SYS-CON

STANDARDS • MODELING PROCESS• PORTALS

WebSphere® JOURNAL

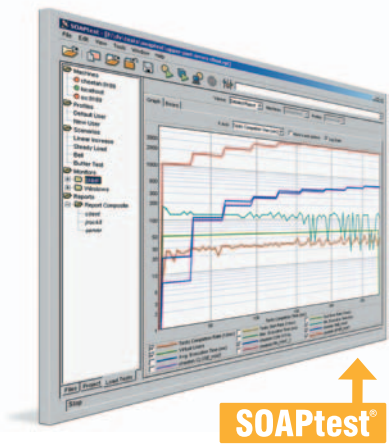
www.SYS-CON.com/WebSphere

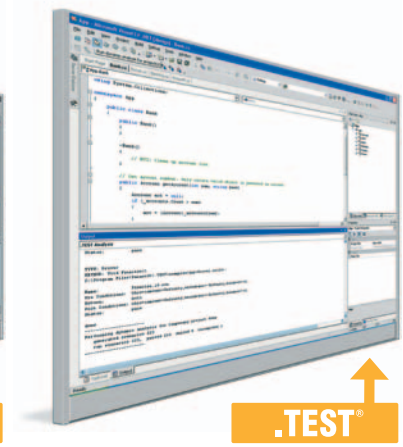
The World's Leading Independent WebSphere Resource

APRIL 2005 VOLUME 4 ISSUE 4

Deliver better, safer software

in less time, with fewer resources.





Parasoft® Automated Unit Testing & Coding Standards Analysis: specifically designed for under-staffed, over-committed development organizations like yours.

Parasoft products ensure that your entire development organization is meeting requirements and adhering to a common standard. Parasoft products automatically identify security violations, memory errors, runtime exceptions, common defect patterns, and violations of best coding practices.

Increase development effectiveness by allowing:

- **Developers** to correct problems early and reduce time spent chasing bugs
- **Architects** to ensure that development is adhering to a common standard
- **Managers** to improve project visibility, decision making and predictability

The Parasoft product family supports all your major development environments including Java, Web Services, C/C++, .NET and more.

For details go to: www.parasoft.com/Products

Email: Products@parasoft.com Call: 888-305-0041 x3307

PARASOFT®
We make software work.™

Founded in 1987, Parasoft's clients include IBM, HP, DaimlerChrysler and over 10,000 companies worldwide.

Open Up!

Portals–Get what you want easily

Ask the Experts

Prolifics is 1 of only 3 WebSphere Service Providers retained by IBM. Open up with portals! With 26 years of industry expertise, Prolifics' certified WebSphere portal experts deliver a personalized, secure, unified view to your enterprise assets.

Our portal solutions build agility into your business — enabling you to reach new markets, achieve customer satisfaction beyond expectation, and ease collaboration within your enterprise. Our teams of specialists work together with your business experts from gathering the business requirements, translating them to a technical specification, to delivering the final production application.

Portal Executive Assessment

Prolifics experts help you articulate the value of portals to your business sponsors and perform a quick business review identifying portal benefits as it applies to your organization.

Portal Workshops and Proof of Concepts

Prolifics demonstrates valuable portal features and translates the breadth of your portal objectives into a step-by-step roadmap to meet your current and future needs.

Portal Jump Start

Prolifics' consultants work with you from the start to install, configure and jump start your team on using WebSphere Portal Software.

Portal Development and Mentoring

Prolifics' innovative approach to building portals drives down your overall cost and transforms the way you conduct business. We mentor and arm your team with the skills to manage, maintain and extend the portal environment.

Prolifics

Winner of 2003 IBM BUSINESS PARTNER LEADERSHIP AWARD

116 John Street, New York, New York 10038
800.675.5419
solutions4websphere@prolifics.com
www.prolifics.com

Prolifics IBM Business Partner

IBM Is Ahead on All the Scorecards

BY ROGER STRUKHOFF

"If it were a fight, they'd stop it." This expression is often employed by sports announcers during a particularly lopsided football or baseball game, and also by market watchers when one business gets a clear upper hand versus its competitors.

Maybe it's time to use it when discussing IBM and BEA. Various SYS-CON publications have covered the IBM-BEA battle over the years, watching as report after report shows Big Blue overcoming an early runner-up position to emerge as the dominant player. The most recent report we've seen shows IBM expanding its number one position in the overall application integration and middleware market, capturing 37.2% of the global market vs. BEA's 7.2%. The third and fourth positions are held by Oracle with 4.4% and Microsoft with 4.3%.

Wow, wasn't that fun? You know, seeing Microsoft as the apparent bottom-feeder, with Oracle barely ahead, and BEA's pulse slowing so dramatically?

Well, another truism – one that serves as well for sports as for business and life – is "don't get cocky." And don't start to believe all those positive reviews about yourself.

The management at IBM, despite having to report a disappointing first quarter 2005, thereby sending stock-market investors to the exits for a few days, shows no signs of letting up as it continues to define and re-define the concept and execution of application development in the 21st century. Knowledgeable IT managers will seriously consider WebLogic, for example, or .NET, or the emerging Linux-driven LAMP strategy.

They will also use WebSphere as the benchmark from which to compare everything else. The world of application is insanely complex, with different approaches in different vertical markets, widely disparate IT budgets throughout the spectrum of small, medium and large businesses, and customer demands that range from straightforward to a full-on "customer360" environment 24/7/365. We're happy to report a diverse group of articles in this month's issue of WebSphere Journal that address some of the numerous important topics facing developers. We are featuring the first part of an interview with Doug Wilson, the CTO of IBM Lotus, with a sidebar by WebSphere Portal Senior IT Architect Richard Gornitsky. Readers will be familiar with their work in these pages, including an article they wrote for us in the March 2005 issue. (Look for another installment of the interview and the article next month!)

We have also developed features that discuss the value of modeling, how to move from business modeling to web implementation, the importance of industry standards, and how to add logging to portlets. And as another common expressions states, "But wait! There's more!" Indeed. More content in this issue, continuing coverage of WebSphere in particular and web services in general online at SYS-CON Media, and more success to WebSphere developers and IBM, if, in the words of yet another expression, "the numbers don't lie." For IBM and WebSphere, we believe the numbers speak the truth. We're still in the early rounds, but IBM is clearly winning this fight.

Roger Strukhoff, editor-in-chief of ***WebSphere Journal***, is West Coast Bureau Chief for the SYS-CON News Desk, and President of www.wdva.com. He spent 15 years with Miller Freeman Publications and The International Data Group (IDG), then co-founded CoverOne Media, a custom publishing agency that he sold in 2004. His work has won awards from the American Business Media, Western Press Association, Illinois Press Association, and the Magazine Publishers' Association. Read his blog at http://www.rssblog.linuxworld.com. Contact him at roger@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 ROGER STRUKHOFF
CONTRIBUTING EDITOR PATTI MARTIN
PRODUCT REVIEW EDITOR JAY JOHNSON
EDITOR NANCY VALENTINE
ASSOCIATE EDITOR SETA PAPAZIAN

WRITERS IN THIS ISSUE
Gary Cernosek, Chris Lockhart, Minesh Manilal, Tilak Mitra, Brian Sennett, Fred Simmons, Roger Strukhoff, Jeng Yoong Tan

SUBSCRIPTIONS
For subscriptions and requests for bulk orders, please send your letters to Subscription Department.
888-303-5282 201-802-3012
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
CREDITS & ACCOUNTS RECEIVABLE GAIL NAPLES
ACCOUNTS PAYABLE BETTY WHITE
ADVERTISING DIRECTOR ROBYN FORMA
ADVERTISING MARKETING DIRECTOR DENNIS LEAVEY
ADVERTSING SALES MANAGER MEGAN MUSSA
ADVERTISING SALES MANAGERS KRISTIN KUHNLE, DOROTHY GIL, KIM HUGHES
LEAD DESIGNER ANDREA BODEN
ART DIRECTOR ALEX BOTERO
ASSOCIATE ART DIRECTORS RICHARD SILVERBERG, TAMI BEATTY, LOUIS F. CUFFARI
WEB DESIGNERS STEPHEN KILMURRAY, PERCY YIP
ONLINE EDITOR MARTIN WEZDECKI
CIRCULATION SERVICE COORDINATORS EDNA EARLE RUSSELL, LINDA LIPTON, MONIQUE FLOYD
CUSTOMER RELATIONS MANAGER BRUNI STAROPOLI

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

NEWSSTAND DISTRIBUTION CONSULTANT:
GREGORY ASSOCIATES, W.R.D.S.
732 607-9941, BJGASSOCIATES@CS.COM

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

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

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

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

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

Get Wily.™

1 888 GET WILY | wilytech.com

wily technology

From Business Modeling to Web Implementation

Part 1 - Modeling business process

BY TILAK MITRA

Tilak Mitra is a certified senior IT architect in IBM. He specializes in mid- to large-range enterprise and application architectures based on J2EE, MQ and other EAI technologies. tmitra@us.ibm.com.

First published on developerWorks WebSphere at http://www.ibm.com/developerWorks

A software development lifecycle begins by trying to understand the system you want to build. One of the de facto ways to understand a system is to model the business processes that collectively constitute its functionality. Each business process is an effort to streamline an enterprise's operations and functions. System architects and designers then use these business process models to come up with a system architecture and implementation that realizes the business processes.

Business process analysts have used various methods to capture business process models. Until recently there was a gap between business process modeling and system architecture, design, and implementation. A standardized mechanism/language had to be devised that captured the artifacts created during modeling so they could be used more handily in system design and implementation. Enter the Business Process Execution Language (BPEL).

With BPEL on the scene, companies started looking for novel ways to tie BPEL output to the definition of skeleton code artifacts that would form the basis of an application's interfaces and implementation.

Among the rich set of products in IBM's Business Integration Suite, the WebSphere Business Integration Modeler (WBI Modeler) and WebSphere Studio Application Developer Integration Edition (WSADIE) aim at tying together business process modeling and system implementation by supporting BPEL. WBI Modeler supports process modeling in BPEL mode; WSADIE has a feature called Process Choreography that uses BPEL to choreograph business processes and create Web Services definitions (WSDL) for BPEL-modeled processes. These Web Service definitions can then be implemented with J2EE technology.

In this two-part article, we'll use a sample scenario in which a simple business process is modeled using WBI Modeler. Part II will show how WSADIE uses WBI Modeler artifacts to define Web Service definitions. I hope to help the reader understand how to implement a business process using J2EE technologies that can be invoked by any service consumer. I'll cover how to create and tie business process modeling artifacts to the definition of a system's services and subsequently its implementations. I'll expose the reader to the WBI Modeler tool and how to create not-too-complex business process workflow. I'll then use the WSADIE tool to show how to create a Web Service definition for the business process created by the WBI Modeler and subsequently how to invoke it, all using J2EE technology.

Install Prerequisites

For business process modeling, I used WBI Modeler 5.1. WBI Modeler comes as both a standalone product and as a plug-in for WSADIE. The look-and-feel of both is exactly the same. Here I used the plug-in so there would be only one toolset (WSADIE), not two, to work with. To implement the business process, I used WSADIE 5.1. One needs a computer with a minimum of 1GB RAM. I used a 2GB RAM machine.

Sample Scenario

Let's sketch a scenario as the basis of our problem statement and its proposed solution. CricketGear, Inc., is a small business in New York that sells equipment for the game of cricket. In an effort to make the

game popular in the United States, it wants to create a Web presence so aficionados could surf its Web site and place orders. To start small, it decides to start by selling cricket bats and balls. A Web order business process is envisioned that will let Web users submit orders. The order will be priced when the buyer's credit card is validated. Based on the validation, a payment transaction will be processed. The priced order will be saved and an acknowledgement sent as the result/output of the process.

A business process implements a set of business rules. An example of the business rule here would be: A payment transaction is based on the validity of the credit card. If the credit card is invalid, no transaction will be initiated and the status of the order may be updated to reflect that. There may be many other business processes (e.g., Ship Order, Process Back Order) that may be modeled and subsequently implemented. For the sake of this article though, we're going to concentrate on modeling and implementing a single business process.

Modeling the Process: The Initial Tasks

There are various steps involved in modeling a business process using the WBI Modeler plug-in (to WSADIE). The high-level steps start by creating a business modeling project, creating the business items, creating the process model (which involves identifying individual tasks and defining their inputs and outputs) and creating the process flow by connecting the tasks together (as sequential or parallel steps). To illustrate:

- Create a new WSADIE workspace and then open the *Business Modeling Perspective* in WSADIE.
- Create a new business model project with the attributes shown in Figure 1. A Process Catalog is a folder that contains all the processes that can be created. Each process has a name (e.g., WebOrder).
- Select the BPEL mode by clicking Modeling->Mode->BPEL mode from the main menu. This lets the process be modeled only with BPEL artifacts so the final output is BPEL-compliant. Also select Modeling->User Profile->Basic from the main menu. (See Figures 2 and 3.)

Business items need to be created. Our business items are Order and CreditCard. There's another business item we're going to create called RequestData. This needs an explanation. A task in a process modeled with WBI Modeler can have more than one output. However, if a task is implemented in a language like Java (as an interface method of a class), the multiple outputs need to be mapped to a complex data object that contains references to the multiple output data types. The process we're going to model has a task with multiple outputs. So we create a RequestData business item whose contents is a reference to the Order and CreditCard business items. This way we can simulate a single output.

- Click on the *Business Item* folder. Right-click New->Business Item. Create the Order business item with the attributes shown in Figure 4. The attribute type can be defined by double-clicking the attributes row then clicking the button on the extreme right of the row that lists the attribute types. (Refer to the figure for the attributes for RequestData.) The attribute types may be Basic (e.g., Integer, or String) or Complex DataTypes (e.g., user-defined types like Order, CreditCard)
- Create the CreditCard business item with the attributes shown in Figure 5.

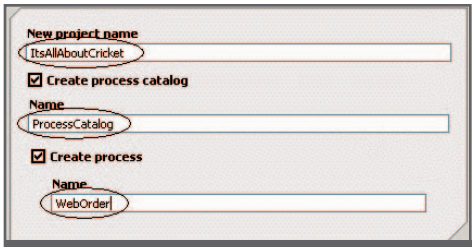


FIG 1: NEW BUSINESS MODELING PROJECT

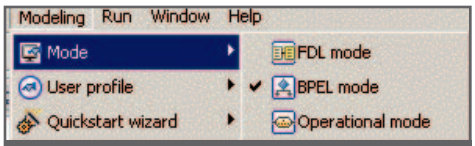


FIG 2: SETTING THE MODELING MODE BPEL

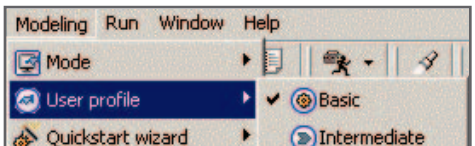


FIG 3: SETTING THE USER PROFILE TO BASIC

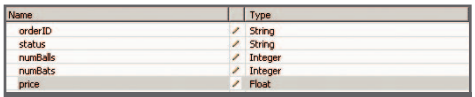


FIG 4: ATTRIBUTES OF THE ORDER BUSINESS ITEM

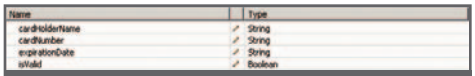


FIG 5: ATTRIBUTES OF THE CREDITCARD BUSINESS ITEM

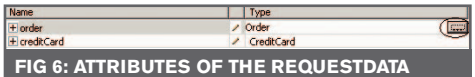


FIG 6: ATTRIBUTES OF THE REQUESTDATA BUSINESS ITEM

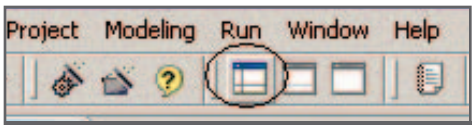


FIG 7: SWITCHING TO THE FOUR-PANE LAYOUT

"There used to be a gap between business process modeling and system architecture, design, and implementation until they invented BPEL"

- Create the RequestData business item with the attributes shown in Figure 6.
- To get a better layout of the perspective, switch to the four-pane layout shown in Figure 7.

Now we have to create the individual tasks that constitute our WebOrder process. In the BPEL editor for any new process (e.g. the WebOrder process) there's a *Start* (in green) and a *Stop* (in red and black) node. Delete these two nodes before drawing the model.

Drawing the Model

With the project created and the business items defined, the process model needs to be drawn with its individual tasks, connections, and data input/output types. Use the Task icon from the palette in the BPEL editor to create the individual tasks. The icon immediately below the Task icon ( ) is that of a Tower depicting a notification broadcaster. Notice the small arrowhead ( ) on the icon. Click that for a dropdown list of icons. The only icon that should be active is the Map. Use this to create a Map as part of our process.

We'll create the model in three stages. The first stage is to identify all the individual tasks and any other BPEL artifacts (e.g., the Map) that may be used. The second stage is to connect the tasks (and other BPEL artifacts) together to determine the workflow of the process. Once this stage is complete, the third stage is to determine and attach the data inputs and outputs to each of the tasks (and other BPEL artifacts). The three stages are illustrated below.

Figure 8 depicts the tasks that constitute the WebOrder process. We also used a Map element. Task output may be a complex datatype. In situations where a task requires input from a complex datatype, a Map can be used to extract only the data required from the complex datatype (e.g., RequestData).

The *Submit Order* accepts the process input. The *Price Order* task prices the order and sets the Order's price attribute. The *Transact Payment* task checks the validity of the customer's credit card while the *Save Order* task saves the order and returns the result of the saving process as the output of the process

We now connect the model artifacts together in a process flow. Figure 9 depicts how the process workflows through its individual tasks. The BPEL editor has a black-bordered boundary. Any connection from the left-vertical border denotes an input to the process while a connection that ends in the right-vertical border denotes the output result of the process.

Note that the input and output connections have an

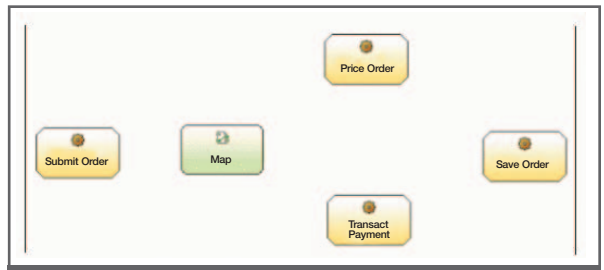


FIG 8: INDIVIDUAL TASKS FOR THE WEBORDER PROCESS

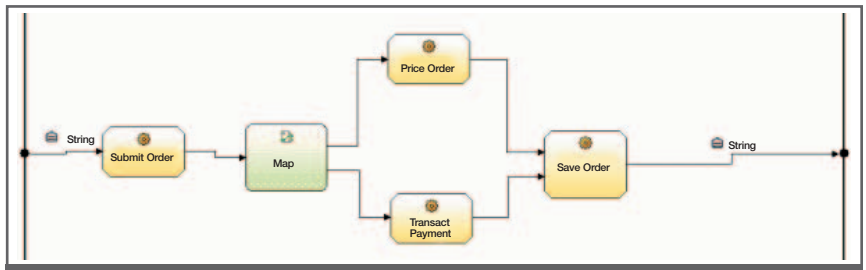


FIG 9: WEBORDER PROCESS FLOW

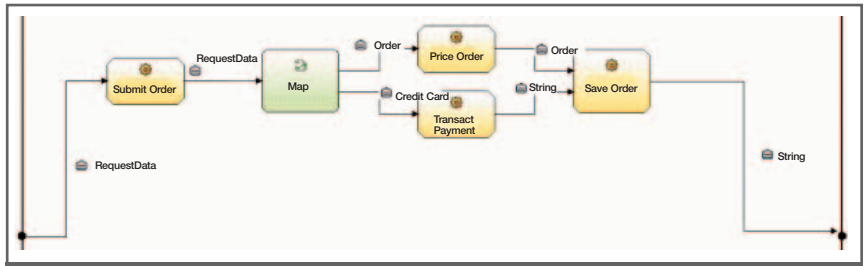


FIG 10: WEBORDER PROCESS MODEL WITH ALL CONNECTIONS, INPUTS, AND OUTPUTS DEFINED

NEW!

