SYS-CON

DEPLOYMENT • RELIABILITY • SECURITY• VALIDATION

WebSphere® JOURNAL

www.SYS-CON.com/WebSphere

The World's Leading Independent WebSphere Resource

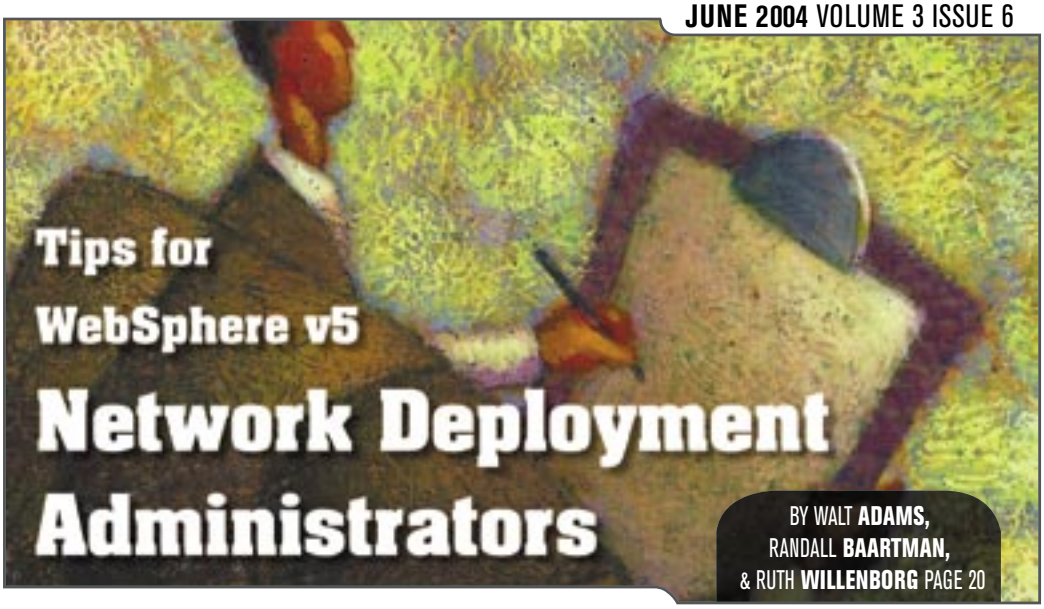
JUNE 2004 VOLUME 3 ISSUE 6

DISPLAY UNTIL AUGUST 31, 2004

$9.99US $9.99CAN

06

0 74470 03422 3



SOAs Help Businesses Deal with Change

BY JACK MARTIN

IBM announced some new software and services to make it easier for companies to adapt their information-technology infrastructures to changing business conditions. The latest IBM offerings will help companies build service-oriented architectures (SOAs), which are collections of business processes that use reusable standard interfaces to integrate applications within a company as well as with customers and suppliers.

IBM's focus on SOAs is an extension of its existing strategy to help customers take advantage of standards-based technologies to drive down costs, create new opportunities for growth, and transform to on-demand business. IBM has been laying the groundwork for SOAs through WebSphere Application Server and integration software, Tivoli's infrastructure management and security software, and supporting the developing industry standards for Web services, combined with their enormous footprint knowledge and experience of solving problems for businesses.

Today, lots of companies use hard-coded connections to communicate with applications both inside and outside of their offices. These connections can make it difficult for companies to respond to change. Because SOAs are collections of standards-based software components, such as Web services, they enable companies to create flexible, reusable connections to address specific business issues such as a manufacturer simplifying its supply chain or a bank processing a mortgage application.

By breaking down business components into basic building blocks, just as the standardized software interfaces allow the IT infrastructure to be componentized, flexible business processes can then be matched with flexible IT processes. The time it takes to process data by linking the business and IT processes in an SOA has the potential to be reduced to days.

Companies can start using SOAs to solve a small number of business problems to get their feet wet and roll out the SOA implementation at their own comfort level and pace.

You can use SOAs to turn existing functionality into a single, reusable component by creating an environment where multiple groups can access the same low-cost infrastructure and save each group or company money.

Early adopters who have implemented SOAs say that they have been able to see dramatic cost savings through reductions in development and operating costs.

The standardization of the architecture, has enabled consumers to consolidate resources using WebSphere Business Integration Server. It allows them to build and integrate applications within service-oriented architectures and it offers native support for the Business Process Execution Language, which is an industry-standard specification executing business logic. WebSphere Business Integration Server also enables companies to create reusable services out of their existing Web services as well as to package applications and combine services to link business processes with software applications.

Planning before embarking on an SOA implementation is key, as you must assess both functional and technical aspects of the planned SOA implementation to meet desired quality of service. Planning for SOAs that follow a path beginning and ending with specific objectives while identifying potential SOA business service candidates is key to a viable SOA vision.

Using the SOA functional building blocks,

–continued on page 30

Jack Martin, editor-in-chief of ***WebSphere 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. Jack is coauthor of ***Understanding WebSphere***, from Prentice Hall. jack@sys-con.com

WebSphere® 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
Executive Editor	GAIL SCHULTZ
Contributing Editor	PATTI MARTIN
Product Review Editor	JAY JOHNSON
Special Projects Editor	RICHARD ARONE
Editor	NANCY VALENTINE
Associate Editors	TORREY GAVER
	JAMIE MATUSOW
	JENNIFER VAN WINCKEL

Writers in This Issue
Walt Adams, Randall Baartman, Kulvir Bhogal, Caroline Buck, Paul Della Maggiora, Jim Doherty, Jim Liddle, Angus McIntyre, Thina Natarajan, Dwight Peltzer, Mike Sick, Ruth Willenborg

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

PRESIDENT AND CEO	FUAT KIRCAALI
PRESIDENT, SYS-CON EVENTS	GRISHA DAVIDA
GROUP PUBLISHER	JEREMY GEELAN
SENIOR VP, SALES & MARKETING	CARMEN GONZALEZ
VP, INFORMATION SYSTEMS	ROBERT DIAMOND
VP, SALES & MARKETING	MILES SILVERMAN
PRODUCTION CONSULTANT	JIM MORGAN
FINANCIAL ANALYST	JOAN LAROSE
ACCOUNTS PAYABLE	BETTY WHITE
ADVERTISING DIRECTOR	ROBYN FORMA
ADVERTISING MANAGER	MEGAN MUSSA
ASSOCIATE SALES MANAGERS	KRISTIN KUHNLE
	BETH JONES
	DOROTHY GIL
CONFERENCE MANAGER	LIN GOETZ
LEAD DESIGNER	LOUIS F. CUFFARI
ART DIRECTOR	ALEX BOTERO
ASSOCIATE ART DIRECTORS	RICHARD SILVERBERG
	TAMI BEATTY
WEB DESIGNERS	STEPHEN KILMURRAY
	MATTHEW POLLOTTA
CONTENT EDITOR	LIN GOETZ
CIRCULATION SERVICE COORDINATORS	SHELIA DICKERSON
	EDNA EARLE RUSSELL
	LINDA LIPTON

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® Journal (ISSN# 1535-6914) is published monthly (12 times a year).
Postmaster send address changes to:
WebSphere Journal, SYS-CON Publications, Inc.
135 Chestnut Ridge Road, Montvale, NJ 07645

WORLDWIDE NEWSSTAND DISTRIBUTION
CURTIS CIRCULATION COMPANY, NEW MILFORD, NJ

FOR LIST RENTAL INFORMATION:
KEVIN COLLOPY: 845 731-2684, KEVIN.COLLOPY@EDITHROMAN.COM
FRANK CIPOLLA: 845 731-3832, FRANK.CIPOLLA@EPOSTDIRECT.COM

Two years without a vacation.
The application's up. It's down.
It's up. It's down.

I'm to blame. Steve's to blame.
Someone's always to blame.

Not any more.

Get Wily.™

Enterprise Java
Application Management
1 888 GET WILY
www.wilytech.com

wily technology

Expanding Web services and creating J2EE Enterprise applications, Part 2

WebSphere's Studio Application Developer Architecture Explained

BY DWIGHT PELTZER

Dwight Peltzer is a professor of computer science, developer, lecturer, and author on Microsoft and J2EE software solutions. Presently, he is a faculty member at the School of Engineering and Computer Science at the C.W. Post campus of Long Island University. He teaches seminars on C++, Visual Basic, .NET, C#, SQL, ASP.NET, and J2EE-related technologies such as JAXP, JavaServer Pages, and XML. dpeltzer@optonline.net

In part 1 of this article, we discussed Microsoft's .NET Framework. This month, we examine WebSphere's multi-product architecture with focus on the WebSphere Studio Application Developer (WSAD) for Windows, the Integrated Development Environment (IDE) that replaces IBM's Visual Age for Java. WSAD, a collection of development tools based on the Eclipse technology, provides an environment designed to enhance Web services development and build J2EE Enterprise applications. The overall WebSphere Application Server (WAS/WSAD) architecture is component-based and layered in multi-tiers. The Eclipse technology, a plug-in based approach, is integrated with WAS to allow users and third parties to extend the base IDE with their own plug-in technologies.

When you start WSAD, the opening view consists of several panes that represent the various components contained within your workspace, i.e., editors, properties, and perspectives. Editors facilitate the creation and modification of resources. Views present a way to navigate through resources. For example, you can rename files or folders from the navigator view. Modifications made in views are reflected immediately, whereas changes to your work do not show up in the navigator view. The properties view displays file properties such as the date the file was last accessed or modified. Perspective views present a series of editors and views organized for executing a specific task.

Each perspective contains its own unique view, displaying tools and editors appropriate for a particular view. For example, if you select the Web perspective, it provides quick access to the Web deployment descriptor and displays the specified hierarchy of folders that have been conformed to J2EE, such as the resources folder, the Web content folder, the images folder, the META-INF and WEB-INF folders, etc. To be more specific, the Java browsing perspective displays projects, packages, types, and members. It goes without saying that the debug perspective provides a server's view and displays its status, variables, breakpoints, expressions, and registers. It also provides a console view with a tasks tab. It is worth taking the time to explore the perspectives and options available to developers. The IDE is a full-service, one-stop environment for carrying out any task necessary to create an enterprise application. In many ways, it is similar to Microsoft's Visual Studio.NET, where a developer is presented with multiple choices for developing Web services or enterprise applications.

Before continuing the discussion on WSAD's infrastructure, let's look back to the mid 1990s and examine the early client/server paradigm. Originally, it was intended to increase productivity. Unfortunately, it failed primarily because business processes were designed to solve technological shortcomings rather than update a system to keep up with technological advances. Unknowingly, developers addressed the wrong issues. Mixing business logic and its implementation with presentation technology was the primary source of the problem.

The decision as to where to place data functionality and transactional processing was a major issue. Most efforts were focused on positioning business solutions on the back end with the user. Developing "thick client" applications was both costly and difficult to scale and distribute. Fortunately, today's Web-based technologies have resolved many previous issues.

The Web-Based Client/Server Design Paradigm

The present Web-based client/server architecture is component-based with large collections of abstract interfaces that promote

component reuse. They represent several vertical slices of the actual business process comprising the following items.

- Remote object presentation uses either server-side scripting or Java servlets. The client makes requests via a Web browser and HTTP. The Java servlet is called to sit in memory and interpret incoming client messages. The servlet then dispatches the request to the appropriate JavaBean or Enterprise JavaBean (EJB) for processing. Finally, the servlet delegates a JavaServer Page (JSP) to return the result to the client's browser in HTML format. This represents a "thin client" solution.
- Remote object access and data management utilizes RMI-IIOP, JNDI, and/or RPC server-side calls, whereas the client uses either a Java applet or a Web application to achieve its task.
- Distributed data management places business logic/data on both the client and on the server.

By viewing these components from both a server and client perspective, we can understand how the components play an integral role in forming a multi-tiered architecture. Clearly, the business process has shifted from client-side processing to the server side. Only presentation functionality remains with the client.

WSAD conforms to the J2EE specifications for constructing a system architecture. The specifications define interfaces, their respective life cycles, and interaction with the various components' functionalities. Most importantly, the specifications introduce the concept of containers. The containers are designed to host J2EE components within a layered boundary. The containers serve as component managers by defining component relationships and their interdependencies within a specific tier. The following example demonstrates how the application containers can be partitioned (see Figure 1).

The individual components residing within each container interact with other container components based on business model requirements. This graphic introduces the Model/View/Controller (MVC) design pattern we shall examine shortly.

WSAD provides a built-in WebSphere Test Environment (WTE), which is a servlet and test development environment built into WAS. Three types of servlet engines exist.

- ***Standalone servlet engines:*** These contain the usual HTTP server functionality and provide support for servlets.
- ***Add-on servlet engines:*** These types of engines function as plug-ins to an existing Web server. WSAD provides a plug-in that installs into an HTTP server. By doing so, the HTTP server will determine which URLs refer to applications deployed in WebSphere and forward the servlet requests to a standalone application server (WAS) via HTTP.
- ***Embeddable servlet engines:*** These engines are servlet deployment platforms that can be embedded in other applications. Subsequently, the applications become the servers. IBM's WebSphere Test Environment is such a servlet engine.

Explaining the MVC Design Paradigm

WebSphere has adopted the MVC design pattern created by Xerox PARC for Smalltalk-80 in the 1980s. This model is becoming increasingly popular with the programming community. Significantly, MVC was selected by Sun Microsystems as the recommended model for its J2EE specifications. MVC can be used by any programming language or platform and enforces the separation between (1) the view, (2) the model, and (3) the controller. It separates data input and processing from data output.

The view interface presents users with a wide range of options such as making a selection from a series of radio buttons or a drop-down list. For example, he/she can select the type of delivery service, i.e., UPS, FedEx, USPS, or DHL from a list. Traditionally, the view is presented in HTML format. However, an interface may be also presented as XHTML or XML/XSLT templates. Because the view is display-neutral, it can present many diverse views of the same data. A view always reflects any event changes occurring in the "model" layer.

The model contains business objects that interact directly with JavaBeans or EJBs. They implement transactional business rules such as data persistence or retrieval from a relational database.

The controller serves as interpreter by intercepting client requests and calling a servlet's service() function to fulfill the client's request. The controller applies no formatting nor does it output anything. It serves only as a notification service both to the view and model to respond accordingly.

Applying the Mediator Design Pattern

The WebSphere architects have taken the MVC concept one step further by applying the mediator design pattern introduced in *Design Patterns* by Erich Gamma. The mediator/controller pattern places another layer of indirection between the view and the model/domain. A mediator/controller captures and decouples application-specific functionality from presentation technology by implementing behavior that would usually exist as methods in presentation classes. The mediator can be better understood by viewing it as an abstract interface that must be implemented by providing a concrete mediator object. When a client submits a request, the mediator

intercepts the request and delegates the task to a concrete class of objects to fulfill mediator/controller-directed tasks. The mediator is responsible for controlling and coordinating the interactions of groups of objects. As Gamma explains, "the mediator serves as an intermediary that keeps objects in the group from referring to each other explicitly. The objects only know the mediator, thereby reducing the number of interconnections." Doing so places behavior in one class, thus allowing the behavior to be modified or replaced locally. Abstract interface substitutability is the key to object reuse.

There is more than one approach to designing an MVC-based enterprise application. Let's assume that we have an application designed especially for registering and managing new students in an academic environment. The first step is identifying the model classes; in our case we have only one model class, namely, student. The attributes for the student are his/her ID, name, age, address, and academic class. The next step is determining the views in our application. We assume they are JSP pages. Any dynamic changes in a page will be presented in a JSP page. We now have three JSP pages:

- PeruseStudents
- NewStudents
- StudentDetails

The next step is looking for transitions between pages. They occur as a result of selecting a button to make the transition. The button selected is either an HTML button that submits a form or executes a GET or URL link between the JSP pages. In a normal context, each button has its own controller. If the content is dynamic, the controller must implement the transition to a specific page. In other words, the controller must act as a filter for each individual set of data. For example, examining the logic for page transitions suggests the link from PeruseStudents to NewStudents does not alter the way the NewStudents page is presented. Therefore, we can identify the controllers for each unique transition:

- PeruseStudentList
- DisplayDetails
- CreateNewStudent
- UpdateStudent
- DeleteStudent

Now we have two approaches: (1) use a controller to implement each servlet, or (2) employ a single controller to handle all servlet requests and pass the processing to an ancillary class. This solution is called a gateway servlet. With this method, a unique controller is identified for each individual request. This is accomplished by employing a polymorphic controller along with an

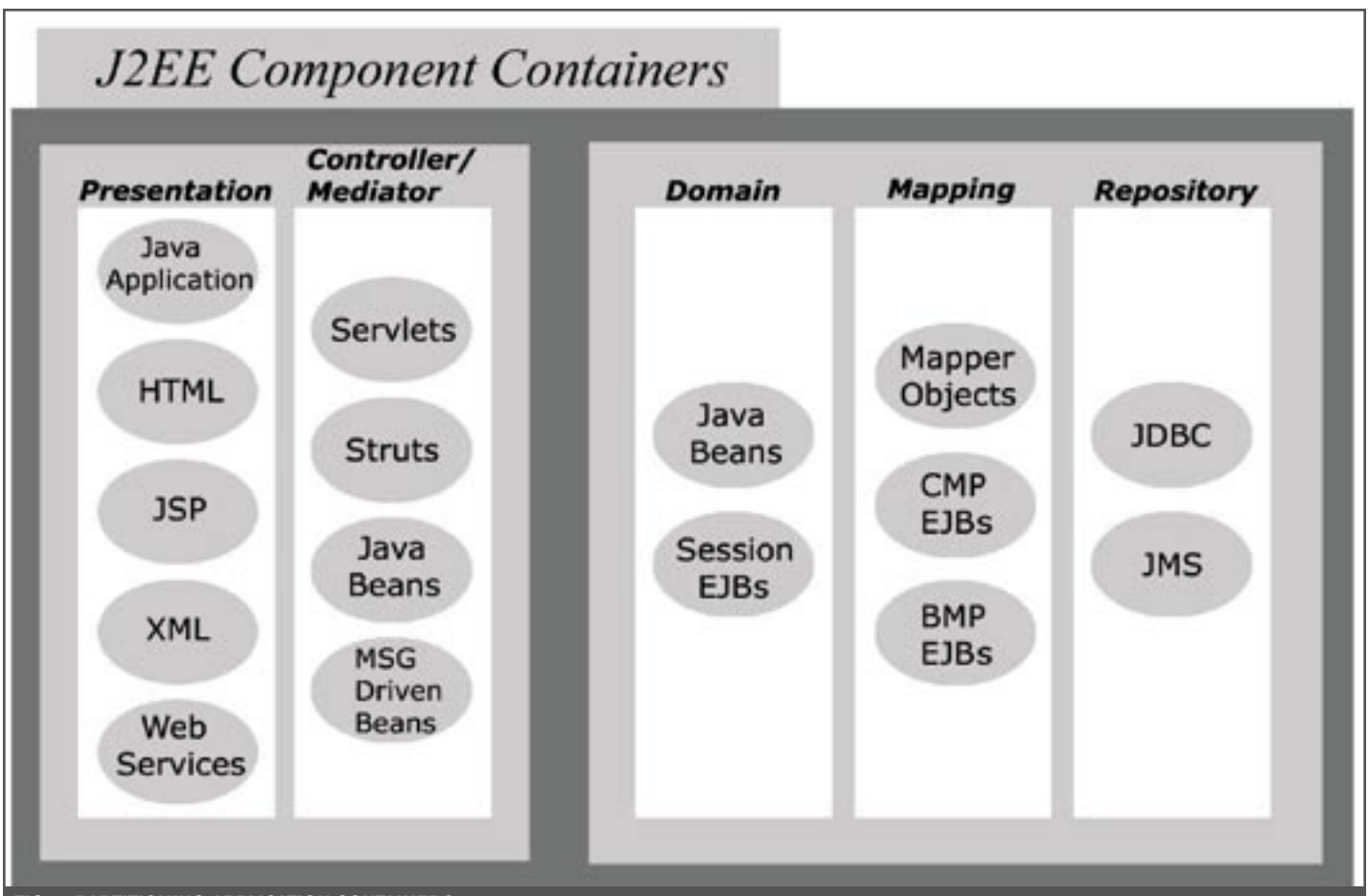


FIG 1: PARTITIONING APPLICATION CONTAINERS

object factory to create new object instances. The benefit obtained from applying this type of approach is obvious – the ability to add new functionality to your application without requiring reconfiguration of the application server. This results in either the implementation of a simple interface or extending an abstract controller class. The Struts Framework we introduce next is built on this model.

