JOCELYNE ATTAL LOOKS AT THE FUTURE OF WEBSPHERE PAGE 24

WebSphere® DEVELOPER'S JOURNAL

The World's Leading Independent WebSphere Developer Resource

WebSphereDevelopersJournal.com

AUGUST VOLUME 2 ISSUE 8

FROM THE EDITOR

New Adapters Make It Easier than Ever to Harness the Power of WebSphere

BY JACK MARTIN • PAGE 4

PRODUCT REVIEW

Bowstreet Portlet Factory: Creating Complex Functionality from Simple Components

BY JAY JOHNSON • PAGE 20

New Adapters Make It Easier than Ever to Harness the Power of WebSphere

BY JACK MARTIN

There is no better time than now to implement WebSphere in your enterprise, and integration is a good place to start. Recently, IBM unveiled more than 50 WebSphere Business Integration Adapters that they have developed (www-3.ibm.com/software/ integration/wbiadapters) and more than 300 that were developed by IBM Business Partners.

The new application and technology adapters address four industries – health care, energy/utilities, automotive, and electronics. Additionally, some cross-industry solutions in the areas of CRM, ERP, and supply-chain management (SCM) were announced. All the adapters connect with WebSphere Application Server version 5 (upgrade!).

Adapting is a good thing. Hey, there's even an adapter that complies with the USA Patriot Act. As always, any implementation should really take into account the current demands of the business and give some consideration to what the future might require as well.

Shall we take out the crystal balls and Ouija boards now? I firmly believe that you must deliver operational improvements and save money if a deal is ever to garner critical management support.

It appears that in our wonderful service economy, technology is being asked to wring out every possible cent of wasted time and money. Additionally, technology is supposed to hold the answer to most of the world's problems.

Let me digress with a headline I saw recently, "Knowledge Management rushes to SARS cure." It made me smile. The article discusses how exposures are being tracked and how scientists worldwide are able to share information to develop a vaccine for the syndrome.

Another claims that a technology product can reduce the incidence of medical errors – well, at least provide better patient medical information. That way the surgeon won't operate on the good side of the patient's brain.

These days management views streamlining as its holy grail, in the name of increased operational effectiveness and improved customer satisfaction. And that is where WebSphere and the timing of its implementation is so perfect.

I have a great deal of contact with IBM and have been involved in many discussions regarding the design, development, and eventual release of WebSphere products. I've spent time with Don Ferguson, in whose brain the first flash occurred; Jocelyne Attal, who sits at the head of WebSphere's marketing; and scores of developers, marketers, and sellers who all had a hand in crafting the product.

People like to take swings at IBM for missing opportunities or for how they do business in general. But I have to give credit where credit is due: IBM created WebSphere and, ultimately, its uses are perfectly on target.

The moment of realization came to me during a recent meeting where I watched an enthusiastic development team explain how they implemented WebSphere and brought together their departmental development efforts. And how – as word is spreading through their company – they are looking to advance their work across other departments.

Their implementation was almost verbatim from a Don Ferguson scenario: a huge mainframe operation with multiple platform developers and disparate systems. I thought, "Is this déjà vu?" How refreshing it was to listen to their excitement in describing how they overcame each obstacle or challenge until the task was completed.

Most managers just don't seem to get as excited as this group does when asked to explain their WebSphere implementation. I think that may be because they are using it to make information sharing faster, customer service more accurate, and to reduce costs in every way they can.

So here is the difference: if you are the direct beneficiary of WebSphere technology, you are more efficient. If it makes your job easier, you are happy. If it saves your company money, you are secure. Are you getting all this?

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, David Caddis, Mike Curwen, Devi Gupta, Lloyd Hagemo, Barbara Johnson, Shannon Lynd, James Martin, Doug Stephen, Victor Valle

EDITOR-IN-CHIEF JACK MARTIN
EDITORIAL DIRECTOR JEREMY GEELAN
EXECUTIVE EDITOR GAIL SCHULTZ
CONTRIBUTING EDITOR PATTI MARTIN
PRODUCT REVIEW EDITOR JAY JOHNSON
SPECIAL PROJECTS EDITOR RICHARD ARONE
MANAGING EDITOR JEAN CASSIDY
EDITOR NANCY VALENTINE
ASSOCIATE EDITORS JAMIE MATUSOW, JENNIFER VAN WINCKEL

WRITERS IN THIS ISSUE
Lloyd Hagemo, Jay Johnson, Ravi Kalidindi, Jack Martin, Edward McCarthy, Sandy Minocha, David Samuel, Andrew Sondgeroth

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
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 DIRECTOR ROBYN FORMA
DIRECTOR OF SALES AND MARKETING MEGAN MUSSA
ADVERTISING SALES MANAGER ALISA CATALANO
ASSOCIATE SALES MANAGERS CARRIE GEBERT, KRISTIN KUHNLE
CONFERENCE MANAGER MICHAEL LYNCH
SALES EXECUTIVE, EXHIBITS JAMES DONOVAN
ART DIRECTOR ALEX BOTERO
ASSOCIATE ART DIRECTORS LOUIS F. CUFFARI, RICHARD SILVERBERG, TAMI BEATTY
WEB DESIGNERS STEPHEN KILMURRAY, CHRISTOPHER CROCE
CONTENT EDITOR LIN GOETZ
CIRCULATION SERVICE COORDINATORS NIKI PANAGOPOULOS, SHELIA DICKERSON, EDNA EARLE RUSSELL
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

WORLDWIDE NEWSSTAND DISTRIBUTION
CURTIS CIRCULATION COMPANY, NEW MILFORD, NJ

ILLUMINATING WEBSPHERE

The ServiceLocator Design Pattern

Using Design Patterns to Streamline Future Development

BY LLOYD HAGEMO AND RAVI KALIDINDI

Developers researching best practices for optimizing Java 2 Enterprise Edition (J2EE) environments can often find only general, one-size-fits all suggestions for coding and tuning specific application components. Although broad best practices can provide a general direction for developers, the complexity associated with J2EE programming requires a detailed set of guidelines to effectively address specific development issues. Many software engineers and organizations that have perfected development processes have created templates – called design patterns – to streamline future development while ensuring high performance levels.

ABOUT THE AUTHOR

Lloyd Hagemo is a senior director for Candle Corporation's Application Infrastructure Management Group. He is responsible for WebSphere tools development. Lloyd has led the successful development of more than 20 products for the WebSphere environment, including operating system utilities, network performance and tuning products, WebSphere MQ configuration and management tools, and application integration solutions.

E-MAIL
lloyd_hagemo@candle.com

Design patterns are proven ways to solve problems common to a number of software projects. Christopher Alexander, et al. first presented patterns in 1977 in their landmark book, *A Pattern Language*, which focuses on construction and architecture of towns and buildings. By using Alexander's patterns as a foundation, Gamma, et al. – known as the Gang of Four – introduced software design patterns in their book, *Design Patterns: Elements of Reusable Object-Oriented Software*, published in 1994. Since then, design patterns have proven to be handy solutions to software problems.

Advantages of Design Patterns

Design patterns represent a proven way to solve recurring problems in the implementation of applications, yielding several important benefits:

- Provide proven and reusable design and architectural artifacts
- Ensure code reuse and reduce redundant code
- Reduce design errors by using a proven solution for an application design problem
- Accelerate the learning curve because design pattern implementations are understood
- Provide well-documented design patterns
- Reduce application problems because the solution works, sample code is available, and engineers have implemented the same design pattern previously

In this article, we will explore the ServiceLocator design pattern, which can improve performance and Java Naming and Directory Interface (JNDI) code reuse in your J2EE application. Before using any design pattern, it is always good to know the problem it solves and why that is important.

A Case Study

Candle architected a solution that allowed customers to build IBM WebSphere MQ applications as components. This solution leveraged WebSphere MQ as a transport and registered components in a Lightweight Directory Access Protocol (LDAP) directory (see Figure 1). These components were WebSphere MQ–based applications. The application program would pass the name of the service that the application developer wanted to invoke. Within the infrastructure and below the programming interface, a directory lookup was executed based on the name to determine the WebSphere MQ queue manager and queue required to process the request. Candle engineers knew that directory lookups are expensive, even though LDAP is optimized for fast read and search access. To reduce this overhead, they implemented client caching. This caching retained the service lookup in memory so that when the application invoked the same service, the infrastructure did not need to reread the directory. The information was retrieved from the cache. Although other problems were addressed in this WebSphere MQ implementation, the specific details of the complete set of features will not be covered in this article.

ServiceLocator Context

The J2EE platform mandates the use of JNDI to get services such as EJBHomes, DataSources, JMS Connection factories, and JMS Destinations. These services act as static factories to create/remove resources. For example, the EJBHome factory is used to create/remove/find EJBObjects, and the DataSource factory is used to get

database connections from the connection pool. These services must be mapped with user-friendly names in the deployment descriptors so that the J2EE server binds these services with these names when the services are deployed in the server. Clients can then look up these services through user-friendly names rather than hard-coded service properties. This type of naming and access mechanism ensures location transparency to the services. It allows clients to be decoupled from the original services and services to be changed without revising the client's code base. This capability facilitates flexible J2EE component development, assembly, and deployment.

Problems and a Solution: the ServiceLocator Pattern

Flexibility comes with a performance overhead and redundant code across clients' code base since clients have to access JNDI for every service call and write JNDI code in each client class. Some vendors attempted to solve this problem by providing cache support within their application servers. This support does not apply to all JNDI reads and is very limited in scope. Servlets, Java applications, and JavaServer Pages (JSPs) all use a JNDI directory interface to get services. By reducing the need to read the directory, enterprises can dramatically improve an application's performance. The ServiceLocator design pattern is used to provide local memory cache support for service names and service values that have been read from the JNDI directory.

This process also encapsulates the JNDI access logic for the services so that the client classes do not have to write JNDI code.

ServiceLocator Pattern Implementation

The ServiceLocator is implemented using the Singleton design pattern, probably the most widely used design pattern. It is intended to ensure that a class has only one instance, and to provide a global point of access to it. Using a Singleton ensures that only one object is instantiated per Java Virtual Machine (JVM) for a class. The Singleton design pattern continues to play an important role in J2EE environments, but it must be used properly to avoid bottlenecks in a multithreading application server environment. A bottleneck can occur because the Singleton design pattern allows only one thread at a time to access the data if it is synchronized. Although we have addressed thread safety, a Singleton pattern can be implemented without synchronization in some scenarios.

In a ServiceLocator implementation, the client applications – JSPs, servlets, Swing, other Enterprise JavaBeans (EJBs) – would instantiate the class that will use the first ServiceLocator object created in a particular JVM. The ServiceLocator constructor creates a Hashtable that is used to hold the service that is indexed by JNDI name. Several methods are included in the ServiceLocator class, as defined below. The class constructor instantiates the Hashtable used to hold an object in memory indexed by a JNDI reference. A Hashtable was used to hold the cached service because it is a synchronized collection that does not allow null keys. There are three basic methods provided within the code; for example, the method getRemoteService (see Listing 1), which can be used to store and return the home address for a remote EJB.

A similar method, getService, is used to return other objects that have been stored in the Hashtable. The last two methods are utility methods used to update a service entry (updateService) and remove an entry (removeService). It is important to call the removeService method when making an object that is no longer required for the cache available for garbage collection. All method calls throw a user-defined exception, ServiceLocatorException, and recovery must be provided by the calling application.

Clients use the getRemoteService method as shown in Listing 1 for getting remote service objects such as EJBHome objects. Because these are remote objects, the application developer needs to call the narrow() method of the PortableRemoteObject class to ensure that the object can be cast to the desired type. The following client code is used to call ServiceLocator to retrieve a remote home EJB reference to the caller.

```
ServiceLocator locator =
ServiceLocator.getInstance();
AccountHome home =
(AccountHome)
locator.getRemoteService("jav
a:comp/env/ejb/Account",Accou
ntHome.class);
```

The clients call the getService() public method, passing in the JNDI name and service class. This method will use the JNDI name as the key to look up the service for the service from the cache using the Hashtable get method. This method is used for getting services such as DataSources and environment variables. The following client code is used to call ServiceLocator to retrieve an object stored in the Hashtable based upon JNDI lookup.

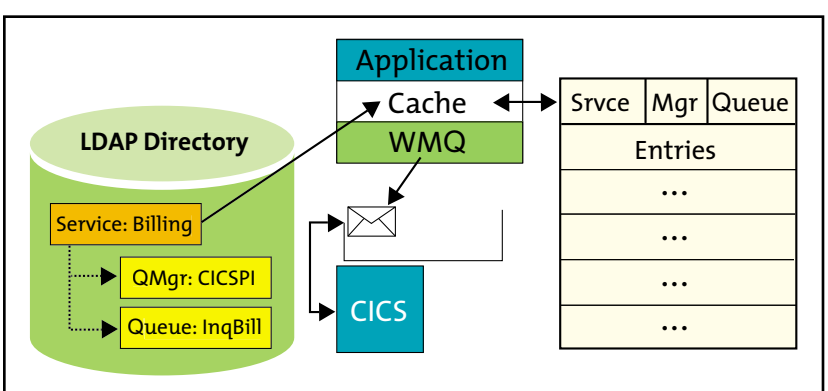


FIG. 1: READ LDAP DIRECTORY ONCE, THEN RETRIEVE FROM LOCAL CACHE

ABOUT THE AUTHOR

Ravi Kalidindi is a senior software engineer for Candle Corporation's Application Infrastructure Management Group. Ravi has worked with Java and J2EE since its inception on a wide variety of J2EE projects and has published several articles that focus on Java and J2EE best practices.

E-MAIL
ravi_kalidindi@candle.com

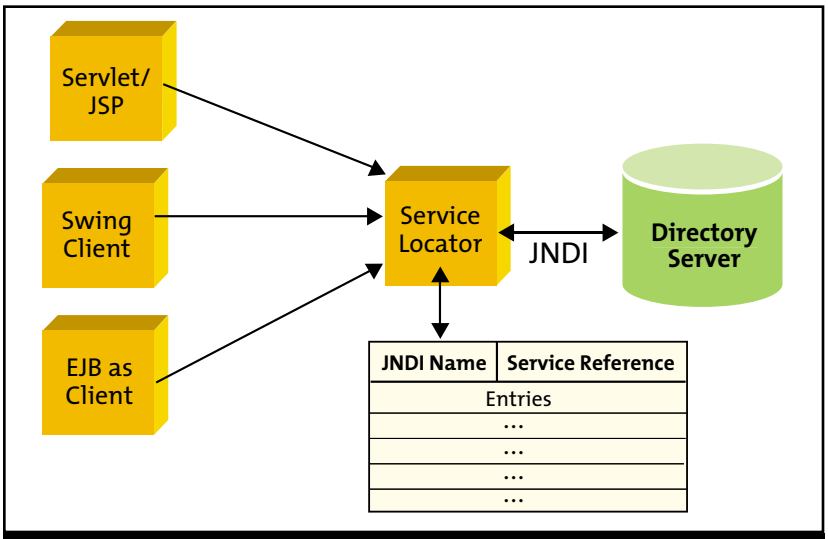


FIG. 2: THE FLOW FROM THE CLIENT TO THE SERVICELOCATOR, WHERE THE LOOKUP REQUEST IS MET EITHER THROUGH THE CACHE OR THE DIRECTORY SERVER

```
ServiceLocator locator =
ServiceLocator.getInstance();
DataSource ds = (DataSource)
locator.getService("java:comp
/env/jdbc/CandleDB",DataSourc
e.class);
```

If the service object is returned, the request has been satisfied using the ServiceLocator local cache memory. If a null entry is found, then the getService() method will access the Web application server directory using an initial context as the starting point to the JDNI tree. The context lookup method is used to get the reference to the service object. The object returned by the JNDI lookup is stored – indexed by the JNDI name – into the Hashtable. The next caller of that service will get the cached version returned. Our implementation included a method to update the cache based on a service that is being relocated to another application server or that is no longer available on the application server. updateService() has the same signature as the getService() method. The user application can call this method to refresh the cache upon receiving an error. updateService() will delete the entry from the Hashtable and re-create it from the directory, returning the new service. This new service is added to the cache.

Practical Results for the ServiceLocator Pattern

We always like to see the results of the ServiceLocator pattern in real-world implementations. Candle worked with a client that had more than 100 clients per minute accessing a database through an EJB entity bean. The application was implemented using standard J2EE practices that executed a JNDI lookup for each access. This process resulted in 100 JNDI calls per minute to read the same EJBHome object reference. Candle recommended that the customer implement the ServiceLocator design pattern to encapsulate the EJBHome object references in memory to resolve the application performance problem. The results were very impressive. Before the implementation, it took, on average, 600 milliseconds to read the directory. This figure translates to roughly a half-second delay per minute. Once Candle implemented the ServiceLocator, response time for an EJBHome lookup averaged 25 milliseconds – a 2,400% improvement in performance over the previous environment.

Conclusion

The benefits of using a ServiceLocator to look up and cache services are obvious. It is important to note that any resource or parameter stored in the directory server can be cached using this design pattern. The ServiceLocator pattern acts as a single point of control for the clients, improves J2EE application performance, and reduces redundant JNDI code across client classes. The result is that applications are easier to maintain because of the minimal code required to perform JNDI lookups and handle remote exceptions. Additionally, applications perform better using a local memory cache to hold directory services, which reduces the network overhead and the number of searches of the directory on disk.

LISTING 1: METHOD USED TO RETURN A REMOTE OBJECT REFERENCE TO THE CALLER

```
public Object getRemoteService(String jndiName, Class
  serviceClass)
      throws ServiceLocatorException {

      // validate method arguments
      if(jndiName == null || serviceClass == null){
            throw new
            ServiceLocatorException("Invalid method
            arguments");
      }
      Object home = services.get(jndiName);

      try{
      if(home == null){
if(context == null){
                        context = new InitialContext();
            }

            Object objref = context.lookup(jndiName);
            home = PortableRemoteObject.narrow
            (objref,serviceClass);
            services.put(jndiName,home);

      }
      }catch(NamingException ne){
            throw new ServiceLocatorException(ne);
      }
      return home;
  }
```

Understanding Tivoli Access Manager for WebSphere Application Server

Component offers centralized management of EJBRoles

– BY EDWARD McCARTHY –

ABOUT THE AUTHOR

Edward McCarthy is an e-business specialist for IBM Global Services Australia. For the past three years he has provided technical support and solution design for WebSphere and MQSeries across all platforms. Prior to that, he was a CICS systems programmer for nine years. He has contributed to a number of IBM Redbooks, including *z/OS WebSphere and J2EE Security Handbook* (SG24-6846) and *IBM WebSphere Everyplace Server Service Provider and Enable Offerings: Enterprise Wireless Applications* (SG246519).