CodePro AnalytiX™

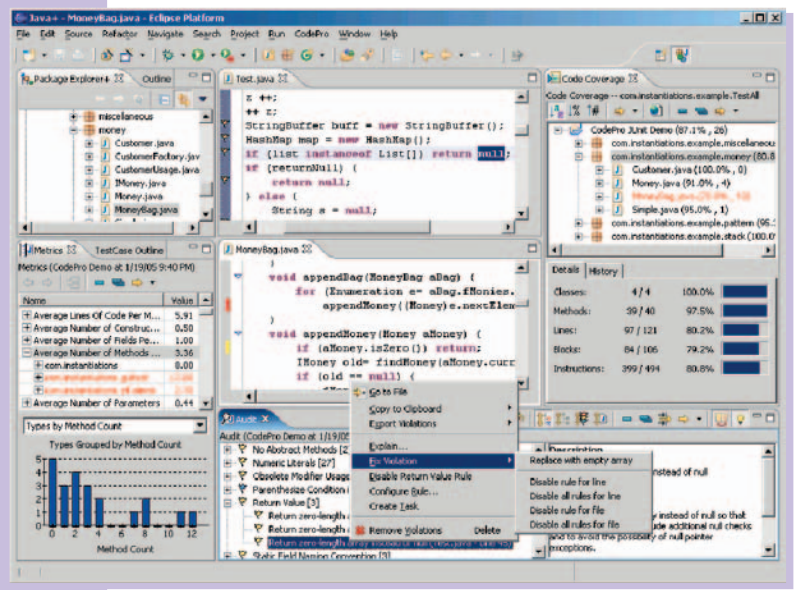
Continuous Collaborative Code Analysis for Rational®, WebSphere® & Eclipse

Imagine your software team creating higher quality, more reliable, Java systems... with less time and money.

Sound like a fantasy?

...*CodePro AnalytiX* lets you do exactly that, or

Your money back.



***AnalytiX* automates the process of building quality into your software as it's developed. Using the latest software development technology *AnalytiX* monitors Java code development in real-time, as its written. It identifies and automatically fixes hundreds of issues that would reduce quality and reliability.**

Powerful wizards generate unit and regression tests and analyze code coverage, helping drive out problems early in the development process.

Less experienced developers code like experts, experts create tight, maintainable systems.

***AnalytiX* also allows you to define, distribute, and enforce corporate coding standards and quality measures across your development teams...no matter where they're located.**

Early identification and correction of code quality issues dramatically improves your application development process and reduces the time and money it takes to develop your high performance Java systems.

From the development tool experts at Instantiations. No one outside of IBM has longer or deeper experience in creating Eclipse-based Java development tools.

Put our experience to work for your team.

www.instantiations.com

1-800-808-3737

Key Product Capabilities

- **Detect and correct code quality issues ...*Automatically***
- **Define, distribute and enforce code quality standards across development teams**
- **700+ rules and metrics/300+ Quick Fixes**
- **Code metrics with drilldown and triggers**
- **Custom rules and rule sets**
- **JUnit test case generation**
- **Code coverage analysis**
- **Dependency analysis and reporting**
- **Integrated team collaboration**
- **Javadoc analysis and repair**
- **Seamless integration with IBM Rational Developer, WebSphere Studio and Eclipse**

Risk Free Trial Copy

www.instantiations.com/analytix

eclipse READY · IBM Business Partner · eclipse FOUNDATION MEMBER

instantiations

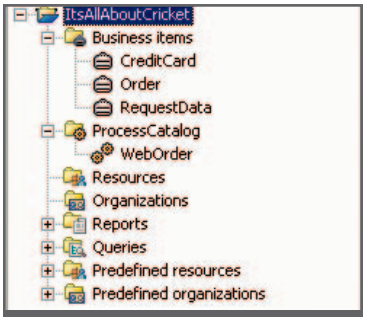


FIG 11: PROJECT TREE REPRESENTING THE ARTIFACTS CREATED FOR THE WEBORDER BUSINESS PROCESS

associated String datatype. This is the default datatype that the editor associates with any input and output of a process connection. It can be changed later to denote any other datatype. You may notice that the input and output connections to the process can't be seen together unless the editor is scrolled horizontally. Scrolling can be minimized by clicking the Decrease Horizontally icon in the editor's icon palette.

With the workflow of our process defined, we now need to associate the proper input and output datatypes to each task. Drag the RequestData business item from the Business Item folder on top of the input connection element of the *Submit Order* task. The appearance of a black dotted-bordered rectangle indicates that the editor has identified the datatype with the connection. Once you see the black dotted-bordered rectangle, you can release the mouse. The RequestData element will then be associated with the input connection. Repeat this step for all the connection elements, with the business items as shown in Figure 10.

The connection between *Transact Payment* and *Save Order* has a String as the associated datatype. For this, click on the connection element and right-click the mouse to get a list of the functions that can be done. Click on the Associate Data menu. It brings up a dialog box from which the datatype can be chosen. Choose String as the Basic Datatype. This will associate the String datatype to the connection.

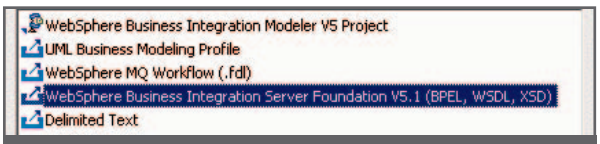


FIG 12: CHOOSING THE WBISF MODE AS THE EXPORT FORMAT

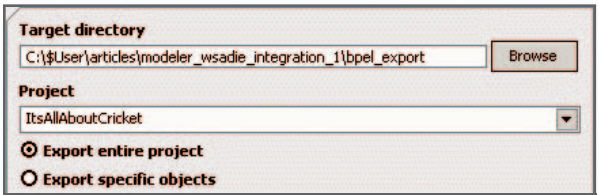


FIG 13: DEFINING A TARGET EXPORT DIRECTORY

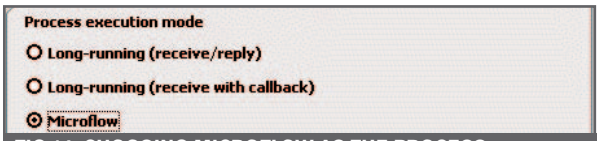


FIG 14: CHOOSING MICROFLOW AS THE PROCESS EXECUTION MODE

Notice how the Map element associates the Order to the *Price Order* task and CreditCard to the *Transact Payment* task. This completes the creation of our WebOrder process. The project tree in Figure 11.

Processes similar to our WebOrder process can be created under the ProcessCatalog folder. Notice how we have kept the process simple. We haven't associated any Resources or Organizations to our process since we're just illustrating the process modeling-to-implementation stage of a scenario.

Getting Ready for Implementation

A BPEL-compliant business process model was created in the previous section. The process artifacts need to be exported in a format that can subsequently be imported into WSADIE. The import process artifacts can then be used to implement the business process.

Right-click on the project name ItsAllAboutCricket in the project tree and choose the Export menu item. This brings up a dialog box with the different modes of export. Choose the mode shown in Figure 12. This makes the exported artifacts compliant with a format that can be imported into WSADIE. Select a target directory where the process artifacts can be exported and check the Overwrite files checkbox (which is not shown in figure 13).

Our process is a short-running business process following the request-reply paradigm. Short-running processes that run in a single transaction (or without a transaction) are known as Microflows. In the subsequent dialog box for the Process Execution Mode, select Microflow as shown in the Figure 14.

Click Finish. An information dialog appears denoting that the project has been successfully exported. This marks the end of our business process modeling. The exported process artifacts can now be used for WSADIE development.

Conclusion

This article took us through a sample business process creation scenario. It showed how to create a process by defining tasks, inputs, and outputs and how they can be linked to a workflow. It also showed how to export the process artifacts created in WBI Modeler in a way that's readily imported into WSADIE to implement business processes. In the next part of this article, we will illustrate how the exported BPEL artifacts can be imported into WSADIE and how Java-based code is generated and modified to include business logic. We will also show how Web Services definitions can be created for the business process and subsequently invoked by a Java client through a Web Service proxy. Stay tuned.

Resources

The WebSphere Business Modeler Portal is the single most comprehensive site to get information about WBI Modeler. The BI Management with WBI Modeler Redbooks provide good coverage on how WBI Modeler's capabilities can be harnessed.

Imagine seeing EAI's impact on business process performance in real-time...

Now imagine implementing it in minutes, *instead of* months

To get your free copy of the informative new article

"3 Steps to Uncomplicating Business Process Management"

go to www.nastel.com/uncomplicate
or call 800-580-2344 or
+631-761-9100

Business happens in real time.
If something impacts your business processes, you need to know. *Now.*

AutoPilot™ from Nastel monitors and manages all of the complex events that comprise your business-critical transactions, giving you deeply granular visibility from end to end for all your business processes. AutoPilot lets you define customized, actionable alerts in the context of each user's specific business function to streamline your business processes and eliminate costly failures, bottlenecks and redundancies. With AutoPilot, you can see the impact of EAI on your business process performance in real time, on-demand.

Unlike other monitoring software, though, AutoPilot's unique architecture takes the complexity out of implementation. It doesn't require months of programmers and consultant's time before you reap the benefits. With AutoPilot, you simply point and click, and within minutes you're in total control.

NASTEL™ TECHNOLOGIES, INC.

Put Your Application Infrastructure On AutoPilot

WebSphere Portal Server 5.1

XMLAccess 101

Part 2: The Blank Portal

BY CHRIS LOCKHART

Chris Lockhart is a senior technical resource at Perficient, a firm recognized for its expertise around IBM technologies. Chris has worked with IBM's WebSphere, Tivoli, and Lotus Software platforms for more than six years. For more information, please visit www.perficient.com.

As a WebSphere AppServer or PortalServer administrator, your world can be a constant learning curve as new applications come along that utilize different aspects of the IBM middleware platforms. You may, for example, have mastered the art of deploying portal configurations using full or partial portal exports. This is convenient and easy to do. However, it may not give you the flexibility required for that new upcoming portal application that is being conceived in the minds of your company's Java developers. Luckily for you, XMLAccess is quite flexible.

Assuming you are familiar with the basic concepts of XMLAccess, the following refinements to the process should be straightforward. Instead of taking a full export from your production environment, suppose you wish to only export the portlets? In XMLAccess, the portlet is represented by the object type "web-app." Naturally you could generate an export of the portlets using the following XML (which is included in the samples shipped with Portal):

```
<?xml version="1.0" encoding=
  "UTF-8"?>
<request
    xmlns:xsi="http://www.
w3.org/2001/XMLSchema-instance"
    xsi:noNamespaceSchemaLocation
      ="PortalConfig_1.3.xsd"
    type="export">
    <portal action="locate">
        <web-app action="export"
          objectid="*"/>
    </portal>
</request>
```

By importing this xml file, we direct the portal to export only the web-app object called '*' which is a wildcard for all web-apps. The command would be as follows:

```
xmlaccess.bat –in C:\temp\export_
  all_portlets.xml –user wpsadmin
  –pwd wpsadminpwd –url http://
  portal.example.com/wps/config
  -out C:\temp\outputfile_all_
  portlets.xml
```

Right, so very easy. However, this sort of thing is useful only if we already have a portal configured with all of the portlets. If we're exporting all "web-app" objects, this assumes we have some to export in the first place. Suppose you have 50 new portlets that all need to be installed, but you have no existing portal that has all of them in place. Are you going to use the portal admin interface and click-click-click your way through installing 50 portlets, assigning permissions, creating pages, putting the portlets on the pages, etc. etc. etc.? Probably not – at least, I know I wouldn't.

How can we use XMLAccess to build a portal configuration if we are deploying portal elements for the very first time? This task won't involve exporting a config from somewhere, since we don't have things configured in any environment yet. Instead we're going to construct our XML files from scratch and import them into a blank portal in order to build our configuration.

The various XML files referred to from this point forward are printed at the end of this article.

The Blank Portal

When installing WPS, it is often desirable to deploy the portal without the sample pages and labels and portlets; it is simply too much clutter. To install a blank portal, use the following command when invoking the installation program:

```
install.bat -W installPortlets
  Sequence.active="false"
```

Proceed through the portal install sequence as you normally would. When complete, start the portal and go the portal home page. You should now see the blank portal.

Before continuing, be sure to have all of your necessary files in place. That is, if you are installing a theme, place the folder containing the theme on the PortalServer in the correct location. The same goes for skins, portlet WARs, etc.

Theme folders go into: C:\WebSphere\AppServer\installedApps\<nodename>\wps.ear\wps.war\themes\html\

Skin folders go into: C:\WebSphere\AppServer\

installedApps\<nodename>\wps.ear\wps.war\skins\html\

By default, portlet WAR files need to be placed in: C:\WebSphere\PortalServer\installableApps

With our blank portal installed, and with our component files in place, let's begin constructing our new portal. We'll start with the skins and themes.

Install Skins and Themes

The XML to install the theme will of course include the standard XML header for WPS 5.1. After that, we will do the required locate action on the "portal" object. After this we will instruct XMLAccess to perform an "update" on our skin with the parameters that we specify as show in Listing 1.

The skin object we specified includes the objectid, uniquename, and resourceroot parameters. Objectid will be the skin's referential id within the portal database. For ease of use, we also have a uniquename specified for this skin so that we can easily refer to the skin from this point forward. The resourceroot indicates the case-sensitive folder name of our skin that we placed in the skins\html\ directory.

Let's save this file as import_skins.xml. Once we import this file, we will have installed the skins into our portal:

```
xmlaccess.bat –in C:\temp\import_
  skins.xml –user wpsadmin –pwd
  wpsadminpwd –url http://portal.
  example.com/wps/config
```

To install our themes, we need an XML file with similar information (see Listing 2).

Since a theme has default and allowed skins assigned to it, we first have to find these skins before we can assign them to the theme. So we do a 'locate' on our two skins that were added in Listing 1. Once we find them, we perform an "update" on the theme object.

The update uses defaultskinref to define the default skin for the theme. For the first theme, we used Skin_A as the default skin. The objectid and uniquename for the theme are defined (and can be different, as indicated). The resourceroot is the case-sensitive folder name of the themes\html\ directory.

Notice that in addition to defining the default skin for each theme, we can also define the other skins that can be used with this theme. Notice that for Theme_Blue we are using Skin_A as the default, but we allow Skin_A as well as Skin_B. This is defined in the <allowed-skin skin="…"> section. Notice that for Theme_Green we are using Skin_B as the default and we only allow that skin. After all, Skin_A is red. Mixing a red skin with a green theme is just plain ugly.

Now let's save the xml file as import_themes.xml and import our themes into the portal configuration:

```
xmlaccess.bat –in C:\temp\import_
  themes.xml –user wpsadmin –pwd
  wpsadminpwd –url http://portal.
  example.com/wps/config
```

Install Portlets

We have our themes and skins imported, now let's put those pesky portlets in place.

Constructing the XML file for portlet install is a little trickier. It involves knowing some key information about the portlets. This information is found in the portlet.xml and web.xml inside the portlet.war file. For this step Winzip is especially useful, as you will need to be able to open the portlet WAR file (see Listing 3). You may also be able to get this info from the portlet developer. After all, he can see this info in his portal toolkit inside WebSphere Studio. However, developers being as busy as they usually are, good luck extracting this info from them.

Okay. This is slightly more complex. Here we have 3 portlets that we're installing. You can see we're locating the portal (of course), and then we do an update on object "web-app" which represents the WAR file.

The next parameter is the web-app uid parameter. The uid must match the uid attribute of portlet-app in portlet.xml in the portlet WAR file.

Following the uid, we have to specify the location of the portlet WAR file. When we transferred our portlet WARs to the server, we placed them in C:\WebSphere\PortalServer\installableApps. To refer to this location, we can use the $server_root$ variable which gets set when the XMLAccess interface is invoked.

Next, we have to perform an update on the portlet-app object. This represents the portlet application and is the parent of all the portlets contained within. The next parameter is the portlet-app uid parameter. The uid must match the uid attribute of concrete-portlet-app in portlet.xml in the portlet WAR file.

Finally we have the update to be performed on each portlet in the portlet application. In this example we only have one portlet in each WAR file, but we could potentially have several in each one. If this were the case, we'd have multiple "portal" objects defined within the "portal-app" object. We give the portlet an objectid and we define the name. The portlet name parameter must match the content of the portlet-name subtag of concrete-portlet in portlet.xml in the portlet WAR file.

So let's go ahead and save the xml file as import_portlets.xml and execute:

```
xmlaccess.bat –in C:\temp\import_
  portlets.xml –user wpsadmin –pwd
  wpsadminpwd –url http://portal.
  example.com/wps/config
```

We should now have deployed skins, themes, and portlets. Let's create a page and drop some portlets on it.

Create Pages

This is very straightforward. We need pages on which to place our portlets. Pages and labels are represented in the XML files as "content-

node." Seems pretty logical if you think about it. A node of content. A content-node. Brilliant.

Our XML will look like Listing 4. Think of it logically. We first locate the resources we need (themes, portlets) and then we create our pages using those resources. Piece of cake. In this example, we'll create a label in the Content Root called Home (Content Root is the highest node of the content hierarchy. It exists by default). Hanging off this label is a single page called Welcome. On this Welcome page is single row containing our portlet, PortletCharlie.

You can see that we start off by locating the necessary resources – first the theme, Theme_Blue and then the portlet, PortletCharlie. Finally we have to locate the Content Root.

The next step in the process is to create the Home label. We give it an objectid and a uniquename. We define the content-parentref to be this label's immediate parent. In our case, this is the objectid of Content Root. Because we're creating a label at this point and not a page, we can assign a theme to the label. So we define themeref to be Theme_Blue.

Once created, we must immediately "locate" the new label before it can be used in the rest of the XML file. Next, we want to create the page called Welcome that is on the label called Home. You can see the values we use to define this page. With a page, however, we can also define the columns and rows containers that compose the structure of a portal page. Within these containers, we place the portlets.

The rows and columns on the page are defined in the "component" section in that page. In this example, within the page definition for the content-node Welcome, we define a component with an orientation of "H," which stands for horizontal, which is therefore a row. Setting the orientation to "V" would of course create a column. Rows can contain columns and vice versa. Tweaking these settings is how you could create an incredibly complex nested row/column page structure, if you so chose.

There are other parameters you can tinker with. Changing "ordinal" will change the order in which this content-node appears (since we only have one content-node at this level, it doesn't matter, but if we had a whole row of pages on the toolbar, we could order them using the ordinal parameter). Setting the "active" parameter to false would de-activate this content-node; and on and on…

```
<component action="update"
  active="true"
objectid="container1"
  ordinal="-1"
  orientation="H"
type="container" width="300">
<portletinstance
  action="update" handle=""
  objectid="portletinstance1"
  portletref="Portlet_
  Charlie">
  </portletinstance>
</component>
```

Inside the component definition for the row on our page, you can see that we have put a portletinstance object with portletref set to Portlet_Charlie. This value must match the uid of the portlet that you wish to put in this container.

So here it is. Let's save this file as import_pages.xml and execute:

```
xmlaccess.bat –in C:\temp\import_
  pages.xml –user wpsadmin –pwd
  wpsadminpwd –url http://portal.
  example.com/wps/config
```

Once our import is complete, fire up a browser and go check out your brand new portal! Wunderbar! You should see your label with a page that has a portlet on it. Experiment.

Of course you can also go further and assign permissions to your objects using XML, or you could create XML files to remove portlets but leave the pages and so forth and so on.

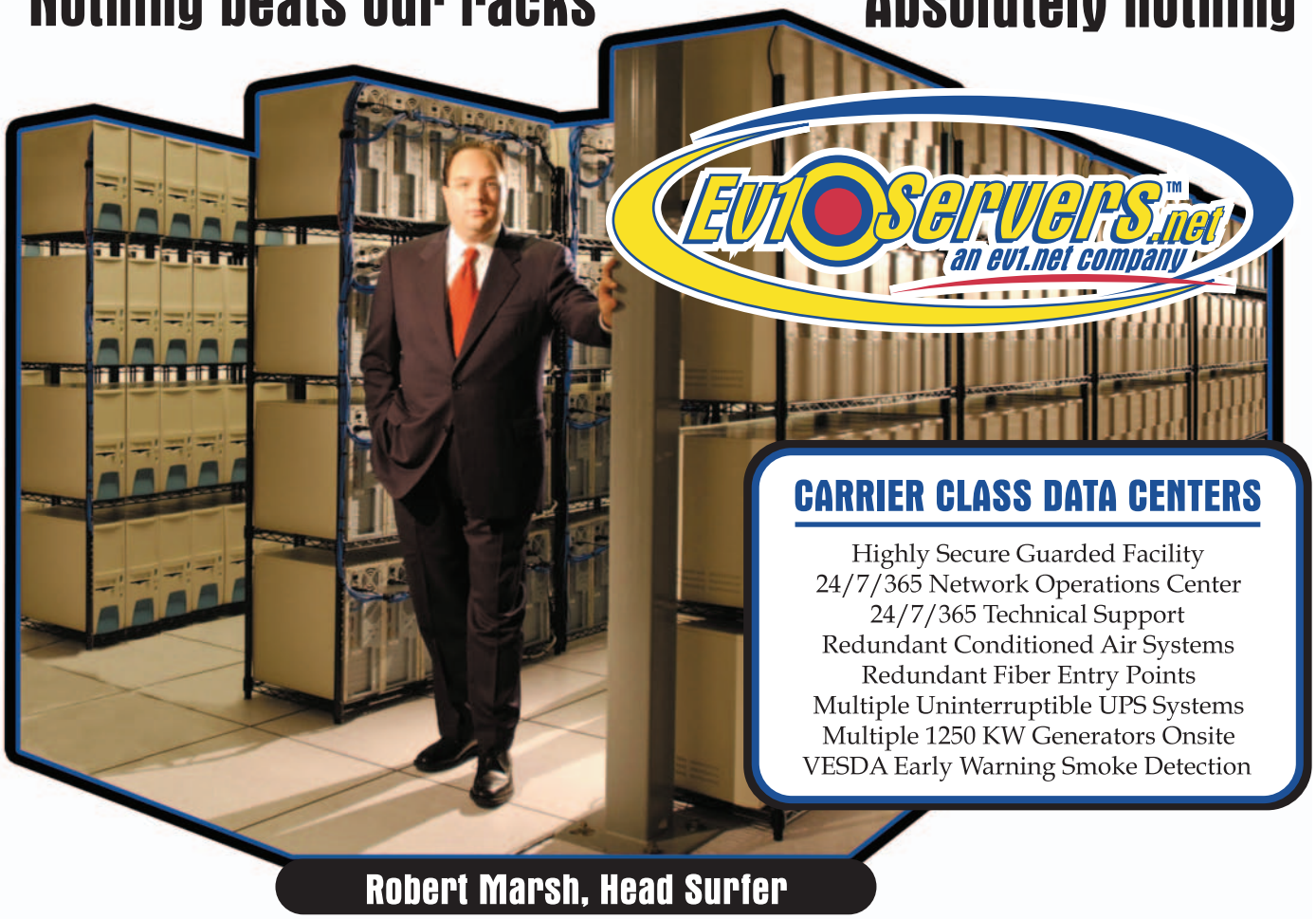
Make liberal use of the example files in C:\WebSphere\PortalServer\doc\xml-samples, or the InfoCenter documents regarding XML Access. These pages are currently located at http://publib.boulder.ibm.com/infocenter/wp51help/topic/com.ibm.wp.ent.doc/wps/admxmlai.html.

