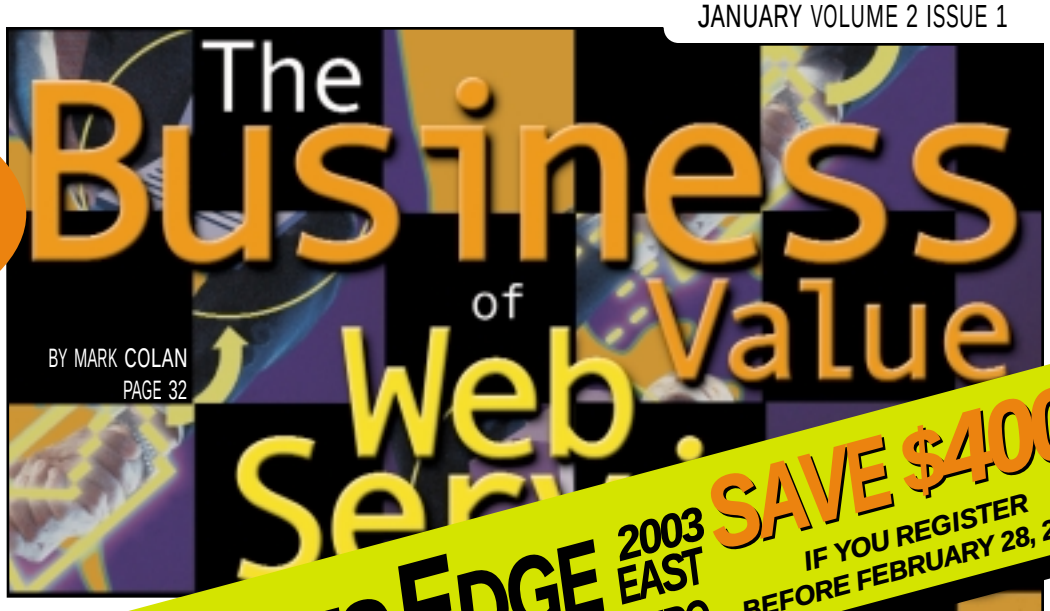
SPOTLIGHT ON WEBSPHERE 5.0 PERFORMANCE

WebSphere DEVELOPER'S JOURNAL

The World's Leading Independent WebSphere Developer Resource

WebSphereDevelopersJournal.com

JANUARY VOLUME 2 ISSUE 1



BY MARK COLAN
PAGE 32

DISPLAY UNTIL MARCH 30, 2003
$8.99US $9.99CAN
01>
0 09281 03422 3

SYS-CON MEDIA

An Elegant Solution

BY JACK MARTIN

Ring out the old, ring in the new: 2003 is upon us and the world continues to change. But some things remain the same; Java should remain the programming language du jour; Linux will continue as the first-choice open-source OS; and Apache, TCP/IP, e-mail, and the venerable Macintosh will hold onto their respective positions in the industry.

Wireless will continue to be the next really big thing that still won't happen again all year. Last year we waited for 3G to roll out across the country and offer all of these really cool services. It never did. Maybe 3G really stands for "Going, going, gone." Who knows, maybe 4G will work out. Next-generation wireless services have a lot of company, including tele-medicine, geodesic domes, electric cars, and world peace. Every year starts with great promise, only to end up with not much changing. Except that this year our government is starting off with a plan to go to war with anyone who in any way might be a threat to us at any time in the future. That should cover everyone everywhere – a lot like the universal service that 3G promised.

2003 will be a year that sees many software houses closed or consolidated. A few lucky ones will be acquired by IBM or other industry giants. And we will also see thousands of new businesses spontaneously formed each month all year long by entrepreneurs. Entrepreneurs, not big business, make up the backbone of the U.S. economy. Small businesses have always and will always drive the economy. In business – as in nature – as one thing goes extinct, seemingly out of nowhere something steps in to fill the void. Think about it – business, like nature – is always ready to try a new combination.

Who'd have thought that deploying universal e-mail in Laos would work? The vast majority of the country does not have electricity, much less telephone service. What a stupid idea. Everyone knows you need electricity and access to the telecommunications infrastructure to run a computer and connect to the Internet. You may ask why people living in a country with no electricity and with an endless supply of unexploded ordinance left over from the Vietnam War lying all over would even want e-mail service. The people are very poor and the majority make their living growing and selling small quantities of rice.

The rice market is like any commodity market; there is endless demand, but the price moves up and down over time. If there were a way to know when rice prices were on the way up, it would be invaluable business information for Laotian rice growers.

One very innovative Laotian did not see e-mail in Laos as a stupid idea. He knew that many people would find it very useful to know the current price of rice. So he set himself to the task and came up with an elegant solution. He gathered up all of the old and seemingly useless computers he could find and connected them to $30 antennas, the kind you can buy at Radio Shack. He created his own ad hoc wireless network. He could do this because the Laotian government has not auctioned off the entire wireless spectrum to businesses to create 3G networks. The man solved the electricity problem by setting up his service in such a way that it takes two people send an e-mail: one person connects his or her bicycle to a very basic generator that supplies enough electricity to power up the computer while the other person types out an e-mail.

This ad hoc network has spread all over Laos and now connects to the Internet via the capital, which has Internet access. So while the world's largest telecommunications companies struggle to deploy universal wireless Internet access in the U.S. and Europe, Laos has already deployed a free wireless network for anyone who needs to know the price of rice. It is predicted that spamming will be kept to a minimum, as someone would have to pedal the bike for the duration of the spam blast.

So while you travel through 2003, try to find elegant solutions that will really change people's lives, and maybe this time next year something will be different. It is up to you.

Happy New Year!

ABOUT THE AUTHOR... Jack Martin, editor-in-chief of ***WebSphere Developer's Journal***, is cofounder and CEO of Simplex Knowledge Company, an Internet software boutique specializing in WebSphere development. Simplex developed the first remote video transmission system designed specifically for childcare centers, which received worldwide media attention, and the world's first diagnostic-quality ultrasound broadcast system. E-MAIL... jack@sys-con.com

WebSphere® DEVELOPER'S JOURNAL

ADVISORY BOARD
Richard Arone, James Martin, Greg Williams

EDITOR-IN-CHIEF JACK MARTIN
EDITORIAL DIRECTOR JEREMY GEELAN
EXECUTIVE EDITOR GAIL SCHULTZ
CONTRIBUTING EDITOR PATTI MARTIN
SPECIAL PROJECTS EDITOR RICHARD ARONE
MANAGING EDITOR JEAN CASSIDY
EDITOR NANCY VALENTINE
ASSOCIATE EDITORS JAMIE MATUSOW
JENNIFER STILLEY

WRITERS IN THIS ISSUE
Mark Colan, Gennaro Cuomo, Cynthia Dunlop, Arthur Hicken, Jim Liddle, Jack Martin, Jim Martin, Carolyn Norton, Srini Rangaswamy Jian Tang, Ruth Willenborg

SUBSCRIPTIONS
For subscriptions and requests for bulk orders, please send your letters to Subscription Department.
SUBSCRIBE@SYS-CON.COM
Cover Price: $8.99/Issue Domestic: $149/YR (12 Issues)
Canada/Mexico: $169/YR Overseas: $179/YR
(U.S. Banks or Money Orders)
Back issues: $15 U.S. $20 All others

PRESIDENT AND CEO FUAT A. KIRCAALI
COO/CFO MARK HARABEDIAN
SENIOR VP, SALES & MARKETING CARMEN GONZALEZ
VP, INFORMATION SYSTEMS ROBERT DIAMOND
VP, SALES & MARKETING MILES SILVERMAN
PRESIDENT, SYS-CON EVENTS GRISHA DAVIDA
PRODUCTION CONSULTANT JIM MORGAN
FINANCIAL ANALYST JOAN LAROSE
ACCOUNTS RECEIVABLE KERRI VON ACHEN
ACCOUNTS PAYABLE BETTY WHITE
ADVERTISING ACCOUNT MANAGERS MEGAN RING-MUSSA ROBYN FORMA
ASSOCIATE SALES MANAGERS CARRIE GEBERT ALISA CATALANO KRISTIN KUHNLE
CONFERENCE MANAGER MICHAEL LYNCH
REGIONAL SALES MANAGERS MICHAEL PESICK RICHARD ANDERSON
ART DIRECTOR ALEX BOTERO
ASSOCIATE ART DIRECTORS LOUIS F. CUFFARI RICHARD SILVERBERG TAMI BEATTY
WEB DESIGNERS STEPHEN KILMURRAY CHRISTOPHER CROCE
CONTENT EDITOR LIN GOETZ
CUSTOMER SERVICE REPRESENTATIVE MARGIE DOWNS
CUSTOMER RELATIONS MANAGER RACHEL MCGOURAN

EDITORIAL OFFICES
SYS-CON Publications, Inc.
135 Chestnut Ridge Road, Montvale, NJ 07645
Telephone: 201 802-3000 Fax: 201 782-9637
SUBSCRIBE@SYS-CON.COM
WebSphere® Developer's Journal (ISSN# 1535-6914) is published monthly (12 times a year).
Postmaster send address changes to:
WebSphere Developer's Journal, SYS-CON Publications, Inc.
135 Chestnut Ridge Road, Montvale, NJ 07645

Getting the most from your Web site

Tuning WebSphere 5.0 Using the Performance Advisors

BY CAROLYN NORTON AND SRINI RANGASWAMY

In today's Internet environment, performance is critical for a successful Web site. Identifying the causes of poor performance and tuning the system accordingly is challenging and often requires expert knowledge of available monitoring data. The Performance Monitoring Infrastructure (PMI) in IBM's WebSphere Application Server 5.0 provides more than 150 performance metrics describing runtime and application resources. As part of IBM's autonomic computing initiative, the performance advisors provide a starting point to the application server tuning process. This article is an introduction to those performance advisors and their role in tuning WebSphere 5.0 for optimal performance.

ABOUT THE AUTHOR

Carolyn Norton is manager of the WebSphere Solutions Performance Technology department. Prior to that she was chief technical officer for the Sydney Olympics, and part of the Worldwide Olympic Technology Team in Nagano. Carolyn, a member of the IBM Academy of Technology, holds a PhD in mathematics from MIT and an AB from Princeton University.

E-MAIL

nortonc@us.ibm.com

Performance Advisors – Overview

Tuning WebSphere Application Server involves analyzing performance data and determining the optimal server configuration. This requires considerable knowledge about the various components in the application server and their performance characteristics. The performance advisors encapsulate this knowledge, analyze the performance data, and provide configuration recommendations to improve the application server performance.

The performance advisors analyze WebSphere PMI data using general performance principles, best practices, and WebSphere-specific rules of thumb. The goal of both performance advisors is to identify common performance problems such as misallocation of resources. For example, the performance advisors check the ORB (object request broker) thread pool size. If it is too small, the thread pool can be a bottleneck, unnecessarily restricting concurrency in the system. If it is too large, the additional memory needed and the extra context switching can hurt overall performance.

The performance advisors are technology previews for WebSphere Application Server version 5, on Windows and Linux platforms. There are two performance advisors: the runtime performance advisor and the performance advisor in Tivoli Performance Viewer (TPV advisor).

The runtime performance advisor runs in the application server process, and the TPV advisor runs in the Tivoli Performance Viewer client. Figure 1 shows a simplified architecture view of the performance advisors. Both advisors provide advice on the following application server resources: thread pools, connection pools, HttpSession size and cache, prepared statement cache, and JVM heap size. The performance advisors can be downloaded from www7b.software.ibm.com/wsdd/downloads/performance_advisors.html.

Runtime Performance Advisor

Running in the background of the application server, this advisor gathers the necessary performance data from PMI, analyzes it, and provides recommendations based on the current system load. Since the advisor uses PMI data, you must first enable the performance monitoring service on the application server; then the runtime advisor automatically enables the individual PMI counters required. The runtime advisor is configured using the WebSphere administrative console. Some of the primary settings include enabling the runtime advisor itself, setting the number of CPUs on the application server machine, and setting the frequency at which the advisor is run. It is also possible to enable and disable specific advice. Figure 2 shows the runtime advisor configuration in the WebSphere administrative console.

The recommendations from the runtime performance advisor are issued as standard WebSphere warning messages. These warnings appear both in the WebSphere Status panel in the administrative console under WebSphere Runtime Messages, and in the SystemOut.log file in the appserver log directory (e.g., c:\WebSphere\AppServer\logs\server1). The runtime advisor is designed to have a minimal performance impact so it can be used in a production environment. In our

lab tests, the impact of running the runtime advisor is around 1% on top of any PMI overhead.

TPV Advisor

The second performance advisor in the technology preview is integrated with the Tivoli Performance Viewer (TPV), which is a stand-alone performance monitoring tool that retrieves and displays the PMI data. The performance advisor in TPV uses this PMI data and displays its recommendations in the TPV GUI. After-the-fact analysis can be obtained by running the advisor on a recorded TPV log.

Unlike the runtime advisor, the TPV advisor is run manually by the user at a specific point in time. Because it runs in the TPV JVM and not during the application server process, this advisor has more freedom than the runtime advisor to do computationally intensive analysis without impacting server performance. In addition to the advice provided by the runtime advisor, the TPV advisor checks the WebSphere configuration for settings that deviate significantly from the norm. The TPV advisor also provides a performance summary of various runtime resources in a graphical format. This performance summary includes average response time and throughput for the Web and EJB containers, CPU utilization, and pool utilization for Web container threads, ORB threads, and database connections. Figure 3 shows a typical screen shot from the TPV advisor.

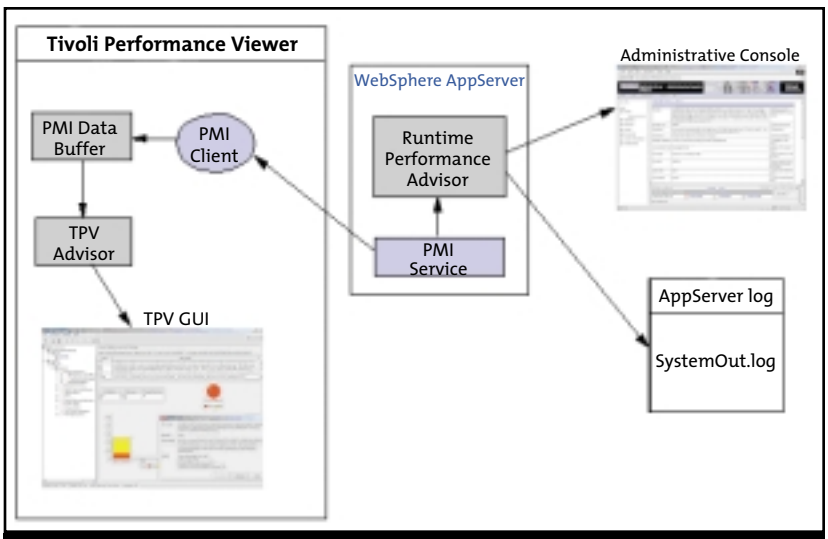


FIG. 1: PERFORMANCE ADVISORS: A SIMPLIFIED ARCHITECTURE

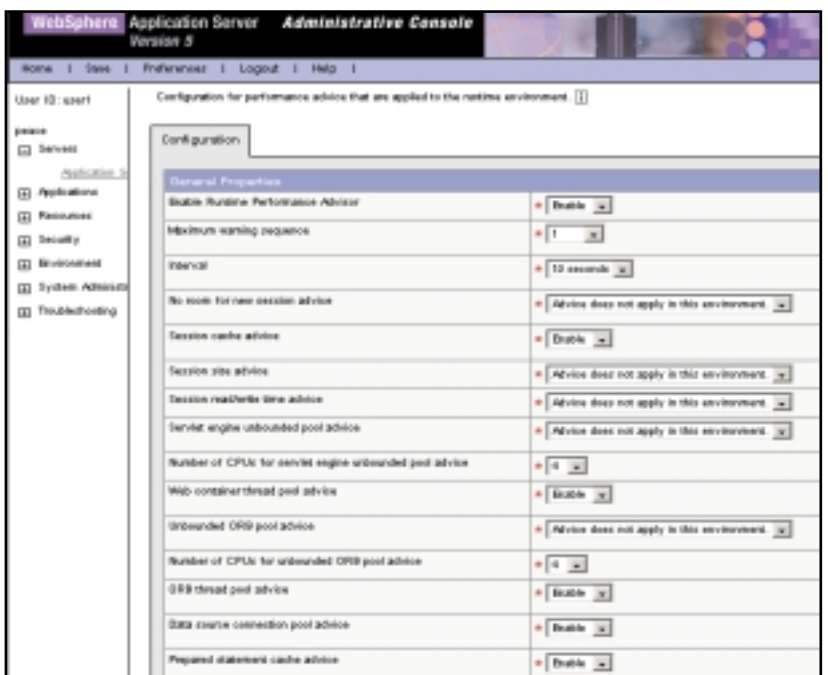


FIG. 2: RUNTIME PERFORMANCE ADVISOR CONFIGURATION SCREEN

Performance Advisors – A Detailed Look

The performance advisors are based on the principle that getting the best performance from an application server requires optimal utilization of the runtime resources. The advisors use the PMI data to calculate the current resource utilization level and provide recommendations to achieve optimal resource utilization.

Let's look at some examples in detail. Consider the data source prepared statement cache, which helps to decrease the cost of executing a prepared statement by caching the compiled statement. If the cache is full, an old entry in the cache is discarded to make room for the new one. Best performance is generally obtained when the cache is large enough to hold all of the statements used in the application. The PMI counter "prepared statement cache discards" indicates the number of statements that have been discarded from cache. The advisors check this counter and provide recommendations to minimize the cache discards.

Now let's look at the pools in the application server. The idea behind pooling is to use an existing thread/connection from the pool instead of creating a new instance for each request. Since each thread/connection in the pool consumes memory and increases the context-switching cost, the pool size can be an important configuration parameter. A pool that is too large can hurt performance as much as a pool that is too small. The advisors use PMI information about current pool usage, pool minimum/maximum size, and application server CPU utilization to recommend efficient values for the pool sizes.

A third area of concern is HttpSession. The HttpSession cache can be configured to be bounded or

ABOUT THE AUTHOR

Srini Rangaswamy is a software engineer in the IBM WebSphere performance development group. He is responsible for the design and development of Tivoli Performance Viewer. He holds a master's degree in software systems from BITS Pilani, India.