E-MAIL

edwardmc@aul.ibm.com

J2EE Security provides a mechanism called EJBRoles that can be used to provide security for applications running in J2EE-compliant application servers, including WebSphere Application Server. Use of EJBRoles requires that users, or groups of users, be mapped to EJBRoles so that WebSphere can perform security checks when applications are running. It is common to find several WebSphere environments in a large organization. Management of EJBRoles across these environments can become complex and expensive.

IBM's Tivoli Access Manager provides a software component that can be integrated with WebSphere Application Server to provide centralized management of EJBRoles. When WebSphere is configured with this component it relies on Tivoli to determine if access to an EJBRole is allowed. This article describes how Tivoli Access Manager (TAM) for WebSphere Application Server works with WebSphere on distributed platforms. For the purposes of this article, I used WebSphere Application Server v4 with Fix Pack 3, and Tivoli Access Manager v4.1 with no fix packs.

IBM provides a range of solutions within the Tivoli Access Manager family that build to form a comprehensive security management and enforcement environment that extends from Web applications to messaging applications and the operating system platforms on which they run. IBM's WebSphere Application Server (WAS) is a high-performance and scalable transaction engine for dynamic e-business applications. Using WAS coupled with Tivoli Access Manager, customers can finally build tightly integrated, centralized identity management solutions that can protect their J2EE, Web, and legacy resources.

More on EJBRoles and J2EE Security

Before describing how TAM for WAS works, it is worth spending a few moments to explain EJBRoles in a little more detail.

An EJBRole typically identifies a logical resource of some sort. Using a bank as an example, different types of employees, such as tellers and managers, have different roles. For instance, a manager may be authorized to withdraw larger amounts of money than a teller. In years past, an application would perhaps hard-code some test to determine if the user running the application was a teller or a manager, or instead look up the user in a table.

EJBRoles allow a Java programmer to use a programmatic or declarative approach to security. For example, a programmer could define two EJBRoles, Teller and Manager, and create two different methods in an EJB, called handleLargeAmounts and handleSmallAmounts.

Declarative Security

The developer would specify in the deployment descriptor that users have to be authorized to use the "Manager" EJBRole in order to invoke the "handleLargeAmounts" method, and users have to be authorized to use the "Teller" EJBRole to invoke the "handleSmallAmounts" method. This approach is called declarative security. A servlet can also be protected with an EJBRole, meaning that before an authenticated user can invoke a servlet, the user must be authorized to the EJBRole that protects the servlet.

When a method that has been marked with an EJBRole is invoked, it is up to WebSphere to determine if the user running the application is authorized to that EJBRole.

Programmatic Security

In the programmatic approach, some sort of logic test in the code is used to determine if the user running the application is authorized to run a section of the code. This can be done by using the statements isUserInRole (for use in a servlet) and isCallerInRole (for use in an EJB).

When these statements are executed, it is up to

WebSphere to check if the authenticated user is authorized to a particular EJBRole. WebSphere returns a value of true or false to the application to indicate the result of the authorization check. To enable this programmatic approach, role names need to be mapped to EJBRoles in the deployment descriptor.

Without Tivoli Access Manager

When you are running WebSphere alone, it is up to WebSphere to handle the process of checking if an authenticated user is authorized to an EJBRole. WebSphere does this based on which users and groups have been assigned to the EJBRole. Typically, during deployment of an application WebSphere will detect that there are EJBRoles in the deployment descriptor. It will then give you an opportunity to find the users and groups that are defined, and map them to EJBRoles. It is also possible to specify users in the deployment descriptor at application assembly time.

If you are mapping users and/or groups to EJBRoles during deployment of the application, then you are faced with having to do this each time you deploy a new version of the application. To deploy a new version of the application, you have to remove the current version, which also removes the existing mappings.

With Tivoli Access Manager

With Tivoli Access Manager configured into WebSphere, the process of checking if a user is authorized to an EJBRole is delegated to TAM. In TAM, objects are defined that represent the EJBRoles. Users and/or groups are then assigned to access control lists (ACLs), which are then attached to EJBRoles objects.

When WebSphere needs to check authorization, it calls TAM, which checks the ACLs and objects to determine if the user has access to the EJBRole.

The Deployment Descriptor Always Wins!

The key point with regard to the use of EJBRoles in WebSphere is that they have an effect only if your servlet/EJB is running as authenticated in WebSphere. The only way a servlet can run as authenticated in WebSphere is if it has been marked as such in the application's deployment descriptor.

In an EAR file containing servlets and EJBs, a web.xml file acts as the deployment descriptor for the servlets, and an ejb-jar.xml file acts as the deployment descriptor for the EJBs. These XML files indicate which EJBRoles are used and whether a servlet runs as authenticated. Listing 1 is a snippet from a web.xml file showing a security constraint for a servlet. The listing shows that for any URL containing "secure/", "BASIC" authentication will be required, and the authenticated user will have to be granted access to the "Employee" EJBRole. Listing 2 is an example of XML from an ejb-jar.xml file showing how a method of an EJB is being protected with an EJBRole. The example shows that the "runAsRoleCEO" method of the "EJBSample" EJB requires the user to be authorized to the "CEO" EJBRole in order to invoke it.

Performing some sort of authentication operation outside of WebSphere, such as in WebSEAL or the TAM plug-in for WebSphere Edge Server, has no effect on WebSphere's determination as to whether a servlet is to run as authenticated.

When WebSphere receives a request to run a servlet, it will check the deployment descriptor to determine if a security constraint is configured. If so, then WebSphere will try to determine if authentication information is present in the HTTP object it has received, and if not, it will send back a response to obtain it.

z/OS WebSphere and J2EE Security Handbook, an IBM Redbook, offers detailed information about how to set security constraints in the deployment descriptors. Although this redbook focuses on WebSphere Application Server on z/OS, the concepts it describes for setting security constraints and J2EE security are the same regardless of what platform WebSphere is running on.

Installation of Tivoli Access Manager

Installation of TAM is described in IBM *Tivoli Access Manager for WebSphere Application Server User's Guide* (SC32-1136). Follow the instructions carefully! Be sure to use the manual corresponding to the version of the product you are using. Tivoli manuals can be viewed online at: http://publib.boulder.ibm.com/tividd/td/tdmktlist.html.

Defining EJBRoles in TAM

TAM supplies a tool called migrateEAR, which takes an application EAR file as input; for each EJBRole defined in the EAR file it defines an object in TAM, and creates an ACL. However, this should be seen as very much a one-off process. When a new version of the application EAR file is delivered, you probably do not want to use the migrateEAR tool. For example, if the application EAR file now no longer uses a particular EJBRole, that EJBRole is no longer defined in the deployment descriptor and migrateEAR will not detect this. The tool will not remove from TAM an EJBRole that is no longer used.

In a situation in which an organization is planning to use EJBRoles in its applications, a process must be developed through which the application area creating EJBRoles passes on this information to an area responsible for managing TAM resources. The organization would also need to identify which users/groups are to have access to the EJBRoles.

Tivoli Access Manager Objects

In Tivoli Access Manager, objects that represent an EJBRole are all anchored off what is called the root object "/", and must start with:

```
/WebAppServer/deployedResources
```

The next part of the object name for an EJBRole object is the EJBRole name itself, for example the object for an EJBRole called "Employee" would be:

```
/WebAppServer/deployedResources/Employee
```

When TAM is invoked by WebSphere, the object it constructs to be checked follows this format:

```
/WebAppServer/deployedResources/<ejbRole>/<appName>/<hostname:LdapPort>/<hostname>/<moduleName>
```

When TAM receives this as the object to check authorization against, it searches from the start of the object tree at "/", looking for the most exact match.

You could simply define an object of this form:

```
/WebAppServer/deployedResources/<ejbrole>
```

This would be sufficient for a match. Tivoli Access Manager searches down the object tree until it finds the object that most closely matches the received object. When TAM finds this object, it then checks the ACL being "enforced" for this object. The ACL will specify the users or groups that have permission to access this object.

Whether or not to use an <appName> in the TAM object depends on whether the EJBRole is to be treated as a common EJBRole across the applications it is defined in. For example, a bank may have two applications, AppA and AppB, deployed into WebSphere, with the "Teller" EJBRole defined in both. In such a situation it would make sense to define an object called /WebAppServer/deployed Resources/Teller rather than define two objects, such as /WebAppServer/deployedResources/ Teller/AppA and /WebAppServer/deployedResources/Teller/ AppB. Using the former approach, the ACLs would each be attached to only one object, as opposed to two.

However, consider another company that is running two applications that have different users in the company, but both define the same EJBRole, "Employee". In order to ensure that only the correct sets of users have access to their respective EJBRoles, you would define two objects:

```
/WebAppServer/deployedResources/Employee/AppA
/WebAppServer/deployedResources/Employee/AppB
```

ACLs

ACLs can be called anything. The migrateEAR tool creates ACLs with names like _WebAppServer_deployed Resources_<ejbRole>_<appName>_ACL. However, there is no need for the ACLs to be of this format. For example, an ACL could be called WAS_EJBRole_Employee.

The WebServer Action Group

When configuring Tivoli Access Manager, part of the process is to issue these two TAM commands:

```
action group create WebAppServer
```

and

```
action create i invoke invoke WebAppServer
```

When giving a group or user access to an ACL, use this type of command:

```
acl modify
_WebAppServer_deployedResources_AdminRole_admin_ACL set group
 pdwas-admin T [WebAppServer ]i
```

The T[WebAppServer]i command breaks down into:

T	*Traverse bit*
WebAppServer	*Action group name*
i	*A permission*

THE TRAVERSE BIT

When WebSphere invokes Tivoli Access Manager to check if a user has access to an EJBRole, TAM constructs an object and then checks for authorization. For example, to determine if a user has access to the object/WebAppServer/deployed Resources/EJB Role, the authorization engine must traverse from the root object "/" down to the "/WebAppServer/deployedResources/EJBRole" object. If at any point the user isn't allowed to traverse beyond a particular object, then regardless of any other permission on subsequent objects, the user is not allowed access to the object.

The traverse bit part of the permission in the ACL command permits this traversing of the object tree.

THE WEBAPPSERVER ACTION GROUP

The "[WebAppServer]" part of the permission is the action group. It is a value used to denote that the permissions granted by the ACL are for use only by Tivoli Access Manager. When TAM checks authorization it makes its calls using the WebAppServer action group.

It is possible to define any number of action groups to TAM. For example, you could define an action group called "MyApplication". You then could set up two ACLs and control access to WebSphere and MyApplication resources like this:

```
acl modify
_WebAppServer_deployedResources_Employee_ACL
set user z12345 T[WebAppServer]i
acl modify
_MyApplication_Resources_Widgets_ACL set user
z123456 T[MyApplication]i
```

Both ACLs give the same user access to the "i" permission, but map it to two different action groups. The permissions after the action group (the characters that come after the last "]") are relevant only for those applications using that action.

Other applications could call TAM to check authorization, and use the "MyApplication" identifier. This approach allows TAM to support the setting up of ACLs to control access for many different applications.

THE "I" PERMISSION

The "i" permission is a setting defined in the ACL. Typically, permissions in the ACL correspond to an action. For example, ACLs used in conjunction with the TAM WebSEAL or TAM Edge plug-ins have permissions that correspond to HTTP GET or POST requests.

When TAM does its authorization check, it is asked by WebSphere to check if a user has access to an EJBRole. TAM checks if the user has permission for the "i" action for the WebAppServer action group.

```
   acl modify
_WebAppServer_deployedResources_Employee_IBM
EBiz_ACL set user citizen T[WebAppServer]i
```

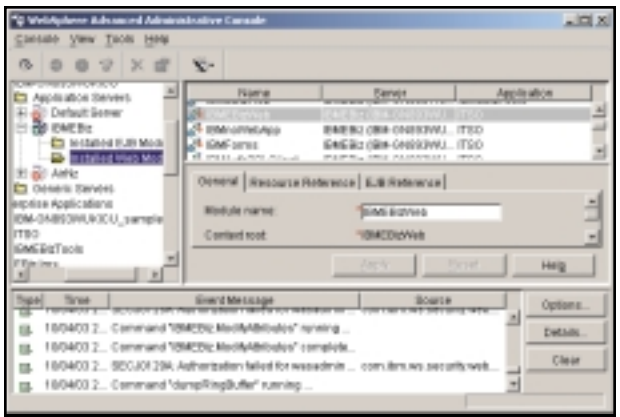


FIG. 1: DISPLAY OF THE ITSO APPLICATION, WHICH HAS BEEN DEPLOYED INTO WEBSPHERE

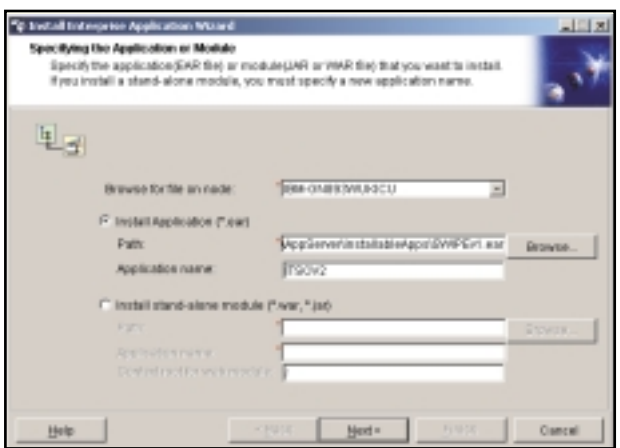


FIG. 2: DEPLOYING A NEW VERSION OF THE ITSO APPLICATION WITH A NEW APPLICATION NAME

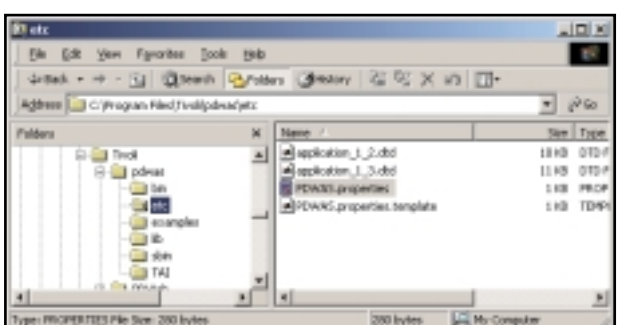


FIG. 3: LOCATION OF THE TIVOLI ACCESS MANAGER PROPERTIES FILE

This Tivoli command grants access to the "Employee" EJBRole to the userid "citizen", because the ACL has the WebAppServer action group defined, and has also been assigned the "i" permission.

Application Name and the TAM Object

Previously I discussed how the objects in Tivoli that represent EJBRoles can contain the application name. There is another important consideration here to do with the name that an application is called when it is deployed into WebSphere.

Figure 1 is a view of the WebSphere administration console showing an application, "ITSO", which has been deployed.

When this application requires an access check on an EJBRole to be performed by WebSphere, TAM will construct object names that contain the value "ITSO" for checking authorization.

Now suppose we have a new version of the application, and when we deploy it we decide to call it ITSOV2, as shown in Figure 2.

When the ITSO application requires WebSphere to perform an EJBRole authorization check via Tivoli Access Manager, the object name will contain the appName "ITSOV2".

If you have defined TAM objects with an application name of ITSO, then those objects will no longer match the new objects being checked for by TAM. Therefore, if you are going to use the application name as part of the TAM objects, the application – regardless of version – must always be deployed into WebSphere with the same name. If you are not going to use the application name as part of the TAM objects, then you can call the application anything when you deploy it into WebSphere.

Note: During testing to confirm the above, I noticed that after deploying the application again with a new name TAM was still testing for objects using the old application name. After restarting the application server, TAM started using objects with the new application name.

Tivoli Access Manager Properties

Chapter 5 of the *IBM Tivoli Access Manager for WebSphere Application Server User's Guide* describes how TAM runtime properties can be adjusted by modifying the PDWAS.properties file. Figure 3 shows where this file is located on a Windows system.

On an AIX system it is located in the directory where the Tivoli Access Manager software is installed. Properties that can be adjusted are:

1. Limit simultaneous connections
2. Enable static role caching
3. Define static roles
4. Configure dynamic role caching
5. Specify logging mechanism type
6. Specify logging level
7. Specify root object space name
8. Specify document type definition directory

Incorrect Authorization

If, after setting up Tivoli Access Manager, you find that authorization decisions are not what you had expected, you will need to find out what object and userid TAM is using to check authorization.

Tracing of TAM can be enabled by setting values in the PDWAS.properties file, as mentioned previously. However,

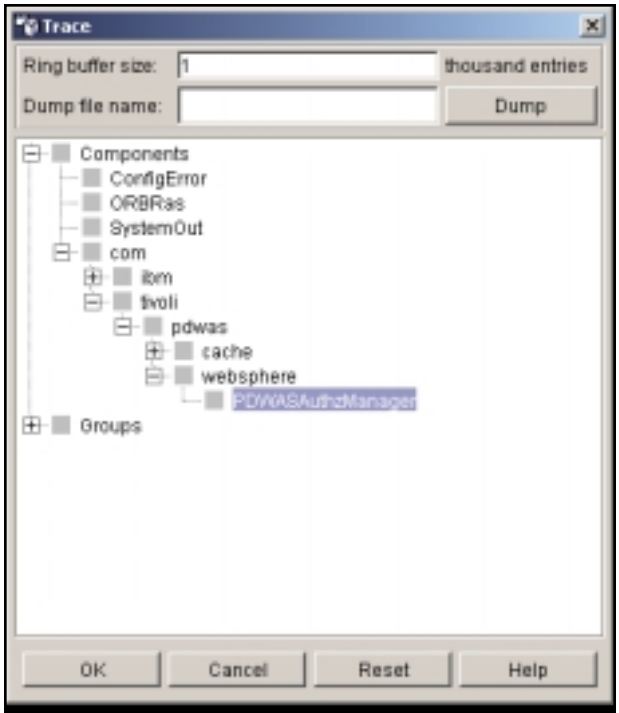


FIG. 4: ENABLING TRACING OF TIVOLI ACCESS MANAGER

that approach requires restarting WebSphere to pick up the change in the PDWAS.properties. An alternative is to use the WebSphere admin facility to set and capture a trace.

In the WebSphere admin facility, select the application server you want to trace activity in, and then select the Trace service. Click on "trace specification" and you will get a display similar to that shown in Figure 4.

TAM now shows up under the Components->com headings. Right-click on "PDWASAuthzManager" and select "All". Click OK, then OK on the Trace service panel, and finally the Apply button.

Run the servlet/EJBs, then go back into the Trace service and dump the trace to a file. You can then view the captured trace to determine which object and userid TAM used for the authorization check. The sample trace shown in Listing 3 shows that the object for which TAM is requesting authorization is:

```
/WebAppServer/deployedResources/Employee/ITSO/IBM-GN893WUKICU:389/IBM-GN893WUKICU/IBMEBiz
```

The userid is "wasadmin" and the EJBRole is "Employee". You can then use this information to check what has been defined in TAM, for example:

- Is the userid "wasadmin" defined?
- What object maps to the resource being checked by TAM?
- Does the userid have access to that object via an ACL?