LISTING 1

```
<?xml version="1.0" encoding="UTF-8"?>
<request
    xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
    xsi:noNamespaceSchemaLocation="PortalConfig_1.3.xsd"
    type="update" create-oids="true">

    <portal action="locate">

          <skin action="update" active="true" objectid=
           "Skin_A"
                    uniquename="wps.skin.mySkin1"
resourceroot="My_Skin1">
                   <localedata locale="en">
                        <title>Skin 1</title>
                          <description>A skin that is red</
                                description>
                   </localedata>
          </skin>

          <skin action="update" active="true" objectid=
            "Skin_B"
                    uniquename="wps.skin.mySkin2"
                      resourceroot="My_Skin2">
                   <localedata locale="en">
                        <title>Skin 2</title>
                          <description>A skin that is
                            blue</description>
                   </localedata>
          </skin>

    </portal>
</request>
```

Nothing beats our racks

Absolutely nothing



START YOUR OWN WEB HOSTING BUSINESS TODAY!

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

Dedicated Server

Dual Xeon 2.4 GHz

2 GB RAM • 2 x 73 GB SCSI HD
Remote Console • Remote Reboot
2000 GB Monthly Transfer Included

Over 20,000 Servers!

1-800-504-SURF | ev1servers.net

PLESK7 RELOADED
Preferred Control Panel

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

LISTING 2

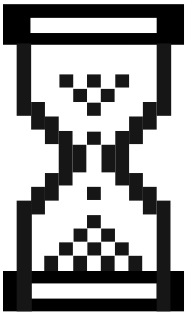
```
<?xml version="1.0" encoding="UTF-8"?>
<request
     xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
     xsi:noNamespaceSchemaLocation="PortalConfig_1.3.xsd"
     type="update" create-oids="true">

     <portal action="locate">

    <skin action="locate" objectid="Skin_A" uniquename=
      "wps.skin.mySkin1"/>
    <skin action="locate" objectid="Skin_B" uniquename=
      "wps.skin.mySkin2"/>

    <theme action="update" active="true" defaultskinref=
      "Skin_A"
          objectid="Theme_Blue" uniquename="wps.theme.
            BlueTheme"
          resourceroot="Theme_1">
                    <localedata locale="en">
                              <title>Big Blue theme</title>
                    </localedata>
                    <allowed-skin skin="Skin_A"
                      update="set"/>
                    <allowed-skin skin="Skin_B"
                      update="set"/>
    </theme>

    <theme action="update" active="true" defaultskinref=
      "Skin_B"
          objectid="Theme_Green" uniquename="wps.theme.
            GreenTheme"
          resourceroot="Theme_2">
                    <localedata locale="en">
                              <title>Big Green Theme</title>
                    </localedata>
                    <allowed-skin skin="Skin_B"
                      update="set"/>
    </theme>

     </portal>
</request>
```

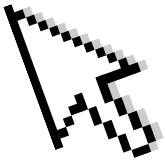
LISTING 3

```
<?xml version="1.0" encoding="UTF-8"?>
<request
     xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
     xsi:noNamespaceSchemaLocation="PortalConfig_1.3.xsd"
     type="update"
     create-oids="true">

     <portal action="locate">

      <web-app action="update" active="true"
      uid="com.yourco.portlet.PortletAlpha">
         <url>file:///$server_root$/installableApps/
           PortletAlpha.war</url>
         <portlet-app action="update" active="true"
         uid="com.yourco.portlet.PortletAlpha.1">
              <portlet action="update" active="true"
              objectid="Portlet_Alpha"
          name="Portlet_Alpha Application">
              </portlet>
         </portlet-app>
       </web-app>

      <web-app action="update" active="true"
      uid="com.yourco.portlet.PortletBravo">
         <url>file:///$server_root$/installableApps/
           PortletBravo.war</url>
         <portlet-app action="update" active="true"
         uid="com.yourco.portlet.PortletBravo.1">
              <portlet action="update" active="true"
              objectid="Portlet_Bravo"
          name="Portlet_Bravo Application">
              </portlet>
         </portlet-app>
       </web-app>

      <web-app action="update" active="true"
      uid="com.yourco.portlet.PortletCharlie">
         <url>file:///$server_root$/installableApps/
           PortletCharlie.war</url>
         <portlet-app action="update" active="true"
         uid="com.yourco.portlet.PortletCharlie.1">

              <portlet action="update" active="true"
              objectid="Portlet_Charlie"
          name="Portlet_Charlie Application">
              </portlet>
         </portlet-app>
       </web-app>

     </portal>
</request>
```

LISTING 4

```
<?xml version="1.0" encoding="UTF-8"?>
<request
     xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
     xsi:noNamespaceSchemaLocation="PortalConfig_1.3.xsd"
     type="update"
     create-oids="true">

     <portal action="locate">

           <theme action="locate" active="true"
uniquename="wps.theme.BlueTheme"
          resourceroot="Theme_1" objectid="Theme_Blue"/>

          <web-app action="locate" active="true"
          uid="com.yourco.portlets.PortletCharlie">
             <url>file:///$server_root$/installableApps/
               PortletCharlie.war</url>
             <portlet-app action="locate" active="true"
             uid="com.yourco.portlets.PortletCharlie.1">
                  <portlet action="locate" active="true"
                  objectid="Portlet_Charlie" name="Portlet
                    Charlie Application">
                  </portlet>
             </portlet-app>
          </web-app>

          <content-node action="locate" objectid="6_0_A
            Content Root"
          uniquename="wps.content.root"/>

          <content-node action="update" objectid="HomeTab"
          uniquename="wps.HomeTab" content-parentref="6_0_A
            Content Root"
          active="true" ordinal="first" allportletsallowed=
            "true"
          create-type="explicit" type="label" themeref=
            "Theme_Blue">
              <supported-markup markup="html" update="set"/>
              <localedata locale="en">
                  <title>Home</title>
              </localedata>
    </content-node>

          <content-node action="locate" objectid="HomeTab"
          uniquename="wps.HomeTab"/>

          <content-node action="update" objectid="WelcomePage"
          uniquename="wps.HomeTab.WelcomePage"
            content-parentref="HomeTab"
          active="true" ordinal="100" allportletsallowed=
            "true"
          create-type="explicit" type="page">
              <supported-markup markup="html" update="set"/>
              <localedata locale="en">
                  <title>Welcome</title>
              </localedata>
              <component action="update" active="true"
              objectid="container1" ordinal="-1"
orientation="H"
              type="container" width="300">
                  <portletinstance action="update" handle=""
                  objectid="portletinstance1"
portletref="Portlet_Charlie">
              </portletinstance>
              </component>
          </content-node>

     </portal>
</request>
```





World Wide Wait.

Don't Leave Your Customers Hanging — Ensure a Positive End User Experience with Quest Software.

They don't care where the problem is. All that matters is their experience using your site. But can you quickly detect and diagnose a problem and deploy the right expert to fix it?

Quest Software's solutions for J2EE application performance management shed light on the true end user experience and help you isolate and fix problems across all tiers. Whether a problem lies in your server cluster, database, framework or Java code, nothing gets you to resolution and optimal performance faster.

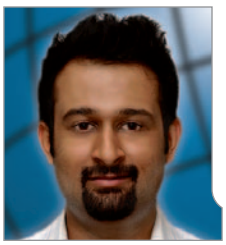
Don't just cross your fingers and hope for the best. Get more with Quest.

Download the free white paper,
"Measuring J2EE Application Performance in Production"
at www.quest.com/wsdj

Optimizing Web content delivery with WebSphere portal server

Between the Covers with WebCM

BY MINESH MANILAL



Minesh Manilal, a specialist in Workplace Web Content Management and J2EE architectures, has led and participated in a number of complex portal, collaboration, and content management projects for the government of Nassau County in New York, AMP Inc., Schering AG, Linpac Group Ltd., Royal & Sun Alliance Insurance Group, and Aviva. Minesh is a key member of Prolifics' highly specialized team of IBM WebSphere experts retained to build highly customized portal solutions as well as install, administer, develop, and migrate run-the-business apps and Web sites using WebSphere and Lotus products.
mmanilal@prolifics.com

The evolution of Web content management is not unlike the evolution of Web application servers and Web portal servers and Web anything else. The software was created to solve a single problem – the problem of managing content. But, when combined with the demands of the Web all your middleware capabilities quickly become critical, such as ensuring scalability, reliability, and security.

Web content management software was originally designed to make Web publishing easier, more dynamic, and more personalized. In fact, one of the pioneers and original market leaders was Vignette. When Vignette was formed in 1995 the founders acquired technology from CNET, the publishing house that had developed a sophisticated online publishing system for its high-tech news.

Other kinds of content management solutions also existed in the market such as Documentum, which came from a document management heritage. The various solutions have extended their functionality into each other's niches by acquiring, partnering, or building it. For example, the ability to do indexed search, database management, repositories, and workflow are appearing in more and more solutions in support of content-rich Web sites.

However it's not just about content. It's about architecture. With the demand for secure content-rich Web sites that scale, support B2B, and integrate, companies are looking to deploy a robust J2EE architecture that give them a level of infrastructure support. This is where IBM demonstrates a lot of strength – it can provide the added value on top of its Web content management solution, IBM Workplace Web Content Management (WebCM).

Remember, it's not just about content. It's also about portal integration. Why? The new desktop that we see now and going forward is a portal – an integration of an organization's assets in a unified and personalized view, using a single sign-on. So the topics go together. You don't want to aggregate your company's information in a portal except for your content and documents. On the contrary – it's essential to integrate your Web content in a portal so that the portal becomes the content's delivery vehicle. This is where IBM continues to demonstrate strength – it has the market-leading portal server integrated with its Web content management solution, WebCM.

This article is the first of a two-part series. The first part will discuss WebCM, its benefits, and how it integrates with IBM WebSphere Portal Server. The second part will detail the repeatable steps involved in migrating from Vignette to WebCM, and a redeployment strategy.

WebCM Overview

WebCM is relatively new. WebCM (or Aptrix, as it used to be called) originated on the Lotus Domino platform in the early days of Web content management and then migrated to the WebSphere platform making it more accessible and more likely to grow into a true enterprise solution.

Content management is becoming increasingly important for businesses competing in an on-demand environment. The ability to manage and deliver Web content efficiently and effectively is crucial for their success, and WebCM facilitates this from the initial authoring stage to the final presentation through its collaborative design, approval, caching, and staging features. The flexibility of WebCM allows both technical and non-technical people to easily create Web site content.

Another, and maybe the most important, benefit of WebCM is its integration with WebSphere Portal Server which lets it be integrated seamlessly with other IBM products, such as the IBM Portal

Document Manager (PDM), so businesses can fulfill their strategic objectives.

WebCM's Features

WebCM's key features are listed below:

DYNAMIC PRESENTATION OF CONTENT

WebCM lets content be delivered dynamically, meaning that as Web content is created or modified it dynamically changes the Web site by using dynamic navigation through Navigator, Menu, and JSP components.

PERFORMANCE AND CACHING

WebCM provides different caching options and has a pre-rendering feature. The basic-to-advanced caching options store content in either memory and/or on the hard disk. The options can be used in conjunction with memory/hard disk caching to improve performance. Basic caching, which renders a page at a time, is a more flexible solution for a large Web site making its content more current because of expiration management. However, it can't cache dynamic, personalized, or secured content. Advanced caching, which can selectively cache content by session or users, provides the same output, but includes a 'connect' tag to selectively cache (or un-cache) components in a Web page. It's useful for a Web site whose content is largely static with some dynamic components.

WebCM's pre-rendering collates design and content components and stores them on a physical disk as complete HTML files. The pre-rendering engine renders an entire site (based on the site framework selected) and can't be broken down into partial pre-rendering. It's done by enabling the <Cacher class=….> module in the connect.cfg. The resulting HTML files can be stored on another device and used by other Web servers. Pre-rendering doesn't have expiration management and a pre-rendered site is replaced when the pre-rendering engine is triggered. It's suitable for small and mid-sized Web sites that contain static information with no personalization.

INTEGRATION OF CONTENT FROM DISPARATE SYSTEMS

WebCM can connect to systems such as DB2 and SQL Server and Web applications, and by using its just-in-time parsing of HTML and XML elements, it can integrate information from these sources as Web content.

PRESENTATION AND DELIVERY OF AGGREGATED CONTENT

Unlike other content management solutions, WebCM has decoupled its presentation layer from the content layer. By allowing for the creation of HTML templates, called Page Designs in WebCM, you can have different presentations of aggregated content and still enforce a consistent look-and-feel.

Since the presentation layer is separated from the content, it lets content creators create content that can be used in various sections of the site. While the same content can be associated with multiple sections of the site with WebCM, you can select different page designs for these sections and present the same content differently.

Another benefit of this separation of presentation and content is that you can use multiple delivery mechanisms. For example, content can have various page designs associated with it so that it can be viewed using WAP devices as well as Web browsers.

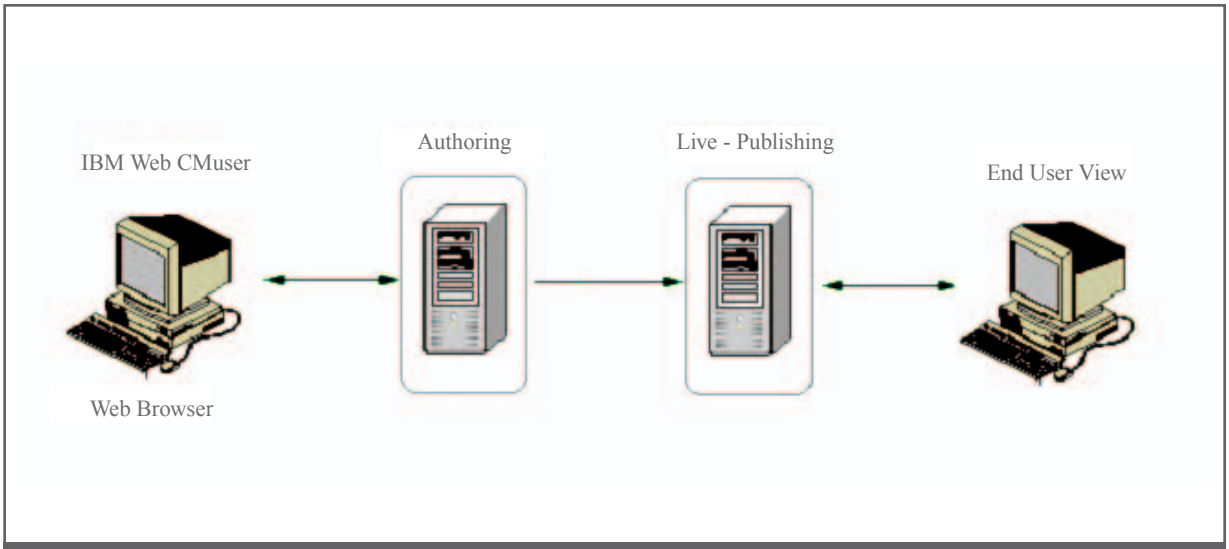


FIG 1: SAMPLE TWO-STAGE ENVIRONMENT FOR WEBCM

RAPID DEPLOYMENT (SYNDICATION)

WebCM can rapidly deploy changes across its staging environments through its syndication abilities. Syndication is the ability to push or pull content from various environments.

It does it by letting Syndicators and Subscribers be created in the various environments. Let's look at a simple example of a centralized two-stage environment architecture, as shown in Figure 1.

We would set up a Syndicator in the authoring environment so content can be pushed to the production environment. A subscriber would have to be set up in the production environment so the information from the authoring environment can be inserted in the database.

There are two ways to send data between two environments: All Items or All Live Items. All Live Items sends all items that don't have a workflow associated with them or items published in the workflow. All Items syndicates all items in the WebCM database whether or not they're in a workflow.

Choosing which to use to gather items for syndication depends on how you want to move the data between environments and the architecture you've deployed for the system. For example, in a simple two-stage centralized system that consists of an authoring environment and a production environment, you may only want to push live items in which case you would use All Live Items. However, in a three-stage environment that has authoring, staging, and production environments, you may want the items in the authoring and staging environments to share the same information so you would use authoring to syndicate All Items to staging.

The time it takes to syndicate data across environments depends on a number of variables like the number of items that have to be syndicated, the environment the system was configured in, and the network that the environments were set up in.

WORKFLOW

WebCM uses a workflow model that can manage content from draft to publishing. The complexity of the workflow can vary from a very simple two-stage draft/publish process to a more complex one where approval stages are introduced and e-mails are sent at various stages.

This ability to manage content through a workflow process is important for a content management system and lets the right stakeholders be involved in the approval process.

VERSION CONTROL

WebCM has versioning capabilities covering everything from design components and information architecture to content objects. It can version everything with its version-all functionality and restore everything through its restore-all functionality. It also provides a label that describes the version.

When versions are created, the system associates a timestamp with the items so that individual versions, which can be associated with a point in time by the label, can be rolled back to a specific time.

Because of WebCM's flexibility and configurability, the amount of versioning can be controlled through its configuration files, so if you only want content and the information architecture to be versionable it can be set up that way.

The system can also preview old content versions, the only drawback being that they would have to be previewed through the presentation layer of the current site.

WebCM and Portal Integration

With the launch of WebSphere Portal 5.1 and future dot releases, WebCM has moved from a standalone application in the WebSphere Application Server that could integrate into WebSphere Portal to being part of the Portal offering. This integration is strategic to IBM's plan for Lotus Workplace. WebCM is a key component for content delivery and by integrating it IBM can use its J2EE code base to grow and enhance its capabilities with the IBM Workplace platform.
Several key changes resulting from

"Content management is not just about content; it's about architecture and portal integration"

this integration should be highlighted.

With WebCM part of the WebSphere Portal, the install process is simpler. Gone are the days when you had to install each component separately, copying files from directory to directory, and creating shared libraries to access the WebCM API. Now the whole process is done in one install, making it consistent with other IBM installers.

WebCM's user management has moved to WebSphere and the WebSphere Member Manager letting it take advantage of the security features that go along with the WebSphere platform. And by using the same security model, integration with other Portal applications is seamless. Once you've logged into the WebSphere Portal it knows who you are.

There's also better integration with Portal search. In the old versions of WebCM you could integrate with Portal Search but it was limited. Now Portal can index its searches from WebCM better and the Portal Search result sets can aggregate content from various sources.

Historically WebCM, even as an application in WebSphere Application Server, couldn't be clustered, so in a cluster environment, a separate instance of WebCM had to exist on each of the servers and syndication had to be used to send the same information to these instances. Now, with integration, clustering is possible.

As a standalone product on the WebSphere Application Server, WebCM used applets through a browser interface as its user interface but now it uses portlets, which has made it more customizable. Actually it's the first time developers have been able to customize the user interface.