E-MAIL
sriniri@us.ibm.com

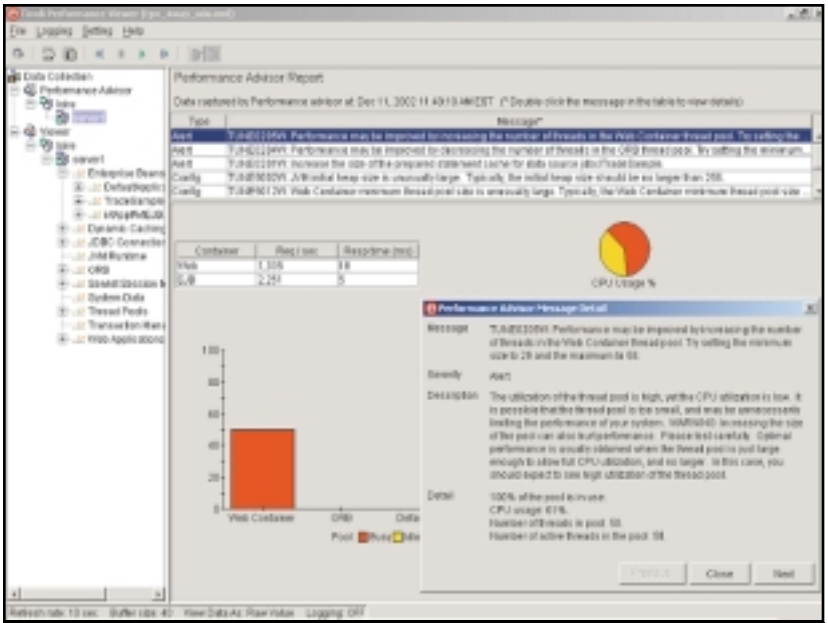


FIG. 3: TPV ADVISOR: A TYPICAL SCREEN SHOT

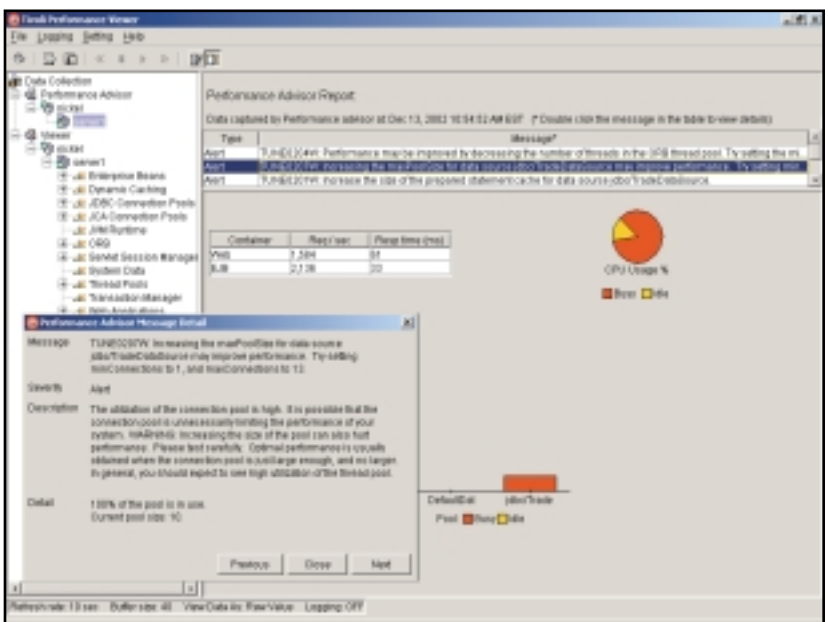


FIG. 4: TPV ADVISOR OUTPUT WITH OUT-OF-THE-BOX SETTINGS

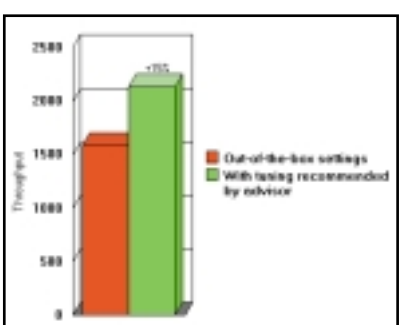


FIG. 5: WEB CONTAINER THROUGHPUT COMPARISON

unbounded. If it is bounded (the "Allow Overflow" flag is false), then sessions may not be available when the application needs them. When this happens, the performance advisors check the PMI counter "no room for new session" and recommend increasing the cache size. Another common performance problem occurs when the individual sessions become too large. When using persistent sessions, the advisors track the average session size and warn when the size gets too large.

Using the Performance Advisors

In this section, we walk through an example that shows how the performance advisors help in improving the performance of an application. We use the Trade3 application with WebSphere Application Server version 5, running on a four-way Microsoft Windows NT system. First, we run the application with the out-of-the-box WebSphere settings. In the second run we use the recommendations from TPV advisor.

Figure 4 shows the output from the TPV advisor when running the application using out-of-the-box settings with 50 users. The TPV advisor recommendations are to decrease the maximum number of threads in the ORB pool from 10 to 3, increase the maximum pool size of jdbc/TradeDataSource from 10 to 13, and increase the prepared statement cache size of jdbc/TradeDataSource from 10 to the number of prepared statements in the application. During this test, the Web container throughput is 1,584 requests/sec., with an average response time of 61 ms.

Now let's apply the recommendations by setting the ORB thread pool maximum size to 3, the jdbc/Trade DataSource maximum pool size to 13, and the jdbc/TradeDataSource prepared statement cache size to 25. We restart the application server and rerun the 50-user test. Figure 5 shows the results. The throughput increases by 15%, and average response time decreases slightly.

These performance advisors base their recommendations on the current system load. In a production environment, you should run the advisors several times to see how the advice changes with normal load fluctuations. In a test environment, run the advisors with a load that closely resembles the expected production load. For further information about using the performance advisors, refer to the documentation that comes with the download package.

Summary

Tuning your WebSphere Application Server correctly is a critical part of getting the best performance from your Web site. These performance advisors help you to do this without requiring you to become a product expert.

WebSphere 5.0: What's New on the Performance Front

BIG IMPROVEMENTS STEM FROM SUPPORT FOR EJB 2.0

BY GENNARO CUOMO AND RUTH WILLENBORG

ABOUT THE AUTHOR

Gennaro (Jerry) Cuomo is a distinguished engineer at IBM. He has been a core member of the WebSphere development and architecture team since its inception in 1998. Jerry is CTO of the WebSphere Technology Institute, which is responsible for pursuing the next generation of WebSphere technologies.

E-MAIL
gcuomo@us.ibm.com

IBM WebSphere Application Server (WAS) version 5.0 continues the tradition of improved performance from release to release. These improvements come from several key areas, which include:

- WAS's implementation of new J2EE 1.3 APIs, notably EJB 2.0
- Web services using SOAP
- Dynamic caching and edge componentry
- HTTP session management
- Extensions for the enterprise

WAS 5.0 complements these performance enhancements with an enhanced collection of runtime tools for performance, including Java Management Extension (JMX) interfaces for performance monitoring and performance advice. This article presents these features, highlighting their contribution to WAS 5.0's performance.

J2EE 1.3

WAS 5.0 supports the J2EE 1.3 programming model. The most significant area of performance advancement is within the EJB 2.0 portion of J2EE 1.3. There are two aspects of EJB 2.0 that support building higher-performing EJB-based applications: local interfaces and container-managed entities.

LOCAL INTERFACES

EJB 2.0 local interfaces provide optimized EJB-to-EJB access. By definition, EJB 2.0 local interfaces are local method calls that use call-by reference semantics. Unlike traditional EJB remote or "location-transparent" methods, local interface methods have no intramethod overhead and do not create copies of parameter or return objects. Lab measurements show a 20% to 2x improvement, depending on the size and complexity of parameters and/or return values.

CONTAINER-MANAGED ENTITIES

Container-managed entities receive a boost in both usability and performance by allowing for container-managed relationships (CMR). CMRs define a relationship pattern between EJBs. Consider an application in which customers place orders. There is a relationship between customers and orders, in that every time the application accesses the Customer EJB, it also accesses the orders that a given customer has made. Furthermore, using the IBM WebSphere Studio tooling, you can define a read-ahead policy to improve the performance by preloading associated objects.

Figure 1 shows an association between customers and orders. The implementation is a CMR with one customer having many orders. A read-ahead hint on this relationship optimizes performance by preloading orders with the related customer in one generated SQL SELECT statement with a SQL JOIN for both underlying tables. This halves the number of database accesses for this logically related data. Using the read-ahead hint yielded a 10–15% performance improvement in lab measurements involving similar CMRs with 10 associated objects.

Base EJB 2.0 supports one read-ahead hint per association. WebSphere Application Server Enterprise Edition adds the capability to define multiple read-ahead hints per CMR relationship. This added capability is part of *application profiling*, discussed later in this article.

Web Services

The most significant Web services performance optimization is in the SOAP support in the Web Services Technology Preview. In particular, the JAX-RPC, which is the standard programming interface to Web services, uses an optimized Apache Axis as the core of the WAS Web services stack. Early benchmarks on this base exhibit significant improvement in performance, depending on the complexity of the SOAP messages. This is illustrated in Figure 2, which shows a 2x improvement in throughput between WAS 4.0 and the WAS 5.0 Technology Preview. In this scenario SOAP messages containing a Java array containing 100 strings are exchanged between a Microsoft .NET client and WAS.

The new Axis implementation also allows for pluggable XML parsers following the Java API for XML Processing (JAXP) standard. By default, WAS configures the standard Xerces parser as the parser used for Web services. However, specialized parsers can be placed under JAXP to produce even better performance results for Web services. IBM is currently testing specialized parsers for future updates. Upgrading to a faster parser is transparent to the applications using Web services.

Dynamic Caching and Edge Components

The most efficient interaction is the interaction never done. This is the spirit of dynamic caching. WAS 5.0 introduces a powerful set of integrated cache-enabled APIs and services within the application server and Edge Server components, as shown in Figure 3. Utilizing these APIs and services can easily lead to doubling or tripling your application's performance.

WAS CACHING PROGRAMMING MODEL

WAS 5.0 allows for result caching of servlets/JSPs and CMP Entities and EJBs. Caching of SOAP responses for Web services and caching of command results (when using the com.ibm.websphere.command interface) are also supported. Resulting servlet/JSP fragments can also be cached on Edge Servers using Edge Side Includes (ESI) technology.

Result caching of servlet/JSP, EJB, and SOAP requests can be accomplished without the use of explicit caching APIs. Caching policies are described declaratively, using XML deployment descriptors. For example, the WebSphere Application Assembly Tool (AAT) is used to express how to cache and invalidate cached content. A Cache policy XML file, cachespec.xml, provides the caching metadata required by the dynamic caching service.

WAS DYNAMIC CACHING SERVICE

The runtime foundation for caching in WAS is the WebSphere dynamic caching service (Dynacache). Dynacache is a Java object cache that caches objects directly in WAS's (i.e., the JVM's) memory with the ability to "overflow" objects to disk to avoid over-utilizing memory. Using the Replication domain facility in WAS, results are optionally replicated across a cluster to avoid expensive regeneration on each server. Invalidation events to remove stale data from Dynacache are distributed using the same facility.

EXTERNAL CACHES

Contents in Dynacache can be shared with external caches, including the Caching Proxy Server (which is a WebSphere Edge Component), the IBM HTTP Server Cache Accelerator (HTTP kernel cache), or the WebSphere Web Server Plug-in (which plugs into most standard HTTP servers). These external caches support full-page caching of content generated by servlets/JSPs.

In addition, WAS 5.0 supports distributed fragment caching and assembly based upon the open ESI technology (www.esi.org). This enables WAS customers to improve delivery of personalized, dynamic Web content by exploiting ESI-capable "edge" servers, including the WAS Web server plug-in and in-network service providers such as Akamai. For example, with this powerful new feature, a page composed from a collection of servlets/JSPs (e.g., a portal-style page) can be assembled in the Web server from either cached or noncached components. By marking cached JSP/servlet results as "edgeable," using the assembly tool or by directly editing cachespec.xml, the corresponding cached results within Dynacache are copied into the plug-in memory (i.e., Web server).

EDGE COMPONENTS

WAS 5.0 is the industry's first J2EE application server to bundle edge-of-network functions specifically designed to add value to J2EE applications. The added functionality extends WebSphere's J2EE capabilities out into the network, placing content closer to the end user and enabling a richer Web site experience.

WebSphere version 5.0 Network Deployment introduces the Edge Components CD-ROM. This new offering concentrates the edge-of-network functions – which are typically deployed in a DMZ setting – into a single installable package.

Dispatcher is a general-purpose load balancer for routing and balancing of HTTP(s) requests entering into the enterprise. It is the point of entry into the data center, typically sitting in front of the Web server or proxy tier. Dispatcher monitors many aspects of the resources that are being load-balanced and dynamically adjusts the workload distribution, ensuring that traffic is routed to the most appropriate place as the load within the site changes.

For example, Dispatcher is used to distribute HTTP requests across a cluster of HTTP servers fronting WebSphere Application Servers. Using the Dispatcher technology typically provides near-linear scaling as servers are added to the cluster. In addition, the smart-routing capabilities within Dispatcher optimize workload distribution, using metrics such as CPU, as well as custom advisors sensitive to the actual performance of WebSphere applications.

In addition to the Dispatcher workload management capabilities, WebSphere includes workload management distribution between HTTP servers and Web containers, and EJB clients (including servlets) and EJBs. Version 5 now supports weighted workload distribution in addition to round-robin capabilities.

Caching Proxy provides the required functionality to front-end a cluster of WebSphere Application Servers. SSL termination, authentication, high-speed persistent caching, and intelligent routing functions are combined within a single network hop, thus alleviating the need to deploy multiple proxies that span multiple hops. Integration with Dynacache provides caching of servlet and JSP pages, while integration with Dispatcher technology and WebSphere's

ABOUT THE AUTHOR

Ruth Willenborg is a senior programmer manager at IBM. She has 17 years of experience in software development and is currently a member of the WebSphere Development Organization. Ruth is responsible for the WebSphere Application Server Performance team. She is coauthor of *Performance Analysis for Java Web Sites* (Addison-Wesley, 2002).

E-MAIL
rewillen@us.ibm.com

J2EE HTTP session provides optimized workload management and routing of stateful WebSphere sessions.

High-Performance HTTP Session Management

One of the most performance-sensitive aspects of a J2EE application is the management of stateful applications. Application state is typically programmed using the servlet HTTP session APIs and managed using the WAS Session Manager. Managing such sessions in clustered configurations is one of the traditional value propositions of an application server. The WAS Session Manager capabilities allow choices between various degrees of performance and HTTP session data integrity (failover) in clustered environments.

The option with the best performance characteristics operates on a *server affinity–based* model without failover support. With this model, a front-end server (e.g., Web server/plug-in, caching proxy server, or Dispatcher) uses the JSession token generated by the Session Manager and typically stored in an HTTP cookie header to route requests to the particular server in the cluster that has the requester's session data.

WAS builds upon this model, adding failover support by backing up session data to a relational database. Affinity-based routing matches user requests to the servers containing their session data.When the application modifies the user's session data, an update is sent to the database in order to keep a fresh backup of the session. The backup version is used by another server in the cluster in the unlikely event of a failure in the server containing the session data. Though persisting HTTP session to a database is a very effective failover technique, some customers requested a lower-cost alternative that doesn't require a highly available database.

For this reason, WAS 5.0 offers the option to replicate HTTP session state in a cluster using a memory-to-memory data replication scheme. This scheme does not require a relational database. Instead it uses the Replication domain facility, a JMS-based, publish-and-subscribe model, to flexibly replicate data between servers in a cluster. (Remember, this is the same service that is used to replicate cached data and send cache invalidation events in WAS clusters.)

Within a replication domain, groups of servers are defined as being part of a replication group. A replication group can contain all servers within the cluster, a single "buddy," or some other arbitrary subcollection. An instance of WAS is configured as a replicator, acting as a message broker for the replication domain. Furthermore, a second replicator can be configured for availability purposes. This provides an efficient, flexible, standards-based mechanism for replicating session, without the need for added investment in relational database technologies.

The performance characteristics of persistent session, either shared through database or in-memory, are highly dependent on the size of the HTTP session object. Best practices advise keeping HTTP session objects small for optimum performance. Figure 4 shows the performance results for a cluster test using Ping Session, a servlet with a small HTTP session. In this test, only minimal performance impact is seen using in-memory replication across a five-node cluster.

WebSphere Application Server Enterprise

The WebSphere Application Server Enterprise Edition, available soon, offers programming model extensions to the base and Network Deployment editions of WAS. These extensions offer significant functional, productivity, and performance capabilities. Functional capabilities include process choreography (workflow) and dynamic query. Productivity

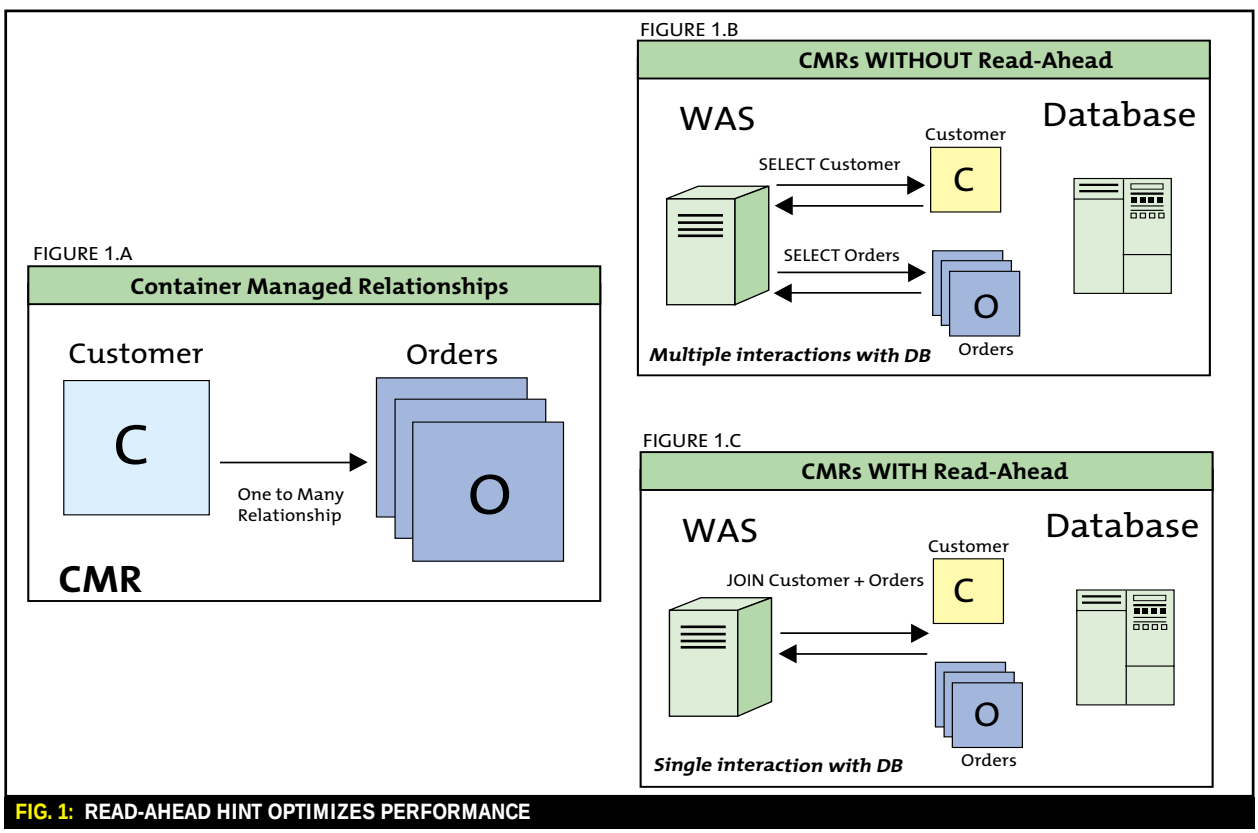


FIG. 1: READ-AHEAD HINT OPTIMIZES PERFORMANCE

capabilities include support for business rule beans, work area, and object pooling. This edition also offers opportunities for improving performance, scalability, and efficiency through application profiling, asynchronous beans, and other capabilities. Let's look specifically at the application profiling service and capabilities for startup and asynchronous beans.

APPLICATION PROFILING

The application profiling service enables deployment-time configuration techniques that enable your application to run more efficiently, with better scalability and performance. You can configure tasks that identify incoming requests, access intents that determine concurrency, and other data-access characteristics. Access intents enable you to specify data-access characteristics on a method-level basis. The application profiling function supports multiple access intents per method, optimizing access to the data for specific callers.

For example, some callers invoke a method with only the intent to read the data, while others can invoke the same method for read with intent to update. Using the application profiling service allows the programmer to develop a single method having different usage-based intents. Using a simple primitive test on a workload that is two-thirds read-intent, one-third read-for-update, the primitive runs three times faster with application profiling. This is compared to the same primitive if all accesses require a pessimistic read for update intent.