Summary

The Tivoli Access Manager component provides centralized management of EJBRole security in WebSphere. Understanding how EJBRoles are mapped by objects in Tivoli Access Manager is the key to how you will set up Tivoli to manage EJBRoles.

Resources

- *Security Management:* www.ibm.com/software/tivoli/solutions/security
- *WebSphere software platform:* www.ibm.com/software/websphere

LISTING 1: A SECURITY CONSTRAINT FOR A SERVLET

```
<security-constraint>
    <web-resource-collection>
      <web-resource-name>Secure Tools</web-resource-name>
      <description>Basic Authentication</description>
      <url-pattern>/secure/*</url-pattern>
      <http-method>GET</http-method>
      <http-method>POST</http-method>
    </web-resource-collection>
    <auth-constraint>
      <role-name>Employee</role-name>
    </auth-constraint>
    <user-data-constraint>
      <transport-guarantee>NONE</transport-guarantee>
    </user-data-constraint>
  </security-constraint>
  <login-config>
    <auth-method>BASIC</auth-method>
    <realm-name>WASWebContainer</realm-name>
  </login-config>
```

LISTING 2: PROTECTING A METHOD WITH AN EJBROLE

```
  <method-permission>
      <role-name>CEO</role-name>
      <method>
        <ejb-name>EJBSample</ejb-name>
        <method-intf>Remote</method-intf>
        <method-name>runAsRoleCEO</method-name>
        <method-params>
          <method-param>java.util.Vector</method-param>
          <method-param>itso.utility.Tools</method-param>
        </method-params>
      </method>
```

LISTING 3: A SAMPLE TRACE

```
[10/04/03 21:00:25:717 EST] 4b3963bb PDWASAuthzMan D
performImplies: PDSecurityRole
[/WebAppServer/deployedResources/Employee/ITSO/IBM-
GN893WUKICU:389/IBM-GN893WUKICU/IBMEBiz]
[10/04/03 21:00:25:717 EST] 4b3963bb PDWASAuthzMan D
performImplies: PDSecurityTag [[WebAppServer]i]
[10/04/03 21:00:25:717 EST] 4b3963bb PDWASAuthzMan D
performImplies: Perform the implies
[10/04/03 21:00:25:777 EST] 4b3963bb PDWASAuthzMan D
performImplies: Return Value [false]
[10/04/03 21:00:25:777 EST] 4b3963bb PDWASAuthzMan <
performImplies: (EXIT)
[10/04/03 21:00:25:777 EST] 4b3963bb PDWASAuthzMan D
isGrantedAnyRole: Return value [false]
[10/04/03 21:00:25:777 EST] 4b3963bb PDWASAuthzMan <
isGrantedAnyRole: (EXIT)
[10/04/03 21:00:25:777 EST] 4b3963bb WebCollaborat <
checkAuthorization: throw AccessException
[10/04/03 21:00:25:777 EST] 4b3963bb WebCollaborat D
checkAuthorization() failed, here is the message in the
exception: Authorization failed, Not granted any of the
required roles: Employee
[10/04/03 21:00:25:777 EST] 4b3963bb WebCollaborat A
SECJ0129A: Authorization failed for wasadmin while invoking
GET on default_host:/IBMEBizWeb/secure/ejbCaller,
Authorization failed, Not granted any of the required roles:
Employee
```

PRODUCT REVIEW

Creating complex functionality from simple components

Bowstreet Portlet Factory

BY JAY JOHNSON

For years, the holy grail of programming, at least from a CIO point of view, was a magic software tool variously called a "code generator," a "fourth-generation language," or an "autocoder." This tool would generate the correct executable code to fulfill a user's wishes, based on relatively high-level requirements. Such a tool would definitely fulfill a project manager's wishes, since creating software is a labor-intensive process requiring high-priced talent.

ABOUT THE AUTHOR

Jay Johnson is a J2EE architect with Tier Technologies, which combines deep domain expertise with technical capabilities to solve problems for their clients in government, health care, utilities, and insurance. In his spare time Jay works as the product review editor for *WebSphere Developer's Journal.*

E-MAIL

jay2ee@hotmail.com

But the big questions were always "Upon what criteria is the code generation based?" and "What is the range of functionality that can reasonably be generated?" The answers to these questions were always, respectively, "very limited" and "extremely narrow." Then came J2EE, proving that in many ways Java was made to generate itself, thanks to introspection, metadata, the JVM, and other features. Now a new type of autocoder is emerging. Not a monolithic code builder, but sets of J2EE component generators strung together in a workflow or a model. Just as did assemblers, then compilers, then JVMs, now component generation is raising the level of abstraction between developers and executable code. How far will this abstraction go? No one knows for sure, but in eventually implementing full J2EE component autogeneration, we may see the limits of human comprehension.

I have used and reviewed several different component generators, and each has features that extend beyond merely generating servlets from XML configuration files. Each of these primarily addresses the issue of J2EE being too complex for the average COBOL, Visual Basic, or PowerBuilder programmer. Java generators open up application programming to novice J2EE programmers, and I think this is important for Java to continue to thrive. Of course, these tools don't do the full job by any means, and Bowstreet Portlet Factory (BPF) is no exception. However, BPF brings several interesting ideas to the table. BPF generates code for a portlet-based architecture, and provides a new slant on component reusability with its idea of multiple profiles for one builder or a set of builders.

BPF itself consists of three main components: the Designer, the Automation Engine, and the Profile Manager.

Designing Portlets

Using the Designer, programmers create models and profiles. This component runs as a plug-in to WSAD, JBuilder, or Eclipse (see Figure 1).

During the course of development, you will use the Designer to create the portlet functionality and run the Portlet Factory's deployment scripts to create a WAR file that contains the portlets you created. You then install the generated portlet WAR as a portlet application using the WebSphere Portal Administration tools. In summary, developing WebSphere portlets with the Portlet Factory consists of these steps:

1. Implement the portlet functionality using Designer to model and profile the application. You can also create customized Configure and Edit functionality with the Designer.
2. Use the WebSphere Portal Administration tools to deploy your portlet by creating a new instance of the Bowstreet Portlet Adapter.

As you make changes to the portlet models, you can log out and log back in to the portal to see the effects of those changes on your portlets.

Use Portlet Factory's Designer to develop the portlet's functionality. The Designer includes several Builders useful in developing portlets for WebSphere, in addition to the more general Builders used for any J2EE application development. These Builders include:

- ***WPS Portlet Adapter:*** Adding this Builder call to a model allows for tighter integration with WebSphere Portal and allows you to define portlet information, such as a name and description, as well as specifying the HTML pages or Portlet Factory models to be used when the user configures or edits the portlet.
- ***WPS Event Declaration:*** Adding this Builder call to a model allows you to declare an event that can be handled through the WebSphere Portal Server's event processes.

Other portlets, whether they are created with the Portlet Factory or not, may handle these events.

- ***Portlet Customizer:*** Adding this Builder call to a model allows you to quickly create custom Configure and Edit models for a particular portlet.

Once you have created the models for your portlets in the Designer, you can deploy them to WebSphere Portal by creating new instances of the Bowstreet Portlet Adapter. Under the Manage Portlets tab in the WebSphere Portal Administration tool, perform a "copy" of the Bowstreet Portlet Adapter for each portlet, and simply specify the name of each model in the Modify Parameters screen. The portlets are instantly available inside the WebSphere Portal. Alternatively, Bowstreet Portlet Factory also provides scripts for packaging portlets by generating traditional portlet WAR files.

The Automation Engine

The Automation Engine is a servlet that handles a request for a model. The first time the Automation Engine gets a request for a model, it generates a Web application according to the profile(s) associated with the request. On subsequent requests for the same model and profile(s), the Automation Engine simply serves the previously generated Web application. For additional requests to the model with different profile associations, the Automation Engine generates new Web applications.

The Profile Set Manager

The Profile Set Manager allows you to create and modify profile sets and profiles (see Figure 2). When you create or open a profile set in the Designer, the Profile Set Manager replaces the view pane.

The Manage Profiles tab allows you to set the profile entry values for the profiles in a profile set. The Manage Profiles tab also provides you with the ability to lock and otherwise manage the profile entry values for the profiles in a profile set.

Developing Portlets

From the developer's point of view, BPF's approach is component based. The core component type is called a "Builder." Builders generate the Java and XML code needed for a specific

Bowstreet Portlet Factory

COMPANY INFO:
200 Ames Pond Drive
Tewksbury, MA 01876
E-mail: info@bowstreet.com
Web: www.bowstreet.com
Toll free: 877 663.2978

SPECIFICATIONS:
Application Server:
IBM WebSphere Application Server

Portal Server:
IBM WebSphere Portal Server

Browsers:
Microsoft Internet Explorer
Netscape Communicator

Operating Systems:
Microsoft Windows 2000
Microsoft Windows NT
Sun Solaris
IBM AIX
Red Hat Linux

IDEs:
JBuilder
Eclipse
IBM WebSphere Studio Family

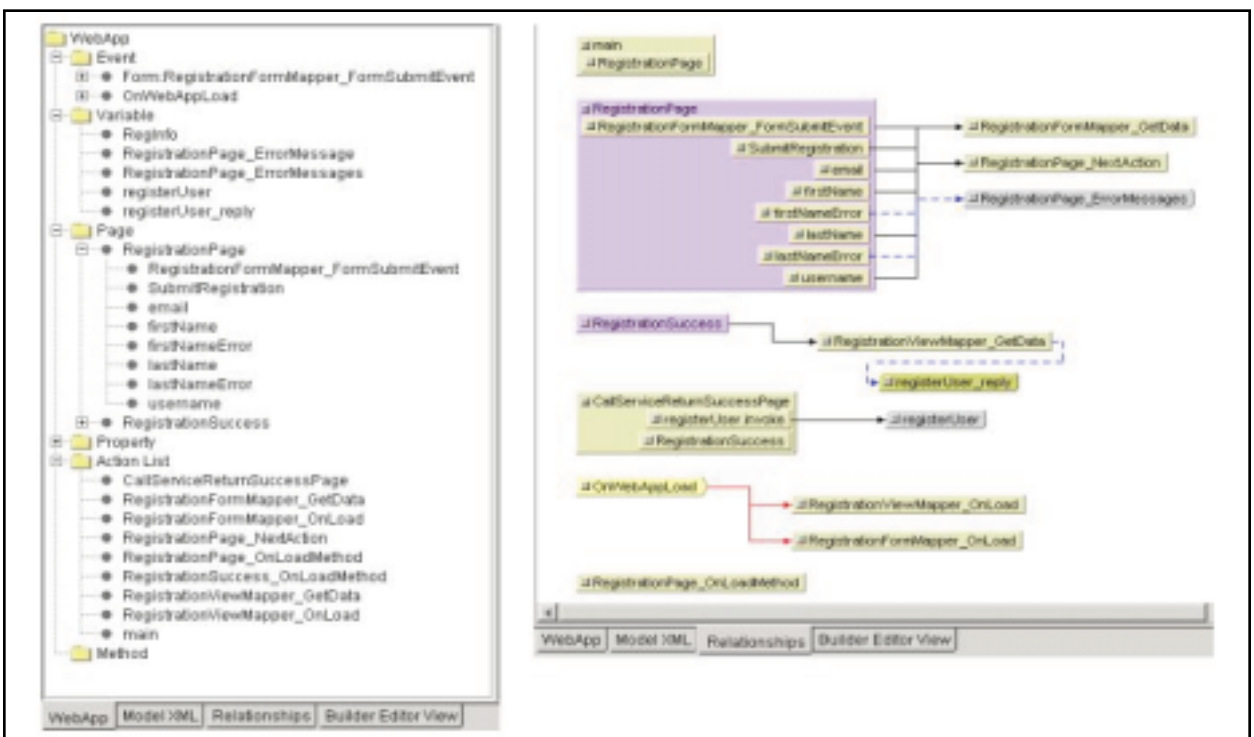


FIG. 1: THE BOWSTREET PORTLET FACTORY DESIGNER

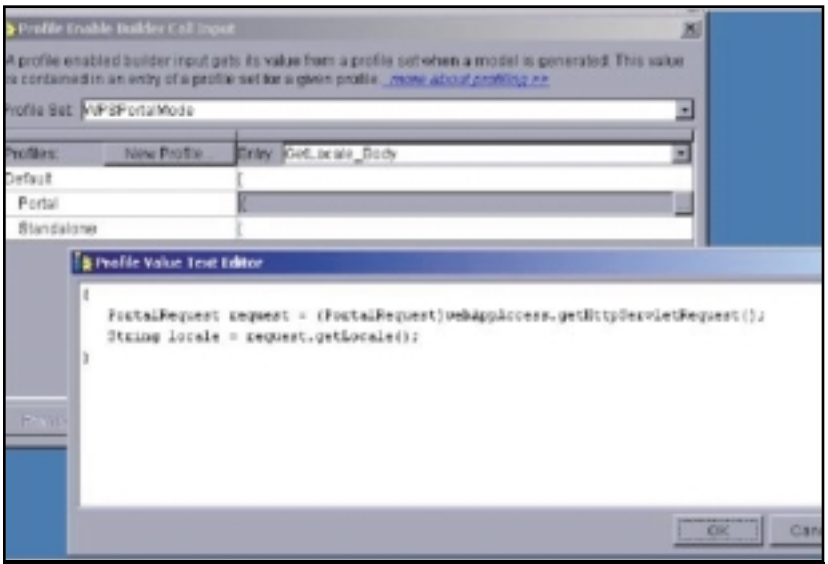


FIG. 2: PROFILE ENABLE DIALOG BOX

application function. Developers can select Builders from a palette, provide parametric values, and launch the Builder to create the code to perform the specified behavior.

Out of the box, BPF provides over 100 Builders. These represent a wide range of functionality, from creating a button to building a Web service call to creating a view portlet connected to a data source. Providing an extensive set of reusable code generators is a great idea, but BPF's ability to combine these into new components is even better.

Of course, someone needs to configure each of these Builders before they are used. This is where Profiles come in. This Profile configuration process may be simple or relatively complex depending on what needs to be generated, but BPF facilitates the idea of combining simple components to create complex functionality. Ideally, a Builder editor would support both a fill-in-the-blanks wizard mode and a more advanced configuration language. BPF provides this feature by allowing a set of Profiles to be associated with each Builder. This is similar in theory to deployment descriptor files for EJBs or instantiations of a C++ template.

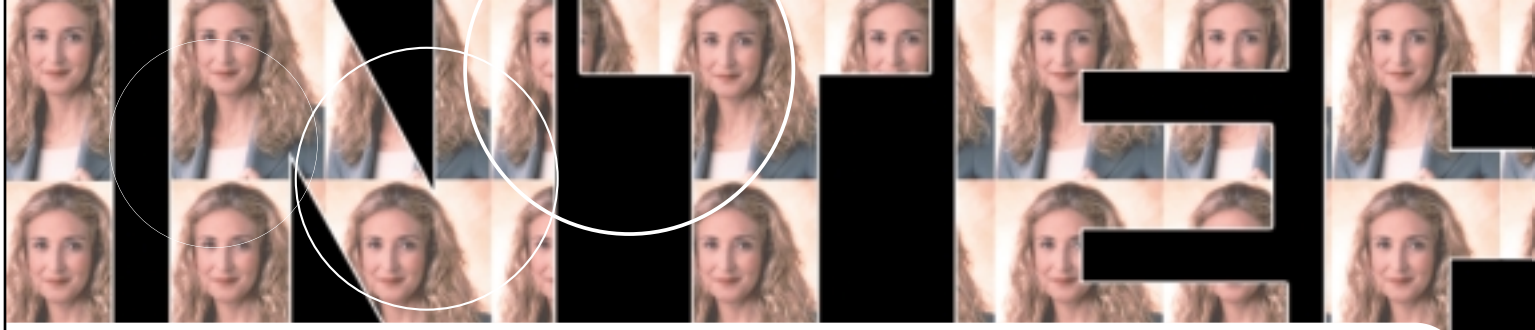
I was impressed that BPF integrates with either JBuilder or WSAD, but Portlet Factory is suitable primarily for application programmers who are relative novices in J2EE development. The work of creating EJB container components must be left to experienced developers using a tool such as WSAD. This work includes transactions, object/relational mapping, connections to legacy applications, and other external data sources.

Portlets are driven by Models. In Bowstreet parlance, an application is made up of Models, each of which is simply an ordered set of Builders that together automate the creation of a portion of an application. For example, an application could have a Model that imports a page, connects to a database, accesses data from the database, formats the data into a table, and displays the table on a Web page. Models and Builders are dynamic. This means that inputs to the Builders are easily changed, and a change to a Builder input automatically propagates through the entire model. For example, if the database schema changes in this model, the JSP that writes the table on the Web page will automatically update.

A Profile is a level of abstraction above, and captures inputs to Builders. When a Model regenerates with a specific Profile, a new version of the application is dynamically generated. In this way, it becomes easy to generate different behaviors from a single Model – for example, to enable customization, user entitlement, or syndication – by creating a new Profile for your application. Profiles can allow people who are not familiar with the details of a Bowstreet Model to supply inputs to the Model in order to change the behavior.

One Model, driven by multiple Profile sets, can drive multiple portlets. As an example, you may wish to allow a marketing manager to create customized versions of your portlet that are co-branded and have unique business processes for each new partner, or to allow sales reps to create a new portlet for each new customer that shows only the products that the specific customer is entitled to purchase. So, Profiles allow multiple variations of a portlet to be easily created from a single code base, helping to make large-grained components reusable.

"Providing an extensive set of reusable code generators is a great idea, but BPF's ability to combine these into new components is even better"



WebSphere Leads Business Transformation with On-Demand Platform

Interview with Jocelyne Attal, vice president of Marketing for IBM's WebSphere software division

Jocelyne Attal is IBM's chief marketing executive for WebSphere, the leading software platform for e-business on demand. Attal focuses on the WebSphere e-business platform, which includes WebSphere Application Server, WebSphere Studio development environment, WebSphere Business Integration, and WebSphere Portal, as well as wireless, commerce, host access, and voice solutions. Under Attal's marketing leadership, WebSphere has become one of the most successful and recognized business software brands.

Having celebrated its fifth anniversary in June, the WebSphere family of products holds the top spot in market share for the application server, portal, and integration markets. Attal, who is a member of the leadership team for IBM's e-business on-demand strategic initiative, recently discussed WebSphere's role in the on-demand environment, and specifically how WebSphere products address the integration, virtualization, and automation aspects of the on-demand environment.

Q: On April 30th IBM announced the next steps in the e-business and on-demand initiatives. Where does WebSphere fit into the on-demand environment?

Jocelyne Attal: E-business on demand is a state of business. On April 30th we announced how we enable business transformation, allowing companies to achieve this state of business through the on-demand operating environment with the capabilities of integration, virtualization, and automation. IBM software, with WebSphere as its core component, provides proven technology customers can use to create and run an on-demand operating environment. WebSphere is designed to enable first, integration of people, process, and information across and beyond the enterprise; second, expanded access to computing resources on demand; third, dynamic building and deployment of new business services that leverage existing business assets.

