

# JAVA<sup>TM</sup> DEVELOPER JOURNAL

The World's Leading Java Resource

December 2003 Volume:8 Issue:12

www.JavaDevelopersJournal.com

## International Conference & Expo

**Edge 2004**  
EAST

Feb. 24-26, 2004  
Boston, MA

SEE PAGES 50-57

### From the Editor

**Haven't We Got Enough to Remember As It Is?**

Alan Williamson pg. 5

### Viewpoint

**Inversion of Control Rocks**

Paul Hammant pg. 6

### J2EE Insight

**For the Coming Year...**

Joseph Ottinger pg. 8

### J2SE Insight

**A Christmas Wish List**

Jason Bell pg. 24

### J2ME Insight

**A New Year, a New Future**

Glen Cordrey pg. 44

### Industry News

pg. 64

**RETAILERS PLEASE DISPLAY UNTIL FEBRUARY 29, 2004**

\$5.99US \$6.99CAN

0 1 >



BY MARCUS S. ZARRA PAGE 34

For a more responsive GUI



## Functionality: It's the Aspects

A new paradigm that makes OOP multidimensional

Bill Burke

10



## Feature: Non-Stop EJB Services

Deploy new releases at your leisure

J. Bradley & D. Raal

16



## Java Framework: PircBot 1.2.5 Java IRC API

Have fun with Java and create your first IRC bot

Paul Mutton

26



## Feature: Building a Connected MIDlet

...and installing on a mobile device

A. Bourgeois & R. Kasperowski

46



## Labs: Teamstudio Analyzer for Java

For a uniform coding style

Maxim Chpakov

58



## Labs: XML Tunneling Technology

(XTT) Version 3.0 by InsiTech Group, Inc.

Robert Carten

60



## JSR Watch: From Within the Java Community Process Program

From elections to final ballots

Onno Kluyt

62

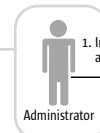
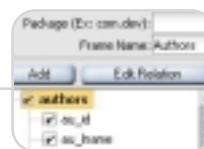


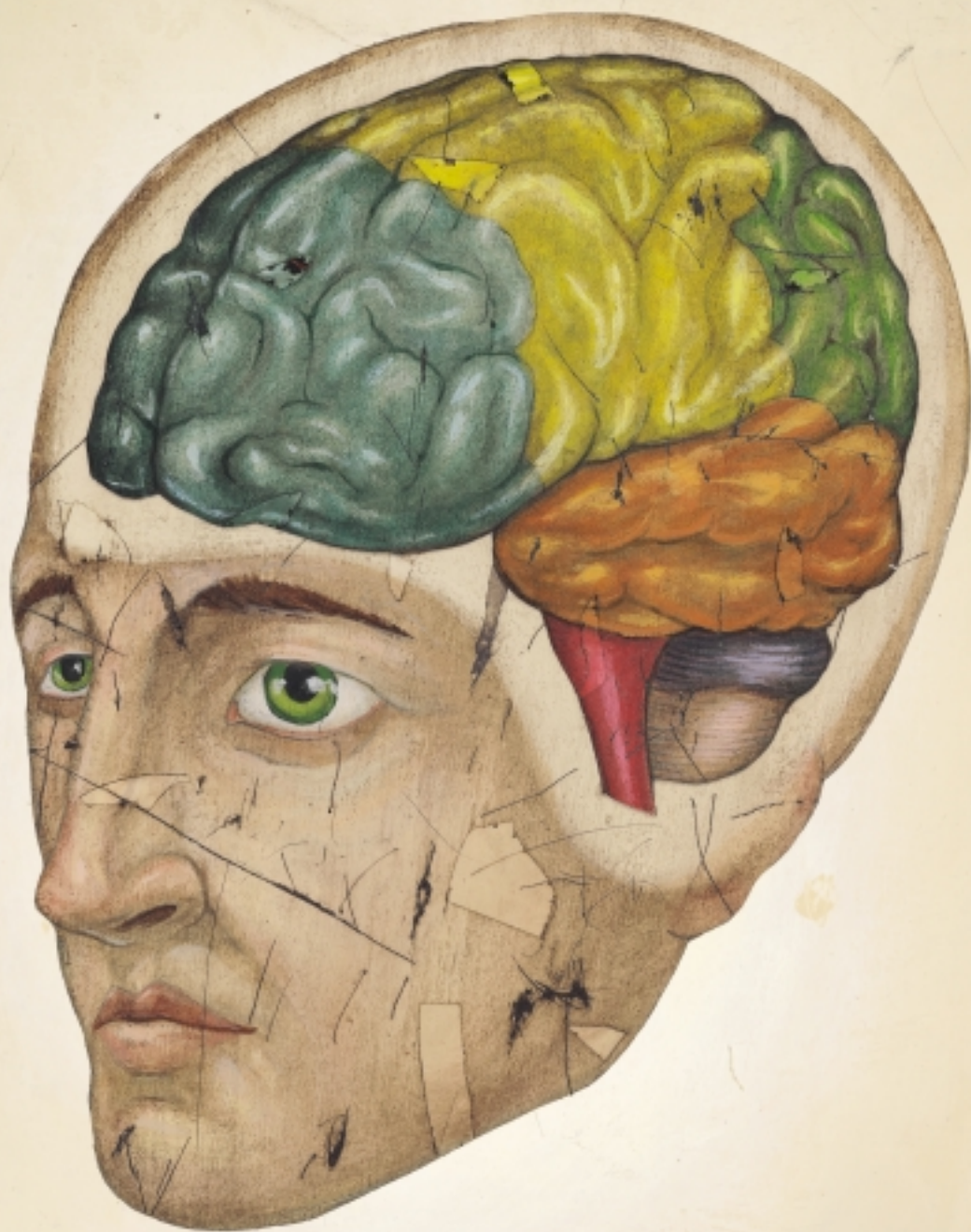
## From the Inside: Dude, Where's My Language?

Sun plays the "Do or Die" card

Henry Roswell

66







# DO YOU HAVE A BRAIN?

## CEREBRUM: CONTROLS THOUGHT.

*What is 2+2? What color is the sky?  
If you can answer these, then you have a cerebrum.*

## MEDULLA OBLONGATA: CONTROLS INVOLUNTARY FUNCTIONS.

*Check your pulse. Do you have one?  
Then you have a medulla oblongata, too.*

## BROCA'S AREA: CONTROLS LANGUAGE.

*Say the following sentence: "Frankly, Hector, I'm a bit surprised."  
Did it work? Then you have a Broca's Area.*

## PITUITARY GLAND: CONTROLS HORMONES.

*When you were about 13, did your voice change?  
Did you grow hair in special places?  
Then you've got a pituitary gland, friend.*

# YOU HAVE A BRAIN.

So why spend so much time on mindless coding?

Download WebLogic Workshop Now.

BEA WebLogic Workshop™ 8.1 is a more efficient way to develop in J2EE.  
And that means less grunt work and more real work. [dev2dev.bea.com/cerebrum](http://dev2dev.bea.com/cerebrum).







# Monitored

The right approach to application life cycle management can transform your business.

## AllFusion™ Life Cycle Management Software

The key to great development isn't just great developers, it's great management. That's why we created AllFusion, a comprehensive application life cycle management solution. AllFusion has unprecedented flexibility that allows your projects to change along with the market. And that helps you do something a lot more important than just minimize aggravation. It lets you maximize productivity. To learn how to improve your development process, or to get a white paper, go to [ca.com/lifecycle](http://ca.com/lifecycle).



**Computer Associates®**

© 2004 Computer Associates International, Inc. (CAI). All rights reserved.



# Maximized





**Alan Williamson**  
Editor-in-Chief



J2ME



J2SE



J2EE



HOME

## Haven't We Got Enough to Remember As It Is?

**T**he journey of a developer is never completed – due to constantly learning, retooling, and grasping new concepts. It's this continual learning that lures many to our profession. Sometimes, though, it can be tiresome trying to keep up and hoping what you're learning today will still be there tomorrow. At times, software disciplines come and go quicker than fashions.

For my own part, I took a sabbatical of sorts for six months. I had quite a lot going on in my life and didn't have the time to be looking in any great depth at the new disciplines coming through. Fortunately, though, my role here at **JDJ** keeps me informed of all trends so I wasn't completely out of the game.

Java is a wonderfully diverse and very comprehensive language. Keeping up with the latest developments in the JDK is a full-time job in itself. The rate at which it is growing is enough to make even the seasoned developer scratch his or her head in exasperation. It's a common complaint we hear from developers new to the language, but that's the price of progress.

A new area that is gaining a lot of traction at the moment is AOP or Aspect-Oriented Programming. I've been hearing more and more on this, and this month Bill Burke from JBoss has written our first AOP article, introducing this new, powerful complement to Java.

AOP is something I've been casually reading about on various blogs and sites, but for some reason I just wasn't clicking on the usefulness of it. Maybe I was reading the wrong level of articles. But at a recent No Fluff Just Stuff conference, I made a point of attending Ron Bodkins' introduction to AOP, and I'm thankful I did. Instantly all the pieces fell into place.

AOP is a wonderful complement to the whole OO philosophy and Java is very capable, through AspectJ, of tak-

ing advantage of what AOP has to offer. What can it do? In a nutshell, it has the ability to give you god-like control over the object calls within your JVM. You can set up "filters" that can be triggered before and after method calls. This allows you to not only inspect parameters but also affect them.

It's often touted as a clean way to perform logging or control security, but one of the cool features is the ability to create "wormholes" in your code to pass references around. Instead of carrying around a reference to a particular class that you need in a deep call, remove it from the call and simply reinsert it when the method needs it.

Sounds like a hack doesn't it? In many respects it is. You could think of it as a legalized, completely authorized "goto." Possibly that's not entirely fair, but the main drawback I hear about AOP is that when you do it right, it's a wonderful addition; do it wrong and it's a nightmare to trace. So you have to respect the power it offers and use it only for the greater good, as opposed to short-cutting a poor design.

On that note, I would like to wave goodbye to the look'n'feel of this **JDJ** because next month we introduce **JDJ** v3. We've been busy working on a new style for the magazine that will reflect the changing Java landscape. We have restructured the sections and assigned new editors to cover a wider range of subjects. In addition to focusing on new technologies, we will be looking at the application of Java in the field and interviewing those behind the decisions to deploy Java over and above the alternatives.

There are those who would lead you to believe Java is dead. Believe none of it. From January on you will be reading success stories and, if your faith has been flagging a little, we'll perk it up for you.