Along with integrating with the Portal Server, WebCM's own functionality has been enhanced. With WebSphere Portal 5.1, WebCM now has an enhanced user interface, national language enhancements, managed reuse of Portal documents, and advanced content personalization:

- Enhanced user interface – WebCM is now deployed as a Portal Application, which consists of two portlets, an authoring portlet and a rendering portlet. The authoring portlet lets users create and mange their content. The rendering portlet lets them view and preview content. Through this new interface, specific items in WebCM can be searched for and the rich text editor has been dramatically improved so links and images can easily be put in the content without anyone knowing HTML.
- National language enhancements – WebCM now provides support for more languages offering better global deployment strategies for localized content. This enhancement also enables local content creators to create content in their local languages.
- Managed reuse of portal documents – Documents created in Portal can be reused in WebCM because of their integration and new WebCM components that explicitly allow reuse. Reused documents can be opened using Portal Document Manager (PDM).
- Advanced content personalization – Due to the integration, content can be personalized using the personalization rules built into WebSphere and new WebCM capabilities that personalize content spots in Web pages for delivery to targeted audiences based on the user profiles in the system.

WebCM Roadmap

IBM is focusing on integration this year, including making it possible to leverage from other Portal offerings, such as Process Choreographer, to manage business workflows.

The future for WebCM is looking exciting with the promise of an integrated development environment (IDE), repository syndication, and task-based workflows, all enabling WebCM to position itself for the next evolution of Web content management and help businesses around the world realize their potential using the IBM Portal and Workplace.

Managing complexity

The Value of Modeling

BY GARY CERNOSEK

Modeling can be an effective way to manage the complexity of software development. It enables you to understand and analyze your requirements, explore possible architectures, develop and communicate your designs, assess changes, and create robust, flexible systems. In spite of these virtues, mainstream software development has yet to take full advantage of modeling in everyday practice.

Gary Cernosek is a Market Manager in the Rational software division of IBM Software Group where he is responsible for analyzing and responding to market trends taking place in software development. Gary recently served as the Market Manager for IBM Rational visual modeling and model-driven development products. His current focus is promoting how the latest industry standards and tools make Java development easier. Gary also held positions in Rational brand field sales, field technical training, and customer consulting.
gcernosek@us.ibm.com

What is Modeling?

For many years, business analysts, engineers, scientists, and other professionals who build complex structures or systems have been creating models of what they build. Sometimes the models are physical, such as scaled mock-ups of airplanes, buildings, or automobiles. Sometimes the models are less tangible, such as business financials models, market trading simulations, and electrical circuit diagrams. In all cases, the model serves as an abstraction — an approximate representation of the real item to be built.

Why Model?

Should you model everything before you build it? Absolutely not. You probably don't need a model to create things such as a simple checkbook register, a currency conversion utility, a doghouse, or a simple macro that opens up a set of routinely used word processor files. Such projects share all or most of the following characteristics:

- The problem domain is well known.
- The solution is relatively easy to construct.
- Very few people need to collaborate to build or use the solution (often only one person is involved).
- The solution requires minimal ongoing maintenance.
- The scope of future needs is unlikely to grow substantially.

But suppose none of these characteristics apply? Why do some professional disciplines bother to create models? Why not just build the real thing right away? The answer relates to the degree of complexity, the risk, and whether those who envision the project are appropriate or available to complete the task

It's neither technically wise nor economically practical to build complex systems without first creating a design, a blueprint, or some other abstract representation. Although most professional architects might build a doghouse without a design diagram, they would never construct a 15-story office building without first developing an array of architectural plans, diagrams, and mock-ups for visualization. Modeling helps architects and others visualize entire systems, assess different options, and communicate designs more clearly before they and their clients take on the risks — technical, financial, or otherwise — of actual construction.

Why Model Software?

For years, the practice of software development was exempt from many of these modeling issues. By its very nature, software can be easily created and easily changed. It requires little capital equipment and incurs virtually no manufacturing costs. These attributes cultivated a do-it-yourself culture among developers. "All I have to do is imagine it, build it, and change it as often as necessary," they reasoned. "There is no 'final' system anyway, so why even try to conceive of one before writing code?"

This is not an effective way to approach today's software systems, which have become very complex. In many cases, developers must integrate them with other systems to run the machines we use in our everyday lives. Automobiles, for example, are now heavily equipped with computers and associated software to control everything from the engine and cruise control to

new onboard navigation and communication systems. Developers also build software to automate business processes of all kinds — both those in the backoffice and those that customers see and experience.

Software systems that support important health-related or property-related functions are necessarily complex to develop, test, and maintain. And some systems are critical to businesses. In many organizations, software development is no longer a cost-center overhead line item — it's integral to the companies' strategic business processes. For them, software has become a key discriminator for competing in the marketplace.

For these reasons and more, developers need a better understanding of what they are building, and modeling can provide that. Customers and business users still expect software to be delivered on time and to perform as expected on-demand. To meet these "fast and good" expectations, software developers can follow four imperatives: Develop iteratively, focus on architecture, continuously ensure quality, and manage change and assets. They can model software for the same basic reasons professionals in other fields model complex, high-risk systems — to manage complexity and understand the design and associated risks. More specifically, by modeling software, developers can:

- Create and communicate software designs before the organization commits additional resources.
- Trace the design back to the requirements, helping to ensure that they are building the right system.
- Practice iterative development; using models and other higher levels of abstraction facilitate quick and frequent changes.

What People Are Saying About the Value of Modeling

Like any technology, UML had early adopters who led the charge in discovering its value. Here are a few comments from IBM Rational customers about the value modeling contributed to their businesses:

"We are trying to reduce the overall cost of insurance to our members. One of the ways to do that is to reuse information and reuse the assets that we build as we go through our business modeling… Model-driven architecture is really at the core of what we're doing from a business modeling perspective. When we begin projects from a software development perspective without a clear business model, without a clear set of business objectives or business goals, we are finding that the customers don't get what they think they have asked for."

- Sue Nelson, director of business modeling for Blue Cross and Blue Shield of Florida

"Enterprise architecture presents its own very unique modeling challenges. You are modeling at multiple levels. You are modeling with large groups of people and different teams. And the models at each level tend to have to be customized for the individual stakeholder types. [Modeling with UML] provided us with the flexibility [to meet] our unique needs and demands at each level of the enterprise architecture."

- Frank Armour, president of ArmourIT, LLC

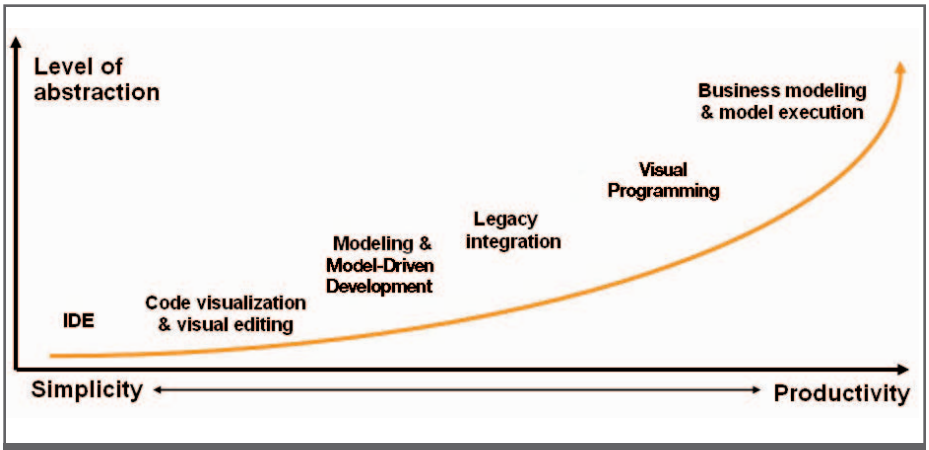


FIG 1: A SPECTRUM OF MODELING APPROACHES

Testimonials like these illustrate how modeling enables organizations to reduce risk, and they should assure those who are just getting started with modeling that their efforts will pay off. Ultimately, spreading the word about modeling's value will help bring the practice into the software development mainstream.

Why Some Developers Choose Not To Model Software

Despite the many reasons and virtues behind modeling, most software developers still don't use any form of abstraction higher than source code. Why? As we noted earlier, sometimes the problem doesn't warrant it. But in the world of software, systems that are simple when first implemented often become increasingly complex as developers make changes in response to new requirements or integrate with other systems. In other cases, developers choose not to model because they simply don't perceive a need for it – until it's too late.

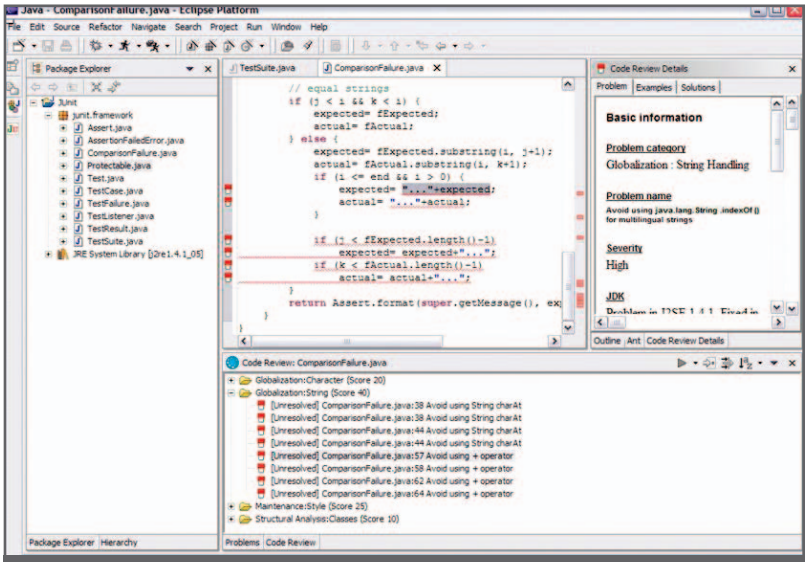


FIG 2: IDE-CENTRIC DEVELOPMENT

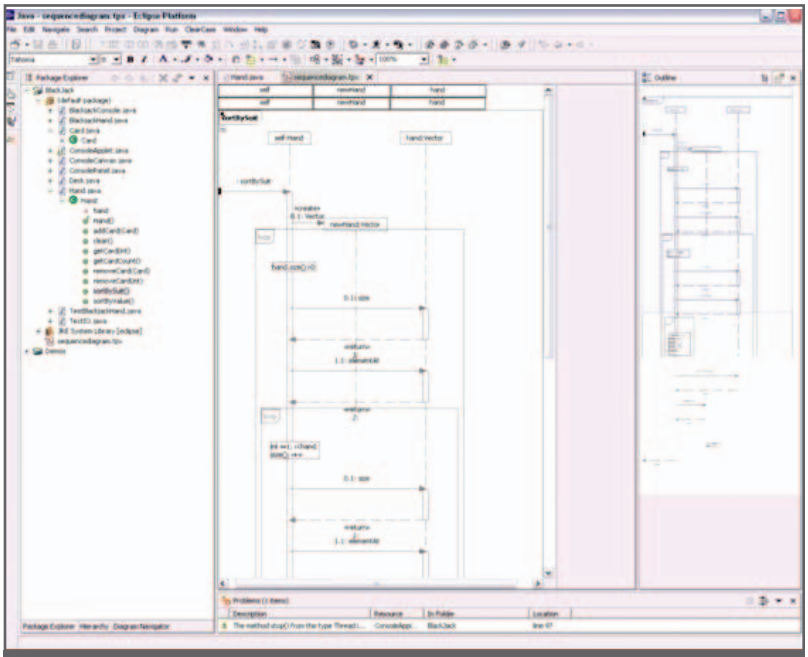


FIG 3: VISUALIZATION IN IBM RATIONAL APPLICATION DEVELOPER

Many experts argue that resistance to modeling software is more cultural than anything else. Traditional programmers are proficient at conventional coding techniques. Even when the complexity level rises unexpectedly, most developers are comfortable sticking with their integrated development environment (IDE) and simply working more hours on the problem. Because modeling requires additional training and tools, it also requires a corresponding investment of time, money, and effort – not at the time of toil, but early in a project's development lifecycle. Traditional developers don't proactively pursue modeling because they believe it will slow them down. In the next section, we'll attempt to dispel this notion.

When Do I Model?

Modeling complex applications has several general benefits. Specifically, modeling can help software development organizations to:

- Better understand the current business or engineering situation ("as-is" model) and craft a better system ("to-be" model).
- Build and design an efficient system architecture.
- Create visualizations of code and other forms of implementation that enable better communication among project team members.

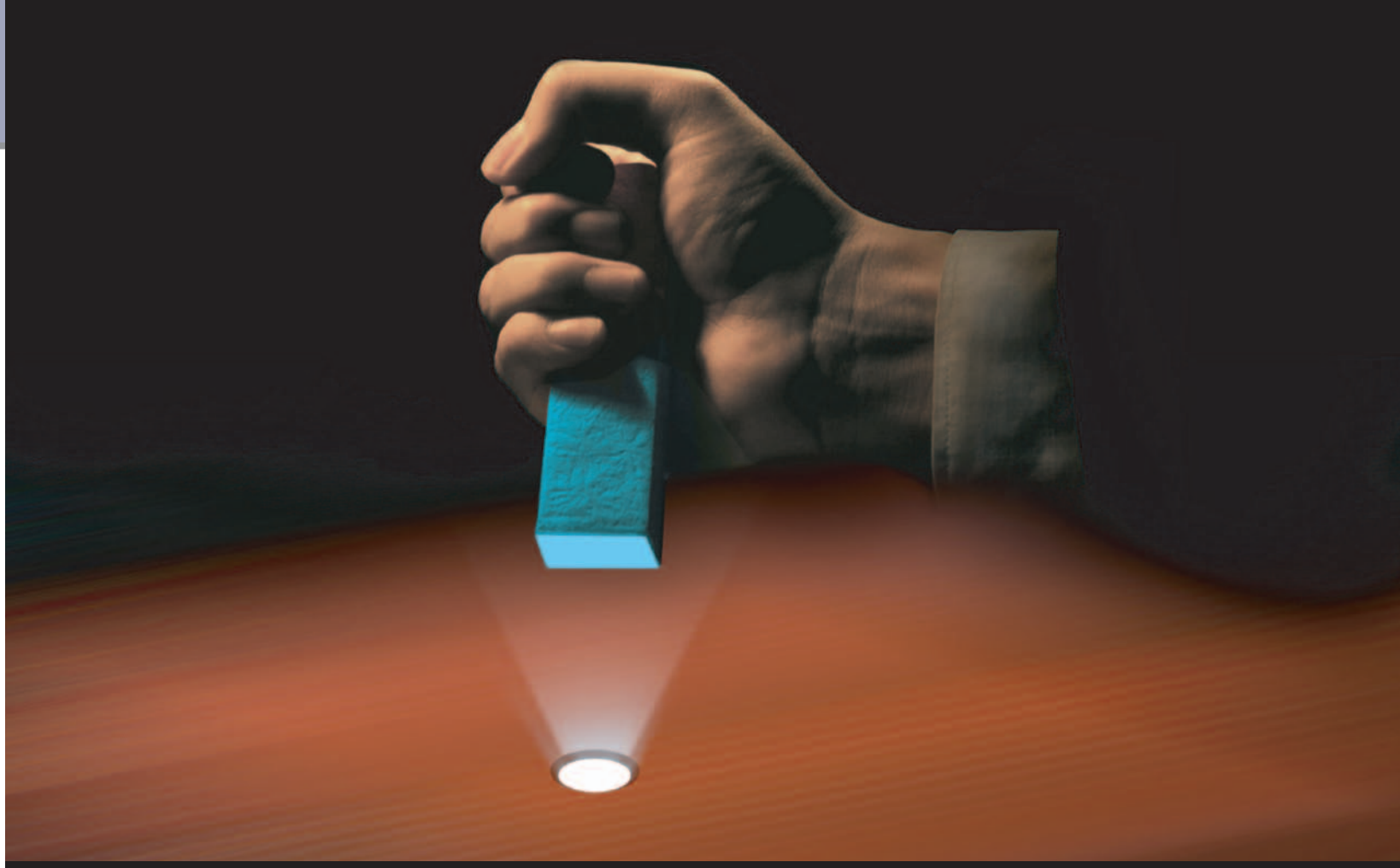
However, it's important to remember that modeling isn't an all-or-nothing proposition. Software development organizations can pick and choose how to use models in the software development process, as shown in Figure 1. We'll examine each of these modeling opportunities below.

Integrated Development Environments

In the loosest notion of modeling, an organization's IDE is a logical entry point into the practice of model-driven development. Modern IDEs already offer several mechanisms that raise the abstraction level for creating and maintaining code: language-sensitive editors, wizards, form builders, and other GUI controls. Like models, these tools help developers be more productive, create more reliable code and institute a more effective maintenance process (see Figure 2). All of these attributes are the essence of model-driven development.

Code Visualization and Visual Editing

A step above the basic IDE functions is visualizing source code graphically. Developers have used graphical forms of abstraction above their code for years. Traditional flow charts are a common method for depicting the algorithmic control flow of code.



Think Crystal is a good fit for your Java application? Think again.

Think JReport.

Forcing a Windows reporting solution into a J2EE environment is never going to result in a perfect fit. Only JReport is built from the ground up to leverage J2EE standards and modular components for seamless embedding in any Web application.

JReport is ready out-of-the-box, with all the tools necessary to empower your application with any reporting requirement. From reusable, shared report components to flexible APIs, JReport is the most complete embedded reporting solution available.

With the ability to scale to multi-CPU and server clusters, JReport is a perfect fit for any reporting workload. Load balancing and failover protection provide peak performance and uninterrupted access to critical business data. In addition, JReport integrates with any external security scheme for single sign-on.

JReport's ad hoc reporting lets users access and analyze data on demand, from any browser. And, with cascading parameters, embedded web controls for dynamic sorting and filtering, drill-down and pivot, JReport ensures users get the information they want, when they want it, and how they want it.

See for yourself why over half of the world's largest organizations have turned to JReport to enable their J2EE applications with actionable reporting.

When it comes to embedded Java reporting, JReport is the perfect fit. Download a FREE copy of JReport today at www.jinfonet.com/wj4.

JReport
JINFONET SOFTWARE

Structure charts, or even simple block diagrams with arrows, are often used on whiteboards — with boxes to represent functions and subprograms, arrows to indicate calling dependencies, and so on. For object-oriented software, boxes typically denote classes (or object instances of classes), and lines between boxes denote relationships of various types (see Figure 3).

Coupled closely with code visualization is visual editing, which lets developers edit code through diagrams instead of through conventional IDE text windows. Visual editing is well suited to changes that have systemic effects on other pieces of code. Using conventional code editors to enact such changes can be tedious and error-prone. But an effective visual editor lets developers, for example, drag and drop a member function from one class to its base class and automatically adjust all code affected by such a change.

In one sense, code visualization and visual editing are simply alternative methods for viewing and editing the code. Changes to the code are immediately reflected in corresponding diagrams, and vice versa. Although some may argue that such depictions don't constitute a "model," the essence of modeling is abstraction, and any visualization of code is indeed an abstraction — selectively exposing certain information while suppressing unnecessary or unwanted details.

"It's neither technically wise nor economically practical to build complex systems without first creating a design, a blueprint, or some other abstract representation"

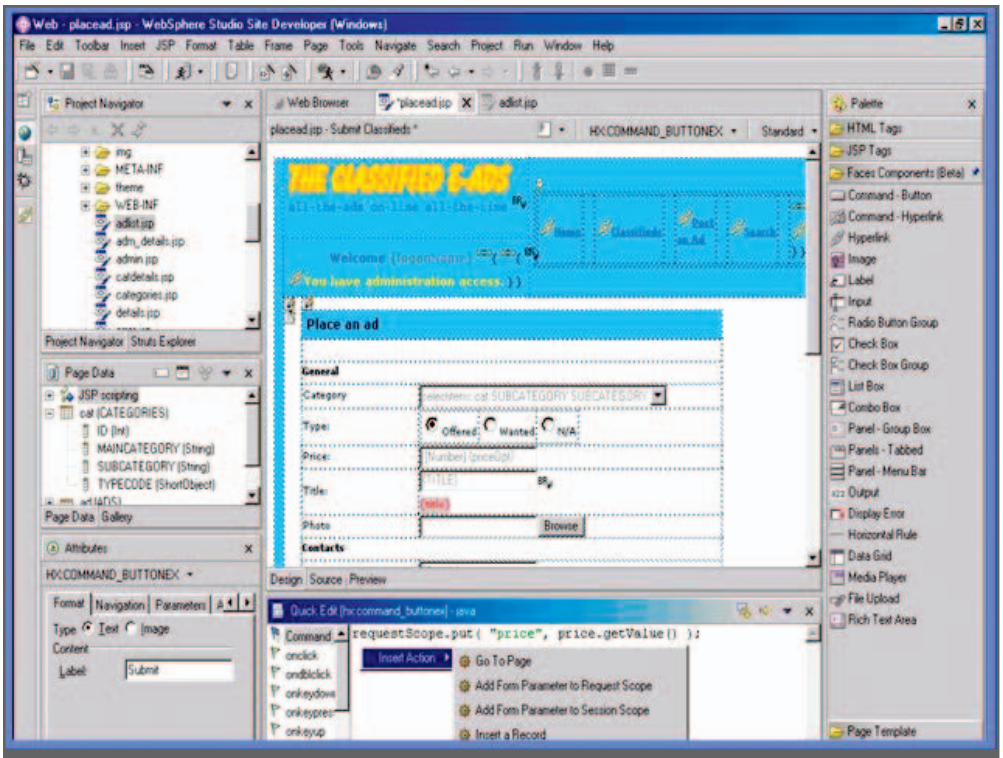


FIG 4: VISUAL PROGRAMMING IN IBM RATIONAL APPLICATION DEVELOPER

Modeling and Model-Driven Development

The next step on the modeling spectrum represents the state of conventional model-driven development. Here, visual models are created from a methodological process that begins with requirements and delves into a high-level architectural design model. Developers then create a detailed design model from which skeletal code is generated to an IDE. The IDE is used to complete the detailed coding. Any changes made to the code that affect the design model are synchronized back into the model; any model changes are synchronized into the existing code.