Q: Please elaborate on the on-demand integration, virtualization, and automation activities and how WebSphere products address these areas.

JA: Integration refers to the integration of people and processes, allowing customers to access any information or application from any device over any network. Elements of this include collaboration, transaction processes, information management, and pervasive computing.

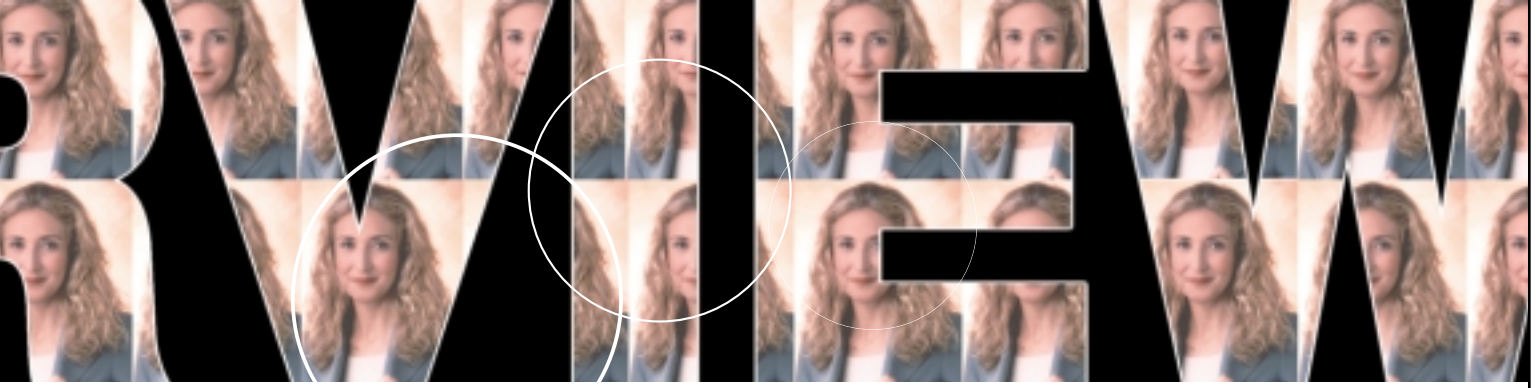
At the core of the on-demand integration capability is WebSphere, which enables integration of applications and automation of business processes, and WebSphere Business Portals, which enable integration of people via collaboration, and integrate access to mission-critical information and business processes.

Virtualization allows the automated consolidation and pooling of resources by creating a logical view of IT assets – including servers, storage, networks, and distributed systems – and virtualizing them in order to better allocate workload and safeguards. WebSphere is at the heart of virtualization because it offers customers the ability to balance workloads and allocate resources on demand for WebSphere business applications with the WebSphere Application Server, delivering clustering capabilities that provide a virtualized, highly scalable, distributed transaction engine.

Automation encompasses availability, security, optimization, and provisioning. For this capability, WebSphere offerings are closely aligned with IBM Tivoli offerings to enable high availability and easy provisioning, all with high levels of security. In addition, WebSphere offers profiling and tuning tools.

Q: How do ISVs fit into the e-business on demand operating environment?

JA: Our top priority is working with the IBM partner ecosystem, which includes the ISV community, to make the on-demand operating environment real for customers. We must partner closely with both large and small ISVs to embed the core capabilities of the operating environment into their applications and industry-specific solutions.



The Developer Relations organization is currently working with 150 ISVs to help them understand the benefits of on-demand computing. For WebSphere ISVs, we have been extremely successful with WebSphere Application Server – Express and our WebSphere Portal – Express and are on track in the registration of our developer program.

Q: What are the direct benefits of e-business on demand for ISVs?

JA: The direct benefits of on demand for ISVs are flexibility, simplification of solution development, and process efficiencies. The IBM on-demand operating environment provides ISVs with a flexible infrastructure because it is based on open standards. The ISVs benefit from building on a strategic, heterogeneous platform that meets the needs of more customers and allows them to enter new markets. Customers will look for an infrastructure that won't lock them up in one model, and this is very important. With on demand, the ISV is offering the customer a set of integrated technologies and capabilities that are flexible, allow end-to-end management, are based on open standards, and are extremely scalable and resilient.

On demand also benefits ISVs because it allows for the simplification of solutions development and integration of their business processes and applications. We are working on a single development platform within IBM that combines the software brands – WebSphere, Rational, Tivoli, DB2, and Lotus – and provides the best tools across the operating environment. Shared components will be implemented in everything we're doing.

The ISVs will realize process efficiencies as well as market flexibility as technology advances evolve in the operating environment, allowing them to seamlessly upgrade their products and solutions to take advantage of new on-demand functionality. ISVs that adopt and build on the on-demand model will provide their customers with next-generation solutions that offer benefits such as reduced costs, improved asset utilization, and the ability to reinvest in new business opportunities.

Q: What are the direct benefits to customers of ISVs?

JA: Customers are facing challenges in the business environment. They have to adapt to changes and at the same time have to continue to invest, grow, and be competitive. They have to look at their processes and make changes, minor and major. They have to become on-demand businesses to respond to an on-demand environment. In order to do that, they need to go through a business transformation. For this we have aligned this integration of the different components for the on-demand operating environment with their business processes and their business policies – not the opposite. This is the number one advantage. With on demand, the customer gets a set of integrated components and capabilities that are extremely resilient, extremely flexible, and based on open standards.

Q: What WebSphere-specific programs are currently available to help ISVs enable on the WebSphere platform?

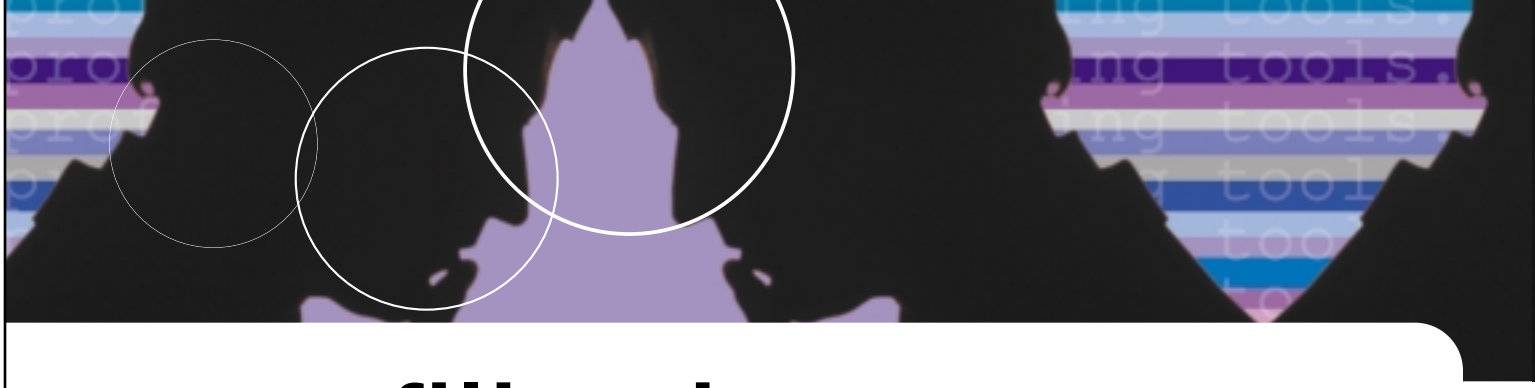
JA: We work directly with Developer Relations, along with the PartnerWorld program, which provides a comprehensive program to address ISV needs, including technical and sales enablement, go-to-market activity, and recognition. Our main objective is to provide the technical content our ISVs are looking for during the deployment of an application. Through PartnerWorld we have a high-volume, Web-based enablement component specifically focused on the midmarket. This gives ISVs access to free education, technical support, and porting assistance for our midmarket products such as WebSphere Application Server – Express and WebSphere Portal – Express. For in-person technical support we work with the IBM Solution Partnership Centers worldwide.

Because we are number one in application servers, integration, and business portals, we are in a good position to help our ISVs gain market exposure through the high market awareness of our brand, and we are, in fact, making this awareness more real through a program called Powered by WebSphere. The Powered by WebSphere Mark allows our partners to associate their solutions with an IBM logo that signifies quality.

Also, the ISVs get marketing and sales support from the IBM Business Partner WebSphere Innovation Centers located around the world, and can access many marketing tools online at www.ibm.com/websphere/businesspartners.

Q: How are you working with the new IBM software brand, Rational?

JA: The IBM goal is not to simply produce complementary tools from WebSphere and Rational. We are creating a well-integrated set of tools that addresses the full life cycle of software development – from modeling and design through coding and testing to quality assurance and deployment to environments like the WebSphere Application Server and pervasive and embedded devices. IBM WebSphere started working with Rational more than three years ago on the Eclipse project. Together, we helped create a new market for Eclipse software tools that now involves more than 150 vendors and has produced software used by millions of developers. Now that they are part of the IBM company, the WebSphere and Rational teams are working more closely than ever to ensure that the Rational tools that help minimize software development risk augment the extendable, multiplatform, and standards-based application development tools for WebSphere.



Profiling in WebSphere Studio 5.0

Part 1: Learning to use the powerful toolset

– BY ANDREW SONDGEROTH –

ABOUT THE AUTHOR

Andrew Sondgeroth has been an instructor and developer for over seven years. He is currently employed with Intertech-Inc. (www.intertech-inc.com), based in Minnesota. Andrew teaches and develops courses on Java-related topics including Enterprise JavaBeans, WebSphere, and Java security. Andrew is visiting faculty on emerging technologies at the University of St. Thomas Graduate School of Business Management Center in Minneapolis/St. Paul, MN. He is also a regular presenter for the Twin Cities Java Users Group.

E-MAIL

asondgeroth@intertech-inc.com

IBM's goliath enterprise tool, WebSphere Studio Application Developer (WSAD) 5.0, has a powerful, full-featured profiling toolset for developers. However, learning how to use the tools and how to interpret the information takes some time.

In Part 1 of this two-part series, I will focus on understanding the information presented by the WSAD 5.0 profiling tools. In Part 2 (in an upcoming issue), I will discuss how to optimize and fix code using the gathered information.

The Purpose of Profiling

In some ways, profiling is similar to debugging. Both are used to examine application flow as well as isolate problems in code. However, debugging is used to check the correctness of code, while profiling is used to inspect the performance of code.

Profiling allows analysis of application behavior in order to recognize how to make the application more efficient. This analysis would be impossible to accomplish by debugging alone.

Profiling has several distinct advantages and uses. It helps identify many behaviors and problems, including:

- Memory leaks/inefficient memory usage
- Poor method response times
- Thread deadlocks
- Frequent code block usage ("hot spots")

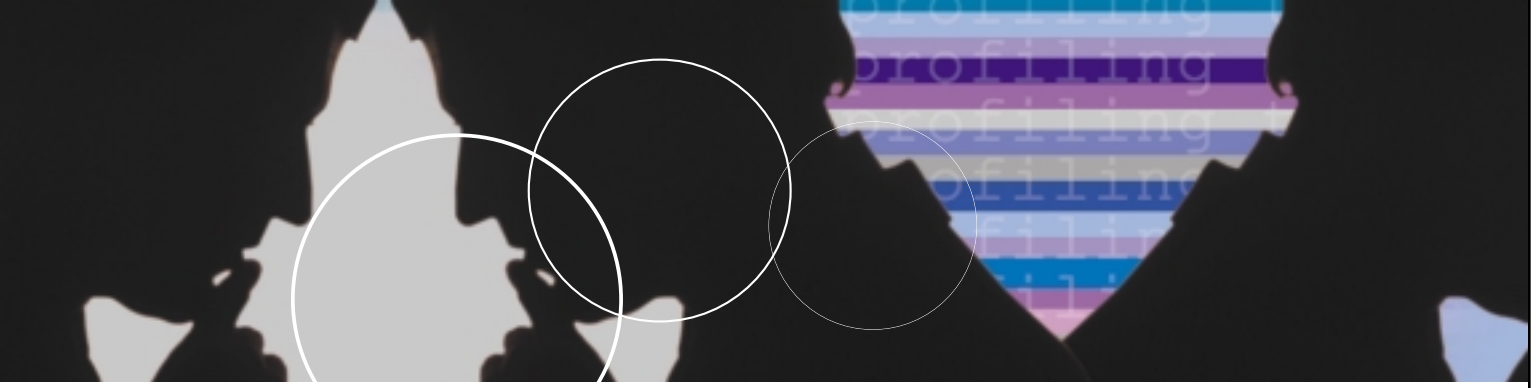
Another benefit of profiling an application is that it can provide early detection of any problems long before production or even QA. Profiling code early on is considered a best practice for application development. Unfortunately, profiling is often done much later in the development life cycle (if at all). Profiling becomes the victim of the common "if we have time" constraint. However, taking the time to profile during proof-of-concept or development milestones helps avoid major architectural changes late in the application life cycle. Thus, significant reworking is prevented, along with bloodshot eyes, last-minute hair-pulling, and the need to gulp bottles of antacids.

Overview of How Profiling Works

Profiling with WSAD involves four major pieces:

1. The JVM
2. An agent running in the JVM (via JVMPI)
3. An agent controller
4. WSAD's Performance Analyzer

Profiling in Java is made possible through Sun's Java Virtual Machine Profiler Interface (JVMPI). A JVMPI implementation (known as the agent) runs as part of a standard JVM. The agent receives and records information from the JVM, such as thread activity and objects being garbage collected. The agent may also make requests for additional information or deactivate further notifications of specific events. The profiling agent is managed by an agent controller, which resides outside of the JVM and acts independently. In the case of WSAD, an extra component, called the Performance Analyzer, acts as an interface for the development environment and the agent controller. The Performance Analyzer can monitor applications locally or on remote machines, as long the remote machine has an agent controller installed and running. Figure 1 illustrates how profiling in WSAD



works. *Note:* Not all JVMPI profilers have an external agent controller.

WSAD allows you to profile applications on JVMs that have the agent controller already installed. This includes applications that are:

- Stand-alone
- Running remotely (including server-side apps)
- Running across multiple JVMs

Data Collection

The profiling process can collect a lot of information, and therefore must keep track of many items, including:

- Instantiated objects
- Garbage-collected objects
- Method calls
- Method execution time
- Memory consumption

Due to the amount of information that must be recorded, profiling can be very slow. Each application reacts differently when profiled. The degree to which performance is affected depends on a number of factors such as connection quality to the remote host, database connectivity, or application complexity. To help optimize the time-consuming process of data collection, we use filters.

FILTERS

Filters allow you to specify which information to profile. For instance, when concerned only with the behavior of classes in a particular package, the agent can be configured to ignore information related to all other packages. Limiting the number of packages to keep track of reduces the amount of data being collected and recorded. Having less data to record helps increase profiling performance. In WSAD, filters are set when using the profiling wizard or by going to Window –> Preferences –> Profiling and Logging –> Profiling. *Note:* Filters are usually set per package – not for individual classes.

EXECUTION TIME

There are two main ways of recording the amount of time spent executing a method: base time and cumulative time. Base time is the actual amount of time that the method spends executing. Cumulative time is the total amount of time that the method spends executing itself, plus the time of all methods that it calls. *Note:* WSAD uses wall time – not CPU time.

The Profiling Perspective

Collecting the data is cumbersome enough, but organizing the information and presenting it in a logical format can be very difficult. WSAD uses over a dozen views to illustrate profiled data. Each view is used to arrange data as a general overview or according to a specific topic. The views in the Profiling Perspective come in two formats: graphical or tabular. Table 1 categorizes the views into their respective formats.

Many of the views correspond to the same exact data but present it in different ways. For example, the Execution Flow view displays a graphical representation of the behavior of all method calls and threads. The Method Invocation Table view uses a table to show the same data but limits it to only data associated with one type of method. Unfortunately, this can cause confusion. Knowing the purpose of each view and interpreting the information displayed is not completely intuitive.

To help familiarize you with all of the views, I will describe what each view does and how it represents data. The descriptions of all the views are organized by the type of data that they correspond to.

Graphical Views

For the most part, each of the graphical representations used is significantly different from the others. However, each format helps you identify relationships and behaviors within your data. I describe each of the views and the data that they represent below.

HEAP VIEW

The Heap view is used to identify objects or methods using the most resources. The information displayed is

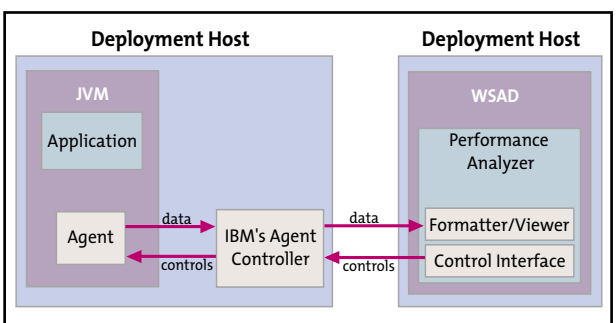


FIG. 1: HOW WSAD AND THE PROFILING AGENTS WORK

Graphical	Tabular
Execution Flow	Class Instance Statistics
Heap	Class Method Statistics
Method Invocation	Execution Flow Table
Method Execution	Instance Statistics
Object References	Method Invocation Table
Sequence Diagram	Method Statistics
	Object Reference Table
	Package Statistics

TABLE 1: PROFILING VIEWS CATEGORIZED BY TYPE

organized by object statistics or method statistics and is the most flexible of the graphical views. Colored blocks in the lower right portion of the view represent objects and method calls. Class names and array types are displayed to the left of their respective colored blocks.

The "Color By" property determines what the color of the blocks means. Each shade of color represents a numeric value. Blue represents the low end of the threshold, and red represents the high end. Numeric values between blue and red are shown by shades that are the result of mixtures of blue and red. Black represents zero and is the only other color used. A slide bar at the top of the view is used to adjust the threshold at which blocks are given a particular color.

For example, when viewing method information, selecting the "Number of Calls" in the Color By property means blue blocks are methods that have been called less than methods represented by red blocks. Assume that red represents 25. Moving the slider down to 15 will readjust the color of the blocks accordingly.