WSAD Provides Support for the Struts Framework

Jakarta's Struts Framework was created by Craig McClanahan and donated to the Apache Software Foundation in 2000. The framework is made up of approximately 300 classes divided into core packages. The top level packages consist of the following components.

- ***Action:*** Contains controller classes ActionForm, ActionMessages, and several other framework components
- ***Actions:*** Contains the action classes such as DispatchAction, which can be used to extend your application
- ***Config:*** Includes the configuration classes that are in-memory representations of the struts configuration file
- ***Taglib:*** Contains the tag handler classes for the struts tag libraries
- ***Tiles:*** Contains classes used by the tiles framework
- ***Upload:*** Includes classes used for uploading and downloading files from the file system using a browser
- ***Util:*** Contains general-purpose utility classes employed by the framework
- ***Validator:*** Contains the struts-specific extension classes used by struts when deploying the validator

In short, the Struts Framework is based on the Java servlet technology. Additionally, the framework utilizes a servlet to process incoming requests. However, it relies on other components that are part of the controller domain. The Struts Framework is considered a Model II MVC architectural model, meaning it uses servlets as the mediator/controller layer, and delegates JSP pages to function as the presentation layer. The servlet is the recipient of HTTP post requests and passes the posted data to the model domain for processing. Furthermore, it selects a specified JSP to return and display the results. Struts definitely makes it easier for developers to build Web applications. And this explains precisely why WSAD supports this framework.

The Struts Framework solves problems developers usually face when using servlets as the mediator. Passing HTTP parameters to JavaBeans is cumbersome because this process requires invoking a JSP to manage the HTTP request. This violates the Model II MVC architecture. Listed below are some of the challenges developers encounter.

- ***Servlets/JSPs:*** These provide no mechanism for page or form validation. The developer must implement his/her own validation procedures manually.
- ***Hard-Coded JSP URIs:*** Modification or re-organization of JSPs in a Web site is difficult without updating Java code in the servlets.

The Struts Framework resolves these issues. Unfortunately, we cannot examine the entire framework here. Therefore, we recommend that you locate the source code by entering the following URL in your browser: http://jakarta.apache.org/struts.

WSAD includes a test environment for testing servlets and JSPs. WSAD also provides Web interfaces using XML/XSL technology.

We have not discussed WSAD's support for building enterprise applications with EJBs. This topic will be reserved for a future article. However, we have discussed the presentation and model layers in considerable detail. Additionally, we have considered the MVC design paradigm, its impact on program design, and how it is implemented in WSAD.

The next article in this series will contain a direct comparison of .NET's ASP.NET technology with Java servlets and JavaServer Pages. Several coded examples will provide a comparison of how both .NET and WSAD implement the MVC architecture. Until then, happy computing.

Resources

- Kyle Brown et al. (2003). *Enterprise Java Programming with IBM WebSphere*. Addison-Wesley.
- Nillson, Dale R. and Mauget, Louis E. (2003). *Building J2EE Applications with IBM WebSphere*. John Wiley & Sons, Inc.
- Cavaness, Chuck. (2002). *Programming Jakarta Struts*. O'Reilly Media, Inc.
- Peltzer, Dwight. (2004). *Interoperability Between .NET & J2EE*. McGraw-Hill Osborne Media.
- Hunter, Jason and Crawford, William. (1998). *Java Servlet Programming*. O'Reilly Media, Inc.

"The IDE is a full-service, one-stop environment for carrying out any task necessary to create an enterprise application"

Say That Again... (In a Different Language)

An introduction to programmatic interaction with WebSphere Translation Server

BY KULVIR SINGH BHOGAL

Kulvir Singh Bhogal works as an IBM Software Services for WebSphere consultant, devising and implementing WebSphere-centric solutions across the nation. He has over 60 patents pending in a myriad of technology areas. kbhogal@us.ibm.com

& BRAD BOULDIN

Brad Bouldin works in the Portal Practice in IBM Software Services for Websphere. bbouldin@us.ibm.com

The phenomenon of globalization has unearthed a major question. With a huge market of non-English speaking customers, how can we facilitate communication to such a large subset of individuals without paying an arm and a leg for translation? In this article, we'll introduce you to IBM WebSphere Translation Server (WTS) for Multiplatforms v5.0. As you will see, WTS can address the translation needs for an enterprise organization. This product can deliver content to end users in their native languages. The delivery is performed at a fraction of the cost of professional human translation and done on the fly.

We will show you how to install WTS and programmatically interact with it using Java. For our studies, we used the Windows environment. However, WTS is also available for Sun Solaris, AIX, and Linux.

Some Background on WTS

WTS supports bidirectional translation of content for English to/from French, Italian, German, Spanish, Japanese, Chinese (simplified), and Chinese (traditional). For the languages of Korean and Brazilian Portuguese, only unidirectional translation (from English) is supported. WTS is a machine translation (MT) offering. Machine translation refers to the science of automatic translation of human language by computers. WTS consists of MT engines that translate from one language to another. You can also add to the vocabulary of a domain using the User Dictionary Manager tool and change how words or phrases are translated. The User Dictionary Manager is a client-side Java application that allows you to add to the core lexicons of WTS. For example, this extensibility feature might be used to add slang, industry-specific terms, jargon, or trademarks.

For the translation of Web content, WTS has built-in support for WebSphere Application Server (WAS), IBM HTTP Server, as well as HTTP servers from Netscape and Apache. WTS offers on-the-fly and on-demand translation. On-the-fly translation takes place automatically for the end user of a Web page if the user has set the language preference in the Web browser.

On-demand translation prompts the end user to translate from within a Web page. This translation approach can be implemented on an HTTP server as well as WebSphere Application Server (WAS). In the case of an HTTP server, Web server plug-in modules are used to facilitate translation. On the other hand, WAS uses a translation servlet. A sample servlet is provided with the product, so modifying or integrating with your own Web applications is straightforward. Web server/WAS interaction is beyond the scope of this article. However, the Information Center provides a good breakdown of how to couple the services of WTS.

Installing WTS

The installation of WTS is pretty straightforward. When you get to the screen that asks you to select the products you would like to install, choose the options for Gateway and InfoCenter. Also, pick the languages for which you want to have support. For our demonstration purposes, we checked the Spanish option (see Figure 1).

In the connection settings page, enter the hostname or IP address of the machine running WTS.

In the screen asking for the type of trace messages you would like to be displayed, this depends on what messages you want showing up in your trace files. We chose

"Add Warning Messages", which will also display Error and Logging messages.

When asked to provide the path to your default configuration file, enter the correct value corresponding to your setup.

Clients accessing via Java require Java 1.3 to interact and operate with WTS's components.

Verifying Installation of WTS

To start the WTS Gateway, simply launch the application from the Start menu and click the start icon. After doing so, you should see an initialization screen (see Figure 2).

You can edit the properties of the translation server using the Properties panel (see Figure 3).

By default, WTS listens on port 1099. WTS can also be started directly from the command line (either utilizing a GUI or not):

1. ***Create the server with the WTS GUI:*** java -classpath wts.jar -ss1M com.ibm.lt.LTserverPanel start
2. ***Create the server with no GUI:*** java -classpath wts.jar -ss1M com.ibm.lt.LTserver wts.properties

For the server with no GUI, the startup properties for the translation server are obtained from the wts.properties file (located in the C:\Program Files\IBMWTS directory under default installation).

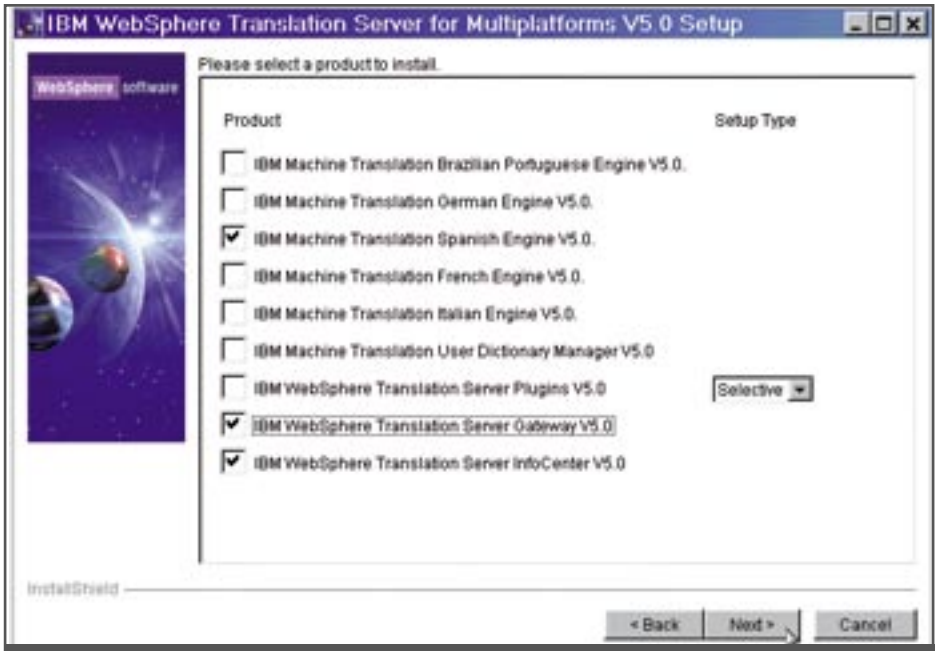


FIG 1: SELECTING THE PRODUCTS TO INSTALL

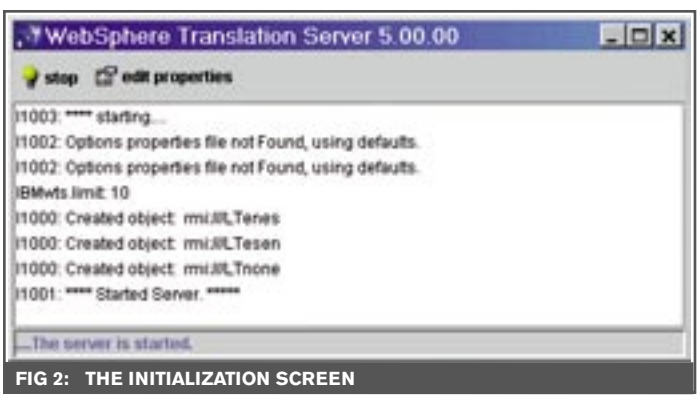


FIG 2: THE INITIALIZATION SCREEN

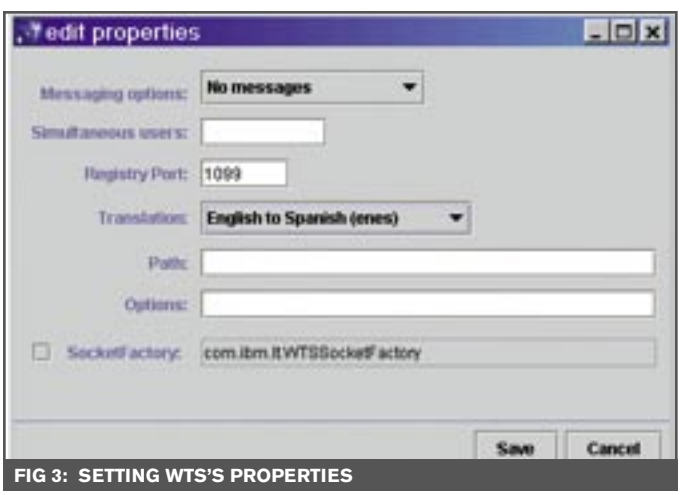


FIG 3: SETTING WTS'S PROPERTIES

Programmatic Interaction with WTS

WTS has exposed APIs in both the Java and C programming languages. We will focus our study on the Java API. WTS is implemented in Java and requires a JVM for its habitation. Interaction using C is facilitated by linking with the supplied WTS library and header file. To start programming a Java client application, you need to have the wts.jar file in your classpath. By default, this file is installed in the C:\Program Files\IBMWTS directory. As shown in Figure 4, clients interact with WTS through the RMI protocol.

The client is provided with a Java API that abstracts the user from the intricacies of the underlying complexity of interaction with WTS.

The code in Listing 1 shows a simple program that takes a string as a program argument. As you can see, the code is rather straightforward (see Listing 1).

To use the program, you need to provide the string to be translated as a command line argument. We used the string "My cat likes milk." Make sure your string is enclosed in double quotes (see Figure 5).

In the code, we use the GetService method (providing arguments stating our WTS hostname and our translation method [Spanish to English, signified by enes – refer to the Information Center for a complete list of language abbreviations]). From the resulting LTinterface object, we can call the jltTranslate method, providing our string to be translated as one of our arguments. The result of this method call is our translated string (see Figure 6).

If you are curious, go ahead and provide the result you got, "A mi gato le gusta la leche," as an argument to the program. Also change the TRANSLATION_METHOD variable to a value of "esen", stating that you want to translate from Spanish to English. The result? You should see a translation that reads "My cat likes the milk." If you recall, our original string was "My cat likes milk." Not bad at all!

There are other utility methods provided by the Java APIs that you may find helpful. For example, you can use the jltFile() method to convert a text file in one language, and output the resulting translation to a new file:

```
service.jltFile(handle, "c:\\in.txt", "c:\\out.
txt");
```

The jltFile method can also convert HTML files, intelligently skipping HTML tags:

```
service.jltFile(handle, "c:\\in.html", "c:\\out.
html");
```

This provides a convenient way to translate multiple files into many other languages. Using this capability, you could convert a set of marketing reports, help documentation, or a set of Web pages or text files to nine other languages quickly and easily.

Guidelines for Writing Content

WTS does its best work when writers of the source material follow some basic guidelines. First, writers should be notified that their text will be translated into a different language, so it is best to pick words and phrases just as if they were speaking to someone who is not fluent in their native language. Using shorter sentences

"WTS does its best work when writers of the source material follow some basic guidelines"

and generally avoiding idioms and slang expressions will make translation easier (whether translation is performed by machines or by humans). It's important for Web designers to remember that .gif files that contain words cannot be translated by a machine, so it's best to avoid placing text inside graphic files for buttons or icons. If writers keep these rules in mind as they write content, your translation results will improve. For a more extensive list of guidelines, see the document "WTS – Guidelines for Writing Content that Will Be Machine-Translated" mentioned in the Resources section.

Conclusion

WTS offers a set of flexible APIs and features that give developers the freedom to leverage the product in many useful ways. Translation can be used for static content and dynamic Web content, and can even be integrated into real-time chat applications such as Lotus Instant Messaging. IBM's WebSphere Portal product includes built-in support for WTS, and there is no programming necessary to enable the translation of individual portlets, or if you prefer, the entire portal page. Using the techniques presented in this article, you can write your own programs that take advantage of WTS's powerful features. Depending on your translation needs, the cost savings can add up quickly. Whereas human translation typically costs between $0.15 and $0.45 per word, depending on the languages and type of document, WTS deployed in your organization can cost as little as fractions of a penny per word. Other advantages of machine translation are that translation services are always available and can translate up to 500 words per second (a typical human translator may top out at 2,500 words per day). Many organizations decide to reduce costs by using a combination approach, first translating by machine, followed by proofreading by human translators.