Until next month, have a great Christmas and see you in 2004. ☺

**Alan Williamson**, when not answering your e-mails and working on the next issue of **JDJ**, heads up a small team dubbed the "Thunderbirds of the Java industry," providing on- and off-site rescue for Java projects in trouble. For more information visit [www.javaSOS.com](http://www.javaSOS.com). You can also read his blog: <http://alan.blog-city.com>.

[alan@sys-con.com](mailto:alan@sys-con.com)

### International Advisory Board

Calvin Austin (Sun)  
Jason Bell (Independent)  
Jason Briggs (Independent)  
Jeremy Geelan (SYS-CON)  
Thorsten Laux (Sun)  
Rickard Öberg (Independent)  
Joe Ottinger (Independent)  
Bill Roth (E.piphany)  
Ajit Sagar (Independent)  
Eric Stahl (BEA)  
Jon Stevens (Apache)  
Aaron Williams (JCP)  
Alan Williamson (SYS-CON)  
Joe Winchester (IBM)  
Blair Wyman (IBM)

### Editorial

Editor-in-Chief: **Alan Williamson**  
Executive Editor: **Nancy Valentine**  
J2EE Editor: **Joe Ottinger**  
J2ME Editor: **Glen Cordrey**  
J2SE Editor: **Jason Bell**  
Contributing Editor: **Jason R. Briggs**  
Contributing Editor: **Ajit Sagar**  
Founding Editor: **Sean Rhody**

### Production

Production Consultant: **Jim Morgan**  
Associate Art Director: **Tami Beatty**  
Associate Editors: **Jamie Matusow**  
**Gail Schultz**  
**Jean Cassidy**  
**Jennifer Van Winkel**  
Assistant Editor: **Kelly Flynn**  
Online Editor: **Lin Goetz**  
Technical Editor: **Bahadir Karuv, PhD**

### Writers in This Issue

Jason Bell, Alex Bourgeois, Joe Bradley, Bill Burke, Robert Carten, Maxim Chpakov, Glen Cordrey, Paul Hamman, Richard Kasperowski, Onno Kluyt, Paul Mutton, Joseph Ottinger, David Raal, Henry Roswell, Alan Williamson, Marcus Zarra

To submit a proposal for an article, go to <http://grids.sys-con.com/proposal>

### Subscriptions

For subscriptions and requests for bulk orders, please send your letters to Subscription Department [subscribe@sys-con.com](mailto:subscribe@sys-con.com). Cover Price: \$5.99/issue. Domestic: \$69.99/yr. (12 Issues) Canada/Mexico: \$99.99/yr. Overseas: \$99.99/yr. (U.S. Banks or Money Orders) Back Issues: \$10/ea. International \$15/ea.

### Editorial Offices

SYS-CON Media, 135 Chestnut Ridge Rd., Montvale, NJ 07645  
Telephone: 201 802-3000 Fax: 201 782-9638

Java Developer's Journal (ISSN#1087-6944) is published monthly (12 times a year) for \$69.99 by SYS-CON Publications, Inc., 135 Chestnut Ridge Road, Montvale, NJ 07645. Periodicals postage rates are paid at Montvale, NJ 07645 and additional mailing offices. Postmaster: Send address changes to: Java Developer's Journal, SYS-CON Publications, Inc., 135 Chestnut Ridge Road, Montvale, NJ 07645.

### ©Copyright

Copyright © 2003 by SYS-CON Publications, Inc. All rights reserved. No part of this publication may be reproduced or transmitted in any form or by any means, electronic or mechanical, including photocopy or any information storage and retrieval system, without written permission. For promotional reprints, contact reprint coordinator Carrie Gebert, [carrieg@sys-con.com](mailto:carrieg@sys-con.com). SYS-CON Media and SYS-CON Publications, Inc., reserve the right to revise, republish and authorize its readers to use the articles submitted for publication.

Worldwide Newsstand Distribution  
Curtis Circulation Company, New Milford, NJ  
For List Rental Information:

Kevin Collopy: 845 731-2684, [kevin.collopy@edithroman.com](mailto:kevin.collopy@edithroman.com)  
Frank Cipolla: 845 731-3832, [frank.cipolla@epostdirect.com](mailto:frank.cipolla@epostdirect.com)

Java and Java-based marks are trademarks or registered trademarks of Sun Microsystems, Inc., in the United States and other countries. SYS-CON Publications, Inc., is independent of Sun Microsystems, Inc. All brand and product names used on these pages are trade names, service marks or trademarks of their respective companies.



# Inversion of Control Rocks

Paul Hammant



**I**nversion of Control (IoC) is about software components doing what they are told, when they are told. Your OO application could well become unmaintainable without it.

IoC is a pattern that helps teams avoid the dependency hell that results when an application grows into a large pseudo-platform without taking care to adequately decouple logic; that thing that ultimately only a couple of its omnipotent architects or old-lag premium-rate contractors really understand; that system that Heath Robinson and Rube Goldberg might have made together (look them up).

The problem with small working applications that become large is that static-method entanglement does not scale. One part of a system that's otherwise fairly self-contained statically accesses another part of the system and cannot be instantiated without invoking methods in the latter. Thus it can't easily be isolated for unit testing. What's more, it can't be developed as an independent component. Components that are developed separately can have their own development teams and may well be part of a larger design. These components will have their own source control directory and can be developed against mock implementations of their dependent components. All this will help overall development become faster, in terms of both the team's efficiency and the build time.

As an example, let's take a Personal Information Manager (PIM) application that has UI and persistence elements. The naïve implementation might have inline JDBC statements among graphical code. A componentized application would have that persistence logic separated into a persistence component with user interface logic enshrined in a view-controller component. Clearly, if the two teams developing their respective components agree on a slowly evolving interface/implementation-separated API for persistence, they can develop at their own pace and ship versions of their components whenever it's appropriate.

A third piece, which is not a component, would be the bootstrap for the application. That bootstrap may well be entirely contained in a static main

method of a simple class and would merely instantiate the two components, passing one into the other's constructor before invoking `setVisible(true)` or similar. With the introduction of this bootstrap we can see the control aspect of the IoC pattern.

In a noncomponentized version of our example, the view-controller may well have its own main method, might instantiate a fixed version of the PIM store, or access it via an unnecessary singleton factory (public static methods are generally bad), i.e., the control is very much inside the component in question.

The word "inversion" from the pattern name is about getting control back. The containing application (often a true container or a proper framework rather than a main method) controls when a component is instantiated and which implementations of its dependent components it is passed.

IoC also dictates the configuration of components. A JDBC version of the PIM store above would clearly require some JDBC settings. Classically, developers may write a mechanism to retrieve them from a fixed properties file. IoC would insist that the configuration is passed into the component. In our hypothetical example, it would take it via the constructor and be interface/implementation separated.

Thus the configuration is a component. It would be tightly coupled to the component that requires it, but subject to multiple implementations, one of which may be the "from properties file".

Inversion of Control has moved to the center stage in the last six months after a five-year gestation period. There are three types of Inversion of Control. Type 1 uses configuration data (Avalon, JContainer); Type 2 uses bean introspection (Spring Framework, WebWork2, HiveMind); Type 3 uses constructor signatures (PicoContainer).

For a componentized system that uses singletons or similar for component resolution, the dependency is obscure. In IoC it is declarative. The application would have loosely coupled components and be more scalable, more maintainable, and more testable. It's a very small investment for a very large return. Inversion of Control rocks. ☺

Paul Hammant is a still-coding architect for ThoughtWorks in London and has been programming professionally since 1989. He was a former committer on Apache's Avalon project, but left and cofounded the open-source PicoContainer project and its sister NanoContainer as well as remaining involved in the post-Avalon JContainer project.

paul@thoughtworks.com

**SYS-CON  
MEDIA**

President and CEO:  
**Fuat Kircaali** fuat@sys-con.com  
Vice President, Business Development:  
**Grisha Davida** grisha@sys-con.com  
Group Publisher:  
**Jeremy Geelan** jeremy@sys-con.com

## Advertising

Senior Vice President, Sales and Marketing:  
**Carmen Gonzalez** carmen@sys-con.com  
Vice President, Sales and Marketing:  
**Miles Silverman** miles@sys-con.com  
Advertising Sales Director:  
**Robyn Forma** robyn@sys-con.com  
Director, Sales and Marketing:  
**Megan Ring** megan@sys-con.com  
Advertising Sales Manager:  
**Alisa Catalano** alisa@sys-con.com  
Associate Sales Managers:  
**Carrie Gebert** carrie@sys-con.com  
**Kristin Kuhnle** kristen@sys-con.com  
**Beth Jones** beth@sys-con.com

## Editorial

Executive Editor:  
**Nancy Valentine** nancy@sys-con.com  
Associate Editors:  
**Jamie Matusow** jamie@sys-con.com  
**Gail Schultz** gail@sys-con.com  
**Jean Cassidy** jean@sys-con.com  
**Jennifer Van Winckel** jennifer@sys-con.com  
Assistant Editor:  
**Kelly Flynn** kelly@yachtchartersmagazine.com  
Online Editor:  
**Lin Goetz** lin@sys-con.com

## Production

Production Consultant:  
**Jim Morgan** jim@sys-con.com  
Lead Designer:  
**Tami Beatty** tami@sys-con.com  
Art Director:  
**Alex Botero** alex@sys-con.com  
Associate Art Directors:  
**Louis F. Cuffari** louis@sys-con.com  
**Richard Silverberg** richards@sys-con.com

## Web Services

Vice President, Information Systems:  
**Robert Diamond** robert@sys-con.com  
Web Designers:  
**Stephen Kilmurray** stephen@sys-con.com  
**Christopher Croce** chris@sys-con.com

## Accounting

Accounts Receivable:  
**Charlotte Lopez** charlotte@sys-con.com  
Financial Analyst:  
**Joan LaRose** joan@sys-con.com  
Accounts Payable:  
**Betty White** betty@sys-con.com

## SYS-CON Events

President, SYS-CON Events:  
**Grisha Davida** grisha@sys-con.com  
Conference Manager:  
**Michael Lynch** mike@sys-con.com  
National Sales Manager:  
**Sean Raman** raman@sys-con.com

## Customer Relations

Circulation Service Coordinators:  
**Shelia Dickerson** shelia@sys-con.com  
**Edna Earle Russell** edna@sys-con.com  
JD Store Manager:  
**Brunilda Staropoli** brunilda@sys-con.com



# **FINALLY, BUSINESS** **SOLUTIONS THAT** **WORK WITH EXISTING** **TECHNOLOGIES** **AND NONEXISTENT** **BUDGETS.**

You need to get more out of what you have. We have just the thing: solutions based on our open technology platform, SAP NetWeaver<sup>®</sup>. Because it's preconfigured to work with your current IT investments – and it's fully operable with .NET, J2EE and IBM WebSphere – SAP NetWeaver reduces the need for custom integration. That lowers your total cost of ownership for your entire IT landscape and gets you quicker ROI. Everything a CIO wants (and a CFO didn't think was possible). Visit [sap.com/netweaver](http://sap.com/netweaver) or call 800 880 1727 for details.