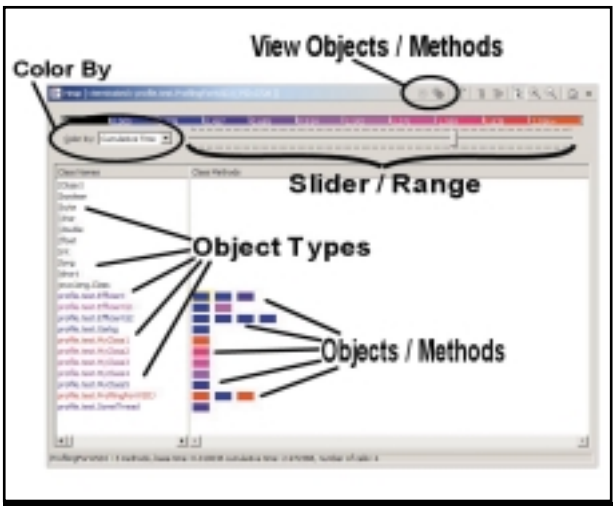


FIG. 2: MAIN FEATURES OF THE HEAP VIEW

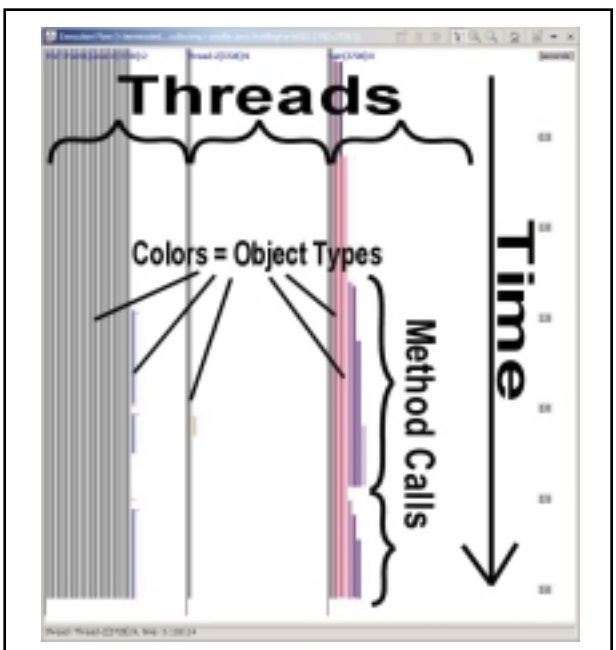


FIG. 3: HOW TO READ THE INFORMATION IN THE EXECUTION FLOW, METHOD INVOCATION, AND METHOD EXECUTION VIEWS

When viewing object information, you have Color By options for:

- Base time
- Cumulative time
- Number of calls
- Number of threads
- Active memory

The Color By choices for method data include:

- Base time
- Cumulative
- Number of calls.

Figure 2 illustrates a labeled example of the Heap view.

EXECUTION FLOW

The Execution Flow view shows all threads in the application. For each thread, vertical bars represent methods invoked. Time is represented vertically. The placement and length of each method bar indicates when the method was called and how long it ran for. By using this view to show all threads, method calls, and the length of method calls, you can get an overview of how your entire application is running. In other words, you can see such details as thread activity, how long method calls are taking, and which ones are taking the longest – all at a single glance. *Note:* Class types can be given unique colors to distinguish what each vertical bar represents.

Figure 3 shows how to interpret the graphical information in the Execution Flow, Method Invocation, and Method Execution views.

METHOD INVOCATION

The Method Invocation view is nearly identical to the Execution Flow view. This view, however, shows more specific data and has the added functionality of navigating method calls. In the Method Invocation view, a single method is examined. Toolbar options allow you to automatically switch the data displayed to:

- The calling method
- A method being invoked by the selected method
- The next time the selected method is called
- The last time the method was called

METHOD EXECUTION

The Method Execution view is similar in nature to the Method Invocation view. The main difference is that the data displayed in this view is basically a compacted overview of the data shown in the Method Invocation view. The Method Execution view displays patterns of functionality. Time has a different meaning than in the other views, representing either the total or average amount of time the method was functioning. Think of this view as an abstract summary of all calls made to a single method.

OBJECT REFERENCES

The Object References view can help identify memory leaks and locations where references to objects are still being held. Because the data displayed in this view is similar to that in the Heap view, the Object References view can be considered a detailed version of the objects shown in the Heap view. The type of objects shown can be nar-

rowed by the choices in the Display dropdown box. Figure 4 shows an example of the Object References view. *Note:* In order to view this data, you need to select "Collect Object References" by right-clicking on the desired process in the Profile Monitor view while profiling.

SEQUENCE DIAGRAM

The Sequence Diagram view uses the Unified Modeling Language (UML) form of sequence diagram to present much of the same data as the Execution Flow view. Method calls, returns, and overall flow of an application can be examined from this view. The Sequence Diagram view can also be used to identify hot spots and thread issues such as deadlocks. Unlike most of the other graphical views, the lengths of vertical bars (representing method calls) do not signify elapsed time. However, method call times can still be viewed by placing your mouse over the method calls.

Tabular Views

All of the tabular views have a similar look and feel to them. In fact, many of them even display some of the same data. The following is a short description of the data presented by the tabular views. Due to the variety of information available in each of the tabular views, Table 2 provides a list of each type of information that is displayed per view.

EXECUTION FLOW TABLE

This view is the tabular equivalent of the graphical Execution Flow view, which displays threads, method calls, method call times, and class types. The Execution Flow Table view displays method information in a tree format. Profiled threads make up the root of each tree. Each method that calls another method makes up another level of the tree that can be expanded or collapsed. When a method node is collapsed, only the start time of the method and total cumulative data are displayed. When the node is expanded, the same type of data is displayed but is also shown for individual statistics of every method called. Figure 5 shows an example of the Execution Flow Table view.

METHOD INVOCATION TABLE

The Method Invocation Table view is the tabular representation of data shown in the Method Invocation view.

OBJECT REFERENCES TABLE

The Object References Table view is the table form of the data presented in the Object References view.

CLASS INSTANCE STATISTICS

The Class Instance Statistics view displays information related to class types. This includes data concerning the number of instances created, amount of memory used, garbage collection, and amount of time spent executing functionality.

INSTANCE STATISTICS

The Instance Statistics view is similar to the Class Instance Statistics view. What makes the Instance Statistics view different is that it lists actual object references. Note that the variable names used in code are not recorded. Instead, each object instance will have its individual statistics displayed.

METHOD STATISTICS

The Method Statistics view simply displays information such as method names, how many times each method was called, and the method stack tree. Of all the information shown, base time and cumulative time for each method are most likely what you will want to use this view for.

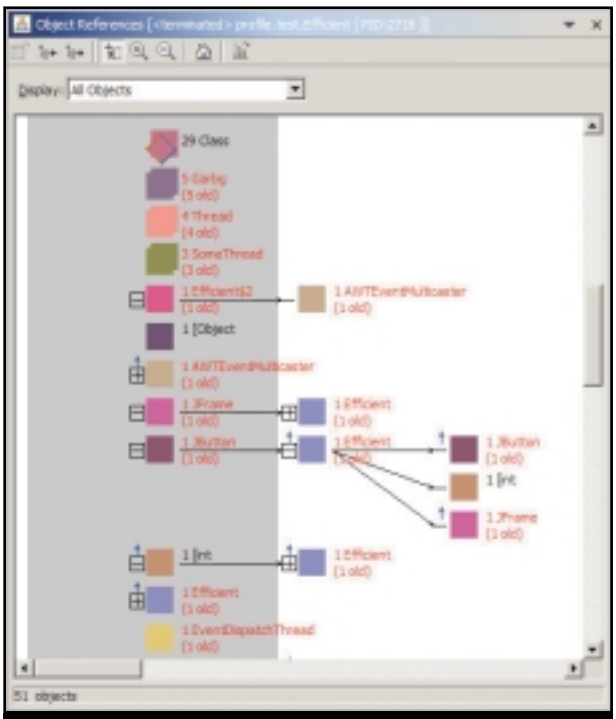


FIG. 4: THE OBJECT REFERENCES VIEW

View Name

Type of Data Displayed	Class Instance Statistics	Instance Statistics	Method Statistics	Class Method Statistics	Method Statistics	Package Statistics	Execution Flow Table	Method Invocation Table
Class/Instance Names	●	●	●	●	●	●	●	
Base Time	●	●	●	●	●			
Cumulative Time	●	●	●	●	●	●		
Package Name	●	●		●	●			●
Method Names			●	●			●	
Average Base Time			●	●				
Calls	●	●	●	●	●			
Total Instances	●				●			
Live Instances	●				●			
Size/Active Size	●				●			●
Collected (Garbage)	●	●						
Instance Size	●	●						
Thread Names						●	●	
Start Time						●	●	
Reference Names								●
Number of References								●

TABLE 2: TYPES OF DATA DISPLAYED IN THE TABULAR VIEWS

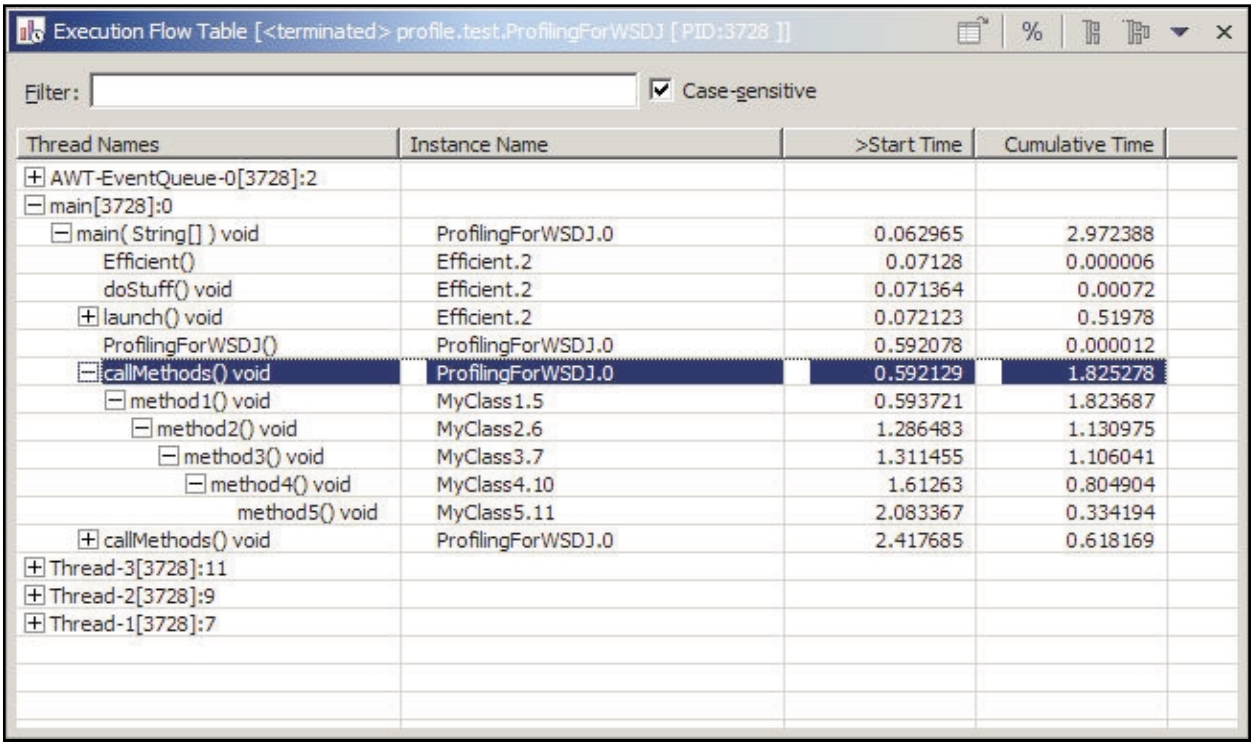


FIG. 5: THE EXECUTION FLOW TABLE

CLASS METHOD STATISTICS

The Class Method Statistics view displays method information organized by class, and is nearly identical to the Method Statistics view. Other than being organized by class type, the only difference between the two views is that this view also displays package information.

PACKAGE STATISTICS

The Package Statistics view allows you to examine information at a package level. This lets you take a high-level glance at what groups of classes may be causing problems. Once identified, you may expand the information of a specific package to possibly identify a class (or classes) that may be causing performance issues. Otherwise, you can use the other views to more accurately pinpoint the source of problems.

Log Views

There are three other views worth mentioning: Activity Log, Default Log, and Message Log. Logging views are used to view errors, exceptions, and messages that may occur during profiling. Log files can further help identify performance issues and architectural problems. WSAD also allows log files to be used in conjunction with a symptom database (an XML file with explanations and possible solutions to information contained in a log file). *Note:* Log files by default can be found at: "[yourWorkspace]/.metadata/LoggingUtil[someID].log".

Conclusion

As you can see, WSAD 5.0's profiling tools give us a plethora of facts and information about our application. We looked at how WSAD presents all the information collected, as well as how the information is collected behind the scenes. By now you can already see what an advantage profiling has over hundreds of System.out.println()s.

Next time we will look at using WSAD's profiling tools to identify and fix problems in our code.

References

- WSAD 5.0 Help Documentation
- *WebSphere Studio Application Developer Version 5 Programming Guide (Redbook SG24-6957-00):* www.redbooks.ibm.com
- *Self-Study Guide: WebSphere Studio Application Developer and Web Services (Redbook SG24-6407-00):* www.redbooks.ibm.com

Things to Consider...

- Java does not allow profiling and debugging to be used concurrently.
- In order to profile a remote application, the agent controller must be installed and running on the remote machine.
- Upgrading to the WSAD v5.0.1 Fixpack takes care of a number of memory/performance–related issues when using WSAD 5 for profiling.
- Some views may need to be refreshed before the most current data will be displayed.
- Be aware of potential firewall issues when attempting to attach to a remote process.
- If you encounter a StackOverflowError when profiling, try disabling the Just-in-Time compiler.

Java Served Here...

March 2003 Issues:

22%

...of all Java pages in March were published by the other Java magazine publisher

78%

...of all Java pages in March were served by SYS-CON Media

SYS-CON MEDIA

Miles Silverman
VP Marketing

Carmen Gonzalez
Senior VP Marketing

Contact | Carmen Gonzalez: carmen@sys-con.com or 201 802.3021

JAVA DEVELOPER'S JOURNAL

Migrating WebSphere

Enabling WebSphere Studio Application Developer Integration Edition v4.11 JCA applications to run in WebSphere Application Server Advanced Edition v4.04

– BY SANDY MINOCHA –

ABOUT THE AUTHOR

Sandy Minocha works for the WebSphere Studio Jumpstart ISV Enablement team at the IBM Toronto Software Lab. The team engages with key WebSphere Studio tools partners, providing technical guidance and support as they integrate into the WebSphere Studio set of products. The team also develops the integration criteria for the Ready for WebSphere Studio partner program. Sandy holds a bachelor's in electrical engineering from the University of Waterloo and a master's in Telecommunications from the University of Toronto, and is a PEng.

E-MAIL

minocha@ca.ibm.com

Since the release of IBM VisualAge for Java Enterprise Edition v3.53, the Enterprise Access Builder (EAB) tools have been able to generate code that conforms to the J2EE Connector Architecture (JCA) specification, which allows Java programs to access Enterprise Information Systems (EIS), such as SAP, CICS, IMS, and other back-end systems, using a standard interface.

IBM WebSphere Studio Application Developer Integration Edition (hereafter called Integration Edition) provides the next-generation tooling for enterprise connectivity. It features a WebSphere Test Environment that targets the powerful WebSphere Application Server Enterprise Edition.

A fix pack is now available that enables you to run JCA applications developed in Integration Edition on the WebSphere Application Server Advanced Edition (hereafter called Advanced Edition). This article describes best practices for migrating your Integration Edition JCA applications to Advanced Edition.

Flow-Based JCA Applications

Business Process Flow technology is an important piece of the middleware stack that has recently been offered as part of the WebSphere platform. IBM has been offering flow technology in a number of products (most notably WebSphere MQ Workflow for people-based workflows, WebSphere MQ Integrator for message flows, WebSphere MQ Adapter Offering for MQ-based adapters, and Enterprise Access Builder for Java-based adapters), but no tight integration of flow technology has been available in the WebSphere platform until now. The pure-Java flow engine, written as a WebSphere enterprise service, allows you to tightly integrate flow technology with all the other services offered by J2EE.

Integration Edition provides visual flow-based tools in order to visually define the sequence and flow of information between application artifacts such as application adapters, JavaBean and enterprise bean components, Web services, or other flows. In Integration Edition v4.11, only noninterruptible, short-running, and synchronous flows are supported. This type of flow is referred to as a microflow or a service flow. Interruptible, long-running, and asynchronous flows are being planned for future releases of Integration Edition.

Flows can only be run in the WebSphere Application Server Enterprise Edition. Therefore, it is important to note that this article is only relevant to non-flow-based JCA applications.

Overview

CREATE, DEPLOY, AND TEST THE APPLICATION IN INTEGRATION EDITION

Integration Edition v4.11 contains a local copy of the complete runtime environment of the WebSphere Application Server Advanced Single Server Edition (hereafter called Advanced Single Server Edition) v4.02 with enhancements to support JCA and some features of the WebSphere Enterprise Extensions. This environment is called the WebSphere Test Environment. The JCA tooling combined with the Test Environment provides you with a complete environment to develop and test your JCA applications. The high-level steps to creating, deploying, and testing the JCA application in Integration Edition are depicted in Figure 1.

DEPLOY AND TEST THE APPLICATION IN ADVANCED SINGLE SERVER EDITION

The release of Fix Pack 4 makes it possible to take JCA applications developed in Integration Edition and deploy and execute them in Advanced Edition. You simply export the JCA application from Integration Edition as an Enterprise Archive EAR file and then use the Administrative Console to deploy it into Advanced Edition. However, before moving your JCA application directly to an Advanced Edition

application server, it is recommended that you test the JCA application in the steps outlined in this article in order to make the porting process as smooth as possible.

In addition to testing JCA applications in the Test Environment's local copy of the WebSphere Application Server, Integration Edition provides the capability to test your JCA applications on a remote copy of the WebSphere Application Server product. Consequently, by configuring an Advanced Single Server Edition with Fix Pack 4, you can use Integration Edition to remotely deploy and run your JCA application. WebSphere Application Server Advanced Single Server Edition is required since the Test Environment does not support it. The high-level steps to deploying and testing the JCA application in Advanced Single Server Edition are depicted in Figure 2.

DEPLOY AND RUN THE APPLICATION IN ADVANCED EDITION

Once you have verified that the JCA application executes successfully in Advanced Single Server Edition, it is ready to be deployed in Advanced Edition. The high-level steps to deploying and running the JCA application in Advanced Edition are depicted in Figure 3.

The rest of this article takes you through a real example, the Phone Book IMS application included in Integration Edition.

Getting Started

The following software is required. Follow the installation instructions included with the software.

- WebSphere Studio Application Developer Integration Edition v4.11
- WebSphere Application Server Advanced Edition (AE) v4.01, v4.02, or v4.03
- WebSphere Application Server Advanced Single Server Edition (AEs) v4.01, v4.02, or v4.03
- WebSphere Application Server v4 Fix Pack 4 (v4.04) for AE
- WebSphere Application Server v4 Fix Pack 4 (v4.04) for AEs
- DB2 Universal Database v7.2 Fix Pack 5, 6, or 7
- Microsoft Windows 2000 Service Pack 1 or 2, or Windows NT 4.0 with Service Pack 6a or higher

To apply Fix Pack 4, go to the WebSphere Application Server Support link, www.ibm.com/software/webservers/appserv/support.html. In the Search this product field, type "Version 4.0.4" and click Submit. This will bring up a link to the Fix Pack 4 download site.