The global Internet allows e-business to reach the corners of the earth. The problem is, not every corner of the earth understands English. Not catering to such a large clientele is the equivalent of shutting the doors for your own e-business. In this article, we showcased WebSphere Translation Server and showed you how the product can be easily installed and configured to provide translation services programmatically. The offerings of WTS are quite extensive. You

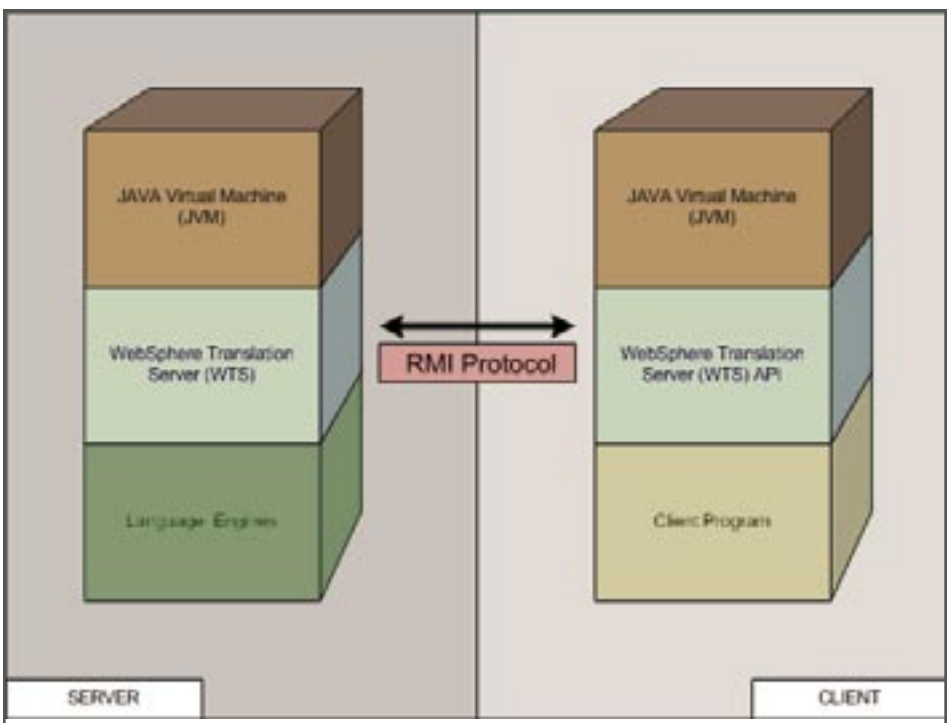


FIG 4: CLIENT INTERACTION WITH WTS

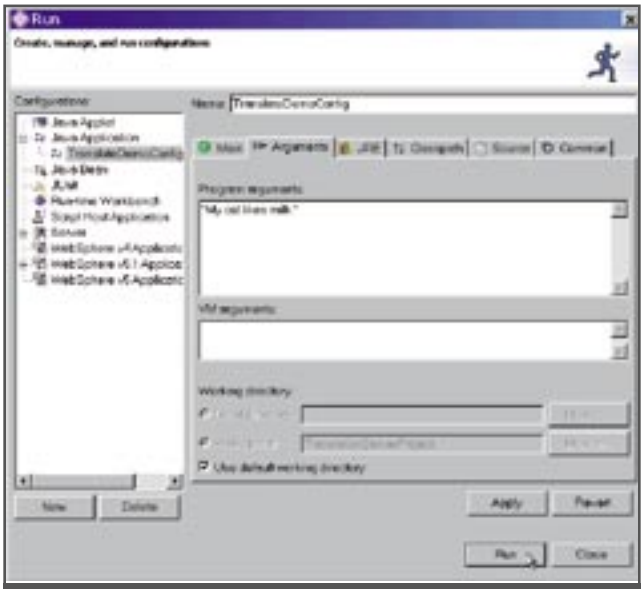


FIG 5: PROVIDING THE SAMPLE PROGRAM WITH RUNTIME ARGUMENTS IN WEBSPHERE STUDIO

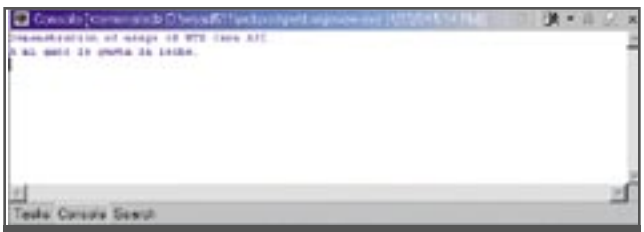


FIG 6: THE ENGLISH STRING TRANSLATED TO SPANISH USING THE SAMPLE APPLICATION

can learn more about the offerings by viewing the product Information Center. The Information Center is also found in the InfoCenter directory of the product's first installation CD. For IBM customers who have WebSphere Portal Server Enable or Extend, they should be aware that WTS comes bundled with the product.

Resources

- IBM WebSphere Translation Server Product Page: www-306.ibm.com/software/pervasive/ws_translation_server/
- WebSphere Translation Server InfoCenter v2.0 (G210-1527-00): www.elink.ibmlink.ibm.com/public/applications/publications/cgibin/pbi.cgi?CTY=US&FNC=SRX&PBL=G210-1527-00
- User Dictionary Manager v2.0 InfoCenter: www-1.ibm.com/support/docview.wss?rs=425&context=SSMQRN&uid=pub185256df80076c61f85256ba300573e46&loc=en_US&cs=utf-8&lang=en+en
- WebSphere Translation Server Support Site: www-306.ibm.com/software/pervasive/ws_translation_server/support/
- WTS - Guidelines for Writing Content that Will Be Machine-Translated: www-1.ibm.com/support/docview.wss?rs=425&context=SSMQRN&uid=swg21115627&loc=en_US&cs=utf-8&lang=en+en
- Globalizing Your e-business http://gwareview.software.ibm.com/software/globalization/index.jsp

LISTING 1:

```
public class Translatedemo {
   // hostname of translation server
   static final String WTS_SERVER = "localhost";
   // provide translation method - esen refers to Spanish to
       English
   static final String TRANSLATION_METHOD =
          "enes";
   public static void main(String args[]) {
          System.out.println(
                    "Demonstration of usage of WTS Java
                       API");
          com.ibm.lt.LTinterface service = null;
          Object handle = null;
          String translation;
          String whatToTranslate = args[0];
          try {
                    // obtain a translation server engine
                    // note the specification of the
                       translation method
                    service =
                              (com.ibm.lt.LTinterface) com
                                        .ibm
                                        .lt
                                        .LTengine
                                        .GetService(
                                                  WTS_SERVER,

TRANSLATION_METHOD);
          } catch (Throwable t) {
                    t.printStackTrace();
                    System.out.println(
                              "No service available");
                    System.exit(0);
          }
          try {
                    // inform handle that we are beginning
                       translation
                    handle =
                              service.jltBeginTranslation(
                                        "*format=text");
                    // API call to phrase translation
                       function
                    translation =
                              service.jltTranslate(
                                        handle,
                                        whatToTranslate);
                    System.out.println(translation);
                    // clean up
                    service.jltEndTranslation(handle);
          } catch (Exception e) {
                    e.printStackTrace();
                    System.exit(0);
          }
          System.exit(0);
   }
}
```

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

diagnosys

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

Sign up today at:
www.hwcs.com/wsj10.asp

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

You need DiagnoSys.

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

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

H&W
www.hwcs.com | 1.800.338.6692

Bridging the Enterprise Computing Gap For 25 Years

Choose your tools and processes wisely

Developing Reliable Code

BY ANGUS MCINTYRE

Angus McIntyre is the director of business development for embedded technology at IBM and has spent his entire career on application development tools (OTI, IBM Canada Lab, VisualAge for Java, VisualAge Micro Edition).

As convenient and connected pervasive devices become more exciting and imaginative, the applications that run them become more challenging and complex. The need to balance function with the constant pressure of time-to-profit are the motivating forces behind the development team, making the selection of tools and processes critical.

The developers who remain chained to traditional command-line interfaces can easily get lost in the details. To keep pace with the rapid changes in the marketplace, the familiar cycle of development functions (create, edit, build, integrate, deploy, run, debug, and release) must occur more quickly. Simply throwing more people at the problem only makes the process slow and complex.

Using an Integrated Development Environment (IDE), together with strong version control tools and proven object-oriented techniques is fundamental for the quality of output from your team. Choose an IDE tuned for the development of device applications that presents the development team with a simple, logical interface to the functions, and provides them the much-needed version control elements.

The process of development follows several discrete steps. In each phase, quality tools, effective team collaboration facilities, and sound development practices can accelerate iterative development. Work to resolve issues early to avoid high costs and delays later in deployment.

Design for Reuse

Isolation of the data that an application uses from the business logic and the user interface is vital for code reuse and quality (Model/View/Controller design pattern). Write once, run anywhere (WORA) rings hollow throughout the J2ME community, as each device has the features and functions (Navigation, GUI, buttons) that differentiate it from other devices (the wonders of a free enterprise system – choice).

Designing the code elements (GUI, business logic, data, class libraries) up front that make up an application will increase the reuse of code from project to project, and potentially from device to device. Reuse of objects from project to project across an organization is essential to reducing duplicate effort and maintaining multiple code bases. Good reference materials on this include Michael Juntao Yuan's book, *Enterprise J2ME: Developing Mobile Applications* and Martin Fowler's work on refactoring (www.refactoring.com).

Develop for Multiple Targets

Several base class libraries of various sizes, designed specifically for the embedded market, offer choices when dealing with resource-constrained devices. They all provide the basic classes needed to run Java applications. Starting small and moving to larger profiles and configurations as necessary keeps the footprint of the target device down.

Ensure that the profiles and configurations shipped with the tools and the target runtime environment have been tested and certified to meet the specifications as defined by the Java Community Process and that they carry the Java-powered logo. In small devices, this would be Constrained Limited Device Configuration (CLDC) and Mobile Information Device Profile (MIDP). In larger devices, it would be Connected Device Configuration (CDC), Foundation Profile, and Personal Profile. These offer the assurance of a consistent set of APIs, regardless of how they are implemented by the JVM vendor. It does not guarantee the quality of the target environment or the portability of the application code.

For applications in which the device must connect and transact over the air (OTA) with enterprise systems, the use of open standards (JDBC, JMS, RMI, UDDI, SOAP, XML, etc.) and middleware to connect devices to data, transactions, and applications can facilitate the connection to existing sources of content

and information. Also needed in the development stage is the ability to maintain a single version control database that coordinates the code to be deployed to the device with the gateway or server. Once your end-to-end solution is delivered, each end knows how to talk to the other.

Deploy, then Debug

Much of the code can be tested right on the development workstation and portions can be run before the program is complete. Serious application development can begin before the actual target device is available in limited supply, or even while it is still being refined.

Having an emulator for the target device that integrates with the development environment is key to building applications for PDAs and cellular handsets. The Universal Emulator Interface, as well as the eclipse.org extensible architecture, have been used by PalmOne and Nokia to integrate emulators and simulators into IDEs. These provide developers with the tools needed to create, test, and make the application ready for OTA certification testing by the content aggregator, carrier, or service provider.

Using an emulator or a simulator can quickly get the development team over the hurdle of ensuring that the system works inside either one, but is no indication that the code will work as designed and written on the target device.

For deep embedded projects (or even a Smartphone that allows a socket connection), development tools that support remote OTA debugging as the application runs live on the target are essential to avoiding errors in production – before millions of devices are shipped. But errors happen. Software provisioning and maintenance architectures, such as OSGi (on larger devices – CDC), can enable bundles of software to be installed, maintained, and removed from a device. Systems on which bundles of software can be pushed have quite the advantage.

Analyze

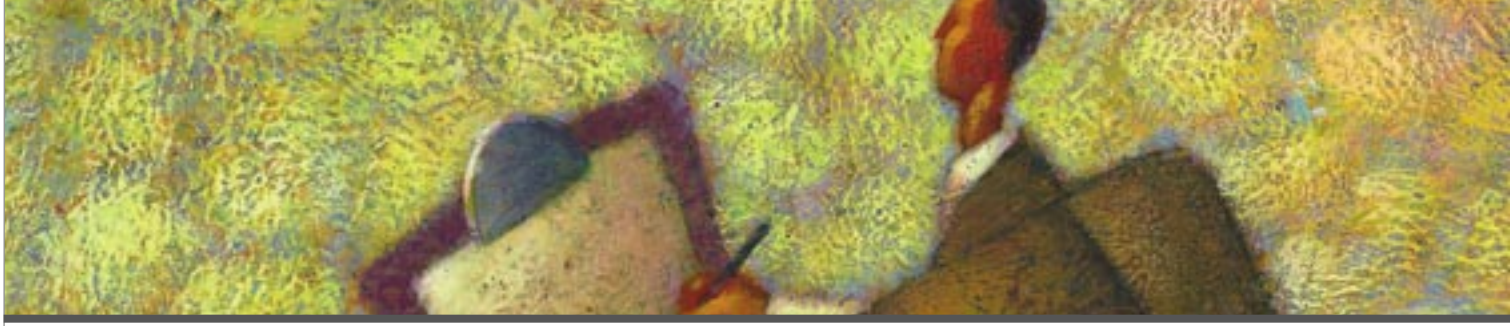
Analyzing applications can show interesting details about application execution and memory usage, giving you critical information about the performance bottlenecks in your application. It helps you achieve optimum balance between speed and memory constraints. By inspecting code and profiling, you become aware of code that needs to be optimized. Profiling can also point out the "big eight" problems of application performance that can avoided up front by the use of good coding practices:

1. Are floats and doubles supported by the device? Doing these calculations in software is time consuming.
2. Are floats and strings overused? Loading blocks of information into the system may be more efficient.
3. Ensure that hash tables with linked lists are not sparsely populated, or seek out alternative data structures that conserve valuable memory.
4. Pick synchronization points carefully. An application waiting on a race condition can cause aggravating wait states for end users or, even worse, hold up critical information from being acted upon (e.g., voice navigation commands back to a driver).
5. Static code blocks can cause initialization times to increase dramatically.
6. Be careful not to mismatch the size of strings passed to the device.
7. Validate at every stage of the process – rather than on entry/exit from a subsystem.
8. Refactor class libraries for reuse.

Conclusion

There are some excellent examples of wireless Java applications that have been deployed into production today. AirToolz has a wireless field-force automation system for the construction industry. Banksys has deployed WebSphere Micro Environment into their CZAM and SMASH wireless payment terminals. Construction workers and wireless banking payment systems are customer segments that count on reliable code. The secret for developers is to select the right tools and to look for patterns of integration in the industry. These patterns can be the samples and examples, and are documented to the point that you can use them for your own purposes. Use code that works, and modify it until it works for you.

"The secret for developers is to select the right tools and to look for patterns of integration in the industry"



Tips for WebSphere v5 Network Deployment Administrators

Expanding the capabilities of WAS ND

BY WALT ADAMS

Walt Adams is a senior software engineer at IBM. He has 20 years experience in software development and is currently a member of the WebSphere Application Server Performance team. He is currently working on WebSphere scalability and can be reached at hwadams@us.ibm.com.

RANDALL BAARTMAN

Randall Baartman is an advisory software engineer at IBM. He has 19 years of experience in software development and is currently a member of the WebSphere Development Organization, specializing in WebSphere performance and scalability. You can e-mail Randy at baartman@us.ibm.com.

The WebSphere Application Server Network Deployment version 5.0 (WAS ND) provides an infrastructure for you to centrally administer multiple WAS servers, resources, and other elements of your topology. Your managed topology can include support for clustered servers with workload management and failover. WAS ND's support for centrally administering topologies provides significant benefits for both large-scale and small-scale topologies.

This article provides tips to help you better exploit the administrative capabilities of WAS ND. We discuss options for deciding on the scope of your topology, and advice on administering the topology. If you are planning or administering a topology based on WAS ND (or WAS Enterprise), these tips, which are based on lab experiences with large ND deployments, should be helpful.

Tips for Planning Cells

Applications and the resources they require are the key elements in any topology. WAS ND provides flexibility for spreading your applications across cells, nodes, and application servers. There is no magic formula that tells you exactly how to build your cell topology, but your solution needs to combine performance and scalability with your site requirements and administrative processes.

WAS ND allows for many nodes, with multiple application servers on each node and multiple applications in each server. In fact, the WAS ND architecture allows as many as you want. Lab experiments have achieved hundreds of nodes and server elements without experiencing any "hard" limits (see Figure 1).

While large configurations are impressive, they may not be entirely practical for you. As you add elements to your topology, it's logical that you'll experience effects of a larger topology. When you plan your topology, consider the practical limits. In terms of the number of nodes and servers, practical limits include complexities in administering a large topology, as well as physical constraints such as memory and network bandwidth available.

Let's take a look at a few tips for you to consider when you plan your topology.

Place the Deployment Manager on a Dedicated System

The deployment manager for your cell can be co-located on a system with a node agent and application servers. But, sharing a system can introduce contention for system resources and interfere with administration performance. Plus, remember that your deployment manager is your central point of administration for your entire cell, and anything that compromises your deployment manager process also compromises your ability to manage and monitor your cell. Therefore, consider placing the deployment manager on a dedicated system.

Put One or a Small Handful of Applications in Each Application Server

While it is entirely possible to put many, many applications in each server, the more typical and preferred deployment is a single application, or perhaps a small set of closely related applications. Keeping applications separated is best from a standpoint of security, isolation, manageability, and so on.

Provide for Fail-Over

Though we won't go into all of the various fail-over and high availability alternatives and options for your site, you'll typically want at least two nodes (at a minimum), to provide for fail-over. Using WAS ND cluster capabilities, each application is set up and managed as a cluster across the nodes.

Meet Each Application's Performance and Scalability Requirements

While many application performance and scalability requirements will be satisfied by just two application servers across two nodes, others have additional needs.

In terms of your performance needs, make sure you consider them in case of maintenance or failures. For example, if one server in a two-node cluster is brought down, your site is immediately at 50% capacity. If two nodes are required to meet performance goals, a three-node cluster is the appropriate number to maintain ongoing capacity.

Next, consider how many servers you need to handle your user load. If two servers are insufficient, the easiest scalability solution is to add additional nodes, taking care to scale any shared resources such as databases and directory servers.

Maximize Each Server's Hardware Utilization

Once you have determined how many nodes you need for each application, and therefore, how many application servers you'll need, you can turn your focus to maximizing the hardware resources at your disposal. The first step is to utilize the processing capacity of your servers. Under expected load conditions, check the processor utilization. If the processor is already heavily utilized, do not add more application servers. Adding more servers to an already utilized machine does not improve performance. Typically, if you have a small amount of hardware (2-4 CPUs), running multiple application servers is not necessary for hardware utilization.

However, if you have many applications, but these applications aren't heavily used, you may want to put multiple application servers on a single node to extend the use of the hardware. When doing this, pay special attention to the physical memory discussion that follows.

Keep Application Servers in Physical Memory

Determining the number of application servers per node requires an understanding of the available memory. Never create more application servers than the available system memory can support. This condition leads to paging that adversely affects performance across the system. The default WAS Java Virtual Machine (JVM) maximum heap size is 256 MB. However, when an application server starts up, it does not use its maximum 256 MB heap size. Thus, application servers exceeding available memory may not be apparent at startup time but can result in excessive paging as the servers increase their heap utilization under load.

For planning purposes, recognize that the heap size grows. As the heap size grows, the Java process for the server uses more system memory. The total memory used is equal to the current heap size plus the Java interpreter. The size of the Java interpreter depends on the heap size. For example, a 256MB heap typically has approximately 64 MB additional from the interpreter, whereas a 128MB heap will have approximately 42 MB. Therefore, an application server with a maximum heap size of 256MB can grow to approximately 320MB of physical memory. A 128MB node agent will grow to approximately 170MB.

