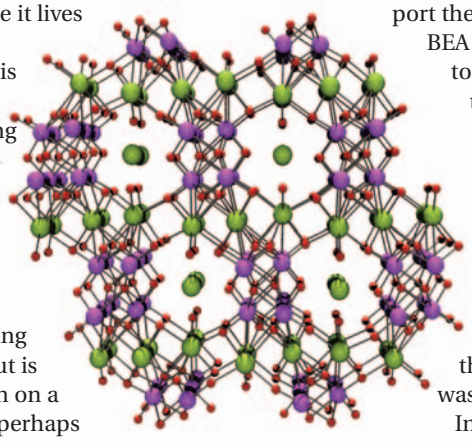
www.pantero.com

Simply SOA

By Joe Mitchko

It's all quite simple you see. In a major move last year, BEA made a significant and very welcome contribution to the open source community by donating the Beehive framework to the Apache software foundation. Beehive, perceived by many in the industry as somewhat proprietary in nature, is the driving technology behind the BEA WebLogic Workshop IDE and was engineered with one thing in mind: to help make your life easier as a WebLogic and J2EE developer. As I have heard over and over from the marketing folks, it takes care of all of the plumbing for you. And with true respect, I believe that to a certain degree it lives up to its promises.

Moreover, at the time of this writing, the Apache Beehive Incubation Project is humming along quite nicely, and has already produced an alpha release that is available for download. All of a sudden, easy J2EE programming is no longer just some proprietary solution from one of the leading application server vendors, but is now open and available to run on a number of platforms. At last, perhaps things will get simpler you say, when the software industry finally accepts this technology and it catches on like other open source success stories, like for instance Struts. And one day it may even officially become part of the J2EE standard. Well, that day may come sooner than you think.



You can sum up the up-and-coming J2EE 1.5 release with two words: Java annotations. It seems that the Java development community is yearning for simplicity too. It is just too complex they say. Java annotations, which are based on JSR 175, provide the developer with more power, and with the wave of a wand and declaring a few things in the Java comments, you can make a simple Java class turn into a Web service. Now, as a programmer, I can truly savor the simplicity of not having to worry about such things as WSDL generation and SOAP marshalling. It lets me concentrate on solving the business problems (the core coding), and lets the annotations take care of the rest. This, of course, is the essence of how WebLogic Workshop and its underlying Beehive framework work, and it is just a matter of time before it becomes officially part of the J2EE JDK. At last, the developer's plea for simplicity is being heard by the vendors, the open source community, and the standards committees that are driving the future of J2EE development.

In this lies one big monkey wrench that will surely make us yearn for the good old days of editing deployment descriptors by hand, and that is what the SOA world has in store for us. Everyone is expounding on how their product will support the SOA world of tomorrow, and BEA is no exception when it comes to this. But when you break the thin veneer of the SOA hype, and dive down into the details of the still emerging set of WS-* standards, the visions of simplicity quickly begin to disappear. I would tend to agree with the old timers who say SOA is just rehashed technology from yesteryear, with the likes of CORBA, etc. We'll, it wasn't so simple back then either.

Integration is one of the most difficult tasks to deal with as a developer or architect, and the same old problems that plagued us back then are still around to haunt us today. To address these complexities, emerging standards such as WS-CAF define transactional activity across multiple Web services, and introduce old concepts, such as transactional context, into a new world of Web services. BEA has done the industry a great service by leading the way for simpler J2EE development. Unfortunately, it will not be nearly enough to make our life easier in the SOA world of tomorrow.

Author Bio:
Joe Mitchko is the editor-in-chief of ***WLDJ*** and a senior technical specialist for a leading consulting services company.

Contact: joe@sys-con.com

wldj THE LEADING INDEPENDENT MAGAZINE FOR WEBLOGIC PROFESSIONALS

EDITORIAL ADVISORY BOARD
Lewis Cirne, Wayne Lesley Lund, Chris Peltz, Sean Rhody

EDITORIAL

Founding Editor
Peter Zadrozny

Editor-in-Chief
Joe Mitchko joe@sys-con.com

Executive Editor
Gail Schultz gail@sys-con.com

Editor
Nancy Valentine nancy@sys-con.com

Associate Editors
Jamie Matusow jamie@sys-con.com
Natalie Charters natalie@sys-con.com

Assistant Editors
Seta Papazian seta@sys-on.com

Research Editor
Bahadir Karuv, PhD bahadir@sys-con.com

PRODUCTION

Production Consultant
Jim Morgan jim@sys-con.com

Lead Designer
Abraham Addo abraham@sys-con.com

Art Director
Alex Botero alex@sys-con.com

Associate Art Directors
Louis F. Cuffari louis@sys-con.com
Richard Silverberg richards@sys-con.com
Tami Beatty tami@sys-con.com
Andrea Boden andrea@sys-con.com

CONTRIBUTORS TO THIS ISSUE

Anjali Anagol-Subbarao, Peter Holditch, Prakash Malani, Joe Mitchko, Helena Aberg Ostlund, John-Axel Strahlman, Raman Sud, Sudhir Upadhyay

EDITORIAL OFFICES

SYS-CON MEDIA
135 Chestnut Ridge Rd., Montvale, NJ 07645
Telephone: 201 802-3000 Fax: 201 782-9638
WLDJ (ISSN #1535-9581)
is published bimonthly (6 times a year)
by SYS-CON Publications, Inc.

Postmaster: send address changes to:
WLDJ: SYS-CON MEDIA
135 Chestnut Ridge Rd., Montvale, NJ 07645

Worldwide Newsstand Distribution
Curtis Circulation Company, New Milford, NJ
FOR LIST RENTAL INFORMATION:
Kevin Collopy: 845 731-2684,
kevin.collopy@edithroman.com
Frank Cipolla: 845 731-3832,
frank.cipolla@epostdirect.com

REPRINTS

For promotional reprints, contact reprint coordinator Kristin Kuhnle, kristin@sys-con.com. SYS-CON Publications, Inc., reserves the right to revise, republish and authorize its readers to use the articles submitted for publication.

Why is it that some Java guys are more relaxed than others?

These days, with everything from customer service to sales and distribution running on Java, keeping your enterprise applications available can be pretty stressful.

Unless of course, you've discovered the power of Wily. No other software offers our level of insight. Which means you'll be able to monitor transactions and collaborate with colleagues in real-time. Even more important, you'll be able to optimize performance across the entire application—end-to-end.

So put Wily to work. And keep performance problems from pushing you over the edge.

Get Wily.™

1 888 GET WILY | wilytech.com

wily technology

Create a Real-World Business Process Model, Part 4

By Anjali Anagol-Subbarao

HOW CHANGING AN ORDER CAN BE ADDED TO AN ERP-BASED SYSTEM

In the first article in this series (Vol. 3, issue 6), I gave you an overview of business process management (BPM) and covered the specifications in this area. I described the order change example and the steps needed to create the business process in WebLogic Integration (WLI). In the second article I looked at how to create a process application (orderChange). In this application I created a new process called orderChange.jpd. To start the process we added a ClientRequest received. Next we added the Web service validate config.

In the third article I added a decision point to handle the result from validate config Web service. The decision point helps in handling both the positive and negative outcomes of the result from the process. Then I added a database control to the process, which checks the status of the order to be changed. Finally, I added another decision node to handle the result from the database Control. In this article we will see how the change order is written out to a file. We will also see how this change order can be added to an ERP-based system (SAP). The code for the process will be examined.

Add a File Control

First let's see how to add the file control to the process.If the orderStatus allows for the order to be changed, the order can be written out to a file through a File Control. Alternatively, you can change an order in SAP using the Application View Control as decribed below.

A File Control makes it easy to read, write, or append to a file in a file system. You can also use the File Control to copy, rename, and delete files. To write to a file control you can add the File Control-ChangeorderFile to the process as shown in Figure 1. This writes an XML file to c:/bea directory, which can then be used to change the order. Once you have written the XML file to the directory, you can use it to update your order management systems.

Add an Application View Control for SAP

To write the order change to SAP, we need to create an integration solution from WebLogic to SAP using an SAP adapter. The integration framework in WLI is based on J2EE Connector Architecture 1.0. WLI provides adapters, application views, and Application View Control for integrating with an EIS.

The SAP adapter provides integration with SAP Business APIs (BAPIs), which are interfaces you can use to link your applications to SAP components. BAPI calls are synchronous and return information. This information is either error notification or a well-formed XML document containing the result of the BAPI call. The adapter also provides integration to Intermediate Documents (IDocs). These calls are asynchronous and do not return any information synchronously. The third integration is to Remote Function Calls (RFCs). RFCs are calls in which the application establishes a connection to the SAP sys-

Author Bio:
Anjal Anagol-Subbarao works in HP's IT organization as an IT architect. She has 12 years of IT experience, the last 5 in Web services. Her book on J2EE Web services on BEA WebLogic will be published this year.

Contact:
anjali.anagol-subbarao@hp.com

SYS-CON MEDIA

PRESIDENT & CEO
Fuat Kircaali fuat@sys-con.com

VP, BUSINESS DEVELOPMENT
Grisha Davida grisha@sys-con.com

GROUP PUBLISHER
Jeremy Geelan jeremy@sys-con.com

ADVERTISING
Senior VP, Sales & Marketing
Carmen Gonzalez carmen@sys-con.com
VP, Sales & Marketing
Miles Silverman miles@sys-con.com
Advertising Director
Robyn Forma robyn@sys-con.com
Advertising Manager
Megan Mussa megan@sys-con.com
Advertising Sales Manager
Dennis Leavey dennis@sys-con.com
Associate Sales Managers
Kristin Kuhnle kristin@sys-con.com
Dorothy Gil dorothy@sys-con.com
Kim Hughes kim@sys-con.com

SYS-CON EVENTS
President, Events
Grisha Davida grisha@sys-con.com
National Sales Manager
Jim Hanchrow jimh@sys-con.com

CUSTOMER RELATIONS
Circulation Service Coordinators
Edna Earle Russell edna@sys-con.com
Linda Lipton linda@sys-con.com
Monique Floyd monique@sys-con.com
Manager, JDJ Store
Brunilda Staropoli bruni@sys-con.com

SYS-CON.COM
Vp, Information Systems
Robert Diamond robert@sys-con.com
Web Designers
Stephen Kilmurray stephen@sys-con.com
Matthew Pollotta matthew@sys-con.com
Online Editor
Martin Wezdecki martin@sys-con.com

ACCOUNTING
Financial Analyst
Joan LaRose joan@sys-con.com
Accounts Payable
Betty White betty@sys-con.com
Credits & Accounts Receivable
Stephen Michelin smichelin@sys-con.com

SUBSCRIPTIONS
subscribe@sys-con.com
Call 1-888-303-5282
For subscriptions and requests for bulk orders, please send your letters to Subscription Departme
Cover Price: $8.99/issue
Domestic: $149/yr (12 issues)
Canada/Mexico: $169/yr All other countries $179/
(U.S. Banks or Money Orders)
Back issues: $12 U.S. (plus S&H)

JANUARY BUZZ

Combining AEP Practices for Effective Error Prevention

As I mentioned in last month's Buzz, the AEP practices of coding standards and unit testing should be considered two sides of the same coin. Each practice can uncover problems that the other cannot. Consequently, to identify and prevent a wide range of software problems, it's important to perform *both* coding standard analysis and unit testing, not just one or the other.

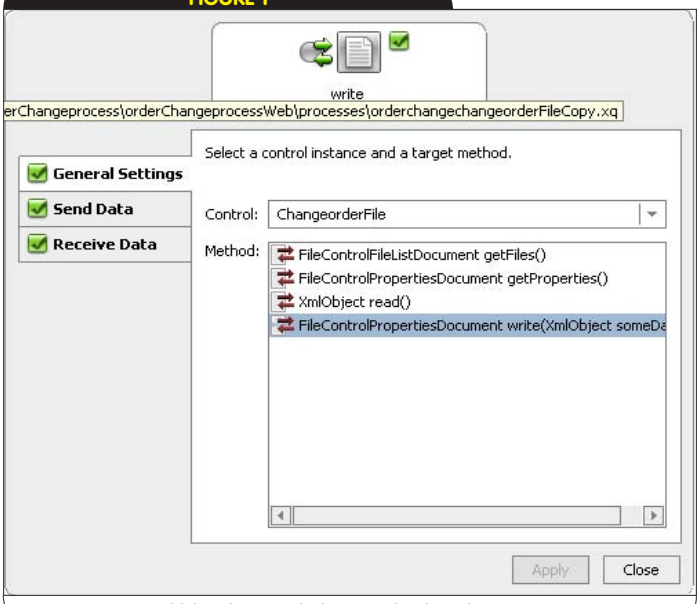
This month, I'd like to elaborate on how combining specific AEP practices is the most effective way to identify and prevent specific errors. For example, let's consider resource or memory leaks. Application-level runtime error detection and/or profiling can identify leaks, but typically not until late in the development process, when correcting the leak is typically time-consuming, difficult, and expensive. With the appropriate technologies, leaks could be detected at the unit level. However, unless the development team is required to close all resources that are opened within a unit, a leak found at the unit level might not indicate a true application-level leak.

One effective solution to this dilemma is to combine coding standard enforcement, unit-level leak detection, and application-level leak detection/profiling. First, implement a coding standard that requires that the same class or function which allocates a resource must also deallocate that resource. By checking compliance with this coding standard, you can ensure that the team's developers write code in a way that prevents leaks. If this coding standard does not make sense in all situations, individual exceptions can be documented and suppressed. You can then use unit-level leak detection to dynamically verify that leaks do not occur within the unit. After all units are developed in this manner, you can use application-level leak detection and/or profiling to verify whether leaks occur at the application level. When the team works in this manner, significantly fewer leaks reach the application level, where they are more time-consuming, difficult, and expensive to fix.

The Parasoft AEP Methodology is constructed to combine multiple practices in this supportive manner. It encourages development teams to determine which types of errors they need to prevent, then combine practices to create a "leak-proof" system for identifying and preventing those types of errors. For details about this methodology, visit **www.parasoft.com** .

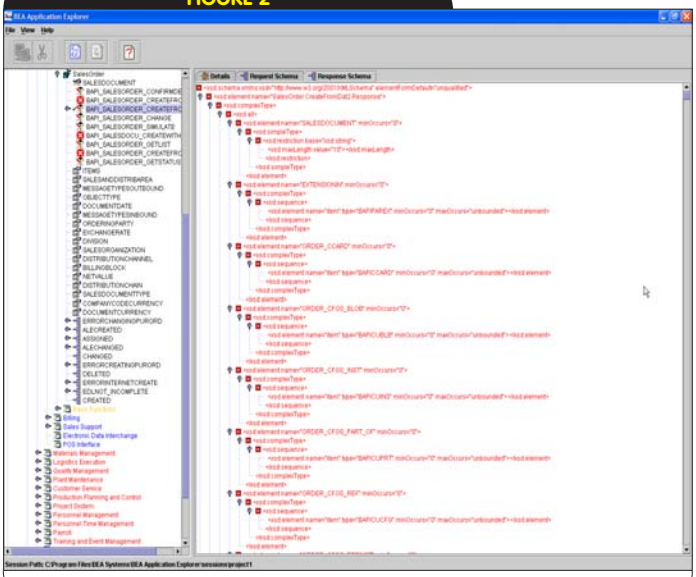
– Adam Kolawa, *Ph.D.*
Chairman/CEO of Parasoft

FIGURE 1



Add the File Control-ChangeorderFile to the process

FIGURE 2



Requst and response schemas

tem (using a valid User ID) and then issues a call to an SAP function. RFC calls are synchronous and usually return information.

To design an application integration solution with an SAP adapter, you must first download the SAP Adapter from the BEA Web site and the SAP JCo from the SAP Web site. To place an order in SAP from BEA WebLogic, you need to generate schemas for the Create Sales Order BAPI. This BAPI will facilitate creation of the order in SAP.

To generate schemas for SAP Business Objects (which will facilitate creation of the order), you need to install BEA Application Explorer. To create schemas you first need to either establish a new connection with SAP or use an existing connection. For a new connection you must name the connection (e.g., D7b), application server system number, client number, username, and password. When you connect to SAP, all the application components, IDocs, and RFCs are pulled into the Application Explorer. We specifically want to create a schema for the Change Sales Order BAPI. We can do that by right clicking on the BAPI and creating the request and response schemas (see Figure 2). These schemas and the manifest.xml file are stored in the working directory.

Next we need to define an RFC remote destination in SAP. You must define an SAP remote destination so that the SAP system can send IDocs to the adapter and respond to RFCs and BAPIs. This SAP remote destination must be defined before you create your application view.

Now create the application view by using the Application View Console. Select the SAP adapter and then create a new browsing connection. You then need to configure your service with or without load balancing. Figure 3 shows an example of SAP without load balancing. To test a service, go to the Application View Administration page and click the Test link next to the service to be tested. In the Test Service window, copy the appropriate XML strings from the SAP request. When you click Test, the results appear in the Test Results window.

After you have created an application view to send and receive schemas, a Control can be created from the application view. This Application View Control for SAP can be used in business processes.

Code for orderChange Business Process

In this section we'll look at the JPD for the orderChange business process and the code for the different parts of the process that we have created in the last few articles. Let's look at the details.

Start of the process:

```
/**
 * @jpd:process process::
```

Name of the process:

```
 * <process name="orderchange">
```

OrderChangeRequest as ClientRequest:

```
 *   <clientRequest name="orderChangeRequest" method="orderChangeRequest
"/>

Process calling the validateConfig Web service:

 *   <controlSend name="validateConfig" method="validateConfignewValidateC
onfig"/>
```

First decision point to check if the configuration is valid:

```
 *   <decision name="Is configuration Valid?">
 *     <if name="true" condition="cond_outValidateConfig_1($outValidate-
Config)">
```

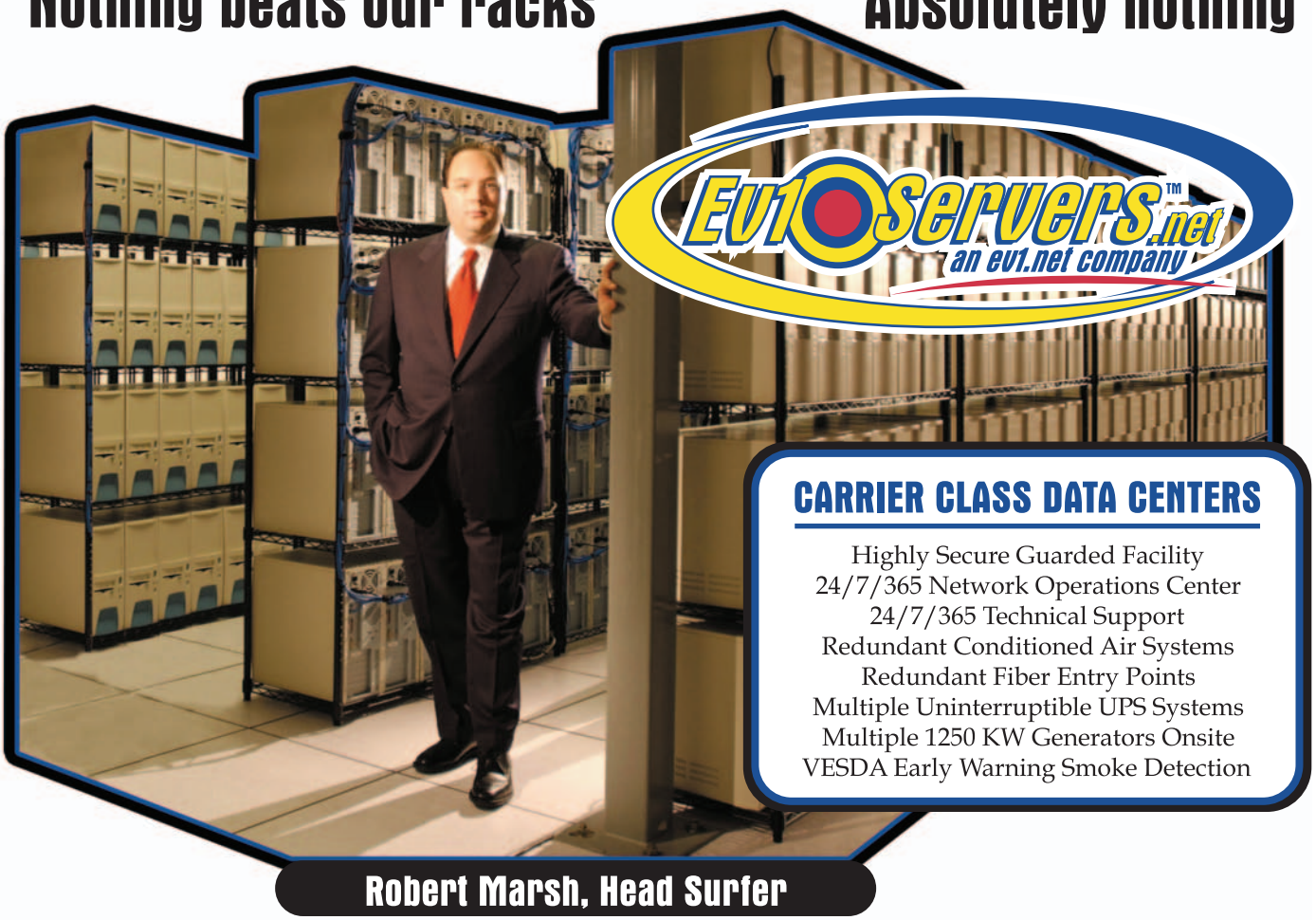
Process calling the order status database control:

```
 *       <controlSend name="OrderStatus" method="orderstatusGetJNDIName"
/>
```

– continued on page 31

Nothing beats our racks

Absolutely nothing



EV1Servers.net™
an ev1.net company

CARRIER CLASS DATA CENTERS

Highly Secure Guarded Facility
24/7/365 Network Operations Center
24/7/365 Technical Support
Redundant Conditioned Air Systems
Redundant Fiber Entry Points
Multiple Uninterruptible UPS Systems
Multiple 1250 KW Generators Onsite
VESDA Early Warning Smoke Detection

Robert Marsh, Head Surfer

START YOUR OWN WEB HOSTING BUSINESS TODAY!

from **$299*** *Instant Activation!*

Dedicated Server

Dual Xeon 2.4 GHz
2 GB RAM • 2 x 73 GB SCSI HD
Remote Console • Remote Reboot
2000 GB Monthly Transfer Included

Over 20,000 Servers!