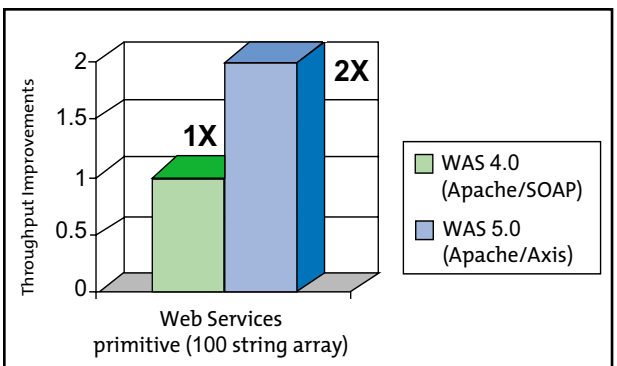


FIG. 2: APACHE/AXIS IMPLEMENTATION DOUBLES THROUGHPUT

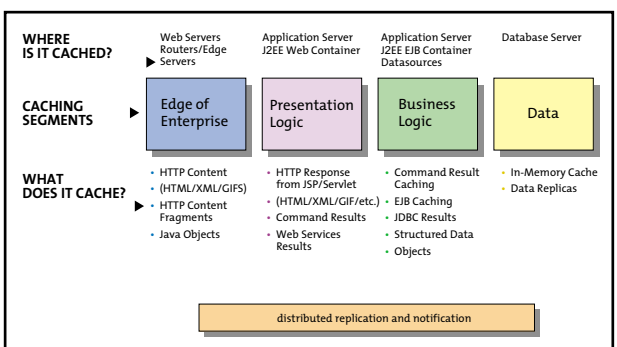


FIG. 3: WAS 5.0 INTRODUCES A POWERFUL SET OF INTEGRATED CACHE-ENABLED APIS AND SERVICES WITHIN THE APPLICATION SERVER AND EDGE SERVER COMPONENTS

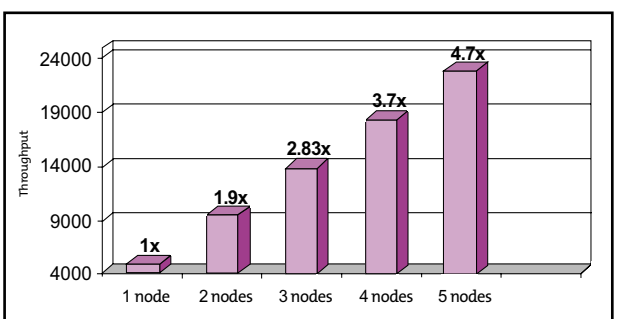


FIG. 4: IN-MEMORY SESSION REPLICATION

STARTUP AND ASYNCHRONOUS BEANS

Enterprise adds a new set of APIs that allow J2EE applications to execute business logic when an application starts and when an application stops normally. For example, a startup bean can be used to warm up a CMP EJB cache by executing finder methods on the required entity beans. Startup beans also support asynchronous operations. The API set enables creation of a work object asynchronous bean to execute the cache warmup operation asynchronously from other work.

Asynchronous (async) beans also provide other parallel-processing mechanisms to an application. For example, writing log records to a database requires time to execute each logging transaction. Instead, the application writes logging records to a vector. Meanwhile, the async bean monitors the vector and writes the log records to the database. An example application using async beans to log such records shows a 2.5x improvement in the rate of logging.

Monitoring and Autonomic Optimization

WAS v5 extends the performance monitoring and tools capabilities of earlier releases. New capabilities include the Performance Advisor Technology Preview, offering tuning and configuration recommendations, as well as new performance monitoring data and interfaces. A brief overview is provided here, and a more detailed article, "Tuning WebSphere 5.0 Using the Performance Advisors" by Srini Rangaswamy and Carolyn Norton, is included in this issue.

NEW PMI METRICS AND JMX SUPPORT

The Performance Monitoring Interface (PMI) is extended to provide additional performance data. In addition, all the performance data is now available through JMX interfaces (as well as the original PMI client APIs). For example, the new performance data includes persistent HTTP session size and JDBC read/write times. New performance modules for Dynacache, Workload Management, and the ORB (object request broker) were also added.

The Tivoli Performance Viewer can be used to view the PMI data. For example, an administrator can view the Dynacache cache hit rate on a given servlet or JSP.

RUNTIME PERFORMANCE ADVISOR

A new technology preview provides warnings to the administrator for mistuned pools, queues, and caches. This advisor seamlessly integrates into the WAS runtime and is configured through the admin console. The advisor runs in the background, accesses PMI and configuration data, and exercises a set of tuning best practices. When the advisor detects tuning issues, a warning, along with advice, is generated to the admin console. For example, if the advisor detects a high discard rate on the prepared statement cache, a warning is generated recommending that the administrator increase the cache size.

TIVOLI PERFORMANCE VIEWER ADVISOR

This technology preview integrates performance advice into the Tivoli Performance Viewer (TPV), formerly called Resource Analyzer. In this scenario, a tuning report is generated by the user's request, using the data collected by the TPV. A superset of the runtime advice is applied to the data, and the analysis report includes a set of tuning advice for modifying the settings on pools, queues, and caches.

Because this capability is integrated with the TPV, the reports are generated in real time during load testing or production. In addition, the TPV logging capabilities support running the report offline for analysis.

REQUEST METRICS

WAS 5.0 introduces a new type of performance data, request metrics. While the PMI provides average response times for servlets/JSPs within an application server and values for WAS runtime resources, request-based views of performance data are often valuable. Request metrics provides this data, tracking servlet, EJB, and JDBC entry and exit calls. As a request crosses JVM boundaries, a correlator is passed to allow end-to-end request timing. Request metrics is enabled for specific IP addresses or URLs. The data is logged to an Application Response Measurement (ARM) agent or the WAS logs.

Tools such as Tivoli Monitoring for Transaction Performance (TMTP) provide the ARM agents and produce end-to-end transaction monitoring reports. Request metrics is extremely powerful when enabled for the specific clients generating TMTP synthetic transactions. In this scenario, if a TMTP synthetic transaction shows a response-time spike, TMTP supports detailed transaction drilldown into the ARM data as shown in Figure 5.

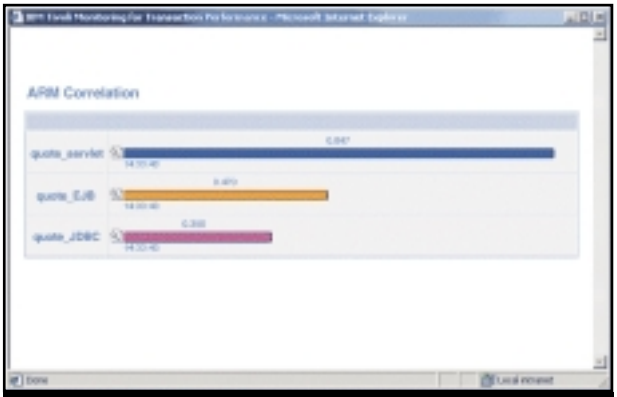


FIG. 5: EXAMPLE OF TIVOLI MONITORING FOR TRANSACTION PERFORMANCE ARM SUPPORT

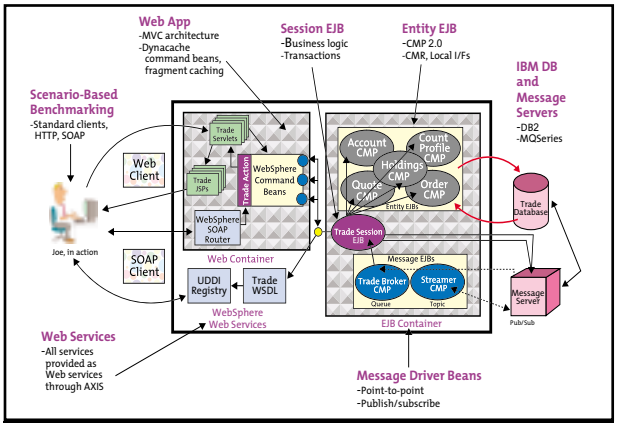


FIG. 6: TRADE 3 PERFORMANCE BENCHMARK AND APPLICATION

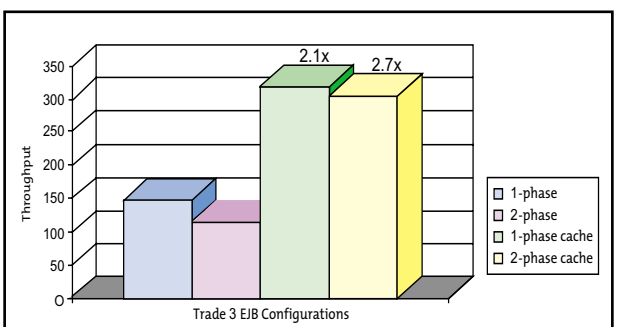


FIG. 7: TRADE 3 PERFORMANCE RESULTS

Putting It All Together: Trade3 Performance Benchmark

Trade3 is the third generation of the WebSphere performance benchmark and sample application, a highlight of the WebSphere performance site: www.ibm.com/software/webservers/appserv/performance.html.

For WAS 5.0, Trade was redesigned to illustrate the significantly expanded programming model and new performance technologies. Trade3 provides a real-world workload, enabling performance research in J2EE 1.3 and Web services, including key WAS performance components and features. Trade3 builds on the fundamental established in Trade2, which is being used for performance research on a wide range of software components and platforms, including WebSphere, DB2, Java, Linux, and more.

Trade3, as shown in Figure 6, includes many of the new J2EE 1.3 and Web services features. New EJB 2.0 capabilities, including CMRs, local interfaces, and EJB Query Language, are exploited. Command-bean and fragment-caching techniques are leveraged. Trade3 also uses JMS and can be run in either synchronous or asynchronous, two-phase commit modes.

Trade3 performance results are collected in many configurations. Figure 7 shows comparisons of Trade3 in both synchronous and asynchronous modes. Two-phase commit performance is improved in version 5, and as you can see, the difference between one-phase commit and two-phase commit for Trade3 is quite small.

Trade3 also provides an excellent example implementation for Dynacache exploitation. The results show greater than 2x performance gains through Dynacache. Additional gains, up to 4x, depending on the workload mix, are obtained if ESI is also used.

Summary

WebSphere Application Server version 5.0 offers exciting new capabilities for developing high-performance, transactive, Web-based applications. Web sites. New functions, such as EJB 2.0 and Dynacache, provide opportunities for significant performance gains. The Edge components, in-memory session replication, and other new capabilities offer flexible, high-performing end-to-end deployment capabilities. These new performance capabilities, combined with new monitoring support, make version 5 the best platform for your Web site.

Excerpted from Chapter 10: Enterprise Process Management

Professional IBM WebSphere 5.0 Application Server

Professional IBM WebSphere 5.0 Application Server provides a perspective on the philosophy and rationale behind WebSphere Application Server, taking readers through the programming and deployment model and familiarizing them with the WebSphere Studio Application Developer tool set. The book explains how to use the application server to build business applications, and how to integrate businesses. The following excerpt is from Chapter 10 on Enterprise Process Management and introduces the concept of Process Choreography as a mechanism for more efficiently creating, running, and changing business processes.

ABOUT THE BOOK

Professional IBM WebSphere 5.0 Application Server, by Rob High Jr., Eric Herness, Chris Vignola, Tim Francis, Jim Knutson, and Kim Rochat

ISBN: 1861005814
Price: $79.99
Publisher: Wrox Press
Available: January 2003

In the early '90s, many companies invested in traditional client/server architectures by building fat-client applications with rich graphics that offloaded legacy-system processing time.

Business Process Management

Business Process Management (BPM) is about modeling, implementing and managing the execution of automated business processes. Automated means that the process is driven by an application that orchestrates the interaction of the various human resources and software components which are required to perform the business process. With a few "green field" exceptions, BPM application development generally is a "meet-in-the-middle" process, which comprises these two elements:

- A high-level requirements specification of the business functions that an enterprise wants to offer or perform to achieve a business goal within certain constraints (cost and time limits, quality of service, and so on). Often this requirements specification is derived from the results of a more or less formal business process analysis effort. In other cases, the functions to be provided are dictated by public standards or external partners. In any case, the process specifications are likely to change frequently.
- A set of existing application components that encapsulate operational data or perform business functions that need to be integrated into process execution. These components can be legacy applications, packaged applications, new J2EE components, or web services provided by external partners. Since they were not built specifically for use in a particular business process, some adaptation needs to be done to fit these components into the overall process.

The goal of BPM application development is to fill the gap between these two ingredients and make it easier to do the following:

- Incorporate existing application components and resources into the process and manage the interactions of human process participants.
- Add new business logic and modify previously defined behavior of the process without affecting other components used by the process.
- Manage execution of the processes from a business user's perspective. This includes capability to suspend execution of the process or parts of the process if necessary.
- Monitor status of the process at any point in time, providing information about progress of the overall process, status of individual process tasks, and resources used by the process.
- Analyze business process execution, providing an execution trace of the process that offers sufficient information to analyze performance of one or more processes. This can be used to evaluate process performance, compare it to the objectives defined earlier and to enable process optimization based on lessons learned.

Further background and explanation of business process management would be be going into too much detail. It suffices to say that the challenge in the application server space is to meet these BPM goals

in a way that is complementary and appropriate for application server-based business solutions. This really means that we need to expose and present these capabilities in a way that is natural and reasonable to those engaged in J2EE centered development activities. The preceding statements are actually useful as a manifesto and set of guidelines for any infrastructure base that intends to support BPM. In our case, the infrastructure base is J2EE and web services.

As is usually the case, existing specifications for J2EE 1.3 could be pushed to the limits and used to implement many of the concepts just described. However, to more appropriately address these challenges, additional services and capabilities are called for. Let's see some of the services supported by WASD, starting with Process Choreography, and later Business Rules.

Process Choreography

The model for execution of a business process is often called a workflow. In software, workflow represents the automation of a business process or part of a business process. Sometimes workflow is referred to as just flow, when talking about a specific automated business process. WASD has a new feature called Process Choreographer. The workflow capabilities of WebSphere are categorized and measured under this umbrella. Process Choreographer in the runtime involves execution and management of flow-based applications. WebSphere Studio Application Developer, Integration Edition (WASD IE) offers tooling that assists developers creating flow-based applications for later execution in the WASD runtime.

WebSphere's support for workflow brings about the true integration of the Java and J2EE worlds with the workflow world. Accessing non-Java artifacts during business processes is possible and simplified through the use of service-oriented architecture. However, when new activities are necessary, they can be constructed in Java using the full power of the J2EE programming model.

Workflow can be depicted graphically using a directed graph. The graph nodes represent individual steps in a flow. The graph edges or connectors describe the execution order of those steps (including conditional and concurrent execution), as well as the data that is needed by the activities. Each of the activities references an "activity implementation" which can be a sub-flow or an elemental operation, for example, a method of an EJB or a web service. People can be assigned to activities if humans perform the respective steps. Actions performed by humans demand that the infrastructure have in place an organizational model and demand that the workflow runtime have access to that. Figure 1 shows a very simple process as a series of steps shown in WebSphere Studio Application Developer, Integration Edition. Each step executes a different kind of activity just for purposes of demonstration. More refined and complete examples will follow later in this chapter.

At runtime, the business process engine executes instances of business processes, either in memory for short running flows, or with persistent state for long running flows. The individual activity implementations are invoked when needed, the state of the business process is tracked, work items are assigned to people if and when needed, and access to this information is provided via a worklist handler interface and associated GUI. The Process Choreographer capabilities of WebSphere can be used to describe a variety of "executables," from short running scripts involving a couple of service invocations up to long running business processes involving B2B interactions and people.

Many existing IBM products use flows of various types as their execution model – some examples are WebSphere MQ Integrator (formerly MQSeries Integrator), Lotus Workflow, WebSphere Adapters (formerly MQSeries Adapter Offering), Enterprise Access Builder, and MQSeries Workflow. Each of these products is specific to certain environments and certain types of flows. WebSphere MQ Integrator, for example, is optimized to flows that manage the routing and handling of messages in a messaged system.

WebSphere's Process Choreographer adds a general-purpose flow engine (Business Process Container) to the application server, which allows for the seamless management and efficient execution of all kinds of flows in the application server. Business

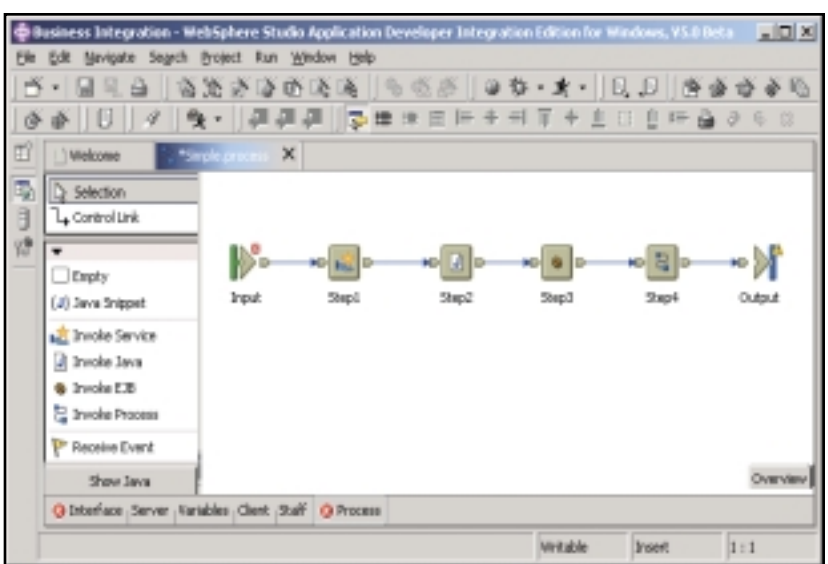


FIG. 1: A PROCESS IN WEBSPHERE STUDIO APPLICATION DEVELOPER

processes, a term we will use interchangeably with flows, are deployed as part of EAR files, and are managed by the WAS administration console. WebSphere's Process Choreographer is based on a business process engine written in pure Java that was designed with extensibility, flexibility, and performance in mind. With these design principles in place, it is intended that numerous kinds of flows can be executed using this technology to solve many business process problems in a wide variety of environments. The next section provides more details on the possible role for automated business processes.

The Role of Business Processes

Automated business processes can play a key role in architectures which are based on the application server. This section will elaborate upon some of those architectures, and suggest some of the features that are necessary and available in WebSphere's Process Choreography support. These features make the creation of solutions which conform to the suggested architectures possible.

First, if we look back to our discussion of the business model earlier in this chapter, we will see that the premise was made that the basic elements of the business process are implemented with session beans. The simplest application of flows involves implementing that session bean using flow technology. What happens is that the basic steps necessary to execute a business process are scripted using a flow. This flow probably also involves conditionally calling and leveraging EJBs that are in the solution architecture. Most simple flows are generally short-lived in nature and are synchronously executed. These synchronous flows are called micro-flows or non-interruptible processes.

Secondly, we begin to see more complex business processes being represented by workflow. These processes involve a flow that choreographs or scripts the usage of a number of sub processes. These sub processes may also be non-interruptible processes or just traditionally created session beans. What is important is that the system supports the ability to conditionally execute various sub processes based on business rules. These more complex flows may also run for longer periods of time and are able to live beyond the lifetime of a given server instance. Activities in these flows often contain asynchronous invocations also. These are what would be characterized as macro-flows or interruptible processes. An interruptible process is shown in the Figure 2. It shows four major activities and suggests that each activity implementation is a session bean calling some entities. Obviously, actions that are more complex are allowed in the activity implementations. This will be further detailed later in the chapter. The smiling faces in the figure represent human-interaction as part of one or more of the activities performed within the business process.

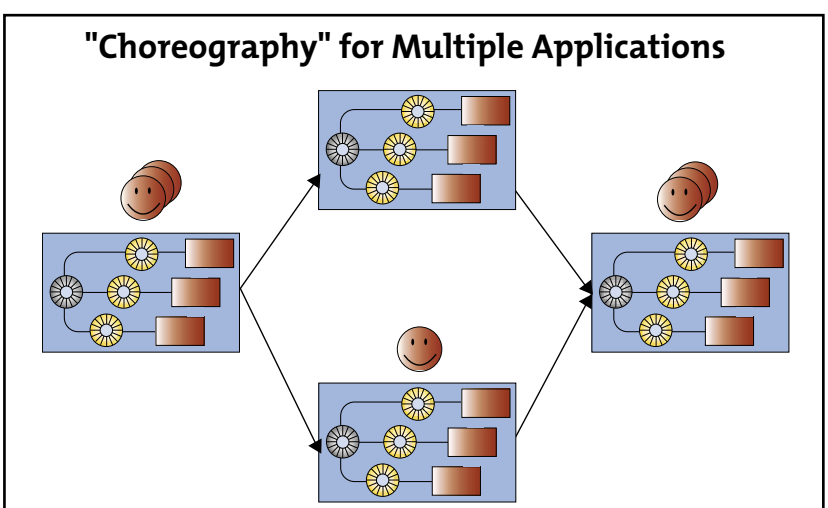


FIG. 2: AN INTERRUPTIBLE PROCESS

Next, we begin to see these basic short- and long-lived business processes taking on various additional roles, leveraging some different features along the way. It is possible that flows will have a series of user-interactions which are necessary to complete one or more of the steps in the business process. This requires workflows to play a more direct role in the presentation of the business process. The business process then is reacting directly to an ongoing stream of inputs driven by user interactions.