Legacy Integration

When developers are ready to integrate systems — whether the systems are all legacy or a mix of legacy and new — they must first understand the systems in place, know how the business wants all systems to work together, and prioritize the integrations. Modeling

legacy systems doesn't necessarily mean depicting all components of all systems; developers need just enough information to understand how the legacy systems' architectures work and interface with others. Understanding what the system does and what other software is dependent on it will help determine suitable steps moving forward.

Developers use several methods to model legacy systems. They can reverse engineer code into models to understand them, manually model them, or use some combination approach.

Visual Programming

The practice of visual programming has been around since the early 1990s. The premise is simply to provide highly productive ways to generate and maintain code. The practices are based on easy-to-use, highly graphical features found in advanced IDEs (see Figure 4). Visual programming raises the level of abstraction above the code but doesn't use models per se.

Business Modeling and Model Execution

Even before they realize that there's a need to develop additional software, business and engineering analysts often find it useful to create models of how their systems work today. From these models, they can analyze what works and what needs improvement. Special-purpose tools can simulate the models along several key variables, such as time, cost, and resources. Then, by examining these models, the analysts can build additional models to prescribe how new improved processes should work. Typically, new software development is needed to implement the new processes, and the models serve as key drivers for that development. For some application domains, the process models are specified in enough detail to generate complete applications from the models. Modeling at this level of abstraction offers the greatest potential for productivity and the greatest level of integration between the business or engineering problem domains and the technology or implementation domains.

How Do I Model?

The software industry has adopted the Unified Modeling Language (UML) as its standard for representing software models and related artifacts. Software architects, designers, and developers use UML for specifying, visualizing, constructing, and documenting all aspects of a software system (see Figure 5). Key leaders from IBM Rational led the original development of UML, which today is managed by the Object Management Group (OMG). That organization consists of representatives throughout the world who help ensure that the specification continues to meet the software community's dynamic needs. Adopting a standard notation such as UML is an important step in taking a model-driven approach to software development. UML is more than just a graphical notational standard — it's a modeling language. As with all languages, it defines syntax and semantics (the underlying meanings of the symbols and text).

Trends and the Future

Ask any software development professional, "Where is the software industry heading?" and you will probably get a wide array of responses. But one trend seems to be quite common:

Software development continues to grow in complexity, and developers must work at increasingly higher levels of abstraction to cope with this complexity.

Modeling software is - and will continue to be - a key way that developers work at those higher levels.

Below, we will note other current trends that relate to modeling as well.

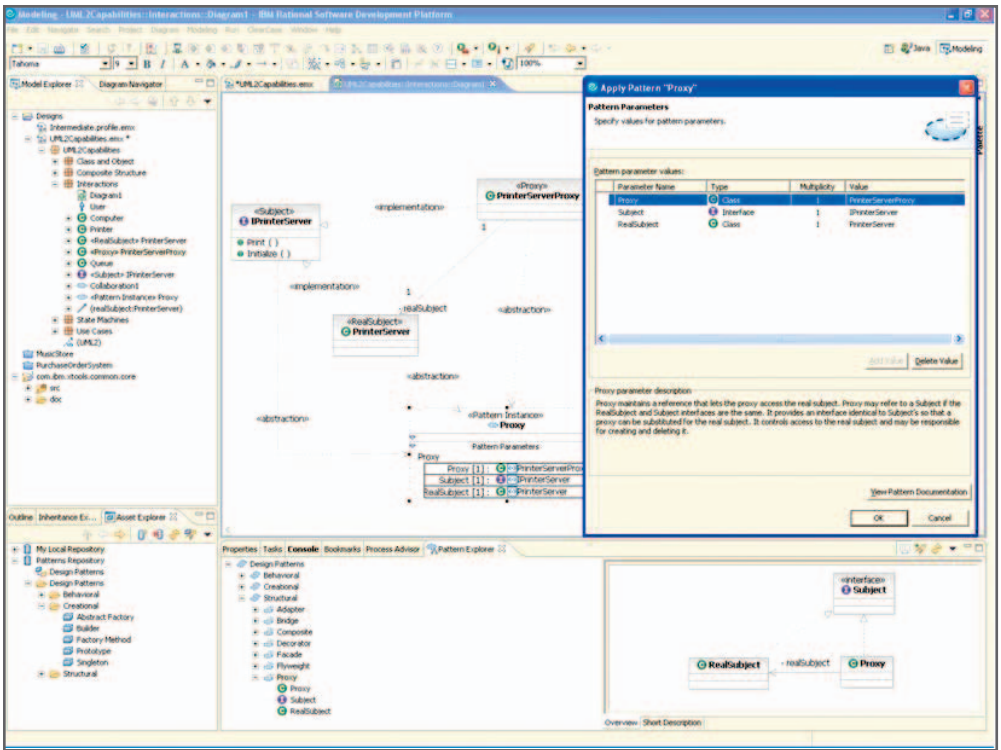


FIG 5: MODELING WITH UML 2.0 IN IBM RATIONAL SOFTWARE ARCHITECT

Unifying Software, Data, and Business Modeling

Although this article has focused primarily on the value of modeling software, we should remember that organizations have recognized the value of data modeling and business modeling even longer. The problem is that the modeling languages and tools these disciplines use are typically worlds apart. However, there is now a promising prospect for unifying these separate disciplines, not necessarily with a single modeling language or tool, but through a combination of multiple, open, converging industry standards (see Figure 6).

As standards continue to evolve, modeling will become applicable to an even broader range of activities across the software development lifecycle. Modeling technology is already driving testing and other quality assurance activities earlier in the lifecycle. And as business modeling becomes more standardized and integrated with data and software, a business-driven development discipline will likely emerge.

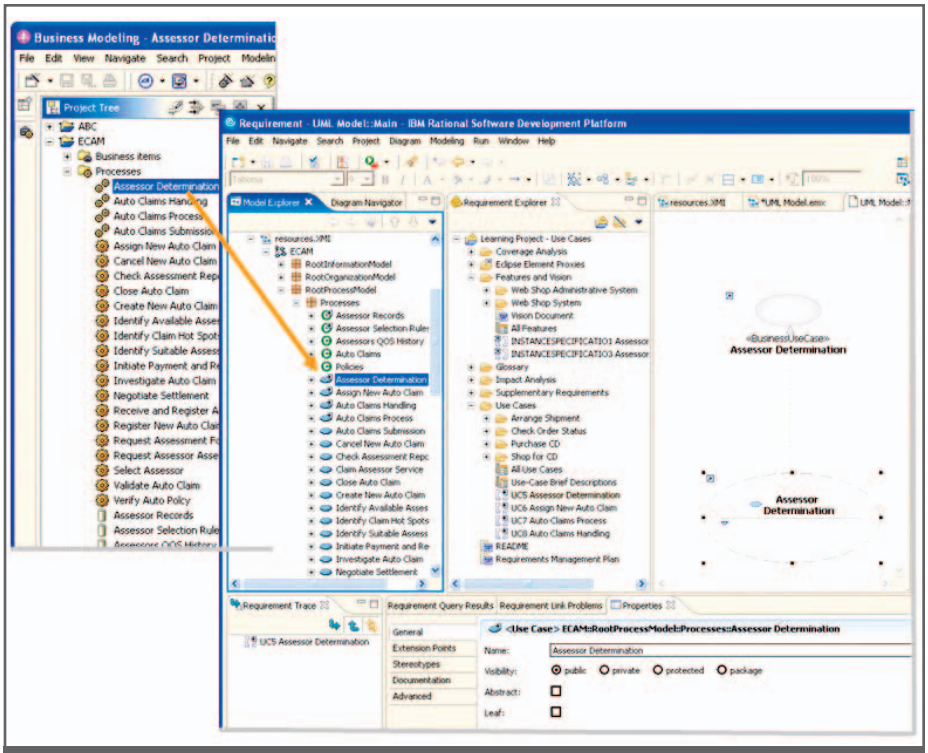


FIG 6: BRINGING BUSINESS AND SOFTWARE MODELING TOGETHER IN IBM RATIONAL SOFTWARE ARCHITECT

Domain-Specific Modeling Languages

As we noted earlier, UML and other modeling languages let developers focus on levels of abstraction above implementation details (see Figure 1). Taken to the highest level of abstraction, a business or domain model focuses not on software, but on the nature of the problem under consideration. Here, the model should use terms and icons familiar to the people in that particular business or application field.

The industry is currently considering domain-specific languages — special-purpose modeling languages dedicated to their respective areas of use. More often, however, organizations extend a general-purpose modeling language—UML, in particular—in standard ways to meet domain-specific modeling needs through innovations such as profiles. Both approaches deliver on modeling's inherent value: to provide abstractions for specifying problems and solutions in a more productive and effective manner.

Software Development as a Global Activity

Many have called software development a "team sport." In fact, it's now an "international team sport." With today's technology, software development has no geographical boundaries, and it will likely grow increasingly distributed and global. Modeling and other higher forms of abstraction will be crucial for helping practitioners handle the associated complexity.

Model-Driven Architecture (MDA): The Next Step

MDA is an initiative led by the Object Management Group. While still in its early adopter stage, MDA can be considered the next logical step in the evolution of modeling and model-driven development technologies. Based on UML and other related standards, MDA focuses on defining models at varying levels of abstraction and on the transformations defined between these different levels. Automated tool support is crucial to the evolution and successful application of MDA.

Conclusion

The purpose of this article has been to define modeling in the most fundamental way, and to emphasize the value of applying modeling to software development. Modeling is about thinking and working at higher levels of abstraction. It is a tried and true technique that has proven successful for many years across all forms of engineering and technical disciplines.

No matter where your organization is on the modeling spectrum, modeling your applications will certainly improve your ability to deliver high-quality, resilient, and useful software systems.

They can help pave the road to compliance

The Importance of Industry Standards

BY BRIAN SENNETT

Brian Sennett is a senior consultant at Lighthouse Computer Services in Lincoln, RI. He is a frequent speaker on how technology can ease the burden of complying with new regulatory issues.

Hundreds of new industry-specific mandates, such as the Sarbanes-Oxley Act and privacy requirements based on the Health Insurance Portability and Accountability Act, are either in place now or scheduled to take effect in the near future. While complying with these mandates may seem daunting, businesses actually now have an opportunity to make improvements that go beyond mere compliance to increase the efficiency and predictability of their operations.

Unfortunately, many organizations don't view compliance with optimism, and it's easy to see why. Compliance means managing and interpreting large amounts of information from disparate sources. It may require significant changes in technology infrastructures, financial reporting, records management and retention, and risk management practices. AMR Research estimates that spending on Sarbanes-Oxley compliance in 2005 could reach almost $6 billion.

But consider the upside. Accurate, real-time records and content management systems do more than help companies address the requirements of complex regulation. They can spur productivity, enhance customer service, and boost return on technology investments.

Compliance Help from SOA

New regulations mean that businesses may have to invest in improving their data management. They can take an ad hoc tactical approach and deal with individual regulations as they come along or seize the opportunity to improve overall business operations in the process. The tactical approach is tedious, expensive, and time-consuming. Ad hoc techniques don't fully utilize - or gain insight from - company information on-demand.

Web Services, which is software that connects application and data regardless of underlying technologies, consist of a set of industry-standard technologies that can ease the burden of regulatory compliance. An IT infrastructure composed of collections of reusable Web Services to connect data from various sources - both inside the enterprise and outside at customer, partner, and supplier locations - to solve specific business problems is often referred to as a Service Oriented Architecture (SOA).

So what is an SOA and why is it invaluable?

At its simplest, an SOA involves having common business processes available in a central repository for use and reuse, all in a secure well-managed environment. An SOA provides an enterprise with the flexibility to take elements of business processes in the underlying IT infrastructure and reuse them to address changing business priorities. Previously, when a company needed to change a business process, like complying with a new regulatory directive, the IT department would need months to adapt silo'd manually coded connections to move the data in a way that would assure compliance. The loosely coupled connections common to an SOA create a flexible IT infrastructure that can cut the time to implement those same new business process to days.

Gartner predicts that by 2008 more than 60% of enterprises will use SOA as the "guiding principle" for IT infrastructures that support critical applications and processes. Another study found that 75% of companies plan to start investing in SOAs in the next year.

An SOA relies on industry standards, which provide a layer of instant integration capability that lets all types of software and hardware work together and share data. The SOA approach itself is a significant change from the traditional IT model. Instead of structuring applications based on functions,

components, and objects, the SOA approach comes down to how a company can actually structure applications around services. How each SOA works depends on the requirements of the business deploying it, since it's the business functions in the applications that are being integrated. For example, if your company's IT infrastructure follows an SOA approach, it will include software applications that are exposed through the Internet that can be made available to all the customers, partners, and others you do business with. Adopting an SOA can remedy compliance issues as well as evolving IT infrastructure to embrace technology advances as they happen. To do this, you'll need the right technology in place.

SOA isn't a product, and it isn't a platform; it's an architectural approach. There's no one set of SOA blueprints to follow or SOA product suites required for a company to build an SOA. If your company wants to build an SOA, you first need to assess the technology products you're using and determine if they are based on industry standards that support Web Services. It's essential to have productivity tools that support Web Services and SOA as well as to support the compliance initiatives that your organization may be dealing with. While some standards are open source, it's wise to lean toward the vendors that contribute to standards bodies and work on developing Web Services specifications.

Every organization should have a server foundation that's scalable, secure, and adaptable to changes in business and technology. While most organizations have already selected a J2EE foundation layer, those that haven't must think about what will be the foundation of their business needs today and business growth tomorrow. Some of the largest and most trusted vendors are IBM, BEA Systems, and Oracle. According to both Gartner and IDC, IBM is the most popular choice for application servers, integration servers, and portal servers. BEA typically does well in the application server market, but lately there's been uncertainty about its long-term viability given the drop in its market share, a steady stream of executive departures, and continued speculation that it's an acquisition target. Oracle is a solid database vendor, but it isn't best of breed for application/IT infrastructure, and the company will need to deploy many resources this year integrating the recently acquired PeopleSoft.

"While you're busy complying with the stiff new regulations, you might as well improve your overall business operations in the process"

The next step your company should take toward establishing an SOA is to map out a blueprint for a customized architecture, one that's going to best serve not only your business needs but also the needs of the parties you conduct business with. Knowing and understanding business processes is just as important as pure technology in successfully deploying an SOA, so it's essential to your SOA investment to select systems integrators that you trust and that know your businesses. Industry standards that support Web Services must be integrated with industry semantics, which vary by vertical industry. Systems integrators teaming with partners with vast vertical industry expertise, like IBM's vertical market strategy for products, could add greater value for customers deploying an SOA.

With an SOA, IT will be tightly integrated with both business objectives and compliance regulations, the government can be assured that mandates are being adhered to, customers will be more satisfied, projects can be delivered faster and cheaper, costs can be reduced, and applications can be extended out to partners and customers.

YOUR TESTING SUCKS.

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

LISA from iTKO.

Because everyone should own quality.

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

We understand.

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

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

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

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

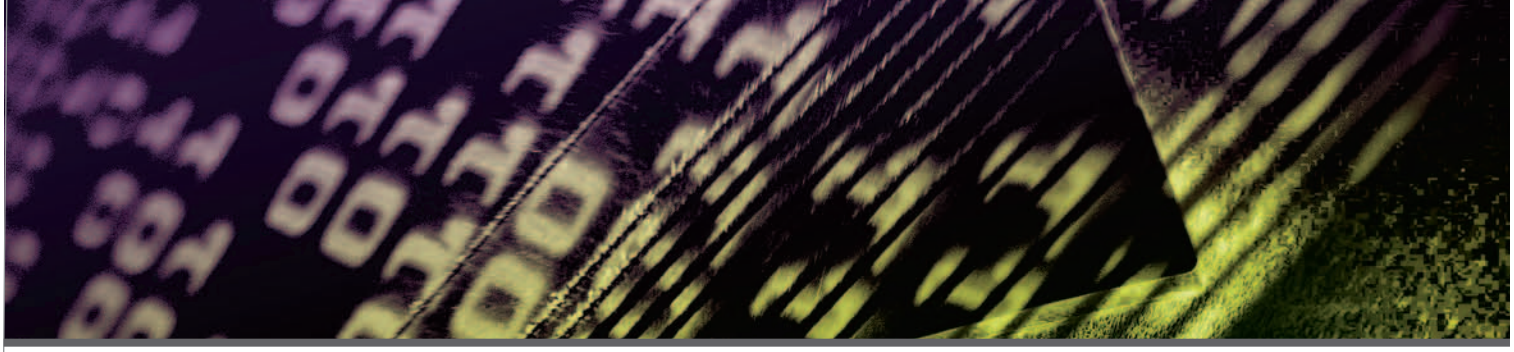
LISA is different.

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

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

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

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



Adding Logging to a WebSphere Portlet

Three ways to do it: good, better, and best

BY FRED SIMMONS & JENG YOONG TAN

Adding logging to portlet code is usually an afterthought during the development cycle. The issue of logging usually only comes up after a portlet has been deployed in production and an error occurs. Then you hear, "Well, the problem didn't happen during testing."

Fred Simmons is a software services consultant for WebSphere at IBM. fredsid@us.ibm.com

Jeng Yoong Tan is a software services consultant for WebSphere at IBM. jtan@us.ibm.com

This paper steps you through the three different logging mechanisms available for portlets. The first demonstrates the WebSphere Portal's built-in logging mechanism. The second is a roll-your-own version of the base WebSphere Portal logging. The third shows you how to add Log4j to your portlet.

Add WebSphere PortletLog to a Basic Portlet

Start by creating a basic portlet using the IBM Rational Software Development Platform. All of the examples in this article should be run in the J2EE perspective. Using the navigation bar, go to **File=>New=>Other=>Portlet Project**. Click **Next**. Enter the project name *Test Log*. Press the **Next** button. Select **Basic Portlet**. Press the **Finish** button.

Using the native WebSphere *PortletLog* logging for your portlet development has a few drawbacks. So you understand both its strengths and weaknesses, you can try this logging feature with the *Test Log* project.

Open *TestLogPortlet.java* and add the declaration for the *PortletLog* and *sClassName*. Replace the *init()* **[A]**, *doView* () **[B]** and *actionPerformed()* **[C]** methods with the code shown in Listing 1 at the end of this article. Remove the annotations **[...]** from the listing before you compile.

Besides instantiating a reference to the *PortletLog* **[D]** in *init()*, the logging methods *info()* **[E]** and *error()* **[F]** are added to both *doView()* and *actionPerformed()*. To prevent a negative impact on performance always check to see if logging is enabled using

```
if (logger.isInfoEnabled()) ...
```

before invoking *info()*. Also limit the number of *info()* calls to significant events such as method entry and exit to prevent the portlet from spending more time tracing than actually performing its function. Since *error()* is only invoked under extraordinary circumstances, you don't need to check if error logging is enabled before invoking this method.

Logging in the WebSphere Portal Server consists of three distinct levels and one all-encompassing level (all):

- low
- medium
- high
- all

Each level can be assigned the value *enabled* or *disabled* to determine which logging method calls are written to the log. The first three levels map to common logging methods. If the low level is enabled then logging from all the logging methods will display. The medium

LEVEL ASSIGNMENT	DEFINITION
low=enabled	Log messages associated with either *info()*, *warn()* or *error()* methods are written to the log.
medium=enabled	Log messages from *warn()* and *error()* methods will be enabled.
high=enabled	Log messages associated with the *error()* method are written to the log.
all=enabled	All logging is enabled
all=disabled	All logging is disabled

TABLE 1: LOGGING LEVEL ASSIGNMENTS AND DEFINITIONS USING *PORTLETLOG*

level means that logging from the *warn()* and *error()* methods will be displayed. Only the *error()* method will display messages if the high level is activated.

Although *PortletLog* supports different logging levels, the best rule for the *PortletLog* tracing is to keep the assignments simple. Limit your level assignments to *all=enabled or all=disabled*.

Enabling PortletLog Tracing Using the Rational Software Development Platform

Test the *Test Log* portlet in IBM Rational Software Development Platform. Right click on *Test Log* and choose **Run=>Run on Server.** Choose **Manually define a server** and **WebSphere Portal v5.0 Test Environment**. Select the **Set server** as project default checkbox. Press **Next**.