1-800-504-SURF | ev1servers.net

PLESK7 RELOADED
Preferred Control Panel

IP Compliant. Price subject to change. Quantities Limited.
*Per month. Set-Up fees apply. See web site for complete details.

Measuring the Value of Software Infrastructure

WHAT DO YOU GET FOR YOUR LICENSE FEE?

By Peter Holditch

As I write, the noise level that continues to be generated around open source application servers and their claims to be coming into the world of enterprise computing continues. In my view, the main reason why the noise travels so far and seems so loud has nothing to do with the reality of the situation and everything to do with the media's love of controversy. That said, however, to take a position it is necessary to understand why you stand where you stand. Given that, and since my feet don't seem to want to walk me anywhere, I had better start charting my position… break out the sextant and get ready for some enterprise cartography!

So, What Is WebLogic?

A quick peek at my laptop shows me that it's a 32.3MB JAR file with supporting JARs that bring it up to about 73MB of Java. So, presumably to work out the license cost you divide 73MB by the per-CPU cost and you have the value of the product. Sounds like a rough-and-ready measure perhaps, but not totally crazy. So, I download my favorite open source alternative and I see a load of JAR files, totaling about 66MB. Since the license cost is effectively zero and there is broadly the same amount of "product" there, what are you waiting for? The choice is clear!

Clearly, a static analysis of this type done on any software product will not yield a very meaningful result. Open source is clearly a good mechanism of generating code bulk, and it is obviously cheaper to take a handout from an altruistic developer than to buy the fruits of an employed development team.

There must be something wrong with this logic… Otherwise, why does BEA Systems, who do nothing more than sell infrastructure licenses, have a market cap of over $3 billion, and why (despite all the noisy protestations to the contrary) are open source application servers not rampaging across the mission-critical environments of the world?

There is clearly intrinsic value to software beyond the lines of code that constitute its makeup.

So, is it support? BEA offers 24x7 mission-critical support to its customers, providing them with the assurance that if one of their critical systems fails, there are technicians on hand around the clock to diagnose problems, whether they lie inside or outside of the middleware layer, and to repair them if they lie within. Clearly, this is important – it cannot be cost effective for an organization such as a bank to employ enough people with enough in-depth technical skill to diagnose arbitrary problems on a 24x7 basis. Having the neck of the product author available to choke where necessary (and having the assurance that the vendor's engineers will be online to fix problems as needed) is a necessary part of the effective running of a technology-enabled business (i.e., any business at all). Here pure open source software presents a problem – development is done on an ad hoc basis by a community of enthusiasts. They are clearly not going to be able or willing to stop their world when a user of their software hits a problem. As a result, an organization whose developers have used an open source infrastructure must ensure that it has the internal resources to provide this level of support. Not only is this not cost effective (as noted above), but skilled systems developers are pretty thin on the ground anyway – as evidenced by the move to simplify Enterprise Java programming over the last few years in the face of Microsoft's low-skill entry point alternative – and it is only the most highly skilled developers who will be up to this kind of task. Better to underpin your business with a business relationship than to become dependent on a set of high-cost specialist individuals who, at the very least,

Author Bio:
Peter Holditch joined BEA as a consultant in the Northern European Professional Services organization in September 1996. He now works as a presales architect in the UK. Peter has a degree in electronic and computer engineering from the University of Birmingham.

Contact:
peter.holditch@bea.com

BEA's Workshop can be even more amazing for the phone.

One Application.
Web. Voice. It's Easy.

AppDev™ VXML for BEA's WebLogic™ Workshop

Delivering Web applications to phone users is as easy as clicking a mouse.

Download 30 Day Free Trial
www.sandcherry.com

For additional information:

SandCherry, Inc.
1715 38th Street
Boulder, CO 80301

+1 (720) 562-4500 Phone
+1 (866) 383-4500 Toll Free
+1 (720) 562-4501 Fax

Info@sandcherry.com
www.sandcherry.com

SandCherry

will live in perpetual fear of falling under the proverbial bus…

Live in Perpetual Fear of the Proverbial Bus…

That all said, however, software support is a recurring charge, separate from the license fee, so while its availability and underpinning by a technology-specialist business provides a very good set of reasons why licensed infrastructure software is good, it cannot be the whole story since, despite the emergence of organizations who undertake to provide production support for open source software infrastructures, there is still no visible migration of mission-critical applications to open source software platforms.

I believe that the answer to this riddle does lie in support, but not in the kind of production support provided under a traditional production support contract. Allow me to take a step back to describe what I mean…

"Software support is a recurring charge"

If you look at the software architecture of a typical production system, you will see many moving parts. Typically these systems consist of a set of business components that receive messages for processing (usually from incoming reliable queues, or from end user–facing applications). The processing will typically consist of applying some business rules (by the way, these are the principal value of the system!) and then updating one or more databases and potentially dispatching some more reliable messages. Usually, these messages represent flows of large amounts of business value (money for a bank, stock for a manufacturer, shipments for a logistics operator). Because of the value of these things, care is needed to ensure that each message is processed only once and that once processed, the results of the processing are not lost and are consistent from wherever they are viewed. The provision of these guarantees is usually the job of the infrastructure (often the transaction manager – I bet you wondered if I was ever going to mention that in this issue?!). After all, as already stated, the value of the application from a business perspective is the business rules – the business will just assume that the technology works.

To take this architecture into the realm of concrete examples, let's imagine that the incoming messages are arriving on an MQ Series queue, that the application data is held in an Oracle database, and that the customer reference information (which can occasionally be updated) is held in Sybase.

Our infrastructure software not only has to work flawlessly within itself, but the interfaces to MQ, Oracle, and Sybase have to work too.

Okay, you say, that's fine. As a developer, I will build and test this application from end to end before I put it into production, and then I will know that it works. What's the difference between an open source application server and WebLogic? The difference, it turns out, is huge…imagine your test plan. It must include all fo the business use cases to make sure they successfully complete. It must include all of the edge cases in the business data to make sure edges and errors are handled gracefully. So, now we have a test plan that will give us an appropriate level of confidence that the application meets our business requirements. We're done, right? Well… Yes, if we are using WebLogic and supported versions of the databases and MQSeries. You know that BEA has tested this from an infrastructure perspective and that it all works, because that's what the documentation says (http://e-docs.bea.com/platform/suppconfigs/configs81/81_over/supported_db.html#1129093).

If you use an infrastructure platform that isn't explicitly certified against your external systems, you need to do this testing yourself. But, you say, didn't we do the testing earlier? Well, no. Not from an infrastructure perspective. Given that we require transactional access to three resources, we need to test that all three can commit the work in the success cases, and roll it back in the failure cases – and we have done that with our application-level testing – but we also need to test that if {MQSeries|Sybase|Oracle} fails to prepare, we roll back the transaction, and that if {Sybase|Oracle|MQSeries} fails after a successful prepare but before the transaction is logged that they are rolled back correctly, and that if {Oracle|Sybase|MQSeries} prepares and then fails after the transaction is logged then they are correctly rolled forward on recovery. That's quite a test matrix. And if you change versions on one of these external systems, you will have to run through at least a subset of these tests again. This is clearly very time-consuming, and that's before you factor in how technically tricky it can be to simulate failures at these various points during the transaction processing life cycle. This is a *lot* of work added onto your project plan, or a *lot* of risk introduced into your deployment, if you simply choose to assume that this stuff will work…and that assumes that the tests all pass. What if they don't? Well, you can resort to whatever developer support you might get from the open source community, but if your problem is specific to your version of Sybase, you'd better hope that whoever understands the code out there has access to that same version too…and that they have time to reproduce and fix your problem. But never mind…you have access to the source, you can fix it yourself! In that case, Good luck!

These problems are notoriously difficult to track down and complex to fix. Is that really what your employers want you to be spending your time on? If you don't believe me, take a look at some of the settings that WebLogic offers to work around xa implementation issues in various database drivers – do you fancy finding and fixing one of these (http://e-docs.bea.com/wls/docs81/jta/thirdpartytx.html#1038034)? Not to mention what happens when you start introducing embellishments like Oracle RAC – how long do you think it would take you to test and debug your way to the conclusions about that documented in the WebLogic Server manual?

This effect ripples forward too – if you do meet problems in the production system, not only will you have access to the engineers, etc., in the way that we talked about before, but they will in turn have access to and experience with the databases and other external resources necessary to reproduce and solve your problem for you.

–continued on page 30

What//: are your Web applications saying to your customers?

diagnosys

Are your Web applications open 24 hours? Get the tips and best practices you need to keep your apps running at peak efficiency 24 hours a day with WebApp News, the free newsletter from the performance experts at H&W.

Sign up today at:
www.hwcs.com/wldj08.asp

You might as well be closed for business if your Web applications aren't running at top speed 24 hours a day. With poor Web performance costing businesses $25 billion a year, you need to find problems, and you need to find them fast.

You need DiagnoSys.

DiagnoSys is the first intelligent performance management solution that analyzes your J2EE and .NET applications from end to end. DiagnoSys starts analyzing quickly, gathers data directly from all application resources, and proactively finds and helps fix the real cause of problems – before they cost your business big time.

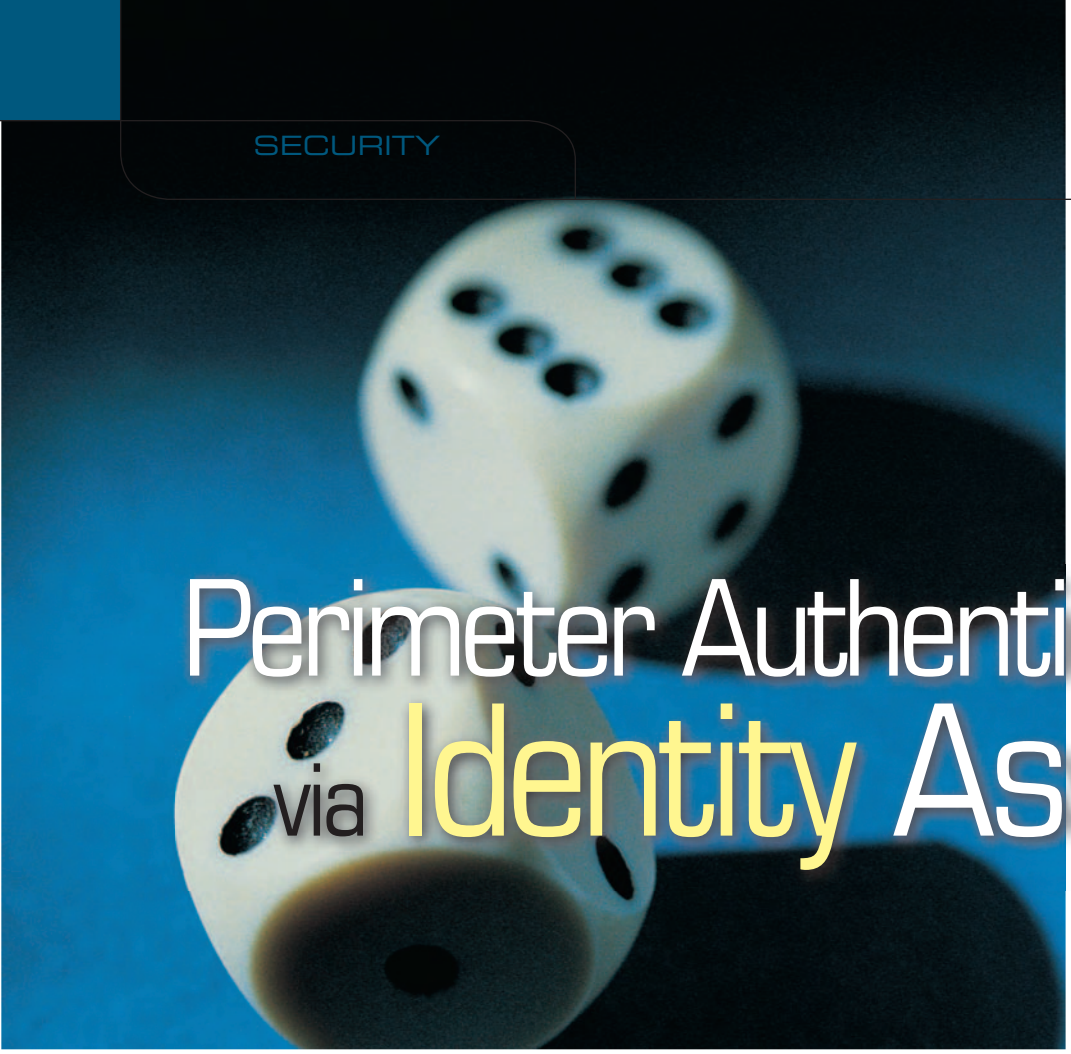
So when it's absolutely essential that your Web applications say "Open 24 hours," trust DiagnoSys.

SILVER ANNIVERSARY 1979 - 2004

H&W

www.hwcs.com | 1.800.338.6692

Bridging the Enterprise Computing Gap For 25 Years



Perimeter Authentication via Identity Assertion

WHERE THE BACK END TRUSTS THE FRONT END

By Sudhir Upadhyay

In some of the typical corporate Web application security deployments, users accessing a protected application are authenticated via enterprise identity/access management products, such as Netegrity's SiteMinder, IBM's WebSEAL, and Oblix's Oblix COREid. The authorization service, however, is delegated to the provider of the application itself, or to the application server.

The application server authorizes a user based on security constraints defined in a Web application's deployment descriptor. However, before the configured security constraints can be applied to provide authorization, the Web containers need to assert that the user is authenticated to begin with. In other words, the front access managers need to provide some information to the back-end Web containers in order to ensure that the Web containers do not attempt to reauthenticate an authenticated user. To accomplish that, it is possible that the front-end access managers may pass the user credentials (username/password) to the back-end servers. This, however, is discouraged for two reasons:

1. In typical deployments, the connectivity between the front-end identity management software and the back-end application servers is not over secured socket layer (SSL), implying that the user credentials would be passed in unencrypted clear text.
2. Even if the connection is made secure, the back-end server will still perform another authentication request for the already authenticated user, using the credentials passed from the front-end access managers, unless it can verify that the request is coming from a trusted source that has already authenticated the user.

This scenario calls for a perimeter authentication, i.e., a user is authenticated only at the front-end access managers. BEA WebLogic Server provides Identity Assertion for achieving identity assertion. This article provides you with a detailed analysis of Identity Assertion implementation in WebLogic Server. You are encouraged to review the product documentation on Identity Assertion providers at http://e-docs.bea.com/wls/docs81/dvspisec/ia.html.

Introduction

From a high-level view, Identity Assertion is implemented by passing tokens from the front-end access managers to the back-end application serv-

Author Bio:
Sudhir Upadhyay is a principal consultant with BEA Professional Services where he helps customers design and implement enterprise J2EE solutions. He is a BEA Certified WebLogic developer and Sun Certified Java developer.

Contact:
sudhir.upadhyay@bea.com

ers. An Assertion is fundamentally a declaration of one or more facts (statements) about the subject, e.g., a user. A token is primarily a piece of information passed from one system to another that can be used to assert the identity of a particular user. Combined, the tokens and the assertion form the basis for Identity Assertion. In essence, an Identity Assertion provider is a specific form of authentication that allows users or system processes to assert their identities using tokens.

Web Application Security

Before moving to the implementation details in Identity Assertion, let's recap the elements of the Web application security model. The servlet specification provides guidelines for implementing Web application security by the servlet containers. The two forms of security prescribed by the specifications are declarative and programmatic. When the Web application is secured declaratively, the security is configured outside the application via deployment descriptors (web.xml). The authentication method, or the way in which a user is prompted for login, is specified by the <login-config> element in the Web descriptor file, web.xml. If the login-config is present and contains a value other than NONE, the user must be authenticated before he or she can access any resource that is constrained by a <security-constraint>.

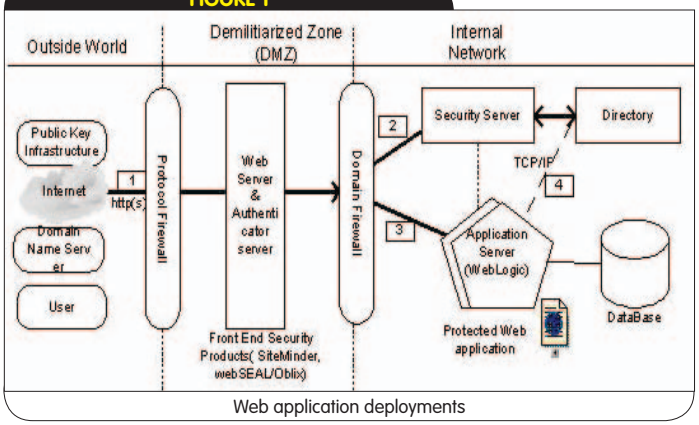
When a user tries to access a protected Web resource, the Web container activates the authentication mechanism that has been configured for that resource in the deployment descriptor (web.xml) between <login-config> elements within <auth-method> tags, like this:

```
<login-config>
  <auth-method>BASIC</auth-method>
</login-config>
```

The following are the valid authentication methods for Web application security.

- ***NONE:*** The user will not be prompted for authentication.
- ***BASIC:*** The Web Server will prompt the user for username/password and authenticate against the information provided. In this form of authentication, a user cannot customize the form that collects the username/password.
- ***FORM:*** In this case, you can customize the login screen and error pages that are presented to the end user by an HTTP browser.

FIGURE 1



Web application deployments

- ***CLIENT-CERT:*** This is a more secure method of authentication than either basic or form based. It uses HTTP over SSL, in which the server and, optionally, the client authenticate one another with Public Key Certificates. Secure Sockets Layer (SSL) provides data encryption, server authentication, message integrity, and optional client authentication for a TCP/IP connection. As you will see later, it is this form of authentication that WebLogic Server uses to perform Identity Assertion.

When the Web application is secured programmatically, the servlet containers implement certain methods of the HttpServletRequest interface to authenticate/authorize users attempting to access a protected resource. The servlet containers may provide additional methods for programmatic security over and above what is mentioned in the specifications. In WebLogic Server, one such class providing the additional methods is weblogic.servlet.security.ServletAuthentication. ServletAuthentication allows both form-based authentication and programmatic authentication in servlets. It performs the authentication call through the Security Realm and sets the user information for the session. The weak() methods are used for password authentication and the strong() methods are used for certificate-based authentication. Later, we will use methods from this class to allow a Web application user to be authenticated via multiple authentication mechanisms for the same Web application.

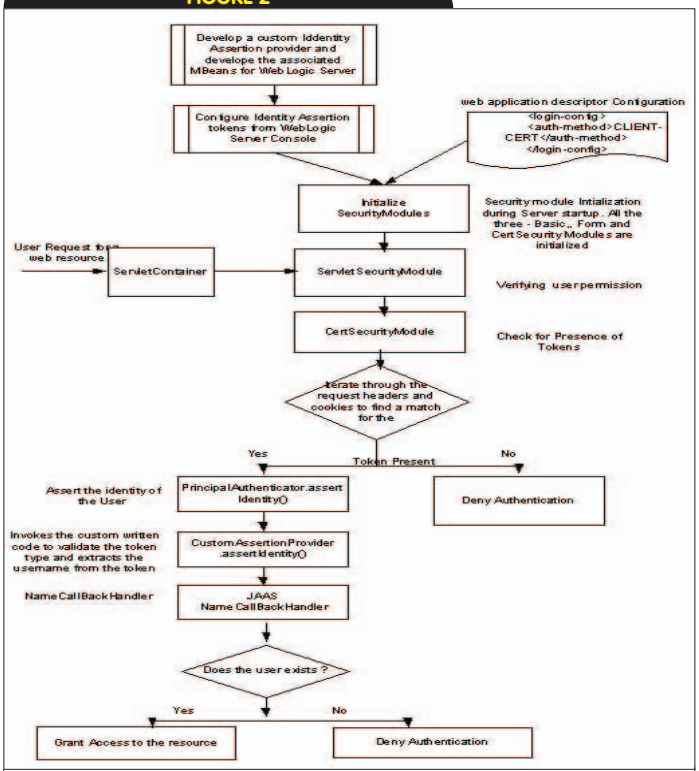
Web Application Deployment

Figure 1 represents the frequently used deployment model for Web applications. As depicted in the diagram, when a user attempts to access a protected Web application, the front-end access manager authenticates the users, and routes the authenticated user to the back-end application server. If the Web application has security constraints defined in its deployment descriptor and the roles/attributed are stored in a directory server, the servlet container invokes security APIs to retrieve user/group information from the underlying directory.

Let's walk through the sequence of steps necessary before a valid user can access the protected resource, as presented in the deployment model.