To plan the maximum number of application servers for a single node, include the memory requirements for the operating system, node agent, deployment manager (if co-resident), and the maximum size of the application servers (including the Java interpreter). The calculations are shown in Table 1. Leaving extra memory is safer than exceeding server memory capacity, so we recommend including a buffer (our calculations used 0.8 as the buffer, but you can increase the value for an even larger margin of safety).

With an available 4GB of physical memory, the calculations show that memory requirements permit up to eight servers (256MB maximum heap) on this system. You should not have more (and as discussed earlier, you may not need to use eight).

Know When to Stop

After you understand what each application requires, how to utilize each server, and your fail-over and maintenance strategy, you still need to determine exactly where to put your applications. If you have a lot of applications, how many is too many to cram into a cell? When do you split across multiple cells? There is no right or wrong answer here. There are both performance and administrative considerations.

With ND-centralized administration, each node has a node-agent process that communicates with the deployment manager to provide status on servers and applications. In addition, the deployment manager and node agents continuously compare notes on the configuration. Background communication between the deployment

COMPONENTS	CALCULATION	EXAMPLE
1. Total Physical Memory		4096 MB
2. Other processes (operating system and other processes) memory	OS + process 1 + process 2 …	400 MB
3. Deployment Manager Memory (if co-resident)	DM Max heap + Java interpreter	0 (DM not co-resident)
4. Node Agent Memory	NA Max heap + Java interpreter	128 + 42 = 170 MB
5. Memory Available for Application Servers	Total memory – other processes – node agent memory	4096MB – 400 – 170 = 3430 MB
6. Single Application Server Memory	App Srv Max heap + Java interpreter	256 + 64 = 320 MB
7. Maximum Number of Application Servers	(Available memory /Single application server memory) * margin for safety	(3430 / 320) * .8 = 8 application server maximum

TABLE 1. CALCULATING APPLICATION SERVERS

RUTH WILLENBORG

Ruth Willenborg is a senior manager at IBM. She has 17 years of experience in software development and is currently a member of the WebSphere Development Organization, where she is responsible for the WebSphere Application Server Performance team. She is co-author of ***Performance Analysis for Java Web Sites*** (Addison-Wesley, 2002). You can e-mail Ruth at rewillen@us.ibm.com.

manager and the node agent is lightweight and has very little effect on application-server performance, but a large number of nodes in the topology can result in longer response times for routine administrative actions and monitoring.

In addition, a large number of servers increases administration-console navigation. By default, the administration console lists 20 servers per page. If there are 200 servers, the administrator traverses 10 pages to get to server 200. As we will discuss later, features exist to make this task easier, such as server filtering by name or node.

Our personal preference was to limit the number of servers in the cell to approximately 60, either using quite a few small servers, with just a couple of application servers on each (i.e., 20 nodes with 3 servers each) or a few larger servers, with quite a few application servers (i.e., 3 nodes with 20 servers each). The choice of approach here depends on hardware availability, operating system choices, cost, etc. Find the right balance to meet your needs by understanding your administrative processes and performance requirements.

Tips for Administering Cells

No matter what size WAS ND system you have, here are a few tips that will make administering it easier and faster.

DEFINE RESOURCES AT CELL LEVEL

You can take advantage of the WAS ND cell architecture by defining resources at the cell level. Resources defined at the cell level are available to application servers throughout the cell. When a resource can be used by multiple application servers, defining it at the cell level enables configuration changes or updates to that resource to be made only once and apply across the cell. For example, often all members in a cluster use the same JDBC driver and data source. Defining the JDBC driver and data source at the cell level enables you to make tuning changes once at the cell level, rather than once per application server.

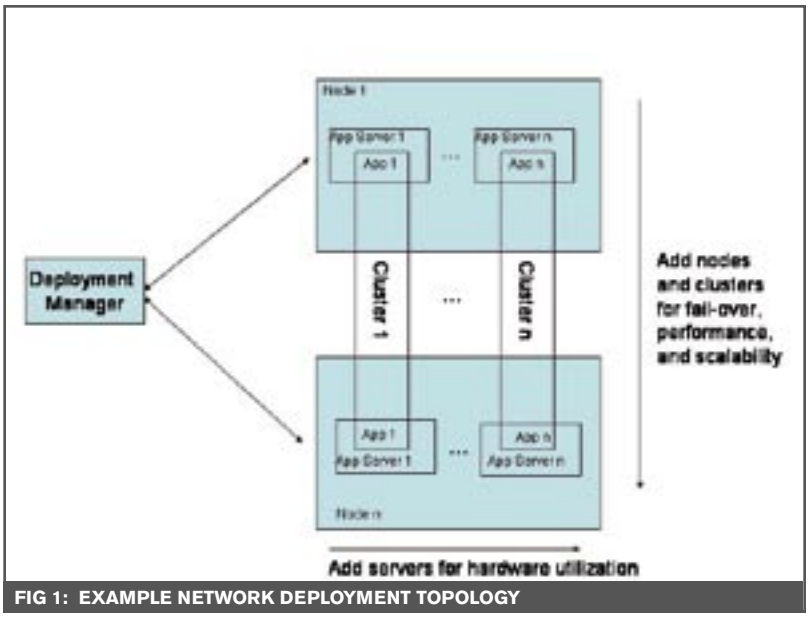


FIG 1: EXAMPLE NETWORK DEPLOYMENT TOPOLOGY

Use Naming Conventions

To assist in administering multiple servers, establish a good naming convention to keep better track of your topology. For example, include a number in the hostnames of your nodes, such as System1 and System2, and so forth. Then, name your cluster members ClusterMember1 and ClusterMember2.

Also consider renaming your log files with the corresponding number suffix, such as SystemOutNA1.log and SystemOutNA2.log, and SystemOutCM1.log and SystemOutCM2.log for your node agents and cluster members respectively. The larger your deployment, the more important it is to establish a good naming convention.

USE FILTERS TO NAVIGATE ADMINISTRATIVE CONSOLE

With a large number of applications, default settings on the administrative console require you to navigate with next and view entries one page at a time. Faster navigation is possible by setting a larger number of applications per page or filtering the application servers on this list. For maximum usability, we preferred navigating no more than three pages by keeping the total number of application servers listed less than 60.

USE A SCRIPT TO IMPROVE STARTUP FOR MULTIPLE, NON-CLUSTERED SERVERS

The $AdminControl startServer script command issues a request to start a server, and by default, waits for up to 10 minutes for the request to complete. This command works fine for starting one or two servers, but becomes a bottleneck when starting many individual servers; the next server cannot start until the previous server starts.

To speed up startup for multiple servers, you can use a script with a reduced wait time. A shorter wait time parameter setting returns control to your wsadmin script where you can start additional servers while previous start requests are processed. Note that setting a wait time of zero actually produces the same 10-minute wait time as the default.

In our 60-application server, single-node test environment, lowering the wait time to 1 second decreased server startup from over 40 minutes to under 5 minutes. A small wait time enables server start requests to process in parallel, which takes advantage of multiple CPUs in a system, or multiple systems. Starting multiple servers in parallel is the technique WebSphere uses when clusters are started.

Listing 1 includes a sample startup script. Logic is also provided to check server status.

Forget something?

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

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

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

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

netiq
Work Smarter.

"There is no magic formula that tells you exactly how to build your cell topology"

Use a Script to Improve Stop Time for Multiple, Non-Clustered Servers

The $AdminControl stopServer script command issues a request to stop an application server and waits for the request to complete. This command works when stopping a few servers but can be a long process when stopping a large number of servers.

Using the stop method of the server MBean is faster. This method returns control to your wsadmin script without waiting. This approach enables multiple stop requests to proceed in parallel.

Figure 2 shows the time it takes to stop from 5 to 100 application servers. As shown, the server MBean stop method is much faster than using the $AdminControl stopServer request. Stopping multiple servers in parallel is the same technique used by WAS when clusters are stopped. Listing 2 provides a sample stop script as well as logic to check for server shutdown.

DON'T AUTOSTART APPLICATIONS UNNECESSARILY

By default, WebSphere server startup processing starts all applications on the server. Choosing to defer the startup of the applications accelerates the initial server startup.

If there are applications installed that do not need to run until later, disable autostart for these applications and start the applications as needed. For example, you might want WebSphere samples available on a development server, but they are not required to continuously run. In our test of application server startup with 20 simple applications, startup time is reduced by half as a result of disabling autostart on the applications. Remember that time saved at startup is a cost later when the applications are started. To disable autostart, change the target mapping of the application to enabled=false.

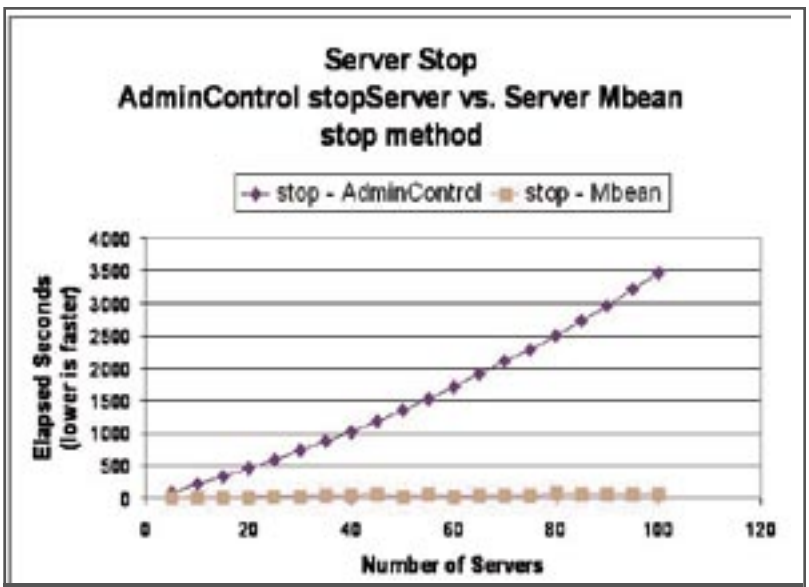


FIG 2: ADVANTAGE OF USING SERVER MBEAN STOP METHOD

Using the Administrative Console:

(1) Select the application: Applications > Enterprise Applications and then select an application to change.
(2) On the Configuration tab, go to Additional Properties and select Target Mappings
(3) Select the target and clear the checkbox to disable the target mapping

Using wsadmin script:

```
set deployments [$AdminConfig getid /
Deployment:$appName/]
set deploymentObject [$AdminConfig showAttribute
$deployments deployedObject]
set targetMappings [lindex [$AdminConfig showAttrib-
ute $deploymentObject targetMappings] 0]
$AdminConfig modify $targetMappings {{enable false}}
$AdminConfig save
```

Conclusion

WAS ND v5 provides a new infrastructure to manage and administer your servers and applications. Improvements in the architecture create a faster and more flexible administration experience when managing large WebSphere environments. New scripting capabilities provide you with the capability to quickly and efficiently manage your entire topology with automation and reliability. The administrative console can be used to check on your topology or modify it from any Web browser.

WAS ND v5 contains no hard limits on what you can do with your topology, but you should consider practical limits such as memory and administrative complexity as important elements in optimizing your network deployment performance.

References

- Joines, Stacy; Willenborg, Ruth; and Hygh, Ken. (2002). *Performance Analysis for Java Web Sites*. Addison-Wesley.
- Alcott, Tom. "J2EE Application Deployment: One or Many Applications per Application Server." http://www.software.ibm.com/wsdd.

LISTING 1: SAMPLE START SCRIPT

```
# Set the number of servers to start
set maxServer 50
set nodeName arcturus1
set event_flag 0
# In this loop, start all servers
set servNum 2
while {$servNum <= $maxServer} {
    set serverName serverX[format "%3.3d" $servNum]
    $AdminControl startServer $serverName $nodeName 1
    incr servNum
    }
# Start server1 also
$AdminControl startServer server1 $nodeName 1
puts "Start request issued to all servers"#
# Wait for start to complete
# First wait for maxServer Application Server MBeans to be
  created#
while {1} {
    after 10000 {incr event_flag}   ;# wait 10 seconds to
     check MBean creation
    vwait event_flag
    set serverMBeans [$AdminControl queryNames type=Server,
      processType=ManagedProcess,*]
    if {[llength $serverMBeans] == $maxServer} {
        break
    }
}
puts "MBeans created for all servers"#
# Then wait for each server to have a state of STARTED#
puts "Waiting for each server's state to be STARTED"

set waitForStart 1
while {$waitForStart == 1} {
    set waitForStart 0
    foreach server $serverMBeans {
        if {[string compare [$AdminControl invoke $server
         getState] "STARTED"] != 0} {
            after 10000 {incr event_flag}   ;# wait 10
             seconds to check MBean state
            vwait event_flag
            set waitForStart 1
        }
    }
}

This script checks for server startup every 5 seconds:

set template [$AdminControl queryNames  cell=$cellName,node=$
 nodeName,type=ApplicationManager,process=$serverName,*]
while {[llength $template] == 0} {
after 5000
set template [$AdminControl queryNames cell=$cellName,node=
 $nodeName,  type=ApplicationManager,process=$serverName,*]
}
```

LISTING 2: STOP SCRIPT

```
#
# Stop running application servers
#
set nodeName arcturus1

set event_flag 0

set serverMBeans [$AdminControl queryNames node=$nodeName,
 type=Server,processType=ManagedProcess,*]

#  Remove those servers from the serverMBeans list that
   should not be stopped
#  ....  For this example assume that no other servers were
         removed from
#  serverMBeans
set numOtherBeans  0

#  Stop the servers

foreach server $serverMBeans {
    $AdminControl invoke $server stop
}

#  Wait for servers to finish stopping - wait for
    Application Server MBeans to #  vanish

while {1}  {
    after 5000 {incr event_flag}   ;# wait 5 seconds to
     check MBean existence
    vwait event_flag
    set serverMBeans [$AdminControl queryNames node=$nodeName
     ,type=Server,processType=ManagedProcess,*]
    if {[llength $serverMBeans] == $numOtherBeans}  {
        break
    }
}
The following example checks for server shutdown every 5
  seconds:

set template [$AdminControl queryNames  cell=$cellName,node=$
  nodeName,type=ApplicationManager,process=$serverName,*]
while {[llength $template] != 0} {
after 5000
set template [$AdminControl queryNames cell=$cellName,node=
  $nodeName,  type=ApplicationManager,process=$serverName,*]
}

The script ends when the server starts, or continues with
  more script logic.
```

Solutions address industry-specific business needs

Having IT Your Way

BY PAMELA KLYM
& KENNETH LEUNG

To get a sense of the challenges facing retail and consumer product businesses, consider the once straight-forward sports shoe. Not too long ago, the choices regarding these shoes, quaintly called sneakers, consisted of a couple of brands (remember Keds vs. PF Flyers?), high tops or not, color, and later, canvas or leather. Today, athletic shoes are a large, varied, and extremely dynamic product category. Retailers and consumers alike must make choices based on function, brand, style, materials, price, and image. Furthermore, the choices change as quickly as does the preferred athlete to endorse a shoe and the hot brand.

The same burgeoning and quick-changing array of choices exists with soda, laundry detergent, electronics, and almost any consumer product you can name. And it means consumer product companies and retailers must work continually to produce and sell the right product, at the right place, the right time, and the right price.

"In the past, a consumer products sales person could go to a retailer and say 'Here are my products. Here's the price. How many can I place on your shelves?'," said Kenneth Leung, Marketing Executive, Consumer Products Industry, IBM Software Group. "Now, both sides must deal with promotions, co-marketing, discounts, special packaging, the need for better logistics, and much more."

In addition, modern consumers are "much harder to please. They want more convenience, higher quality, better service, but not necessarily to pay more," said Leung. And all this takes place as these industries deal with rapid globalization and retailer consolidation.

"The result is intense pressure to cut costs and get products to stores more quickly – without over-producing or over-buying," said Pamela Klym, Marketing Executive, Retail Industry, IBM Software Group. "Making their businesses more efficient can be a matter of survival," she said.

New Technologies Address New Industry Challenges

Consumer product and retail companies have a track record of bringing information technology (IT) to bear on business challenges. New technologies, such as integration servers, wireless support, and the ability to monitor business processes in real time, are bringing key capabilities to the effort. IBM has built many of these capabilities into its Middleware Solutions for Consumer Products (CP) and Retail. IBM's experience with these customers is the foundation for the solutions that are designed to address the industries' most pressing challenges.

The vertically-aligned solutions are part of IBM's effort to deliver industry middleware solutions. Customer buying behavior indicates that they prefer to buy solutions designed specifically to address business issues in their industry. Each solution contains the optimum combination of IBM's WebSphere, Lotus, Tivoli, DB2, and Rational middleware brands, combined with IBM's industry-specific middleware, applications from independent software vendors (ISVs), and industry-expert services.

There are five Consumer Products solutions and six for Retail. They are all modular, allowing companies to implement pieces at their own pace, based upon open standards and platform independence. They include industry-critical technologies such as Radio Frequency Identification tag (RFID), item synchronization, and product information management capabilities.

IBM's consumer products solutions include the IBM Middleware Solution for Consumer Products Item Management. It deals with the critical information that flows between suppliers and retailers without which no product could make it to store shelves. Traditional processes have resulted in 30% of product data to retailers being inaccurate and 60%

of orders needing to be reworked because of errors. As a result, it has taken an average of 10 days for a new item or price change to reach stores.

"IBM's solution for item management improves supply chain effectiveness by using Web technologies and item synchronization, product information management, and integrated RFID, which is the next generation bar coding technology," said Leung. "This improves collaboration with retailers."

Specifically, the solution increases item data accuracy and decreases order errors, improves inventory tracking, and decreases losses during transport. It also provides support for participation in the industry's third-party data pools like UCCnet. The result is less time between production and the retail store shelf, improved customer service from CP companies to retailers and retailers to consumers, and improved consumer satisfaction because the right product is more often at the right place at the right time.

"IBM's development team started working on these capabilities in December 2001," said Sesh Murthy, director of distribution sector solutions. "We worked on how to store the data in a two-part process that allowed the transfer of information in and out of an enterprise and the means for companies to use the data for critical internal capabilities," said Murthy.