THE BEST-BUY BUSINESS IS WITH SAP







Joseph Ottinger

J2EE Editor

## For the Coming Year...

It's that time of year, when the air is crisp and cool, and lights fill the air with the glint of good cheer and renewal. It's when wishes are fulfilled; when revitalization is just around the corner. I'd like to take some time to share some of the things I'd like to see for the next year, and about the people for whom I'd like to see them.

*For Sun:* I'd love for them to come up with a unified business and marketing plan, and stick to it. They have a marvelous suite of products, filled with potential; however, poor marketing and occasional shoddy construction mangle it in the field for the low- to mid-range markets. With a decent business plan and more cohesive price structures, they could easily go head-to-head with their competition...including the other four-hundred pound gorillas, Microsoft and IBM. It would also free them from paying unnecessary attention to small fry.

*For JBoss:* Patience! JBoss Group is a bunch of techies; hard-core programmers who maintain their reputation by being crusty, demanding, and hard to deal with. A lot of people lambaste them for it, while not realizing that when it gets down to brass tacks, these are exactly the sort of personalities you want running the trains, as opposed to a lot of feel-good marketers who don't know what they're talking about.

*For Apache:* I hope your endless confusion about what the world needs in components becomes clear. Perhaps you could do some of the clarification yourselves. Some of the stuff hosted by Jakarta is really useful; a lot of it is worth the derision that occasionally gets thrown at it by the monkeys.

*For Microsoft:* My fondest wish for these guys is that someone – anyone – would drop by and teach them something – anything – about security besides how to expose data unwillingly again and again and again and again. I have to hand it to Microsoft: they're great at protecting most of their own data while providing the means to expose virtually

everyone else's stuff. I'm glad I rarely use Windows, even though I think it has a very acceptable user interface. My next wish for Microsoft would be that they'd realize how much fun it can be to try to dominate through technical prowess, instead of through marketing gimmicks and stupid advertising campaigns. That stupid MSN butterfly costume has to be one of the scariest ad campaigns ever – because my fear is that they expected people to take it seriously. Compounding that fear is the thought that people might have fulfilled that expectation.

*For specification writers everywhere:* I would put forth hope that they realize that, while surely not being mortal themselves, those who use their specifications most likely are fallible humans, and that their specifications are written with implementors and users in mind. Sometimes I think this is the crucial flaw in the JCP – with the apparent attitude repaired, all other failures could be forgiven and rectified in time.

*For JDJ:* I hope we can continue to inform and focus the Java community, such that it is, for the betterment of all.

*For application vendors not necessarily in the Java space:* I would love to see clues about Java's speed, reliability, and ease of use descend upon them like manna from above. I've been using a "less-supported" platform recently, and I'd like nothing better than to see common applications that I'd rely on written in Java so I could use them. What would be the icing on this particular cake is if some as-yet-unnamed vendor provided assistance – for a small fee, of course – to help make this happen.

*For you:* I wish for you all peace, love, a Coke, and a smile.

*For myself:* I wish I didn't keep mixing my similes. I want the best possible world for my children, health and happiness for my family, a happy working environment, enough freedom to do as I please while harming no one else, and enough room to crank up the volume every now and then without annoying the neighbors too much. ☺



### For the Coming Year...

It's that time of year, when the air is crisp and cool, and lights fill the air with the glint of good cheer and renewal. It's when wishes are fulfilled; when revitalization is just around the corner. I'd like to take some time to share some of the things I'd like to see for the next year, and about the people for whom I'd like to see them.

### It's the Aspects

Aspect-oriented programming (AOP) is a promising new paradigm that came out of Xerox PARC a few years ago and is just now becoming mature and mainstream. A natural complement to object-oriented programming, it has the promise of easing the management of complex systems and making their organization much more intuitive, extendable, and flexible.

Joseph Ottinger is a consultant with Fusion Alliance ([www.fusionalliance.com](http://www.fusionalliance.com)) and is a frequent contributor to open source projects in a number of capacities. Joe is also the acting chairman of the JDJ Editorial Advisory Board.

[josephottinger@sys-con.com](mailto:josephottinger@sys-con.com)



# The **only** Java components you need for J2EE or Swing development

## JClass®

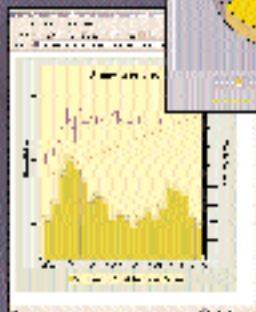
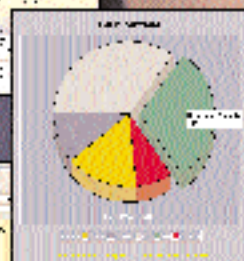
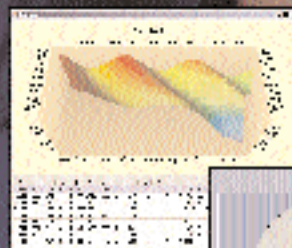
Rich client user interface and utility components. Server-side web client interface and reporting components. Whatever type of Java development you're doing, JClass can help.

### JClass ServerViews

Add professional content to your Servlet, JSP or J2EE applications. Generate interactive charts with JClass ServerChart and dynamic PDF reports with JClass ServerReport. Now fully XML and Web Services ready!

### JClass DesktopViews

Essential components for client-side Java applications and applets: 2D/3D charts, tables/grids, data-entry fields, data base access and much more.



Evaluate and experience JClass today – visit  
<http://java.quest.com/jclass/jdj>



Bill Burke

# It's the Aspects

## A new paradigm

**A**spect-oriented programming (AOP) is a promising new paradigm that came out of Xerox PARC a few years ago and is just now becoming mature and mainstream. A natural complement to object-oriented programming, it has the promise of easing the management of complex systems and making their organization much more intuitive, extendable, and flexible. AOP makes OOP multidimensional.

What is an Aspect? An Aspect is a common functionality that's scattered across methods, classes, object hierarchies, or object models. Functionality that your class or object model shouldn't be concerned about, functionality that doesn't belong as it's not what the object is all about. The AOP-ites like to call this type of functionality crosscutting concerns, as the behavior is cutting across multiple points in your object models, and yet is distinctly different from the classes it's crosscutting. AOP allows you to abstract and seamlessly componentize these concerns and apply them to your applications in a unique way that regular object-oriented programs cannot achieve very easily.

A simple example of a crosscutting concern is timing and metrics. Let's say you wanted to add code similar to Listing 1 to your application that would measure the amount of time it would take to invoke a particular method.

There are a few problems with this approach:

1. You have to manually add this code to multiple different files, methods, and classes, which is a pain; if you want to change which methods are profiled you have to manually edit those files. In other words, this is hard to turn off and on and difficult to maintain.
2. The profiling code really doesn't belong sprinkled throughout your application code. It makes your code bloated and harder to read as you have to enclose the timings within a try-finally block.
3. If you want to add other metrics like a method count or a failure count,

you would have to modify all the files where you manually inserted the profiling code. It's very difficult to maintain, expand, and extend your metrics functionality as it's dispersed throughout your entire code base.

This is a tiny example of how you can have common code that is sprinkled across many unrelated modules of your application; code that intrudes on the overall purpose of the Java class you're implementing. Aspect-oriented programming provides a way to pull together these common behaviors into a manageable unit and apply them to your code base. Let's look at how AOP would implement and solve this problem.

### Defining an Aspect

The first thing that should be done to aspectize the metrics functionality would be to create an Aspect. The try-finally block that we originally had within the `BankAccount.withdraw` method should be extracted and encapsulated into its own object. Having this code within its own object enables us to easily expand and maintain any additional metrics we may want to calculate later on in the development cycle. For this object to work, it must be able to wrap around and obtain contextual information about the particular method you want to add profiling to so that metrics can be displayed. There are a few AOP frameworks out there, so rather than picking one framework to give an example in, let's look at some pseudo code that could be easily translated into a real framework later on (see Listing 2).

The `MetricsAspect` class pulls together the metrics functionality into one maintainable, extendable unit. The `invoke` method at line 3 should be called in place of the actual method you want to provide metrics for. All AOP frameworks should provide some form of abstraction for wrapping/intercepting a method call. Line 8 wraps and delegates to the actual method. Line 13 assumes that you can obtain contextual information about the method call from the AOP framework you are using.

### Applying an Aspect

Now that we have extracted out the metrics functionality into a componentized Aspect, how can we apply it? This is where a pointcut comes in. A pointcut defines an entry point within your code base. It describes an event. An entry point could be a field access, a method call, or a constructor call. An event could be an exception being thrown. A pointcut is a way for you to define where you want your aspects applied. Let's look at some pseudo XML configuration for a pointcut that any AOP framework should be able to do in some form or another.

```
1. <method-pointcut
   expr="com.mc.BankAccount.withdraw(double
   amount)">
2.   <attach-aspectclass="com.mc.
   MetricsAspect"/>
3. </method-pointcut>

4. <method-pointcut expr="com.mc.billing.*">
5.   <attach-aspectclass="com.mc.
   MetricsAspect"/>
6. </method-pointcut>
```

Lines 1–3 define a pointcut that applies the metrics aspect to the specific method `BankAccount.withdraw`. Lines 4–6 define a general pointcut to apply the metrics aspect on all methods in all classes under the `com.mc.billing` package name. Most AOP frameworks have a rich set of pointcut expressions that you can use to apply your aspects. You can attach your aspects on an individual one-on-one basis to each Java class in your application, or you can use a more complex pointcut to specify a wide range of classes with one expression.

What this example shows is that with AOP, you're able to pull together crosscutting behavior into one object and sprinkle it easily and simply throughout your code base without making code unreadable or polluted with functionality that doesn't belong with the business logic you are implementing. Common crosscutting functionality can be maintained and extended in one place.