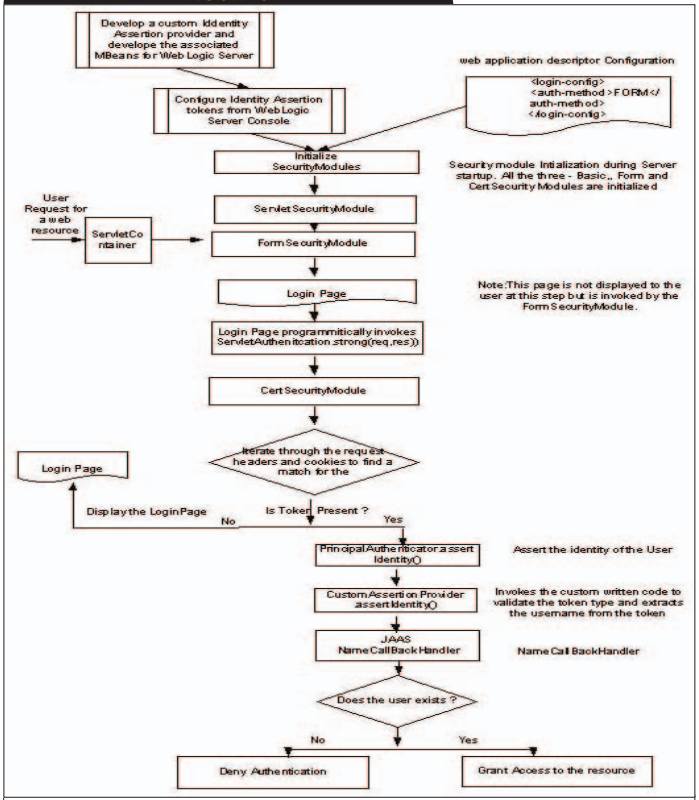
1. User requests a protected Web resource.
2. The front Web server/authenticator server authenticates the user.
3. If the user is authenticated, the request is then forwarded to the back-end application server.
4. However, because the authorization rules/security constraints are configured in the Web application descriptor, the servlet container needs to retrieve the user/group information from the underlying directory store.
5. For the application server to associate the incoming subject with attributes from the underlying store, it needs to ensure that a user is an authenticated user.
6. To enforce the above security requirement, the application server should either reauthenticate the user or assume that the request coming from the front-end access manager is that of an authenticated user.
7. However, making an additional reauthentication call from the application server is not recommended due to its performance

FIGURE 2



Identity Assertion sequence

FIGURE 3



Sequence flow when using multiple authentication methods

overhead. To avoid additional reauthentication calls, the other option for the application server is to assume that the request coming from the front-end access managers is from an authenticated user and bypass the authentication step.

Identity Assertion or perimeter authentication provides a mechanism for the application server to assert that the request coming from the front end is from an authenticated user and thereby avoids reauthentication. This is accomplished by configuring a token that can be passed with the HTTP header request from the front-end access managers to the back-end application server. The following sections explain this in detail.

Token Overview

A *token* is primarily a secret code passed between the two parties. In essence, a token is the assertion or a statement about a principal as asserted (trusted) by an asserting party. When the receiving party receives a token, it performs a set of operations to validate the incoming token. If the token is validated, it is assumed that the request is coming from a trusted source and no additional authentication is done by the receiving party. In a typical Web environment, the token is passed via HTTP header or via cookie.

In its current implementation, WebLogic Server's default Identity Assertion provider provides support for the following tokens: X.509, CSI.PrincipalName, CSI.ITTAnonymous, and CSI.X509CertChain CSI.DistinguishedName. "Supporting" token types essentially mean that the Identity Assertion provider's runtime class (that is, the IdentityAsserter SSPI implementation) can validate the token type in its assertIdentity method.

Since the above tokens may not meet the need for all the applications, you can easily create a new, custom token for your environment. Creating a custom token requires writing a custom Identity Assertion provider that implements the IdentityAsserter Security Service Provider Interface (SSPI). SSPI in WebLogic Server provides a way to develop custom security providers. The custom token type can be as simple as a piece of string. For example, the following piece of code defined in the Custom Identity Assertion Provider Implementation defines a new token type.

```
public final static String MY_TOKEN_TYPE = "MyCustomIAToken";
```

After a new token type is defined, it can be configured via WebLogic Console as described in the BEA product documentation. Once configured and active, these tokens can then be used by the WebLogic Security framework to assert the identity of the incoming requests.

Sequence Flow

Figure 2 represents the sequence of steps that happen during Identity Assertion process:

1. During the deployment of the Web application, the Web container identifies that the Web application's authentication method is configured as CLIENT-CERT.
2. The Web container creates and initializes a corresponding SecurityModule – in this case it creates CertSecurityModule.
3. When the Web resource is accessed by the client, the server's ServletSecurityManager invokes the CertSecurityModule to check the user permissions to access the resource.

YOUR TESTING SUCKS.

1. Writing test code isn't testing – it's coding.
2. Your developers aren't making testable software.
3. It's impossible to get the business side involved in quality.
4. Your existing testing tools have become shelfware.
5. Your customers have become the primary testers.

LISA from iTKO.

Because everyone should own quality.

iTKO Try a free eval of LISA 2.5 at **www.itko.com**

We understand.

You've probably invested considerable time and money trying to automate your testing. But it just didn't work.

You were sold a tool that let you point and click to test on Day 1, and on Day 2 you were maintaining a test script - inserting bugs into your tests at the same rate as you inserted bugs into the code during development.

The tools were too hard for the QA teams to use, they required intense training and developer knowledge, and the seats were more expensive than a presidential fundraiser.

It's OK. We've seen this scenario at every customer we talk to.

LISA is different.

LISA is a comprehensive, no-code automated testing solution ideal for distributed applications and teams.

Using "inline testing" technology, LISA allows even non-developers to instrument and test web services, J2EE apps, databases and websites.

LISA automates test design, staging and monitoring for unit, regression, load and performance tests. And since LISA is pure Java, she's equally stylish on Linux, OS X, Unix or Win clients.

But don't take this ad's word for it. Try LISA for free and talk to the experts at iTKO. We'll show you how everyone can own quality.

4. The CertSecurityModule then invokes the server's security module to find any tokens present in the request object. CertSecurityModule looks for tokens in the HTTP header or in the cookie that were configured during the Identity Assertion configuration. In essence, it iterates through the header values and attempts to match header values with predefined tokens from the WebLogic console.
5. If there is any valid token present in the request, the SecurityModule then attempts to assert the identity of the user.
6. At this time, the server's security module passes the control to the custom Identity Assertion provider implementation.
7. Custom Identity Assertion provider verifies the validity of the token and extracts the username from it.
8. The custom Identity Assertion provider generates JAAS CallBackHandler. Java Authentication and Authorization Service (JAAS) is a set of APIs that enable services to authenticate and enforce access controls upon users. Refer to http://java.sun.com/products/jaas/ for more details on JAAS.
9. The CallBackHandler verifies that the list of callbacks contains a name call back (else it throws an exception) and appropriately sets the username.
10. The NameCallBack ensures that the user is present in the underlying security realms and grants access to the user based on the defined rules.

Relationship with JAAS

It is important to note that the Identity Assertion does not fall under the JAAS umbrella – i.e., JAAS does not provide any specific guidelines on implementing Identity Assertion. The only correlation between the Identity Assertion and JAAS is that Identity Assertion's implementation returns a javax.security.auth.callback.CallbackHandler (http://java.sun.com/j2se/1.4.2/docs/api/javax/security/auth/callback/CallbackHandler.html). As you may recall, when a JAAS LoginModule needs to communicate with the user – for example, ask the user for a username and password – it does so by invoking a CallBackHandler. The CallBackHandler's handle() method then generates the appropriate CallBack to obtain the requested information. In case of Identity Assertion, the custom security provider verifies that the token passed in the request is valid, and subsequently generates a NameCallBack alone.

Multiple Authentication Mechanism

Earlier in this article, we reviewed the Web application security model. One problem with specifying the authentication method in the login-config is that, at a given time, only one authentication method can be configured for one Web application. In certain cases, however, there is a need to serve users from multiple sources, i.e., the users coming from a trusted source (front-end access managers) containing the defined tokens and the users not coming from the trusted source. This may include users from within the internal network who can access the application server directly. In other words, depending on where the request is coming from, the servlet security implementation may need to generate two different CallBackHandlers. However, there is no provision in the servlet specification that allows for multiple authentication methods. This calls for a programmatic solution that may be specific to Web container providers. As I indicated earlier, WebLogic provides weblogic.servlet.security.ServletAuthentication for programmatic authentication from within Web applications. The ServletAuthentication method provides two important methods for authentication:

- ***weak():*** Returns an int value for AUTHENTICATED or FAILED_AUTHENTICATION after pulling the username and password from the request, authenticating the user, and setting it into the session
- ***strong():*** Strong authentication uses the client-side certificate chain as the credential for authentication against the "WebLogic" (default) realm.

Figure 3 provides the sequence of steps in this configuration. In cases of Identity Assertion, the servlet container invokes ServletAuthentication's strong() method. To accomplish multiple authentication methods (CLIENT-CERT and FORM-based) for the same Web application, the option includes invoking the methods programmatically in the login form. In other words, the login form performs a strong authentication (client-cert) first. If a valid token is present in the request header or in the cookie, the user is given access to the resource. If the strong authentication fails, the user is then prompted for a username and password via the login form. Note that because the BASIC form of authentication does not provide a capability to customize the authentication form and provide any programming capabilities in the form, a programmatic combination of BASIC and CLIENT-CERT authentication methods is not possible.

Alternative Solutions

Identity Assertion is not the only possible solution for this problem. Some of the other solutions include passing a shared secret (or a token) between the two parties. In this case, both the sender and the receiver agree upon a shared secret before hand. If any requests arrive with the correct value of the shared secret, the receiving party can trust that the request is coming from a trusted source and allows access to the underlying resource.

Other solutions, such as IP-based trust, can be used for this as well. In this case, the application checks whether or not the request is coming from a preconfigured, trusted IP address before the request is granted access. This typically would be the IP of the Web server in the DMZ.

Conclusion

This article offered details on how BEA WebLogic Server provides a mechanism to perform perimeter authentication via Identity Assertion. This method is useful in situations when the front-end access managers perform the authentication and multiple back-end servers trust that the requests coming from them are authenticated and therefore do not need additional authentication.

References

- *WebLogic Security Documentation:* http://e-docs.bea.com/wls/docs81/dvspisec/ia.html
- *Sample Custom Security Providers:* http://dev2dev.bea.com/codelibrary/code/security_prov81.jsp
- *Java Authentication and Authorization Service (JAAS):* http://java.sun.com/products/jaas/
- Servlet Specifications: http://jcp.org/aboutjava/communityprocess/finaljsr154/index.html

Want to spend more time developing software and less time supporting it? Spend some time discovering hp OpenView—a suite of software management tools that enable you to build manageability right into the applications and Web services you're designing.

Find out how the leading-edge functionality of hp OpenView can increase your productivity.

http://devresource.hp.com/d2d.htm

hp invent

Strategies for WebLogic Domain Configuration

SCRIPTING OPTIONS, PART 2

By Prakash Malani

In my previous article (**WLDJ**, Vol. 3, issue 8), I gave you a detailed overview of the different strategies available for domain creation and configuration and evaluated manual and templating options. In this article, I employ tools like WLShell, WebLogic Scripting Tool, Silent Scripts, and Ant for domain configuration. These tools leverage simple, high-level scripting languages.

Note: This article relies heavily on the common steps such as Domain Creation, Database Configuration, and Verifying the Domain Configu–ration described in part 1.

Scripting Language

There are many questions regarding the structure and semantics of the scripting language. How do you write comments? What, if any, is the line separator? What data-types does the language support? Does the language have variables and assignments? How do you perform query and navigation? How do you invoke behavior (i.e., methods)? Is there support for branching and looping? Fortunately, many scripting alternatives exist for domain configuration. Let's look at some of them.

WLShell

According to the WLShell Web site, "WLShell is a shell for WebLogic." WLShell uses a very simple scripting language to provide access to JMX Mbeans, and a familiar file and directory navigation model to navigate MBeans.

Installation of WLShell is easy. Download the appropriate installer and execute it. Follow the installer steps and complete the installation.

WLShell scripting language supports single-line comments. The comment character is a hash-mark (#). Everything from # is commented out. There is no special end-of-line character. The echo command is either print or echo.

WLShell uses the familiar concept of directories and files for navigating the WebLogic Server instance where the MBeans correspond to directories and the attributes of MBeans correspond to files. Hence, navigation happens with recognizable commands like cd and ls. The command to create an MBean is md and the command to delete an object is rd. For example, in order to create a connection pool named semJDBCConnectionPool, execute the following command:

```
md /JDBCConnectionPool/semJDBCConnectionPool
```

What if the name contains a forward slash? The forward slash is escaped by using an additional forward slash. For example, to create a data source named jdbc/semJDBCDataSource, execute the following command:

```
md /JDBCTxDataSource/jdbc//semJDBCDataSource
```

Setting the variable attribute of an object is done with the set command. To set a variable of the boolean type, invoke set variable-name true-or-false. For example, to set StdoutDebugEnabled variable to true, invoke the following command:

```
set StdoutDebugEnabled true
```

To set a variable of integer type, invoke set variable-name integer-value. For example, to set StdoutSeverityLevel to 64, invoke the following command:

```
set StdoutSeverityLevel 64
```

To set a variable of string type, invoke set variable-name "string-value." For example, to set URL to jdbc:pointbase:server://localhost:9093/workshop, invoke the following command:

```
set URL "jdbc:pointbase:server://localhost:9093/workshop"
```

Author Bio:
Prakash Malani has extensive experience in architecting, designing, and developing object-oriented software, and has done software development in many application domains, such as entertainment, retail, medicine, communications, and interactive television. He practices and mentors leading technologies such as J2EE, UML, and XML. Prakash has published various articles in industry publications.

Contact:
pmalani@malani.org

Setting variables of primitive types is easy, but how do you set a variable of java.util.Properties type? To set a variable of java.util.Properties type, use the special syntax. Invoke set variable-name (java.util.Properties) "name=value pairs separated by semicolon." For example, to set the properties for the connection pool, use the following:

```
set Properties (java.util.Properties) "user=weblogic;databaseName=jdbc:pointbase:server://localhost:9093/semdb"
```

So far, I have examined how to set variables of types boolean, int, String, and Properties. How do you set a variable of another MBean type? Just invoke set variable-name path-to-the-mbean. For example, to set the connection pool property of the JMS JDBC store, invoke the following command:

```
set ConnectionPool /JDBCConnectionPool/semJDBC-ConnectionPool
```

By executing the ls command, WLShell not only lists attributes that can be get or set, but also operations that can be invoked. How do you execute an operation? Using a special command called invoke. The syntax for the invoke command is: invoke method-name parameter-list-separated-by-space. For example, deploy the connection pool to the server, using the following command:

```
invoke addTarget /Server/$server
```

In the above example, the method name is addTarget. The method takes one parameter, the MBean of the server.

Deploying an application with WLShell is involved. Get a reference to /Deployer–Runtime/DeployerRuntime instance and create an instance to hold the deployment data. Associate the data with the server, and activate the application.

WLShell can be used in the script mode as well as in the interactive mode. In the interactive mode, a WLShell session is created and connected to a running WebLogic Server instance. Changes are made to the WebLogic Server instance by invoking commands in the WLShell session. To launch WLShell in interactive mode, execute the following command:

```
%wlsh
```

In the script mode, a script of WLShell commands is created and executed. The script can be created once and executed any number of times later to set up similar domains. The syntax to execute a script is wlsh –f script-file-name. To create the sample domain using script, execute the following command:

```
%wlsh -f sem_domain_wlshell.txt
```

In the above example, sem_domain_wlshell.txt file contains WLShell statements to configure the domain.

Before executing the WLShell script, configure an empty WebLogic Workshop domain by following the steps in the Domain Creation section using SEMDomain-WLShell as the Configuration Name. Setup the database instance by following the steps in the Database Configuration section. In order to run WLShell, just verify that wlshell/bin directory is available in the PATH environment variable. Now, execute the script:

```
wlsh -f sem_domain_wlshell.txt
```

Verify that the configuration is correct by running the tests as described in the Verifying the Domain Configuration section.

WLShell supports converting an existing config.xml file into WLShell script and commands via c2w command. WLShell supports working in off-line mode and connecting to config.xml instead of the running server. However, the implementation is not feature complete. For example, creation of a connection pool in the off-line mode results in a "feature not implemented" message.

WebLogic Scripting Tool (WLST)

WLST uses Jython as its scripting language. The Jython site states, "Jython is an implementation of high-level, dynamic, object-oriented language Python written in 100% Java, and seamlessly integrates with the Java platform." Instead of reinventing the wheel, WLST makes a very smart decision regarding the scripting infrastructure. WLST leverages the powerful scripting language of Jython.

Here are the installation instructions:

- Download WLST (wlsScripting.zip) file and unzip the file (e.g., c:\WLST). The wlsScripting.zip file contains wlst_doc.pdf file. The wlst_doc.pdf contains all the details about WLST such as installation, configuration, and execution. Refer to the References section for detailed instructions on downloading and installing WLST.

Here are the configuration instructions for WLST:

- Set the weblogic.jar in the CLASSPATH by executing setWLSEnv (e.g., %WLS_HOME%\server\bin\setWLSEnv.cmd).
- Set wlst.jar in the CLASSPATH.
- Set jython.jar in the CLASSPATH.
- Verify that WLST is working by executing java weblogic.WLST. This executes WLST in the interactive mode.

Refer to the wlst_doc.pdf file for detailed instructions on configuring WLST.

WLST can be used in script mode as well as in interactive mode. In the interactive mode, a WLST session is created in offline mode. In this mode, WLST is not connected to a running instance of WebLogic Server. Once WLST is connected to a running WebLogic Server instance changes are made to the WebLogic Server instance by invoking commands in WLST session. In the script mode, a script of WLST commands is created and executed. Just as in the interactive mode, the script contains Jython statements. The script file is a Jython file. The script can be created once and executed any number of times later to setup similar domains.

To invoke WLST in interactive mode, invoke the following command:

```
java weblogic.WLST
```

To invoke WLST in script mode, pass in the name of the script. For example:

```
java weblogic.WLST sem_domain_wlst.py
```

Do not execute the above commands yet. At the end of this section, you will do additional setup and then execute the WLST script.

The following are the quick and basic notes on Python (and Jython). Specifying an end of statement character (;) is optional unless multiple statements are specified on a single line. The comment character is hash mark (#). All the characters from # onwards on a single line are considered part of the comment. The echo command is print. Strings can be passed in either single quotes (') or double-quotes (").

WLST uses the familiar concept of directories and files for navigating the MBeans. Hence, navigation happens with recogniz-

able commands like cd and ls. However, every command is a function. Therefore, the commands look like cd('Servers/cgServer') and ls(). WLST defines a very special variable named cmo for the current managed object. Initially, cmo is assigned to the root. The value of cmo changes as you navigate the MBeans. In file system parlance, the value of cmo is the current working directory. Execute ls('a') to find attributes whose value can be changed. Prefix the name of the attribute with the word set to change the value of the attribute. For example, to change the value of attribute StdoutSeverityLevel to 64, invoke the following method: setStdoutSeverityLevel(64). However, on what object should the setStdoutSeverityLevel(64) method be invoked? Of course, cmo:

```
cmo.setStdoutSeverityLevel(64)
```

The other option is to assign cmo to a variable. The variable can later be used at any time throughout the current session. For example:

```
cgServer = cmo
cgServer.setStdoutSeverityLevel(64)
```

Suppose you want to set StdDebugEnabled value to true. However, neither Python nor Jython supports the boolean type. Fortunately, the integer equivalents, 1 for true and 0 for false, can be used. Therefore, set the value of StdDebugEnabled to true as follows:

```
cgServer.setStdoutDebugEnabled(1)
```

How do you create resources such as JDBC connections pools and JMS queues? Use WLST's built-in create() method. The create() method accepts two parameters. The first parameter is the name of the resource, whereas the second parameter is the type of resource. For example, to create a JDBC connection pool with semJDBCConnectionPool as its name, execute the following command:

```
create('semJDBCConnectionPool','JDBCConnection
Pool')
```

Note: There is also a corresponding delete() method that works similarly.

WLST provides a special method named makePropertiesObject() to create a properties object out of a string of name and value pairs separated via semicolon. Therefore, set the properties of a JDBC connection pool as follows:

```
semCPProperties = makePropertiesObject('user=web
logic;databaseName=jdbc:pointbase:server://loc-
alhost:9093/semdb')
semCP.setProperties(semCPProperties)
```

Deploying the newly created resource, such as the connection pool, to the server is easy. Invoke the addTarget() method of the source as follows:

```
semCP.addTarget(cgServer)
```

Deploying an application is very easy with WLST using the deploy() method. The deploy() method takes three parameters. The first parameter is the name of the application. The second parameter is the path to the location of the EAR file. The third parameter is the target or the server name. The following is the example of the deploy() method:

```
deploy('SEMApp','c:/Automate/SEMApp/SEMApp.
ear','cgServer')
```

Before executing the WLST script, configure an empty WebLogic Workshop domain by following the steps in the Domain Creation section using SEMDomain-WLST as the Configuration Name. Setup the database instance by following the steps in the Database Configuration section. Now, execute the script:

```
java weblogic.WLST sem_domain_wlst.py
```

Restart the WebLogic Server instance and verify that the configuration is correct by running the tests as described in the Verifying the Domain Configuration section.

Is it possible to reverse engineer an existing domain into a WLST script? Absolutely. Using the configuretoscript() command, the contents of an existing domain are converted into a WLST script file. Use this feature to quickly incorporate WLST into your current process.