When installing the Fix Pack for Advanced Edition, ensure that you run the install -connectors command if you have not already installed the J2C (J2EE Connector) runtime support in the application server.

To install the J2C runtime support in Advanced Single Server Edition, copy the following JAR files from the <AdvancedEdition_Installdir>/lib directory| to the <AdvancedSingleServerEdition_Installdir>/lib directory:

- recjava.jar
- jca.jar
- j2c.jar
- eablib.jar
- ccf2poolmgr.jar
- ccf2.jar
- ccf.jar

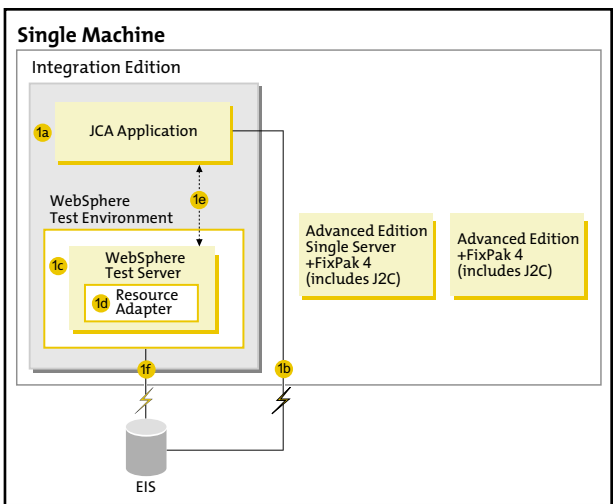


FIG. 1: THE HIGH-LEVEL STEPS TO CREATING, DEPLOYING, AND TESTING THE JCA APPLICATION IN INTEGRATION EDITION

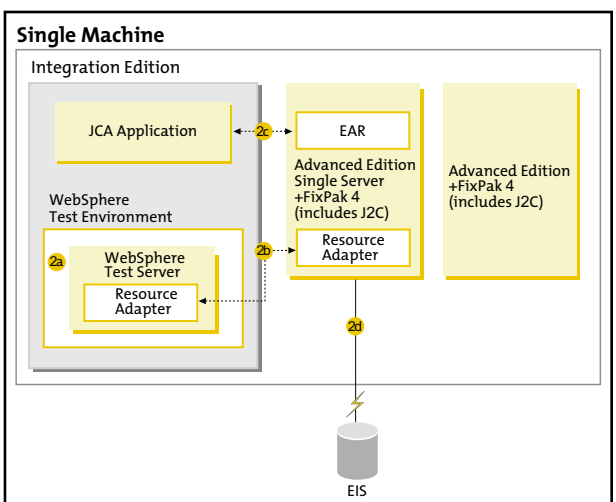


FIG. 2: THE HIGH-LEVEL STEPS TO DEPLOYING AND TESTING THE JCA APPLICATION IN ADVANCED SINGLE SERVER EDITION

Create, Deploy and Test the Application in Integration Edition

CREATE THE APPLICATION

Integration Edition includes a sample that documents how to build, deploy, and run the Phone Book IMS application in the Test Environment. Since the intent of this article is to show how JCA applications can be deployed and executed in Advanced Edition, this sample is not built but loaded into the workbench instead. To install the Phone Book IMS application in the workbench, complete the following steps:

Install the Phone Book IMS Application

1. Start Integration Edition. The workbench launches the Enterprise Services perspective (see Figure 4).
2. Select File –> New –> Other.
3 Expand Examples from the left frame and select Enterprise Services Samples.
4. From the right frame, select Phone Book.
5. Click Next.

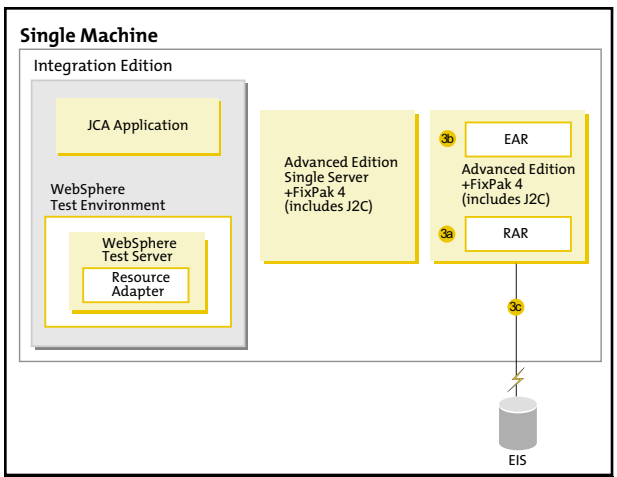


FIG. 3: THE HIGH-LEVEL STEPS TO DEPLOYING AND RUNNING THE JCA APPLICATION IN ADVANCED EDITION

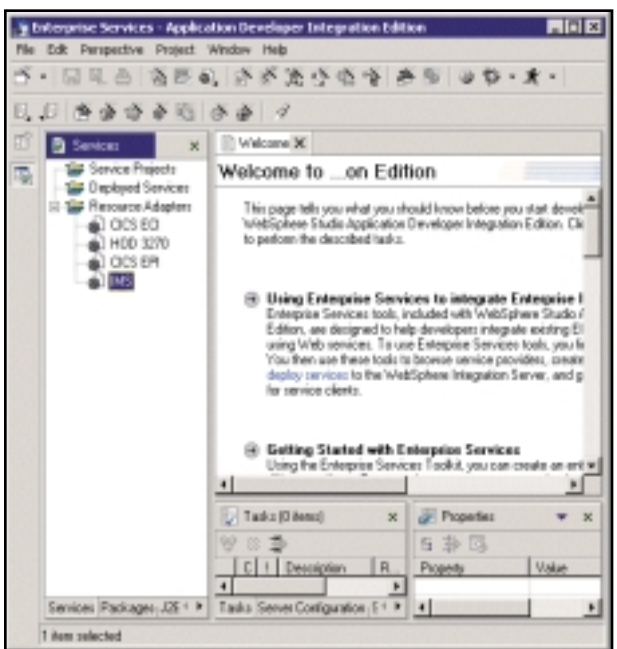


FIG. 4: THE ENTERPRISE SERVICES PERSPECTIVE

6. Click Finish. This will load the Phone Book IMS sample in the workbench. It may take several minutes. Once the sample has been loaded, the Help perspective opens and you will see four errors in the task list, which can be ignored.
7. Go back to the Enterprise Services perspective.
8. To remove the list of referenced libraries from this view, go to the top right-hand corner of the Packages view, click the Menu icon and ensure that Show Referenced Libraries is unchecked.

TEST THE APPLICATION

Before the Phone Book IMS application is deployed and executed, it is a good idea to verify that the Phone Book IMS application was installed correctly, and to verify connectivity to IMS.

Configure Connection Information in the Phone Book IMS Application

1. From the Enterprise Services perspective, click the Packages tab. Expand the IMSSample project, sample.ims package, and double-click PhoneBookIMS.wsdl. PhoneBookIMS.wsdl will open in the WSDL editor.
2. In the Services section, click PhoneBookIMSService.
3. In the Ports section, click imsPort.
4. Click Extensibility Elements.
5. Select IMSAddress. The Extensibility Element Properties appear (see Figure 5).
6. Type the appropriate connection information for your IMS system.
7. Close the editor and click Yes to save your changes.

Run the Java Client Application

The Phone Book IMS application includes a fat Java client application that communicates directly with the IMS system.

1. In the same sample.ims package, select TestPhoneBookIMS.java class and expand the Run icon on the toolbar by selecting the arrow beside it. From the pop-up menu, select Run –> Java Application.
2. Verify that the output is similar to that shown in Figure 6.

ADD A LOCAL WEBSPHERE TEST SERVER

The Test Environment uses server instances and server configurations to test J2EE applications. Server instances identify servers that can test J2EE applications, whereas server configurations contain setup information. Server instances and server configurations are created using the Create a New Server Instance and Configuration wizard.

Create a Server Instance and Server Configuration

1. From the Enterprise Service perspective, click the Server Configuration tab to open the Server Configuration view. Right-click Server Configurations. Select New –> Server Instance and Configuration. The Create a New Server Instance and Configuration wizard opens.
2. In the Server name field, type ServicesServer.
3. In the Folder field, type ServicesServer.
4. Expand WebSphere Servers from the Server instance type list.
5. Select WebSphere v4.0 Test Environment. Leave the template set to None. Click Next.

6. Click Yes to create a ServicesServer server project.
7. Click Finish. The new server instance appears in the Server Configuration view and in the Servers view. The server configuration appears in the Server Configuration view.

INSTALL AND CONFIGURE THE RESOURCE ADAPTER

Integration Edition comes with four resource adapters: CICS ECI, CICS EPI, IMS, and HOD 3270. In order for a WebSphere Test Server to use these resource adapters, they must be installed and configured appropriately:

1. Expand Server Configurations in the Server Configuration view.
2. Double-click the server configuration ServicesServer. The server configuration editor opens.
3. Click the J2C tab.
4. Click Add beside the J2C Resource Adapters pane. The Create Resource Adapter window opens.
5. In the Resource Adapter Name dropdown list, select IMS.
6. Click OK.

To add a connection factory:

1. Click Add beside the J2C Connection Factories pane.
2. In the Create Connection Factory window, click Name in the Name column, and then click in the Value column beside Name. Type ims_cf.
3. Click JNDI name in the Name column, and then click in the Value column beside JNDI name. Type "myIMS".
4. Click on another field to set the new values, and then click OK.
5. In the Resource Properties pane, complete the necessary resource properties for your IMS system (see Figure 7).
6. Close the editor and click Yes to save the changes.

DEPLOY THE APPLICATION

Add the JCA Application to the Local WebSphere Test Server

Deploying a JCA application involves the simple step of adding the EAR project to the server configuration:

1. In the Server Configuration view, expand Server Configurations and right-click ServicesServer.
2. Select Add Project –> IMSClient.

TEST THE JCA APPLICATION

Start the Local WebSphere Test Server

1. Click the Servers tab to open the Servers view.
2. Under Server Instance in the Servers view, right-click ServicesServer, and then select Start from the pop-up menu. The server status should change to "Started". (If it doesn't, click the Console tab and look for exceptions or load module errors.)

Run the Java Client Application

The Phone Book IMS application includes a Java client application that invokes the application's session bean.

1. From the Enterprise Services perspective, click on the Packages tab. Expand the Services_CLIENT project, appClientModule folder, and sample.ims.client package.
2. Select TestPhoneBookEJB.java and expand the Run icon on the toolbar by selecting the arrow beside it. From the pop-up menu, select Run –> Java Application.
3. Verify that the output is similar to that shown in Figure 8.
4. Under Server Instance in the Servers view, right-click ServicesServer, and then from the pop-up menu, select Stop. The server status should change to "Stopped". (If it doesn't, click the Console tab and look for exceptions.)

At this point, you have completed building your JCA application and verified that it executes successfully with your EIS. The next step is to test the same JCA application in a different WebSphere Test Environment.

Deploy and Test the JCA Application in Advanced Single Server Edition

ADD A REMOTE WEBSPHERE TEST SERVER

The process of adding a remote WebSphere Test Server to your Test Environment is quite similar to adding a local test server.

Create a Server Instance and Server Configuration

1. From the Enterprise Service perspective, click the Server Configuration tab to open the Server Configuration

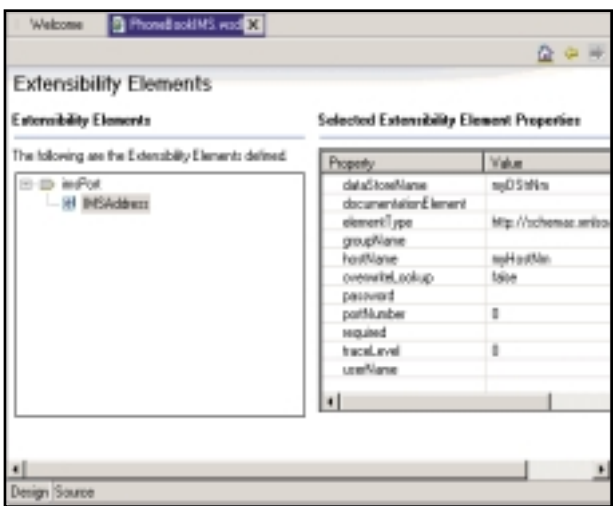


FIG. 5: THE EXTENSIBILITY ELEMENT PROPERTIES

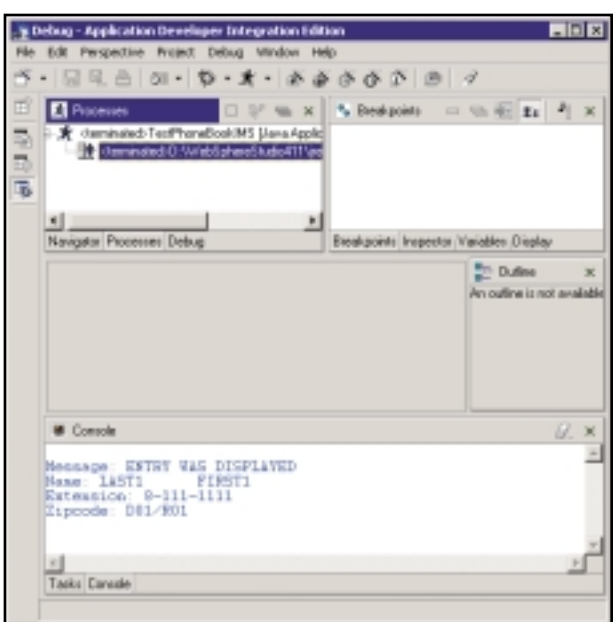


FIG. 6: RUNNING THE JAVA CLIENT APPLICATION

view. Right-click Server Configurations. Select New –> Server Instance and Configuration. The Create a New Server Instance and Configuration wizard will open.
2. In the Server name field, type AEs.
3. In the Folder field, select ServicesServer.
4. Expand WebSphere Servers from the Server instance type list.
5. Select WebSphere v4.0 Remote Server. Leave the template set to None.
6. Click Next (see Figure 9).
7. In the WebSphere installation directory field, type the path where Advanced Single Server Edition is installed.
8. Click Next twice.
9. In the Remote target directory field, type the path of the Advanced Single Server Edition seen on the local machine. Most likely, the Single Server Edition is installed on the same physical machine as the Integration Edition. This means the remote target-directory will be same as the WebSphere installation directory.
10. Click Finish. The new server instance appears in the Server Configuration view and in the Servers view. The server configuration appears in the Server Configuration view.

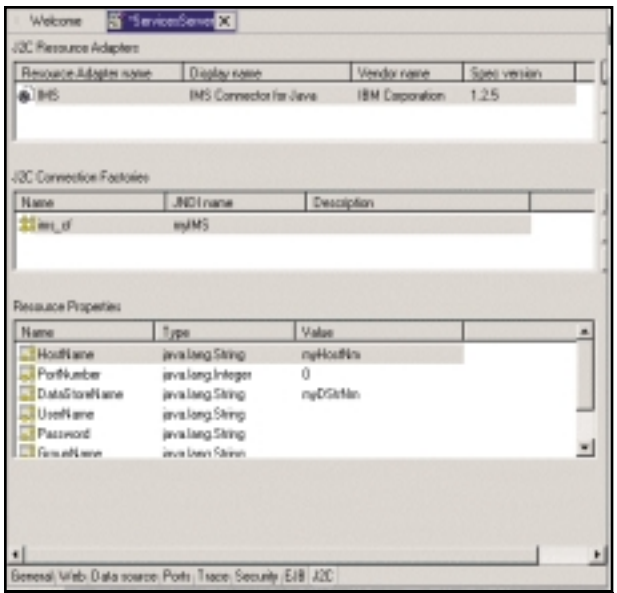


FIG. 7: THE RESOURCE PROPERTIES PANE

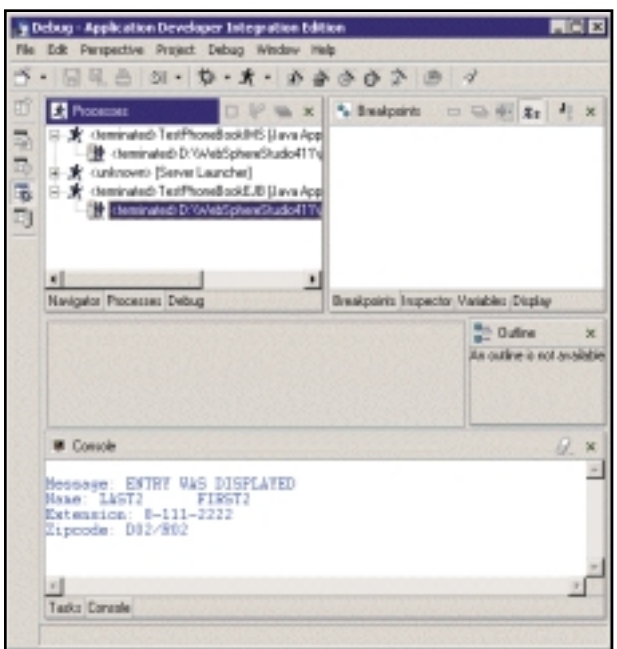


FIG. 8: YOUR OUTPUT SHOULD BE SIMILAR TO THIS

INSTALL AND CONFIGURE A RESOURCE ADAPTER

Installing and configuring resource adapters in a remote test server is identical to installing and configuring resource adapters in a local test server. Consequently, follow the steps as outlined above to install and configure a resource adapter in a local test server.

If the remote test server is not on the same physical machine as the Integration Edition, then additional steps are needed to configure the resource adapter. If this is the case, then see the "Additional Steps to Configure a Resource Adapter in a Remote Test Server" section in the "Hints and Tips" section of this article.

DEPLOY THE JCA APPLICATION

Add the JCA Application to a Remote WebSphere Test Server

Deploying a JCA application to a remote WebSphere Test Server involves adding the EAR project to the server configuration, which is done as follows:

1. In the Server Configuration view, expand Server Configurations and right-click AEs.
2. Select Add Project –> IMSClient.

TEST THE JCA APPLICATION

Start the Remote WebSphere Test Server

1. Click the Servers tab to open the Servers view.
2. Under Server Instance in the Servers view, right-click AEs, and then from the pop-up menu, select Start. The server status should change to "Started". (If it doesn't, click the Console tab and look for exceptions or load module errors.) In the Console window, you will notice that the Advanced Single Server Edition with the applied Fix Pack 4 will start up (see Figure 10).

Run a Java Client Application

The same Java client application used earlier with the local test server is invoked here.