Notice the port number. The portal login page for this test environment will have the address http://localhost:9081/wps/myportal. Click **Next**.

Add *Test LogEAR* to the list of **Configured projects.**

The WebSphere Portal Server test environment will display the portlet. In the **Servers** view, right click on the **WebSphere Portal v5.0 Test Environment**. Press **Stop**. After the server stops, right click on the server and select **Open**. Choose the **Portal Options** tab. Select the **Enable base portlets for portal administration and customization** and **Enable console logging** checkboxes. Press **Ctrl-S** to save your configuration changes and close the server configuration panel.

Right click on the *Test Log* portlet project again and choose **Run=>Run on server**.

Due to the design of *PortletLog* tracing, you need the administration console to enable portlet logging. Once the portlet renders, navigate to **Administration=>Portal Analysis=>Enable Tracing**. You enable tracing by adding the line *org.apache.jetspeed.portlet.PortletLog=all=enabled* to the **Append these trace settings** field. Press the plus (+) button. Log out of the portal and then logon. After the page with the *Test Log* portlet displays, press the **Submit** button and select the **Console** tab. You should see messages similar to those shown in Figure 5.

Results of Using PortletLog Logging

Notice that in Figure 5 all log entries generated by the *PortletLog* are prefaced with *org.apache.jetspeed.portlet.PortletLog*. It's important that any log messages include a unique string. In the example above, the logging text for both the *info()* **[E]** and *error()* **[F]** methods include the *sClassName* variable in addition to the method name. Also, if all your portlets use the *PortletLog* methods, then logging for all the portlets in your portal will be enabled. This can be a nuisance if you're just trying to solve a problem in a single portlet.

The benefit of using the *PortletLog* is that it's simple. It's included in the base portlet class. You can get a reference to the *PortletLog* using:

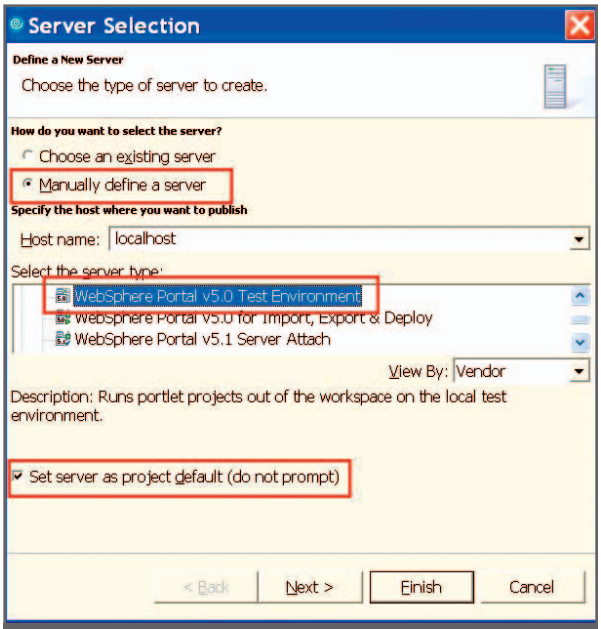


FIG 1: DEFINE THE WEBSPHERE PORTAL TEST ENVIRONMENT

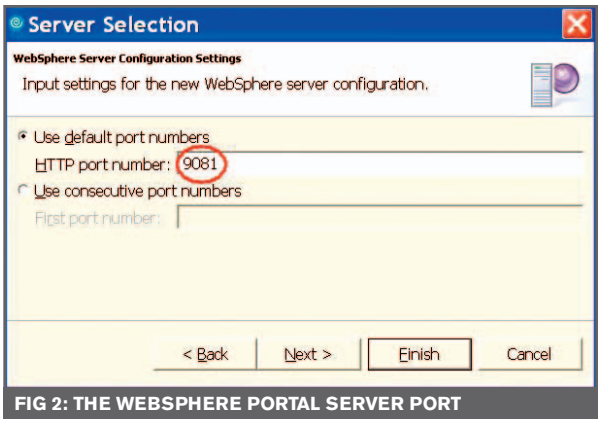


FIG 2: THE WEBSPHERE PORTAL SERVER PORT

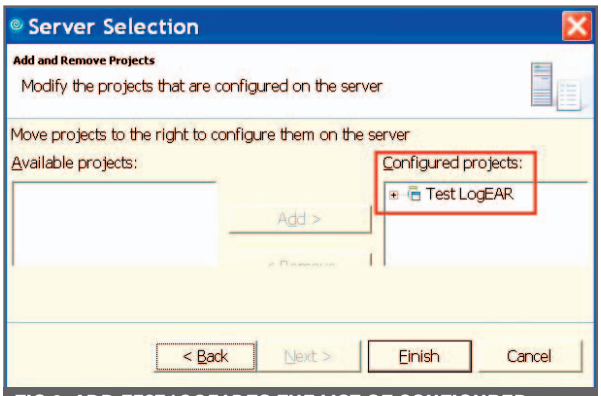


FIG 3: ADD *TEST LOGEAR* TO THE LIST OF CONFIGURED PROJECTS

- *portletConfig.getContext().getLog()*, or
- *getPortletLog()*, which is available in the *PortletAdapter*.

Adding the *PortletLog* methods are better than not having any logging in the portlet but it's definitely not the best method.

LEVEL ASSIGNMENT	WHICH FUNCTIONS WILL LOG FROM *LOGGINGUTILS.JAVA*
low=enabled	This will enable all log messages from *debug()*, *entry()*, *exit()*, *warn()*, *info()* and *error()*methods.
medium=enabled	Log messages from *warn()*, *info()* and *error()* methods.
high=enabled	Log messages associated with the *error()* method are written to the log.
all=disabled	All logging methods are disabled.

TABLE 2: LOGGING LEVEL ASSIGNMENTS AND DEFINITIONS USING *LOGGINGUTILS*

Roll-Your-Own WebSphere Portlet Logging

The basis of the logging mechanism used in the *PortletLog* is WebSphere Portal's *LogManager* and *Logger* classes. By removing the *PortletLog* class layer you can increase the flexibility of your portlet logging. Creating your own instance of the *Logger* in each class lets you associate the source of the logging with your portlet class name and not the *PortletLog* class name.

The following example demonstrates the *LogManager* and *Logger* classes. Create a new basic portlet using the Rational Software Development Platform. From the menu navigate to **File=>New=>Other=>Portlet Project**. Click **Next**. Enter the project name *Test Log WPS*. Select **Basic Portlet**. Press the **Finish** button.

Make the following changes to *TestLogWPSPortlet.java*. First, add the following two import statements.

```
import com.ibm.wps.logging.LogManager;
import com.ibm.wps.logging.Logger;
```

Then create an instance using the name of the portlet class, *TestLogWPSPortlet.class*. All entries in the log file will display this class name as the source in the message.

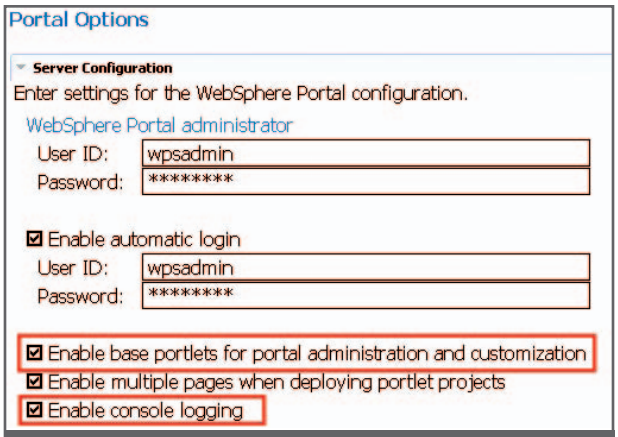


FIG 4: ENABLE BASE PORTLETS FOR PORTAL ADMINISTRATION AND CUSTOMIZATION AND CONSOLE LOGGING

```
A org.apache.jetspeed.portlet.PortletLog info test_log.TestLogPortlet Exit doView()
A org.apache.jetspeed.portlet.PortletLog info test_log.TestLogPortlet actionPerformed() called
A org.apache.jetspeed.portlet.PortletLog info test_log.TestLogPortlet Entry doView()
A org.apache.jetspeed.portlet.PortletLog info test_log.TestLogPortlet Exit doView()
```

FIG 5: CONSOLE WITH *PORTLETLOG* MESSAGES

```
// This is the portlet logger class instantiated
    for this particular class
private static Logger logger =
        LogManager.getLogManager().getLogger
          (TestLogWPSPortlet.class);
```

Replace the *doView()* **[G]** and *actionPerformed()* **[H]** methods with the one shown in Listing 2 at the end of this article. The logging functions added to these methods *entry()* **[I]**, *exit()* **[J]**, *info()* **[K]** and *error()* **[L]** are found in the utility class *LoggingUtils* included with this article. Remove the annotations **[...]** from the listings before you compile.

The *LoggingUtils* class contains the generic logging functions. To create this class in your project right click on *test_log_wps* and select **New=>Class**. The name of the class is *LoggingUtils*. Both the **public** and **abstract** modifiers checkboxes should be selected. Click on the **Finish** button. Insert the code for *LoggingUtils* as shown in Listing 3 at the end of this article.

The methods in the *LoggingUtils* class correlate with the log levels supported by the WebSphere Portal Server. Below is a table showing the level and the logging functions enabled at that level.

Enabling Roll-Your-Own Portlet Tracing in the Rational Software Development Platform

Test the portlet logging in *Test Log WPS* using the IBM Rational Software Development Platform prior to deploying it on your production portal server.

Right click on the *Test Log WPS* portlet project again and choose **Run=>Run on server**. Choose **Manually define a server** and **WebSphere Portal v5.0 Test Environment**. Select the **Set server as project default** checkbox. Press **Next**. On the next panel add the *Test Log WPSEAR* to the **Configured projects** and remove *Test LogEAR*. Press the **Finish** button.

The *Test Log WPS* portlet will display. You can enable all the tracing by adding the line *test_log_wps.*=low=enabled* to **Administration=>Portal Analysis=>Enable Tracing**. Paste that line into the **Append these trace settings** field. Then press the plus (+) button. Log out and then logon again using the user id *wpsadmin* and the password *wpsadmin*.

Press the **Submit** button in the *Test Log WPS portlet*. The output of the tracing won't display in the **Console** view. In the portal trace file **<Rational Install Dir>***runtimes\portal_v50\log\trace.log*, you will see:

```
[3/23/05 21:04:06:180 EST] 55ffc21b PortalServer
  A test_log_wps.TestLogWPSPortlet info info in
  actionPerformed()
[3/23/05 21:04:06:210 EST] 55ffc21b PortalServer
  e test_log_wps.TestLogWPSPortlet entry Entry
  doView()
[3/23/05 21:04:06:220 EST] 55ffc21b PortalServer
  e test_log_wps.TestLogWPSPortlet exit Exit doView()
```

Test the logging capabilities by returning to the **Enable**

[Engage and Explore]
the technologies, solutions and applications that are driving today's **Web services** initiatives and strategies...

www.sys-con.com/edge2005

web services EDGE conference&expo
FALL SERIES

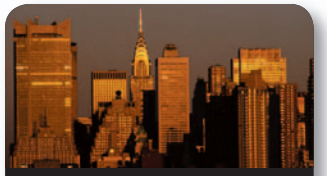
CALL FOR PAPERS NOW OPEN!*

Coming to a City Near You ►

Web Services Edge
Fall Conference Series

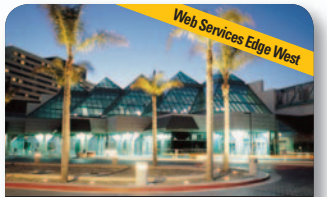
The Westin Washington, D.C.
Washington, D.C.
September 7-8, 2005

1



Helmsley Hotel
New York, NY
September 19-20, 2005

2



The Westin Santa Clara Convention Center
Santa Clara, CA
October 24-25, 2005

3

3 Dynamic Conference Programs
Targeting Major Industry Markets

20+ seminars within 5 tracks will address the hottest topics & issues:

- Web Services: The Benefits and Challenges
- Web Services Security
- SOA (Service-Oriented Architecture) and ESB (Enterprise Service Bus) Strategies
- Interoperability, Incremental Integration, & Open Source
- The Management Process in Developing a Web Services Strategy

Program Features:

- Keynotes
- Tutorials
- Panel Discussions

Why Attend:

- Improve the return on your technology investment
- Develop & sharpen your strategy and identify key action steps
- Find new ways to reach and impress customers with Web services
- Maximize the power of your enterprise
- Protect your business from security threats
- Assess Web services as a viable option

Attention Exhibitors:
- An Exhibit-Forum will display leading Web services products, services, and solutions

Register Today! www.SYS-CON.com/Edge2005

Sponsored by

WebServices JOURNAL · XML JOURNAL · .NET DEVELOPER'S JOURNAL · eclipse developer's journal · WebSphere JOURNAL · information STORAGE+SECURITY journal · wldj · JDJ · LINUXWORLD MAGAZINE · MX developer's journal · asp.netPRO · SD Times · CoDe · Software Test & Performance

*Call for Papers email: grisha@sys-con.com

For Exhibit and Sponsorship Information ► Call 201 802-3066

Produced by SYS-CON EVENTS

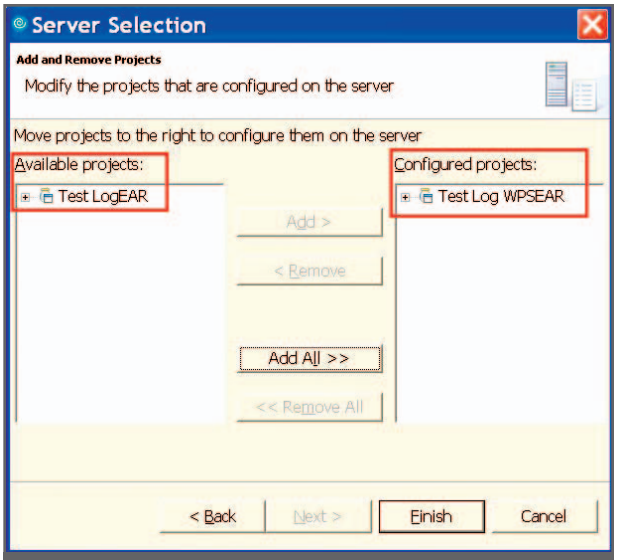


FIG 6: ADD *TEST LOG WPSEAR* TO THE LIST OF CONFIGURED PROJECTS

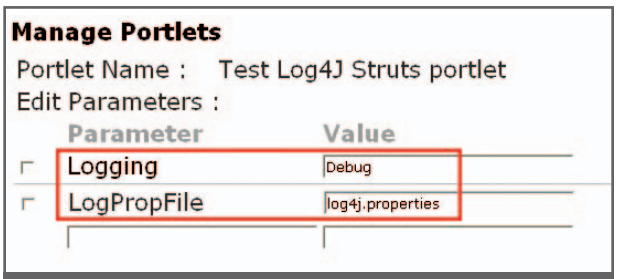


FIG 7: MODIFY PORTLET SETTINGS FOR THE *TEST LOG4J STRUTS* PORTLET

Tracing page under **Administration** and modifying the trace string to *test_log_wps.*=medium=enabled*. This alters the logging to let the *info()* method call in *actionPerformed()*be displayed, not the entry and exit ones. Log out and logon. Press the **Submit** button. In **<Rational Install Dir>**\ *runtimes\portal_v50\log\trace.log* you will have a single trace entry:

```
[3/23/05 21:29:01:861 EST] 705dc215 PortalServer
  A test_log_wps.TestLogWPSPortlet info info in
  actionPerformed()
```

If you set the trace string to *test_log_wps.*=high=enabled* then none of the trace messages in the portlet are displayed unless an exception occurred.

Results of Roll-Your-Own Portlet Logging

This version of the portlet logger lets you enable tracing for a specific portlet or portlets. In fact you could limit the tracing to a single class. For example, *test_log_wps.TestLog WPSPortlet=low=enabled* will only allow logging from the *TestLogWPSPortlet* class. This improves the portlet logging in the base WebSphere *PortletLog*.

The added benefit of using the roll-your-own logging is that:

1. the WebSphere Portal Adminstration lets you easily modify the tracing and
2. beyond the addition of the *LoggingUtils* class it doesn't require any modifications to the base portlet.

Add Log4j to a Basic Portlet

If you aren't satisfied with the native WebSphere Portal logging capabilities, you can easily add the Jakarta Log4j package to your portlet. This section will give you a general overview of Log4j and how to integrate it.

As an example, create a new basic portlet using IBM Rational Software Development Platform. From the menu navigate to **File=>New=>Other=>Portlet Project**. Enter the project name *Test Log4J*. Press **Next**. Make sure Basic Portlet is selected and then press the **Finish** button.

Select the project name in the **Navigator** view, right click to bring up the context menu and choose **Import...**. Select **File System**. In the **From Directory** entry field, browse to the directory where your *log4j jar* file is located. For the **To Directory** entry field browse to the *Test Log4J/WebContent/ WEB-INF/lib* directory. Press the **Finish** button. The *log4j jar* file will be imported into your WAR file.

Edit the *Test4JLogPortlet.java* file. At the top add the import for Log4j and statically declare the logger **[M]**. If the logger does not exist it will be created for you. There will always be one instance of this logger in your Java Virtual Machine. If you wanted to log from other modules in the portlet application you could declare the logger at the top of each of those classes. Remove the annotations **[...]** from the listing before you compile.

```
import org.apache.log4j.Logger;

public class TestLog4JPortlet extends PortletAdapter
  implements ActionListener {

[M]private final static Logger logger =
  Logger.getLogger(test_log4j.TestLog4JPortlet.class);
```

Replace the *doView()* **[N]** and *actionPerformed()* **[O]** methods with the code found in Listing 4 at the end of this article, which includes Log4j logging.

To enable logging you must create a *log4j.properties* file under the *Java Resources/JavaSource* directory.

From the menu bar choose **File=>New=>Other=>Simple=>File**. Press **Next**. The parent folder should be *Test Log4J/ JavaSource*. Enter *log4j.properties* for the file name. For purposes of this example, the logging level in *log4j.properties* is set to debug so that you can see the debug messages.

Paste the text in Listing 5 at the end of this article into the *log4j.properties* file and modify it to suit your environment. Remove the annotations, **[...]**.

Test the Log4j Tracing in the Rational Software Development Platform

Run the *Test Log4J* portlet using the IBM Rational Software Development Platform. Right click on the *Test Log4J* portlet project. Choose **Run=>Run on server**. Select

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

Manually define a server and **WebSphere Portal v5.0 Test Environment**. Also select the **Set server as project default** checkbox. Press **Next**. On the next panel add the *Test Log4JEAR* to the **Configured projects**. Press the **Finish** button.

If you run this portlet using the IBM Rational Software Development Platform environment, the log file *mytestportlet.log* **[P]** will be located in the **<Rational Install Dir>** directory. In this example the logging is made to the *mytestportlet.log* file using the RollingFileAppender **[Q]**. Press the portlet **Submit** button. Here is a sample of the resulting output:

```
[2005-03-23 21:59:12,414] Servlet.Engine.Transports :
  2 test_log4j.TestLog4JPortlet DEBUG - Exit
  actionPerformed()
[2005-03-23 21:59:12,495] Servlet.Engine.Transports :
  2 test_log4j.TestLog4JPortlet DEBUG - Entry
  doView()
[2005-03-23 21:59:12,505] Servlet.Engine.Transports :
  2 test_log4j.TestLog4JPortlet DEBUG - Exit doView()
```

Log4j follows a parent-child relationship pattern. A root logger is provided by default. All of the loggers defined in the portlets will ultimately inherit from the root logger. A parent-child relationship is followed by using a name pattern. For the class *TestLog4JPortlet* in package *test_log4j* instantiate a logger with *test_log4j.TestLog4JPortlet.class*. Any packages created under *test_log4j* will be its children. The only ancestor of the *test_log4j.TestLog4JPortlet* logger is the root logger provided by Log4j.

Adding Log4j to a Struts Portlet

Start the development of the struts portlet from the main menu. Navigate to **File=>New=>Other=>Portlet Project**. Press **Next**. Enter the project name *Test Log4J Struts*. Click **Next**. Choose the **Struts portlet** radio button. Press **Finish**. Expand the project and import the *log4j jar* file into the *WebContent/WEB-INF/lib* directory.

To dynamically change the Log4j logging in struts you must extend the base struts portlet, *WpsStrutsPortlet*. Under Java Resources for the struts portlet application project, add a package named *com.ibm.strutsextend*. In this package create a java class named *TestLog4JStruts*. Make its superclass *com.ibm.wps.portlets.struts.WpsStrutsPortlet* **[R]**. Replace the code inside this class with the code in Listing 6 at the end of this article. Remove the annotation, **[R]**.