One of the components in the IBM Middleware Solution for Consumer Products Item Management is WebSphere Product Center. IBM worked on this solution with Trigo Technologies, Inc. IBM's acquisition of Trigo closed on April 6 and since the planned acquisition was announced in March, it has contributed to several major customer signings.

Product information management middleware is increasingly playing an important role in retail and consumer products companies, among others. Standards in the retail and consumer products industries call for every item in a store to have 153 core attributes when in reality each item has anywhere from 300 to 4,000 attributes. A supermarket might have 300,000 items and a large supermarket chain may have 3,000 stores. All this data needs to be managed and that's one area where WebSphere Product Center comes into play. It acts as a central repository, enabling companies to centrally manage comprehensive product information, laying the foundation for global data synchronization and RFID initiatives.

Tapping Key Data Critical to Sales Teams

Consumer products companies also need to increase the effectiveness of their sales forces. Typically, sales teams lack key information to assist customers and react to critical changes such as pricing challenges. These buyers rely on data from numerous applications and data sources that are not integrated. As much as 60%–80% of their time is spent on administrative tasks. Best practices are not shared. There is also little career development structure or the ability to develop new skills where needed.

The Customer Management Workplace solution provides sales forces with critical information such as account profitability, product acceptance, trade promotion, joint forecasts, etc. This data is presented in a single, customized interface and applications and data are linked based on the user.

The solution improves collaboration and workflow management and has the potential to save buyers up to 10 hours per week because it is easier to access systems and find information. Thus, IBM's customer management solution allows sales teams to focus more on business development and manage opportunities and negotiations using dynamic information on sales and volume.

Consumer products companies have also faced challenges tapping consumer data to improve strategies and tactics. The IBM Middleware Solution for Consumer Products Information Management enable SCP firms to harness product and consumer data from retailers for insights that help them merchandise their items better. The solution's Integrated Point-of-Sale (iPOS) technology intelligently combines retail POS data with manufacturer information to create meaningful and timely product and category insights.

iPOS, which will support multiple retail POS formats, handles huge data volumes to quickly produce timely reports. The delivery of richer information faster allows more time for insight, so less time needs to be spent on data management. It also enables more effective management of product categories and use of measurements based on consumption rather than shipment.

Protecting Brands with Workflow, Governance

Brand management is also a challenge for CP companies. Many lack control in content creation and workflow involving areas such as product logos and packaging design, yet they need to maintain brand consistency across multiple channels and produce multiple images for different mediums. IBM's Middleware Solution for Consumer Products Brand Management provides workflow, planning, and data management solutions. Its Marketing Asset Management capabilities provide features for managing and collaborating with digital brand assets.

The brand management solution's workflow and governance features reduce the risk to brand equity based on elimination of misuse. They also lower the cost of holding, managing, distributing, and updating digital content because there is no need to recreate images. This enables faster product launch, pro-

motion, and marketing campaigns because workflow reduces approval times. Other benefits include streamlined version control; more capability to control the definition, tracking, and security of assets; and improved monitoring and management of intellectual property rights.

Like many industries, CP companies are facing increasing risk and regulatory requirements. Compliance can be challenging due to difficulty in consolidating and managing information from multiple units across geographic regions and a lack of documentation of business processes and controls.

The Middleware Solutions for Consumer Products Risk and Compliance Management provides real-time audit and traceability around certified revenue and income compliance requirements. Its Business Controls & Reporting Workplace provides workflow-based capability, allowing sales and finance departments to collaborate on certification of revenue. This is especially important in areas such as compliance with the Sarbanes-Oxley Act of 2002.

These capabilities allow businesses to identify key issues before they become critical, to respond faster to report requirements, and to retain history of events information for legal purposes.

Streamlining Supply Chains & Synchronizing Product Information

IBM's middleware solutions for retail companies are based on its long experience in the industry, which includes inventing the UPC Barcode in 1973. Today, 61 of the top 100 global retailers use IBM Point-of-Sale (POS) hardware (over 1.9 million installed in more than 100,000 stores in over 100 countries), more than 80% of the top retailers in the world use WebSphere, and more than 70% of the Fortune 100 use IBM MQSeries.

But "retailers are facing challenges like never before," said Klym, "which boil down to enormous competitive pressure to cut costs and the need to accelerate the process of getting items on store shelves because trends change so rapidly today."

One of the biggest challenges is to efficiently purchase the right products based on rapidly changing market demand. To get data to do this, retailers go to numerous internal and external sources. These processes are very costly, lack agility, and have resulted in poor forecasting and planning. The effect has been low product availability, forcing retailers to rely on mark-down selling, which cuts profitability.

The IBM Middleware Solution for Retail Merchandising provides more efficient processes through role-based, real-time access to a fully integrated merchandising environment. It allows retailers to maximize in-stock percentages and thus increase sales and improve customer service through on-time fulfillment. Retailers are also better able to minimize markdowns, protecting their margins, and more easily differentiate themselves.

Like consumer products companies, retailers need to streamline their supply chains. Today's manual item management processes are costly and result in slow product-to-store times. Retailers need to more quickly synchronize product information, maximize the percentages of goods and materials that are in stock, and improve customer service through on-time order fulfillment.

The Middleware Solution for Retail Item Management helps retailers accomplish these tasks by providing a fully automated and integrated set of data synchronization and product information management processes. By eliminating manual processes and resulting errors, retailers can trade item information with suppliers seamlessly, eliminating bottlenecks, manual labor, and costly errors.

Cutting Costs, Improving Return of Marketing Dollars

Retailers also need to cut costs associated with advertising, marketing, and promotions. They are faced with soaring print advertising costs and the need to rationalize ad space, high-cost static in-store signage outweighing its benefits, ineffective promotions and offers, and no single repository for product images.

The Middleware Solution for Retail Advertising, Marketing, and Promotions provides a collaborative project workspace that organizes and links these marketing efforts. It allows retailers to personalize and target promotions to attract customers and increase overall traffic. This solution also provides the capability to use digital media in stores.

Inventory management is another area in which retailers need to improve efficiency. Manual processes have traditionally made tracking inventory through warehouses, distribution centers, and stores very difficult. Retailers have been unable to locate inventory and match customer demand for a given item, regardless of channel and inventory location because of this lack of item visibility. They also lose revenue whenever an item is out of stock and theft is a problem.

IBM addresses these issues by providing better item visibility. The Middleware Solution for Retail Inventory Management integrates the current use of barcodes to track individual stock-keeping units (SKUs) with new radio frequency identification (RFID) technology, to track items at the pallet and case level, and to automate

June 28–
July 1, 2004

Moscone Center
San Francisco, CA

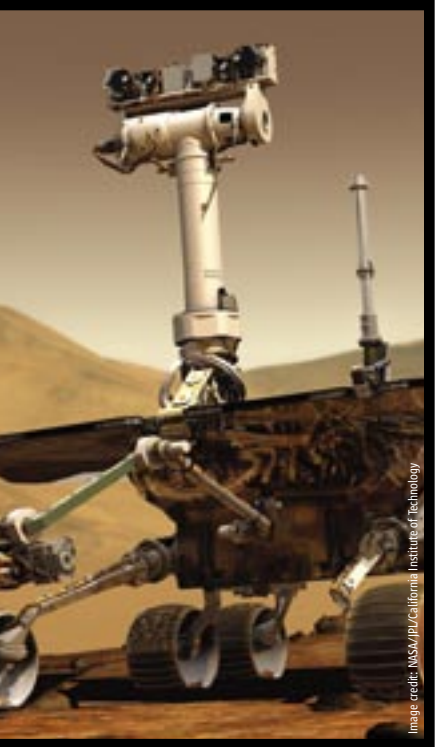


Image credit: NASA/JPL/California Institute of Technology

Benefit from four days of in-depth training and networking with the foremost leaders and creators of Java™ technology in:

- Enterprise
- Mobility
- Web Services
- Desktop

Java™

Go Anywhere You Want with Java™ Technology

It all starts at the JavaOne℠ conference.

Java™ technology is everywhere, improving the digital experience for everyone. It all starts at the JavaOne℠ conference, your source for cutting-edge knowledge and proven solutions. You'll learn how to tap into the full power of the Java platform: platform independence; compatibility with and between editions; open toolsets and technologies supported by a global developer community.

The JavaOne conference offers hundreds of in-depth technical sessions in:

Topic 1—The Foundations: Core J2SE™ Technologies
Topic 2—Core Enterprise Technologies
Topic 3—Java™ Technology on the Desktop
Topic 4—Java Technology for the Web
Topic 5—Java Technology for Mobility
Topic 6—Dissecting the Implementation: Solutions
Topic 7—Intriguing and Unexpected: "New and Cool"

JavaOne℠

Sun's 2004 Worldwide Java Developer Conference℠

Save $100! Register by June 27, 2004, and receive $100 off the full Conference package. Priority code: ADBLZKND

Register at java.sun.com/javaone/sf

Sponsored by

Sun microsystems

Produced by

MediaLive™ INTERNATIONAL

manual shipping and receiving processes, such as matching the purchase order, invoice, and shipping documents when goods are received. These new capabilities improve supply chain efficiency and automate document-matching.

Retailers have also faced challenges when attempting to streamline and synchronize business processes across disparate channels. Retailers' channels typically have fragmented order processes, varying views of inventory and orders, inconsistent buying experiences, and little ability to access inventory and order information regardless of location.

The Middleware Solution for Multi-Channel Management helps retailers consistently support multiple sales and fulfillment channels by providing a consolidated order platform and a fully integrated set of sales channels. It allows retailers to provide customers with a seamless and personalized cross-channel buying experience and allows them to work more cost-effectively with their suppliers.

Helping Over-Burdened Managers Improve Operations

Retailers also need to improve the efficiency and cost-effectiveness of store operations. Customers are demanding higher quality service while labor costs need to shrink. There is costly high employee turnover, while overburdened managers are tied to back-office systems whose disconnected applications provide poor visibility of inventory, sales and customer data.

The Middleware Solution for Retail Store Operations provides a Web platform that is affordable, manageable, and allows retailers to deploy game-changing applications. The solution increases access to information by using in-store devices. It speeds responses to inquiries that can improve employee knowledge and efficiency, increase customer satisfaction and loyalty, and streamline operating procedures, including sharing critical sales data with corporate systems.

Several pieces of industry-specific middleware needed to be built to create the valuable and efficient flow of data between the store and corporate headquarters, according to Murthy. "An adapter to the POS system was developed to access the data, IBM MQSeries Everyplace was used for the reliable transfer of the data in and out of the stores, and IBM WebSphere Message Broker was used to convert POS data to an industry-standard format," he said.

To develop the Manager's Portal for the store operations solution, the development team first built an architecture that laid out where to house functions, how to make connections, etc. Then they built portlets for connections to store and corporate applications and then technology to tie together data and display it. The team is continuing to develop the Store Integration Framework. "We had to deal with a shortage of IT skills and bandwidth in the stores," said Murthy. "In building this, we seek to move this functionality into the core middleware to get capability to deploy, monitor, and manage the key applications at the outlet (store)."

–continued from page 4

SOAs Help Businesses Deal with Change

BY JACK MARTIN

SOA governance model, an SOA reference model architecture, and an SOA transition plan to realize your specific SOA is an absolute must.

If you have data in a legacy system, and almost everyone does, you should think about where the potential value is in exposing legacy data and linking it to new business processes in the SOA. Leveraging the legacy applications you have with SOA can give you a huge bang for your buck.

Once your business has been divided into a set of discrete activities supported by people, processes, and systems, the pieces can be measured and benchmarked, thereby identifying the weaknesses and strengths of individual business activities, allowing you to better select areas for process improvement.

The challenge of any information technology project is effectiveness – making sure the deployed technology solves the original business problem as efficiently as possible. The use of SOAs is a major step in the right direction.

Reach Over 100,000

Enterprise Development Managers & Decision Makers with...

Offering leading software, services, and hardware vendors an opportunity to ***speak to over 100,000 purchasing decision makers*** *about their products, the enterprise IT marketplace, and emerging trends critical to developers, programmers, and IT management*

Don't Miss Your Opportunity to Be a Part of the Next Issue!

Get Listed as a Top 20* Solutions Provider

For Advertising Details Call 201 802-3021 Today!

*ONLY 20 ADVERTISERS WILL BE DISPLAYED. FIRST COME FIRST SERVE

SYS-CON MEDIA

The World's Leading i-Technology Publisher

Personal Java Development with WebSphere Studio Application Developer

Utilizing WSAD to develop and deploy Java code

BY JIM LIDDLE

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

This article is intended to help you understand what support the Sony-Ericsson P800/P900 has for Java and to show you how to install the necessary Symbian software required for developing Java solutions for these platforms. It will discuss how WebSphere Studio Application Developer can be used to develop the Java code, how this code can then be packaged using the Symbian development kit utilities, and how to deploy it to the Symbian P800/P900 emulator.

The Sony-Ericsson Px range is one of the new breeds of smartphones that combines phone functionality with that of a PDA. The Px smartphones provide you with GSM tri-band capability along with GPRS support. The phones use Symbian OS v7 with 12MB of internal user storage and a maximum of 128MB external user storage, provided by the Sony Memory Stick Duo card format. The screen size is 208 x 320 pixels in 4096 colors (65K colors for the P900). The phones have a flip cover that becomes a "virtual" flip if removed. With the flip closed, 208 x 144 pixels are visible. When the flip is open, the screen becomes touch sensitive.

The Symbian Px range provides support for Personal Java and MIDP/CLDC. The P800 provides support only for MIDP 1.0, whereas the newer P900 offers support for MIDP 2.0. MIDP 1.0 applications developed for the P800 should be compatible with the P900 as well.

For MIDP development, you can use IBM's Websphere Device Developer or Sun's Wireless toolkit to produce MIDP-compliant JAR files. These can then be deployed directly to the phone or to the emulator. Note that the P800 does not use the JAD file. The JAD file itself is not required by the MIDP 1.0 specification. Instructions on the various ways to install Midlets to the P800, including over-the-air (OTA) provisioning, can be found in the Symbian Knowledge Base entry listed in the Resources section.

The Personal Java implementation is based on JDK1.1.6, with some device-specific extensions as well as some restrictions. Personal Java has significant support on both PDAs and smartphones and provides a fully-fledged JVM that is not tied by the MIDP size restrictions, and has the capacity to utilize the external storage format. This, coupled with the ability to use AWT from a UI perspective, provides a capable platform for Mobile Application Development.

I find that one of the biggest challenges to individuals wishing to pursue mobile development is understanding what is needed and how to set up the environment. To enable development of Personal Java applications for the Sony Ericsson Px series, you first need to download the full Symbian/UIQ software development kit. This is free of charge and is available at http://developer.sonyericsson.com. The P800 supports the Symbian OS 7.0 UIQ 2.0 and the P900 supports Symbian OS 7.0 UIQ 2.1. UIQ is an acronymn for "User Interface Quartz". Quartz refers to one of the three original reference designs produced by Symbian. It defines a family of devices. UIQ 2.0 and 2.1 are used by the Px Series whereas Series 60, for example, is used by the Nokia Communicator Smartphones.

We will choose to develop against UIQ 2.0 because anything compiled and deployed with it will work on both the P800 and the P900. The SDK is comprised of a Perl Interpreter, a P800 emulator, as well as the required Java libraries and examples. Exactly why the environment is such a cluster of technologies is not entirely clear, and hopefully this will be updated and streamlined sometime soon.

After you have downloaded the separate installation packages, you need to install these onto a Windows-based OS. There are instructions provided on how to do this, but I would also like to add that, on installing the SDK, when asked what to install, select 'everything' when you are asked what to install. Even if you don't plan to develop in C++, I would suggest installing this part, as my experience suggests that you will have problems later only if you fail to install it. For those of you who are using Linux as your development platform, it is possible to

enable development for Symbian UIQ devices on Linux but it is outside the scope of this article.

Once you have installed the Symbian software, there are a few things that need to be done to ensure that the development environment and emulator are set.

The first thing you need to do is to set the 'EPOCROOT' environment variable. I suggest you do this via the Windows environment variables settings. Set this to \Symbian\UIQ_70\ .

As Symbian OS Kits are designed to support developers who need to work on different projects targeting different types of phones, you next need to register the correct device with the environment. This is done using the 'devices.exe' command at the command prompt. To do this, run the devices command: 'devices -setdefault @UIQ_70:com.symbian.UIQ'. You need only do this once, as the device details are saved in the environments configuration file. This is actually an XML file which can be found at 'c:\Program Files\Common Files\Symbian\devices.xml.'

The emulator maps the Symbian OS file system onto the PC's file system. It performs this drive mapping at run-time. The emulator needs to have two default drives: z:, which maps to the phone's ROM, and another which represents the primary writable file system on the phone. Both drives should be pre-configured, but if you are having problems, then you can look inside epoc.ini with a text editor. This can be found at <Symbian Install DIR>\UIQ_70\epoc32\data\. If it does not already exist, you can create it. An example is given below:

```
_epoc_drive_j D:\Symbian\UIQ_70\epoc32\java
```

You can add further virtual drives, and reconfigure the standard drives to map to alternative locations, however any mapping for the z: drive must specify a directory named z.

The emulator can now be launched by invoking the emulator launcher 'epoc' from the command line. This invokes and runs the emulator executable file epoc.exe for the currently active device, as defined by the devices command that we previously set. If everything has been set correctly, the emulator should launch (see Figure 1).