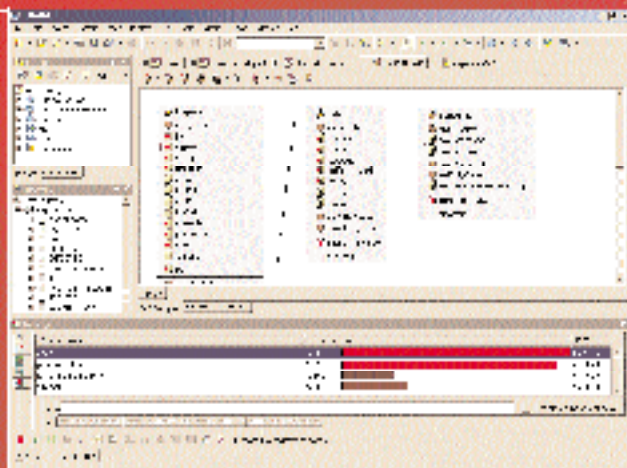


# ANNOUNCING JBuilder X MADE IN BORLAND

The World's #1 Java™ IDE Just Got Better.

- 1) Two-way Struts designer to visually create multi-tier Web applications
- 2) Visual Web Services designer to rapidly create, find, import/export, publish, and deploy Web Services
- 3) New visual deployment descriptor editor to manage deployment of J2EE™ and Web apps
- 4) Easy access to new refactorings helps you manage change
- 5) Configurable, streamlined working environment
- 6) TagInsight for JSP,™ HTML, and XML designed to speed coding
- 7) Award-winning Borland® Optimizer™ Suite performance tools included to help you build quality applications
- 8) Extensible solution that integrates the application lifecycle
- 9) Supports the entire spectrum of Java development including Web, mobile, JSP, database, EJB,™ XML, and Web Services
- 10) Integrates with all leading application servers from BEA, Borland, IBM, Oracle, Sun, Sybase, and now JBoss

Try Borland® JBuilder® X today. Download a copy at [go.borland.com/j6](http://go.borland.com/j6)



Visit [go.borland.com/j6](http://go.borland.com/j6)

**Borland®**  
Excellence Endures™



DEFINE  
Borland® Delphi® VCL



DESIGN  
Borland® Together®



DEVELOP  
Borland® JBuilder



TEST  
Borland® Optimizer Suite



DEPLOY  
Borland® Enterprise Server



MANAGE  
Borland® StarTeam®

Made in Borland® Copyright © 2005 Borland Software Corporation. All rights reserved. Java and all Java-based marks are trademarks or registered trademarks of Sun Microsystems, Inc. in the U.S. and other countries. All Borland brand and product names are trademarks or registered trademarks of Borland Software Corporation in the United States and other countries. © 2005 J1.0



Another thing to notice is that the code within the `BankAccount` class has no idea that it's being profiled. The application developer was allowed to focus on writing business logic rather than being distracted with writing the code candy and syntactic sugar of profiling. Needed orthogonal behavior could be snapped on after the fact quite easily without even touching this existing code base. This is a very subtle significant part of AOP as this complete obliviousness allows aspects to be layered on top of or below the functionality they are crosscutting. A layered design allows you, as a system designer, to more easily snap on or remove functionality or behavior that you need. For instance, maybe you only snap on the metrics functionality when you're doing some benchmarks but want to remove this within production. Or, if the AOP framework allows for it, maybe you want to turn on metrics in production to determine where bottlenecks are.

### Real-World AOP

In the early days of object-oriented programming, it was user-interface applications that helped to scope and discover object-oriented patterns and techniques. If you look at the Gang of Four's *Design Patterns* book (the bible of object-oriented programming), you'll see that GUIs are used in many of the coding examples that describe the patterns in the book. As GUIs helped formulate the early patterns of OO, middleware is shaping up to be the killer app for aspect-oriented programming.

Middleware, by nature, is crosscutting. It has functionality that's common across object hierarchies that really should not be mingled with business logic. The evolution of middleware has always been to abstract out how it is applied to regular simple objects. AOP completes this evolution as middleware functionality can be applied after the fact without changing the code or design of the existing business model. Packaging up middleware into a set of aspects frees developers to focus on writing the plain Java objects that make up their application's specific behavior rather than forcing them to work under an API dictated to them by their system architecture.

Take J2EE, for instance. It can be sliced and diced and served à la carte to your object model rather than going through the sometimes cumbersome and unnecessary process of imple-

menting an EJB. For instance, let's say you were using EJB solely for the purpose of defining transactions. Transaction demarcation lines could be drawn within any class at any point using AOP. Instead of extending `SessionBean` and writing home, remote, and local interfaces; deciding on a JNDI binding; and defining all your `<ejb-ref>`s in XML; all you would have to do is define a pointcut for the method of the class you want a transaction started from and attach the transactional aspect to trigger the desired behavior.

```
1. <method-pointcut
   expr="com.mc.BankAccount.withdraw(double
       amount)">
2.     <attach-aspect
       class="org.vendor.transaction.
           RequiredAspect"/>
3. </method-pointcut>
```

You can apply these same techniques to a multitude of middleware technology like remoteness, ACID, replicated caching, oneway, simple asynchronous invocations, role-based security, and persistence. AOP prevents system programming from intruding into your object model. It has the potential to completely separate the concern of middleware from your application logic. This can make your code easier to maintain and read, and more flexible as you can

make system architecture decisions later on in the development process. It's a pure layered approach to applying middleware.

### Conclusion

AOP is a new paradigm for expanding code reuse and easing the maintainability of your code base. It provides mechanisms to easily componentize code that is scattered throughout your object model and really needs to be organized centrally into one set of objects. When combined with something like middleware, it has the ability to isolate your business logic from the confines of system architecture, thus making your applications even more resistant to change as the landscape of APIs and public specifications changes over time. As framework developers focus on providing their functionality through aspects, the term pointcut will be morphed into pointclick as aspects are applied to an object model through the point-and-click interfaces of an IDE. ☞

### References

- Gamma, E., et al. (1995). *Design Patterns: Elements of Reusable Object-Oriented Software*. Addison-Wesley.
- JBoss: [www.jboss.org](http://www.jboss.org)
- AspectJ: [www.aspectj.org](http://www.aspectj.org)
- AspectWerkz: [www.aspectwerkz.codehaus.org](http://www.aspectwerkz.codehaus.org)

#### Listing 1

```
1. public class BankAccount
2. {
3.     public void withdraw(double amount)
4.     {
5.         long startTime = System.currentTimeMillis();
6.         try
7.         {
8.             ...actual method body
9.         }
10.        finally
11.        {
12.            long endTime = System.currentTimeMillis() - startTime;
13.            System.out.println("withdraw took: " + endTime);
14.        }
15.    }
16. }
```

#### Listing 2

```
1. public class MetricsAspect implements some.aop.framework.Aspect
2. {
3.     public Object invoke(Invocation invocation) throws Throwable
4.     {
5.         long startTime = System.currentTimeMillis();
6.         try
7.         {
8.             return invocation.proceedWithMethod();
9.         }
10.        finally
11.        {
12.            long endTime = System.currentTimeMillis() - startTime;
13.            java.lang.reflect.Method m = invocation.getMethod();
14.            System.out.println("method " + m.toString() + " time: " + endTime
+ " ms");
15.        }
16.    }
17. }
```

Bill Burke is the chief architect at JBoss Group, LLC.

[bill@jboss.org](mailto:bill@jboss.org)





Get  
Optimized

# Optimize J2EE.

The J2EE revolution is here to increase performance, value, and lower IT costs.  
It's a solution from Mercury Interactive that makes your whole J2EE applications ecosystem work right.

It's the fastest way to find the toughest J2EE problems from the development to the live applications. Even pinpoints root cause and the line of source code.

It's about more than delivering J2EE applications. It's about delivering applications that work and yield real business value.

It's the Business Technology Optimization revolution.

And Mercury is the only one bringing you a revolutionary solution for J2EE.

Download our free white paper, "Diagnosing J2EE Performance Problems Throughout the Application Lifecycle,"

at [www.mercuryinteractive.com/optimizej2ee](http://www.mercuryinteractive.com/optimizej2ee)

Get Optimized™



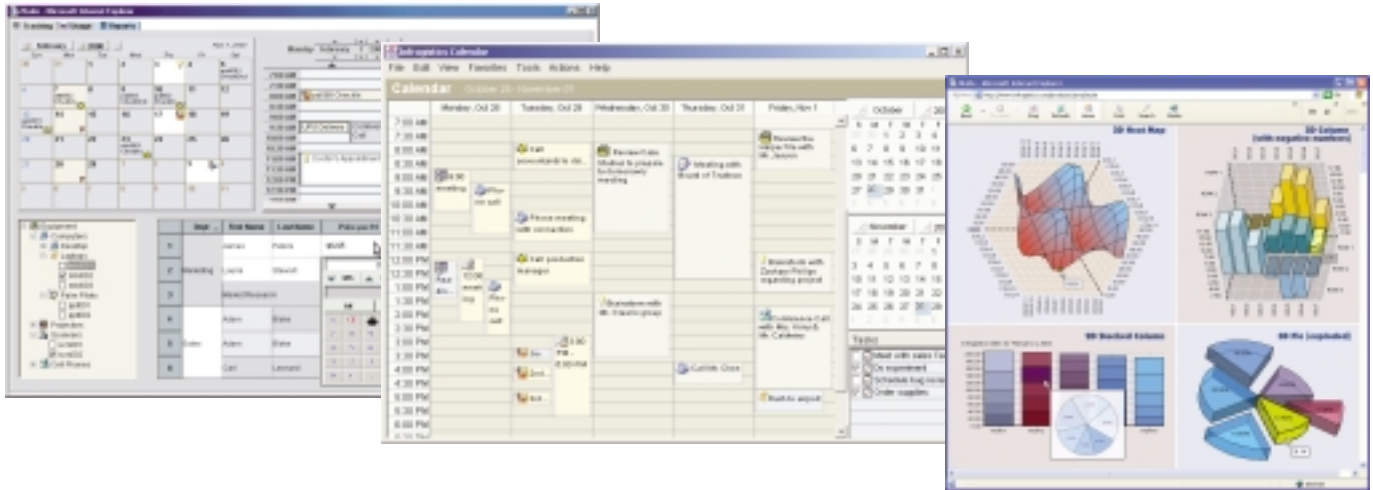






# Infragistics JSuite 7.0

## Tune up Your Presentation Layer!



## The Only Integrated Toolset for Presentation Layer Design

The Infragistics JSuite 7.0 is the only comprehensive framework of presentation layer components you will ever need for building visually superior front-end interfaces for any Java application. Create robust, eye-pleasing client- or server-side applications with the sophistication and usability of today's most recognized commercial applications.