Some business processes will begin executing based on an inbound message rather than some direct invocation from within an existing servlet or session EJB. There is a role for flows in message-oriented applications. The flow continues to drive a series of steps that execute a business process. The difference is that the flow is started by an inbound message. In message-based flows, the results of a flow are also often transmitted with messages as well.

These basic patterns for flow usage begin to suggest some powerful ways to implement the processes that make up our business model. Beyond an individual organization, flows that span businesses or business units should also be considered. The same basic building blocks apply. A business process that spans businesses is called a public process. Each step in the public process may indeed involve utilizing some of the patterns we have just described.

With this set of possibilities in place, the details on the more basic elements that make up the business process programming model can be described. This is the topic of the next section.

Cache your connections

Using the Unshareable DataSource in WebSphere Application Server 5.0

BY JIAN TANG

In Java 2 Enterprise Edition (J2EE) 1.3, a new concept of a resource sharing scope has been introduced. A resource sharing scope can be either shareable or unshareable, the default being shareable. If the resource is configured as shareable, applications can share connections to this resource; if the resource is unshareable, connections will not be shared. Of all resources, the datasource is the most widely used. This article focuses on how to use unshareable datasources in WebSphere Application Server (WAS) 5.0.

ABOUT THE AUTHOR

Jian Tang works in IBM's WebSphere Application Server Connectors, JDBC, and Resource Adapters department. His main responsibility involves writing J2EE applications for function verification testing. He holds a master's degree in computer science from the University of Nebraska-Lincoln.

E-MAIL

jian.tang@us.ibm.com

A resource is configured as shareable by setting <res-sharing-scope> of the deployment descriptor to "Shareable". If a datasource is configured as unshareable in WAS 5.0 by setting <res-sharing-scope> to "Unshareable", every getConnection() call will access a connection handle with a different physical connection, even within the same sharing scope (I will discuss this in greater detail later). There is a one-to-one relationship between the connection handle and the physical connection to the datasource. In other words, there is an exclusive physical connection for each connection handle's use. If a datasource is configured as shareable, multiple getConnection() calls may access different connection handles with the same physical connection to the datasource in the same sharing scope. In other words, these connection handles may share the same physical connection.

There are several reasons for using unshareable connections. The first is to isolate unexpected behavior, for example, when you don't want other Enterprise JavaBeans (EJB) to access an EJB's physical connection, in order to avoid unexpected behavior related to the connection. The second reason is that some applications may need to make changes to connection attributes that are not allowed for shareable connections. One example of these attributes is the readOnly attribute. Other examples include character settings, localization configurations, etc. If applications fall into any of these categories, unshareable connections should be used.

Since all connections are shared in WAS 4.0, some programming models that worked in WAS 4.0 might not work without modification in WAS 5.0 if these applications use unshareable data sources. In this article, I will discuss which models work and which won't work when unshareable data sources are used.

Problems with Unshareable Connections

Before I go into detail about sharing connections, let me first clarify what a sharing boundary is. It's a boundary within which a physical connection can be shared by multiple connection handles. In other words, for two or more connection handles to share the same physical connection, they must be within the same sharing boundary. If getConnection() is called twice using the same datasource within the same sharing boundary and their connection sharing attributes are equal, the two connections are eligible to share one physical connection. Based on the resource sharing scope value, the runtime decides whether or not to share the physical connection.

A sharing boundary can be a global transaction (either a user transaction or a container-managed transaction), or a Local Transaction Containment (LTC). Notice that an LTC is different from a Resource Manager Local Transaction (RMLT), which is a resource's view of a local transaction. An LTC is a scope in which multiple RMLTs may be accessed in WAS 5.0. An example of an LTC is an execution of a bean method with the transaction attribute NotSupported, while setting a connection's autocommit attribute to false and later committing the connection is an example of an RMLT.

To better understand the case in which a data source is unshareable, assume we have two getConnection() calls to the same datasource within a sharing boundary. If the first connection is not closed before the second getConnection() is called, the second call will definitely get a different physical connection. What if the first connection is closed before the second getConnection() is called? Will the sec-

ond connection have a different physical connection from the first connection, or will they have the same physical connection? Notice that LTC and global transactions handle this differently. In LTC, when the first connection is closed and there is no outstanding work associated with this connection (either this connection never did any work, or the work has been committed or rolled back), the physical connection is released back to the connection pool. When the second getConnection() is called, a free connection is picked up from the connection pool. Therefore, the same physical connection may be used to service the request. However, in global transactions, when the first connection is closed, the physical connection is not released since the global transaction has not completed. The second getConnection() will definitely access a different physical connection.

Therefore, if a datasource is configured as unshareable, connections will not be shared within a sharing boundary. This means some applications that work in WAS 4.0 won't work in WAS 5.0 if they use unshareable datasources. The following code is an example of an application that might not work with an unshareable datasource.

```
userTransaction.begin();
java.sql.Connection conn1 =
datasource.getConnection();
java.sql.Statement stmt1 =
conn1.createStatement();
stmt1.execute("....");
...
java.sql.conenction conn2 =
datasource.getConnection();
java.sql.Statement stmt2 =
conn2.createStatement();
Stmt2.execute("....");
...
userTransaction.commit();
```

This code works in WAS 4.0 without any problem, because all the datasources in WAS 4.0 are shareable. In this example, there is no difference between the connection attributes, and two instances of getConnection() are called within one sharing boundary, in this case a user transaction. Therefore, two connection handles, conn1 and conn2, will share the same physical connection. Although from the user's point of view, there are two connections, conn1 and conn2, doing different work, they are actually using the same physical connection to talk to the datasource.

If the data source is configured as unshareable and the same code is executed in WAS 5.0, connection handles conn1 and conn2 will have different physical connections. The application may fail in one of the two following scenarios:

1. If the datasource does not support XA transaction, i.e., the datasource is a non-XA-capable datasource, such as an implementation of java.sql. ConnectionPoolDataSource, the application will fail because two non-XA-capable resources are not allowed to be enlisted in one global transaction. This will cause WAS 5.0 to throw a SQLException that includes "javax.resources.ResourceException: enlist: caught exception". There will also be a more detailed exception message in the log file: "Method enlist caught java.lang. IllegalStateException [specific exception stack trace] while trying to enlist resources from datasource [datasource JNDI name] with the Transaction Manager for the current transaction, and threw an Exception."

2. Even if the datasource supports XA transaction, if stmt1 tries to insert a record and stmt2 tries to update the same record, it will lead to database deadlock in some circumstances, for example, when the connection isolation level is TRANSACTION_REPEATABLE_ READ. In this case, you can either see an exception thrown from the database back end indicating a deadlock, or a javax. transaction. TransactionRolledBackException after the transaction is timed out.

You should pay careful attention to your applications if you want to use unshareable datasources. Watch out for applications that have the following two characteristics: (1) an instance of global transaction; (2) two or more getConnection() calls within this global transaction.

Existing Programming Models

WAS 5.0 supports all J2EE 1.3 application components. They are:

- ***Servlets:*** Servlets support both global transactions and LTC. In WAS 5.0, all servlets run by default within an LTC boundary. Servlets can look up "java:comp/UserTransaction" to get a javax.transaction.UserTransaction object. Servlets can also look up datasources and call getConnection() to the datasource multiple times in one global transaction. Therefore, some servlets that work in WAS 4.0 might not work well in WAS 5.0 if the datasource is unshareable.
- ***EJBs:*** EJBs that access datasources include session beans, bean-managed persistence (BMP) beans, and container-managed persistence (CMP) beans. All of these EJBs can run within a user transaction or a container-managed transaction if the transaction attribute is set to Required, RequiresNew, Mandatory, or Supports. In addition, bean-managed transaction (BMT) session beans can also directly access the javax.transaction.UserTransaction object from SessionContext. All of these EJBs can directly look up data sources. With CMP beans, persistence is managed by the EJB container, so CMP bean developers usually don't write database access codes into the methods of a bean class. Therefore, I will only discuss session beans and BMP beans later in the article.
- ***Other application clients:*** Other application clients include J2EE application clients, applets, thin clients, etc. Since application clients don't have access to "java:comp/UserTransaction", there is no way to start a user transaction. It is true that these client applications can still start global transactions by calling EJBs, which could potentially cause the multiple-enlistment problem or deadlock problem I discussed before. However, it is the EJB's code that we need to fix, not the client's code.

For servlets and session beans, there are situations in which applications that work in WAS 4.0 won't work in WAS 5.0. The example shown earlier is a typical one.

For BMP beans, if the datasource is configured as unshareable, it is possible that programming models that work in WAS 4.0 won't work in 5.0. To clarify these situations, let's have a look at some programming models that may not work in WAS 5.0.

Many BMP bean developers use the get/use/close model in every method. According to this model, in every business method, such as ejbLoad(), ejbcreate(), and ejbStore(), to name a few, connections are used in the following way:

1. A new connection is acquired from the datasource at the beginning of the method.
2. This connection is used to access or update the database.
3. At the end of the method, usually in the final block, this connection is closed.

This model has many advantages; it is easy to write, understand, and maintain. In fact, if the datasource is always shareable, this get/use/close model will work perfectly. However, if the datasource is unshareable, we will have the same problem as with the application shown earlier. Consider creating a BMP bean with the transaction attribute set to Required in a user transaction. As you know, creating a BMP bean requires two methods, ejbCreate() and ejbStore(), to be called in this transaction. If the get/use/close model is used, it will end up trying to access two physical connections within one global transaction. If the datasource is non-XA-capable, WAS 5.0 won't let the second connection's resource be enlisted in the transaction. Even if the datasource supports XA, there might be a deadlock, since usually ejbStore() updates the record created in ejbCreate().

Recommended Programming Models

In order to avoid these problems, I recommend using the following programming models and tips when the datasource is configured as unshareable. The goal of the following programming models is to cache the connection so that only one physical connection will be used in one global transaction.

SERVLETS

The recommended model for using unshareable connections with servlets is very straightforward. Avoid using multiple connections within a global transaction. If possible, applications should use one connection handle across the global transaction. However, sometimes servlets have to make more than one getConnection() call within a global transaction. For example, the servlet needs to access two different databases within a global transaction. In that case, make sure the datasources support XA transaction.

SESSION BEANS

As with servlets, using multiple connections within a global transaction is discouraged. If you must do this, make sure the datasource is XA-capable and that there won't be any deadlock. Session beans can be executed in a container-managed transaction as well as a user transaction. Therefore, be aware of the consequences when you specify the transaction attribute for session beans.

A session bean can be stateful or stateless. Stateful session beans can contain conversational states, while stateless session beans cannot. Since stateless session beans cannot contain conversational states, you should not cache the connection in the bean class. Therefore, we can follow the get/use/close connection usage pattern in every method. Session bean users should be aware that there is a getConnection() called in every method of a stateless session bean if that method accesses a datasource. If you try to group several methods in one global transaction, you should be aware of the potential problems.

For stateful session beans, you can cache the connection in the session bean class. You can get the connection in the setEntityContext() method and close the connection in ejbRemove(). Therefore, all methods in this stateful session bean use the same connection if the session bean accesses only one datasource. However, in this case you should call ejbRemove() after you finish using the session bean. Otherwise, the connection will be leaked in the system.

There can be some improvements on the stateful session bean to avoid this problem. A container-managed transaction stateful session bean can implement the javax.ejb.SessionSynchronization interface. With this interface, "the afterBegin notification signals a session bean instance that a new transaction has begun. The container invokes this method before the first business method within a transaction (which is not necessarily at the beginning of the transaction). The afterCompletion notification signals that the current transaction has completed," according to the Enterprise JavaBeans 2.0 Specification. Therefore, you can get the connection in the afterBegin() method and close it in the afterCompletion(boolean) method so you won't have any connection-leaking problems. Similarly, if your stateful session bean is a bean-managed transaction session bean, the bean can get the connection after the user transaction is started and close the connection before the user transaction is committed or rolled back.

For these reasons, the recommended connection usage pattern for session beans is as described in Table 1.

STATEFUL	Cache the connection. Get the connection in *setEntityContext()* and close it in *ejbRemove()*	
	Container-managed transaction session beans that implement *SessionSynchronization*	Cache the connection. Get the connection in *afterBegin(), and close it in afterCompletion (boolean)*
	Bean-managed transaction	Cache the connection. Get the connection after starting user transaction and close it before committing or rolling back user transaction session beans
STATELESS	Follow get/use/close connection usage pattern in every method	

TABLE 1: PROGRAMMING MODELS FOR SESSION BEANS

BMP BEANS

The goal of the model for using unshareable connections in BMP beans is the same as with the other models: to cache the connection across the BMP life cycle. Therefore, all the methods called in one global transaction will use this cached connection.

The simplest way to cache the connection handle in BMP is to get the connection in the setEntityContext() method and close it in unsetEntity Context(). This guarantees that one connection is used across this bean's life cycle. If in some business methods you want to access another datasource, you can simply get another connection. However, you should make sure both datasources support XA transaction. If this BMP bean accesses only one datasource, a non-XA-capable datasource can be used.

The EJB specification states that setEntityContext() and unsetEntityContext() can only access EntityContext methods [getEJBHome() and getEJBLocalHome()], and JNDI access to java:comp/env. It also says, "if an entity bean instance attempts to access a resource manager or an enterprise bean, and the access is not allowed, the behavior is undefined by the EJB architecture." Fortunately, WAS 5.0 allows you to access a resource manager in all methods. Therefore, we don't have any problem getting the connection in setEntityContext() and closing it in unsetEntityContext().

The programming models recommended above are also applicable when the datasource is configured as shareable. In fact, caching connections to avoid calling getConnection() several times can improve performance. Therefore, if you require high performance from your applications, you should cache connections.

Summary

In this article, I discussed some scenarios demonstrating why applications that work in WAS 4.0 won't work without modification in WAS 5.0 for an unshareable datasource. When the datasource is unshareable, every getConnection() will return a different physical connection, which leads to invalid enlistment and potential deadlocks. In order to avoid these problems, I recommended programming models for different applications, such as servlets, session beans, and BMP beans. These recommended programming models minimize problems by caching connections. This article focuses on datasources, but the principle applies to other resources as well.

References:

Java 2 Enterprise Edition 1.3 Specification: http://jcp.org/en/jsr/detail?id=58

Enterprise JavaBeans 2.0 Specification: www.jcp.org/aboutJava/communityprocess/maintenance/jsr019/EJBAccepted.html

Verifying Web Site Accessibility

Methods for ensuring dynamic sites work correctly in a variety of contexts

BY CYNTHIA DUNLOP AND ARTHUR HICKEN

ABOUT THE AUTHOR

As director of business development for Parasoft, Arthur Hicken works with the executive management team to influence the strategic direction of product and business development. In addition, he has been responsible for managing the Parasoft corporate Web site since its inception. The Web site includes online purchasing and product evaluation capabilities as well as a consumer-based Internet site for the wireless Web.

E-MAIL

ahicken@parasoft.com

Are you confident that your site works correctly for someone using a text-to-speech browser that vocalizes Web page text? For someone using a device that instantly translates Web page text into Braille? For someone using a special device or software (such as voice recognition software) for navigation and input?

If not, you are taking two serious risks:

- People with disabilities (approximately 19% of the U.S. population) might not be able to access your site.
- If the site is produced for or by a U.S. government agency, you might be violating a federal law and could lose government contracts and/or attract costly lawsuits.

You can make your site accessible to people with disabilities (and thus mitigate these risks) by following the guidelines the U.S. government and the World Wide Web Consortium (W3C) have established for making Web sites accessible in a wide variety of contexts. Ensuring the accessibility of a static site or even a simple dynamic site is fairly straightforward – you automatically verify whether the pages comply with many accessibility guidelines, then manually verify compliance with the remaining guidelines. However, verifying the accessibility of the highly dynamic sites that WebSphere developers typically create is more challenging. If the pages change depending on variables such as username/password, user preferences, click paths, time, and so forth, how do you verify that the site will be accessible to every user all of the time? This article attempts to provide a practical solution to those challenges.

Understanding the Standards

You can ensure a basic level of accessibility by following accessibility guidelines during design and implementation. There are two main sets of Web site accessibility guidelines: the U.S. government's Section 508 Web Standards and the Web Accessibility Initiative's Web Content Accessibility Guidelines. Both sets of guidelines generally prompt you to follow design and navigation best practices and add optional or redundant elements to ensure that the site functions on a variety of different devices and settings. In fact, following these guidelines not only makes a site accessible to people with disabilities, but also makes it more accessible to users on desktop systems, PDAs, Web-enabled phones, kiosks, and other existing and upcoming internet devices.

The Section 508 amendment to the Rehabilitation Act requires that any technology produced by or for U.S. government agencies must be accessible to people with disabilities. Section 508 Web Standards detail 16 specific requirements that a Web site must satisfy to comply with Section 508 legislation. Most of these guidelines focus on ensuring graceful page transformations to a variety of devices and making content understandable and navigable on a variety of devices. Although these guidelines were designed specifically for federal agencies and vendors producing technology for federal agencies, they have become the standard that many Web developers use to gauge site accessibility. These guidelines are listed in the Section 508 Guidelines sidebar.

The Web Accessibility Initiative (WAI), a W3C organization committed to making the Web more accessible for people with disabilities, developed the Web Content Accessibility Guidelines. These guidelines are similar to Section 508 guidelines because Section 508 guidelines were largely based on the WAI guidelines. These guidelines are listed in the WAI Guidelines sidebar.

Accessibility Testing Fundamentals

You can automatically verify compliance with the majority of the Section 508 and WAI guidelines by using an automatic verification tool. However, while accessibility testing tools can be a tremendous aid in verifying accessibility, a gold stamp from even the best verification tool does not mean that the site is actually accessible. Some degree of human inspection is always required to fully guarantee compliance because the criteria for some guidelines are too subjective and/or complex for today's systems to evaluate. For example, machines cannot automatically verify WAI Guideline 13. ("Provide clear and consistent navigation mechanisms – orientation information, navigation bars, a site map, etc. – to increase the likelihood that a person will find what they are looking for at a site.")

For both automatic and manual verification, you can save yourself considerable time and grief by starting your testing early in the development process – before accessibility problems have the opportunity to multiply as the result of code reuse or interactions.

Automatic Inspection

The W3C maintains a list of tools that automate many aspects of accessibility verification; this list is available at www.w3.org/WAI/ER/existingtools.html.

The automatic verification process is simple – provided that your tool can correctly access the versions of site pages that will be sent to client devices (we discuss specific strategies for loading and testing dynamic pages in the next section). Assuming that the verification tool you select has a way to access the various pages you need to test, you can verify compliance with many accessibility guidelines by telling the tool how to find those pages, and then clicking the Start button. The tool will then request the specified pages, load the code delivered in the responses, and then analyze the code to determine whether it violates any of the Section 508 and/or WAI accessibility guidelines it is configured to enforce. After each accessibility test, the tool will report code that violates the selected accessibility rules. Messages vary by tool, but typically include a description of the violated guideline, as well as the filename and line number in which the violation occurs.