Silent Scripts

A silent script consists of statements that are executed by the configuration wizard in silent mode. The silent script uses a very simple language. The statements are terminated by semicolons (;). The language supports Java-style comments. Double forward slash (//) is used to comment to the end of the line whereas /* … */ is used to comment sections. The language has no concept of data-types; everything is a string enclosed in double quotes. For example, to set boolean or integer types just enclose the value in double quotes, as shown below:

```
set cgServer.StdoutDebugEnabled "true";
set cgServer.StdoutSeverityLevel "64";
```

The simple types can be converted to their string equivalents, but what about objects? In the case of objects the name of the object is used. For example, to set the error destination for a queue:

```
set aq.ErrorDestination "SEMAppWeb.queue.Async-
Dispatcher_error";
```

There is no support for branching and looping. There is no navigation. The structure of the silent script file is flat. The language does support variables. A variable can be assigned upon either creating a new object or finding a preexisting object. For example, a variable assignment on finding a preexisting object:

```
find Server "cgServer" as cgServer;
```

And, a variable assignment on creating a new object:

```
create JDBCConnectionPool "semJDBCConnection-
Pool" as semCP;
```

The script uses special syntax to deploy resources to their targets, i.e., calling the addTarget() method of an MBean. The special syntax is assign <resource type> "<resource name>" to Target "<server name>";. For example, to deploy semJDBCConnectionPool to cgServer use the following syntax:

```
assign JDBCConnectionPool "semJDBCConnection-
Pool" to Target "cgServer";
```

The silent script enables domain creation very nicely. Therefore, you can either create the domain using the Configuration Wizard as described in the Domain Configuration section or use the silent script. I will illustrate domain creation with a silent script. The sem_domain_silent_create.txt script illustrates domain creation. The silent script reads Basic WebLogic Workshop Domain template and creates a domain. But, how do

Post-launch is NOT the time to be verifying web applications.

The wild blue yonder of operational monitoring and management is extremely unforgiving. Which means that going live with the monitoring software you used in development is a great way to go dead—quickly! You simply can't support operations if your staff is drowning in details provided by development profiling tools and debuggers. **Let NetIQ cover your apps...with AppManager.**

AppManager—the industry's easiest-to-use Systems Management suite—is a proven management system for monitoring J2EE application servers, databases, operating systems and even end-user response time. NetIQ's AppManager monitors ALL application components—not just your server. **NetIQ. Nobody does UNIX better. Nobody.**

Visit us at www.netiq.com/solutions/web to learn how we can help you address the challenges of your operational monitoring and management.

netiq
Work Smarter.

you invoke the script? The syntax to invoke the silent script is:

```
> %BEA_HOME%/weblogic81/common/bin/config.cmd
-mode=silent -silent_script=full path to script
file -log=full path to the log file
```

Now, create the domain using the sem_domain_silent_create.txt script.

The –log flag is optional, but highly recommended. When there are errors, the messages logged to the console are very terse, but the complete exception trace is logged to the log file. The log file is immensely important in debugging silent configuration scripts.

Please note that in sem_domain_silent_create.txt file, if the WebLogic user is not assigned a password, the script will fail. Also, newly created users by default are assigned to group Administrators. Therefore, upon creating a user system, the user system belongs to group Administrators. If the script tries to write a domain to a location that already exists, the script fails. In order to replace a preexisting domain, set the OverwriteDomain variable to true.

Next, set up the database as described in the Database Configuration section. Now, shut down the server if the server is running. Configure the server by executing the sem_domain_silent_config.txt script. Be aware that the order of setting driverName and URL is important. Otherwise, the creation of the connection pool fails!

Note the update domain; command at the end of sem_domain_silent_config.txt script. This command is very important. Without it, changes made to the domain will not be committed, and the domain will not be changed!

How do you deploy the application while configuring the domain using the silent script (Note: the server is not running)? Create a custom template that contains the application. The custom template is a JAR file that contains the EAR file, the relevant section of config.xml that will be merged into current config.xml, and some metadata in the template-info.xml file. In the silent script, add the template using the add Template … command. Adding at least one template when updating the domain is necessary, otherwise the changes made to the domain are not persisted! A common technique, if you do not have your own template, is to use the DefaultWebApp template.

Verify that the configuration is correct by running the tests described in the Verifying the Domain Configuration section.

Special Ant Tasks

Apache Ant is an invaluable tool for building and deploying applications. An Ant file is an XML file that contains many targets. The targets invoke tasks. WebLogic ships special tasks such as wlconfig and wldeploy. The wlconfig Ant task helps with domain configuration. The wldeploy Ant task helps with application deployment.

WebLogic Server installation also includes Ant. Ant is automatically set up when setting up the environment by executing the setWLSEnv (bat or sh) script. The taskdefs for wlconfig and wldeploy are not required if this version of Ant is used.

The language of the Ant file is XML. Therefore, language specifics like comments are XML comments. Other features, like the echo are Ant-specific.

The wlconfig task contains nested elements such as query and create. The query element is used to find preexisting MBeans, whereas the create element is used to create new MBeans. Both query and create support get and set elements to get and set MBean attributes. The syntax for the set attribute is: <set attribute="MbeanAttributeName" value="NewValue"/>. All attribute values, regardless of type, are enclosed in double quotes and passed in as string. Consider setting the MBean attribute StdoutDebugEnabled value of type boolean to true.

```
<set attribute="StdoutDebugEnabled"
value="true"/>
```

Similarly, consider setting the MBean attribute StdoutSeverityLevel value of type integer to 64.

```
<set attribute="StdoutSeverityLevel" val-
ue="64"/>
```

Once an object is queried, the object can optionally be set as an Ant property that is available throughout the file. For example, the server is queried and set into cgServer property. Later, in the build file, the cgServer property is utilized to target resources to the server.

Note: The wlconfig does not currently support the invoke command. Therefore, for example, targets are set as attributes instead of invoking the addTarget operation.

The wldeploy Ant task is used to deploy and undeploy the application. The wldeploy task takes actions such as deploy, undeploy, redeploy, cancel, start, stop, and distribute.

In order to execute the included Ant file, first create the domain by following the steps described in the Domain Creation section, giving a meaningful name like SEMDomain-Ant as the Configuration Name.

Note: Since the name of the domain is used in the Ant file, either use SEMDomain-ANT as the domain name or change the Ant file to match your domain name. Start the WebLogic server instance and use the steps described in the Database Configuration to configure the database. Launch a new command or shell window and execute the %WLS_HOME%/server/bin/setWLSEnv (cmd or sh) file to set up the environment. Execute the Ant using the following command:

```
ant -v -f sem_domain_ant_config.xml
```

Run the tests as described in Verifying the Domain Configuration section.

The verbose (-v) option to the Ant command displays the MBean commands issued to the server. The information is very valuable in debugging and troubleshooting.

An Ant file encourages very modular development of the automated script. Each individual resource is created and configured using a separate target. Using Ant dependencies, the all target calls the individual Ant targets and does the complete configuration. Ant encourages the creation of a clean target. The order of dependencies in the clean target is opposite that of the all target.

Conclusion: Choices, Choices, and More Choices

This article described domain configuration using WLShell, WLST, Silent Scripts, and Ant tasks. These tools employ simple and powerful high-level scripting languages. Whereas WLShell uses a custom scripting language, WLST leverages Jython. Once a domain configuration script is created, setting up another identical domain is a simple matter of executing the script. Scripting domain configuration is the way to go for automating domain configuration.

References

- *To discuss the article and ask questions start here:* www.bartssandbox.com. Free membership is required.
- *WLShell:* www.wlshell.net/
- *Download WLST from BEA dev2dev code library:* http://dev2dev.bea.com/codelibrary/code/wlst.jsp

THE WORLD'S LEADING INDEPENDENT WEBLOGIC DEVELOPER RESOURCE

Helping you enable intercompany collaboration on a global scale

- Product Reviews
- Case Studies
- Tips, Tricks and More!

SAVE $69.89 OFF REGULAR NEWSSTAND PRICE

OFFER SUBJECT TO CHANGE WITHOUT NOTICE

***Only $49.99 for 2 years (12 issues)**

Now in More Than 5,000 Bookstores Worldwide

Go Online & Subscribe Today! www.WLDJ.com

SYS-CON.COM/WEBLOGIC/

SYS-CON Media, the world's leading publisher of *i*-technology magazines for developers, software architects, and e-commerce professionals, brings you the most comprehensive coverage of WebLogic.

SYS-CON MEDIA

Distributing Tasks in a Clustered Application Using JMS

MUCH MORE THAN ASYNCHRONOUS DATA TRANSFER

By John-Axel Stråhlman

Decoupling and delaying processing in a request-driven environment is one of the key strategies in creating a robust and scalable distributed application. Many services rely on clustering alone to ensure scalability, but they frequently run into trouble when newfound requirements keep application complexity growing.

Although server clustering is an essential technology that facilitates scalability, it can be rather inefficient when all processing is done synchronously. Throughput can be increased, but responsiveness is a tougher nut to crack.

In this article, I discuss asynchronous processing and illustrate how clever task management can increase the performance, availability, scalability, and manageability of your application. We will create a generic task distribution framework that can send any task to either one or every server in your cluster, in a highly configurable fashion. Our framework will implement the well-known Command pattern by using polymorphism and the Java Message Service (JMS).

Author Bio:
John-Axel Stråhlman is the founder and CEO of Sanda Interactive Ltd (www.stc-interactive.com), a software consulting company based in Espoo, Finland. He is a distributed systems specialist and has been working as a consultant for his clients' projects for more than five years.

Contact:
john-axel.strahlman@stc-interactive.com

What Does Decoupling Mean in Practice?

When a server receives a client request, it usually needs to perform several individual tasks before a response can be returned. Decoupling means that instead of performing all tasks at once, some are instead queued and processed asynchronously. Because queuing is normally a low-cost operation, the synchronous request will finish quicker.

And the Benefits?

Processing tasks sequentially and in parallel is generally more efficient than processing them randomly (whenever clients happen to make requests). The positive impact is greater than what is immediately apparent. In theory, decoupling can increase performance in the following areas:

- ***Robustness:*** Increases because requests will rely on fewer processes that can fail
- ***Responsiveness:*** Partial post processing of requests decreases the time between receiving a request and returning a response
- ***Scalability:*** All decoupled processes can grow in complexity without the threat of decreased responsiveness
- ***Availability:*** Failures can be handled without the client ever knowing that something went wrong

Automatic retries are easily configured for situa-

tions where subsystems are unavailable.

Naturally, the difference between theory and practice will vary from application to application. However, it is clear that almost every implementation will have at least some of the aforementioned benefits.

Sounds Great, But Are There Any Pitfalls?

As with most good things, there are some caveats. One of the most severe is that you might actually find yourself decreasing availability if you don't ensure that you have enough hardware to clear busy processing queues. Queues grow very rapidly if more asynchronous requests come in than your system is able to process. Care has to be taken in the design, and automated monitoring of queues is certainly advisable. Another obvious problem is that most processes in request-driven environments aren't very good candidates for decoupling. In fact, most of the processing might be required to return a response. Sometimes it will require some out-of-the-box thinking and maybe even a change in the way you serve your clients.

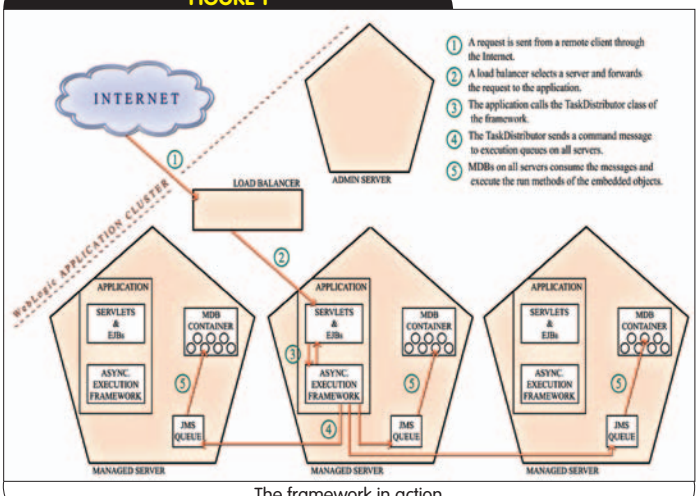
Which Processes Can We Decouple?

From a purely technical perspective, nearly all processes can be decoupled. For instance, you can decouple an ordering transaction by queuing a list of purchased items together with customer details – the asynchronous process will take care of the rest. The downside is that you can't include any processing details in the response. Thus, meticulous prevalidation of data is important to ensure that nothing will go wrong.

An increasingly popular implementation is to queue requests immediately and then keep polling the server to know when a response can be retrieved. Although this approach is actually synchronous in nature and won't improve request processing times, it has the psychological benefit that a progress bar can be displayed during polling.

In addition to decoupling integral business logic (which can be a formidable challenge), less central processes such as logging and sending e-mail are very good candidates to consider. There is no reason to have a client wait for such tasks to complete when performance is of the essence. E-mail in particular is a very good candidate for decoupling. Let's have a closer look.

FIGURE 1



The framework in action

Case Study: Asynchronous E-mail

Sending e-mail the traditional way (as part of a synchronous request) poses some problems. First of all, connecting to an e-mail server requires a network round-trip and it might be slow, especially if the server is very busy. An overloaded e-mail server can even make a service that relies on e-mail temporarily unavailable.

XA Transaction Support

Another easily overlooked issue is that e-mail servers are commonly nontransactional by nature. This can cause inconsistent notifications when transactions are rolled back – a message can't be cancelled after it has been queued. Fortunately, JMS supports transactions and can fix this problem by delaying the delivery of a message until its underlying transaction is committed.

Take into account that when accessing the database and transaction-aware JMS, you will need to use XA and two-phase commit (2PC) transactions. It is possible to emulate XA with non-XA resources, but you might end up with inconsistent data. Enabling XA is a configuration issue and usually requires no changes in code. See the WebLogic documentation for details.

Sending E-mail Through JMS

To use JMS for sending e-mail, we need to have JMS components configured (e.g., JMS server, JMS queue, and connection factory). We also need to write a Message Driven Bean (MDB) to perform the actual sending of our letters. When we want to send an e-mail from within our code, we create a JMS message that contains the properties and content of our letter. After that, we send it to our processing queue.

That's a lot of work! Fortunately, BEA WebLogic JMS provides us with all we need to create a framework for decoupling almost any process.

A Framework for Asynchronous Execution

It's time to roll up our sleeves and look at some code. We will create a framework that will enable the execution of any piece of code asynchronously on either one server or all servers in a cluster. The implementation does require some effort, but, once the framework is in place, asynchronous execution is as easy as pie.

The idea is to write classes that contain one public method with runnable code and another for initializing parameters – possibly a constructor. Encapsulated within JMS object messages, instances of these prewritten classes (command messages) will be sent to JMS queues configured on your servers. At that point, consumers will pick them up and execute them asynchronously (see Figure 1).

Let's look at all the pieces of this framework one by one:

1. ***JMS queues:*** A JMS queue for receiving command messages should be configured on every server. Error queues should also be configured for storing repeatedly failing messages.
2. ***JMS connection factories:*** To allow runtime selection of transactional behavior, two connection factories should be configured: one with XA enabled and another without.
3. ***Command object interface (CommandMessage):*** This is a simple Java interface that all command objects need to implement. It extends the java.io.Serializable interface, which is required for our commands to be embeddable in JMS object messages. Now, because we want to run our commands without actually knowing their exact types, we also implement the java.lang.Runnable

interface and later simply convert them to Runnable objects and execute their run methods. We run the code without knowing exactly what we are running. It's polymorphism at its best!

4. ***Command executor (CommandExecutionManager):*** We will use an MDB for command processing. Instance pooling circumvents recurring JMS initialization, which makes MDBs very powerful message listeners and perfectly suited for this task. Writing the bean class doesn't require much effort; we only need to write a couple of lines in the onMessage method (see Listing 1).

This casts the received message to an ObjectMessage, retrieves the embedded command object, and executes its run method. You can configure a retry count by setting the redelivery limit of your queues to a value greater than zero in your config.xml file. Redelivery is triggered by throwing a runtime exception from your command object. Moreover, by configuring a redelivery delay, you can control the retry frequency as well.

A Helper Class for Sending Messages (TaskDistributor)

Technically, this part is not completely necessary; it is possible to do the JMS queuing manually every time. However, that would be tedious at best, and the helper is actually what makes this framework so practical. The helper is a regular Java class with static methods for queuing command messages. You can write separate methods for handling different scenarios, but for conciseness, I chose to write a single method that can handle most situations:

```
static void execute(CommandMessage cm, long delay, boolean runEverywhere,
boolean persisted, boolean enableXA, int priority)
```

This static method has several parameters for precise execution control. Let's go through these individually:

- ***CommandMessage cm:*** A command message instance.
- ***long delay:*** Represents the time at which to deliver the property, which is set with the weblogic.jms.extensions.WLMessageProducer class. This way, a command could be executed during the night or at some other convenient time. Accepting a Date object instead could also make sense.
- ***boolean runEverywhere:*** Decides whether the message will be sent for execution to a single, randomly selected server or to all servers in the cluster.
- ***boolean persisted:*** Will choose the delivery mode of the message by using the setDeliveryMode method of the queue sender. Business-critical messages should always be persisted so they aren't lost in case of a server crash. However, persistence always entails a performance penalty, which should be taken into account.
- ***boolean enableXA:*** Will choose whether the method will use an XA-enabled JMS connection factory. When set to true, the queuing will take part in an underlying transaction (if one exists), and the message won't be queued before the transaction is committed.
- ***int priority:*** Decides the JMS priority of the message. The setJMSPriority method of the javax.jms.Message class will be called with the given value prior to sending. The valid range is 0–9. Assigning different priorities to command messages may seem like overkill for most applications, but I've included the option here for completeness.

The implementation of the TaskDistributor helper class should be tailored to your specific execution needs. An example would be too long to include in this article, but you can download one from the *WLDJ* Web site at www.sys-con.com/wldj/sourcec.cfm. Several additional parameters could be added to control the execution with even more precision, but, on the other hand, you might be content with fewer options.

Hello Asynchronous Execution!

Once the framework is in place, we can start to implement our command messages. Let's look at a simple example. First, we need to create a class that represents our command message (see Listing 2). To invoke the execution, we use our TaskDistributor class (see Listing 3).

When the execute method in the example is called, an ObjectMessage (with JMS priority set to 4) containing an instance of the DistributedLogger class will be delivered after a one-second delay to all servers in the cluster. Consequently, the logger will print a string to stdout on all servers. With the framework in place, asynchronous execution becomes remarkably accessible and hassle free. Node-to-node communication is easier than ever before.

Container-Managed Task Distribution

We could create an analogous service by using pooled threads and virtual in-memory queues to process asynchronous requests. However, it is strongly recommended to let the application server take care of all thread management.

Furthermore, because JMS provides us with a very elegant and flexible solution, there is no reason not to let our server handle the intricacies of the process. In fact, we could call this methodology Container-Managed Task Distribution.

What About Performance Issues?

BEA WebLogic can handle a heavy message load, and performance is usually not an issue. Nevertheless, when producing a very large number of commands for processing, the use of nonpersistent messages and pipelining is strongly recommended. In addition, message flow control can provide alleviation for situations in which temporary peaks in service utilization rates cause message processing to consume too many resources.

Parallel Processing

A tremendous benefit of using MDBs is that they do parallel processing of messages automatically. You can fine-tune the amount of expended processing resources by limiting the amount of pooled consumer beans.

WebLogic offers numerous valuable JMS extensions and configuration options—many of which can be used in different implementations of task distribution. Great care should be taken when choosing and optimizing JMS parameters for paging, redelivery, persistence, and throttling (flow control).

JMS is a very sophisticated service and careful study of its features is likely to pay off. For more information on improving performance, see the WebLogic JMS performance guide.

Summary

We have discussed decoupling and asynchronous messaging. As a rule of thumb, we can say that a server processing asynchronous

requests is performing more efficiently than one exclusively processing synchronous requests. Although decoupling may not always be easy, or even feasible, it can be a very powerful mechanism when implemented thoughtfully. Not only do we get several performance-related benefits, we are also able to design our applications with more flexibility.

BEA WebLogic JMS is far more than simply a service for asynchronous data transfer. In addition to being remarkably configurable, it offers many useful features, such as automated redelivery, persistence of messages, schedulability, XA support, throttling, transient paging, and redirection of recurrently failing messages. By taking advantage of this enormous versatility, we can create a powerful and extensible framework to handle almost any situation in which asynchronous processing is needed.

References

- Gamma, Erich; Helm, Richard; Johnson, Ralph; and Vlissides, John. (1994). *Design Patterns: Elements of Reusable Object-Oriented Software*. Addison-Wesley.
- Hohpe, G. and Woolf, B. (2004). *Enterprise Integration Patterns: Designing, Building, and Deploying Messaging Solutions*. Addison-Wesley.
- Yochem, Angela; Carlson, David; and Stephens, Tad. (2004). *J2EE Applications and BEA WebLogic Server*. Prentice-Hall.
- *BEA WebLogic JMS Performance Guide:* http://dev2dev.bea.com/productswlserver/whitepapers/WL_JMS_Perform_GD.jsp
- *Programming WebLogic 8.1 JMS:* http://e-docs.bea.com/wls/docs81/jms/index.html

Listing 1: Command processor MDB

```
public void onMessage(Message msg) {

  ObjectMessage om =
    (ObjectMessage) msg;

  Runnable command =
    (Runnable) om.getObject();

  command.run();
}
```

Listing 2: Example of a command class

```
public class DistributedLogger
  implements CommandMessage {

  private String text = null;

  public void setText(String text) {
    this.text = text;
  }

  public void run() {
    System.out.println(text);
  }
}
```

Listing 3: Framework sample usage

```
DistributedLogger logger =
  new DistributedLogger();

String text =
  "Hello asynchronous execution!"

logger.setText(text);
TaskDistributor.execute(
  logger, //Command instance
  1000,   //delay
  true,   //runEverywhere
  false,  //persisted
  false,  //enableXA
  4);     //delay.
```

Measuring the Value of Software Infrastructure

– continued from page 12

For any organization to invest in this level of testing of an open source framework (as opposed to just agreeing to handle production issues) would be a significant commitment of time and effort, and therefore money. And what would be the payback? If this fictional organization spent its life testing and applying fixes to the open source product in open source, who would pay for it? Everyone would be grateful and benefit from their endeavors, but how would they show their gratitude? Or maybe this organization would keep the fixes to itself and charge end users who wanted the additional assurances its fix packs brought. Oh, but isn't that a software license?!