<b>Data Models</b>	<b>Gantt</b>
<b>Tables/Grids</b>	<b>Editors</b>
<b>Charting</b>	<b>Other GUI Controls</b>
<b>Trees</b>	<b>Applet Wrappers</b>
<b>Schedule</b>	<b>Server Classes</b>
<b>Explorer UI</b>	

## AWT JFC JavaBeans™ Server

### JSuite 7.0

#### Enterprise Edition

includes AWT, JFC, JavaBeans & Server product, subscription, as well as guaranteed priority support

è **includes ALL Java source code** **\$1495**

### JSuite 7.0

#### with annual subscription

includes AWT, JFC, JavaBeans & Server a year's worth of updates, upgrades and new products

è **includes ALL Java source code** **\$995**

### JSuite 7.0

#### product only

includes AWT, JFC, JavaBeans & Server **\$795**

Java Source Code is included in the JSuite 7.0 with Subscription or Enterprise Edition!

### Version 7.0 includes ALL NEW client- & server-side charting featuring ONE core charting engine:

- Both client- & server-side charts feature the same rich look-and-feel and object model.
- 2D/3D chart types - All major chart types: bar, stacked bar, column, stacked column, line, area, pie, scatter, bubble, heatmap, candle, hi/low, and open/close financial.
- All charts are fully customizable and extensible.
- Unique session-based security for protecting rendered chart data, making any and all sensitive charts secure.
- Bound and unbound modes, robust aesthetic enhancements, advanced legends, labels and tooltips, client- & server-side events, and much, much more!
- All source code for Java components (Subscription and Enterprise Editions)
- Licensed per developer (multiple installs for non-concurrent use). No runtime or server deployment charges
- RAD customizers maximize productivity, minimize code
- Create robust server-based thin client applications for the web with virtually the same rich user experience found in thick client implementation
- Create user interfaces connected to Web Services using XML and SOAP for seamless application integration

**Download a free, full-featured trial version!**  
**Order online, or contact us!**

**www.infragistics.com**

**800-231-8588**

**Infragistics™**



# Non-Stop EJB Services

DEPLOY NEW RELEASES AT YOUR LEISURE

by Joe Bradley and David Raal

**S**ervice-oriented architectures (SOA) provide numerous benefits: reuse of business logic by many clients, location transparency of business logic, simplified unit testing, better scalability through distributed and load-balanced processing, and the composition of new services from existing services. Enterprise JavaBeans are a favorite platform on which to base service-oriented architectures because of their enterprise-class features.

As many new SOA applications are now developed on the J2EE platform, a problem arises: how to maintain 100% availability while deploying maintenance fixes and new versions of the services. Most application server vendors do not recommend hot deployment of applications in production; problems may occur with unloading classes, class loaders, and resources being used by existing deployments. Instead, the vendors recommend restarting the server or cluster of servers after a redeployment; however, the total time to redeploy, test, and restart a cluster of servers can be substantial. This downtime is unacceptable for many production sites due to loss of revenue and customer goodwill, and the mission-critical nature of the services.

A solution to this problem is to provide a mechanism for dynamically switching clients from a cluster of application servers running the old version to another cluster of application servers running the new version. We refer to this as dynamic cluster switching. This can be accomplished by some enhancements to commonly used J2EE patterns in conjunction with JMS-based messaging. The result is that most deployments of new releases can be made without interruption of services to the client.

Why bother with non-stop EJB services? We have all experienced the issues associated with EJB application upgrades and deployments, such as unexpected outages due to limited testing, poor fall-back strategies, and planned downtime for maintenance in the wee hours. For businesses selling commodity goods and services on the Web, downtime directly translates to lost revenue when customers can easily surf to other sites to buy the same product. With non-stop EJB services, you can reduce if not eliminate downtime while seamlessly rolling out new versions of your services. Imagine redeploying and upgrading your EJBs without impacting your clients and their

Web sites, Web services, consumers, and business partners. If there are issues with the new software, fallback is low-risk and easy to accomplish. All of this occurs during peak traffic periods when using non-stop EJB services. This article describes how this was accomplished on a large consumer Web site handling over 15,000 concurrent sessions during peak times.

## Dynamic Cluster Switching

This solution uses JMS messaging to control a plug-in used by clients of the EJB services. When it's time to release a new version of software, an alternate cluster of servers is deployed with the new code on the same hardware platform as the existing servers. A console or command-line program publishes a "cluster switch" message to the client plug-ins that subscribe to a JMS admin topic. The client plug-ins then start to open connections to the new cluster and allow connections to the old cluster to "die off" as sessions or connections are released. In a short time, all the client plug-ins are seamlessly connected to the new cluster. While this approach sounds simple at a high level, the implementation needs the confluence of many design patterns to be successful in practice.

A basic assumption to this solution is that the EJB services are deployed as their own J2EE application, independent of any Web components or J2EE application clients. In environments requiring performance, flexibility, scalability, and reliability, this is likely to be the case anyway.

## Implementation

The implementation of the solution uses several common design patterns and enhancements in combination with JMS messaging. The particular patterns used are Service Locator, Business Delegate, Publish/Subscribe Messaging, and Observer. Figure 1 provides a graphical depiction of how the various components and message flows work together to perform the cluster switch.

Business Delegates are the client's proxy to the services. They use a Service Locator to obtain an EJBHome object and subsequently create a remote reference to an EJB. To be able to create Business Delegates that point to a different cluster of servers, the Service Locator needs to change its provider URL where it looks up EJBHome objects. This can be accomplished by having the Service Locator receive an update configuration message on a JMS topic that contains the new provider URL.

Clients using existing Business Delegates are not affected and their existing remote references to EJBs continue to operate. As the sessions for these clients expire, the remote refer-





Rational. software

See yourself writing code.  
See yourself writing better code.  
See yourself writing better code, faster.

# Can you see it?

You want modeling, diagnostics, debugging, configuration management, all of your development tools—invisibly embedded within your IDE. You want to collaborate seamlessly with your team and eliminate wasted time. Bottom line, you want to write better code faster. You want IBM Rational software. For a free CD with demos, visit [ibm.com/rational/seamless](http://ibm.com/rational/seamless).

 business on demand™ software

**IBM**

IBM, the e-business logo and e-business on demand are registered trademarks or trademarks of International Business Machines Corporation in the United States and/or other countries. ©2005 IBM Corporation. Rational is a trademark of International Business Machines Corporation and Rational Software Corporation in the United States, other countries or both. All rights reserved.





**Joe Bradley** has worked as a senior Java architect with Sun Software Services for the past six years. He has over 18 years of industry experience in architecture and development of distributed enterprise applications as well as scientific modeling and simulation applications.

joe.bradley@sun.com



**David Raal** is a software architect with the North Highland Company with experience in designing and building complex multitier distributed systems using Java, J2EE, CORBA, and C++. Recently, David has focused on creating e-commerce systems in the manufacturing, telecommunications, hospitality, and retail industries on the J2EE platform.

draal@northhighland.com

ences are released and their Business Delegates are garbage collected. New Business Delegates that were created after the Service Locator received the update configuration message are in effect pointing to the cluster identified by the updated provider URL. This is because their EJB handle was created from EJBHome objects looked up at the updated URL.

A JMS subscriber receives update configuration messages and passes them on to a MultiCaster. The MultiCaster becomes the sole point in the client VM for receiving these messages and distributing them to interested components. When the client code first loads, the JMS subscriber is initialized and components, such as the Service Locator, register with the MultiCaster for the type of messages they wish to receive.

A simple command-line program can be used to generate the JMS message that initiates the cluster switch, or this functionality could be part of a more comprehensive management and monitoring console application. The publish-subscribe paradigm is important here because any number of clients can be dynamically reconfigured through their connection to a JMS topic. This approach supports the management of a dynamic and ever-changing set of clients connected to the EJB servers.

Figure 2 is a class diagram of implementations of the various components and patterns. The source code for this article can be downloaded from [www.sys-con.com/java/sourcec.cfm](http://www.sys-con.com/java/sourcec.cfm). The code should be considered fragments, intended only to illustrate the points in this article since it's missing important features such as logging, exception handling, and configurability. The more important classes will now be discussed in detail.

#### Service Locator

The Service Locator pattern, as described in *Core J2EE Patterns*, abstracts all JNDI usage, hides the details of initial context creation as well as EJBHome lookup, and caches EJBHomes for performance reasons. The Service Locator is usually made a singleton so that all clients can access the same EJBHome cache.

For the Service Locator to receive update configuration messages, it must register with the MultiCaster when first loaded. When a message is received, the Service Locator replaces its local copy of the provider URL and the initial context factory class with those obtained from the message. Subsequently, it invalidates its current cache of EJBHome objects. Then, the next time a Business Delegate asks for the EJBHome, it won't be found in the cache and will be looked up at the new provider URL. Once looked up, the new EJBHome object will be placed in the cache.

The implementation of the Service Locator provided in the source code is named *ClientServiceLocator*. As the name indicates, there may be other Service Locators in an application for use in other layers of the architecture (e.g., Services, Foundation, etc.).

#### Business Delegate (BD)

The Business Delegate pattern hides the details of connecting to and using an EJB. Typically each business method in an EJB has a corresponding method in the Business Delegate that delegates client invocations to the EJB. The Business Delegate catches all the exceptions that can result from communicating with an EJB and turns them into application-specific exceptions. It allows clients to use the services as if they were local, and is thus a client-side proxy for a service. Business delegates can also be used to cache frequently requested data and provide other similar performance improvements to the services.

In addition to the normal responsibilities ascribed to the Business Delegate, the following additional responsibilities are required to support continuous availability of services:

1. The BD must automatically perform a client/server ver-

sion compatibility check. The first time a remote reference is retrieved by a business delegate, the client version must be compared to the server version to ensure compatibility. If incompatible, the business delegate must return a specific exception on compatibility mismatch that can be caught by a client. The exception should be logged by the client in the form of an informative error message. This provides a quick indication to support personnel that the client view JAR file is out of date. Without this check, a serialization error will result if the client and server classes are incompatible, and the source of the error will not be obvious to support personnel.