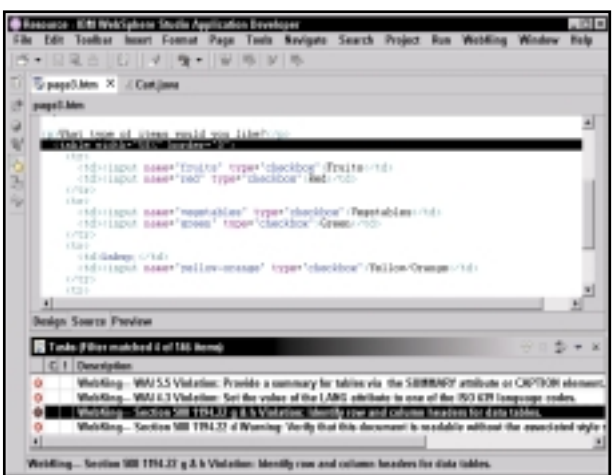


FIG. 1: VERIFYING SECTION 508 COMPLIANCE WITHIN WSAD

WSAD developers can simplify the accessibility verification and improvement process by selecting an accessibility verification tool that integrates into the WSAD environment. This integration offers two main benefits:

- It allows you to test the entire project (or each new page or project element) without manually loading and configuring it in a different tool.
- It expedites the correction process as well as part of the manual verification process. Double-clicking accessibility error or warning messages in the Task panel prompts WSAD to open the related source file in the Page Designer and place the cursor at the line containing the violation. You can then inspect and modify the source code in the Source mode (as shown in Figure 1), or switch to the Design mode to make modifications. Also, you can instantly render a page containing a definite or possible violation by switching to the Preview mode, and then inspect it without leaving WSAD.

If you want to guarantee that new code is verified immediately after it is added, you can configure your tool to automatically perform accessibility testing daily. Setting up such an automatic testing infrastructure typically involves writing a simple command-line script that runs the test, then configuring a cron job or other scheduling program to run this test at the specified time each day.

Manual Inspection

As mentioned in the introduction to this section, manual inspection is required to check compliance with some guidelines, and no site can be deemed "accessible" without some degree of manual inspection. For example, tools cannot make the subjective analysis required to determine whether a page is readable when rendered without a stylesheet or whether the site's navigation aids are clear and consistent. However, tools can help you pinpoint some of the elements that require human inspection.

Most accessibility testing tools can be configured to report occurrences of elements (such as links to stylesheets) that indicate that the related page requires human inspection and possibly additional modification. These warning messages relieve you from having to search for each of these elements.

In addition, some accessibility testing tools reduce the burden of manual inspection by automatically checking custom accessibility guidelines. For example, imagine that you discover a certain company logo used on your site does not provide the necessary contrast for someone with color deficits or someone viewing the site on a black and white screen. Instead of manually searching for all occurrences of this logo, you could automatically identify all pages that use this logo by designing and applying a custom guideline that detects any occurrence of this logo.

ABOUT THE AUTHOR

Cynthia Dunlop is a senior technical author for Parasoft. Since 1998, she has been responsible for crafting Parasoft product manuals and technical papers. Cynthia has also coauthored *Bulletproofing Web Applications* (John Wiley and Sons, 2001) and numerous technical articles. Prior to joining Parasoft, she worked as a writing instructor at Washington State University. Cynthia holds an MA in English from Washington State University and a BA in English from UCLA.

E-MAIL
cynthia@parasoft.com

Verifying Dynamic Sites

Accessibility verification is more challenging if your site is truly dynamic (e.g., pages change according to user preferences, variables, time, inputs, and so on) and your verification tool cannot reach all possible page instances by following links or crawling the site. However, dynamic sites are by no means exempt from accessibility standards.

Section 508 Guidelines

Here is a list of the Web-related guidelines in Section 1194.22 of Section 508:

a) A text equivalent for every nontext element shall be provided (e.g., via "alt", "longdesc", or in element content).

b) Equivalent alternatives for any multimedia presentation shall be synchronized with the presentation.

c) Web pages shall be designed so that all information conveyed with color is also available without color, for example from context or markup.

d) Documents shall be organized so they are readable without requiring an associated stylesheet.

e) Redundant text links shall be provided for each active region of a server-side image map.

f) Client-side image maps shall be provided instead of server-side image maps, except where the regions cannot be defined with an available geometric shape.

g) Row and column headers shall be identified for data tables.

h) Markup shall be used to associate data cells and header cells for data tables that have two or more logical levels of row or column headers.

i) Frames shall be titled with text that facilitates frame identification and navigation.

j) Pages shall be designed to avoid causing the screen to flicker, with a frequency greater than 2Hz and lower than 55Hz.

k) A text-only page, with equivalent information or functionality, shall be provided to make a Web site comply with the provisions of this part, when compliance cannot be accomplished in any other way. The content of the text-only page shall be updated whenever the primary page changes.

l) When pages utilize scripting languages to display content or to create interface elements, the information provided by the script shall be identified with functional text that can be read by assistive technology.

m) When a Web page requires that an applet, plug-in, or other application be present on the client system to interpret page content, the page must provide a link to a plug-in or applet that complies with §1194.21(a) through (l).

n) When electronic forms are designed to be completed online, the form shall allow people using assistive technology to access the information, field elements, and functionality required for completion and submission of the form, including all directions and cues.

o) A method shall be provided that permits users to skip repetitive navigation links.

p) When a timed response is required, the user shall be alerted and given sufficient time to indicate more time is required.

To view a more detailed explanation of these guidelines, visit www.section508.gov/index.cfm?FuseAction=Content&ID=12#Web

The W3C states:

Dynamically generated pages are usually assembled from one or more templates that provide common layout and navigation features, and content provided automatically from a database or other content management system. To achieve full conformance, the accessibility of both templates and generated content must be evaluated. It is not sufficient to evaluate only templates: content may also contain markup, or be required to contain markup in order to be accessible.…If all dynamic content cannot be evaluated, generate broadly representative samples, capture content, and test the output.

(More information is available at www.w3.org/WAI/eval.)

You can find and control many accessibility problems at their source by inspecting templates (with the tools discussed in the previous section) and content management system contents (if applicable), then fixing problems found before you generate and test any actual pages. However, since the final determination of accessibility is based on the code/content delivered to client devices, all testing of dynamic sites must ultimately verify the code/content delivered to a client. This means that if your e-commerce application's shopping cart page can be created a thousand times a day and can be created differently each time it is invoked, you need to verify that a "broadly representative" sample of those possible pages are accessible.

To verify a broadly representative sample of your dynamic pages, you must create and test a number of specific instances of the precise pages that your users are likely to encounter. However, this can be challenging. With most dynamic sites, specific state and input variables are required to invoke certain pages. For example, if you want to invoke and test a checkout page, you must first set valid username and password variables and have at least one item added to a shopping cart. If you want to test the pages returned for both valid and invalid logins, you must set the inputs necessary to create those pages.

Many accessibility testing tools do not allow you to specify the path or inputs required to invoke specific pages, and thus are unable to accurately verify many of the pages that might be sent to user devices. For example, say that you are building an e-commerce site and you want to verify the accessibility of a page built based on the user's past purchases and browsing patterns. Assume that this page will only display the customized information if a valid user login and password are submitted. If you submit the same URL to a tool that does not allow you to enter specific inputs, the tool will not be able to create the customized page, and will instead create and test a generic version of the page. This can be especially troublesome on sites that implement frames. Often, one frame might alter the contents of another frame; if your testing tool loads just the URL of one of the subframes (rather than the complete series of frames), your test results might be seriously skewed.

If you have a dynamic site and want to automate as much of the verification process as possible, look for a tool that can:

- Follow and verify specific click paths through the site, including paths that involve the submission of pass-

words or other specific user inputs. At the least, your verification should involve creating and testing several different versions of the possible click paths representing your use cases. For example, to verify the accessibility of the login functionality, you should submit both a valid and invalid username/password set. To verify the accessibility of the search results page, you should submit search criteria that returns zero results, a small set of results, and an extraordinarily large set of results. One way to streamline the path recording process is to use a tool that can create and test the actual user paths recorded in your log files. If creating paths through your site requires the submission of a large number of inputs, you can further streamline the process by using a testing tool that can create paths by reading inputs from a database or other data source.

- Provide virtual users that automatically create new paths through the site. For most sites, manually creating and testing all possible permutations of every possible path through the site simply is not feasible. Automatic virtual users provide a way to significantly increase the coverage of your testing with little additional effort. You will get the most mileage from intelligent virtual users that can randomly generate inputs for forms whose options you want to exercise, as well as submit specific inputs for forms that require them (such as login forms). Ideally, you want to flush as many possible combinations of page elements, contents, and configurations that could be sent to user devices, then check whether these pages are accessible.

If you are unable to access a tool that can help you with your specific verification needs, you have two main alternatives for accessibility verification:

- Invoke the possible user click paths manually, then inspect the pages created at each path step. If you click the same links and enter the same inputs as your users will, you should receive pages similar to those the users will encounter. However, if you are considering this method, be aware that 100% manual verification is typically considered tedious, time consuming, and error-prone.
- Enlist the help of an accessibility testing consultant. Before agreeing to a contract, consider how the site will be tested and how much of the site will be covered by the specified testing method. Also, ensure that the consultant will test all of your site modifications as they are implemented and that this testing will be comprehensive enough to verify whether a small change in one part of the application causes accessibility problems that might ripple throughout other parts of the application.

Conclusion

Verifying your site's accessibility is mandatory if you are producing technology for a government agency, and it is highly recommended if you want to ensure that your site does not exclude people with disabilities or people accessing your site in a variety of contexts. Automated tools can significantly facilitate the accessibility verification process for even very dynamic sites. However, it is important to realize that today's programs and systems are simply not intelligent enough to verify every accessibility requirement, so tools alone cannot guarantee a site's accessibility. By carefully designing your site, then performing both automated and manual accessibility inspections from the earliest possible phases of the development process, you should be able to efficiently guarantee that your site is accessible to everyone who wants to use it.

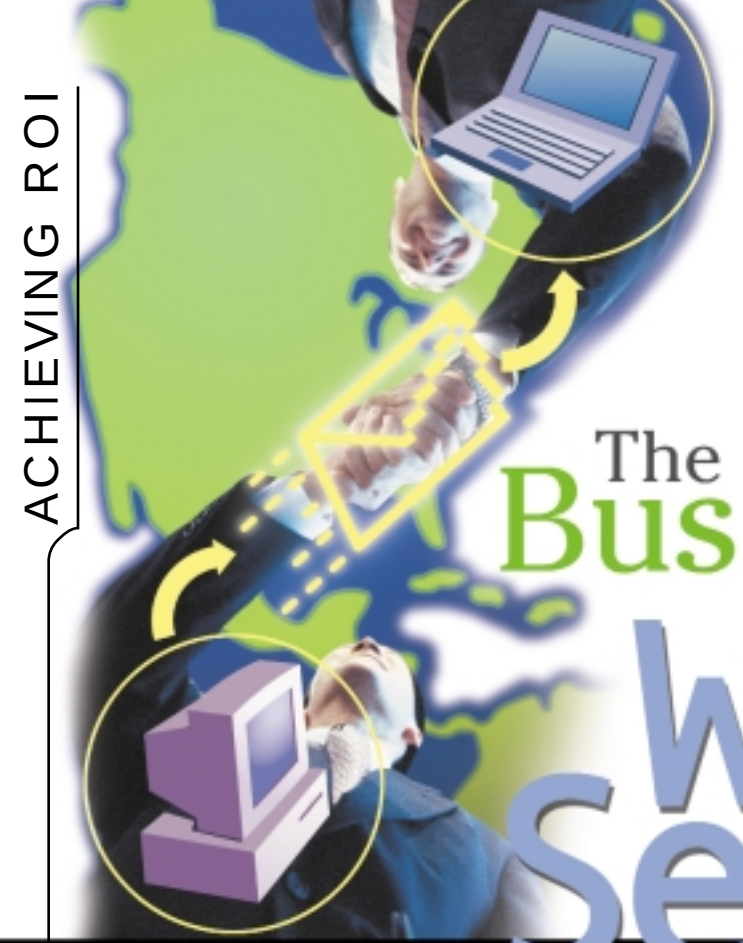
One way to facilitate the accessibility verification process is to test your site with Parasoft WebKing, which automatically identifies code that positively or possibly violates Section 508 and WAI guidelines. This tool is specifically designed for testing dynamic sites, but works equally well with static pages. To facilitate verification of dynamic sites, it automatically creates, follows, and verifies the most frequently accessed paths through the site (based on your log file) or any specific paths you record. In addition, WebKing automatically creates and verifies new paths through the site for increased testing coverage. Inputs for password pages and other forms can be specified prior to the test, or generated automatically. In addition to its accessibility verification features, the tool also automates functional testing, regression testing, and load testing to prevent and expose errors throughout the Web development process. WebKing has completed the WSAD integration process and has applied for Ready for WebSphere validation.

WAI Guidelines

Here is a list of the WAI Guidelines (each of which is divided into a number of specific "checkpoints"):

1. Provide equivalent alternatives to auditory and visual content.
2. Don't rely on color alone.
3. Use markup and stylesheets and do so properly.
4. Clarify natural language usage.
5. Create tables that transform gracefully.
6. Ensure that pages featuring new technologies transform gracefully.
7. Ensure user control of time-sensitive content changes.
8. Ensure direct accessibility of embedded user interfaces.
9. Design for device-independence.
10. Use interim solutions.
11. Use W3C technologies and guidelines.
12. Provide context and orientation information.
13. Provide clear navigation mechanisms.
14. Ensure that documents are clear and simple.

To view a more detailed list of these guidelines and their checkpoints, visit www.w3.org/TR/WCAG10



BY MARK COLAN

The Business Value of Web Services

Improving IT stability, agility, and flexibility

Web services are rapidly gaining traction in both developer mindshare and the overall technology marketplace. Companies like Amazon and Google have opened access to their back-end IT systems through Web services interfaces. IBM has many customers who have started to use Web services to reduce the cost of doing business and solve integration problems while gaining many other benefits that I'll discuss in this article.

If you haven't gotten started with Web services yet, perhaps it's because you've not yet become convinced of the business value of the technologies, or where you'll see a return on investment. In this article I'll focus on the advantages of Web services, how people are using them, and the technical advantages that lead to business value and ROI. I'll also look at two customer stories to demonstrate why many feel that now is the time to get started with Web services.

Cross-Platform Application Integration

Many companies have struggled for years with the limitations of the traditional means of integrating application systems – whether the systems are inside an enterprise or are used to connect business partners. In the absence of a standardized approach like Web services, integration can be difficult and very time consuming; moreover, the result might be unstable in the face of ongoing changes to software.

Integration with Web services addresses these problems using standard technologies such as XML and SOAP. XML provides a universal data representation that can be used with any programming language on any system. SOAP provides a standardized mechanism for process-level integration and the exchange of XML content for data integration, regardless of the underlying operating system. Platform-neutral Web services standards have become very popular for a flexible integration strategy, both between applications within an enterprise and between business partners.

When two computer systems are separated by manual, paper-oriented processes, automation using Web services can provide ROI (see, for example, the Storebrand case study later in this article). Sending updated data automatically – instead of reentering from computer-printed reports – eliminates errors and reduces staffing requirements. Using Web services technology instead of traditional programming integration techniques can substantially reduce the development cost and time for integrating applications, so that the benefits of the work can be realized sooner.

Flexibility and Stability

In traditional integration, programmers make direct calls through the Application Programming Interface (API) that they design for external use, and these calls are compiled directly into the programs. Most file formats use fixed positions for the data, and the file descriptions are often embedded into the programs.

Both APIs and file formats change because of the changing business requirements. When this happens, all programs that use them must be recompiled, retested, and redeployed. If you forget to tell a department or business partner about these changes, someone will get a telephone call in the mid-

dle of the night when the system has crashed – if you are lucky. If you're unlucky, the programs may appear to be working normally but produce incorrect results.

An important advantage of XML over a traditional fixed-file format is that programs become more tolerant to changes in the data. XML data is not positional like fixed-file formats, so changes in the content may not require recompilation. When new data is added to a traditional file format, programs may need to be recompiled; programs that use XML can be designed to ignore unexpected new information.

SOAP can replace the use of conventional APIs or file formats by using the XML message exchange to request actions in the other system, leading to a more stable integration. When conventional APIs change, they may not always require a change to the SOAP message content. It can be difficult and complex to support more than one version of an API or file format, and attempting to do so may introduce errors. SOAP's use of XML means you can easily add versioning information to support older messages temporarily, smoothing the transition to a new message format.

The use of SOAP messaging as an alternative to compiled-in APIs or file formats removes dependencies on technology assumptions between the service provider and the service requester, providing a clean separation between the two. For example, the service provider could replace a legacy implementation of the back-end system with a new version; if the same message format is used for both old and new code, the service requester may not require any changes, even in the extreme case where the back-end system is replaced with new code in a different programming language. We call this separation – inherent in SOAP messaging – "loose coupling" (see Figure 1).

Developers of the back-end functions focus on providing generic services for any kind of client, and can create new services without disrupting requesters of the older ones. New kinds of client applications (such as access via handheld devices or browsers) may be developed and deployed using the same service interface – despite potentially very different technology. A common service interface for a wide range of clients results in a much simpler infrastructure, compared to having different infrastructure code for each kind of client.

All of this adds up to a substantially more robust and reliable integration. Software will always change, reflecting evolving business requirements, but the loose coupling inherent in a SOAP message strategy greatly improves stability. The ROI follows from reduced costs for managing change.

A Unifying Architecture

Web services might be considered a success even if it improved only EAI and B2B integration. The real strength of Web services technology lies in the fact that it can be used for any kind of system-to-system integration.

There are many technologies for connecting handheld devices to our IT systems. Unfortunately, these technologies are often vendor-specific and vary from device to device, and they require special software on the IT side. Similarly, there are many techniques for providing back-end access from browser applications, but these techniques are different from those used by handhelds, or by program-to-program integration.

Web services is a simple and standard architecture that unifies the integration between any kinds of computers and software. You may start by creating Web services interfaces to connect two programs, but the same interfaces could also be used for supporting access from a browser application or a handheld computer. Your developers can be freed from the headaches of creating and maintaining low-level integration technologies, and focus instead on activities providing higher value to the organization. The use of Web services for handheld devices will really take off once those devices routinely have more memory and faster processors, as well as greater wireless communication bandwidth.

Code Reuse and Agility

For companies providing access to business functions with Web services, some forethought into the types and purposes of the functions will lead to code that can be written once but used for many applications. A company that systematically uses Web services to expose business functions will enjoy the advantages of code reuse, and the reused code will be integrated in more flexible and dynamic ways. The standardized interface to the software may be used in many applications for many types of computers, increasing the ROI for deployment of the service. With the new development tools that speed the integration of Web services into other applications, a company can more quickly provide software to support business functions, or experiment with new business models supported by quickly written applications based on internal services.

The proliferation of Web services in a company will require some management. How can a developer search for a useful business function that may have been written within the same company, yet in a different IT department on the other side of the world? When load balancing requires moving Web services to other servers, what happens to the applications that use those services?

The UDDI (Universal Description, Discovery, and Integration) Specification provides a solution to these problems. A UDDI service registry can be used to catalog and organize available Web services to make them easy to find, much in the way that a community's Yellow Pages directory makes it easy to find a doctor or plumber when you need one. Applications that use a Web service but find that it is no longer available on a known server can be designed to look in the UDDI registry to find the new location for the service, and adapt automatically, without requiring programmer intervention.

Leveraging Existing Software Investments

As exciting as the new Web services technology is, many companies will not want to discard all their existing software assets just to get the new features. Often a company depends on this older software to run its daily business-critical functions; bringing in an entirely new system can cause disruption that leads to loss of customers. Fortunately, Web services does not require a "rip and replace" approach.

It is possible to develop Web services interfaces to connect to legacy applications. A small amount of code can be written to handle incoming SOAP messages, and to call the internal entry points of the existing code as required to handle the request. This thin layer of SOAP support then protects the underlying code by controlling access from outside, while simplifying the effort to integrate by hiding the arcane technical details of the implementation in favor of a universal message interface. Some analysts have referred to this as providing "multichannel access" to the application.