It is worth reflecting that all the open source application servers I am aware of do not ship with fully functional transaction managers. I suspect that the kinds of issues raised in this discussion might account for that – transaction managers, like all other software, are after all only code…

Conclusion

So to conclude, open source is an excellent idea for generating code – lots of the world runs on open source – every time you google your search runs on Linux, a good proportion of the J2EE Web app world on Struts, much of the Java world on Log4j, XMLBeans, Xerces, etc., an increasing number of applications on open source databases. But all of these open source projects have two things in common – first, they interface to a bounded number of stable external systems: Linux to the hardware and BIOS (how often does that change?!), Struts to the stable and mature servlet API, and the rest simply to the Java system itself. Second, with the exception of Linux and database engines (both implementations of well-understood systems, whose requirements have been defined for over a decade and so are very stable), they are developer-oriented toolkits, which do not lie in the critical path of transaction processing. Application infrastructure has to interface to a relatively unbounded set of external systems, queues, databases, clusters, and so on, and by definition is of most value when it is managing the transactional critical path.

For these reasons, I think open source application servers will continue to exist to host low-volume, low-risk systems, but when the stakes are high, their licensed cousins will continue enabling operations staff and business owners to sleep at night, knowing that they were designed from the outset to expect and manage the unexpected.

Create a Real-World Business Process Model, Part 4

– continued from page 8

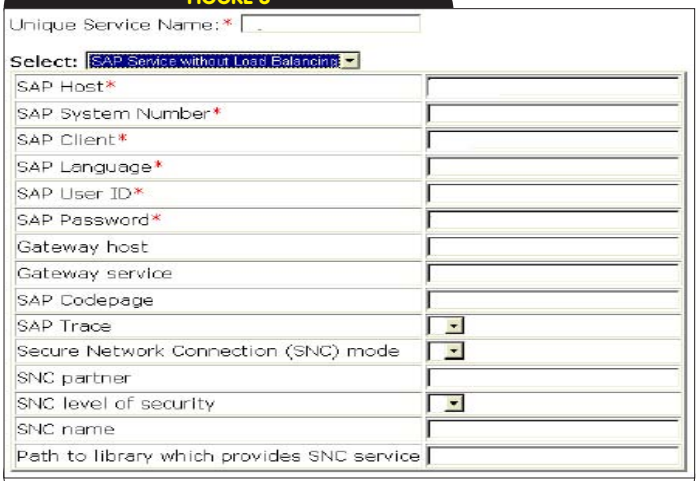
Second decision point to find out if order is changeable:

```
*       <decision name="Is order changeable?">
*         <if name="Yes" conditionMethod="condition"/>
*         <default name="No"/>
*       </decision>
*     </if>
*     <default name="No"/>
*   </decision>
```

Business process writes the file through file control:

```
*   <controlSend name="write" method="changeorderFileWrite"/>
* </process>::
```

FIGURE 3



SAP example without load balancing

Summary

We have seen how to write the change order file through a file control to a directory. This XML file can be used to create a change order in any of your order management systems. I have also shown you how to use an SAP adapter to connect to SAP. We looked at how to use BEA Application Explorer to connect to SAP to create request and response schemas. We saw how the Application View Console uses these schemas and creates an Application View. This Application View can be exposed to WebLogic Workshop to create an Application View Control. The Application View Control can be used in the business process to create a change order in SAP.

In the next and final article, we will look at converting this JPD process into a WSBPEL-specified process and how WLI accomplishes it. This process will be examined to see how it is executed. We will look at tools to monitor the process and see how HP and BEA are partnering to monitor the process in WLI.

References

- *BEA WebLogic Workshop Help:* http://e-docs.bea.com/workshop/docs81/doc/en/core/index.html

Are You Ready to Run with the IT Run Book?

A FIRM GRASP OF EVERYTHING BRINGS BENEFITS

By Raman Sud

Author Bio:
Raman Sud is the vice president of engineering for mValent, developer of the Infrastructure Automation Suite. He has 20 years of experience delivering mission-critical software for enterprises and telecommunication service providers, leveraging distributed development, and building integrated teams in the U.S. and India.

Contact:
rsud@mvalent.com

Any developer or IT organization can attest to the fact that flawless application development does not ensure flawless deployment when the time comes to roll the application out across a WebLogic application server. Successful deployment requires that application configurations in the IT infrastructure stack be correctly dialed for the application to run properly.

Unfortunately for IT organizations, proper application configurations are often lost in translation (or fall victim to human error) between development and production. The result can be slow deployment or application downtime – both of which are mortal sins in the world of critical business applications. Fortunately for businesses, this problem has not gone unnoticed, and an emerging class of technology centralizes and automates the creation of these application configuration files.

In any WebLogic development organization, developers typically use e-mail, text docs, or verbal communication to set forth their recommendations on what configuration parameters the QA team should use to adequately support the latest binary dump in a testing environment. For example, the developer might have the QA team check to ensure that ShrinkingEnabled is set to "true" under the JDBC connection pool, and that the number of connection pool threads is set to 50. Sometimes the info is very straightforward (e.g., "Here's the URL for the DB connection"); sometimes it takes a little research (e.g., "Here's the transactional data source JNDIName"). In most cases, this is not a one-sided exchange. Getting the QA build support folks set up with the best testing environment takes some time. Also, some of the information, such as the Server and SSL listen port settings, is dependent on their environment. But, in the end, the settings are defined, the binaries are checked into the build process, and the developer is essentially removed from the configuration aspects of getting his or her application running properly (see Figure 1).

On the production side of the IT house, configuration settings appear in a very different light. In a typical production environment in which IT managers are busy keeping 1,000 or so mission-critical applications up and running all the time, 99.999% availability is top priority. Unlike the development organization, the folks in IT are closely involved with ongoing maintenance and upkeep of the IT infrastructure stack (Web servers, app servers, databases,

etc.). They have a deep concern for the platforms supporting the J2EE applications in production, because their jobs are on the line if these platforms fail. Moreover, they are hard-pressed to manage the complexities associated with all the underlying configuration files, both within WebLogic and residing across all the other assets in the application infrastructure stack.

In the high-stakes world of application performance and availability, the functioning of a particular application takes a back seat to that which is required to keep that application up and running at maximum efficiency.

It's no secret that the production people will do pretty much anything to stop an application from crashing, or to get a failed application back on line when it goes down. In these cases (although the developer is usually the first one pulled in), the problem is most often the underlying configurations within the supporting IT infrastructure. It's usually not the code itself that causes the crash. The tendency is for lots of blame to be handed out, without much happening to ensure that more configuration-related issues don't crop up in the future.

At a high level, development and IT organizations both want the same thing: to move new applications into production as quickly and smoothly as possible. However, their approaches are totally different. For IT infrastructure team leaders, making ongoing adjustments to low-level parameters around an application that they know little about is just a fact of life. Developers, on the other hand, are relying on coding expertise and industry know-how to put today's leading technologies to good use. They are in the business of creating and delivering industry-leading applications. Developers aren't delving into the intricacies of the various IT assets supporting the applications. In their creative world, new capabilities and competitive advantage are key. Finding the best way to set up the surrounding assets in support of these applications is IT's business.

FIGURE 1

Server	
Name	portalServer
AcceptBacklog	50
AdministrationPort	1
HttpdEnabled	true
ListenAddress	
ListenPort	7001
ReliableDeliveryPolicy	RMDefaultPolicy
TransactionLogFilePrefix	./logs/
TunnelingClientPingSecs	45
TunnelingClientTimeoutSecs	40
DefaultProtocol	t3
DefaultSecureProtocol	t3s
NativeIOEnabled	true
StdoutDebugEnabled	false
StdoutSeverityLevel	32
Log	
Name	portalServer
FileMinSize	20000
RotationType	bySize
KernelDebug	
Name	portalServer
ServerDebug	
Name	portalServer
ServerStart	
Name	portalServer
WebServer	
Name	portalServer
AuthCookieEnabled	true
LogFileName	./logs/access.log
LoggingEnabled	true
COM	
Name	portalServer
ExecuteQueue	
Name	portalRenderQueue
ThreadCount	5
KernelDebug	
Name	portalServer
JTAMigratableTarget	
Name	portalServer
UserPreferredServer	portalServer
SSL	
Name	portalServer
Enabled	true
IdentityAndTrustLocations	KeyStores
ListenPort	7002
ExecuteQueue	
Name	default
ThreadCount	15

Configuration parameters

Is it enough to simply see an application through QA to full GA release, and then let the staging and production folks fend for themselves? Or is this a problem? Should IT and development cooperate more closely on ironing out infrastructure requirements? One giant insurance company, which asked not to be named in this article, thinks they should. Unlike most organizations, this company's IT department maintains control over infrastructure setup across the entire application life cycle, including development, QA, performance testing, staging, and production. This company's developers don't control their own testing sandboxes. If the application they are working on requires a change to the JDBC provider, the developer opens a ticket with the IT department.

This example sounds a bit extreme, but it stems from a desire to capture and log everything required to build what's known as the Application Run Book. To solve the problem of Run Books that don't accurately reflect the production environment, the IT team is working far upstream in the development environment to gather the full set of configuration data that will eventually be used in support of the application running in production. In the case of this company, getting an accurate gauge of the application infrastructure means beginning to gather data in support of the Run Book before the new development project is ever launched.

A second example is a major financial services company. The company needed a more durable tool for IT to use when defining disaster recovery information and troubleshooting application issues in production. The solution they identified meant that all the teams involved in building and delivering a new J2EE application needed to engage very early on and define a set of configuration data that would input directly into their build tracking solution. To this end, they implemented a process by which every dev build checked into PVCS must be accompanied by an electronic set of infrastructure configuration data. This data had to encompass the entire infrastructure stack, including WebLogic. It also had to include Apache, their Oracle database, all the way down to the port settings that would eventually be required on the firewall.

If these companies are any indication, in coming years developers will gradually be required to have input into the production Run Book. This will likely focus on WebLogic configuration information. However, as seen here, developers will also need to scope peripheral technologies they do not currently test or use. Perhaps they will even help define best practices for the IT team to rely on when their application arrives in the production environment.

Of course it's not likely that developers will be defining actual configuration parameters in support of their application after the QA testing cycle is complete. But, in their drive to move Web applications out to production faster, and keep them constantly up and running afterwards, companies are definitely looking for new technologies to facilitate the promotion of an application. It could well turn out that developers have full knowledge of, and perhaps even control over, associated infrastructure parameters throughout the application life cycle. To achieve this, companies will certainly consider the benefit of automation in managing this information. Using automation, whole environments, with their associated low-level configuration settings, can be captured and redeployed. This can be done with minimal involvement from the IT infrastructure team, in a fraction of the time it currently takes to manually configure a new environment. Compared with other new technologies appearing now, automation offers the highest potential return to IT infrastructure managers looking to trim costs and improve organizational efficiency.

Automation needs to help the development and QA side of the house give the IT infrastructure team confidence about the underlying configuration data. Specifically, they need to know that the configuration data supplied along a new application for the Staging

queue has been properly vetted and reflects the current realities of production.

For starters, the work of developing the IT Run Book would already be done. This is because IT would be receiving the same information culled directly from a test environment of which they have full knowledge. There would be no more middle-of-the-road approach involving multiple production scenarios. The Run Book would reflect the production environment exactly (see Figure 2).

To arrive at this happy situation, a determination must be made from the outset about how the various teams will share the information that will ultimately feed into the IT Run Book. Throughout the process, the developer should be free to access and comment on the end-to-end QA testing environment. He or she should also be able to provide input into how the various underlying infrastructure assets need to be configured for optimal performance. This is essentially a standards and policy creation exercise. By gathering feedback from development and QA, the IT infrastructure team can collect perceived requirements pertaining to asset configurations and actually watch as these asset requirements are modified along the path to production. In addition, they will effectively be the hand on the rudder, because the core requirement throughout the testing phase is that the QA environment must match production exactly.

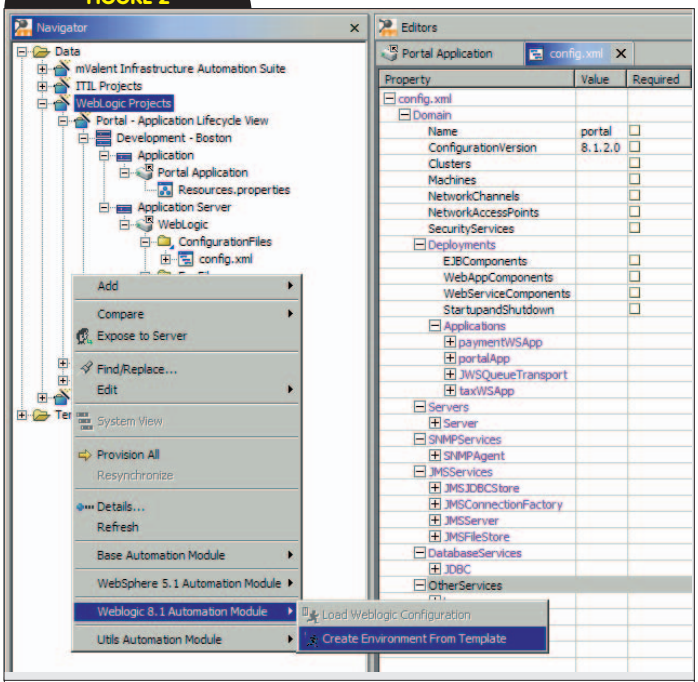
The final stage in this Run Book Automation scenario is for QA to literally push the entire binary set, together with all the "blessed" configurations agreed on by QA and development, out to the UAT phase. Currently, the bottleneck that exists in UAT is largely due to the time it takes to understand what testing took place. It's also created during exploration of how the myriad changes that have happened since the GM milestone will affect the application in the production environment. Now that the Run Book has been made transparent and accessible for all to view, and in some cases modify, this issue goes away. If a quick comparison of the QA test environment can show that the application works well in an environment that mirrors today's production topology, the burden on UAT is virtually eliminated.

A fully verified set of configuration data that mirrors production exactly is possible and can be easily accessed by interested stakeholders across the entire application life cycle. Thus, the notion of a configuration management database (CMDB) must come into play. Moreover, the underlying data must allow for a level of granularity to achieve the type of results described here. That is, namely, the ability to promote an entire application environment, and all the configuration items for its related assets, seamlessly across QA, through to the staging phase and subsequent release to production.

Summary

Technologies that allow companies to achieve faster rollout of new Web applications to production and to automate the development of a meaningful Run Book remain few. However, they are growing in number. Over time, it may turn out that having a firm grasp of the entire spectrum of infrastructure requirements is an important asset to the development organization. At a minimum, such a system would introduce far broader visibility across the organization, in terms of what configurations need to be managed and the preferred attribute values. A powerful upside for development might be fewer invitations to help troubleshoot configuration-related issues in the production environment.

FIGURE 2



Production environment

See what everyone is talking about… ***The Application Server Shoot-Out!***

International Web Services Conference & Expo

Keynotes

Matt Ackley
eBay

Ari Bixhorn
Microsoft

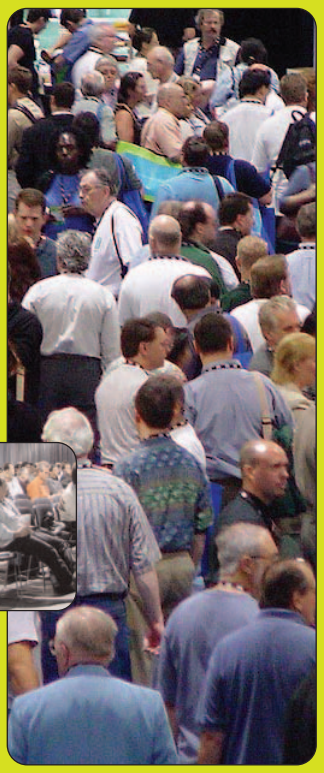
Mike Milinkovich
Eclipse.org

web services conference EDGE & expo

Register by FEBRUARY 11, 2005 SAVE Up To $200

WEB SERVICES EDGE 2005 EAST

Product Teams Delivering SOA, Web Services, Java, .NET, & XML Solutions



New for 2005…

- Cabana Night Developer Exchange
- Application Server Shootout
  facilitated by Burton Group
- Hands On VS and .NET Labs
- Software Giveaway

Free Tutorials presented by critical sites · Novell · Microsoft · Java Community Process

For more information:
visit: www.sys-con.com/edge
call: 201-802-3066
e-mail: jimh@sys-con.com

February 15-17, 2005
John B. Hynes Convention Center
Boston, MA

PRODUCED BY: SYS-CON MEDIA
PRESENTED BY: SYS-CON EVENTS

DIAMOND SPONSOR: Microsoft
GOLD SPONSORS: Novell, SAP
ASSOCIATION SPONSORS: Integration Consortium, OASIS, ineta, eclipse
CORPORATE SPONSOR: Java Community Process
APP.-SERVER SHOOTOUT FACILITATOR: Burton Group
MEDIA SPONSORS: Information Storage+Security Journal, JDJ, WebServices Journal, .NET Journal, Queue, CoDe, XML Journal, wldj, asp.netPRO, WebSphere Journal, ColdFusion Journal, SD Times, MX developer's journal

SPECIAL INSERT: WEB SERVICES EDGE 2005
WWW.SYS-CON.COM/EDGE
FEBRUARY 15-17, 2005
JOHN B. HYNES CONVENTION CENTER • BOSTON, MA

web services conference&expo EDGE

Web Services Edge 2005 East

February 15–17, 2005
John B. Hynes Convention Center
Boston, MA

The Largest *i*-Technology Event of the Year!

Who Should Attend

- Software Developers
- Software Engineers
- Application Developers
- Development Managers
- Technical Directors
- Analysts/Programmers
- IT Managers
- CEOs
- CTOs
- CIOs
- Technical Architects
- Team Leaders
- Software Consultants
- IT Directors
- Project Managers

Features & Attractions

- 3 Days Packed with Education and Training
- Keynotes & Panel Discussions from Industry Leaders
- 50+ Hard-hitting and Informative Seminars
- FREE Web Services Tutorial Presented by Novell
- FREE .NET Tutorial Presented by Microsoft
- FREE Web Services Security Tutorial Presented by Critical Sites
- Cabana Night Developer Exchange Presented by INETA
- Panel Discussions Presented by JCP
- Opening Night Welcome Reception
- Compelling Case Studies & Best Practices
- Hands-On Labs Featuring .NET and Visual Studio Presented by Microsoft Partners
- Featured Product Demonstrations
- Exhibit Floor featuring hundreds of products
- Real-time SYS-CON Radio Interviews

VISIT **www.sys-con.com/edge** FOR TIMES & SCHEDULES

The program, including topics and times, is subject to change. Please refer to www.sys-con.com for all updates.

SPECIAL INSERT: WEB SERVICES EDGE 2005
WWW.SYS-CON.COM/EDGE
FEBRUARY 15-17, 2005
JOHN B. HYNES CONVENTION CENTER • BOSTON, MA

ices Conference & Expo

Keynote Speakers

Tuesday, February 15, 11 a.m.

Matt Ackley
Senior Director, eBay Developers Program

Web Services for eCommerce

eBay, The World's Online Marketplace, has more than 114 million registered users, 10,000 developers, and over 700 live, third-party applications. Four years ago, eBay began allowing third parties to build applications that tap into eBay, and today eBay hosts one of the leading Web services platforms. Through its developer program, eBay enables third parties to create cutting-edge Web services applications that benefit the buyers and sellers on eBay. At present, 40% of eBay's listings come through its API, which handles more than a billion Web services calls a month. Ackley will discuss the rewards and challenges of building and maintaining one of the world's leading Web services platforms, and share insights and practical guidelines for others.

Matt Ackley is senior director of the eBay Developers Program. He supports eBay's vision to be the leading platform for global online commerce, and is chartered with creating a thriving ecosystem between eBay, its community of users, and third-party developers and solution providers. Ackley joined eBay in 2003 as part of eBay's acquisition of FairMarket, which provided technology solutions and services to online marketplaces.

Wednesday, February 16, 11 a.m.

Ari Bixhorn
Director, Web Services Strategies, Microsoft Corporation

Introducing Indigo: The Unified Programming Model for Building Service-Oriented Applications

Indigo is Microsoft's unified programming model for building service-oriented applications on the Windows platform. It enables developers to build secure, reliable, transacted solutions that integrate across platforms and interoperate with existing investments. Indigo combines and extends the capabilities of existing distributed application technologies, including .NET Enterprise Services, System.Messaging, Remoting, ASMX, and WSE to deliver a unified development experience spanning distance, topologies, hosting models, protocols, and security models. This keynote will provide an inside look at Indigo and show you how Indigo will radically simplify the development of distributed, service-oriented applications.

Ari Bixhorn is the director of Web Services Strategy in the Developer and Platform Division at Microsoft Corp. He is responsible for product planning and technical evangelism for Microsoft's Web services offerings, including "Indigo," the code name for a component of the next version of the Windows operating system, code-named Windows "Longhorn." Bixhorn has spent the past five years at Microsoft, driving product management efforts for the Visual Basic and Visual Studio development systems.

Thursday, February 17, 11 a.m.

Mike Milinkovich
Eclipse.org

An Open Development Platform for Web Services

Open source technology runs the Internet. Linux, Apache, PHP and Eclipse are highly successful open source communities that provide the backbone for today's Web applications. All indications point to a continued value proposition for organizations for leveraging open source when developing and deploying SOA-based applications. This keynote will examine the benefits of using open source technologies, the decision-making process used when adopting these solutions and the potential for contributing back to the open source community.

Mike Milinkovich has held key management positions at Oracle, WebGain, The Object People, and Object Technology International Inc. (which subsequently became a wholly owned subsidiary of IBM), assuming responsibility for development, product management, marketing, strategic planning, finance, and business development. Mike earned his MS degree in information and systems sciences and a bachelor of commerce degree from Carleton University in Ottawa, Canada.

VISIT **www.sys-con.com/edge** FOR TIMES & SCHEDULES

The program, including topics and times, is subject to change. Please refer to www.sys-con.com for all updates.