2. The BD provides a *create()* and *release()* method for use by the client. Typically the Business Delegate Factory invokes the *create* method so the client doesn't need to. The client should always call the *release* method, however, when finished with a Business Delegate. For Web component clients (servlets and JSP pages), assuming the BD has been placed in the session, this can be accomplished by catching HTTP session timeouts with the *HttpSessionBindingListener* interface. The *release* method not only invokes *remove()* on the Business Delegate's EJB remote reference, but a *BusinessDelegateReleasedMsg* is sent to the MultiCaster. The MultiCaster in turn notifies objects that have registered to receive this event, notably the Business Delegate Factories. The use of this event by the Business Delegate Factory is described in the next section.

The above responsibilities are implemented in the *BusinessDelegate* base class and should be extended by each Business Delegate in an application. All the business methods of each Business Delegate subclass typically invoke the inherited *getService* method to obtain the remote reference. Rather than store a remote reference to an EJB, which is not guaranteed to be serializable by the EJB specification, *BusinessDele-*

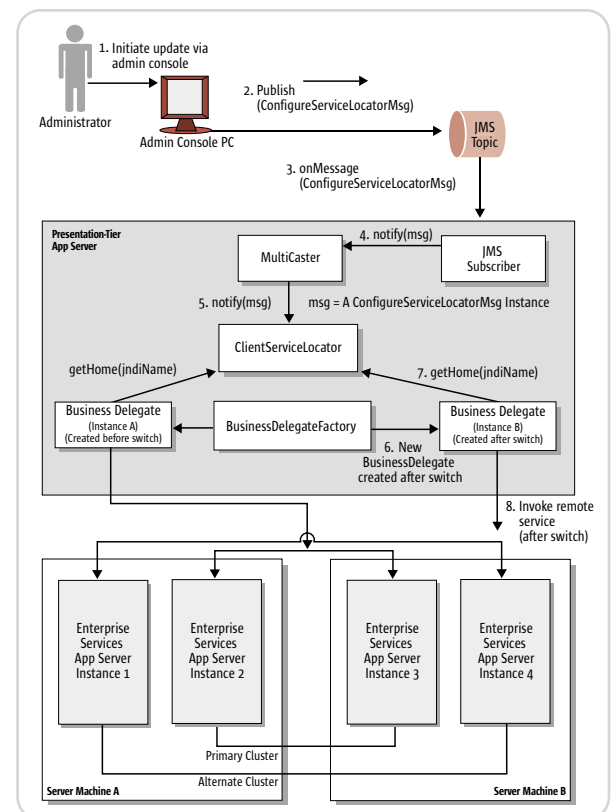


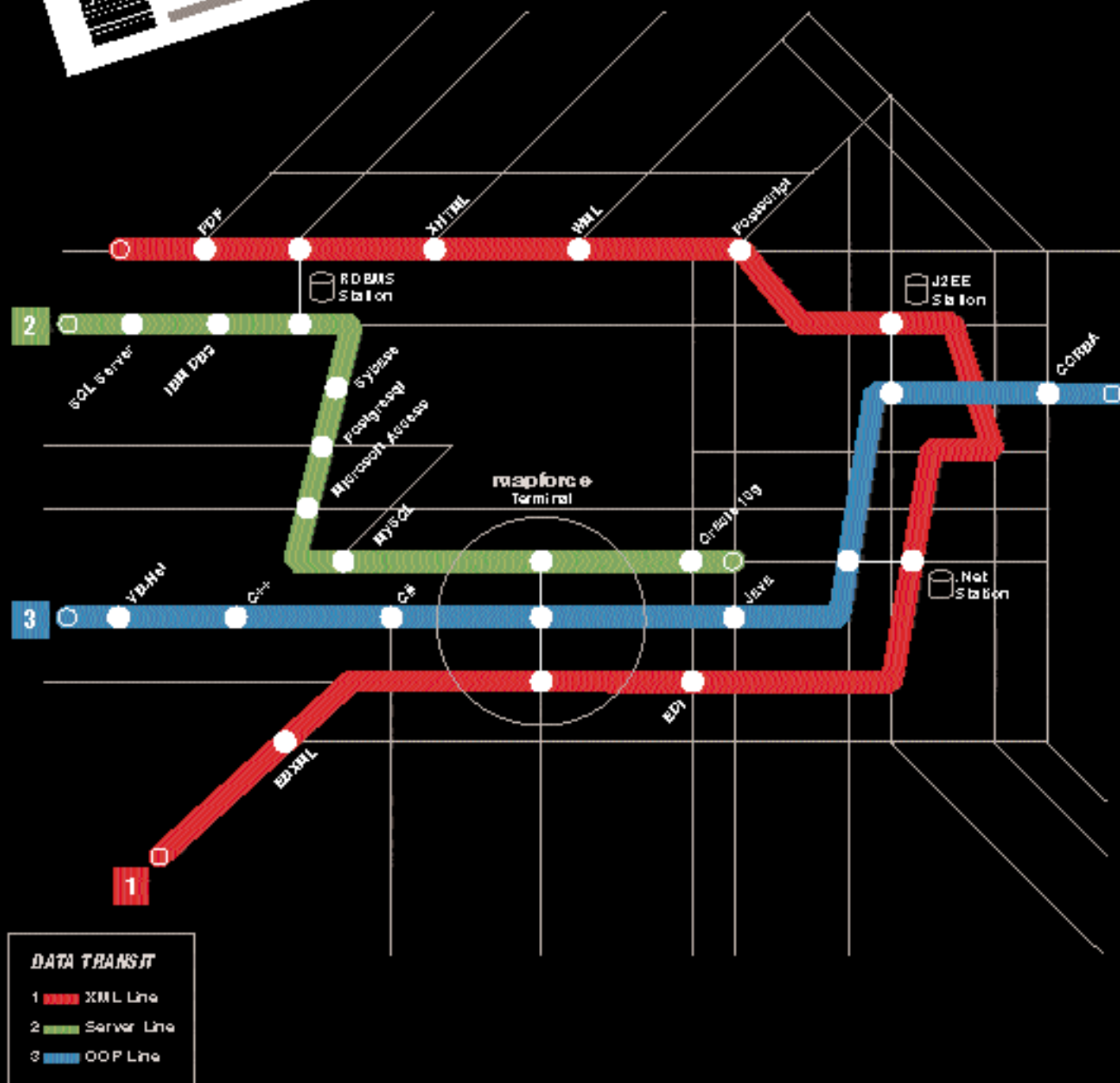
Figure 1 Cluster switch





mapforce™  
2004

Best of Show  
Map Software



Introducing **Mapforce 2004**: A powerful new XOL mapping tool from Altova, producer of the industry standard XOL Development Environment, XOLSPY. Mapforce is a visual data mapping tool, which auto-generates custom mapping code in multiple output languages such as XSLT, C++, C#, and Java, to enable programmatic XOL-to-XOL or XOL-to-Database data transformations. Mapforce 2004 is the hassle-free transit for moving data from one format to another, enabling efficient information reuse. Download a free trial now!

# ALTOVA

gate stores the EJB Handle. `getService()` reconstitutes the remote reference from the EJB Handle on each invocation in case the Business Delegate has been serialized to another server in the cluster between invocations.

### Business Delegate Factory

A Business Delegate Factory is used primarily because it provides the flexibility to hand out other implementations of the Business Delegates depending on the type of client. It also enables a total count to be kept of the number of Business Delegates of each type that have been handed out, as well as a running count of the current number of outstanding Business Delegates.

A subclass of `BusinessDelegateFactory` should be created for each Business Delegate in an application and a singleton should be created for it. The singleton should register with the MultiCaster to receive Business Delegate release messages for the corresponding Business Delegate type. The management of the counters and the reporting of the counts is all inherited from the `BusinessDelegateFactory` base class. The specific mechanism for reporting the counts is outside the scope of this article but could be reported by a JMX agent or published to a JMS topic.

### MultiCaster

The MultiCaster is the central player in the implementation of the Observer pattern. Observers register with the MultiCaster, providing a filter implementation. When the MultiCaster is notified of an event, it applies all filters to it and notifies observers (subscribers) who have matching filters for the event.

The role of the MultiCaster is to deliver Business Delegate–released notifications to each subclass of `BusinessDelegateFactory`, as well as deliver update configuration messages to the Service Locator that was received on a JMS topic.

To receive notifications that a Business Delegate has been released, each subclass of `BusinessDelegateFactory` adds itself as an observer to the MultiCaster with a filter type of `BusinessDelegateReleasedFilter`. This filter type checks to see that the published object is of type `BusinessDelegateReleasedMsg`, and that the BD name in the message is the same as that with which the filter was constructed. This causes each `BusinessDelegateFactory` to receive release notifications only for the type of Business Delegates it creates.

To receive update configuration messages, the Service Locator adds itself as an observer to the MultiCaster with a filter type of `UpdateServiceLocatorFilter`. This filter type checks to see that the published object is of type `ConfigureServiceLocatorMsg`.

## Two Levels of Client Redirection

The solution presented in this article redirects new clients of the services to the new version of the services. Existing clients using the old version are left to slowly bleed off as their sessions expire. A modification to the solution could be made to immediately switch all existing clients of the services to the new version as well. This would mean that every Business Delegate registering with the MultiCaster would receive Service Locator reconfigured messages, which the Service Locator would have to publish after reconfiguration was complete. This enhancement would also involve the additional complication of managing access to BD instances by multiple threads since the client thread using the BD would be distinct from the thread used by the MultiCaster to deliver event notifications to the BD.

## Procedure for Cluster Switch

Now that the architecture of the solution that enables an application for dynamic cluster switching has been presented, we'll discuss the procedure for actually performing a switch. While the procedure might seem obvious, experience has shown the obvious approach is not necessarily the best.

Recall that one of the assumptions stated at the beginning of this article is that clients of the services are running in separate containers from the services. This means that those clients will be using a client view JAR file that has all the classes necessary to be a client of the services. Included in that client view JAR file are configuration resources that point the Business Delegates to a specific application server cluster (subsequently called the “primary” cluster). Assume the new version of the services is deployed to the “alternate” cluster and clients are switched there. It's not unreasonable to assume that at some point, days or weeks later, the client environment (such as a Web container) may need to be restarted. In that case, the clients will get their configuration from their existing client view JAR file, which is pointing to the primary cluster. But the latest services are running on the alternate cluster.