FIG 1: SYMBIAN 7 UIQ 2.0 P800 EMULATOR

Once you have the software installed you essentially have:

- A mobile phone emulator that has been customized for the P800
- A 'virtual' Windows file system-based implementation of the phone directory structure
- Software tools that are able to package the Java JAR files into a 'SIS' file that is the default file format for applications on the P800/900
- Personal Java documentation for the P800/900

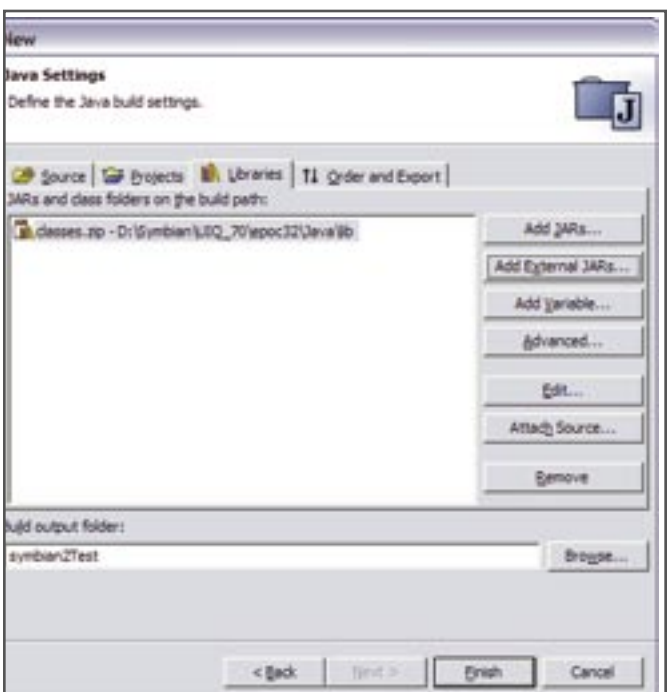


FIG 2: ADDING CORRECT LIBRARY TO WSAD SYMBIAN TEST PROJECT

The UIQ SDK contains several Java libraries. The libraries of note are:

- ***classes.zip:*** The Personal Java API library
- ***qawt.jar:*** Provides support for the UIQ look and feel; a modification of the Java AWT Frame class.
- ***javaphone.jar:*** The JavaPhone API library
- ***util.jar:*** A library to provide access to Symbian OS utilities.

For reference, the classes.zip library is located in the path epoc32/java/lib/ inside the UIQ installation. The other three libraries are located in epoc32/java/ext/.

When you build your Personal Java application on the Px series, you will need a mechanism to debug the deployed application while it is running on the emulator, i.e., receive output from Java's System.out stream. Fortunately, Ericsson provides this functionality in the form of a small application called 'Redirect'. It is not installed by default on the phone emulator, but can be found at <your-sdk-installation>\epoc32\winscw\c. Launch the P800 emulator and install this from

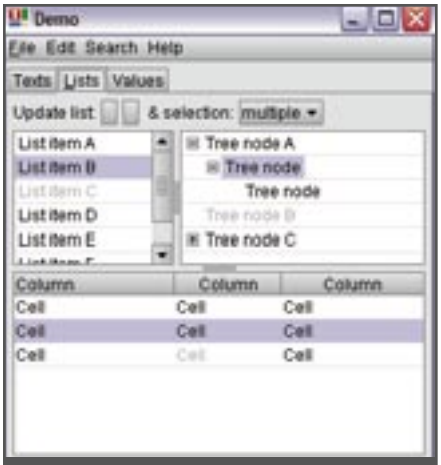


FIG 2: THE THINLET DEMO.JAVA FILE RUNNING WITHIN WEBSPHERE STUDIO.

the application Launcher menu, by selecting the Redirect.sis file.

To recap, we now have the UIQ SDK installed and the correct parameters set to allow us to use the emulator. We are now going to use WebSphere Studio Application Developer to compile a Personal Java application and use the Symbian SDK utilities to package it so that it can be installed to the P800 emulator.

Rather than build a small "Hello World" Personal Java application, we are going to take an existing Personal Java application and import it into WSAD. The Personal Java application we will use also addresses one of the criticisms of mobile development of Java compared to Mobile development for .NET, namely the lack of rich UI components such as tree, tab, etc.

Thinlet is a 'lightweight' UI toolkit that supports Java 1.0 to 1.4, Personal Java, as well as Personal (Basis) Profile. Swing isn't required. It comprises a single Java class that handles user interaction, and calls business logic. The graphic presentation is described in an XML file and the application methods are written as Java code. Its compressed size is 38KB, and it is LGPL licensed.

We are going to download the demo application, import this into WebSphere Studio, verify that it will compile and run against the Symbian Java API Library, and then use the Symbian SDK to package it and deploy it to the phone. The Thinlet download demo can be found at www.thinlet.com.

After you have downloaded the demo file, launch WebSphere Studio and choose to start a new Java project, which we will simply call Test. Ensure that you remove the default rt.jar library and add the classes.zip library from the Symbian directory.

Once the new project is created, we need to import the Thinlet demo into the project. Right-click the root level Test project, choose Import, and select File system. Navigate to the directory where you unpacked the Thinlet demo file and choose the 'src' directory. From the /src/thinlet directory we need to import the Thinlet.java and FrameLauncher.java runtime files. From the /src/thinlet/demo directory select all the files. Once the files have been added, rebuild all the project files from the Tools Menu.

You can now choose to run the demo.java file or the calculator.java file from within WebSphere Studio.

The next thing we need to do is to package this as a JAR file. Within WebSphere Studio, choose to export to a JAR file(file->export->jar file). Deselect the .classpath and .project files and choose the destination location. Note that you will need to generate the .manifest file to make the JAR executable. Choose thinlet.demo.Demo' as the executable main class for the application entry point. The Symbian phone refers to this when launching the application so it is important to ensure that this is correct. Save the JAR file as test.jar. Note that you should not use filenames over 8 characters, as the emulator will not recognize the file.

Now that we have the executable Personal Java JAR file, we need to package it so that it can be deployed and used on the Symbian phone.

AIF Builder is an application that is installed as part of the Symbian SDK and enables you to create the required Symbian files that need to exist before final packaging. These include an Application Information File (.aif), an application file (.app), and a text file that specifies the command line to the JVM when the application is launched. AIF Builder is invoked from the command line using 'AifBuilder'. This can be found at <Symbian Install DIR>\UIQ_70\epoc32\tools\AIFBuilder.bat.

Once AIF Builder is launched, select Java as the application language and enter the application name as 'test'. A unique identifier then needs to be entered for the application. You can use a number in the range 0x01000000 to 0x0fffffff for testing purposes. Each application has to have its own unique ID to run correctly. Commercial UIDs can be obtained from Symbian. We will use 0x01000035 as a test UID. Against the Java Command use Line text enter "-cp \system\Apps\test\test.jar thinlet.demo.Demo".

Under the UI Variants Tab select 'UIQ' and also the location to generate files. Here we can also choose to

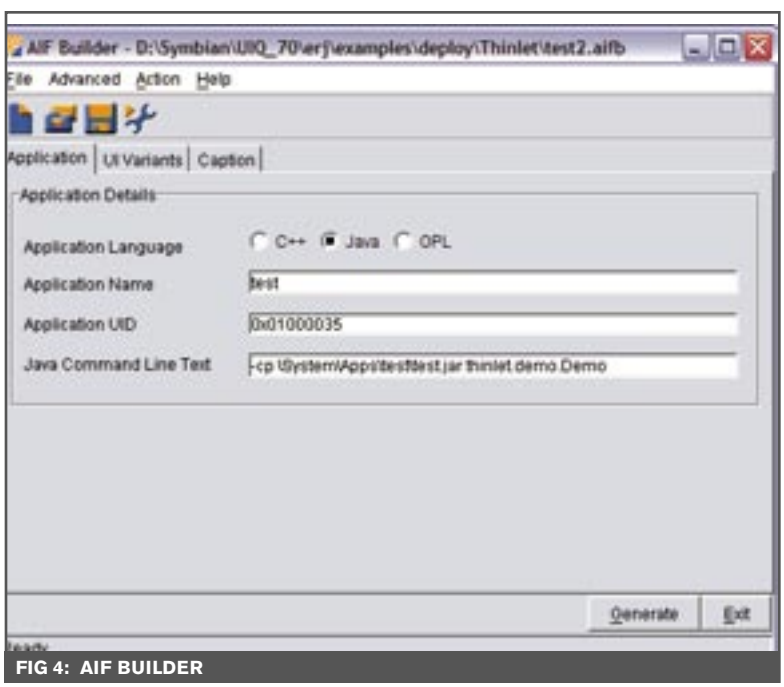


FIG 4: AIF BUILDER

Subscribe Today!

– INCLUDES –
FREE DIGITAL EDITION!
(WITH PAID SUBSCRIPTION)
GET YOUR ACCESS CODE INSTANTLY!

*The major infosecurity issues of the day... **identity theft, cyber-terrorism, encryption, perimeter defense, and more** come to the forefront in* **ISSJ** *the storage and security magazine targeted at IT professionals, managers, and decision makers*

Editorial Mission

Greater Collaboration and Efficiency Through Education

✓ ***ISSJ*'s editorial mission is to showcase proven solutions** that will guide, motivate, and inspire senior IT and business management leaders in the planning, development, deployment, and management of successful enterprise-wide security and storage solutions.

✓ ***ISSJ*** brings together all key elements of data storage and protection, and presents compelling insight into the benefits, efficiencies, and effectiveness gained by focusing on these two critical areas of IT simultaneously.

✓ ***ISSJ*** is an objective, critical source of information that helps storage and security managers make informed management decisions about what is working today, and what they need to plan for tomorrow, and is the only publication that focuses exclusively on the needs of IT professionals who are driving the enterprise storage architecture/infrastructure while pursuing and incorporating the latest security technologies.

✓ ***ISSJ*** achieves our mission by delivering in-depth features, practical "how-to" solutions, authoritative commentary, hard-hitting product reviews, comprehensive real-world case studies, and successful business models, written by and for professional IT storage and security practitioners.

SAVE 50% OFF!
(REGULAR NEWSSTAND PRICE)

Only $39.99 ONE YEAR 12 ISSUES

www.ISSJournal.com
or
1-888-303-5282

SYS-CON MEDIA
The World's Leading i-Technology Publisher

FIG 5: INSTALLING THE THINLET DEMO TO THE P800 EMULATOR

FIG 6: THE THINLET DEMO RUNNING ON THE P800 EMULATOR

select the icons that we have developed for the application or we can invoke an icon creator from the create button. We won't select any that result in a default icon being used. The last thing to do is to select the language(s) that you wish to be supported under the Caption Tab.

Once all of the above has been completed, choosing generate will result in three files being created, test.app, test.aif, and test.txt.

We have one last remaining step to complete before we can generate the sis installation file that the phone emulator recognizes. This is to create a package file. Package control files (extension .pkg) define the files to be packaged in a sis installation file. These currently have to created by hand. Listing 1 shows the package file for our test application with self-explanatory comments.

Once you have typed this information into a text editor, save it in the same directory as the generated files, calling it test.pkg. We can now use this to generate the .sis file. Makesis is the installation file generator provided by Symbian and is located at:

```
<Symbian Install DIR>\UIQ_70\epoc32\tools\MAKESIS.
exe.
```

Open a command line prompt within the directory where your generated files and package file resides. Executing the 'makesis' utility against the package file ('makesis test.pkg') will create a sis installation file called test.sis.

To enable you to install this file to the emulator, it has to be placed correctly within the virtual directory structure to enable the emulator launcher application to find it. The correct place for this is:

```
<Symbian Install DIR>\UIQ_70\epoc32\winscw\c.
```

Also note that you can quickly test the application before packaging it as a sis file by putting the .app, .aif, .txt, and JAR files directly within the emulator virtual directory structure at the following location:

```
<Symbian Install DIR>\\UIQ_70\epoc32\winscw\c\sys-
tem\Apps\<Your App DIR>
```

No icon is installed, but the application can be started by browsing to the correct directory using QFileman, and the Symbian emulator file explorer tool, and selecting the .app file.

Launch the emulator and choose the install option and you will see the 'test.sis' package ready to install. Following the default install instructions will result in a new application being installed called "test". Once you have installed it, launch the application and you should see the same application that we ran within WebSphere Studio running directly on the phone itself (see Figure 6).

Summary

To recap, we have given an overview of the Symbian Smartphone Px series, described how to install the SDK, how to develop Personal java Applications in WebSphere Studio, and how to package and deploy them to the phone emulator.

The age of the smartphone/smartPDA is definitely upon us and there will undoubtedly be more organizations wishing to make use of this technology for corporate applications. Although Sun is positioning MIDP as the prevalent mobile standard, most of the high-end PDAs and smartphones are equipped with a Personal Java-based Virtual Machine. This includes PDAs such as the IPAQ, the Palm, and other smartphones such as the Blackberry. This environment is currently ideally suited as a platform for ubiquitously accessible mobile applications, whatever the device.

Resources

- *Symbian Knowledge Base article on deploying Midlets to P800:* www3.symbian.com/faq.nsf/0/b105ed619a17c35580256d40005693d7?OpenDocument
- *Thinlet download demo:* http://thinlet.sourceforge.net/thinlet-2004-03-07.zip
- *Obtaining Commercial UIDs from Symbian:* UID@Symbiandevnet.com
- *Sony Ericsson Developer Network:* UIQ SDK downloads can be found at http://developer.sonyericsson.com .

LISTING 1: PACKAGE FILE LISTING FOR THINLET TEST APPLICATION

```
;Basic install file for Thinlet Demo Header
; Installation header – contains App Name, UID,  Version (1.0.0) and
type (Application)
#{"test"},(0x01000035),1,0,0, TYPE=SA
;
; Below used for any Symbian based phone and needed by emulator to
run
(0x101F617B), 2, 0, 0, {"UIQ20ProductID"}
;Below is only used when deploying to an actual P800 – above cannot
be then be used
;(0x101F80BE),1,0,0,{"SonyEricssonP80xPlatformProductID"}
;
; Only two files to install for the minimal application
"Test.app"-"!:\system\apps\demotest\test.app"
"Test.aif"-"!:\system\apps\demotest\test.aif"
"Test.txt"-"!:\system\apps\demotest\test.txt"
"Test.jar"-"!:\system\apps\demotest\test.jar"
; If we had generated our own icon then we would install it using
something like below
;"Test.mbm"-"!:\system\apps\demotest\Test.mbm"
;
; Required files
; None
;
; Component .sis files
; None
```

Utilizing portals to extend users' capabilities

Portlets and Portals Design Overview

BY THINA NATARAJAN

Thina Natarajan is a principal solution architect for the Application Infrastructure Management (AIM) group at Candle Corp. Natarajan optimizes IT architecture, development, and deployment phases for global clients. He has recently been focusing on J2EE performance tuning best practices.
thina_natarajan@candle.com

Portals enable users, employees, customers, and suppliers to expand beyond their traditional boundaries by providing a single common interface to access globally dispersed content and applications. Most important, portals also allow users to communicate with other enterprise applications, content factories, and processes. Dynamic portal capabilities also provide personalization that enables users to create tailored views of information or receive customized content. In addition to personalization capabilities, portals allow users to publish, share, and subscribe to various documents in an enterprise, regardless of original format.

Portal technologies, such as WebSphere Portal, support important back-end operations that provide the flexibility, scalability, and security required for any robust Web-based environment. Advanced portals include capabilities such as single sign-on (SSO), authentication and authorization services, directory services, content management, collaboration, mobile device support, search and taxonomy services, accessibility support, and internationalization.

Effective portal implementations require careful planning and execution. At the core of successful portals are personalization policies that help ensure user privacy and data security. This article explores the potential of portal personalization capabilities, security, and privacy considerations when designing portals and portlets, and outlines portlet design best practices that support benchmarks for flexibility, scalability, extensibility, and performance.

Portlets: Building Blocks for Portals

From a technical standpoint, the portlet architecture is an extension of the Java servlet architecture. It is a small piece of an application that is plugged into a Web page or assembled in a portal to create a single enterprise application. Portlets are also assembled into portal pages, which form a portal implementation.

Portlets, in simple terms, can be viewed as reusable sets of Java code that render HTML code. Similar to other Java technologies, the role and benefits of portlets vary based on IT staff and user perspectives. Application server administrators manage portlets as J2EE applications. Similarly, Java developers categorize portlets as configurable servlets that have additional attributes and include the potential to communicate with external content providers.

Personalization Policy

Dynamic and flexible portal capabilities enable users to receive targeted information based on their personal interests and preferences, as well as business requirements. Users must first provide personal information to establish a user profile.

Policies and processes associated with the portal owner's use and protection of site visitor information is a key consideration. Personalization policy guidelines determine how portal user information is handled. This policy applies to the entire portal and affects the design and usage of all portal features.

Portal personalization is a data-intensive process. It requires the use of personal data in a context that provides maximum value to portal users and portal owners. Successful personalization requires a continuous cycle of data collection, user profiling, and tailored content delivery. Most importantly, content delivery must adhere to user requests for information, including the types of content requested and the content delivery mechanism.

Personalization technology inherently gathers user data through explicit and implicit methods, making it crucial for portal privacy policies to encompass all means of personal information collected via multiple portal channels. Trust between users and portal providers is integral to the success of a portal environment. Users must be reassured that their personal information

is safeguarded and that their privacy is protected in order to establish the level of trust required for a successful portal environment.

Because security and privacy considerations provide a foundation for successful portals, J2EE architects and developers must establish the following parameters before designing new portals.

- Determine the level of personalization the portal will provide.
- Clarify the amount of information users will be required to share.
- Determine how the portal will profile users based on portal usage patterns and other implicit methods.

An established personalization policy will help answer these questions and provide a framework for building sophisticated personalization capabilities. Moreover, it will deliver an architectural base for the design of individual portlets. IBM, for example, is establishing a Personalization Value Space (PVS) methodology to evaluate how levels of personalization affect users and portal providers. PVS examines personalization across four parameters:

- Personalization policies (e.g., user control of personal data)
- Feature categories (e.g., clickstream analysis)
- User context (e.g., level of interest or interaction with the portal content)
- Business context (e.g., how portal content supports organizational objectives)

Portlet Application Design Principles

In addition to addressing the personalization policy issues, there are a number of portlet design best practices that will support overall portal benchmarks for flexibility, scalability, extensibility, and performance. There are six key portlet application design principles.

Principle 1: Design Your User Interface (UI), Model Your Portlet

Good portlet design does not equal good portlet UI design. An effective design supports portlet reuse and enables multipage rendering without modifications. Organizations must first evaluate page and site design to eliminate any roadblocks to reuse or rendering problems before building portlets. A sitewide UI policy that identifies and removes these roadblocks will serve as a cornerstone to sound portlet development.

Principle 2: Unleash the Power of Simplicity

The ideal portlet is narrow in its focus, reusable in its entirety, and crystal clear in its message. Developers should limit the information displayed to include only content required for task completion. If a portlet is part of a use-case model that requires the display of large amounts of Web-page information, developers should break the model into standalone "atomic" use cases. Each use case then translates into a single portlet. A group of related portlets can be connected via parameterized URLs, an approach that minimizes the initial amount of content users receive to maintain a high level of performance. Users have the option to access additional personalized information as needed.

Although displaying all of the Web pages as URLs is permissible, the size and amount of information should be manageable. Many Web developers believe the initial splash page should be as low as 30K, which is typically the equivalent of one or two graphics and text on a page.