SPECIAL INSERT: WEB SERVICES EDGE 2005
WWW.SYS-CON.COM/EDGE
FEBRUARY 15-17, 2005
JOHN B. HYNES CONVENTION CENTER • BOSTON, MA

SPECIAL INSERT: WEB SERVICES EDGE 2005

WWW.SYS-CON.COM/EDGE

FEBRUARY 15-17, 2005

JOHN B. HYNES CONVENTION CENTER • BOSTON, MA

DAY 1 FEBRUARY 15

	Java	.NET	Web Services
7:30	Registration		
8:00	**FREE Tutorial** – Ashish Larivee, Novell, **Using a Web Services Framework to Build SOA Applications**		
9:00	(J-1) What's New In JDO 2.0	(.NET-1) Intro to SPOT	(WS-1) Ensuring Web Services Interoperability
10:00	(J-2) Using Java Messaging in Real-Time Trading Systems	(.NET-2) An Introduction to SQL Server Reporting Services	(WS-2) Web Services Standards: Going Behind the Mask
11:00	**Opening Keynote** – Matt Ackley, Senior Director, eBay Developer Program, eBay		
12:00	**EXPO OPEN** (12 P.M.–5 P.M.)		
3:00	**Keynote Panel** Presented by JCP – **Web Services and Security** Moderator: Onno Kluyt, Sr Director & Chair, JCP Program, Sun Microsystems		(WS-2B) Solving Complex Business Problems Though SOA
4:00	(J-3) The ROI of a Java-Rich Client	(.NET-3) Go With The Flow – Human Workflow Services in BizTalk 2004 (.NET-3B) Techniques with Visual Basic.NET	(WS-3) The XML Data Challenge
5:00	Opening Night Reception		

DAY 2 FEBRUARY 16

	Java	.NET	Web Services
7:30	Registration		
8:00	**FREE Tutorial –** Thom Robbins, Microsoft – **The Next Generation of Visual Studio** (free with VIP preregistration)		
9:00	(J-4) Web Services End-to-End Security on J2EE: Gaps and Proposed Solutions	(.NET-4) The Microsoft Framework: An Agile Software Development Process for Building Web Service Applications	(WS-4) How To Bulletproof Your Web Services
10:00	(J-5) J2ME and Eclipse	(.NET-5) Web Services Security for Dummies with WSE2	(WS-5) The Role of Policy in Web Services Integration – It's More Than Just Security
11:00	**Keynote** – Ari Bixhorn, Director, Web Services Stategies, Microsoft Corporation		
12:00	**EXPO OPEN** (12 P.M.–4 P.M.)		
3:00	**Application Server Shootout**		
4:00	(J-6) The Impact of JBoss and Mono on the Application Server Market	(.NET-6) Securing Service-Oriented Architecture with Microsoft's WSE 2.0 (.NET-6B) J2EE to .NET Interoperability and App. Integration	(WS-6) B2B Policy Enforcement: The Third Rail of SOA Implementation
5:00	(J-7) Migrating Enterprise Applications Between J2EE Application Servers	(.NET-7) So You THINK You Know What an Object Is… (.NET-7B) Building and Using Advanced ASP.NET Web Controls	(WS-7) Driving SOA Governance
6:00	**Cabana Night –** Hosted by INETA		

DAY 3 FEBRUARY 17

	Java	.NET	Web Services
7:30	Registration		
8:00	**FREE Tutorial** – Patrick Hynds and Duane Laflotte, Critical Sites – **Security, The New Reality** (free with VIP preregistration)		
9:00	(J-8) Design Patterns and Project Organizational Techniques for "Write Once, Debug Everywhere"	(.NET-8) Migrating ASP to ASP.NET	(WS-8) SOA: From Pattern to Production
10:00	(J-9) Using Grid Computing with Web Services and J2EE to Create Internet-based SOAs	(.NET-9) Smart Client Development with the Offline Application Block	(WS-9) High Performance Web Services – Tackling Scalability and Speed (WS-9B) Effective Risk Abatement and Success in a Service-Oriented World
11:00	**Keynote** – Mike Milinkovich, Executive Director, Eclipse Foundation		
12:00	**EXPO OPEN** (12 P.M.–4 P.M.)		
3:00	(J-10) Java Web Services Programming Tips & Tricks	(.NET-10) CLR Internals	(WS-10) So You Want an SOA: Best Practices for Migrating Toward Service Orientation in the Enterprise
4:00	(J-11) JCP Program: How the Java Technology Binary Software Standard is Managed and Evolves	Visit Web site for update	(WS-11) Four Abilities SOA Will Lack Without a Registry

The program, including topics and times, is subject to change. Please refer to www.sys-con.com for all updates.

	WS Security	Case Study
	FREE Tutorial –Aaron Williams, JCP, **Developing Web Services Using Java Technology**	
	(WSS-1) Identity in SOA	(CS-1) Developing E-Commerce Applications with Web Services
	(WSS-2) Securing Web Services with WS-Security	(CS-2) Developing Enterprise Class Web Services
		(CS-3) Service-Oriented Development on NetKernel – Patterns, processes and product to reduce the complexity of IT systems
	(WSS-3) Anatomy of a Web Services Attack	

Novell

FREE Web Services Security Tutorial Presented by Novell

Using a Web Services Framework to Build SOA Applications

Tuesday, Feb. 15
8:00 A.M. – 11:00 A.M.

FREE Tutorial Presented by Java Community Process

Developing Web Services Using Java Technologies

Tuesday, Feb. 15
8:00 A.M. – 11:00 A.M.

Java Community Process

Free Tutorials with VIP Preregistration ONLY!

	WS Security	Case Study
	(WSS-4) Using Mobile Phones as an SSO Authentication Device in SOA Solutions	(CS-4) Orchestrating FORCEnet Engagement Packs with BPEL for Web Services
	(WSS-5) Building Intelligent Enterprises with Novell's Identity-Driven Computing	(CS-5) CPI: A Globally Integrated Problem-Tracking and Resolution System Using Java Web Services
	(WSS-6) XML Content Attacks	(CS-6) The Transformation of SiteRefresh into a Web Services
	(WSS-7) The Interoperability Challenge of Web Services Security Standards	

Microsoft

FREE
.NET Tutorial Presented by Microsoft

The Next Generation of Visual Studio

Thom Robbins

Wednesday, Feb. 16
8:00 A.M. – 11:00 A.M.

Free Tutorials with VIP Preregistration ONLY!

	WS Security	Case Study
	on)	
	(WSS-8) Transitioning Successfully to SOA and Web Services: Building the Infrastructure for SOA Growth	(CS-8) Using SOA and Web Services to Issue Business Licenses in the District of Columbia
in d	Visit Web site for update	(CS-9) Developing Web Services with Eclipse

critical sites

FREE
.NET Tutorial Presented by Critical Sites

Security, The New Reality

Patrick Hynds

Thursday, Feb. 17
8:00 A.M. – 11:00 A.M.

Free Tutorials with VIP Preregistration ONLY!

The program, including topics and times, is subject to change. Please refer to www.sys-con.com for all updates.

SPECIAL INSERT: WEB SERVICES EDGE 2005
WWW.SYS-CON.COM/EDGE
FEBRUARY 15-17, 2005
JOHN B. HYNES CONVENTION CENTER • BOSTON, MA

WEB SERVICES TRACK

WS-1 Ensuring Web Services Interoperability

Chris Ferris, IBM

Tuesday, Febuary 15, 2005 9:00 A.M. – 9:50 A.M.

Despite the open industry standards that underlie Web services, interoperability has been a key challenge for vendors and customers implementing Web services. One reason for this is that the relevant industry standards often permit multiple acceptable implementation alternatives. This presentation will discuss in detail the challenge of Web services interoperability and the role played by the premier industry organization formed to address it, the Web Services Interoperability Organization. In particular, the presentation will cover the critical importance of WS-I profiles to an organization's Web services initiatives, including the manner in which companies can put WS-I profiles immediately to work.

BIO: Chris Ferris is chair of the WS-I Basic Profile Working Group and a senior technical staff member with IBM's Emerging Technology Group. He has been actively engaged in open standards development for XML and Web services since 1999 and is an elected member of the OASIS Technical Advisory Board. Chris is also a coauthor and editor of the WS-Reliable Messaging specification.

WS-2 Web Services Standards: Going Behind the Mask

Glen Daniels, Sonic Software

Tuesday, Febuary 15, 2005 10:00 A.M. – 10:50 A.M.

Web services and service-oriented architectures (SOAs) are emerging as an integral part of the enterprise IT strategy. According to a recent IDC study, Web services – related revenue is expected to triple from $1.1 billion worldwide in 2003, to $3.4 billion in 2004, and $16.6 billion by 2008. As SOAs proliferate and the number of Web services added to them increases, standards will play an increasingly significant role. This session will look at the state of key Web services standards such as WS-Choreography, WS-Reliability and WS-ReliableMessaging, SOAP/MTOM/XOP, WSDL, XPath, XQuery, and WS-Notification as well as related Java standards and open source efforts. It will also look at the organizational impact of standards adoption in the industry.

BIO: Glen Daniels is manager of standards and consortia at Sonic Software and coauthor of *Building Web Services with Java.* He has been working with Web services technologies since their inception in the late '90s, and in addition to developing products and helping to found Apache's Axis project, he has been an active participant in standards bodies such as the W3C, and a member of the SOAPBuilders interoperability group.

WS-2B Solving Complex Business Problems Though SOA

John Daly, netNumina

Tuesday, Febuary 15, 2005 3:00 P.M. – 3:50 P.M.

BIO: John Daly is a recognized leader in service oriented architecture, enterprise wide integration and business intelligence. As Managing Director of netNumina, a leading firm focused on financial service, pharmaceutical and other fortune 100 companies, he is responsible for helping companies solve complex business problems by delivering sophisticated technology solutions that combine proven technologies with cutting edge thinking. John's ability to deliver technological solutions specifically designed for the user experience separates his approach from those of netNumina's competitors.

WS-3 The XML Data Challenge

Jonathan Bruce, DataDirect Technologies

Tuesday, Febuary 15, 2005 4:00 P.M. – 4:50 P.M.

Most businesses store and query data with relational databases but need to use Extensible Markup Language (XML) to exchange and display data on the Web and with vendors and partners. As a result, programmers need to deal with both relational and XML data, often at the same time. Emerging standards such as XQuery, XQJ, and SQL/XML, promise to revolutionize data exchange and the ways applications are developed, deployed, and utilized. Learn the key facts about these standards, including what they mean, when they will be available, and what you, the developer, can do to prepare.

BIO: As the XML Product Manager at DataDirect Technologies, Nancy Vodicka is responsible for DataDirect Connect for SQL/XML, a database-independent SQL/XML implementation that is currently shipping, and DataDirect XQuery, a database-independent XQuery implementation that is currently in development. Nancy has more than 15 years experience in the software industry working with technologies such as XML, Web services, relational databases, and SQL.

WS-4 How To Bulletproof Your Web Services

David McCaws, Parasoft

Wednesday, Febuary 16, 2005 9:00 A.M. – 9:50 A.M.

Web services are gaining industry-wide acceptance and usage and are moving from proof-of-concept deployments to actual usage in mission-critical enterprise applications. Web services range from major services such as storage management and customer relationship management to much more limited services such as furnishing stock quotes or providing weather information. As companies and consumers begin to rely more and more on Web services, the need for developing reliable, high-quality Web services is even stronger. This session will explain issues specific to Web services and will illustrate solid engineering and testing practices required to ensure complete Web service functionality, interoperability, and security. Whether creating Web services from scratch or integrating a legacy back-end server via Web services, the practices and principles outlined in this session will be of great benefit.

BIO: David McCaw has over eight years of experience in helping software development teams improve quality throughout the development process. Over the last three years, he has led the Parasoft Web Services Solutions team, which has developed an industry-leading approach for Web services testing. He has implemented Web service quality solutions for development groups in organizations such as Sabre-Holdings, Yahoo! Overture, and McGraw-Hill. McCaw has an extensive background in the areas of Java and Web service reliability, performance, and security. He is involved with OASIS and WS-I, and is a frequent speaker at industry events.

WS-5 The Role of Policy in Web Services Integration – It's More Than Just Security

Toufic Boubez, Layer 7 Technologies

Wednesday, Febuary 16, 2005 10:00 A.M. – 10:50 A.M.

Too often today the preferences, terms, and conditions describing how a Web service behaves when discovered and invoked is programmed right into the business logic. Hard-coding this behavior logic however introduces cost, complexity, and rigidity into a Web services architecture. A better approach is to abstract a Web services usage "policy" out of code where this metadata can be managed as need be. This session introduces the concept of Web Services Policy and describes how the construct can be used to implement a more customized and versatile Web service infrastructure.

BIO: Toufic Boubez is a well-respected and renowned Web services visionary. Prior to cofounding Layer 7 Technologies, Toufic was the chief Web services architect for IBM's Software Group and drove their early XML and Web services strategies. He is a sought-after presenter and has chaired many XML and Web services conferences. He is an author of many publications and his most recent book is the top-selling *Building Web Services with Java: Making Sense of XML, SOAP, WSDL ,and UDDI.*

WS-6 B2B Policy Enforcement: The Third Rail of SOA Implementation

Alistair Farquharson, Digital Evolution

Wednesday, Febuary 16, 2005 4:00 P.M. – 4:50 P.M.

One of the great benefits of a service-oriented architecture is the ability it gives you to extend programmatic, integration capabilities to business partners. Going beyond simple sharing of data with partners, SOA enables true B2B application integration. At the same time, this capability creates a vexing security policy enforcement dilemma. How can you be sure that a user from a partner organization is actually authorized to integrate with your applications? How can you authenticate that user? Do you even want that headache in the first place? This session will discuss the issues that arise in B2B security policy enforcement and explore several proven approaches

to solving the problem. In particular, it will focus on the emerging technology of XML Virtual Private Networks (XML-VPNs) and their potential to mitigate security policy enforcement issues in B2B SOA implementations.

BIO: Alistair Farquharson is the CTO of Digital Evolution, where he spearheads product development and provides thought leadership to enterprise customers implementing Web services. His skills span many industries and include designing and implementing system architectures, as well as spearheading initiatives such as development/team lead. He is an expert in custom-application development, distributed environments, architecting scalable hardware and software applications and systems, and Web services application development.

WS-7 Driving SOA Governance

Brent Carlson, LogicLibrary

Wednesday, Febuary 16, 2005 5:00 P.M. – 5:50 P.M.

In the past year, Web services and service-oriented architectures (SOAs) have become mainstream because of their ability to provide business agility and flexibility through integration, productivity, and reuse. With SOA enablement on the rise, IT groups must address SOA governance as a means of controlling what and how services located within an SOA are deployed. This session will discuss SOA governance, specifically how an organization can manage and control assets and artifacts located within an enterprise, while ensuring that deployed assets meet an organization's business and technical architectural standards. It will also outline governance best practices such as monitoring the UDDI publish process in order to seamlessly tie together the development and operational views of Web services within the enterprise.

BIO: Brent Carlson drives the development and delivery of LogicLibrary's products. He is a 17-year veteran of IBM, where he served as lead architect for the WebSphere Business Components project and held numerous leadership roles on the "IBM San Francisco Project." He is a member of the Eclipse Board of Stewards and a BEA Regional Director.

WS-8 SOA: From Pattern to Production

David Chappell, Sonic Software

Thursday, Febuary 17, 2005 9:00 A.M. – 9:50 A.M.

Service-oriented architecture (SOA) represents the opportunity to achieve broad-scale interoperability, while providing the flexibility required to continually adapt technology to business requirements. No small feat, particularly when one considers the extent and complexity of today's IT environments. As both a technology concept and IT discipline, the challenge inherent in SOAs is maintaining the right architectural approach. If all services in an SOA are treated as interdependent point-to-point interfaces, then the complexity of implementing and maintaining them in this spaghetti-like architecture becomes enormous. The enterprise service bus (ESB) has emerged as one of the first true SOA product offerings, bringing SOA from pattern to production. ESBs provide a framework for building and deploying an event-driven, enterprise SOA and accommodates the configuration, hosting, and management of integration components as services across the business.

BIO: VP and chief technology evangelist for Sonic Software, Dave Chappell has over 18 years of experience in the software industry covering a broad range of roles including R&D, codeslinger, sales, support, and marketing. He also has extensive experience in distributed computing, including message-oriented middleware, CORBA, COM, and Web application server infrastructure.

WS-9 High Performance Web Services – Tackling Scalability and Speed

Sameer Tyagi, Sun Microsystems

Thursday, Febuary 17, 2005 10:00 A.M. – 10:50 A.M.

Web services facilitate application-to-application integration and interoperability across different platforms. However, critics usually point to an inefficient processing model and bandwidth requirements for developing Web services. This is often cited as a reason why Web services cannot perform and scale well in production environments. This session takes a detailed look at performance and scalability issues around Web services in the real world, as well as strategies that architects and developers can adopt to mitigate such risks in these applications. Some analytical and modeling strategies that enable acceptable application performance will also be covered.

BIO: Sameer Tyagi works as a senior Java architect with Sun Microsystems. He remains focused on architecture, design, and implementation of large-scale enterprise applications with Java technology. His publications include industry periodicals and books on Java and J2EE technologies including *Java Web Services Architecture.*

WS-9B Effective Risk Abatement and Success in a Service-Oriented World

Paul Lipton, Computer Associates

Thursday, Febuary 17, 2005 10:00 A.M. – 10:50 A.M.

IT leaders are hoping to leverage the benefits inherent in Web services and Service-Oriented Architectures (SOA) to enable their businesses to be far more competitive and to find new operational efficiencies. But, can we depend on these new technologies and approaches? Management and security are a common concern today and this session provides the necessary background and perspective on both the business and the technical issues. We will examine important principles and recommendations using real-world examples to illustrate key concepts.

BIO: Paul Lipton is a senior architect in the Web services and application management team at Computer Associates (CA) as well as a Strategist in the Office of the CTO. He has been an architect and developer of enterprise systems for over 20 years, and has worked closely with key CA customers to solve important business challenges through the creation of manageable, mission-critical distributed solutions.

WS-10 So You Want an SOA: Best Practices for Migrating Toward Service Orientation in the Enterprise

Eric Newcomer, IONA

Thursday, Febuary 17, 2005 3:00 P.M. – 3:50 P.M.

Replacing complex, monolithic applications with nimble applications built from exposed services promises increased developer productivity, greater flexibility, and ultimately reduced cost. The adoption of Web services and SOA can also remove a significant level of complexity and integration problems from enterprise application development projects. But, as with any large-scale project, IT departments must have the right plan and the right resources in place to ensure a successful transformation of their computing infrastructure. This article will explore what IT organizations need to know to be successful in their attempts to migrate the enterprise to a service-oriented architecture.

BIO: In the role of chief technology officer at IONA, Eric Newcomer is responsible for IONA's technology roadmap and the direction of IONA's e-business platforms as relates to standards adoption, architecture, and product design.

WS-11 Four Abilities SOA Will Lack Without a Registry

Luc Clement, Systinet

Thursday, Febuary 17, 2005 4:00 P.M. – 4:50 P.M.

A service-oriented architecture (SOA) is the design blueprint for seamless connectivity between business processes and IT infrastructure, enabling innovation and improving productivity. SOA provides the most efficient, standard way to dynamically interoperate with any customer, supplier, product or employee. SOA makes integration intrinsic. Web services are the foundation building blocks of an SOA, and they are already proliferating inside most enterprises. In an SOA, Web services become business services with the ability to perform a particular function or access data dynamically. This presentation will discuss the four abilities that a registry provides for an SOA.

BIO: Luc Clement is director of product marketing, SOA Registry for Systinet. He is also cochair for the UDDI Specification Technical Committee. Formerly Microsoft UDDI Program Manager, Luc is well known in the UDDI community and has been heavily involved with the UDDI specification for several years.

WEB SERVICES SECURITY TRACK

WSS-1 Identity in SOA

Sekhar Sarukkai, Oblix

Tuesday, Febuary 15, 2005 9:00 A.M. – 9:50 A.M.

The mainstreaming of SOAs requires a more general approach to the notion of identities – beyond simply central management of people identities and into the realm of managing applications, devices, and other identities that represent entities that are first-class participants in this application network while also providing this as a pluggable service into the larger enterprise SOA. Enterprises should view identity as a service that is ubiquitously available and is a shared infrastructure service necessary for application networking, rather than as being managed by a server, such as an Authentication or Access server. While it makes architectural sense to consider an Identity service, there are business and related drivers that may force the need to deploy such an architecture.

BIO: Sekhar Sarukkai is currently a technical architect at Oblix. He was the original founder and CTO of Confluent Software, a leading Web services management company, which was acquired by Oblix in 2004. He holds a PhD in computer science from Indiana University.

Attend a FREE* Tutorial

Thursday, February 17, 2005, 8 – 11 a.m.

Using a Web Services Framework to Build SOA Applications

Ashish Larivee

Ashish Larivee, Novell

Service-oriented architecture (SOA) has quickly taken center stage as the primary development style of the next decade and beyond. Businesses of all types are preparing for the SOA revolution that promises consistency of process, reduction in duplicate work, ease of maintenance, service reusability and broad interoperability. The Web Services Framework (WSF) is the foundation that can deliver on the promise of SOA. Come learn about the components of an SOA including the core WSF standards. Attend this free Novell tutorial and learn about the future of SOA-style development, including legacy system enablement, platform interoperability, open source in SOA and building composite applications that leverage SOA services using Novell exteNd. In this session, we will create SOA application logic that orchestrates legacy services, JBoss4 Web services and MS.Net Web services. We will create Web services in Novell exteNd, Eclipse and Visual Studio respectively. We will then orchestrate these Web services and expose a single course, process level interface to public Web service consumers. Finally, we will cre-

Novell

* Free Tutorials with VIP Preregistration ONLY!

WSS-2 Securing Web Services with WS-Security

Dr. Jothy Rosenberg, Service Integrity

Tuesday, Febuary 15, 2005 10:00 A.M. – 10:50 A.M.