The procedure we've been using in production to solve this problem is as follows:

1. Boot the alternate cluster.
2. Deploy the old services to the alternate cluster.
3. Run regression tests to verify the services are functioning as expected on the alternate cluster.
4. Issue a cluster switch to clients to point them to the alternate cluster.
5. Enable trace-level logging in the old services in the primary cluster to ascertain when existing sessions have bled off the primary cluster. An admin console that is able to query and display the outstanding BD counts from the Business Delegate Factories can also be used as a cross check.
6. Remove the old services from the primary cluster and deploy the new ones to it.
7. Run regression tests against the new services on the primary cluster.

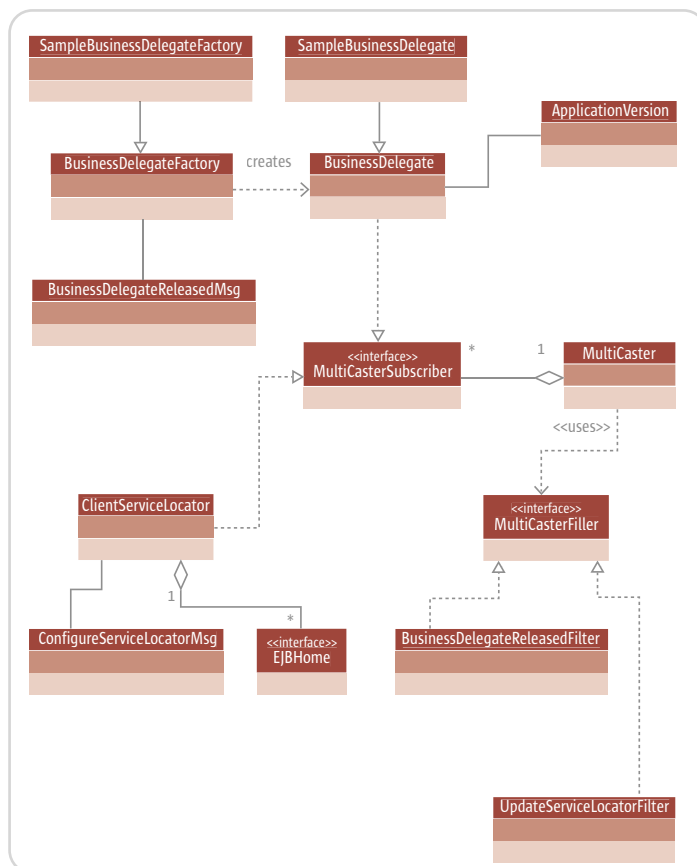
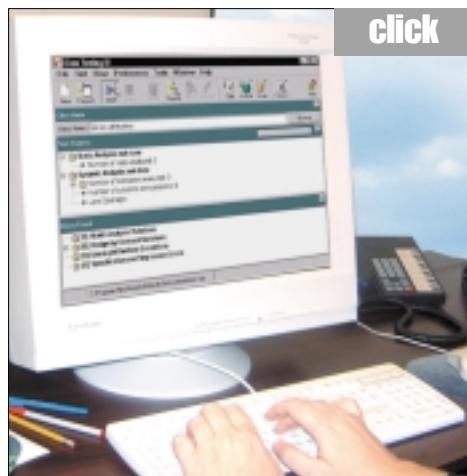
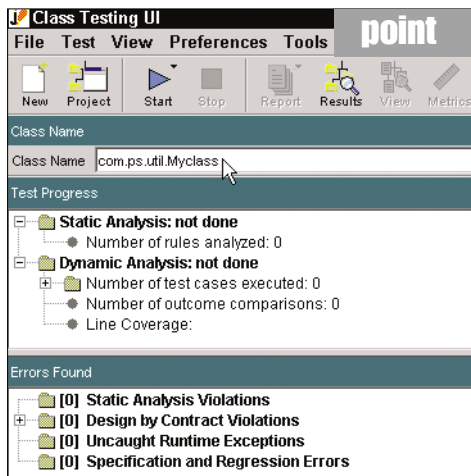


Figure 2 Implementation class diagram



# Automate unit test case generation for JUnit and Java™ with Parasoft Jtest.



**(It's as easy as 1-2-3.)**

**Parasoft Jtest is the first and only  
automated unit testing tool for Java™ development.**

With just a click, Jtest reads and analyzes code — quickly creating harnesses, stubs and test inputs — and tests without user intervention. Jtest also enables you to automate regression testing and static analysis. For loyal JUnit users, Jtest is designed to fully support existing test cases and automate the creation of new JUnit-compatible test cases.

**Learn how Jtest can enhance JUnit capabilities...**

Download a free eval copy of Jtest along with our informative new white paper entitled "Using Jtest With JUnit."

----- **For Downloads go to [www.parasoft.com/jdj\\_12](http://www.parasoft.com/jdj_12). Or call 888-305-0041.**

Copyright ©2003 Parasoft Corporation. All rights reserved. All Parasoft product names are trademarks or registered trademarks of Parasoft Corporation in the United States and other countries. All other marks are the property of their respective owners.

**New for Java development...**

**Parasoft® Jtest® 5.0**

**Features:**

- Fully integrated into JUnit & Eclipse
- Quick Fix enabled finds & fixes errors fast
- Intuitively designed GUI

**Platforms:**

Linux

Solaris

Windows 2000/XP

**A part of Parasoft Automated  
Error Prevention (AEP) Solutions and Services**

**PARASOFT**  
*jtest*®

8. Issue a cluster switch to clients to point them at the primary cluster.
9. Monitor old services on the alternate cluster to determine when incoming traffic has stopped.
10. Shut down the alternate cluster.

In summary, two switches are performed. New clients are first switched to the old code on the alternate cluster, and then subsequently new clients are switched to the new code on the primary cluster. With an HTTP session timeout of 15 minutes on an e-commerce-related site, the authors have found that letting the traffic bleed off after both cluster switches generally takes a total of three hours. Obviously this number may vary greatly depending on the nature of the services. Three hours is thus the total time that both application server clusters must be active, potentially straining resources such as memory, CPU, and connection pools if both clusters are run in a single hardware environment.

### Service Compatibility

A caveat to dynamic cluster switching is that if a change in the public API of the services would cause a serialization or marshaling error between clients using old classes and the new services, the switch cannot be performed. Clients will have to shut down to upgrade their client view JAR files to the new version.

Minimizing the frequency of incompatible builds requires careful attention to application and object versioning. The Java Object Serialization Specification describes exactly what changes to a class make it incompatible with previous versions with regards to serialization. A technique that maximizes long term compatibility of class versions is to manually control their Stream Unique Identifier (SUID).

It's also recommended that a compatibility version number be added to the overall version number for the application. The

version number must be made available to clients through the service API so that the BusinessDelegate base class can automatically retrieve it the first time a Business Delegate of each type is used. At that point, the version number in the client view JAR file is compared with the value returned from the service, and a difference in the compatibility number causes an exception to be thrown to the client. This mechanism can be seen in the BusinessDelegate code fragment in the source code.


### Conclusion

This solution enables you to deploy new releases into production at leisure. A full regression test can be run on the newly deployed services before putting them into production. Care can be taken to assure that the deployment is perfect since there is no time pressure due to a production outage.

We have used the solution presented here to push a half-dozen new releases into production over the past six months at one of the top revenue-generating Web sites. At this particular site, 75% of the new releases of the services have been compatible builds for which this technique was successfully applied. ☺


### References

- Stevens, M. (2002). "The Benefits of a Service-Oriented Architecture." 5 March 2003: [www.developer.com/java/ent/article.php/1041191](http://www.developer.com/java/ent/article.php/1041191)
- Alur, D., Crupi, J., and Malks, D. (2001). *Core J2EE Patterns: Best Practices and Design Strategies*. Prentice Hall PTR.
- The Portland Pattern Repository. "MultiCaster." 5 March 2003: <http://c2.com/cgi/wiki?MultiCaster>
- Sun Microsystems, Inc. Java Object Serialization Specification. 5 March 2003: <http://java.sun.com/j2se/1.3/docs/guide/serialization/spec/serialTOC.doc.html>




# REPORTS!

**Formula One from ReportingEngines**  
**Embedded reporting toolsets for J2EE application servers**



- Embed reports in your applications with Pure Java toolsets that fully integrate with your IDE and leverage your existing application server infrastructure
- No separate report server process to learn, set up, or maintain
- Point, click, and design reports from Java objects, JDBC, and XML
- Easy-to-understand developer pricing
- Full APIs and visual report designers



**Free Trials:** [www.ReportingEngines.com/info/trial7.jsp](http://www.ReportingEngines.com/info/trial7.jsp)  
 888-884-8665 • [sales@ReportingEngines.com](mailto:sales@ReportingEngines.com)

Copyright 2003 ReportingEngines, a division of InetSoft Corporation. All rights reserved. Formula One is a registered trademark of InetSoft Corporation. Java and Java-based trademarks and logos are the trademarks or registered trademarks of Sun Microsystems, Inc. in the United States and other countries. All other trademarks are property of their respective owners. All specifications subject to change without notice.



## Announcing Reactor 5.5!

# Welcome To Workflow Nirvana

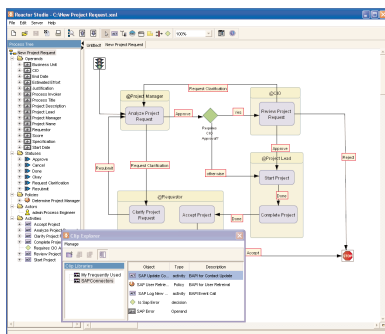
***The Functionality You Want***  
***The Ease of Use Your Business Users Demand***



Already the perfect workflow engine for developers seeking **control, extensibility, and platform neutrality**, Reactor 5.5 is now the perfect tool for business users, giving them the ease of use they demand.

J2EE-based, XML-driven and **available as royalty-free source code**, Reactor 5.5 frees you to follow the path to true enlightenment (and killer gains in productivity) by providing:

- A single, scalable process platform for defining and linking enterprise-wide workflows
- Extensible libraries of re-useable process components
- A user-friendly tool for point-and-click development of custom workflow solutions.



Process design tool, Reactor Studio, enables quick development and deployment of workflows.