Principle 3: Remember, in the End, It's Just HTML

Although portlets are written in Java, the browser renders them through HTML. It is paramount to employ HTML coding best practices because the simpler the HTML, the faster the page load time. While more forgiving than its successor – Extensible Markup Language (XML), it is important to disallow mismatched HTML tags and discourage the use of browser-specific features during portlet development.

Principle 4: Familiarize Portlets Into One Family

Cascading style sheets (CSS) support a browser-independent method for establishing a uniform portlet look and feel. A portal can then use predefined CSS for text formatting directly or through application program interfaces (APIs). However, it is important to ensure that there are no absolute CSS positions and that formatting options, such as fonts and colors, are not hard-coded.

Principle 5: Model Your Data, Control Your View

The importance of separating development and object-oriented design cannot be understated. A Model/View/Controller (MVC) architecture is preferred when rendering content at the UI level through Java Server Pages (JSPs),

"Effective portal implementations require careful planning and execution"

Extensible Style Language (XSL), and related technologies. MVC enables developers to segment the application into the following three sections to simplify development and design issues.

- The controller level oversees the addition of new user data and determines which data should be displayed (imputing, as well as marshalling/demarshalling of data parameters).
- The model level stores the data.
- The model layer abstracts the complexity of data retrieval and information gathering.
- The view level displays the information.

The portlet API provides many features for an MVC-based design, including tag libraries for JSPs. Portlet APIs also support Java Community Process standard JSR 168 features, such as coordination of user action and action sequences.

Principle 6: Promote Diversity Through Integration

A portal typically integrates diverse sets of information into a single user-friendly view. Depending on the data source, integration can become a complex multiphase exercise. (One example is a portal that screen-scrapes CICS applications to provide read-only access to legacy applications.) It is important to note that model-level integration simplifies portlet design. However, if model-level integration is not feasible due to needs for real-time information or other business requirements, organizations should consider adopting a service-oriented architecture (SOA) integration strategy. The SOA model shields integration complexities from portlets. The tradeoff, however, is delayed UI response times.

Conclusion

Noted fashion designer Issey Miyake once said, "Design is not for philosophy – it's for life." A good portal design can be used for the life of the application and extends the life of the application. The principles discussed throughout this article can help organizations create effective designs that translate into scalable applications.

Resources

- Rodriguez, Juan, et al. (2003). "IBM WebSphere Portal v5 – A Guide for Portlet Application Development." IBM Redbooks.
- Hoffing, Larissa. (2002), "Portlet Development for WebSphere Portal 4.1 – Best Practices." IBM Software Group.
- Guidelines for Building Portlets in Oracle9*i*AS Portal: www.oracle.com.
- Karat, C.M. et al. (2003). "Personalizing the User Experience on IBM.com," *IBM Systems Journal*. Vol. 42, No. 4: www.research.ibm.com/journal/sj/424/brodie.pdf

Excerpt from Chapter 9: Security

Cisco Networking Simplified

BY PAUL DELLA MAGGIORA & JIM DOHERTY

Network Security

The following sections describe the different categories of network security.

Identity

Identity is the identification of network users, hosts, applications, services, and resources. Examples of technologies that enable identification include Remote Authentication Dial-In User Service (RADIUS), Kerberos, one-time passwords, digital certificates, smart cards, and directory services.

Perimeter Security

Perimeter security controls access to critical network applications, data, and services so that only legitimate users and information can access these assets. Examples include access lists on routers and switches, firewalls, virus scanners, and content filters.

Data Privacy

The ability to provide secure communication is crucial when you must protect information from eavesdropping. Digital encryption technologies and protocols such as Internet Protocol Security (IPSec) are the primary means for protecting data, especially when implementing virtual private networks (VPNs).

Security Monitoring

Regardless of how security is implemented, it is still necessary to monitor a network and its components to ensure that the network remains secure. Network-security monitoring tools and intrusion detection systems (IDSs) provide visibility to the security status of the network.

Policy Management

Tools and technologies are worthless without well-defined security policies. Effective policies balance the imposition of security measures against the productivity gains realized with little security. Centralized policy-management tools that can analyze, interpret, configure, and monitor the state of security policies help consolidate the successful deployment of rational security policies.

A company's network is like any other corporate asset: It is valuable to the success and revenue of that company. More than ever, the corporate computer network is the most valuable asset of many companies. Therefore, it must be protected. Generally, middle- to large-size companies appoint a chief security officer, whose job is to develop and enforce corporate security policies. Security threats present themselves in many forms:

- A hacker breaking into the network to steal confidential information or destroy corporate data
- A natural disaster such as a fire, tornado, or earthquake destroying computer and network equipment
- A disgruntled employee intentionally trying to modify, steal, or destroy corporate information and devices
- A computer virus
- An act of war or terrorism

Common security threats introduced by people include the following:

- Network packet sniffers
- IP spoofing
- Password attacks
- Distribution of sensitive internal information to external sources
- Man-in-the-middle attacks

Internet security is also a big concern given the exposure of corporate data resources to the publicly accessible Internet. Traditionally, you could achieve security by physically separating corporate networks from public networks. However, with corporate Web servers and databases – and the desire to provide access to corporate resources to employees over the Internet – companies must be especially diligent in protecting their networks.

Another recent area for security concern is wireless networking. Traditional networking occurred over physical wires or fibers. However, the current trend is to

ABOUT THE BOOK

Cisco Networking Simplified by Paul Della Maggiora and Jim Doherty

ISBN: 1-58720-074-0
Publisher: Cisco Press (www.ciscopress.com)

provide networking services over radio frequencies. Companies are installing wireless networking in their buildings so employees can link to the corporate network from conference rooms and other shared locations from their laptop computers. Additionally, service providers are now offering public wireless Internet services.

Identity and Network Access Control

You can define identity terms of authentication and authorization:

- A computer or computer user identifies itself to the network or network resources.
- Authorization occurs after authentication. After the computer or user successfully identifies itself, the network or server authorizes the individual or computer to perform certain things with a certain level of access.

802.1x is a link layer protocol used for transporting higher-level authentication protocols defined by the Institute of Electrical and Electronic Engineers (IEEE).

One form of authentication occurs through the exchange of passwords. This form is generally a one-way transaction in which a user or computer identifies itself to a network or server.

A popular method for securely identifying a machine or individual uses digital signatures. For example, if you send an e-mail to someone, he might want to verify that you were indeed the originator of the e-mail. Algorithms such as Secure Hash Algorithm (SHA), Message Digest 5 (MD5) (similar to checksum), and triple Digital Encryption Standard (3DES) encrypt and securely "sign" the message. Then, the sender and receiver match public and private keys. The combination of these methods allows both parties to trust (or not trust) each other when exchanging information.

Visit www.ciscopress.com for a detailed description and to learn how to purchase this title.

Paul Della Maggiora, CCIE No. 1522, is a manager for the Cisco Systems Enterprise Solution Engineering Group and developed their Networking Bootcamp for new Cisco TAC engineers. He participated in network management product development as well as multiple customers' network designs and assists Cisco customers with managing their networks.

Jim Doherty is a solutions marketing manager with responsibility for routing and switching for Cisco Systems. He has taught professionals in both academic and industry settings on a broad range of topics including electric circuits, statistics, economics, and wireless communication method

TOP 13 SECURITY VULNERABILITIES

1. Inadequate router access control.
2. Unsecured and unmonitored remote access points, providing easy access to corporate networks.
3. Information leakage revealing operating system and application information.
4. Hosts running unnecessary services.
5. Weak, easily guessed, and reused passwords.
6. User or test accounts with excessive privileges.
7. Misconfigured Internet servers, especially for anonymous FTP.
8. Misconfigured firewalls.
9. Software that is outdated, vulnerable, or left in default configurations.
10. Lack of accepted and well-promulgated security policies, procedures, guidelines, and minimum baseline standards.
11. Excessive trust domains in UNIX and NT environments, giving hackers unauthorized access to sensitive systems.
12. Unauthenticated services such as the X Window System.
13. Inadequate logging, monitoring, and detection capabilities.

WebSphere Journal Coming Next Month...

WHY WEBSPHERE

Tom Inman, IBM's Vice President for Product Management and Marketing, IBM WebSphere Software, talks about why companies choose the Websphere family of products over other similar products. He is joined by some of their WebSphere Service Provider partners, who talk about the value WebSphere provides to their customers.

AT THE HEART OF AN ENTERPRISE

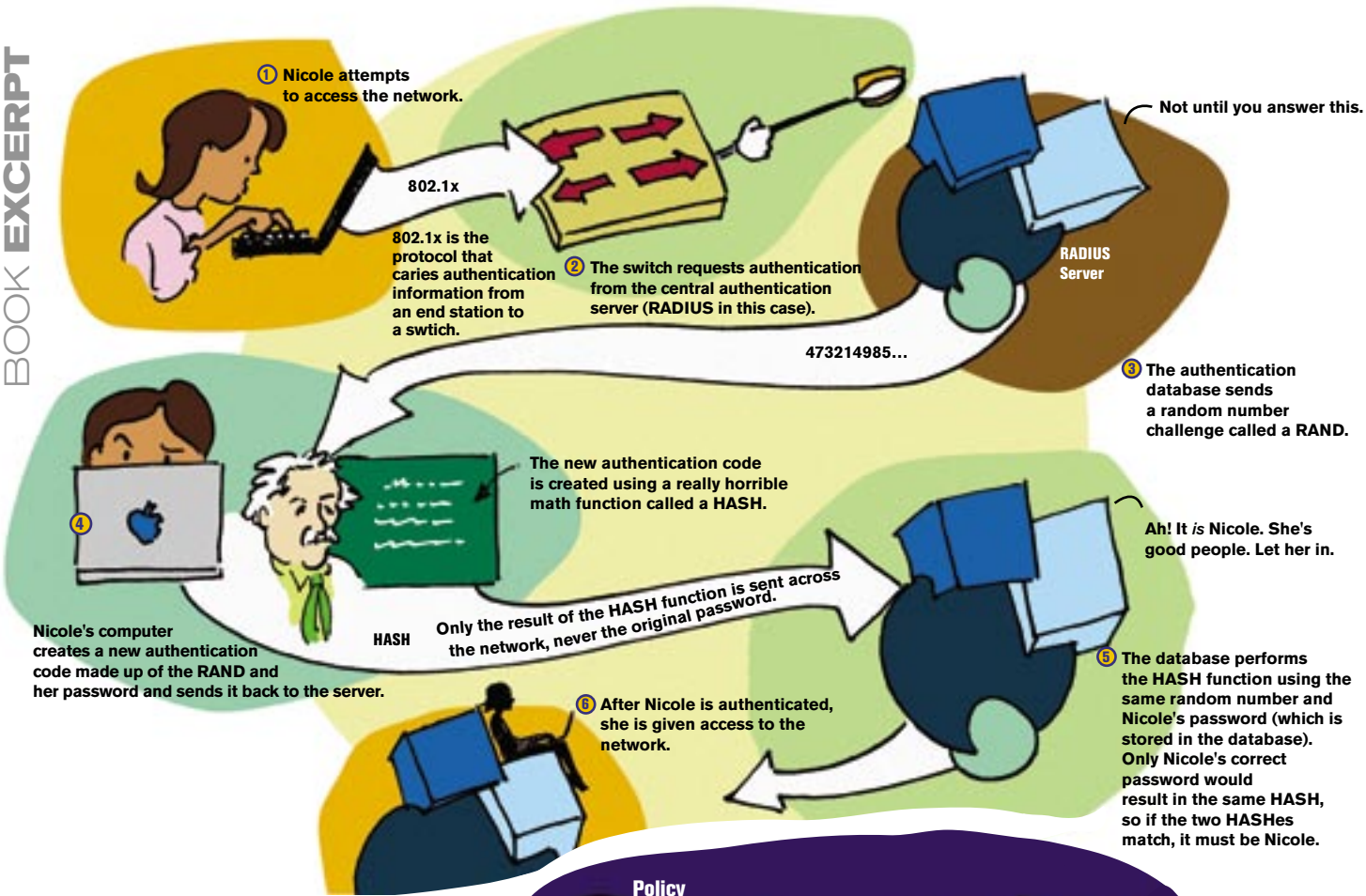
The benefits of hub-and-spoke and IBM's WebSphere Business Integration Server

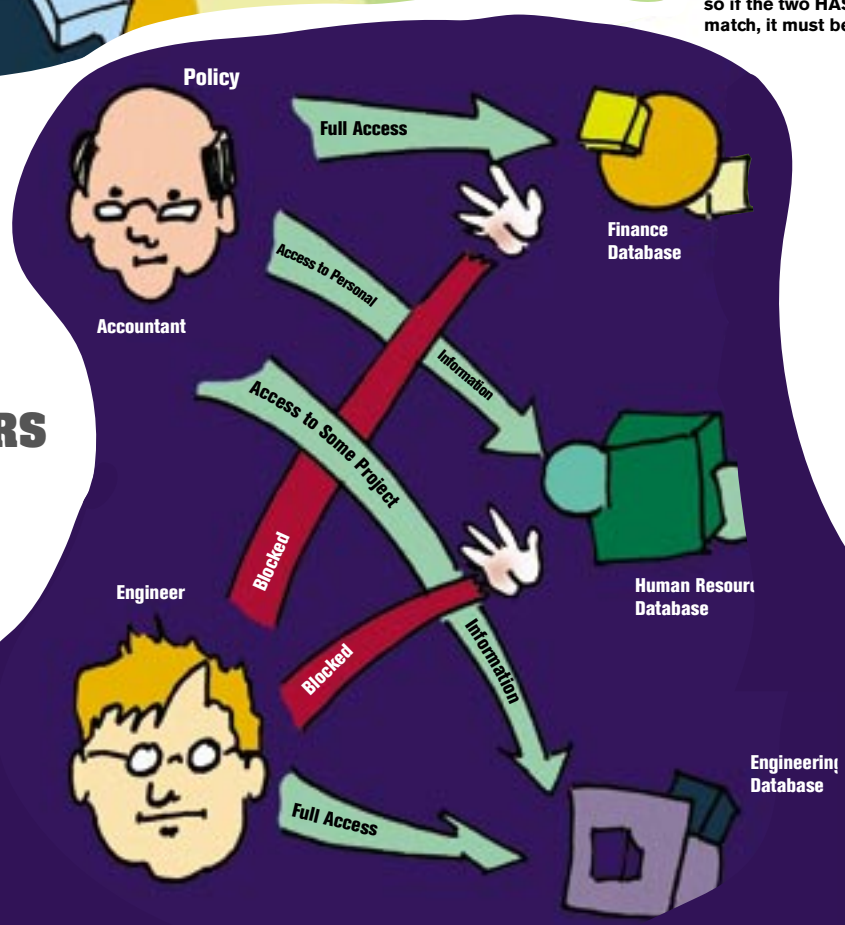
PROPAGATING SECURITY CONTEXT ACROSS A DISTRIBUTED WEB SERVICES ENVIRONMENT

How to accommodate a broad range of established security procedures and legacy technologies

CARE FOR THE CAREGIVERS

Four solutions that cut costs without losing TLC



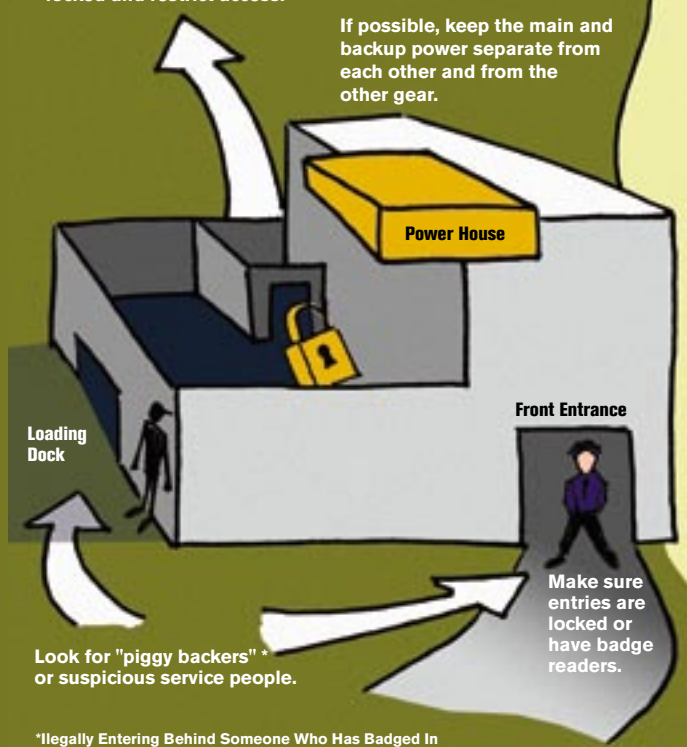


KEEP OUTSIDERS OUT. KEEP INSIDERS HONEST.

PHYSICAL SECURITY

Keep gear and wiring closets locked and restrict access.

If possible, keep the main and backup power separate from each other and from the other gear.



Make sure entries are locked or have badge readers.

Look for "piggy backers" * or suspicious service people.

*Illegally Entering Behind Someone Who Has Badged In

SOFTWARE SECURITY
ENCOURAGE STRONG PASSWORDS

Good

Username: DOHERTY

Password: TZg3B47

Time to Crack: Six Months
(Password Changed Every Four Months)

Strong Passwords = Mixed Upper/Lower Case with Numbers

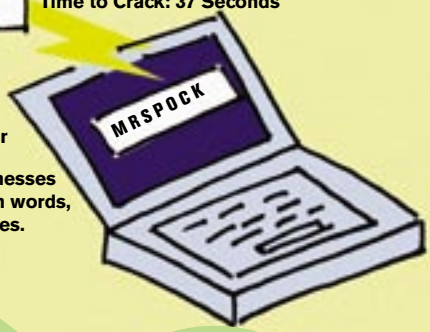
A six-digit password using the above formula gives 9.54×10^{22} possible combinations.

Not Good

Username: DOHERTY

Password: MRSPOCK

Time to Crack: 37 Seconds



Hacker PC with Password Cracker

Looks for weaknesses such as common words, themes, and dates.

AVOID ACTS OF GROSS STUPIDITY

Business rules system as a plug-in for WSAD

Integrated IDE for Applications with Complex Business Logic

BY CAROLINE BUCK

Caroline Buck has spoken at various academic events and CeBIT on topics concerning information distribution and business rules. After gaining seven years of application development experience in the industry and service sector, Caroline is now responsible for bringing visual rules onto the market at Innovations Softwaretechnologie GmbH.
presse@visual-rules.de

This article describes how the IDE can be utilized as a common platform for both developers and business experts, how the development process is accelerated, and how costs are reduced. The integrated IDE supports free choice of architecture for using business logic.