An up-to-date, comprehensive, and practical discussion of Web services security, and the first to cover the final release of new standards SAML 1.1 and WS-Security. Comprehensive coverage and practical examples of the industry standards XML Signature and XML Encryption will be presented.

BIO: Dr. Jothy Rosenberg is a serial entrepreneur. He is a founder and CTO of Service Integrity, a company that helps Web service operators see, measure, understand and fully leverage operational and business information flowing across their Web service networks. Prior to this venture, Jothy cofounded GeoTrust, the world's second largest certificate authority.

WSS-3 Anatomy of a Web Services Attack

Mamoon Yunus, Forum Systems

Tuesday, Febuary 15, 2005 4:00 P.M. – 4:50 P.M.

A broad range of new security threats is facing enterprises implementing XML Web services, leaving the enterprises open to financial risks, loss of property, and tarnished reputations. The basic rules of security – authentication, authorization, and auditing – no longer provide adequate security in the new world of straight-through processing paths into mission-critical systems. What's worse, WSDL documents provide a guide book to security exposure. Most attacks on traditional Web-based applications exploit weaknesses in HTML-enabled custom, or packaged, applications. However, hackers and other malicious users are quickly uncovering new techniques at the SOAP/XML data level that bypass HTML and target weaknesses in Web services programming, technology, and architecture. This session will outline the innovative techniques that hackers use to map out the vulnerabilities of an organization's network, and how Web server security must now complement Web services security in order to provide an adequate defense.

BIO: Mamoon Yunus, CTO of Forum Systems, was previously a global systems engineer for webMethods, where he developed business integration strategy and architecture for Global 2000 companies. He is an industry-honored CTO in advanced technological solutions for enterprise customers.

WSS-4 Using Mobile Phones as an SSO Authentication Device in SOA Solutions

Dr. Michael Juntao Yuan, University of Texas

Wednesday, Febuary 16, 2005 9:00 A.M. – 9:50 A.M.

Federated identity management across multiple single-sign-on domains is a major challenge for SOA-based solutions to fully realize its business potential. The traditional username/password combination is often too weak to protect the extremely sensitive single-sign-on credentials. The new-generation mobile phones could be used to identify and authorize users for SOA services. The device-based authentication scheme depends on not only "what you know" but also "what you own." This session will discuss new advances in Java-based mobile devices to interoperate with Sun's Liberty Alliance Services.

BIO: Dr. Michael Juntao Yuan is an author, developer, and software architect for end-to-end mobile software. He is a contributing editor to *JDJ* and a frequent contributor to many developer forums and publications. He is the author of two books. Michael has a PhD from the University of Texas at Austin and teaches information systems at the college level.

WSS-5 Building Intelligent Enterprises with Novell's Identity-Driven Computing

Ashish Larivee, Novell

Wednesday, Febuary 16, 2005 10:00 A.M. – 10:50 A.M.

Companies are now facing complexities dealing with issues such as regulatory compliance and security while still providing for company-wide collaboration between employees, partners, and suppliers. Identity systems are becoming a crucial component of applications, enabling developers to take advantage of a new set of services that know who you are, where you are, what you are trying to do, and can adapt to your changing business needs. Identity-driven computing addresses these problems by applying best practices learned from Novell's leadership in identity management for the management of people to all aspects of an enterprise, including servers, PCs, devices, applications, and even Web services.This presentation will outline identity-driven computing, describe the attributes of an identity-driven application, and discuss steps enterprises can take to make the transition to an identity-driven computing environment.

BIO: With more than nine years of experience in the software industry, Ashish Larivee has designed and developed many enterprise applications across a variety of platforms, including Microsoft, Lotus Notes/Domino, and J2EE. In 1999, Ashish joined SilverStream Software, acquired by Novell in July 2002, and has served in various roles in consulting, development, and technical marketing. In her current role, she helps define the strategy and product direction across Novell's Web Application Development Products.

WSS-6 XML Content Attacks

Girish Juneja, Sarvega, Inc.

Wednesday, Febuary 16, 2005 4:00 P.M. – 4:50 P.M.

This talk defines a new class of threats, XML Content Attacks, and differentiates these threats from more general Web services attacks and XML security-based attacks. These three related but distinct threat areas are explained. The talk covers XML Content Attacks with regard to tree-based parsing exploits related to coercive parsing, node-depth attacks, and DOM. XML grammar validation exploits such as schema poisoning and lax-content models are discussed, and why traditional schema validation cannot ensure content-model consistency. Web services attacks like WSDL scanning and parameter tampering (SQL Injection, SOAP array attack) are discussed – highlighting common mistakes made when applying message-level security (WS-Security).

BIO: Girish Juneja has more than 15 years' experience in the high technology industry with extensive product management, product strategy, engineering management, and technology marketing expertise. He is the cofounder of Sarvega. Since Sarvega's inception, Girish has led the Sarvega engineering and customer services organizations to develop Sarvega's industry-leading core XESOS technology and XML Networking products.

WSS-7 The Interoperability Challenge of Web Services Security Standards

Eve Maler, Sun Microsystems

Wednesday, Febuary 16, 2005 5:00 P.M. – 5:50 P.M.

The Web Services Interoperability Organization chartered its Basic Security Profile Working Group to develop an interoperability profile involving transport layer security, SOAP message layer security, encryption, signatures, and other security considerations. This session will discuss the interoperability challenges presented by current Web services security standards and the work of the WS-I Basic Security Profile. The session will highlight typical Web services security threats and countermeasures and the related design goals, usage conventions, and conformance testing of the soon-to-be-released Basic Security Profile.

BIO: Eve Maler is an XML standards architect at Sun Microsystems, where she coordinates Sun's involvement with Web services security standards such as SAML and the WS-I Basic Security Profile.

WSS-8 Transitioning Successfully to SOA and Web Services: Building the Infrastructure for SOA Growth

Dan Foody, Actional

Thursday, Febuary 17, 2005 9:00 A.M. – 9:50 A.M.

This session will address how to approach service-oriented architecture (SOA) management from a project-based level while still allowing room for future expansion and incremental growth to an enterprise-wide SOA. The session will provide valuable insight into how SOA management can help organizations ease the complexity of moving toward a loosely coupled environment.

BIO: As CTO at Actional, Dan Foody leverages his extensive experience in enterprise systems software toward designing robust and manageable service-oriented architectures. He is an active participant in the Web services standards community, including WS-I and OASIS, where he spearheads Actional's contributions on the OASIS Web Services Distributed Management Committee (WSDM).

RESERVE YOUR HOTEL ROOM NOW! AT THE SHERATON BOSTON HOTEL

THE OFFICIAL CONFERENCE HOTEL OF EDGE 2005 EAST

THE SHERATON BOSTON HOTEL
PRUDENTIAL CENTER
39 DALTON STREET
BOSTON, MA 02199
PHONE (617) 236-2000
FAX (617) 236-1702

Hotel Arrangements Are Easier Than Ever! Web Services Edge 2005 East has secured special group rates, well below regular rates, at one of the finest hotels in Copley Square, The Sheraton Boston Hotel. The hotel is directly next door to the Hynes Convention Center, and is connected via The Prudential Mall. To learn more about hotel savings, or to make your individual or group reservations please contact Meeting & Travel Consultants:
Toll free (866) 549-3985
Direct (732) 549-3985
Fax reservations to (732) 549-3987.
Make online reservations by Jan. 17, 2005
visit: http://www.mtcreservations.com

SPECIAL CONFERENCE RATES

Single	Double
$196	$196

Credit card information is required to guarantee reservations and expedite confirmation. Confirmations will be mailed directly from the hotel, time permitting. All changes and cancellations should be made directly through MTI.

DRIVING DIRECTIONS TO THE SHERATON BOSTON FROM LOGAN INTERNATIONAL AIRPORT

• Follow direction sign from terminals to Sumner Tunnel/Rt. 93 North • Take the Rt. 93 North ramp as you exit the Sumner Tunnel • Exit at Storrow Drive (26 B) • Follow Storrow Drive approximately 2 miles to the Fenway/Kenmore Exit (1st exit after Massachusetts Ave., on left) • Stay left as exiting Storrow Drive going towards Fenway • Continue to 1st set of lights, staying left, going onto Boylston Street • Go through 4 sets of lights on Boylston Street • The Hynes Convention Center's main entrance driveway will be on the right immediately after going through the 4th set of lights.

WEB SERVICES EDGE CONFERENCE PAST SPONSORS & EXHIBITORS

VISIT **www.sys-con.com/edge** FOR TIMES & SCHEDULES

The program, including topics and times, is subject to change. Please refer to www.sys-con.com for all updates.

SPECIAL INSERT: WEB SERVICES EDGE 2005
WWW.SYS-CON.COM/EDGE
FEBRUARY 15-17, 2005
JOHN B. HYNES CONVENTION CENTER • BOSTON, MA

International Web Services Conference & Expo

Register by FEBRUARY 11, 2005 SAVE Up To $200

For Assistance Tel: 201-802-3066 Fax: 201-782-9601

CONFERENCE & EXPO: February 15 – 17, 2005

John B. Hynes Veteran Memorial Convention Center • Boston, MA

THREE WAYS TO REGISTER FOR CONFERENCE

1) On the Web: Credit Cards or "Bill Me" Please make checks payable to SYS-CON Events
2) By Fax: Credit Cards or "Bill Me" 201-782-9601
3) By Mail: 135 Chestnut Ridge Road, Montvale, New Jersey 07645, Attention: Registration

Please note: Registrations are not confirmed until payment is received.

Please complete sections 1, 2, 3 and 4

1 YOUR INFORMATION (Please Print) ☐ Mr. ☐ Ms. ☐ Mrs.

First Name _____ Last Name _____

Title _____

Company _____

Street _____

Mail Stop _____

City _____

State _____ Zip _____ Country _____

Phone _____

Fax _____ E-Mail _____

2 PAYMENT METHOD: (Payment in full due with registration)

☐ Check or Money Order Enclosed (Registration confirmed upon receipt of payment)

Check # _____ Amount of Check $ _____

Charge my ☐ Visa ☐ MasterCard ☐ American Express ☐ Discover

Name on card _____

Card # _____ Exp. Date _____

Signature _____

Billing Address (if different from mailing address)

_____

3 PLEASE INDICATE YOUR CONFERENCE CHOICE:

Total Registration fee $_____

	Before 12/31/04	Before 1/21/05	Before 2/11/05	On Site
☐ **3D: Three Day Conference** Good for all three days of the conference program, including keynotes, all conference sessions.	$1,195.00	$1,395.00	$1,495.00	$1,695.00
☐ **2D: Two Day Conference** Good for two days of the conference program, including keynotes, all conference sessions.	$995.00	$1,195.00	$1,295.00	$1,395.00
☐ **1D: One Day Conference** Good for one day of the conference program, including keynotes, all conference sessions.	$595.00	$695.00	$695.00	$795.00
☐ **VIP PASS** FREE Tutorials with VIP preregistration ☐ FREE - Novell Tutorial (Feb. 15) ☐ FREE - Developing Web Services Using Java Technology (Feb. 15) ☐ FREE - The Next Generation of Visual Studio (Feb. 16) ☐ FREE - Security, The New Reality (Feb. 17)	FREE	FREE	FREE	$75.00
☐ **EO: Expo Hall Only** Includes keynotes, expo presentations, and workshops	FREE	FREE	FREE	$50.00

SPECIAL GROUP RATES

Registering a group of four or more? Please call Jim Hanchrow at 201-802-3066, and receive an additional discount.

4

A. Your Job Title

☐ Senior Software Developer/Engineer/Architect/Analyst
☐ Director of IT/Development/Engineering
☐ Software Developer/Engineer/Architect
☐ Systems Architect/Engineer/Analyst
☐ Software Consultant/Analyst
☐ Project Manager/Team Leader
☐ Business Development
☐ General Manager/Division or Department Head
☐ CTO,CIO, Chief Architect
☐ CEO, President, Owner, Partner/Principal
☐ VP/Technical Director
☐ Network Administration/Manager/Consultant
☐ Web Developer/Programmer
☐ Database Administrator
☐ VP/Sales & Marketing Director /Product Management
☐ Other (please specify)_____

B. Business/Industry

☐ Agriculture/Mining/Oil/Gas
☐ Computer Hardware and Electronics
☐ Construction/Architecture/Engineering
☐ Consulting (non-computer)
☐ Education
☐ Banking/Insurance/Accounting
☐ Government (including Military)
☐ Healthcare/Pharmaceuticals/Biotech/ Biomedical
☐ Internet/Web/e-Commerce
☐ Manufacturing and Process (other than computer related)
☐ Media/Marketing/Advertising
☐ Non-Profit/Trade Association
☐ Real Estate/Legal
☐ Telecommunications
☐ Transportation/Rail/Sea/ Air/Ground
☐ Travel/Hospitality/Entertainment
☐ Utilities
☐ Wholesale/Trade/Distribution/ Retail
☐ Other (please specify) _____

C. Total Number of Employees at Your Location and Entire Organization (check all users that apply):

	Location	Company
10,000 or more	01 ☐	01 ☐
5,000 - 9,999	02 ☐	02 ☐
1,000 - 4,999	03 ☐	03 ☐
500 - 999	04 ☐	04 ☐
100-499	05 ☐	05 ☐
100 or less	06 ☐	06 ☐

D. Please indicate the value of software products and services that you recommend, buy, specify or approve over the course of one year:

☐ $10 million or more
☐ $1 million - $9.9 million
☐ $500,000 - $999,999
☐ $100,000 - $499,999
☐ $10,000 - $99,999
☐ Less than $10,000
☐ Don't know

☐ $10 billion or more
☐ $1 billion - $9.9 billion
☐ $100 million - $999 million
☐ $10 million - $99.9 million
☐ $1 million - $9.9 million
☐ Less than $1 million
☐ Don't know

E. What is your company's gross annual revenue?
F. Do you recommend, specify, evaluate, approve or purchase mainframe products or services for your organization?
01 ☐ Yes 02 ☐ No

G. Which of the following products, services, and/or technologies do you currently approve, specify or recommend the purchase of?

☐ Application Servers
☐ Web Servers
☐ Server Side Hardware
☐ Client Side Hardware
☐ Wireless Device Hardware
☐ Databases
☐ Java IDEs
☐ Class Libraries
☐ Software Testing Tools
☐ Web Testing Tools
☐ Modeling Tools
☐ Team Development Tools
☐ Installation Tools
☐ Frameworks
☐ Database Access Tools / JDBC Devices
☐ Application Integration Tools
☐ Enterprise Development Tool Suites
☐ Messaging Tools
☐ Reporting Tools
☐ Debugging Tools
☐ Virtual Machines
☐ Wireless Development Tools
☐ XML Tools
☐ Web Services Development Toolkits
☐ Professional Training Services
☐ Other [Please Specify]

_____

If you require special assistance covered under the Americans with Disabilities Act, please call 201-802-3066 by February 4, 2005.

CANCELLATIONS, SUBSTITUTIONS, REFUNDS

Fax written request to SYS-CON Registration 201-782-9601. Requests for refunds received prior to January 7, 2005 will be honored, less a 10% handling charge; requests received after January 7, 2005, and before February 4, 2005, will be honored less a 20% handling charge. No requests for refunds will be honored after February 4, 2005. Requests for substitutions must be made in writing prior to February 11, 2005. No one under 18 is permitted to attend. No warranties are made regarding the content of sessions or materials.

Speakers, sessions and schedule are subject to change without prior notice.

No solicitation by anyone other than official exhibitors, sponsors or marketing partners is permitted. Such behavior is cause for expulsion without refund.

SPECIAL INSERT: WEB SERVICES EDGE 2005
WWW.SYS-CON.COM/EDGE
FEBRUARY 15-17, 2005
JOHN B. HYNES CONVENTION CENTER • BOSTON, MA

Memory Leak Detection with the JRockit JVM

UNIQUE CAPABILITIES WITH THE RIGHT TOOLS

By Helena Åberg Östlund

What causes a memory leak in a Java program? Shouldn't the garbage collector of the Java virtual machine (JVM) take care of unused memory? Yes, it will, but only objects that are no longer referenced can be garbage collected. A typical memory leak in Java occurs when objects that are no longer needed are still referenced somewhere in the system so that they cannot be garbage collected.

Memory leaks in Java code are a common and difficult problem to solve in large enterprise systems. These leaks are often discovered only after deployment and can be a challenge to reproduce in a testing environment. Why is that? One reason could be that the deployed system handles far larger amounts of data, and it could be that only after weeks of execution it is discovered that the Java heap is slowly increasing. Eventually this will cause the system to run out of memory.

How can we find such a leak? There are a number of tools on the market, but most of them are either based on creating a dump of the Java heap and analyzing it offline and/or based on JVMPI, which makes application execution prohibitively slow. In BEA's JVM JRockit, version 1.4.2_05, there is a technology preview of an interactive memory leak detection tool, the JRockit Memory Leak Detector, which allows usage while the system is running at almost full speed. In this article we will take a look at this experimental tool and some of its possibilities. We will also get a taste of what future versions of the tool will be capable of and go over some common types of memory leaks.

Memory Leak Detector

The Memory Leak Detector is designed for use in production environments and induces very little performance overhead on the system. A lot of the JVM-internal memory leak detection functionality is built directly into the JRockit garbage collector for higher performance. JRockit has a management console to monitor and manage one or more JVMs. The memory leak detection tool is currently built into the JRockit Management Console, but will be available as a stand-alone tool in a JRockit JDK 1.5 release later this year. The idea is that the system will help the developer understand three things: Is there a memory leak; if so, what is leaking; and where is it leaking?

Before we dig into the details, I would like to underscore that this is a low-level tool. For example, it will not tell you in what EJB your leak is. The data is passed directly from the JVM to the Memory Leak Detector and the JVM has no notion of containers, EJBs, or any other J2EE stuff. It will only deal with classes, arrays, and instances. As long as you have access to the source code, this should not pose any problems.

Approach

Let's get started. The first step is to determine if there really is a memory leak in the system. To do this we first create a connection in the management console to the JVM we are interested in, connect, and make sure the system is running and under load. Then we go to the "MemLeak Detector" tab in the management console. Here we need to enable the memory leak detection system in JRockit, which will start a trend analysis of memory growth. Each time there is a garbage collection in the JVM, JRockit sends trend data to the Memory Leak Detector that is presented in a trend analysis table (see Figure 1).

A Case Study

The trend analysis shows the most common object types on the heap (or classes if you prefer) and the rate at which memory for these types is growing. It also gives the current memory usage of the type and the number of instances. The longer the trend analysis runs, the more reliable the trend is. The types are sorted in order of growth rate and

Author Bio:
Helena Åberg Östlund is a senior software engineer in the Java Runtime Products Group at BEA Systems, part of the development team for the JRockit JVM. For the past three years she has been developing tools for JRockit. Prior to that she worked with Java consulting and development in Smalltalk.

Contact:
helena.aberg@bea.com

it is the types with a high growth rate that we are interested in – these are the leaking types. In the example in Figure 1 it is clear that we have three types guilty of leaking memory: DemoLeak$DemoObject, Hashtable$Entry, and Hashtable$Entry[]. I will use this example as a case study in order to explain the functionality of the JRockit Memory Leak Detector.

Where Is It Leaking?

So, now we have detected a memory leak and we also know what types of objects are leaking. In this example, it is not very far-fetched to guess that the hash table entries are holding DemoObjects, since the number of instances of these types is approximately equal. Also, it is logical that the arrays of hash table entries are also growing, as there are more and more hash table entries.

To verify that the hash table entries are holding on to the DemoObjects we can ask to see what types are pointing to DemoObject. To do this we need to freeze the updating of the trend analysis. Then we can select the DemoLeak$DemoObject row in the table and right-click to get a popup menu with an option to "Show types pointing to this type." This will display a table of all types pointing to DemoObject. In this case the table only contains the Hashtable$Entry type. (Aha, our guess was correct!) From that table it is possible to drill down and check what types are pointing to Hashtable$Entry and so on.

Traversing this "points-to chain" backwards (see Figure 2) will reveal that Hashtable$Entry types are pointed to by Hashtable$Entry[] (arrays) and also Hashtable$Entry. If you think the latter is weird, remember that in the case of a hash collision in a hash table, a single bucket stores multiple entries, which are set up as a linked list of hash table entries that is searched sequentially.

Now that we know that one or more hash tables are leaking, we want to get down to the instance level. One way is to find the Hashtable$Entry[] instance(s) that grows, which is probably the largest one. To do this we can right-click on the Hashtable$Entry[] type and ask to "Show largest arrays of this type." This action will display a table of the largest array instances and the memory size in bytes of these arrays. In our example it looks something like Listing 1.

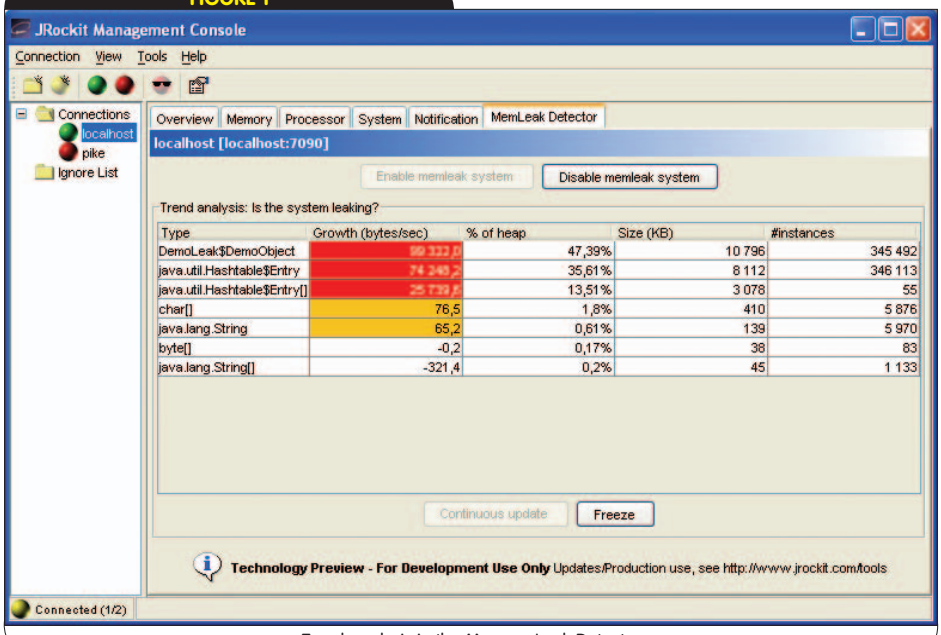
The number in brackets <n> is an object ID that uniquely identifies the object and is used by JRockit's memory leak detection system. What this list tells us is that the array with object ID <10> is clearly a suspect as it consumes way more memory than the others. To be absolutely sure, we can wait for a while and then ask the system once again for the table of largest Hashtable$Entry arrays. We can then verify that the array with ID <10> still consumes the most memory and that it has grown a few hundred bytes.