The logging parameters in order of increasing restriction are as follows:

- **All:** This is the same as Debug. It will display all the messages sent to Log4j.
- **Debug:** Used to display debugging messages that aren't usually enabled in production.
- **Info:** The "verbose" mode of the application.
- **Warn:** Indicates a potential problem.
- **Error:** A serious problem has occurred. The application may continue to function.
- **Fatal:** A severe error occurred that will shut the program down.
- **Off:** Absolutely, positively no message logging at all.

The Logger will only output messages that are at a level greater than or equal to its set logging level. For example, setting the logger level to Warn will only let messages at the Warn, Error, or Fatal levels be logged. If a class doesn't set the level of a logger, then it will inherit the level of the closest ancestor. If a logger is created in the package *com.ibm.strutsextend* and no level is set for it, it will inherit the level of the logger created in *com.ibm*. If no logger was created in the *com.ibm* or *com* packages, then the logger created in *com.ibm.strutsextend* will inherit the level of the root logger. The root logger is always instantiated and available and is assigned the level Debug.

In the portlet's *initConcrete()* method, you can set the logging level and the Log4j properties file.

Add the config parameters shown in Listing 7 at the end of this article to *WebContent/WEB-INF/portlet.xml*. The logging parameter supports the values: Debug, Info, Warn, Error, Fatal, Off, and All. If you omit the LogPropFile entry you will get the default one, *log4j.properties*, found in the Java Resources directory.

In *WebContent/WEB-INF/web.xml* replace the *com.ibm.wps.portlets.struts.WpsStrutsPortlet* servlet class with the *com.ibm.strutsextend.TestLog4JStruts* class.

```
<servlet-class>com.ibm.strutsextend.TestLog4JStruts</
  servlet-class>
```

Also remove all the entries in the *welcome-file-list* except *index.jsp* in *web.xml*.

```
<welcome-file-list><welcome-file>index.jsp</
  welcome-file></welcome-file-list>
```

Add the *log4j.properties* file under the *Java Resources/JavaSource* directory.

You now have a basis for building a struts portlet with Log4j logging that can be dynamically configured.

You can even log from inside of a JSP. Right click on *WebContent*. Select **New=>JSP File**. Enter *index.jsp* as the name and select the JSP fragment checkbox. Insert the code in Listing 9 in *index.jsp*. The *index.jsp* file references the logger defined in the base struts portlet class.

```
<%@ taglib uri="/WEB-INF/struts-html.tld"
  prefix="html" %>
<%@ taglib uri="/WEB-INF/struts-bean.tld"
  prefix="bean" %>
<%@ taglib uri="/WEB-INF/tld/portlet.tld"
  prefix="portletAPI" %>
<%
 // use the logger from the portlet class
 final org.apache.log4j.Logger logger =
        org.apache.log4j.Logger.getLogger(com.
         ibm.strutsextend.TestLog4JStruts.class);

 // log even in a jsp.
 logger.debug("Loading index.jsp");
%>
<p>Hello logger</p>
```

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

Run the struts portlet in the **WebSphere Portal v5.0 Test Environment**. The output log file, *mytestportlet.log* will contain entries similar to this:

```
[2005-03-25 11:30:45,764] Servlet.Engine.Transports :
  1 com.ibm.strutsextend.TestLog4JStruts DEBUG -
  initConcrete()
[2005-03-25 11:30:45,764] Servlet.Engine.Transports :
  1 com.ibm.strutsextend.TestLog4JStruts DEBUG -
  Entry setLevel Debug
[2005-03-25 11:30:45,764] Servlet.Engine.Transports :
  1 com.ibm.strutsextend.TestLog4JStruts DEBUG -
  Exit setLevel debug
[2005-03-25 11:30:45,764] Servlet.Engine.Transports :
  1 com.ibm.strutsextend.TestLog4JStruts DEBUG -
  New Property file : log4j.properties
[2005-03-25 11:30:46,925] Servlet.Engine.Transports :
  1 com.ibm.strutsextend.TestLog4JStruts DEBUG -
  Loading index.jsp
```

You can change the log level or Log4j configuration file for *Test Log4J Struts* by navigating to **Administration=>Portlets=>Manage Portlets**. Select the entry for *Test Log4J Struts* and press the **Modify parameters** button. The values for the *Logging* and *LogPropFile* are displayed. Change the *Logging* to **Error**. Press **Save**. Log off from the portlet and logon using user id *wpsadmin* and password *wpsadmin*. The logging to *mytestportlet.log* will stop.

Summary

This article reviews three different kinds of logging available for a portlet. The first was the use of the *PortletLog*. The second was a roll-your-own version of the base WebSphere Portal logging classes. The third involved directly invoking the Log4j framework.

The simplest form of portlet logging is provided in the WebSphere *PortletLog*. Although it's easy to invoke inside a portlet, the major drawback is that it provides no granularity as to which portlet or portlet class is being traced, but it's better than having no logging in your code.

If you want to use the built-in logging of WebSphere Portal but would like to avoid the pitfalls of the *PortletLog*, then use the roll-your-own logging approach described. It provides you with ease of use while maintaining flexibility in determining which portlets and classes are being traced.

A different approach for portlet logging is invoking Log4j directly. Log4j is an external logging library that provides a flexible framework. Many Java programmers prefer to use Log4j, which can be added to any existing portlet by the following steps:

- statically instantiate a *Logger* at the top of your class file;
- add a *log4j.properties* to your Java Resources directory;
- import the *log4j jar* file into your portlet.

Your portlet can be enhanced using the watch capabilities of Log4j, which lets you easily change the portlet Log4j parameters.

Be prepared. Don't be caught with a portlet without logging capabilities in your production WebSphere Portal Server.

Downloads

Although you can build all of the portlets in the article with just the IBM Rational Software Development Platform, the completed portlets WAR files are available in this zip file.

Resources

Download a trial version of the *IBM Rational Software Development Platform*.

Get the latest version of Log4j, an open source logging framework, at the Log4j Web site.

Share your questions and views on this article with other readers in the *WebSphere Portal Server forum*.

LISTING 1 • CHANGES MADE TO *TESTLOGPORTLET.JAVA* TO ENABLE *PORTLETLOG*

```
//  the WebSphere Portlet logger
//  note that to enable this you must use :
//  org.apache.jetspeed.portlet.PortletLog=all=enabled
private org.apache.jetspeed.portlet.PortletLog logger = null;

/**
  *  Class name for tracing

  */

private String sClassName = ((Object) this).getClass().getName();

/**
 * @see org.apache.jetspeed.portlet.Portlet#init(PortletConfig)
 */
[A] public void init(PortletConfig portletConfig) throws
    UnavailableException {
      super.init(portletConfig);
      // set up the logger.
[D]       logger = portletConfig.getContext().getLog();
}
```

```
/**
 * @see org.apache.jetspeed.portlet.PortletAdapter#doView
    (PortletRequest,
 *       PortletResponse)
 */
[B] public void doView(PortletRequest request, PortletResponse
    response)
          throws PortletException, IOException {

  try {
      if (logger.isInfoEnabled()) {
[E]           logger.info(sClassName + " Entry doView()");
      }

      // Check if portlet session exists
      TestLogPortletSessionBean sessionBean =
        getSessionBean(request);
      if (sessionBean == null) {
          response.getWriter().println("<b>NO PORTLET SESSION
            YET</b>");
          return;
      }
      // Invoke the JSP to render
      getPortletConfig().getContext().include(
            VIEW_JSP + getJspExtension(request), request,
```

```
  response);
  } catch (Throwable e) {
      if (e instanceof PortletException) {
[F]           logger.error(sClassName + " doView() Portal Exception : "
                  + ((PortletException) e).getMessage());
          throw (PortletException) e;
      }
      if (e instanceof IOException) {
[F]           logger.error(sClassName + " doView()I/O Exception :
                  " + e.toString());
          throw (IOException) e;
      }
[F]       logger.error(sClassName + " doView() Exception :
            " + e.toString());
  }

  if (logger.isInfoEnabled()) {
[E]       logger.info(sClassName + " Exit doView()");
  }

}

/**
 * @see org.apache.jetspeed.portlet.event.ActionListener#action
     Performed(ActionEvent)
 */
[C] public void actionPerformed(ActionEvent event) throws
     PortletException {
  if (logger.isInfoEnabled()) {
[E]           logger.info(sClassName + " actionPerformed() called");
  }
  try {
      // ActionEvent handler
      String actionString = event.getActionString();
      PortletRequest request = event.getRequest();
      // Add action string handler here
      TestLogPortletSessionBean sessionBean =
       getSessionBean(request);

      if (FORM_ACTION.equals(actionString)) {
          // Set form text in the session bean
          sessionBean.setFormText(request.getParameter(TEXT));
      }
  } catch (Throwable e) {
      if (e instanceof PortletException) {
[F]           logger.error(sClassName + " actionPerformed()
                Portal Exception : "
                  + ((PortletException) e).getMessage());
          throw (PortletException) e;
      }

[F]       logger.error(sClassName + " actionPerformed() Exception : "
+ e.toString());

  }
}
```

LISTING 2 • CHANGES TO THE *DOVIEW()* AND *ACTIONPERFORMED()* METHODS IN *TESTLOGWPSPORTLET.JAVA*

```
/**
 * @see org.apache.jetspeed.portlet.PortletAdapter#doView
     (PortletRequest,
 *       PortletResponse)
 */
[G] public void doView(PortletRequest request, PortletResponse
     response)
          throws PortletException, IOException {
      try {
          // entry log is limited to debug mode
          // the other choices would be TRACE_LOW, TRACE_MEDIUM,
TRACE_HIGH
          if (LoggingUtils.isDebugEnabled(logger)) {
[I]           LoggingUtils.entry(logger, "doView()");

          }

          // Check if portlet session exists
          TestLogWPSPortletSessionBean sessionBean =
            getSessionBean(request);
          if (sessionBean == null) {
              response.getWriter().println("<b>NO PORTLET SESSION
                YET</b>");
              return;
          }

          // Invoke the JSP to render
          getPortletConfig().getContext().include(
                  VIEW_JSP + getJspExtension(request), request,
response);
      } catch (Throwable e) {
          if (e instanceof PortletException) {
              // log error is performed when the trace level is set
to
              // high or all
[L]               LoggingUtils.error(logger,
                   "error in doView() throw portlet exc", e);
                    throw (PortletException) e;
          }
          if (e instanceof IOException) {
              // log error is performed when the trace level is
                set to
              // high or all
[L]               LoggingUtils.error(logger,
              "error in doView() throw i/o exc", e);
              throw (IOException) e;
          }
          // log error is performed when the trace level is set to
          // high or all
[L]           LoggingUtils.error(logger, "error in doView()", e);
      }
      //
      // exit log is limited to low mode
      // the other choices would be TRACE_LOW, TRACE_MEDIUM,
       TRACE_HIGH
      if (LoggingUtils.isDebugEnabled(logger)) {
[J]           LoggingUtils.exit(logger, "doView()");
      }

}

/**
 * @see org.apache.jetspeed..event.ActionListener#actionPerformed
    (ActionEvent)
 */
[H] public void actionPerformed(ActionEvent event) throws
     PortletException {
      // demonstrates the use of info() method.
      // this is logged only for medium mode
      if (LoggingUtils portlet.isInfoEnabled(logger)) {
[K]           LoggingUtils.info(logger, "info in actionPerformed()");
      }
      try {
          // ActionEvent handler
          String actionString = event.getActionString();
          PortletRequest request = event.getRequest();
          // Add action string handler here
          TestLogWPSPortletSessionBean sessionBean =
            getSessionBean(request);

          if (FORM_ACTION.equals(actionString)) {
              // Set form text in the session bean
              sessionBean.setFormText(request.getParameter(TEXT));
          }
      } catch (Throwable e) {
          if (e instanceof PortletException) {
              // log error is performed when the trace level is
                set to
              // high or all
[L]               LoggingUtils.error(logger,
                  "error in actionPerformed() throw portlet exc", e);
                   throw (PortletException) e;
          }

          // log error is performed when the trace level is set to
          // high or all
[L]           LoggingUtils.error(logger, "error in
                actionPerformed()", e);
      }
}
```

For Listings 3-8, go to www.sys-con/websphere/source.cfm

Interview with Doug Wilson

IBM's Lotus Division CTO Speaks Out

WebSphere Journal recently interviewed Doug Wilson, Distinguished Engineer and CTO of IBM's Lotus Software Division in Westford, Massachusetts.

BY ROGER STRUCKHOFF

Roger Strukhoff, editor-in-chief of *WebSphere Journal*, is West Coast Bureau Chief for the SYS-CON News Desk, and President of www.wdva.com. He spent 15 years with Miller Freeman Publications and The International Data Group (IDG), then cofounded CoverOne Media, a custom publishing agency that he sold in 2004. His work has won awards from the American Business Media, Western Press Association, Illinois Press Association, and the Magazine Publishers' Association. Read his blog at www.rssblog.linuxworld.com. Contact him at roger@sys-con.com.

Doug Wilson Speaks About His and Lotus's Background...

WebSphere Journal: Doug, could you please provide our audience with a little bit on your background and what path you took to your current job and responsibilities with the Lotus Software Division.

Doug Wilson: Like many people, my career path is a bit checkered. I started some time ago working at MIT as Director of Systems Operations and Systems Development for Project Athena, which, as you may recall, originated a lot of the UNIX based and X-Window based user interface technology. From there I attempted to work in the venture capital land, started a company to do computer aided mechanical engineering design, worked for a while at Kodak building office automation systems for the legal profession and then joined IBM at that time, Lotus, in 1991, where I worked on versions of 1-2-3.

In particular, I was the chief architect for Windows versions 1-2-3 and Smart Suite which was then being acquired by IBM. I worked on a variety of projects at IBM but many centered around the componentization of functions and Java technology.

WJ: What overall role does Lotus play within IBM today, and specifically within the company on the on-demand initiative?

DW: It's actually quite exciting for Lotus now. Lotus has taken the responsibility, within IBM, is really to build the face of the On-demand operating system. We're responsible for those parts of the IBM portfolio that deal with the interactions between users and systems, the organization and presentation of information. Products like WebSphere Portal are part of the Lotus Portfolio, as you certainly have known in recent months. We've released the Workplace Client Technologies and have the responsibility for marketing and engineering of that product as well. Of course, our history has been in Lotus Notes and Notes Domino, and we continue to be extremely strong there with a large customer base of a hundred million plus users.

WJ: What level of support can corporate buyers, in particular, and users, as well, expect to see from IBM for Lotus and Domino in the coming months or coming years?

DW: I think the first word would be unwavering. Domino is a critical part of the portfolio and obviously it's a critical part of the IBM business. IBM itself has close to 300,000 users on Notes. So along with our own internal interests we continue to drive and work towards the interest of all of our Notes Domino customers. It's also important to note that any system that customers have today is part of every system they will build in the future.

WJ: When you say system, are you referring the ability of Lotus to work across platforms and infrastructures?

DW: Well, certainly it does do that, but more specifically I was suggesting that customers have built up an incredible inventory of business critical systems on Domino, and those systems are always necessary going forward and will be part of the customers' future. And so the Workplace initiatives fully embrace all of that and extend the reach of those systems.

WJ: So you're respecting the vital aspects of Lotus in the Workplace and different vertical markets and applications.

DW: Absolutely. Lotus has always been in the platform business and the tremendous strength in Notes is attributable to a large partner community and we see

that continuing as we enhance the Domino story with the Workplace story.

About Addressing Customer Needs...

WJ: What specific problems are your customers addressing today to Lotus, and how has this evolved over the past few years?

DW: I think our customers are principally concerned with improving the productivity of their employees. A key differentiation in a market can be driven through the productivity of your people. Once you've driven costs out of as many of your systems as you can, one of the few places you have to make substantial differences is in the productivity of the people. That has been the traditional focus of Lotus products, whether for individual productivity with Smart Suite or team productivity which is a focus of the Notes Domino portfolio. Now with the Workplace initiatives focusing on organizational productivity, we are making people work better and more effectively in the context of the business operations that make up a company.

WJ: You were just talking about because you're always checking the feedback from your customers and responding in terms of product development to their increasing needs and expanding and diversified needs. And that's, I guess, reflected in every subsequent release of Domino. Can you be more specific in how you're meeting those current and anticipated customer needs?

DW: Certainly one of the things that we continue to do with the portfolio is to drive cost out of the operation. Domino now is significantly more efficient and capable than it was before. It is more in-tuned and integrated with modern technologies like Web services, and emerging new standards wherever they're appropriate. We continue to assist customers all the time with business value assessments that help them understand how better to utilize those technologies. Many of our customers are utilizing new capabilities and notes and new platforms to achieve significant cost savings through server consolidation which results in, of course, lower cost of ownership both from hardware acquisition and maintenance standpoint, but also in personnel that it takes to manage and administer those systems.

WJ: So you've seen those numbers actually declining as well on the number of quantity of administrators and support that's required?

DW: Yes, and the numbers are significant: factors of three to five in the reduction of the number of CPUs necessary. That's an accumulative effect of better SMP systems and better utilization of those systems by the software itself.

WJ: Would you say these particular factors have accelerated in the last year, given the current state of our economy? Are you more sensitive to total cost of ownership and cost of management and admin? Is that something that you've always been aware of something that you're trying to reduce?

DW: I think it's always been important but certainly in the last couple of years there's been an increasing focus on IT in driving costs out of the system. Actually, though, as the economy has improved lately, we've seen a swing back to more strategic projects and people really looking for ways to get more capability out of their IT systems and not just driving down costs. And that really leads us to look for new opportunities—and that's the most exciting focus of the Workplace portfolio.

We need to really figure out how to do that business more efficiently. We are embarked on really the first phases of what we see as a fundamental change in the way we approach collaboration in computing systems, and almost all business decision operation kinds of things are collaborative. Very few things are done by people in isolation, and are certainly not done outside of a business context that drives those decisions. We see tremendous opportunity in integrating the collaborative systems with the business systems that provide the context that employees need to make decisions that provide the information feeds all in the context of the particular business operation.

About Customer and Marketing Relationships...

WJ: We understand, Doug, that marketing relationships are critical to the success of all software products in environments today. Could you describe for our audience the extent of

Doug Wilson,

CTO, LOTUS PRODUCT DIVISION, IBM

Doug Wilson is an IBM Distinguished Engineer and chief technology officer for the Lotus Product Division of IBM's Software Group. Prior to this he was responsible for architectural strategy for Domino and WebSphere integration, WebSphere Portal Server and WebSphere content manager, and WebSphere Personalization. He invented and managed the design and development of Lotus InfoBus technology, a data exchange technology for Java components that has been accepted for inclusion in JavaSoft's Java Developer's Kit as a Java standard. Doug holds a degree from M.I.T.'s Department of Mechanical Engineering.

douglas_wilson@us.ibm.com

IBM's Lotus marketing program today? And how many partners, for example, and can you offer an example of great customer success stories? Those are things that our audiences are always interested in hearing about.

DW: Well, there are great success stores all over the place. One good story involves a major insurance company using our Workplace Client Technology to drive increased productivity in their organizations, handling claims more quickly, handling claims on site without the employees having to be tethered to the IT systems, and providing significant increases, significant improvements, in policy handling. We have customers with many, both internal- and external-facing portal implementations that are providing new sources of information, aggregation and feed to their employees to get them information more quickly. We have customers deploying our new Workplace services in order to provide low cost messaging and communication services between mobile employees, people who have no desks, but move from kiosk to kiosk.

WJ: And your marketing programs?

DW: What I think you'll see from IBM is a focus that is not as much software-brand-led as it's been in the past, but rather focused on the problem of the IT and business organization as a whole. The On-demand initiative from IBM is one that's really had resonance with our customers. It describes a set of ideals for business in terms of business agility responsiveness, flexibility, which are particularly important to businesses in the current market conditions, as things change quickly, as organizations merge and divest, in-source and outsource their capabilities. These are all things that are representative of the needs for a company, and correspondingly its IT organization needs to be very agile. The On-demand initiatives in IBM are about responding to that and providing IT the tools needed to do that.

WJ: So will the tools and software take kind of a back seat to just providing an overall solution or capability of creating greater agility for an organization?

DW: I wouldn't say a back seat. I would say that they are foundational, Right? That those software systems are necessary, that integration is a key to all of this. If you look at IT spending projects, tremendously large percentage of those projects are oriented at integration kinds of activities. Consolidation and acquisition force those issues and expanding opportunities drive new modalities and new channels for delivery of services.

About Lotus Workplace...

WJ: Nearly a year after its launch last June, how would you define the business opportunities inherent in Lotus Workplace? We've been hearing more and more about Workplace. That's part A of that question, and part B is has a full set of APIs now been released for Workplace or is that something we can anticipate?

DW: To the first question, as with any new portfolio, you expect there to be people out on the leading edge of that who are achieving success. We have those customers and they're doing quite well. I think we have a number our customers, in the thousands now, for the Workplace product portfolio. It's a very broad portfolio as well and encompasses both stand-alone offerings from Web content management to one of the very exciting offerings which is Workplace for Business Controls and Reporting which is very relevant to corporations today who are faced with compliance requirements through the Sarbanes-Oxley Act and the like.