WebSphere Application Developer (WSAD) is – like Eclipse and other IDEs based on Eclipse – a platform that can be complemented with different tools and customized to meet individual needs. A business rules system has been presented by a German software engineering company that is the first of its kind as a plug-in for these platforms. With WSAD, it presents an integrated IDE for the development of applications with easy-to-change business logic.

As a plug-in, the business rules system uses the functionalities of WSAD and complements the platform with a unique method of graphical development of business logic and automatic rule code generation of the models.

Accelerating the Process Using the Graphical Approach

Different concepts of software engineering target integrating the knowledge of business experts into the application development process. UML is one of them. Relations between classes and objects are modeled graphically to control data flow and communication between objects.

The graphical modeling approach is used by the business rules system for the development of business logic. The whole modeling process – structuring rule flow as well as defining rules as different decisions – is covered with the graphical method. This makes results extremely traceable and easy to actualize and debug. The graphical design of the IDE allows for efficient integration of the business expert into the development process. The complete model represents the precise definition of the business logic. The developer then takes this model and automatically generates rule code for it. Based on simplified coordination between business and IT experts, high-quality development and actualization of business logic is accelerated.

Platform for Developers and Business Experts

The business user has a simple view of the IDE that solely comprises the graphical rule editor. This editor also displays statistics of simulations and tests that are important for the expert to debug the logic from his business view. Modeling the rules in their context, he or she has direct access to the data objects of the application and is supported by dialogues. It is not necessary to define an extra business language.

Figure 1 shows a graphical editor for modeling business logic, in this case for calculating prices of cinema tickets depending on the hall where the film is shown and the seat that has been chosen. This calculation example is extended by a second tree (node at the bottom of the model) that integrates discounts for students and coupons. This ability to extend rule trees is provided for reasons of clarity accompanying complexity.

The graph in Figure 2 shows a similar interface for the developer. It has different views of the business logic (for navigation, for project settings, statistics, etc.), using the previous example after simulation of single data processing, with statistics at the nodes and highlighting of the path taken in the rule tree.

All functions of the integrated IDE are available for the developer's work – for automatic code generation out of the rule models, for integration of rule code into applications, for deployment of the business logic on different target platforms, for debugging and monitoring of operational logic, as well as for versioning of rule projects. Versioning is accomplished with mechanisms that are provided by WSAD.

Integrated in WSAD, the rules system features full function for the development process of business logic. Both developers and business experts use the graphical component for different tasks. After the simulation of rule processing, statistics are shown at each node of the

rule tree. Statistics are also available for operational business logic as well as for tests. Furthermore, the paths that are taken by rule processing are highlighted. The graphical design simplifies actualization and debugging of rules.

With WSAD complemented by the plug-in for developing business logic, there is now a platform on the market used by developers and by business experts alike. The development of applications is accelerated, both by the simplified coordination and by automatic generation of code out of business logic models (Rapid Application Development).

High-Performing Code

By structuring business rules explicitly in hierarchical context, no rule engine is needed to interpret the rules for execution (RETE-algorithm). Instead, the best option for a path is always as defined taken in the rule tree. Rules are processed extraordinarily fast because only those that apply are passed. The graphical modeling of business logic is best qualified for applications with great numbers of rules and metrics.

Free Choice of IT Architecture for Using the Business Logic

The business rules system is the only one of its kind that generates either Java or COBOL code out of the business logic models. This code can be integrated directly into applications (e.g., as a JAR file), mainframes, and client/server architectures. It can be used in central systems – from legacy systems to mobile terminals. This is important for all companies with organically grown IT infrastructures and becomes even more important when migrating systems.

The business logic can also be deployed on different application servers. All servers that conform to the J2EE standards are supported. This includes IBM WebSphere as well as other established application servers. For deployment, the business logic is made available outside of the IDE as an EJB or Web service.

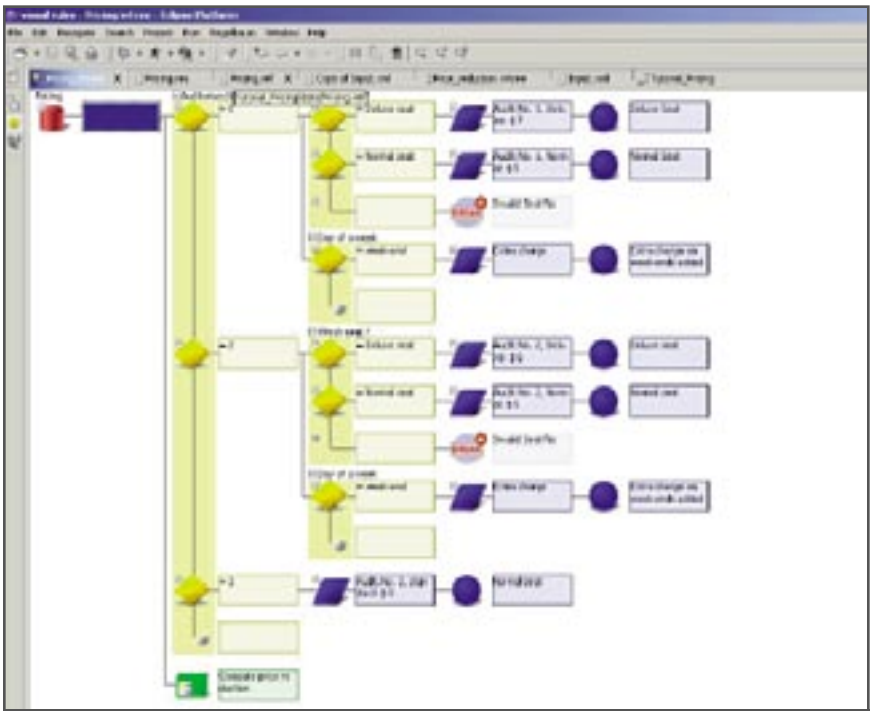


FIG 1: THE INTERFACE FOR THE BUSINESS EXPERT

Enterprise-wide use of business logic is supported technologically. It is also supported with a special licensing model. No extra run-time costs are generated in using it.

WSAD Functions Used by the Plug-In

With the integration of the business rules system into WSAD, the developer continues working with his familiar environment. All steps of the development process can be accomplished directly out of the IDE. Furthermore, the IDE can be used for a variety of functions. An example of this is the resource repository. It can be used not only for versioning the application code but also for the business logic with all project files. This function is completed by the business rules system with the graphical matching of different versions of rule trees.

Conclusion

The market is up-and-coming; business analysts are forecasting a growing market, in both the short and long term, for rule engines. With the full integration of visual rules into WSAD, the graphical development of business logic becomes available to all WebSphere users. Visual rules provide flexible and transparent business logic to create WebSphere-based decision automation systems concerning various business cases. These systems enable companies to react fast to changing market conditions – a crucial success factor of all businesses now and in the future.

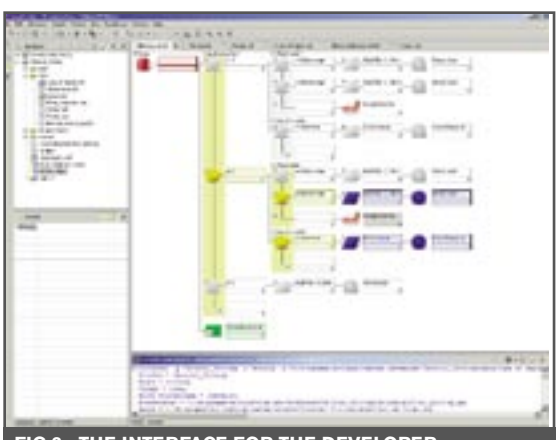


FIG 2: THE INTERFACE FOR THE DEVELOPER

OpenDemand Unveils Web Performance Test Tool

(Newark, NJ) – OpenDemand Systems has released OpenLoad v4.0, which is a browser-based load and functional testing solution to ensure the performance, availability, and reliability of enterprise Web applications and services. It has been developed to address a need in the market for simple, powerful, and affordable automated test tools. OpenDemand v4.0 offers midmarket firms an alternative to complex legacy load testing tools, giving them the ability to perform productive testing of dynamic Web sites without having to develop complex scripts or perform complicated data analysis.

OpenLoad v4.0 automates the scripting process by inspecting the contents of each response and automatically parameterizing the appropriate session variables on the user's behalf. This approach helps to eliminate the need for complex scripting and debugging, allowing users to quickly build robust data-driven user scenarios. The "script-free" recorder offers out-of-the-box support for PeopleSoft applications using versions 8.0 and later, enabling PeopleSoft implementers and administrators to easily test customized modules.

This offering provides full support for functional and performance testing of Web services. OpenLoad's Web service transaction recorder allows users to capture SOAP requests without a GUI and offers a browser-based interface that enables both technical and non-technical users to manipulate XML data elements.
www.opendemand.com

dynaSight Analytics Approved for WebSphere Platform

(Wayne, PA) – arcplan, a business intelligence (BI) software provider and IBM Advanced Partner, received approval of its dynaSight BI platform for use with the WebSphere middleware product environment.

The dynaSight software platform enables the quick creation of custom analytic applications that integrate with various data sources to present business data in a manner that is easily understood – including graphics, tables, and dashboards. Its integration within the WebSphere portal allows enterprises to combine disparate data sources and present the information in a consistent, Web-based, dynamic interface.

According to Jim Burke, arcplan's vice president of business development, "arcplan's dynaSight provides the rapid application development and direct interaction with the WebSphere components required to

IBM SIMPLIFIES SOA SECURITY AND MANAGEMENT

(Austin) TX) – IBM has announced that four new business partners have enhanced their solutions to support IBM's identity management software, bringing additional security and management capabilities to clients who deploy service-oriented architectures (SOAs). IBM is working with XML firewall vendors to ensure their products work with IBM Tivoli Access Manager to enable customers to simplify security and identity management of Web services in SOA deployments.

Customers had requested a centralized security management solution for securing their SOA to help integrate their systems with partners, suppliers, and their own customers. Clients now have the options and flexibility to integrate security management into their Web services solutions, thereby reducing administration costs, accelerate time to market, and increase operational efficiencies in SOA deployments.

Developers and information technology (IT) staff can incorporate security as another component in SOA, allowing them more time to focus on creating reusable business components that can be deployed quickly to respond to changing market conditions.

The products that interoperate with Tivoli Access Manager include:

- Digital Evolution Service Manager: an XML and Web services management and security solution that implements policy enforcement and comprehensive SOA enablement
- Reactivity XML Firewall: a Web service security appliance that helps enterprises deploy secure Web services
- VordelSecure/VordelDirector: an XML gateway and XML security server that enables organizations to deploy and configure security, both inside and outside the organization
- Layer7 SecureSpan: a standards-based solution for coordinating security and integration policy across a Web services transaction

www.ibm.com/ondemand

WILY TECHNOLOGY RELEASES J2EE APPLICATION PERFORMANCE SERVICES

(Brisbane, CA) – Wily Technology, an enterprise management company, has announced the availability of a J2EE Performance Consulting Practice to aid customers and partners in establishing best practices for managing and monitoring Java environments. The J2EE Performance Practice includes a suite of technical services, educational offerings, and business partner programs that have been designed to complement its application management software solutions.

The foundation of the J2EE Performance Practice offerings is Interscope 5.0, the enterprise application management solution, which has been designed to ensure the performance and availability of mission-critical applications and supporting systems. J2EE Performance Practice is intended to help customers and partners understand the roles and responsibilities required for day-to-day operations of Java-based applications, as well as how to define and set best practices for managing such applications.

The professional services offerings are designed to transfer critical technical and process skills to customers. The new offerings include:

- Web Application Management Assessment: This is a service which evaluates current performance management practices to identify opportunities for improvement and deliver findings and recommendations leading to a better control of Java Web environments.
- Java Performance Management Deployment Services: This helps customers develop a comprehensive deployment plan, install and configure products and other relevant solutions, implement monitoring and reporting procedures, develop and document deployment processes, integrate with existing systems management frameworks, develop and execute a rollout plan across life-cycle phases and develop customer reports.
- J2EE Monitoring Business Consultants (JMBC): Consultants will help customers and partners understand best practices and benefit from their expertise.

www.wilytech.com

deliver on-demand business performance management solutions."
www.arcplan.com

IBM Achieves Top UK Innovation Prize

(Hursley, UK) – IBM has received a UK engineering award for WebSphere MQ, its middleware breed of software. The Royal Academy of Engineering's MacRobert prize rewards technological and engineering innovations.

Used by global banks, WebSphere MQ has helped to transform e-commerce by enabling data transfers across computer systems without the need for custom coding. Banks that provide Internet services use the software to ensure that transactions occur without mistakes, even when the computers go offline.

With the software, vital information can be swapped between computer systems, regardless of location and whatever hardware, programming language, or operating system is being used.

The software was developed in 1994 to allow applications on any of over 40 separate platforms to communicate and handle data transfer easily. In the past, systems were connected through custom coding.

Increased End-to-End Transactional Visibility for CICS Applications

(Danbury, CT) – Bristol Technology Inc., a provider of transaction tracking and monitoring solutions, has announced an enhanced CICS TransactionVision Sensor that increases end-to-end transactional visibility for business transactions. It allows customers to track business transactions including stock trades, funds transfers, and insurance claims, through existing CICS applications. With this capability, users can track transactions across CICS, Web services, J2EE, LMS, and WebSphere MQ.

The expanded TransactionVision CICS Sensor allows customers to automatically and non-intrusively discover transactional flows by correlating asynchronous parallel events from all of the applications in the business process.
www.bristol.com

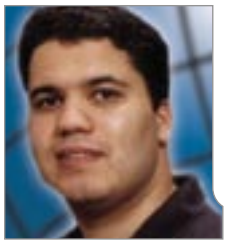
Axeda Extends Support for IBM Products

(Mansfield, MA) – Axeda Systems Inc., a provider of device relationship management software and services, has announced expanded support for IBM products in the release of the Axeda DRM system. Device and instrument organizations can leverage IBM technologies, including eServer xSeries running Linux, WebSphere Applications Server, and Tivoli Directory Server to deploy their remote service solutions.

The Axeda DRM system allows service and support personnel to remotely diagnose, repair, monitor usage, and manage software of deployed devices. It is a distributed software system that securely transmits status and usage data via the Internet from remote equipment to service representatives, notifying them if a device is operating outside preset performance standards. In an effort to increase customer satisfaction, it has been designed to reduce the number of support calls, unnecessary service visits, and improve device uptime.
www.axeda.com

Dice Check

BY MICHAEL SICK



I recently completed my first WebSphere Portal v5 install/implementation for a moderately sized healthcare company. Typically, when I consider new assignments and new technologies, I perform the "Dice Check". The Dice Check consists of some quick and dirty research using dice.com's job posting database to determine the vitality of the software stack with which I might be working. After completing this portal work, I shared some of my experiences with one of Sun's principal engineers, who suggested I take a look at the Sun ONE Portal Server. After reviewing Sun's portal server, I found it to be technically solid, but the product did not seem to pass the "Dice Check".

The Method

Dice Checking is both a fuzzy and flawed research tool, but I still believe that it is also informative and valid. Its purpose is to give a general snapshot of the consulting market for a particular software skill or set of skills. Dice's interface is insufficient for establishing trends, with the exception of recording individual results and mapping them over time. For a quick data point, the Dice Check gives a fairly reasonable picture of available opportunities.

Why Dice?

Dice specializes in contract jobs and I'm a contractor. I also use Dice because I believe that the contract market can serve as a bellwether for the full-time employment market. Sites like Monster, HotJobs, or Career Builder may be better targets for the full-time market. Job seekers looking for only cutting-edge technologies may not even consider the Dice Check valid because a technology may be in use by startups and advanced technology groups for some time before it registers on various job sites.

Searches

Searches for multiple software technologies using Dice's Boolean search features can be complex. In this case, the searches were fairly simple and consisted of the following.

1. ***WebSphere Portal:*** "websphere portal"
2. ***Sun ONE Portal:*** "sunone portal" or "sun one portal"
3. ***Java System Portal:*** "sun java system portal server" or "sun java system portal server"
4. ***WebLogic Portal:*** "weblogic portal"

The Results

Each search was performed on June 1, 2004 and seemed generally consistent with ad hoc searches performed in the previous week.

TITLE	HITS
1 WebSphere Portal	80
2 Sun ONE Portal	8
3 Java System Portal	0
4 WebLogic Portal	61

The first interesting number was that Sun's re-branding of its Sun ONE Portal to Java System Portal has not taken root in the job market. As a result, there are zero postings associated with that product name. Additionally, while I found the Sun One Portal technically sound and easy to work with, it seems clear that the contractor market for WebSphere Portal offers more opportunities than the Sun ONE Portal. When investing in my technical skills, it's difficult to justify spending significant amounts of time learning about a product that offers one-tenth of the contract opportunities.

The surprising number for me was there were fewer hits for WebLogic Portal than for WebSphere Portal. Until this Dice Check, I had assumed that that WebLogic's Portal Server offered more opportunities than the WebSphere Portal Server. I had also expected WebSphere Portal's numbers to be underrepresented because contractors for IBM Global Services would take many of the opportunities.

Dice Checking doesn't keep me from downloading, learning, and/or using new and interesting technologies. However, I do let it guide how I invest a portion of my professional development. It informs me when considering new contracts, professional certification, and continuing education, and will remain a part of how I inform myself about contract opportunities.

Michael Sick is an independent Java architect helping clients solve complex product definition, design and implementation problems. He has over nine years of experience in the construction of distributed information systems and Internet technology. Mike is a contributing editor for ***Web Services Journal*** and a member of their editorial board. msick@sys-con.com

Jumping 2 Enormous Eggplants.

A new way to look at J2EE.

Cyanea/One gives you unprecedented visibility into mission-critical application performance. With the transaction-based performance management capabilities of Cyanea/One, you can drill down to the method level to determine and resolve performance issues. Cyanea/One monitors complex, multi-platform, service-oriented environments 24 x 7 from a single web-based console, all without having to touch the application code. If you are developing enterprise applications, you know the hurdles you sometimes have to overcome; with Cyanea/One, your applications will reach new heights.

For more information, visit www.cyanea.com/j2ee or call 1.877.CYANEA8.

Cyanea®