Let's recap what we've found out so far. The system is leaking memory and there are three types that seem to be leaking – DemoObject, Hashtable$Entry, and Hashtable$Entry[]. There is one single array instance of hash table entries that is growing, so there is obviously only one hash table in the system that is leaking. Which one?

We can find out by asking for points-to information on the largest array instance by right-clicking on it and selecting "Show instances pointing to this array." This will open the window shown in Figure 3. It shows instances or static fields (if any) referring to the Hashtable$Entry[]<10> instance. As we can see, there is a single instance pointing to the array: a hash table with object ID <20>. We have now found the leaking hash table and we can continue our search by asking for points-to information for this hash table. The results are represented visually in the instance graph in Figure 4. The leaking hash table is referred by two DemoThread objects and a number of ObjectMonitors. We can disregard the ObjectMonitors because they are only a result of synchronization and have nothing to do with the application code. DemoThread on the other hand is very interesting, as it is part of the user code.

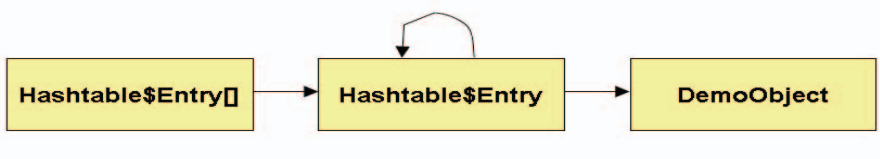
To help us quickly find the correct instance field in the DemoThread that is referring the hash table, we can open an Inspector on one of the DemoThread objects. The Inspector (see figure 5) shows all the fields of the instance and their val-

FIGURE 1



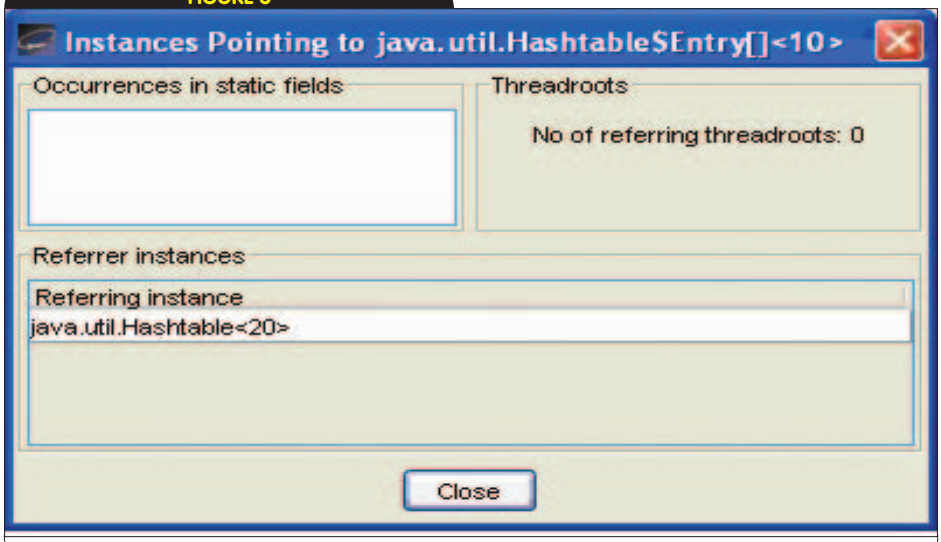
Trend analysis in the Memory Leak Detector.

FIGURE 2



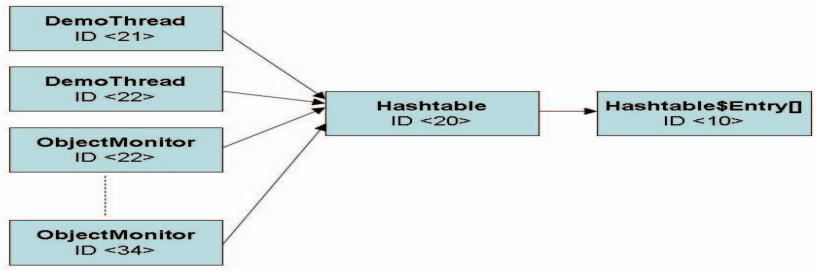
Types pointing to each other

FIGURE 3



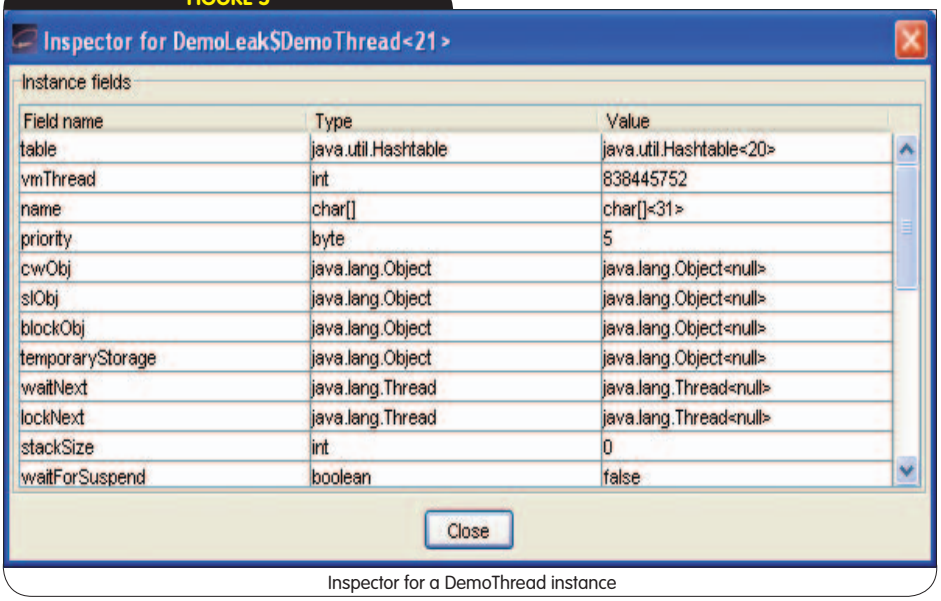
Instance level points-to information

FIGURE 4



Instances pointing to the leaking hash table

FIGURE 5



Inspector for a DemoThread instance

ues. The first field in the list happens to be "table," which points out a Hashtable with ID <20> that was the leaking hash table. This is how far the Memory Leak Detector gets us. Now it is simply a matter of reading the source code of DemoThread (which is an inner class in DemoLeak.java) and seeing what we are actually doing with this "table" field.

Limitations

The program we have been investigating is a simple example chosen for clarity. A real production system is a lot more complex and may need more thorough research.

You should also be aware that there are some limitations with this technology preview of the Memory Leak Detector tool. One issue is that it is now using the Java-based communication protocol of the JRockit Management Console. This means that JRockit needs to create new Java objects to send information over to the management console. This is not an ideal situation, as the system is probably already low on memory in the case of a memory leak.

Another limitation is that when there are large amounts of information to send over, we risk losing the connection to the management console because of timeout problems. This is usually only a problem when requesting a list of all instances of one kind of type pointing to a particular type (which in theory could be hundreds of thousands). The navigation in the user interface also leaves a lot to be desired.

We view this tool as a proof-of-concept and we are eager to get feedback so we can improve future versions of it. Even if it is not perfect, we believe it can still be very useful to people seeking out a memory leak.

Future Work

So, what is planned for the next version of the Memory Leak Detector? First, it will be a stand-alone tool that can be run independently from the management console. It will use a native communication protocol, which should eliminate the out-of-memory risks in the present tool. The major thing, however, will be improvements to the user interface. There will be object and type reference graphs, similar to those in Figures 2 and 4, to visualize the points-to chains and where the user can navigate in a more intuitive way than that of today. It will also be possible to get allocation traces

for where a particular type of object is allocated. This can come in very handy when pinpointing exact locations in the code. The memory leak detection system in JRockit will still be operating in real time with minimal overhead and can be enabled or disabled at any time.

Common Leaks

Memory leaks are difficult to detect and fix, but the good news is that many programmers tend to make the same mistakes. That might not sound good at all, but it means that most memory leaks are quite similar. The set of typical memory leaks is therefore not that large and this will, with experience and the right tools, make them easier to find. Below, I will describe a few of these often seen on the server side.

To put objects in hash tables or hash maps but then forget to remove them when they are no longer needed is a very common error. Our case study is an example of an off-by-one error when removing items from a hash table. The faulty code in DemoThread is shown in Listing 2.

The approach here is to find out which hash table instance is leaking and who is holding onto that instance, as we did in our case study. Hopefully, this will give you enough information to pinpoint the correct hash table in your code.

A similar situation is when there is a complicated data-structure with hash tables inside other hash tables and where one or more of the "inner" tables are leaking. The approach is the same as above, but the reference information will appear more confusing, as a hash table entry can also be pointing to a hash table.

Another common leak occurs when objects are inserted in some other kind of collection, but not all of them are removed after use. Take the example of a java.util.LinkedList. In the trend analysis we will see that the LinkedList$Entry type is growing. Getting the referrers of that type will not get us very far as the type is pointing to itself – it is a linked list after all. This is a good time to use the option "Show instances of this type pointing to type <T>" in the Memory Leak Detector, where T in this case is LinkedList$Entry. This will display a table of entry instances pointing to other entries, together with information on how much data is kept alive by each instance. This data-kept-alive number indicates how much memory the instance is keeping alive (assuming only that object was alive in the system). The entry keeping the most data alive is normally the first entry in the leaking list. Once we have found this entry we can get the instances referring to it to see where this linked list is referenced in the system. This should lead us to the right place in the code.

Summary

The JRockit JVM has some unique capabilities for real-time memory leak detection. Memory leaks in Java are often hard to find but with the right tools it is not an impossible task. I have shown how you can use the JRockit Memory Leak Detector to detect a memory leak, find out what is leaking, and then drill down to what is causing the leak in the code.

Try It Yourself

BEA WebLogic JRockit 1.4.2_05 can be downloaded free at http://commerce.bea.com/products/weblogicjrockit/1.4.2/142_05.jsp. The installation includes the management console with the built-in Memory Leak Detector. Also see http://e-docs.bea.com/wljrockit/docs142/userguide/memleak.html.

Listing 1

```
java.util.Hashtable$Entry[]<10>  6 291 472
java.util.Hashtable$Entry[]<3>   784
java.util.Hashtable$Entry[]<4>   400
java.util.Hashtable$Entry[]<5>   400
java.util.Hashtable$Entry[]<6>   400
java.util.Hashtable$Entry[]<7>   400
java.util.Hashtable$Entry[]<8>   400
java.util.Hashtable$Entry[]<9>   400
java.util.Hashtable$Entry[]<11>  208
java.util.Hashtable$Entry[]<12>  208
```

Listing 2

```
int total = 0;
while (true)
{
for (int i = 0; i <= 60; i++)

table.put(new DemoObject(total + i),
"foo");

 // Below is the faulty line: Should be <=
and not just <
for (int i = 0; i < 60; i++)

table.remove(new DemoObject(total + i));

  total += 61;
}
```

BEA Boosts Enterprise Java Application Development with JRockit 5.0 JDK

(San Jose, CA) – BEA Systems, a world leader in enterprise infrastructure software, has announced the general availability of BEA WebLogic JRockit 5.0 Java Development Kit (JDK). Compatible with the J2SE 5.0 specification, WebLogic JRockit 5.0 is designed to offer-the-highest performing, easiest to use, and most manageable Java Virtual Machine (JVM) optimized for 32-bit and 64-bit Intel Xeon processor and Intel Itanium2 processor–based servers, according to published industry standard benchmarks. Out-of-the-box, WebLogic JRockit 5.0 is designed to allow Java developers to deploy their applications more quickly and efficiently into production, helping to achieve optimal performance through minimal configuration.

WebLogic JRockit 5.0 is available for free download at www.bea.com/download, www.bea.com

BEA Diablo WebLogic Server Beta Available For Download

(San Jose, CA) – BEA has released the beta version of its BEA WebLogic Server. The company's flagship application server, the first part of BEA's next-generation WebLogic product line, can now be downloaded for free. Diablo supports the latest industry standards, such as J2EE 1.4, and the latest Web services standards, helping to allow developers to more quickly create portable and interoperable applications.

BEA said Diablo is designed to provide an ideal platform for the development and deployment of service-oriented architectures (SOA). An improved service security framework is designed to make it easier to set up application security and prevent costly security holes. However, the company has cautioned users that this beta version should not be used in production environments. www.bea.com

Adaptis Connect Enables More Timely Treatment for Patients

(San Jose, CA) – Adaptis Inc., a business process outsourcer for Medicaid and other managed care plan operators, has built its Adaptis Connect Web application on BEA WebLogic Server 8.1 to help eliminate the administrative overhead and financial risk associated with Medicaid. The application also utilizes a BEA WebLogic JRockit JVM.

Adaptis manages numerous tasks for its clients, including approving referrals to specialists, confirming Medicaid eligibility for patients, reviewing the status of provider claims, and tracking inquiries made by doctors. These tasks have traditionally been handled manually and it wasn't unusual for them to drag on for days or even weeks due to process inefficiencies, thus delaying care to patients and stalling or even preventing payments to providers.

The Adaptis Connect Web application is designed to address those inefficiencies by helping make referrals online in real time. The application can also enable physicians to confirm patient eligibility for Medicaid more quickly, and Adaptis' claims processors can use the system to verify provider participation in the plan before authorizing payments. www.adaptisinc.com

Upgraded BEA Tuxedo Mainframe Adapters Improve Interoperability

(Shanghai) – BEA Systems has announced the availability of upgraded BEA Tuxedo Mainframe Adapters. BEA Tuxedo's proven mainframe adapters, which are designed to provide end-to-end transactional support between BEA Tuxedo and mainframe applications, can now support applications built on IMS 8.1 and CICS 2.3. With these new adapters, mainframe customers can continue to leverage their existing IT assets through creating composite architectures, linking their IMS or CICS applications on the mainframe with open, standards-based applications built on BEA Tuxedo. These adapters can also play an important role in supporting a migration strategy to open systems, which may require the mainframe applications to coexist with BEA Tuxedo during the transition period. www.bea.com

Compoze Software Completes BEA Validation Program

(Philadelphia) – Compoze Software, a provider of collaborative software for portals and enterprise applications, has announced that Compoze Portlets, BEA WebLogic Edition, has completed the BEA Validation Program and is now verified to integrate with BEA WebLogic Platform 8.1.

Compoze software is designed to provide companies with the ability to quickly and seamlessly bring popular groupware functionality from Microsoft Exchange and Lotus Domino into BEA WebLogic Portal. Together, Compoze and BEA can help provide customers with a solution designed to drive portal adoption and increase productivity by helping provide convenient Web access to applications used regularly by employees across the enterprise.

The BEA Validation Program provides a testing framework to verify the technical readiness of partner solutions that complement, operate with, or extend the BEA WebLogic Platform 8.1. www.compoze.com

BEA Systems Acquires Incomit

(San Jose, CA) – BEA Systems will enhance its substantial leadership in the telco industry through its acquisition of Incomit, a privately held Swedish company that provides infrastructure software for the telecommunications industry.

The acquisition, BEA's fifth in the past two years, will provide the enterprise infrastructure software company with key technologies to augment the BEA WebLogic Service Delivery Platform, BEA's telecommunications solution framework.

With the BEA WebLogic Service Delivery Platform, the company can respond to telco customers' demands for broader product offerings that enable the creation, delivery, and management of new converged services for next-generation telco networks. www.incomit.com

ICEreader Integrated into BEA WebLogic Workshop 8.1 SP4

(Calgary, Alberta) – BEA Systems will integrate its WebLogic Workshop 8.1 SP4 with ICEreader, ICEsoft's Java Rendering Tool.

BEA selected the ICEreader rendering software because of ICEsoft's leadership in Java-based Web browser technology. The software reduces BEA's time to market by enabling its development team to quickly implement new HTML editing capabilities for building portals and Web applications within BEA WebLogic Workshop.

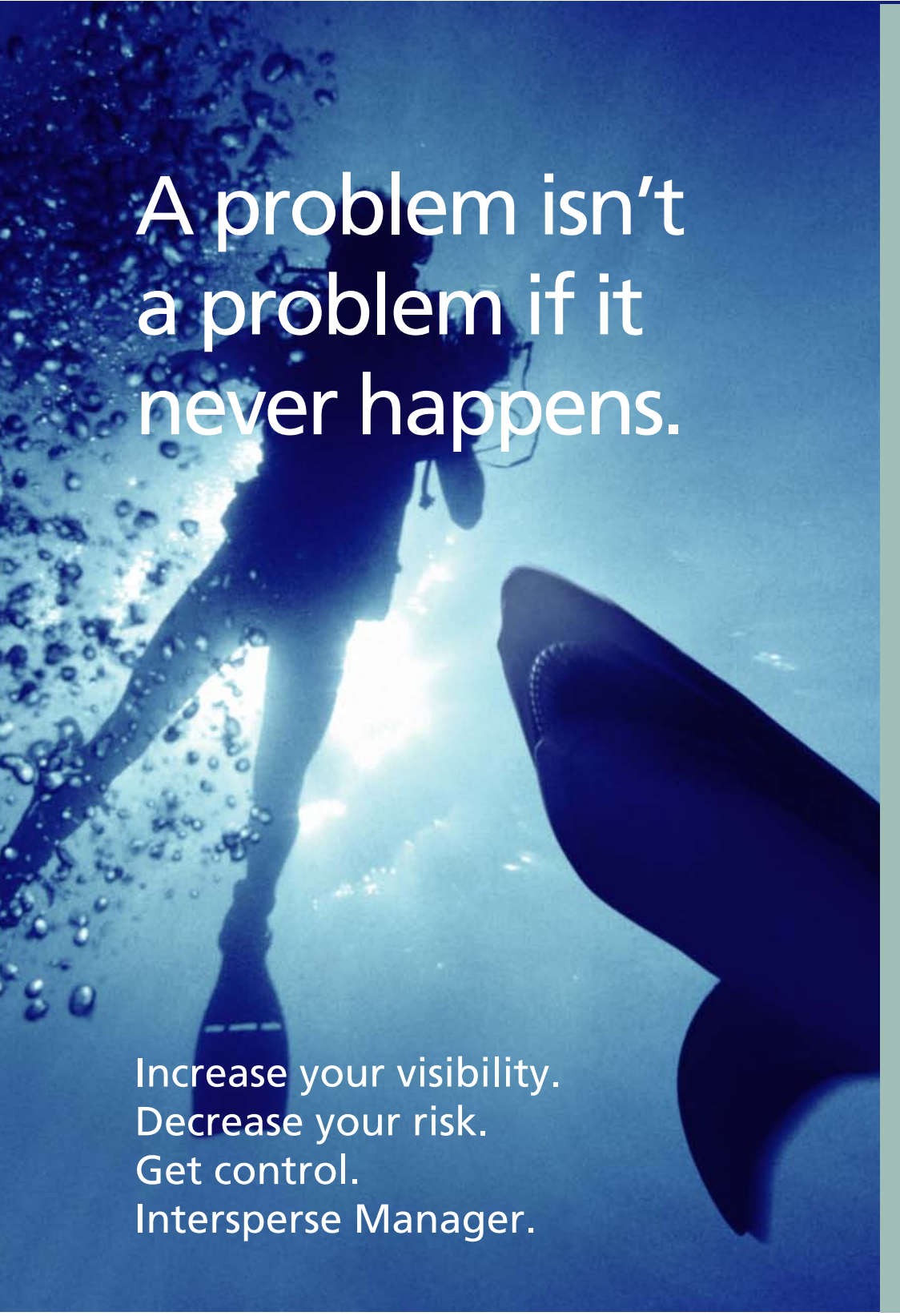
The implementation of ICEsoft's technology allows BEA to offer its customers a more tightly integrated solution and an advanced level of service. www.icesoft.com

Motive brings an enlightened approach to application management, building management intelligence into the application itself. Only Motive transcends the fragmented, disjointed reality of traditional application management, to deliver the visibility, insight and automation that support and development teams need to keep applications running smoothly.

This is no mere vision for the future. It's here now in application management products from Motive. Learn more about Motive's breakthrough approach in a new white paper about built-in management at www.motive.com/within1.

motive

www.motive.com



Enterprise information systems are more complex – and vital – than ever. But visibility into these dynamic architectures is minimal, increasing the chances of system failure and serious risk to your company.

Intersperse Manager™ removes the risks by proactively monitoring and managing distributed and SOA applications deployed on BEA. It gives users the ability to dive deeply into system infrastructure. To quickly pinpoint and correlate the root causes of problems. To automatically or manually correct issues in real time.

Intersperse Manager gives you complete visibility – and that puts you in complete control.

Call Intersperse today for
a free demonstration

866.499.2202

intersperse™

Management solutions
for the evolving enterprise.

www.intersperse.com