1. From the Enterprise Services perspective, click on the Packages tab. Expand the Services_CLIENT project, appClientModule folder, and sample.ims.client package.
2. Select TestPhoneBookEJB.java and expand the Run icon on the toolbar by selecting the arrow beside it. From the pop-up menu, select Run –> Java Application.
3. Verify that the output is similar to that running in the local test server.

At this point, you have verified that your JCA application executes successfully with your EIS in an environment similar to Advanced Edition. The final step is to move the JCA application over to the Advanced Edition.

Deploy and Run a JCA Application in Advanced Edition

The Advanced Edition does not come with the resource adapters that a JCA application requires to access enter-

prise information systems using SAP, CICS, IMS, and other back-end systems. You must obtain the appropriate Resource Adapter Archive (RAR) file for your EIS before you can execute your JCA application in Advanced Edition.

In the case of the Phone Book IMS sample, you can obtain the IMS RAR file from the IMS Connector for Java link, www.ibm.com/software/data/ims/about/imsico/downloads.htm.

The WebSphere Application Server InfoCenter contains extensive documentation on the application server, including how to install resource adapters, how to create JCA connection factories and how to install EAR files. Consequently, some of the instructions below are not detailed. See www.ibm.com/software/webservers/appserv/infocenter. html for more details on these tasks.

INSTALL AND CONFIGURE A RESOURCE ADAPTER

Start the Administrative Server

1. From a command window, type the command admin-server to start the Administrative server. Once the message "Server __adminServer open for e-business" appears in the command window, the server is up

Start the Administrative Console

Click Start –> Programs –> IBM WebSphere –> Application Server V4.0 AE –> Administrator's Console to open the Administrative Console.

Install a Resource Adapter

1. In the Administrative Console, select New –> J2C Resource Adapter. The J2C Resource Adapter Properties window comes up.
2. In the Name field, type ims_ra.
3. Beside the Archive file name field, click the small square button to open the Install Driver page.
4. Select the node that you want to use to install the resource adapter.
5. Beside the RAR file field, click Browse, and then navigate the file system and select the IMS RAR file.
6. Click Install.
7. Click OK.

Add a Connection Factory

1. From the Console menu, select New -> J2C Connection Factory. The J2C Connection Factory wizard appears.
2. In the J2C resource adapter dropdown list, select ims_ra.
3. In the Name field, type "ims_cf".
4. In the JNDI binding path field, type "myIMS".
5. Click the Connections tab and type the property values appropriate to your environment.
6. Click OK.

DEPLOY THE JCA APPLICATION

Export an EAR File

The Integration Edition includes an Export EAR wizard that conveniently packages a JCA application into an EAR file. This EAR file can then be installed in Advanced Edition using the Administrative Console.

1. From Integration Edition, select File –> Export. The Export wizard opens.
2. In the left frame, select EAR file.
3. Click Next.
4. Click Finish.

Install EAR File

Use the Administrative Console to install the EAR file in Advanced Edition. Since the EAR file was packaged with all the deployment information in Integration Edition, installing it into the application server is very simple.

1. From the Administrative Console menu, select Wizards –> Install Enterprise Application. The Install Enterprise Application wizard opens.
2. In the Path field, click Browse, navigate to your EAR file, and select it. Click Open.
3. Click Next 10 times to accept the defaults.
4. Click Finish. When a message appears that asks you if you want to regenerate code, click No.

RUN THE JCA APPLICATION

Start the Default Application Server

1. In the left frame of the Administrative console, expand WebSphere Administrative Domain, Nodes, your_node_name, and Application Servers.
2. Right-click Default Server and select Start.

Run the Java Client Application

The same Java client application used with the local and remote test servers in earlier is invoked here.

1. From a command window, type "launchclient< Advanced Edition_Installdir>/installed Apps/ServicesEAR.ear".
2. Verify that the output is similar to that shown in Figure 11.

Congratulations! You have now successfully deployed and executed a JCA application in Advanced Edition.

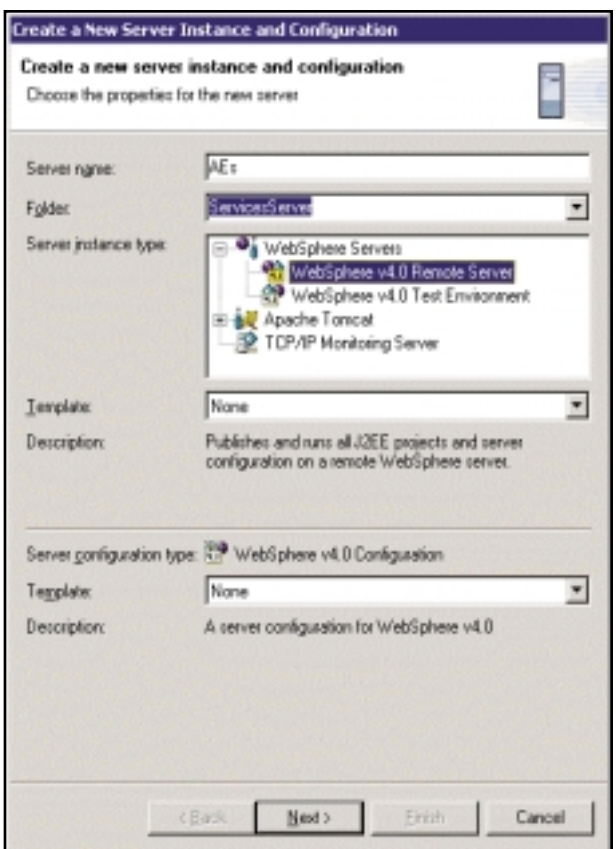


FIG. 9: CREATING A NEW SERVER INSTANCE AND CONFIGURATION

Hints and Tips

PROBLEMS WITH LAUNCHING THE REMOTE TEST SERVER

The most likely scenario is that you already had Agent Controller installed on a machine and then you installed Advanced Single Server Edition on the same machine. In this case, you must edit the settings in the server configuration file after you install Advanced Single Server Edition:

1. Open the serviceconfig.xml file, which is located in the <Agent_Controller_Installdir>/config directory.
2. In the file, search for the Application executable="wte Remote.exe" tag. Change all occurrences of

```
path="%WAS_HOME%"
```

within the tag to

```
path="AdvancedSingleServerEdition_Installdir"
```

3. Save your changes and close the file.

It is also possible that Agent Controller is not running. Agent Controller is a daemon process that provides the mechanism by which client applications either launch new host processes or attach to agents that coexist within existing host processes. Agent Controller is automatically installed with Integration Edition as a Windows service. You do not have to provide any information (such as the installation path and JAVA_HOME environment) when you install Agent Controller with Integration Edition. Integration Edition will provide all necessary information to Agent Controller. Check your list of Windows services and ensure that the Agent Controller service is running.

Additional Steps to Configure a Resource Adapter in a Remote Test Server

Additional steps to configure a resource adapter in a remote test server are necessary when the remote test server is on a different physical machine from the Integration Edition. This is due to a limitation of Integration Edition's Test Environment for publishing resource adapters to the remote test server. When the remote test server is on the same physical machine as Integration Edition, this limitation is transparent because the Test Environment configures the remote test server to point to the pre-installed resource adapters. If the remote test server is on a different machine, then the configuration settings for the resource adapters become invalid because the directory structure will be different on the two machines. In order to configure a resource adapter in the remote test server that is on a different physical machine from Integration Edition, follow these steps:

1. Navigate to the directory <IntegrationEdition_Installdir> \workspace\Installed Resource Adapters. This directory contains all your installed resource adapters, including IMS.
2. Copy the appropriate group of JAR files to the machine where Advanced Single Server Edition is installed.
3. Open the server-cfg.xml file, which is located in the <IntegrationEdition_Installdir>\workspace\<Server_project>\<Server_project>.wsc directory.
4. In that file, search for the installedResourceProviders tag. Modify its classpath attribute to point to the location where the JAR files were moved to in Step 2.

Note: The Integration Edition v4.11 licensing agreement permits you to copy the resource adapter files to Advanced Single Server Edition and use them in conjunction with the Integration Edition product for development and testing purposes only. For additional information, see the licensing agreement located in the <IntegrationEdition_Installdir>\license directory.

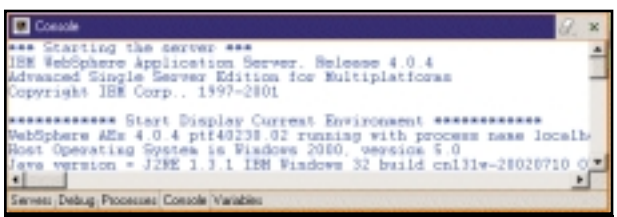


FIG. 10: ADVANCED SINGLE SERVER EDITION STARTING UP

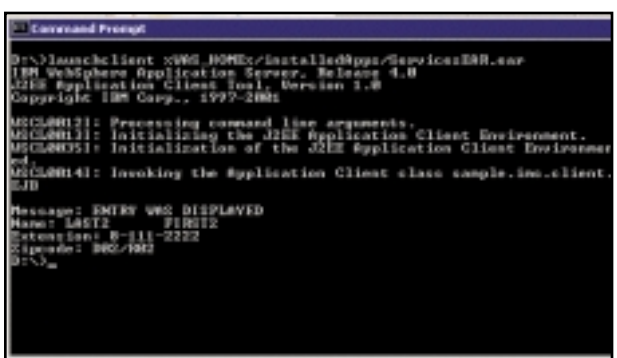


FIG. 11: YOUR OUTPUT SHOULD BE SIMILAR TO THIS

Conclusion

The target platform for applications developed with Integration Edition v4.11 is Enterprise Edition. However, the release of the WebSphere Application Server v4 Fix Pack 4 makes it possible to deploy and run JCA applications in Advanced Edition. This article provides best practices for moving your non-flow-based JCA applications to Advanced Edition.

References

- *WebSphere Studio Application Developer Integration Edition:* www.ibm.com/software/ad/studiointegration
- *Running JCA Applications in Advanced Edition:* www-1.ibm.com/support/docview.wss?uid=swg27002269&aid=1
- *WebSphere Application Server Support:* www.ibm.com/software/webservers/appserv/support.html
- *WebSphere Application Server InfoCenter:* www.ibm.com/software/webservers/appserv/infocenter.html
- *IMS Connector for Java:* www.ibm.com/software/data/ims/about/imsico/downloads.htm
- *Connector Architecture:* http://java.sun.com/j2ee/connector
- *WebSphere Adapters:* www-3.ibm.com/software/ts/mqseries/adapter/ws
- *IBM WebSphere V4 Advanced Edition Handbook:* http://publib-b.boulder.ibm.com/Redbooks.nsf/ 9445fa5b416f6e32852569ae006bb65f/f1d70f465728f28e85256b7300549ca6?OpenDocument&Highlight=0,WebSphere,Handbook

International Web Services Conference & Expo

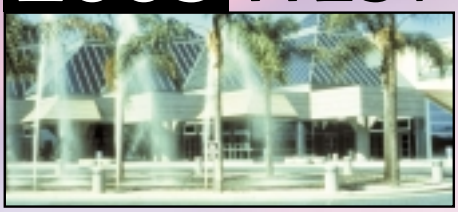
Web Services Edge

3rd Annual

web services conference EDGE &expo

Delivering .NET, JAVA, Mac OS X, & XML Technologies

2003 WEST



SEPT. 30 - OCT. 2, 2003

Santa Clara, CA

web services conference EDGE &expo

REGISTER TODAY!

CALL 201-802-3058

www.sys-con.com

Register before September 1st and

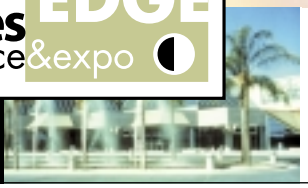
SAVE Up To $200

Produced and Presented: SYS-CON EVENTS

Media Sponsors: WebServices Journal, Java Developer's Journal, WebSphere Developer's Journal, XML Journal, .NET Developer's Journal, WebLogic Developer's Journal, LinuxWorld Magazine, wireless Business Technology, Linux Business Week, CF Advisor, ColdFusion Developer's Journal, PowerBuilder Developer's Journal, market WIRE, HSP Street.com, OASIS, SD Times, LinuxWorld.com, SAMS, Real Industry

WEB SERVICES EDGE CONFERENCE & EXPO

web services EDGE conference&expo



SEPT. 30 - OCT. 2, 2003

Santa Clara Convention Center

It is our pleasure to bring the latest edition of the highly successful Web Services Edge Conference & Expo to the Santa Clara Convention Center, September 30 – October 2, 2003. The Third Annual Web Services Edge West Conference & Expo 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 West 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 dedicated tracks covering Java, Web Services, .NET, and XML - along with the newly added Mac OS X Track. Sessions in this track will highlight the use of the Mac OS X platform, which combines the ease of use of a Mac with the power of Unix, in applications and Web services development, deployment, and management.

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 September 30 and October 1, 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
- Keynotes & Panel Discussions from Industry Leaders
- 60 Hard-hitting and Informative Seminars
- FREE .NET Tutorial with Russ' Tool Shed
- Java University Certification Training
- Industry-Leading Certification Programs
- "Birds of a Feather" Discussions
- Round Table Discussions
- Opening Day Welcome Reception
- SAMS Meet the Authors Hot Topics Lounge
- Compelling Case Studies & Best Practices
- Hands-On Labs
- Featured Product Demonstrations
- Exhibit Floor featuring more than 40 companies and hundreds of products
- Real-time SYS-CON Radio Interviews

WHO SHOULD ATTEND

- Software Developer
- Software Engineer
- Development Manager
- Application Developer
- Technical Director
- Analyst/Programmer
- IT Manager
- Technical Architect
- Team Leader
- Software Consultant

HIGHLIGHTED SPEAKERS

Allan Vermeulen

CTO, Amazon.com

CTO and vice president at Amazon.com directly oversees the Platform Technologies group. This group is responsible for guiding Amazon.com's technology architecture, including building and acquiring foundational components. Prior to his move to Amazon.com, Vermeulen was CTO and vice president of development at Rogue Wave Software. He holds a PhD in Systems Design Engineering from the University of Waterloo.

John Schmidt

Leader of Systems Integration and Middleware, Best Buy Co.

John Schmidt is the chairman of the Methodology Committee for the EAI Industry Consortium and leader of systems integration and middleware at Best Buy Co., a leading specialty retailer of consumer electronics, personal computers, entertainment software, and appliances.

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. Dave has also been published in numerous technical journals, and is currently writing a series of contributed articles for *Java Developer's Journal*.

Anne Thomas Manes

Research Director, Burton Group

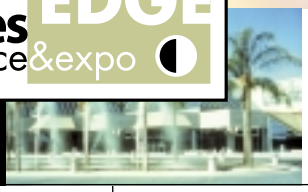
Anne Thomas Manes is a research director at Burton Group, a research, consulting, and advisory firm. Anne leads research for the Application Platform Strategies service. Named one of NetworkWorld's "50 Most Powerful People in Networking" in 2002, and one of Enterprise Systems Journal's "Power 100 IT Leaders" in 2001, Anne is a renowned technologist in the Web services space. Anne participates in standards development at W3C and OASIS. She is a frequent speaker at trade shows and author of numerous articles and the book *Web Services: A Manager's Guide*.

Register Online at

www.sys-con.com

WEB SERVICES EDGE CONFERENCE & EXPO

web services EDGE conference&expo



SEPT. 30 - OCT. 2, 2003

Santa Clara Convention Center

WEB SERVICES TECHNOLOGY

Presentations will include discussions of security, interoperability, the role of UDDI, progress of the standards-making bodies, SOAP, and BPM. Case studies cover the design and deployment of Web services in the marketplace.

Sessions will focus on:

- Interoperability
- Enterprise Networks
- Web Services Management
- Web Services Standards
- Web Services Orchestration
- Security (WS-Security, SAML)
- BPEL4WS
- UDDI: Dead or Alive?
- ebXML & Web Services
- EAI & Web Services
- RPC vs Documents: Uses and Differences
- User Interfaces for Web Services
- Web Services Best Practices
- Service Oriented Architecture

JAVA TECHNOLOGY

The Java Track features presentations aimed at the beginner, as well as the seasoned Java developer. Sessions will explore the whole spectrum of Java, focusing on J2EE, application architecture, EJB, and J2ME. In addition the track will cover the latest in SWT, Ant, JUnit, open source frameworks, as well as an in-depth look into the vital role that Java is playing in building and deploying Web services.

Sessions will focus on:

- Enterprise Java 1.4
- Ant Applied in "Real World" Web Services
- Developing Application Frameworks w/SWT
- Empowering Java and RSS for Blogging
- JUnit: Testing Your Java with JUnit
- JDK1.5: The Tiger
- Simplifying J2EE Applications
- Using IBM's Emerging Technologies Toolkit (ETTK)
- Apache Axis
- Meeting the Challenges of J2ME Development
- Integrating Java + .NET
- Squeezing Java

XML TECHNOLOGY

Presentations will focus on the various facets of XML technologies as they are applied to solving business computing problems. Sessions will include emerging standards in XML Schemas, XML repositories, industry applications of XML, applying XML for building Web services applications, XML/XSLT/XQuery-based programming using Java/.NET, XML databases, XML tools and servers, XML-based messaging, and the issues related to applying XML in B2B/EAI applications. The XML Track is geared for audiences ranging from beginners to system architects and advanced developers.

Sessions will focus on:

- XML Standards & Vocabularies
- Introduction to XForms
- Securing Your XML and Web Services Infrastructure
- XQuery Fundamentals: Key Ingredient to Enterprise Information Integration
- XML and Enterprise Architecture: Technology Trends
- Standards-Based Enterprise Middleware Using XML/Web Services
- XML and Financial Services
- Canonical Documents for Your Business: Design Strategies
- XPath/XSLT 2.0: What's New?
- XML Schema Best Practices
- XML in EAI, Enterprise Portals, Content Management

.NET TECHNOLOGY

Presentations will explore the Microsoft .NET platform for Web services. To the average developer, it represents an entirely new approach to creating software for the Microsoft platform. What's more, .NET development products - such as Visual Studio .NET - now bring the power of drag-and-drop, GUI-based programming to such diverse platforms as the Web and mobile devices.

Sessions will focus on:

- ASP.NET
- Security
- VB.NET
- .NET and XML
- Smart Device Extensions for VS.NET
- Best Practices
- Shared Source CLI
- .NET Remoting
- Smart Devices in Health Care Settings
- Mobile Internet Toolkit
- ROTOR
- Portable .NET
- ASP.NET Using Mono
- Using WSE with IBM's WSTK
- GUI applications Using Mono
- Portals – Windows SharePoint Services/SharePoint Portal Server
- Windows Server 2003 and IIS 6
- .NET and Java Interoperability
- Distributed .NET for Financial Applications
- Developing C# with Eclipse

MAC OS X

OS X represents a new wave of operating systems. It combines the ease of use of a Mac with the power of Unix. Sessions in this track will highlight the use of the Mac OS X platform in applications and Web services development, deployment and management.