ABOUT THE AUTHOR

Mark Colan is senior e-business evangelist for IBM Corporation. He gives technical, keynote, and customer presentations on Web services and XML technologies and strategy, and has spoken at most XML conferences from 2000 through 2002, as well as JavaOne '98 and '99. PDFs of Mark's current presentations can be downloaded from http://ibm.com/developerWorks/speakers/colan.

Of course, one of the goals of Web services is to reduce the amount of new code you need to write and maintain. IBM is building adapters to allow access to older software systems, such as CICS and IMS, which are still in use by our customers. Use of these adapters will make it fast, easy, and reliable to expose more business functions as Web services.

IBM is building Web services interfaces into their products. With Version 8 of the IBM DB2 Universal Database, for example, you can easily create Web services that can query, update, or run stored procedures for data in any tables of your database, without writing code and under control of the administrator. IBM believes that Web services provide a rapid-integration mechanism, and we will support the technologies aggressively across our software product line.

Leveraging Existing Skill Sets

Distributed computing – connecting computer systems together – is a difficult programming problem that traditionally requires highly skilled developers, often working together to debug both sides of a connection at once. SOAP by itself does not address this problem. Creating a requester application "by hand" requires writing code to create request messages, send them across the network, get the response, parse the response from XML, and convert the data into the types needed for the program.

WSDL, the Web Services Description Language, simplifies the integration of Web services. The service provider can create a WSDL document that specifies the complete technical requirements needed for integration. Because WSDL is an XML vocabulary, the WSDL document is machine-readable. Because Web services can be accessed from many programming languages, the WSDL document expresses details of the message format, the types of information, and so on, in a form that is independent of any particular programming language.

The WSDL description can be used by application development environments such as IBM's WebSphere Studio Application Developer (WSAD). WSAD makes it easy to create service requester applications that use Web services by automatically generating Java code that connects to the service. The generated code looks like a local Java class that exposes methods with Java return values, but handles all of the details of the network connection, construction of the request message, and converting the response to native Java types. The generated code looks like a local class implementation, yet it handles all of the technical complexity inherent in remote access. Because it is custom generated for the service from its WSDL description and does not require adding or changing code to make it work, it eliminates complex distributed debugging between the service requester and service provider.

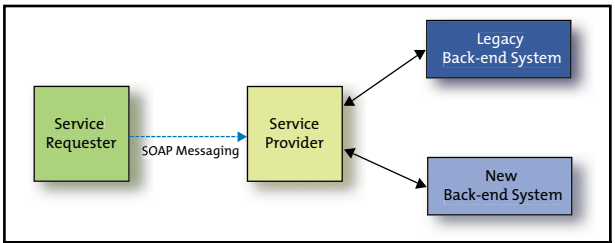


FIG. 1: IN LOOSE COUPLING, SOAP PROVIDES A CLEAN SEPARATION BETWEEN THE SERVICE REQUESTER AND THE SERVICE PROVIDER

Less-experienced developers can thus quickly become productive in building applications that use Web services. Instead of learning all the arcane details required to talk to a remote server, developers can focus on the work they already know best, building applications that use the services. By using the Web Services Invocation Framework (WSIF), an open-source technology available from http://xml.apache.org, developers can write code that supports SOAP but can select other available bindings at runtime based on quality of service requirements.

Deploying a Web service using traditional hand-coding techniques is possible, but there are tools in WebSphere Studio Application Developer to speed this work as well. Much of the WSDL description can be generated automatically, especially if you are starting with Java code. The server code can be installed and configured in a standard way, without requiring a system guru who is familiar with all the arcane details of multiple configuration files, special places to copy files, system environment variables, and the like.

Both SOAP and WSDL were coauthored by IBM and are now being standardized in the World Wide Web Consortium.

Future Uses of Web Services

Although use of Web services focuses on today's IT integration problems, the same technologies can be used for communication between computers across a network for other technologies.

IBM is implementing and driving the standardization of grid computing technology to leverage existing computing resources and provide sophisticated allocation of computing resources on demand. Our hardware and software products are gaining new features to intelligently and automatically detect and manage problems. We call this approach autonomic computing because of its similarity to the human autonomic nervous system's automatic regulation of our bodies. Both grid computing and autonomic computing use Web services to coordinate their work.

New technologies will be invented that require network communications and exchange, and Web services should be ideal for these purposes. If instant messaging had been developed using Web services, we could have had a common mechanism for messaging between computers, between mobile phones and other devices – instead of only between computers or only between telephones, the way it is today. An open instant messaging standard built on Web services could solve this problem now.

Up to now, most of what I have talked about has involved one Web service being invoked by one or more other pieces of software. What if we need to string together a sequence of Web services, perhaps with some logic mixed in, to accomplish a more sophisticated business task?

The newest specifications for Web services support business process modeling and description. BPEL4WS (Business Process and Execution Language for Web Services) defines an XML vocabulary for describing business processes between many cooperating business entities – either independent companies or private processes in one partner company. The partners, their responsibilities, the path followed for handling a service request among multiple partners, and a consistent means of managing outcomes and handling exceptions are all parts of the specification.

Because BPEL4WS is written as an XML vocabulary, it

becomes less dependent on the software that uses it. Business partners could cooperate using a common BPEL4WS description even though the software they use comes from different vendors. BPEL4WS should allow many opportunities for innovation as companies move from ad hoc e-business connections to real cross-enterprise implementations of workflow and business processes. As the business processes evolve, it might be possible to make changes such as adding new partners simply by updating the descriptions and making it available to all partners.

Case Study: Flexibility in Cross-Platform EAI

The IBM jStart team worked with a leading wholesale financial services company, implementing Web services to simplify the company's strategy for supporting internal applications. The client applications in this case were built using Microsoft Word templates to access back-end investment banking data on demand by calling Word macros.

While the client application uses Microsoft technologies, the back-end functions were built with J2EE on IBM WebSphere Application Server and Sybase (with an ongoing project to migrate from Sybase to IBM DB2 Universal Database). The solution had to work with Sybase at rollout and also be able to migrate to DB2 without any client-side code changes.

The ROI is difficult to measure in this case because it does not improve on an existing implementation that we could use for comparison. Still, there are important advantages to this approach that make Web services valuable for this solution.

The use of SOAP messaging to loosely couple the client application to the back end is key to the flexibility that allows one database back end to be replaced with another, without redeploying the client code. By keeping the message format the same, it is possible to change the back-end service for the messages without affecting the client application. SOAP messaging also enables flexible cross-platform interoperability between the client application (Microsoft technology in this example) and the middle tier (J2EE on IBM WebSphere). Finally, the use of Web services for accessing back-end functions means it will be less expensive to develop client access using other technologies, such as browser applications or handheld access, because these kinds of clients can reuse the same Web services.

Case Study: Improved Business Partner Integration

Storebrand ASA has created a company pension plan that allows 6,500 small and mid-size companies in Norway to provide pension benefits to their employees. Naturally, Storebrand requires accurate and current information for the people they insure. The problem is that the current information is maintained on payroll systems in all those partner companies. Because of the difficulty of integrating these payroll systems to the Storebrand infrastructure with traditional programming techniques, the company employed a team of 50 people to handle telephone calls, faxes, e-mails, and printed reports from the payroll systems to list changes that needed to be entered manually.

The IBM jStart team helped Storebrand build an automatic means of synchronizing the information from the payroll systems with the Storebrand databases, built on an IBM infrastructure. The pilot study involved the integration of one popular payroll system, which happened to be written in the Delphi programming language and ran on a Microsoft Windows platform. This was done by developing a COM object using the Microsoft SOAP Toolkit, and installing it into the payroll system so that updates could be sent electronically over the Internet. The personal information about the employees is protected for transmission across the Internet using HTTPS transport-level encryption and client-side authentication via SmartCard. This is but one choice for the client; in the future, clients built using Java could be used because the Web services technology is based on open standards.

Storebrand expects to save up to 10,000 staff hours per year as they automate the synchronizing process with other payroll systems. Eventually, the company should be able to reduce and redeploy the team that handles manual updates, saving substantial staffing costs. Less quantifiable, yet easily as important, Storebrand now has information that is more accurate and more timely, making it a more effective insurer.

In this case we see the value of Web services for automating two computer systems that were formerly separated by manual methods. Web services also provides cross-platform integration between clients using Microsoft technology and a back end based on IBM software.

Conclusion

The Web services architecture unifies connections between all kinds of computing devices using a straightforward, platform-neutral, and standard way of representing information and requesting actions, thereby providing both data- and process-level integration capabilities. While many of the current projects involve integration inside the enterprise or between business partners, the technologies can also be used to extend the reach of IT systems to handheld devices, browser applications, and future applications, all using the same interfaces.

A good way to get started with Web services is with a pilot study. This will make it easier for you to measure the return on investment for your own environment. For example, look for a case in which two (or, even better, three) computer applications are separated by manual, paper-oriented processes, and consider using Web services to automate the connection. There are probably many opportunities to reduce the cost of doing business and improve the quality of information upon which you base your business, but Web services technology becomes even more valuable when the business services can be reused by other applications.

Web services simplifies your infrastructure and can result in less code to write and maintain, while improving IT stability, agility, and flexibility. Web services can make a big heterogeneous world look much more homogeneous. Now is the time to give Web services a serious look and see how it can benefit your company and save money.

Resources

Web Services Invocation Framework:
http://www.xml.apache.org
Current work in standardizing SOAP at W3C:
www.w3.org/2000/xp/Group
Web Services Toolkit for Mobile Devices:
www.alphaworks.ibm.com/tech/wstkmd
Details on the EAI case study in this paper:
www.ibm.com/developerworks/library/ws-best3

WEBSPHERE NEWS

VISUAL SLICKEDIT PLUG-IN ANNOUNCED

(Morrisville, NC) – SlickEdit, Inc., provider of the award-winning multilanguage, multiplatform editing environment, announced Visual SlickEdit for WebSphere Studio, a plug-in editor for both IBM WebSphere Studio and the Eclipse open tools framework. With this product, OEMs and end-user developers using IBM WebSphere Studio Application Developer or the Eclipse Workbench can choose Visual SlickEdit as the default editor to extend the functionality of these integrated development environments (IDEs).

Visual SlickEdit Plug-In for WebSphere Studio takes advantage of the open architecture of Eclipse and allows developers to use the popular Visual SlickEdit editing environment as the default editor within the Eclipse IDE and the WebSphere Studio IDE. Visual SlickEdit includes a rich and innovative set of multilanguage editing capabilities that fosters increased productivity.

www.slickedit.com

IBM UNVEILS NEW VOICE PORTAL TECHNOLOGY

(Somers, NY) – IBM unveiled its new WebSphere Voice Application Access product: middleware that simplifies building and managing voice portals and that more easily extends Web-based portals to voice. Leveraging the scalability, personalization, and authentication features of IBM's WebSphere Portal, it enables mobile workers to more easily access information from multiple voice applications – using a single telephone number.

This new offering includes IBM's WebSphere Voice Server as well as ready-to-use e-mail, personal information management (PIM) functions, and sample portlets. It also supports VoiceXML and Java – including development tools based on Eclipse, the open-source, vendor-neutral platform for writing software – and uses open-standard programming languages to create voice-enabled applications that will interoperate with a range of Web servers and databases.

In keeping with IBM's strategy to provide solutions across multiple platforms, IBM will be working to make WebSphere Voice Application Access interoperable with offerings from third-party VoiceXML vendors, such as Nuance and Cisco. In addition, IBM is also working with independent solutions vendors, including V-Enable, Voxsurf, and Viecore, to extend their current solutions.

www.ibm.com

NEON SYSTEMS ADAPTERS ACHIEVE A WEBSPHERE FIRST

(Sugar Land, TX) – NEON Systems, Inc., a leading provider of software to "liberate the mainframe," announced that the company's Shadow JDBC Adapter for mainframe integration is the first JDBC adapter to pass the innovative WebSphere self-testing process and will be added by IBM to its "Self-Tested Software" support page.

WebSphere Self-Testing is a new program sponsored by IBM that facilitates self-testing of WebSphere complementary technologies through an IBM-endorsed testing process. The ability for third-party vendors to self-test with the WebSphere Application Server helps provide customers with a wide array of access capabilities and the broadest choice of technologies to deliver their applications while minimizing potential problems. NEON Systems' Shadow is the leading software technology for application platform suite to mainframe integration.

www.neonsys.com

INSTANTIATIONS CODEPRO STUDIO OFFERS SUPPORT FOR WEBSPHERE STUDIO 5.0

(Portland, OR) – Instantiations, a leading provider of advanced Java development and deployment solutions, announced that it has begun shipping CodePro Studio 2.0, a comprehensive suite of products that enhance IBM WebSphere Studio and Eclipse development environments. CodePro Studio now fully supports IBM WebSphere Studio 5.0 and ensures that Java developers deliver high-quality software in record time at reduced cost.

CodePro Studio gives developers a rich set of tools that includes powerful Java coding best practices features like code audit, code metrics, and design patterns. CodePro Studio provides hundreds of valuable features in the areas of best practices, usability, collaboration, deployment, and transition, according to Instantiations. The product liberates Java developers by automating repetitive tasks, improving user interfaces, and streamlining development activities.

CodePro Studio 2.0 was developed during the WebSphere Studio 5.0 early access period. The product has been field-tested and enthusiastically embraced by early users of WebSphere Studio 5.0 tools. Developers find that CodePro Studio liberates them from tedious activities and enables them to spend time on the more creative aspects of software development.

www.instantiations.com/codepro

Candle's PathWAI Suite to Accelerate Deployments

(El Segundo, CA) – Candle Corporation, a leading enterprise infrastructure management provider, has introduced PathWAI, a modular suite of best-in-class solutions that empower businesses to rapidly architect, develop, deploy, and manage their WebSphere infrastructures. Incorporating tools and services, PathWAI enables businesses to harness the power and flexibility of WebSphere by streamlining deployment, accelerating return on investment, and enabling consistent performance.

The PathWAI suite offers modular solutions designed to empower enterprises to derive maximum business value from their WebSphere e-business initiatives. The PathWAI offerings span the total "build, deploy, manage" life cycle – accelerating application infrastructure development and deployment, enhancing management control, and minimizing risk.

The PathWAI suite uses proven methodologies to align developers, operations staff, and senior management. In addition, the methodology focuses on knowledge transfer and training, allowing enterprises to gain the internal expertise necessary to optimize their WebSphere investment and freeing them from expensive ongoing service engagements.

www.candle.com

THE LARGEST WEB SERVICES, JAVA, XML AND .NET CONFERENCE AND EXPO IN THE WORLD

Boston 2003 | London 2003 | Berlin 2003 | Hong Kong 2003

JDJ EDGE conference&expo
web services EDGE conference&expo
.NET EDGE conference&expo
XML EDGE conference&expo

Register by February 14th SAVE UP TO $400
Register by March14th SAVE UP TO $200

3rd Annual

Web Services Edge EAST Conference & Exposition

Connecting the Enterprise with Web Services, Java, XML & .NET

March 18–20, 2003
Hynes Convention Center
Boston

Call TODAY TO REGISTER
201-802-3058
www.sys-con.com/webservicesedge2003east/registernew.cfm

MEDIA SPONSORS

OWNED BY SYS-CON MEDIA
PRODUCED BY SYS-CON EVENTS

Special Insert: Web Services Edge East Conference & Expo

Featuring FREE Tutorials, Training Sessions, Case Studies and Exposition

We are pleased

to bring the latest edition of the very successful Web Services Edge Conference & Expo to the city of Boston this March 18-20, 2003. Now in our third year, we will continue to build on our past success to make available the most current and relevant information to you, our valued attendee.

With the widespread adoption of Web services across the industry, developers are facing new challenges. In this year's conference program, we will address these challenges with our most comprehensive program to date. Web Services Edge 2003 will provide practical approaches and solutions to overcome the hurdles in developing and deploying Web services in today's competitive markets.

Once again Web Services Edge will feature four distinguished tracks - Java, Web Services, .NET, and XML - along with the newly added Vendor Track. The Vendor Track will allow specific sponsors, along with their customers, a platform to present their latest technical developments and real world applications across all the related technologies.

Your three days will include highly informative keynotes, conference sessions, tutorials, industry-leading university certification programs, case studies, and demo presentations. The Expo Hall will be open on March 19 and 20, featuring the largest grouping of quality exhibitors prepared to field your questions and solve your development needs.

All the best,
SYS-CON Events

Features & Attractions

TO ALL CONFERENCE & EXPO REGISTRANTS

- **3 Days Packed with Education and Training**
- **5 Keynotes & 3 Panel Discussions**
- **60 Hard Hitting and Current Seminars**
- **FREE .NET with Russ' Tool Shed Tutorial**
- **Web Services & XML Tutorials**
- **Java University Certification Training**
- **HOT** Industry-Leading Certification Programs
- **NEW** Extended Learning: Evening Conference Session & Certification Classes
- **INFORMATIVE** Round Table Discussions
- Opening Day Welcome Reception
- SAMS Meet the Authors Hot Topics Lounge
- **COMPELLING** Case Studies & Best Practices
- **FEATURED** Product Demonstrations on the show floor
- **RIVETING** Real-time SYS-CON Radio Interviews

Keynotes and Highlighted Speakers

Dave Chappell

VP, Chief Technology Evangelist, Sonic Software

Dave Chappell is the vice president and chief technology evangelist for Sonic Software. He has more than 18 years of industry experience building software tools and infrastructure for application developers, spanning all aspects of R&D, sales, marketing, and support services. Mr. Chappell has also been published in numerous technical journals, and is currently writing a series of contributed articles for Java Developer's Journal.

Eric Newcomer

Chief Technology Officer, IONA

In the role of Chief Technology Officer at IONA, Eric is responsible for IONA's technology roadmap and the direction of IONA's Orbix E2A e-Business Platforms as relates to standards adoption, architecture, and product design. Eric joined IONA in November 1999, and most recently served as IONA's Vice President of Engineering, Web Services Integration Products. Eric is a member of the XML Protocols and Web Services Architecture working groups at the W3C and IONA's Advisory Committee representative to UDDI.org.

Simon Phipps

Chief Technology Evangelist , Sun Microsystems

Simon Phipps, currently chief technology evangelist at Sun Microsystems, speaks frequently at industry conferences on the subject of technology trends and futures. He was previously involved in OSI standards in the 1980s, in the earliest collaborative conferencing software in the early 1990s, and in introducing Java and XML to IBM.

John Magee

Vice President, Oracle9i Oracle

John Magee is Vice President, Oracle9i at Oracle. He has more than 14 years experience in the enterprise software industry and has held positions in product development, product management, and product marketing. In his current role, he manages technical product marketing for Oracle's application server and development tools products, and is responsible for evangelizing Oracle technology initiatives around J2EE, XML and Web services.

Event Sponsors

ORACLE · Microsoft · hp invent · Sun microsystems · IONA · ALTOVA · Rational the e-development company · COMPUWARE

REGISTER BY PHONE–201-802-3058

SUN MICROSYSTEMS Java™ University Program

Web Services Programming Using Java™ Technology and XML

Tuesday, March 18, 2003
9:00 am - 5:00 pm

Who Should Attend

Web services designers and programmers, application developers and programmers using the Java programming language who have experience using the Java™ 2 Platform, Enterprise Edition (J2EE™).

Prerequisites:

Experience using the Java programming language and basic knowledge of XML.

Overview:

This one-day seminar provides in-depth knowledge of Web services and shows how to develop Web services using the Java programming language and XML, the technologies of portable code and portable data respectively.

The session will start with an introduction on fundamental concepts and characteristics of Web services. This will be followed by a detailed explanation of how to implement, how to describe, how to register, how to discover, and how to invoke Web services using core Web services standards - Simple Object Access Protocol (SOAP), Web Services Description Language (WSDL) and Universal Description, Discovery, and Integration (UDDI). In addition, the ebXML standard, which defines the framework for the global electronic marketplace, will be talked about in detail. Also, the tools for building and deploying Web services will be discussed. Each topic will be presented with concrete examples and demonstrations when possible.

Attendees will also learn how to use standard Java APIs for Web services, mainly Java API for XML Messaging (JAXM), Java technology API for XML-based RPC (JAX-RPC), and Java technology API for XML Registries (JAXR), for developing and deploying Web services.