WJ: Are you offering solutions that are more vertical?

DW: We offer the Sarbanes-Oxley solution as a platform in conjunction with partners who bring the domain expertise on regulatory compliance and the templates and patterns necessary for achieving compliance in an organization.

WJ: Highly specialized...

DW: Yes. So again, we, as IBM, specialize in foundational and platform software. We turn very much to our partners to drive the solutions across all of the domains that they work in.

WJ: Are there any particular vertical industries that are quicker to adopt Workplace versus others that you're seeing any particular uptake in one versus the other, or is that something you can share with us?

DW: It's a very good spread, actually. The financial institutions have been quite responsive. We see very high interest in professional services organizations as well, which, of course, you'd expect because of the high degree of collaboration between individuals there. But it's quite evenly spread. IBM has a strong industry focus solutions team and every one of them is actively engaged with the Workplace technology so it's very good.

To the second part of your question, APIs for Workplace. Yes, there is an API for the Workplace collaborative services, and again I would probably stress that Workplace is a high level brand in IBM. There are a number of products under the Workplace umbrella and I suspect the one you were specifically asking about is Workplace Collaborative Services, and yes, there is an API for all of the service functions coming in the 2.5 release, which is imminent.

Part 2 of this interview will appear in the May 2005 issue of WebSphere Journal. *Doug Wilson talks about RFID and retail, Lotus and J2EE, Eclipse, and other topics of interest.*

Senior WebSphere Portal IT Architect Richard Gornitsky

WebSphere Journal: There's quite a bit of development, we understand, going on in the WebSphere Portal space. So we'd like to learn about of how Portal has evolved over the last two years.

Richard Gornitsky: Well, this has been a very exciting time for us. Over the last couple of years, WebSphere Portal was focusing on the traditional portal environment space of aggregating Web pages. People would be looking at our product to support B2E applications in their Intranet environment. Because of all the new features we have been adding to the product for scalability, reliability, and availability, our customers are now looking to WebSphere Portal as an overall presentation framework. So now customers are putting large scale applications, like online banking, on WebSphere Portal.

WJ: So it's been kind of growing on its own in an organic way through the different vertical markets that maybe at the onset you didn't necessarily anticipate.

RG: Exactly. And because of this, we have been motivating our lab people to make many interesting modifications to the product to support these new initiatives. So for instance these new applications have a huge user base that need to access portal sites with hundreds or thousands of pages. New features have been added to support the scalability challenges these applications present.

"Our customers have been critical to the development of portal"

WJ: So it seems that Portal development has really been a collaborative effort on behalf of IBM and your existing and expanding customer base.

RG: Yes, it has. Our customers have been critical to the development of WebSphere Portal. Their input has been instrumental in keeping WebSphere Portal a "best-of-breed" product. My feeling is, and I can clearly tell you that I'm very biased about this, none of our competition can touch us.

WJ: Would you say that the marketplace has been finding Portal as a technology or tool to gain some competitive advantage?

RG: I see that it's actually becoming more than just a tool; customers are using WebSphere Portal as the standard presentation framework for their entire IT infrastructure. They're saying, "Okay, if any Web development is going to be done we're going to use WebSphere Portal." Normally we saw it isolated to portal applications. Now companies are seeing a major productivity boom by replacing their multitude of custom web frameworks with a single web presentation framework based on open standards. That of course is WebSphere Portal.

WJ: With the growth and the acceptance, and uptake of Portal, I imagine that you have a repertoire of challenges and experiences on behalf of your customers. Are there any in particular that you'd like to share with our viewing audience today?

RG: Previously our customers were developing applications that were either departmental or not critical to their company. Now they are using WebSphere Portal to develop large enterprise applications. These applications require the same discipline/process that you would use for any conventional critical production application.

You cannot expect to deliver a major application in two weeks. You need to adhere religiously to a project management process and a defined methodology process throughout the portal development life cycle. In reality, the average portal application development cycle is between four to six months, and even some take up to two years. It's not because it's difficult to use; it's because we're talking about large applications that integrate multiple systems

throughout your enterprise.

Another challenge that we find is the lack of emphasis on testing. Testing needs to start at the beginning of the project. WebSphere Portal projects usually touch every critical system in the enterprise. Each time you integrate a different component you should plan at least one to two weeks of stress testing per integration component. Testing can take up to four months with a complex application that has numerous integration points.

WJ: Well, today we hear more and more about acquisitions and mergers in the business marketplace. How quickly can Portal scale in the event of a merger or acquisition of a company that already is using Portal quite extensively when they acquire another company or an expanded division?

RG: That's an interesting question. For a merger or acquisition, I do not believe the tool is a critical element as much as the process. I have seen too much dependency on the tools and not on properly creating and validating the integration process. The process is actually far more critical than the technology, especially when we're talking about large mergers.

WJ: With regards to Portal, how does it support small- to medium-size businesses?

RG: We have a product called WebSphere Portal Express. The product is designed to be installed quickly and easily and fits within the price range that a small to medium size business can afford. It gives you the functionality you need to create sophisticated portal applications. But to be honest with you, WebSphere Portal Express gives you tremendous value for the dollar that you're getting.

WJ: And is Express supported and receiving upgrades alongside the large scale version, if you will?

RG: Yes, it is. And actually it has some really interesting and very nice user interface enhancements (also available in IBM WorkPlace) that makes it easier for the small and medium size businesses to use it.

Part 2 of this interview will appear in the May 2005 issue of WebSphere Journal. *Richard Gornitsky addresses future WebSphere Portal enhancements, including enterprise management and system management tools integration in this concluding portion of the interview.*

WebSphere Journal Advertiser Index...

COMPANY	URL	PHONE	PG
BLOG-N-PLAY.COM	WWW.BLOG-N-PLAY.COM	201-802-3000	21
EV1 SERVERS	WWW.EV1SERVERS.NET	800-504-SURF	15
INFERDATA	WWW.INFERDATA.COM/WDJ	888-211-3421	31
ISSJ	WWW.ISSJOURNAL.COM	888-303-5282	39
INSTANTIATIONS	WWW.INSTANTIATIONS.COM/ANALYTIX	800-808-3737	9
ITKO	WWW.ITKO.COM	877-BUY-ITKO	33
ITEMFIELD	WWW.ITEMFIELD.COM/WBI	800-838-1055	51
IT SOLUTIONS GUIDE	WWW.ITSOLUTIONS.SYS-CON.COM	201-802-3021	41
JINFONET	WWW.JINFONET.COM	301-838-5560	25
MOTIVE	WWW.MOTIVE.COM/WITHIN1	512-531-2527	52
NASTEL TECHNOLOGIES	WWW.NASTEL.COM/UNCOMPLICATE	800-580-2344	11
PARASOFT	WWW.PARASOFT.COM/PRODUCTS	888-305-0041	2
PROLIFICS	WWW.PROLIFICS.COM	800-675-5419	3
QUESTSOFTWARE	WWW.QUEST.COM/WSDJ	949-754-8633	17
SYS-CON MEDIA PUBLICATIONS	WWW.SYS-CON.COM	888-303-5282	49
TANGOSOL	WWW.TANGOSOL.COM	617-623-5782	27
WILY TECHNOLOGY	WWW.WILYTECH.COM	888-GET-WILY	5
WEB SERVICES EDGE 2005	WWW.SYS-CON.COM/EDGE2005	201-802-3066	37

WebSphere Journal Coming Next Month...

Modeling to Implementation, Part 2

High-Performance SOAP Processing with VTD-XML

Using WAS on UNIX

Lotus CTO Doug Wilson Talks About the Enterprise

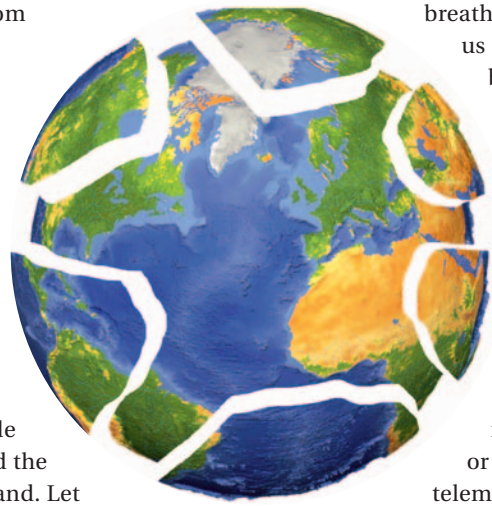
Deconstructing the Sphere

BY ROGER STRUKHOFF

In the spirit of French deconstructionist Jacques Derrida, who died a few months ago, let's indulge ourselves in a little wordplay of our own this month. Don't worry, I haven't read much of Derrida's work, and am hardly qualified to comment on it. The good news is, that thanks to the Internet and World Wide Web, I don't have to be. You can find all the Derridaesque analysis you want through the most basic of yahoo or google searches. As one blog puts it, "we are developing the social individualist meta-context for the future." How could I possibly top that worthy goal?

But mention of the Web and of a blog leads to my own deconstructionist point. The Internet was abstractly thought of as a net (duh!), an interconnected structure through which messages could travel through myriad possible paths to reach its goal. It could simply go from point A to point B, but mostly likely took the scenic route and meandered through point XY and perhaps point 4RQ, which I understand is lovely this time of year.

The Web was abstractly thought of as a…web, that's right, which sounds to me like a more densely-weaved net. Oh, and it is worldwide, in distinction from the Net, which was originally envisioned in the U.S. Never mind that the Net was truly global when the Web was invented, the idea of a "worldwide" web probably sounded very egalitarian to an Englishman living in Switzerland, and it also made for a crackling domain prefix of WWW.



In any case, both of these structures are fundamentally two-dimensional in shape. Now, don't get all pedantic on me and tell me that all physical structures are three-dimensional, even if only a single atom thick, and that therefore the Net and the Web shouldn't be considered part of Flatland. Let me tell the story, kid.

Now, to my point (understated drum roll, please). The great breakthrough in the past few years as been to think of cyberspace in three dimensions. As in WebSphere. As in the blogosphere. As in, well, as in cyberspace. Add the fourth dimension of time, measured in billionths, if not trillionths, of a second, and we can start appreciating what we have wrought. A fully-functioning universe consisting of simple electrons that transmit great thoughts, banal conversations, and perhaps, ultimate truth.

Cyberspace has been thought of as three-dimensional at least since the *Neuromancer* days envisioned by William Gibson. But until we started naming our products and our concepts in these terms, we were just toodling along in a flat, inefficient grid that was better equipped to send words than images.

With the advent of common usage of the word "sphere" in our conversations, we are equipping ourselves with the abstract concepts that will lead to fruition of our great ideas.

By creating an abstract world metaphysically similar to our own, we can start to bring life of sorts into it. We can now solve all problems, small or large, by a simple application of our new stone tools.

We can even talk of IT ecosystems as living, breathing things. Not things carrying the dystopic message of a malignant intelligence that threatens our existence. Not all living, breathing things have intelligence (and most of us are living, breathing proof of that on a daily basis). But they can be dynamic, responsive to conditions, able to evolve, and capable of spawning new ideas (whether we call them revs, iterations, or paradigms).

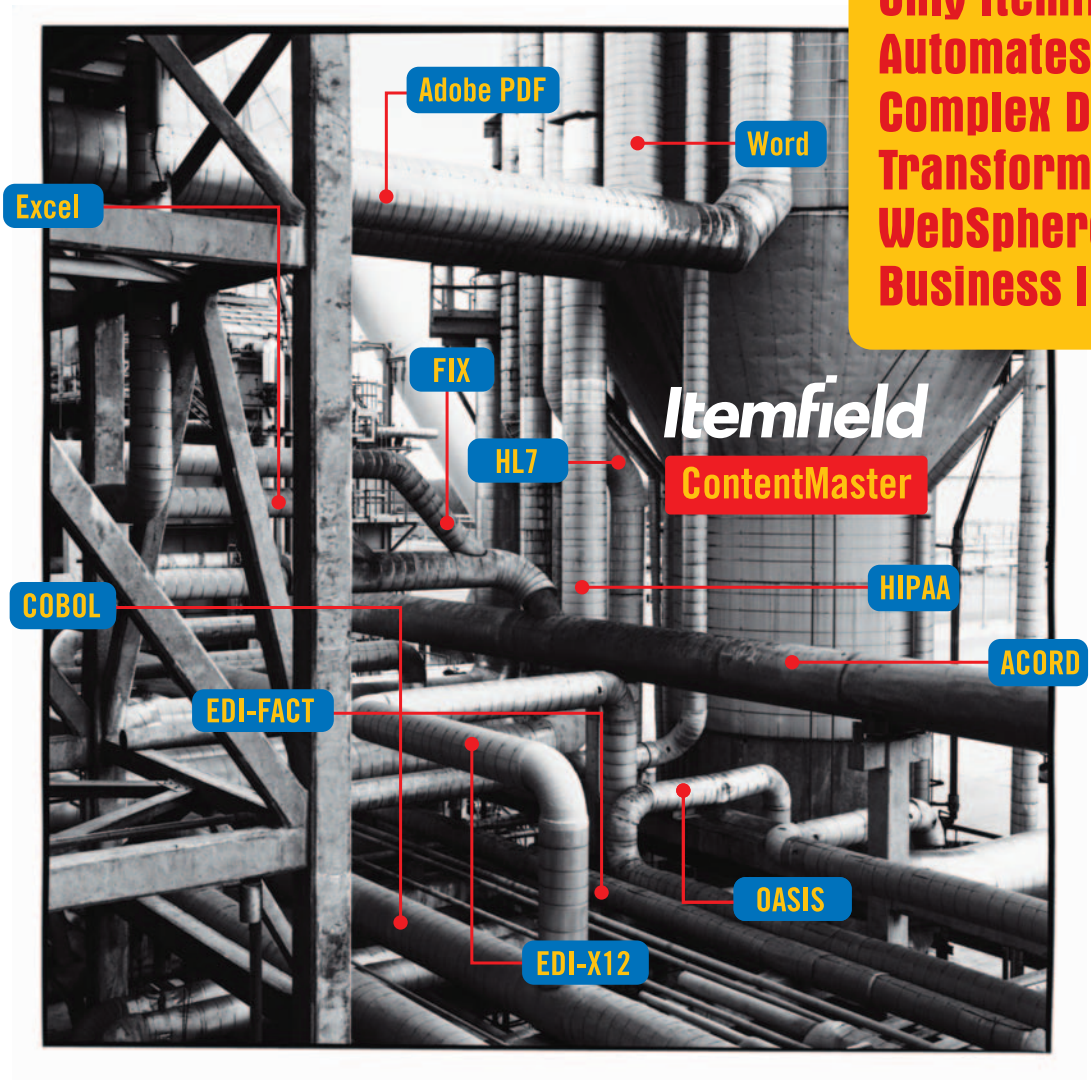
By simple use of the word "sphere," we have reconstructed the older way of thinking in terms of a mere Net or Web. We are thinking as grandly as possible so that we can build things as specifically as possible.

Yeah, sure, sometimes these tools make nothing more than viruses or spam or data correlations that enable annoying telemarketing callers to interrupt dinner or the ballgame. But most of you are doing good in this newly conceived sphere of ours, right?

As Jacques Derrida himself wrote, "That the particularity of the example does not interfere with the generality of my argument is a point which I shall occasionally – try not merely to take for granted." Uh, yeah, I couldn't agree more…

Roger Strukhoff, editor-in-chief of ***WebSphere Journal***, is West Coast Bureau Chief for the SYS-CON News Desk, and President of www.wdva.com. He spent 15 years with Miller Freeman Publications and The International Data Group (IDG), then cofounded CoverOne Media, a custom publishing agency that he sold in 2004. His work has won awards from the American Business Media, Western Press Association, Illinois Press Association, and the Magazine Publishers' Association. Read his blog at www.rssblog.linuxworld.com. Contact him at roger@sys-con.com.

Only Itemfield Automates Complex Data Transformation For WebSphere® Business Integration



Extend The Power And Reach of WebSphere Business Integration

Transform Any Data Format to XML and Back Again—Without Writing Code

Together, IBM and Itemfield solve the costly problem of data transformation to reduce the cost of business integration with a combination of products:

From Itemfield

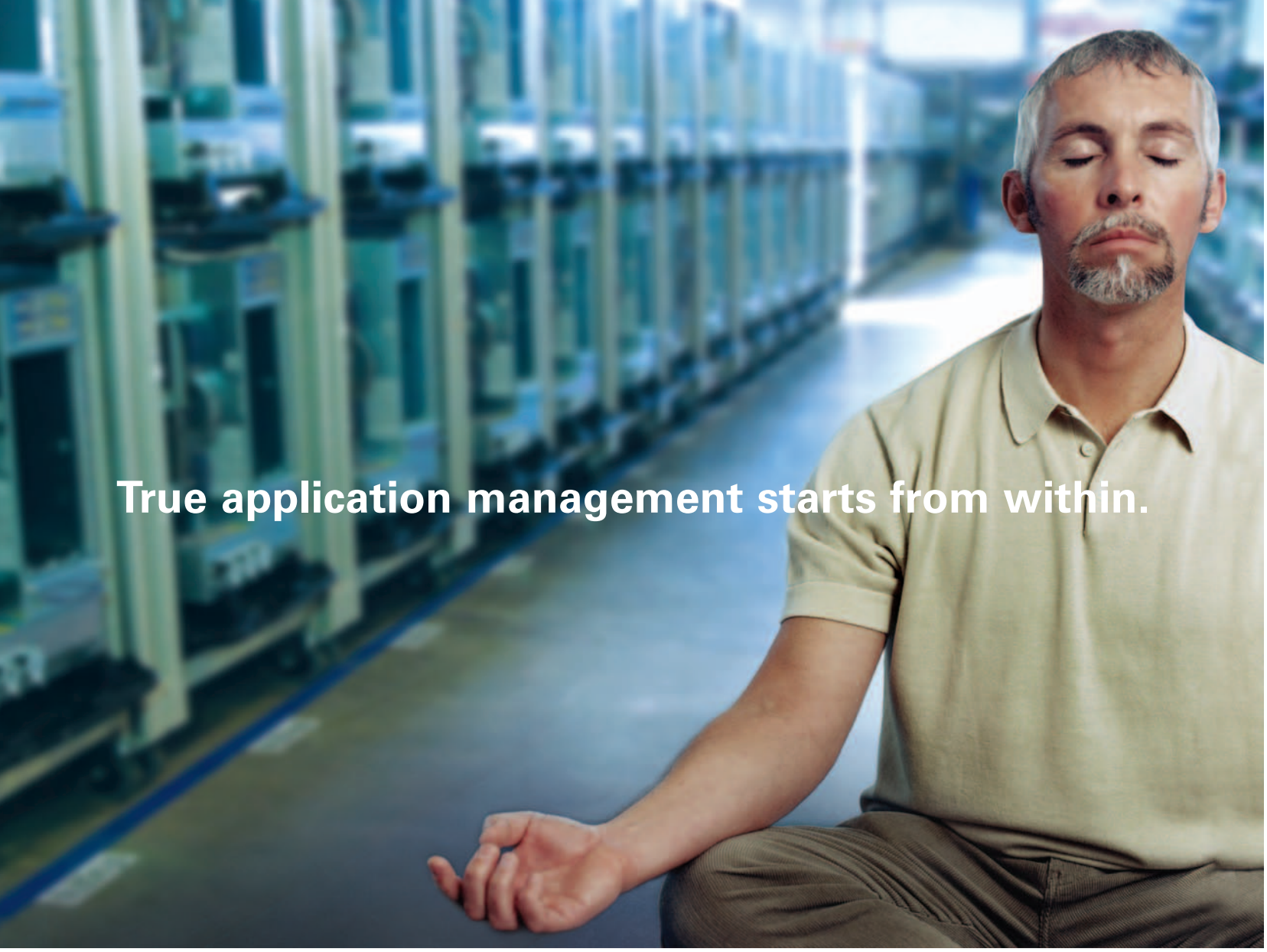
Itemfield ContentMaster™ Agent for WebSphere Business Integration Message Broker simplifies solutions to a broad range of integration needs and data formats with WebSphere Business Integration Message Broker, providing a high throughput interface between any data formats.

From IBM

IBM WebSphere® Business Integration Data Handler for Complex Data leverages Itemfield ContentMaster to enable the bidirectional conversion of a vast number of text and binary formats to and from WebSphere Business Integration business objects and can be used with any server in the WebSphere Business Integration product family.

Visit our website **www.itemfield.com/wbi** to obtain a copy of our latest white paper *"The Complexity Advantage: Solving the Requirements of Complex Data Transformation in Business Integration"* and to view the ContentMaster Solution for WebSphere Business Integration demo.

Itemfield | 1 800 838 1055 | www.itemfield.com/wbi



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

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

motive

www.motive.com