Sessions will focus on:

- Introducing OS X (Panther): What's New?
- Quick Applications using AppleScript
- Enterprise Java and OS X
- Developing Web Services Using WebObjects
- Xserve: Ease of OS X and Power of Unix
- Introducing Quartz: 2D Graphics for Apple
- OS X for the Unix Developer
- Securing OS X Applications
- Java and OS X: A Perfect Marriage
- Programming Rich User Interfaces Using Cocoa

SPECIAL DISCOUNTS AVAILABLE

Take advantage of the Early Bird and Pre-registration values available right now, or save even more with a group of 5 or more. For special group discounts contact Michael Lynch at mike@sys-con.com, or by phone at (201) 802-3058.

For more information visit
www.sys-con.com
or call
201 802-3069

PRODUCED BY SYS-CON EVENTS

Conference at-a-Glance

		JAVA	.NET	WEB SERVICES	XML
TUESDAY, SEPTEMBER 30 DAY 1	8:00AM – 4:00PM	**REGISTRATION**			
	9:00AM – 9:50AM	Enterprise Java 1.4	Using WSE 2.0	Web Services Management	Introduction to Xforms
	10:00AM – 10:50AM	Opening Keynote - Allen Vermeulen, CTO, Amazon.com			
	11:00AM – 6:00PM	**EXPO OPEN**			
	2:00PM – 2:50PM	Keynote Panel Discussion - Enterprise Application Integration			
	3:00PM – 3:50PM	Ant Applied in "Real World" Web Services	Smart Devices in Health Care Settings	Service Oriented Architecture	Securing Your XML and Web Services Infrastructure
	4:00PM – 4:50PM	Developing Application Frameworks with SWT	Using the Mobile Internet Toolkit	Web Services Orchestration	XQuery Fundamentals: Key Ingredient to Enterprise Information Integration
	5:00PM	**OPENING NIGHT RECEPTION**			
WEDNESDAY, OCTOBER 1 DAY 2	8:00AM – 4:00PM	**REGISTRATION**			
	9:00AM – 9:50AM	Integrating Java and .NET	Introduction to ROTOR	Security (WS-Security, SAML)	Standards-Based Enterprise Middleware Using XML/Web Services
	10:00AM – 10:50AM	Morning Keynote			
	11:00AM – 4:00PM	**EXPO OPEN**			
	2:00PM – 2:50PM	Keynote Panel Discussion - Interoperability: Is Web Services Delivering?			
	3:00PM – 3:50PM	JUnit: Testing Your Java with JUnit	Using Portable .NET	WS-BPEL	XML and Enterprise Architecture: Technology Trends
	4:00PM – 4:50PM	JDK1.5: The Tiger	ASP.NET with Mono	UDDI: Dead or Alive?	Using XML Schemas Effectively in WSDL Design
	5:00PM – 6:00PM	Squeezing Java	Using WSE with IBM's WSTK	Web Services Choreography, Management, and Security - Can They Dance Together?	Canonical Documents for Your Business: Design Strategies
THURSDAY, OCTOBER 2 DAY 3	8:00AM – 4:00PM	**REGISTRATION**			
	9:00AM – 9:50AM	Using IBM's Emerging Technologies Toolkit (ETTK)	Distributed .NET for Financial Applications	eAI & Web Services	XML and the Fortune 500
	10:00AM – 10:50AM	Morning Technical Keynote			
	11:00AM – 11:50AM	Apache Axis	Developing C# with Eclipse	RPC vs Documents: Uses and Differences	XPath/XSLT 2.0: What's New?
	12:00PM	**BREAK**			
	1:00PM – 1:50PM	Meeting the Challenges of J2ME Development	Windows SharePoint Services	The Seven Habits of Highly Effective Enterprise Service Buses (ESBs)	ebXML & Web Services
	2:00PM – 2:50PM	Keynote Panel Discussion - Summit on Web Services Standards			
	3:00PM – 3:50PM	Empowering Java and RSS for Blogging	BizTalk 2003	See www.sys-con.com for more information	See www.sys-con.com for more information
	4:00PM – 5:00PM	See www.sys-con.com for more information	See www.sys-con.com for more information	See www.sys-con.com for more information	See www.sys-con.com for more information

MAC OS X

	Introducing OS X (Panther) What's New?
	Programming Rich User Interfaces Using Cocoa
	Quick Applications using AppleScript
	Java and OS X: A Perfect Marriage
	Enterprise Java and OS X
	Developing Web Services Using WebObjects
	Cocoa, Carbon, Java: Application Frameworks for OS X (When to use what)
	Securing OS X Applications
	Xserve: Ease of OS X and Power of Unix
	OS X for the Unix Developer
	Introducing Quartz: 2D Graphics for Apple
	See www.sys-con.com for more information

PROGRAM SUBJECT TO CHANGE

Microsoft .net

FREE Microsoft® Tutorial

RUSS' TOOLSHED

TINKERING WITH VISUAL STUDIO

RUSS' TOOL SHED

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

INTRO TO WEB SERVICES USING VS.NET

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.

ADVANCED WEB SERVICES USING ASP.NET

This session 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 the .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.

.NET REMOTING ESSENTIALS

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 lifetime of remoted objects, and how to secure remoting applications.

Register Online at

www.sys-con.com

REGISTRATION FORM

CONFERENCE: Sept. 30 – Oct. 2, 2003 EXPO: Sept. 30 – Oct. 1, 2003

Santa Clara Convention Center • Santa Clara, CA

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 8/1/03	Before 9/26/03	Onsite
❑ **GP Gold Passport** Good for all three days of the .NET, Web Services, XML, Java, and Mac OS X Tracks, including Keynotes, Panel Discussions, preferred seating for Microsoft .NET's Russ'"Tool Shed" Tutorial, and your choice of one Sun Microsystems Java™ University Class	$1,195.00	$1,395.00	$1,495.00
❑ **3D Three Day Conference** (Does not include Sun Java™ Education)	$1,095.00	$1,295.00	$1,395.00
❑ **2D Two Day Conference** (Does not include Sun Java™ Education) (select any two days: ❑ Tue. ❑ Wed. ❑ Thurs.)	$995.00	$1,195.00	$1,295.00
❑ **1D One Day Conference** (Does not include Sun Java™ Education) (select any one day: ❑ Tue. ❑ Wed. ❑ Thurs.)	$495.00	$595.00	$695.00
❑ **JU1 Sun Java™ University Class** Select one: ❑ Web Services Programming Using Java™ Technology and XML (Sept. 30) ❑ Java™ Fast Path: Programmer (Oct. 1) ❑ Java™ Fast Path: Architect (Oct. 2)	$595.00	$695.00	$795.00
❑ **JU2 Sun Java™ University Class** Select two: ❑ Web Services Programming Using Java™ Technology and XML (Sept. 30) ❑ Java™ Fast Path: Programmer (Oct.1) ❑ Java™ Fast Path: Architect (Oct. 2)	$1,095.00	$1,295.00	$1,395.00
❑ **JU3 Sun Java™ University Class** Select three: ❑ Web Services Programming Using Java™ Technology and XML (Sept. 30) ❑ Java™ Fast Path: Programmer (Oct. 1) ❑ Java™ Fast Path: Architect (Oct. 2)	$1,195.00	$1,395.00	$1,495.00
❑ **EO Expo Only**	FREE	FREE	$50.00

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 Programmer
❑ 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]

Web Services Edge WEST 2003
SEPT. 30 - OCT. 2, 2003
Santa Clara Convention Center
web services conference&expo EDGE
WEB SERVICES EDGE CONFERENCE & EXPO

If you require special assistance covered under the Americans with Disabilities Act, please call 201-802-3058 by September 16, 2003.

CANCELLATIONS, SUBSTITUTIONS, REFUNDS
Fax written request to SYS-CON Registration 201-782-9651. Requests for refunds received prior to August 29, 2003 will be honored, less a 10% handling charge; requests received after August 29, 2003, and before September 12, 2003, will be honored less a 20% handling charge. No requests for refunds will be honored after September 12, 2003. Requests for substitutions must be made in writing prior to September 26, 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.

An IT solution for the energy and utilities industry

WebSphere Business Integration

BY DAVID SAMUEL

Outage. That word is a sharp stick in the eye of the energy and utilities industry. But outages are a way of life for all utilities, especially in geographies that are prone to bad weather. In fact, utilities spend a significant amount of their time and resources maintaining physical assets and recovering from outages when they occur.

ABOUT THE AUTHOR

As general manager of IBM's Global Energy and Utilities industry, David Samuel provides leadership to a diverse IBM team that offers solutions at the intersection of business and technology to the energy industry. In addition, he is accountable for the marketing programs and thought leadership that enable the industry business.

Restoring service is not enough, however. In addition to restoration activities, utilities must communicate the status of an outage and the expected recovery time to emergency departments (police, fire, and hospitals), affected customers, the media, local government, and the public utilities commission. Poor communication has often turned a relatively small-scale outage into a full-scale public relations nightmare. And in today's deregulated, competitive environment, utilities cannot afford to let that happen, or to be ranked low on their ability to manage service outages because of communication lapses.

Unfortunately, today's outage-management process, characterized by a mix of labor-intensive, voice-based operations, is not designed for efficiently communicating recovery activities to all of the utility's constituencies. Low-tech communications management can often be a far more daunting task than managing the physical recovery.

While it's true that most utilities have automated communications between customer-service reps and outage management systems, many have not extended that automation to the field staff. Further, outage management is generally not integrated into an overall information and communications system that can track the recovery process in real time and communicate that progress to all constituents. And when people are kept out of the loop during an emergency, they tend to assume the worst – an impression that lasts even when the emergency has passed, human nature being what it is.

Integration solutions, like IBM's WebSphere Business Integration for Energy and Utilities solutions, will help companies integrate different areas of their business and those of their customers and partners – automating the flow of information and the processing of transactions across an entire value chain and transforming their businesses to become more agile and responsive.

Most companies have 11 or more core applications that need to be integrated to drive an improvement in business performance (see Figure 1). Disconnected business processes are inefficient, expensive, and require a large, specialized workforce to handle multiple steps – making it harder to share staff across functional areas.

Business process integration can help reduce the costs of service, asset operation, and asset maintenance by streamlining processes and leveraging investments in existing systems. Business integration accelerates the execution of business processes while giving employees more complete and accurate information in their interactions with customers. The result is quicker, more effective customer service requests and problem resolution, with fewer errors, reduced customer callbacks, and greater customer satisfaction.

IBM's business integration software for energy and utilities markets provides a comprehensive integration platform for straight-through processing. It addresses many of the industry's pain points through the delivery of value propositions.

1. ***Reduce costs of enabling customer switching:*** WebSphere Business Integration (BI) for Energy and Utilities can help reduce the cost of supporting customer switching by providing an electronic data exchange infrastructure for fast connectivity, improved audibility and management, and standardization of communications between the parties.
2. ***Reduce time required to recover from outage:*** WebSphere BI for Energy and Utilities can help speed the recovery from outages by automating data flow from the field and from remote pervasive sensors, and by automating the integration of systems.
3. ***Reduce costs of settlement and dispute resolution:*** WebSphere BI for Energy and Utilities industries can

help reduce the cost of settlement and dispute resolution with connections to multiple infrastructures by consolidating connectivity through a single network hub.

4. ***Improve customer fulfillment process:*** WebSphere BI for Energy and Utilities can help reduce the cost of providing customer services such as tree trimming or turn on/off services by integrating key systems and providing core capability to schedule work at the time it is requested.
5. ***Improve visibility of enterprise risk and drivers of EPS:*** WebSphere BI for Energy and Utilities can help increase the visibility of risk through system integration and reduce the risk associated with corporate actions by automating complex corporate actions' data streams and reducing the likelihood of mishandled events.

WebSphere Business Integration also addresses integration needs around the processes used to support operations, manage and maintain assets, and provide customer service. For example, using an electrical power outage management system helps to decrease the time required to restore utility services when an outage occurs. It also improves communications to the media, public officials, and customers regarding the status of an outage and the expected recovery time.

With WebSphere Business Integration, IBM delivers an effective way to redesign processes and integrate those processes with operational applications, data, remote user devices, and physical assets. This integration environment will effectively add intelligence to distribution networks in ways that will enable fact-based decision-making in real time. These technologies help make the utilities business and IT infrastructure more integrated and automated, and can result in the utilities reducing operating costs, improving customer service, and extracting more value out of operational assets.

Integrated outage management can accomplish the following for a typical electric or utility company:

- Customers can be updated on outage status on a timely basis.
- Keeping customers in the loop means that call centers will not be overwhelmed with inquiries.
- Public officials and the media can be apprised of recovery times on a continual basis.
- Information can flow from the field to the outage-management team in real time, eliminating the need for data entry.
- With accurate, up-to-date information, the utility can manage communications rather than being victimized by rumors and speculation.
- Pervasive sensors can be deployed that publish information to a utility, enabling it to discover future outages when they occur.

Such improvements not only enable the utility to burnish its image with its constituents, the efficiencies of integration may result in reducing the cost of recovery as well.

Integrating outage management is a four-step process for most utilities. First, the organization must design an outage management model, simulating processes that are tailored to the utility. Second, it must link all of its management, information, and field systems into a single, integrated IT process. Third, it must ensure flexible, transparent communications with its customers, suppliers and subcontractors, public officials, emergency facilities, and the media. And fourth, it must establish command, control, and review processes to manage the system and make improvements and upgrades as warranted.

Key to the success of an integrated outage-management system is its ability to interact fluidly with other systems, to build easily upon existing IT platforms, and to be adaptable enough to allow for future trauma-free change. Integration software – "middleware" – operating on open-standards platforms will

WebSphere Developer's Journal Coming Next Month...

EAI

Demystifying the Art of Integrating Enterprise Applications:

Linking disparate systems

BY TROY HOLMES AND JAY HOTALING

UNIT TESTING

Stable Code Is a Good Thing:

J-Unit testing in WSAD 5

BY KULVIR SINGH BHOGAL

ILLUMINATING WEBSPHERE

The ValueListHandler Design Pattern:

Improving J2EE application performance

BY LLOYD HAGEMO AND RAVI KALIDINDI

WSAD

Profiling in WebSphere Studio 5.0:

Part 2 – Identifying and fixing problems

BY ANDREW SONDGEROTH

ensure a system that can be built upon an existing IT infrastructure and can communicate seamlessly with the outside world, today and tomorrow.

What will an outage scenario look like with integrated management? It will be considerably less chaotic than is currently the case. From the first customer call reporting an outage, information will flow to all involved parties. Field crews will have wireless devices that will enable them to keep data streaming to the outage-management team, which can then issue progress reports and assessments with virtually no time lag. Everyone will know what's going on; and with efficient communications damage control, the team will be able to focus on their primary responsibility: repairing the physical damage.

Sound like Utopia? It's not – it's simply one example of a vision of what IT integration can do by ensuring that all processes work together to create open, real-time communications when they're needed most. After all, it's in the best interests of the utilities industry not to keep people in the dark.

In summary, IBM WebSphere Business Integration helps energy and utility companies:

- Provide sophisticated industry-specific process integration capabilities that reduce operating expenses by removing process inefficiencies and failures out of key operations.
- Integrate best-of-breed and legacy applications quickly using industry-specific adapters to enable easier and more cost-effective application maintenance and upgrades.
- Enable a single Web-channel interface for all users, regardless of the access method.

IBM teams have a broad base of technical expertise in the energy and utilities industry, and are familiar with the industry's key business issues and processes. Moreover, IBM integration products are designed to work with industry-specific applications, data standards, and business processes. This openness translates into faster, easier, and dramatically more efficient results from integration projects. WebSphere Business Integration for Energy and Utilities is enabled by the following IBM software portfolio:

- WebSphere Business Integration Server is the process integration hub that helps organizations automate and integrate business processes using secure and scalable technology to accelerate e-business initiatives.
- WebSphere Business Integration Adapters help customers achieve business responsiveness by rapidly integrating applications, technologies, industry standards, data sources, and partner systems. Adapters are integral to a scalable, secure integration infrastructure that improves business performance and supports e-business goals.

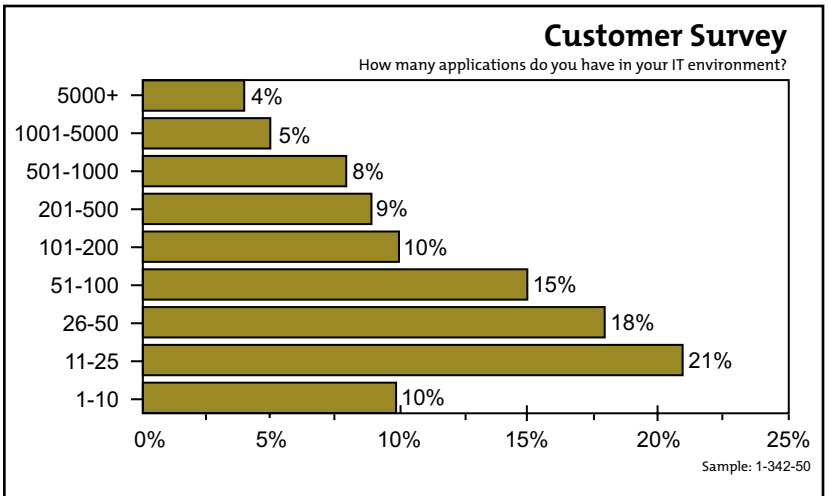


FIG. 1: MOST COMPANIES HAVE 11 OR MORE CORE APPLICATIONS THAT NEED TO BE INTEGRATED TO DRIVE AN IMPROVEMENT IN BUSINESS PERFORMANCE

- WebSphere Business Integration Collaborations are prebuilt process templates that help accelerate the integration of common business processes that span multiple applications.
- WebSphere MQSeries Workflow aligns and integrates an organization's resources and capabilities with its business and e-business strategies. MQ Workflow drives accelerated Business Process Management and enhances business responsiveness, service-level management, and the reuse of business services.
- WebSphere Business Integration Message Broker is a powerful information broker that routes, transforms, and enriches in-flight messages between applications. It is able to handle multiple transports, such as WebSphere MQ messaging, WebSphere MQ Everyplace for pervasive devices, and telemetry integration for remote sensory and control devices.
- WebSphere Business Integration Modeler is used to define, model, analyze, simulate, and report business processes extending IBM WebSphere Business Integration MQSeries Workflow with business tooling to visualize process impact for today's competitive global marketplace.
- WebSphere Business Integration Monitor provides a real-time quantified view of business processes and output, allowing you to track automated business processes and displaying key metrics via convenient dashboards to check the pulse of company performance.

WebSphere Business Integration for Energy and Utilities leverages the IBM middleware portfolio and provides a comprehensive approach that allows for message transformation, message routing, and assured message delivery. It manages workflows for closed-loop and secure business processes. WebSphere Business Integration also provides back-end and legacy integration for true business process management without having to write changes to existing data and application environments.