Benefits

- Learn the fundamental concepts and characteristics of Web services
- Gain detailed understanding on core Web services standards: SOAP, WSDL, UDDI
- Gain detailed understanding on ebXML, the standard framework for electronic business
- Learn Java programming language APIs for Web services – JAXM, JAX-RPC, JAXR

Outline

- Web services and Sun™ Open Net Environment (Sun ONE) overview
- Web services standards
- Java APIs for Web services
- J2EE technology and Web services

Java™ 2 Platform: Architect Certification Fast Path

Wed, March 19, 2003
9:00 am - 5:00 pm

Who Shoud Attend

This session is designed for enterprise application architects, system analysts, experienced technologists and developers using Java™ technology seeking certification as an architect for the Java 2 Platform, Enterprise Edition (J2EE™).

Prerequisites

Understanding the benefits of Java technology solutions; experience with object-oriented analysis and design; familiarity with concepts of distributed computing.

Overview

Gaining recognized competency architecting J2EE platform-based solutions is vital to your success as an architect and increases your career opportunities.

Developed and presented by Mark Cade, this one-day session helps prepare attendees to pass the Sun Certified Enterprise Architect for J2EE Technology exam. Cade provides an overview of the components comprising the J2EE architecture as a whole, emphasizes the incorporation of J2EE technology into an architecture, and reviews the exam's testing objectives. Multiple real-world case studies demonstrate correctly architected J2EE technology-based solutions and pinpoint key topics within the exam.

Additionally, you'll learn how to interpret exam objectives, what each of the three exam phases contains, and guidelines and resources to use after the course.

Benefits

- Receive an intensive review of the topics covered on the Sun Certified Enterprise Architect for the Java 2 Platform, Enterprise Edition Exam
- Increase understanding and knowledge of architecting solutions using J2EE technology
- Understand the system qualities: scalability, availability, extensibility, performance, and security
- Understand trade-offs of different architectural choices
- Describe the benefits and weaknesses of potential J2EE technology-based architectures.
- State benefits and costs of persistence management strategies
- Review case studies of J2EE technology-based architecture
- Review practice tests and questions

Outline

- Architect examination overview
- Part multiple choice
- Part assignment
- Part essay

Java™ 2 Platform: Programmer Certification Fast Path

Thursday, March 20, 2003
9:00 am - 5:00 pm

Who Shoud Attend

This session is designed for programmers who have some exposure to the Java™ programming language, and are ready to prepare for the Sun Certified Programmer for Java 2 Platform exam.

Prerequisites

Object-oriented software development experience and familiarity with the syntax and structure of Java technology-based development.

Overview

The development community recognizes that certified competency in developing solutions using Java technology is vital to productivity, reaffirms your value to your organization, and increases your career advancement opportunities.

This valuable session, developed and delivered by Philip Heller, author of the two leading Java technology certification preparation manuals and president of Philip Heller Associates, helps to prepare you for the Sun Certified Programmer for the Java 2 Platform exam. In a comprehensive one-day seminar, Philip provides code-level, detailed review of the Java skills and knowledge you need to confidently approach the exam.

Benefits

- Receive an intensive review of the advanced topics covered on the Sun Certified Programmer for the Java 2 Platform Exam
- Increase your understanding and knowledge of Java programming language syntax and structure
- Prepare for the exam by reviewing practice tests and questions
- Gain a strong understanding of Java technology fundamentals

Outline

- Operating on data
- Shifting
- Shallow and Deep Comparison
- The Literal String Pool

XML Certified Developer *Fast Path*

Tuesday, March 18, 2003
9:00 am - 5:00 pm

Audience

This tutorial is for programmers who have some knowledge of XML and related technologies and would like to pass the IBM Certified Developer Test 141 on XML and Related Technologies.

Prerequisites

Background in object-oriented programming and knowledge of Hypertext Markup Language (HTML). Exposure to XML and related technologies.

Overview

XML is the foundation of two important emerging technologies: Web Services and the Semantic Web. XML expertise and certification is critical for developers who want to remain competitive in the current tight IT job market. The practice tests and questions in this course are specially designed to teach you XML essentials and the key concepts to successfully pass IBM® Test 141 on XML and related technologies.

Outline

- Well formed XML documents
- XML Infoset
- XML namespaces
- Document analysis and modeling
- Document Type Definitions (DTDs)
- XML schemas
- The SAX API
- The DOM API
- XPath and XSLT
- XSL Formatting Objects (XSL FOs)
- Formatting XML with CSS
- XLink and XPointer
- XML Encryption
- XML Signatures
- SOAP, UDDI, and WSDL
- XML architectures based on business and technical considerations
- Optimization and testing of XML applications

Presenter Bio: Joel Amoussou is Founder and Chief Learning Architect of XMLMentor. Joel is the author of the first XML training course specially designed to prepare developers for IBM® Test 141 on XML and related technologies. Joel has created XML content management applications for the aerospace, pharmaceutical, and publishing industries.

Conference at-a-Glance

		JAVA	WEB SERVICES	.NET
TUESDAY MARCH 18	8:00AM — 4:00PM	Registration Open		
	9:00 — 9:50AM	(JV1) Squeezing the Best Out of Java	(WS1) Security: SAML, WS-Security and related issues	(NT1) .NET Framework Overview
	10:00AM — 10:50AM	John Magee, ORACLE		
	11:00AM — 11:50AM	(JV2) Testing Your Java Using JUnit	(WS2) Web Services Management	(NT2) Introduction to ASP.NET
	12:00PM — 2:00PM	Break		
	2:00PM — 2:50PM	Panel - WS-I A road Map for Web Services Standards		
	3:00PM — 3:50PM	(JV3) Building/Deploying the Ant Way	(WS3) Web Services Integration	(NT3) Introduction to Web Services
	4:00PM — 4:50PM	(JV4) Unlocking the Secrets of JDK1.4	(WS4) Using Web Services to Integrate J2EE and .NET Enterprise Applications	(NT4) How To Build Mobile Solutions Using the Microsoft Mobile Internet Toolkit
WEDNESDAY MARCH 19	8:00AM — 4:00PM	Registration Open		
	9:00AM — 9:50AM	(JV5) Java and .NET	(WS5) Combining BPM and BRM technologies: a major step towards corporate agility	(NT5) ASP.NET with Visual Studio.NET
	10:00AM — 10:50AM	KEYNOTE - Sun Microsystems, Speaker TBA		
	11:00AM — 6:00PM	EXPO OPEN 11:00 a.m. - 6:00 p.m.		
	11:00AM — 11:50AM	(JV6) To Not Swing is to SWT! The Swing Alternative	(WS6) Web Services Fundamentals: UDDI, WSDL, XML	(NT6) Best Practices for .NET Development
	12:00PM — 2:00PM	BREAK & EXPO		
	2:00PM — 2:50PM	Panel - Web Services & .NET		
	3:00PM — 3:50PM	(JV7) Talking Back to the Server; the SOAP Way	(WS7) Portals and Web Services	(NT7) Best Practices for ADO.NET Development
	4:00PM — 4:50PM	(JV8) Unlocking the Power of XML	(WS8) Web Services: Next Steps After the Hype	(NT8) Developing Pocket PC applications using the Smart Device Extensions for Visual Studio .NET
THURSDAY MARCH 20	8:00AM — 4:00PM	Registration Open		
	9:00AM — 9:50AM	(JV9) Writing SOAP Services	(WS9) Web Services Best Practices	(NT9) How to Debug with .NET
	10:00AM — 10:50AM	KEYNOTE - Microsoft, Speaker TBA		
	11:00AM — 6:00PM	EXPO OPEN 11:00 a.m. - 4:00 p.m.		
	11:00AM — 11:50AM	(JV10) Working with Data the JDO Way	(WS10) Web Services Startups: Telltales of the Future	(NT10) XML and Web Enabling Legacy Applications Using BizTalk
	12:00PM — 2:00PM	BREAK & EXPO		
	2:00PM — 2:50PM	PANEL - The Future of Java , Moderated by Alan Williamson		
	3:00PM — 3:50PM	(JV11) Enterprise: The Next Generation	(WS11) Web Services Interoperability: The Last Mile	(NT11) Migrating Visual Basic Applications to Visual Basic.NET
	4:00PM — 4:50PM	(JV12) Moving Around the Limitations of J2ME	(WS12) Web Services Case Study	(NT12) How to Develop an End-to-End .NET-Connected Application

XML	VENDOR
(XM1) XML - A Managers Guide	Session TBA
(XM2) OASIS Standards Update	XMLSPY 5 Altova
(XM3) A Definitive Introduction to XML Schemas	SOAP and Java - Parasoft
(XM4) XML in Print - XSL:FO	Session TBA
(XM5) XML in Financial Services	Session TBA
(XM6) Case Study: XML in Life Sciences Oracle	Pattern Driven Application Development- Compuware
(XM7) Using XML for EAI - Best Practices	Managing the Developer Relationship - Sun Microsystems
(XM8) Take XML with You: XML and Mobile Computing	Session TBA
(XM9) Analyzing XForms IONA	Session TBA
(XM10) XML Query	SOAP Security- Rational
(XM11) XPath & XSLT 2.0 BEA	Why Web Services Management? - HP
(XM12) Third Generation XML Tools	Session TBA

Conference Overview

Java Technology Track

The Java track has been specifically designed to allow you to squeeze as much information out of each session as possible. This track is designed for the Java developer, and will be led by industry-leading speakers and authors. Not a track for the beginner or the novice, this track is designed for the experienced developer who wishes to catch up on the latest techniques and APIs.

The Java Track has been designed with you, the more experienced Java developer, in mind. We know you don't have a lot of spare time, and we've designed the track to ensure that your time is maximized and you are armed with all the necessary tools to take your development to the next level.

Microsoft .NET Track

Microsoft .NET represents a major evolution in how applications are developed deployed and managed on the Microsoft platform. The .NET Framework gives developers an object oriented development environment for building all types of applications including desktop, client/server, dynamic web page, wireless devices, server based as well as complete support for XML Web Services and the related XML standards. The sessions in the .NET Track will give you a broad as well as deep understanding of the capabilities in the .NET Framework and how applications built on .NET are easily integrated with applications running in a heterogeneous environments including mainframe, UNIX and J2EE platforms.

Web Services Track

The Web Service track is focused on issues and topics that are at the forefront of development efforts in Web Services. Although the current specifications provide a minimum set of protocols, issues such as security, transaction management, service management and coordination remain in flux. This track presents some of the leading authorities in the field on these urgent topics and addresses all of the questions that currently concern designers, developers and consumers of Web Services.

XML Technology Track

Whether you're looking to understand different XML standards, application techniques, or development tools; or using XML to develop the next generation of Web applications and services, the XML Track is your ultimate training, collaboration, and innovation ground. Sessions include fast-track, in-depth training on XML Schemas and XSL-FO. We will update you on standards development and offer a comprehensive review of the various technologies related to XML that are essential for today's IT manager. The XML Track is armed with real-world applications of XML in financial services, life sciences, enterprise and B2B integration, and mobile computing. We will discuss new developments around XForms, a recent W3C Standards which marks another era of standards based application development; XPath and XSLT 2.0 XML; and Query.

The XML Track explores the technology and standards, real-world applications, and trends which will set the course for the future.

Microsoft® FREE .NET Web Services Tutorial

Russ' Tool Shed

Wednesday, March 19, 2003
9:00 a.m. – 5:00 p.m.

Join Russ as he shows you how to use Visual Studio.NET

9-12:15 Intro to Web Services using VS.NET by Russ Fustino

One of the key ideas behind the .NET strategy is the concept of software as a service, or in short, Web Services. This session will explain what a Web Service is and provide an overview of its related technologies like XML, SOAP and UDDI. We will demonstrate how the .NET Framework makes it easy to implement them for new and existing applications. This session will also provide concrete best practices for building XML Web Services using Visual Studio.NET. We'll answer many common questions like: How will my Web Service scale? How can my XML Web Services enable interoperability with Web Services from other vendors as well as within my own organization? We'll delve into building highly reliable and secure Web Services. Also, we will discuss issues such as dealing with complex data types using WSDL (Web Services Description Language), as well as securing SOAP messages using encryption. We ll see how developers can use enterprise level XML Web Services to simplify customer solutions.

1-2:30 - Advanced Web Services Using ASP .NET Thom Robbins

This session we will explore some of the more advanced areas of SOAP in ASP.NET s support for Web Services. ASP.NET Web Services are the preferred way for Web developers to expose Web services on the Internet. The goal is quick, easy, and high-performing SOAP services. We will look at how to use the SOAP extension classes to create some very interesting applications on top of the core SOAP architecture found within .NET Framework. For instance, you can implement an encryption algorithm or screen scraping on top of the Web Service call. We ll dig into more advanced topics, explore the SOAP headers, and see ways to ensure security in our Web Services.

2:45-4:15 - .NET Remoting Essentials Thom Robbins

Microsoft .NET Remoting is the .NET technology that allows you to easily and quickly build distributed applications. All of the application components can be on one computer or they can be on multiple computers around the world. .NET Remoting allows client applications to use objects in other processes on the same computer or on any other computer to which it can connect over its network. During this presentation we will discuss what you will need to know to get started with .NET Remoting. We will talk about how .NET Remoting compares with DCOM, how to host remoted objects in a variety of applications, how to call remoted objects from a client application, how to control the life time of remoted objects, and how to secure remoting applications.

To learn more, visit www.sys-con.com/webservicesedge2003

Advisory Board

Sean Rhody
Editor in Chief, Web Services Journal
Partner, CSC

Alan Williamson
Editor in Chief, Java Developers Journal
Chief Technology Officer, n-ary

Derek Ferguson
Editor in Chief, .NET Developers Journal
Chief Technology Evangelist, Expand Beyond Corp.

Bob Familiar
.NET Architect, Microsoft New England

Thomas Robbins
Senior Technology Specialist, Microsoft New England

Hitesh Seth
Editor in Chief, XML Journal, Chief Technology Officer, ikigo

J.P. Morgenthal
Chief Services Architect, Software AG

Hotel Arrangements Are Easier Than Ever!

Special arrangements have been made with some of Boston's finest hotels, priced well below regular rates. Hotels are located within walking distance, if not connected to the Hynes Convention Center. To learn more about hotel savings, call ETI at (800) 829-2281 or (201) 444-0060 (direct) or fax reservations to (201) 444-0062. To make online reservations visit www.expotravel.com by Feb 24th, 2003.

SPECIAL RATES

Official Hotels	Address	Single	Double
Sheraton Boston Hotel	39 Dalton Street	$159	$159
Hilton Boston Back Bay	40 Dalton Street	$149	$149

RESERVATIONS

To make reservations call (800) 829-2281 or (201) 444-0060 (direct). Fax reservations to (201) 444-0062. 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 ETI.

REGISTRATION FORM

CONFERENCE: March 18 – 20, 2003 EXPO: March 19 – 20, 2003

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" 201-782-9651
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.

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 2/14/03	Before 3/14/03	On Site
❑ **GP Gold Passport** includes Edge Conference march 18-20, and Select one: Web Services Programming Using Java™ Technology and XML (Mar.18) Java™ Fast Path: Architect (Mar.19) Java™ Fast Path: Programmer (Mar. 20)	$1,495.00	$1,695.00	$1,795.00
❑ **3D Three Day Conference** (Does not include Sun Java™ Education)	$1,295.00	$1,495.00	$1,695.00
❑ **2 Day Conference** (Does not include Sun Java™ Education) (select any two days: ❑ Tue, ❑ Wed, ❑ Thurs.)	$1,195.00	$1,295.00	$1,395.00
❑ **1 Day Conference** (Does not include Sun Java™ Education) (select any one day: ❑ Tue, ❑ Wed, ❑ Thurs.)	$495.00	$695.00	$895.00
❑ **JU1 Sun Java™ University Class** **Select one: Select one: ❑ Web Services Programming Using Java™ Technology and XML (Mar.18) ❑ Java™ Fast Path: Architect (Mar.19) ❑ Java™ Fast Path: Programmer (Mar. 20)**	$695.00	$895.00	$995.00
❑ **JU2 Sun Java™ University Class** **Select two: Select one: ❑ Web Services Programming Using Java™ Technology and XML (Mar.18) ❑ Java™ Fast Path: Architect (Mar.19) ❑ Java™ Fast Path: Programmer (Mar. 20)**	$1,295.00	$1,495.00	$1,595.00
❑ **JU3 Sun Java™ University Class** **Select three: Select one: ❑ Web Services Programming Using Java™ Technology and XML (Mar.18) ❑ Java™ Fast Path: Architect (Mar.19) ❑ Java™ Fast Path: Programmer (Mar. 20)**	$1,495.00	$1,695.00	$1,895.00

4

A. Your Job Title
- ❑ CTO, CIO, VP, Chief Architect
- ❑ Software Development Director/Manager/Evangelist
- ❑ IT Director/Manager
- ❑ Project Manager/Project Leader/Group Leader
- ❑ Software Architect/Systems Analyst
- ❑ Application Programmer/Evangelist
- ❑ Database Administrator/Programmer
- ❑ Software Developer/Systems Integrator/Consultant
- ❑ Web Programmers
- ❑ CEO/COO/President/Chairman/Owner/Partner
- ❑ VP/Director/Manager Marketing, Sales
- ❑ VP/Director/Manager of Product Development
- ❑ General Division Manager/Department Manager
- ❑ Other (please specify)_____

B. Business/Industry
- ❑ Computer Software
- ❑ Computer Hardware and Electronics
- ❑ Computer Networking & Telecommunications
- ❑ Internet/Web/E-commerce
- ❑ Consulting & Systems Integrator
- ❑ Financial Services
- ❑ Manufacturing
- ❑ Wholesale/Retail/Distribution
- ❑ Transportation
- ❑ Travel/Hospitality
- ❑ Government/Military/Aerospace
- ❑ Health Care/Medical
- ❑ Insurance/Legal
- ❑ Education
- ❑ Utilities
- ❑ Architecture/Construction/Real Estate
- ❑ Agriculture
- ❑ Nonprofit/Religious
- ❑ Other (please specify)

_____

C. Total Number of Employees at Your Location and Entire Organization (check all 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 communications and computer 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

E. What is your company's gross annual revenue?
- ❑ $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

F. Do you recommend, specify, evaluate, approve or purchase wireless 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 201-802-3058 by March 4, 2003.

CANCELLATIONS, SUBSTITUTIONS, REFUNDS

Fax written request to SYS-CON Registration 201-782-9651. Requests for refunds received prior to February 15, 2003 will be honored, less a 10% handling charge; requests received after February 15, 2003, and before March 1, 2003, will be honored less a 20% handling charge. No requests for refunds will be honored after March 1, 2003. Requests for substitutions must be made in writing prior to March 14, 2003. 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.

www.sys-con.com/webservicesedge2003east/registernew.cfm

Special Insert: Web Services Edge East Conference & Expo

Featuring FREE Tutorials, Training Sessions, Case Studies and Exposition

JDJ EDGE conference&expo | web services EDGE conference&expo | .NET EDGE conference&expo | XML EDGE conference&expo

Don't Miss the Developer Event of 2003

Meet with the industry experts, professionals, and managers building today s Web Services enterprises!

Conference
March 18–20, 2003

Exposition
March 19–20, 2003

Hynes Convention Center, Boston

Past Exhibitors & Participants

www.sys-con.com/webservicesedge2003east/registernew.cfm

Multi-Pack Subscriptions

SAVE UP TO $400

Pick a 3-Pack, a 6-Pack or a 9-Pack for incredible savings and receive as many as 3 FREE BONUS CDs!

RECEIVE YOUR DIGITAL EDITION ACCESS CODE INSTANTLY WITH YOUR PAID SUBSCRIPTIONS

☐ 

Two Years / 24 issues - $99.99
One Year / 12 issues - $69.99
One Year - Canada & Mexico - $89.99
One Year - All Other Countries - $99.99

☐ 

Two Years / 24 issues - $89
One Year / 12 issues - $49.99
One Year - Canada & Mexico - $79.99
One Year - All Other Countries - $99.99

☐ 

Two Years / 24 issues - $99.99
One Year / 12 issues - $69.99
One Year - Canada & Mexico - $89.99
One Year - All Other Countries - $170

☐ 

Two Years / 24 issues - $99.99
One Year / 12 issues - $69.99
One Year - Canada & Mexico - $89.99
One Year - All Other Countries - $99.99

☐ 

Two Years / 24 issues - $169.99
One Year / 12 issues - $149
One Year - Canada & Mexico - $169
One Year - All Other Countries - $180

☐ 

Two Years / 24 issues - $169.99
One Year / 12 issues - $149
One Year - Canada & Mexico - $169
One Year - All Other Countries - $179

☐ 

Two Years / 24 issues - $129
One Year / 12 issues - $89.99
One Year - Canada & Mexico - $99.99
One Year - All Other Countries - $129.99

☐ 

Two Years / 24 issues - $89
One Year / 12 issues - $49.99
One Year - Canada & Mexico - $79.99
One Year - All Other Countries - $99.99

☐ 

Two Years / 24 issues - $169.99
One Year / 12 issues - $149
One Year - Canada & Mexico - $169
One Year - All Other Countries - $179

3-Pack

Pick any 3 of our magazines for only $175.00 and save up to $275 for a 1 year subscription plus a FREE CD

- 2 Year – $299.00
- Can/Mex – $245.00
- All Other Countries – $315.00

6-Pack

Pick any 6 of our magazines for only $395.00 and save up to $350 for a 1 year subscription plus 2 FREE CDs

- 2 Year – $669.00
- Can/Mex – $555.00
- All Other Countries – $710.00

9-Pack

Pick all 9 of our magazines for only $495.00 and save up to $400 for a 1 year subscription plus 3 FREE CDs

- 2 Year – $839.00
- Can/Mex – $695.00
- All Other Countries – $890.00

Subscribe online today www.sys-con.com/suboffer.cfm

OFFER SUBJECT TO CHANGE WITHOUT NOTICE

SYS-CON MEDIA

Technology-independent metamodels support BPM

Process-Driven Architectures for WebSphere Application Server

BY JIM LIDDLE

As the industry moves toward a new, process-based application model, it is important to understand the terms and standards involved.

ABOUT THE AUTHOR

Jim Liddle, principal solutions architect for Versata, Inc., has worked within the IT industry for over 14 years, the past 4 working with J2EE technology. Jim's background is in the design of large-scale enterprise systems. He holds an MSc in data communications. www.versata.com

E-MAIL

james_liddle@versata.com

There have been many definitions of business process management (BPM) bandied about, but the one that is the easiest to understand and that most accurately reflects BPM is the following:

Business Process Management is the representation and enactment of business interactions that involve multiple steps, possibly over time, that can occur in parallel across multiple systems or people.

Within the BPM market itself there are different terms that tend to be referenced in BPM literature:

- ***Business process:*** A business process is analogous to an end-to-end use case requirement within a business. It is the definition of the activities involved in fulfilling a business requirement regardless of the people, architecture, or systems involved.
- ***Business process modeling:*** Business process modeling tools provide a graphical way to map out the steps required to complete a business process as described earlier. Note that business process modeling vendors do not necessarily provide an executable engine in which to run the process.
- ***Executable business processes:*** Executable business processes are models that can be loaded and executed by a process engine. Process engines execute processes that are long running, as well as straight-through processing (STP)-type processes. The process engine is able to handle the services, integration, and mediation necessary to complete each activity step that forms the sum of the process. And, most leading process engines are certified to run within WebSphere Application Server.
- ***Collaborative business processes:*** Collaborative business processes specify how two concurrent executable business processes interact at the business level. Examples of collaborative business processes are Microsoft BizTalk, RosettaNet, Ariba, and Commerce One.
- ***Business process activities:*** A business process activity represents an interaction between users (worklists), systems, autonomous agents (EJBs), or possibly Web services. They are triggered by events.

Why Companies Are Embracing a Process-Based Architecture

The industry is in the process of moving to a new application model, one that is fundamentally different from the one we're familiar with.

Today, an "application" is a discrete piece of software that a user opens on his or her computer in order to perform a specific task. In the evolving model, the activities that users click on will call on an underlying layer of interconnected services running within WebSphere Application Server, ultimately providing an infrastructure layer, transparently providing software automation to perform business processes as required. This infrastructure change is driven by:

1. The need to change applications rapidly to meet evolving requirements. This is not new, but has been exacerbated by the Internet and the real desire to have more loosely coupled external trading collaborations.
2. The need to integrate existing applications, data, and processes to automate value chains. This is in essence what the industry is describing when referring to "process driven architectures" – i.e., integrate rather than throw away. Corporations are driving this need to integrate in order to reduce total cost of ownership (TCO) and increase return on existing investments (ROI).

The emergence of a reliable, shared, standardized infrastructure based on the J2EE standard, such as WebSphere Application Server, and a service-oriented architecture (SOA) has been a predominant force in enabling this application infrastructure change described above. And, today no company wants to write an application from scratch. The ability

to use a large number of services common to all applications (security, transaction, connection pooling, messaging, etc.) is the nirvana on which the J2EE platform was originally sold.

This same model can be applied to application service pieces to answer questions such as "Why can't I apply services that are common to my business across different business scenarios?" and "Why can't I do it at a higher level of abstraction?" Let me put it another way: enterprise development of WebSphere applications can involve as much as 18 million lines of code. How are you going to deliver this – on time and to budget, and then maintain and update it – without abstraction?

This touches upon another industry nirvana, component-based development (CBD) and software component reuse. Reuse, a primary focus of process-driven architectures, is the ability to reuse frequently used processes, subprocesses, and frequently used activity or component patterns throughout an organization.

Business process management is right there at the convergence of metadata-driven components, using Web services to solve EAI-type requirements and using process activities as an interface to a service for reuse.

Applications are more maintainable because the "process" is not hard-coded but instead described by an XML definition and deployed in real time. This enables flexibility and responsiveness to change at the most volatile architecture layer, that which implements business policies and procedures.

If applications comply with the same business process metamodel, one has a greater chance to be able to compose and recompose these processes across applications, provided that their respective process engines interoperate (design-time composition/orchestration, runtime choreography).

Declarative component reuse becomes achievable as activities become the interface to services that are "new" or "wrappered" legacy processes/components exposed in the middle-tier platform fulfilled by business rules, i.e., we achieve modular, reusable software components that can be manipulated visually in a declarative design tool.

The idea here is that we take the object-oriented idea of reuse and essentially repackage it with reuse by "specification." Process activities can be customized and their properties altered to be reused in different scenarios.

In addition, BPM addresses another aspect of application development, the ability to support long-running units of work. In the past, transaction processing systems have provided a synchronous link between the consumer of the data and the data itself. As applications are more and more integrated with their environment, one cannot expect to have the whole world synchronously (and transactionally) connected to applications. We need the ability to selectively couple our architecture, i.e., to make sensible choices about where it is necessary to have tightly coupled or loosely coupled endpoints.

Ultimately in today's business and technology environment it's not just about connecting application A to application B; it's about taking a set of applications, applying a business process flow to control the integration and a set of rules to ensure data and transactional consistency, and then exposing this aggregate application using whatever middleware is appropriate.

When to Implement a Separate Process Layer

Many companies that have designed, or are thinking of designing, their middle tier using a granular service approach find that they need a layer that coordinates the services to provide an aggregate service. This has often been done in a piecemeal fashion in which complex code resides in session beans that are viewed as "fat" service objects in the middle tier. Moving "process" from the business logic layer into the process layer considerably increases flexibility for the following reasons:

1. A change in the process definition does not then necessarily require a modification of business services.
2. The processes become "visible" and are able to be rapidly changed/amended and redeployed.
3. BPM provides the ability to monitor, audit, and escalate processes. This provides the ability to identify and deal with processes that are not working, while also providing real metrics to the business: Which are the most popular services? Which services are regularly escalated, etc.
4. Volatile rules in the business services layer can be externalized. Examples of this are decision points that can be externalized in the process model and changed at runtime.
5. A process flow layer can be used to connect different user interfaces to the same business service. Business analysts can do process modeling and maintenance.
6. In using the worklist aspects of BPM, organizations have the ability to "push" work to users – this is being referred to as fulfillment of the "last mile" of workflow and EAI. Many companies that are looking to automate and drive back-office value chains use BPM in this way.

The Relationship Between Process and Web Services

One of the big questions within the IT industry, and particularly in the areas of EAI and BPM, is how do Web services help businesses? In a business process context, Web services themselves are in some ways at the very edges of the business processes in that they provide a loosely coupled invocation mechanism to the "units of work" that are implemented by back-office or legacy applications. These are traditionally exposed via proprietary APIs, but increasingly EAI vendors are allowing access to their connectors over SOAP, enabling them to be composed within the various business processes.

Taken in this context, Web services won't always be appropriate, as it is likely that in certain scenarios you will require robust persistent-based durable messaging. In this case it is likely that you will implement a more traditional EAI hub-and-spoke infrastructure, perhaps using JMS as the standard access mechanism. Your application requirements will dictate the type of coupling that you use.

However, there will also be cases when you have low-volume transactions with existing legacy that you are able to elegantly expose through a Web service entry point. This model enables organizations to quickly, and with little

cost, allow connectivity to legacy, and is increasingly being utilized within internal firewall boundaries.

Consider the case in Figure 1. Here we can see an example of a business process designed in Versata's Process Logic Designer that utilizes Web service process activities to communicate with an external credit card verification system and an internal legacy goods ordering system whose Web service endpoint could be exposed through an EAI vendor's connector. In this scenario we are utilizing Web services within our process flow to rapidly satisfy application and business requirements.

BPM vs EAI

BPM is closely related to the problems that are traditionally addressed by EAI. However, BPM is built on a service-oriented architecture. This architecture, as discussed, changes the economics of integrating disparate applications and business processes.

BPM lends itself to smaller initial projects, allowing companies to build incrementally. Also, most BPM solutions feature server-based pricing with drastically reduced requirements for specialized consulting services, which is a necessity with proprietary adapter and scripting language–based EAI solutions.

For any company wanting to invest in the service-oriented architecture approach, BPM adds the capability to automate long-running, multistep business transactions that span heterogeneous systems.

BPM solutions based on Java and J2EE leverage existing skills within the enterprise. Analysts are able to declaratively specify business processes and in-house developers can, where needed, leverage their Java-based skill sets to implement them.

BPM Standards

Given that an end-to-end business process can extend across technology platforms and encompass several different vendors, it stands to reason that vendors and companies are keen to promote standardization of business process modeling, execution, and interactions.

Currently there are many different standardization initiatives, which has resulted in confusion rather than the compatibility for which they were intended. These include:

WSFL

Web Services Flow Language (WSFL), was defined by IBM and is a proposed standard that addresses workflow on two levels:

1. It takes a directed graph model approach to defining and executing business processes.
2. It defines a public interface that allows business processes to advertise themselves as Web services.

WSFL can be used to model processes that move from one activity to the next, where decisions are made at each control point, using an XML syntax that can be read by both humans and machines. By consuming WSFL, a workflow engine can walk through business processes activity by activity, control point by control point. WSFL has now been superceded by BPEL4WS.

XLANG

XLANG , a Microsoft specification that is an XML-based language, describes business process interactions. XLANG is implemented within Microsoft's BizTalk server and its BizTalk Orchestration Designer can compile XLANG schedule drawings into XML-structured XLANG schedule files. Because XLANG is XML-based, its schedules must comply with XML rules for well-formed documents, which means it must conform to a specification or standard schema. XLANG has also been superceded by BPEL4WS.

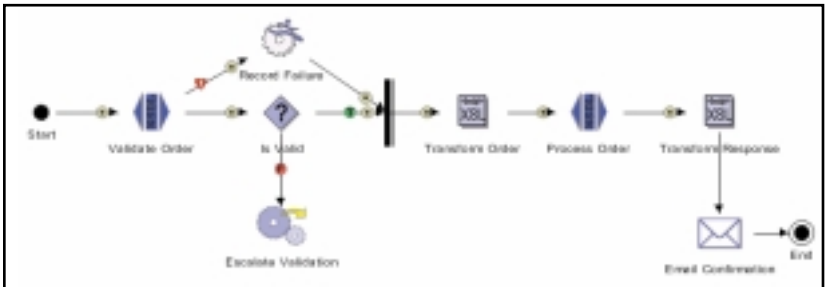


FIG. 1: BUSINESS PROCESS MODEL INCORPORATING WEB SERVICE COMPONENTS

BPEL4WS

BPEL4WS is a relatively new standard from Microsoft, IBM, and BEA. This standard supercedes a previous Microsoft standard, XLANG, and a previous IBM standard, WSFL. It describes the manner in which different participants in a process collaborate from a single participant viewpoint, i.e., rather than relating the process from a holistic viewpoint.

BPMI

The Business Process Management Initiative (BPMI) promotes a standard to describe business processes called Business Process Modeling Language (BPML). BPML describes comprehensive control flow and data flow constructs, providing support for both short and long-running transactions, including compensating activities. BPML also provides support for exception handling and timeouts. However, it does not address such issues as authentication and nonrepudiation.

UML 2.0

It would be relatively easy to forget UML, but process definitions themselves are analogous to a UML activity diagram. UML is extensible and as such can be adapted to many applications. In fact Versata has used this ability to define its own "rules" stereotype.

EBXML

UN/CEFACT and OASIS have collaborated on ebXML, an end-to-end stack of protocols and specifications for conducting electronic business using the Internet. ebXML predates the Web services paradigm and has been discussed as an alternative to Web services, however, this somewhat misses the point. Web services technology currently does not provide adequate support for business transaction services. This, however, is what ebXML was designed for (think EDI-style information exchange) and where it clearly distinguishes itself from Web services.

It is likely that at some point there will be a merging of some of the technologies used in the Web services model and ebXML. An example of this is UN/CEFACT and OASIS recently adopting SOAP as the basis of the ebXML messaging infrastructure. For its part, the W3C is evaluating the ebXML specification and will likely incorporate aspects of the specification that they feel meet requirements missing from the Web services architecture.

EDOC

The Object Management Group's vision for EDOC is to simplify the development of component-based EDOC systems by means of a modeling framework based on UML 1.4 and conforming to the OMG (Object Management Group) Model Driven Architecture. In particular they are striving to achieve:

- A platform-independent, recursive collaboration-based modeling approach
- A business component architecture that provides interoperable business components and services, reuse and composability of components, and reuse of designs and patterns while being independent of choice of technology (e.g., component models)
- Modeling concepts for clearly describing the business processes and associated rules
- A loosely coupled, reuseable business collaboration architecture that can be leveraged by business-to-business (B2B) and business-to-customer (B2C) applications, as well as for enterprise application integration
- A development approach that allows two-way traceability between the specification, implementation, and operation of enterprise computing systems and the business functions they are designed to support
- Support for system evolution and the specification of collaboration between systems
- A notation that is accessible and coherent

This approach has a lot of merit and can be seen as the basis for achieving a Model Driven Architecture (MDA).

WF-XML

The Wf-XML standard, designed and supported by the Workflow Management coalition, is designed to support process-level interoperability. Wf-XML defines a set of operations that can be carried out to start, interrogate, and terminate a process using an Internet/Web services–style paradigm. Although Wf-XML is not based on the SOAP standard, and the operations are not defined in WSDL, it is an applicable standard for Web services, specifically aimed at business process/workflow runtime interoperability. The WfMC recently announced Wf-XML SOAP bindings for interoperability standards support.

The Versata Logic Server is based on a Wf-XML implementation and Versata has done some work in designing additional process artifacts that tie this in with Web services support.

The Workflow Management Coalition has already spent over nine years facilitating this set of standards that enables processes to be used across organizations. We believe that what is essential is the description of the process and the ability for interoperability.

Conclusion

Although a standard business process language would be great to have, the reality is that, with competing standards, this will be in flux until a dominant one emerges. Make no mistake, however. This is fundamental to achieving business process integration and collaboration. The ability to have technology-independent metamodels "behind" BPM is where the "rubber meets the road."

References

BPEL4WS: www-106.ibm.com/developerworks/webservices/library/ws-bpelcol1

BPML: www.bpmi.org.

Positioning paper comparing BPML and BPEL4WS: www.bpmi.org/downloads/BPML-BPEL4WS.pdf

Proposal from the UML 2.0 members concerning process models: www.omg.org/cgi-bin/doc?ad/01-11-01

ebXML: www.ebxml.org

More information on EDOC and MDA: http://omg.org

Wf-XML: www.wfmc.org

Where Will the IT Dollar Be Spent in 2003?

Predicted upturn in tech spending provokes a big question

BY JIM MARTIN

Analysts and suppliers are all predicting a big upturn in tech spending during 2003, which is good news for the industry as a whole. It is certainly welcome news for software vendors and developers, who have been hit hard over the past year. It's also good news for the economy as a whole because new tech spending should spur jobs and thus spur spending. However there is still a huge question that is in need of an answer: Where will those IT dollars be spent, and who will control the purse strings?

Clearly services addressing security will be high on the list, but what about more pure infrastructure plays? In the wake of September 11th, IT spending beyond whatever is absolutely necessary has come to a grinding halt. Now, many organizations are looking at themselves and asking, "Where's the pain," or rather, "where does it hurt the most?" As the purse strings begin to loosen, IT departments wait with bated breath.

However, before they can even begin to assess the list of potential answers, they know that they will have to prove which pain is the most critical so it can be tended to first. The most effective method of proof will lie in one very simple concept: return on investment. ROI has become the way of life for IT departments. In the not-too-distant past ROI was often ignored, and many companies suffered extraordinary consequences as a result. In the next wave of spending, more attention than ever before will focus on ROI.

Projects that have extended ROI are going to be scrutinized at the CXO level very carefully. CRM and ERP systems are still going to be tough sells, considering the operational impact during implementation and the extended time frames for material payback to be realized. There will be some who go down this road, but not before making sure they understand where it is going to take them.

Business process improvement will be the guiding light for the next wave of IT spending. The days of experimental implementations are over. No longer will you hear, "Lets go with it and see what happens." There have been some amazing figures bandied around about the percentage of software that has been sold and delivered but that goes unused, sitting on the shelf. Shelfware will soon become a relic of the past as well.

I say this because a well-managed ROI process should always be an ongoing one in which decisions are studied over time in order to better understand what a successful implementation looks like. If a project is revisited with any frequency, the measurable results will impact how future decisions are made. Something sitting on the shelf provides a zero return, and these days it won't be able to hide in IT development budgets. Gone are the days of opening a storage closet and marveling at the shrink-wrapped ideas that never came to be.

The ROI revolution is taking root in an unexpected place, however. Technology vendors are facing increased competition, and they are responding by becoming better partners with their customers. It has gone beyond lip service and preferred terms. Every day more and more suppliers are becoming integral parts of the process to justify the purchase of their products.

As business accepts the practice of opening itself up to suggestions from multiple vendors, it provides an insight into its own operations that can be learned no other way than by seeing things through different eyes. Vendor competitions are becoming more revealing than ever because both sides have taken a more direct approach with one another. It's no longer just about a customer saying "prove it does what you say it does." Vendors are responding by saying, "Sure, no problem. But first let me make sure I really understand what you want it to do and why you want to do that in the first place."

IT negotiations have become more defined because now we are forced to fill in blanks that we allowed to pass by in the past. Ongoing ROI studies are going to make sure that those negotiations include all aspects of the benefit and risk to the bottom line.

The way that technology is purchased has fundamentally changed – and this change is going to create opportunities for innovation. While sales have lagged many IT suppliers have continued to labor away, refining what was bleeding-edge before the downturn. That technology is now mature and ready for prime time. As spending increases and the newer technology is adopted, business needs will continue to change, and the next wave of newer, better, and faster will begin to rise.

ABOUT THE AUTHOR... Jim Martin has worked in the system integration and communications industry for the past 15 years. Working on design and implementation teams, he has been instrumental in deploying Web-based mission-critical systems.

E-MAIL
jmartin@csstrategies.com