So light some incense, find your mantra and focus your chi. Because Reactor 5.5 is here to take you to Workflow Nirvana. **Download a free evaluation copy at [www.oakgrovesystems.com/jdj](http://www.oakgrovesystems.com/jdj) or call us at (818) 440-1234.**

**We'd love to help you...**

## Declare Your Workflow Independence!™



**oakgrove**  
systems



**Jason Bell**  
J2SE Editor

## A Christmas Wish List

I may not believe in the existence of someone who can span the globe in a number of hours, along with a collection of antler-based creatures (one with a red nose, the others not). However, it doesn't stop me from making a list of stuff that I want for Christmas. Apologies in advance if you do not partake in these celebrations.

- **JSP Support for Eclipse**

If there's one thing that Eclipse is in desperate need of it's better support for Java Web applications. It's not just JavaServer Page creation but also neat things like a .war builder and the construction of the configuration files that go with it. I've built Ant tasks that create the .war files and copied them to the desired location. That's all very well, but after all this time I thought the Eclipse guys (and everyone else who writes plugins for the application) would have sorted this one out. It's odd that some folks have built a PHP plug-in.

Now before everyone writes in and tells me at great length that IDEA already does that; yes, I know. There's a small note of difference between free and lots of \$\$\$ (something I don't have).

- **Blogs with Content, Not Links to Everyone Else's Content**

Take this as advice: if you're going to write a blog, then write your own content. Linking off to everyone else's content is fine, but please accept the fact that, chances are, I've read it already from an aggregator of some form.

If 2003 was the year of anything, it was the year of the blog. I've never seen so many blogging tools, libraries, aggregators, and utilities come out in such a short space of time. Was it worth it? Well, the jury is still out on that one. I think there's an ebb and flow on these sorts of things. I certainly don't read them as much as I used to.

- **For People to Read the API Docs**

I only ask for the small things in life. If I have to utter the words "equals()"

and "String class" in either an e-mail or IRC channel, I think I will scream. That and you'd think that Alan would know this...(just joking E-in-C :)).

It may seem trivial but it is important. If you've read my previous editorials, these things are close to my heart. I don't care what the project is, no matter how small it should be documented one way or another. Javadoc helps a lot and it's pretty easy to use, so there should be no excuse for not creating it let alone reading the existing stuff. Don't even get me started on unit testing. If you're going to bookmark two things, make it the API doc for the JDK you are using and a link to the Java Almanac site. If you get stuck, those two alone should help you on your way.

- **Everyone Should Attempt to Write a Class Before Getting Apache Commons to Do It for Them**

Yes, I know it sounds really harsh, but this is something I know from bitter experience. Let me give you an example. Reading an XML configuration file, everyone seems to jump on Commons Digester, which I assume you'd only ever use 10% of the API. The result is one huge dependency to do one small task. The whole thing can be done with some clever DOM coding; it's a case of taking the time. If you don't know it that well, ask someone (which is what I did).

- **A New Look to JDJ**

Now I know I can dream and wish all I want...but in fact it is happening. Next month we move away from "J2SE" to a new section, "Core and Internals." To start off, we'll have an introduction to object orientation and also an article on the Java Collections framework. As ever I am on the lookout for articles, plus I'm also looking for shorter articles that get the message across on how to solve particular problems. If you think you have something that would be helpful to everyone, let us know. ☺

Jason Bell is the senior programmer for a B2B portal. He's also a keen supporter of people reading the API docs before asking questions. In his spare time he's involved with building RSS development tools.

jasonbell@sys-con.com



### A Christmas Wish List

I may not believe in the existence of someone who can span the globe in a number of hours, along with a collection of antler-based creatures (one with a red nose, the others not); however, it doesn't stop me from making a list of stuff that I want for Christmas.

#### PircBot 1.2.5 Java IRC API

Internet Relay Chat (IRC) is a system that allows groups of people to collaborate and chat from anywhere in the world. Clearly defined by several RFC documents, it's arguably the most standard real-time chat system currently in use. PircBot is a Java framework for writing IRC clients quickly and easily.



# Get the complete picture



## Features, Performance and Control

### Discover the ILOG JViews Graphics Components

You're developing a sophisticated user interface for a desktop, applet or servlet application – it needs to provide displays that go far beyond what Swing and HTML offer. How can you be sure it will have the features, performance, customization and scalability to enable your end-users to make better more informed decisions, faster?

With ILOG JViews, you get comprehensive graphical libraries & tools, resources, and maintenance services so you can focus on the implementation, confidently completing your application in less time and at less cost.

Quickly and easily build:

- Gantt and resource displays
- Graph layouts, diagrams, workflows
- Geographic map displays
- Realtime data charts
- Custom monitoring and control screens
- Network and equipment management screens

**Get a JViews Info Kit – Learn more, test drive an Eval.**  
**Go to: [jviews-info-kit.ilog.com](http://jviews-info-kit.ilog.com) or Call: 1-800-for-ILOG**



Changing the rules of business™



Paul Mutton

# PircBot 1.2.5 Java IRC API

Have fun with Java

Internet Relay Chat (IRC) is a system that allows groups of people to collaborate and chat from anywhere in the world. Clearly defined by several RFC documents, it's arguably the most standard real-time chat system currently in use. An IRC network consists of a set of servers that people can use to connect to IRC. Typically each network is comprised of several servers to help increase the performance and resilience of the system.

In addition to being able to send messages directly from one user to another, IRC users can join a set of channels, which are analogous to rooms. Each channel has a unique name and is usually inhabited by users with a common interest. These users may participate in the shared discussions.

Since its introduction by Jarkko Oikarinen in 1988, IRC has steadily grown in popularity and currently has more than a million users worldwide at any one moment. The IRC protocol was clearly defined five years later in RFC 1459, making the system more accessible. As a result of this, there are now many client programs that allow users to connect to an IRC network. Some of these client programs are called *bots*, a term commonly used to describe an automated IRC client (bot is a contraction of robot).

PircBot is a Java framework for writing IRC clients quickly and easily. It has an event-driven architecture to handle common IRC events, flood protection, DCC resuming, ident, and more. Its comprehensive log file format is suitable for use with programs that generate channel statistics for communities. PircBot can be used to create standalone IRC bots and clients, or be rapidly incorporated into any existing Java program, adding useful functionality while avoiding the need to reinvent the wheel. It can also play an important role in education by making it fun to learn Java. Several unique ideas for both research and plain fun have been implemented using the PircBot package.

Creating your first IRC bot with the PircBot framework takes only a couple of minutes. Simply download PircBot and its documentation from [www.jibble.org/pircbot.php](http://www.jibble.org/pircbot.php) and import the `org.jibble.pircbot` package in the classes that will be using it. This package contains an abstract class named `PircBot`, which implements common IRC functionality and leaves the rest up to you. Several other classes in the package allow easy access to more advanced features, such as file transfers. Writing a bot is a simple case of extending the abstract `PircBot` class.

```
import org.jibble.pircbot.*;

public class MyBot extends PircBot {

    public MyBot(String name) {
        this.setName(name);
    }
}
```

All methods in the `PircBot` class are fully detailed in the supplied Javadoc documentation, with examples of usage where appropriate. The first methods that you're likely to use are the `connect` methods, which allow you to connect to an IRC server (with an option to specify the port number or a password to join password-protected servers). Another prominent method is the `sendMessage` method, which lets you send messages to other users or channels.

After connecting to an IRC server, users typically join one or more channels, each of which contains a set of users. Messages sent to a channel can only be seen by those in that channel. PircBot automatically monitors the channels and users that it has seen, so it's possible to discover which channels your bot is in and, more usefully, who is in each of those channels. This provides important functionality for channel management, where a bot may be responsible for guarding a particular channel against misuse from spamming



Figure 1 Statistics generated from a PircBot log file

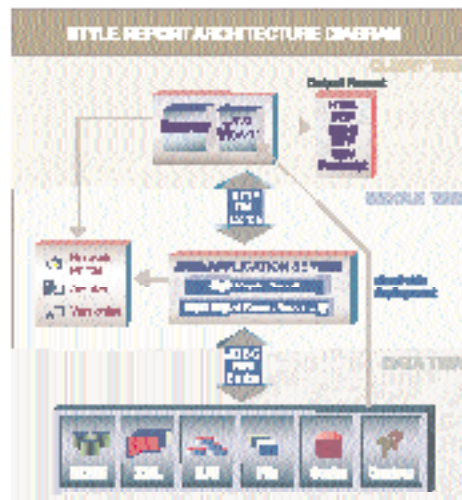
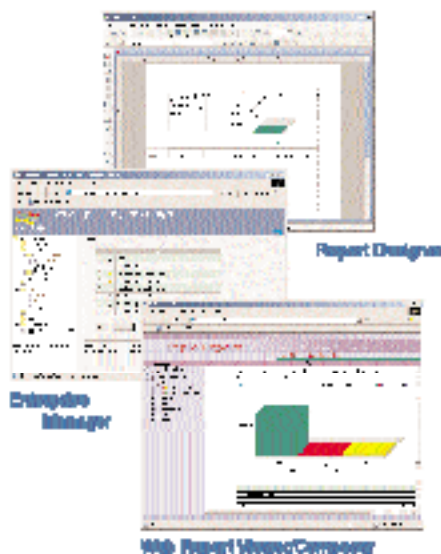


# Java & Web Reporting: with Java flexibility without Java complexity

## Style Report: Information Vantage Point



Style Report blends the best of Java and Web to provide an award winning Web enterprise reporting solution. Based 100% on open standards such as XML, JSP, Java and Web services, Style Report allows you fully leverage existing IT infrastructure and common technical skills. One Report, Multiple Channel delivery and zero client ad-hoc reporting gives end users the freedom and frees IT from time consuming repetitive work. Designed for integration, Style Report is deployed and managed as a standard Web application or Carba/RMI service requiring not special skills. Download a free evaluation copy now!



### Style Report Features

Flexible Visual Designer with JavaScript or VB Script	✓
PDF, Excel, Word, CSV and other Export Format	✓
Database/XML and other Data sources	✓
Zero Client Browser Viewing	✓
Built-In Drilldown/Parameters	✓
Zero Client Ad-hoc Reports	✓
Virtual Private Security Model	✓
J2EE Drop-In Deployment	✓
Advanced Scheduling	✓
Report Versioning/Archive	✓
Demand Paging/Report Cache	✓
Portal/JSP Integration	✓

or some other abuse. Channel management with PircBot is easy, as it becomes trivial to monitor topic and mode changes or even grant operator status to other users.

```
1 MyBot bot = new MyBot("Billy");
2 bot.setVerbose(true);
3 bot.connect("irc.freenode.net");
4 bot.joinChannel("#bots");
5 bot.sendMessage("#bots", "Hello!");
```

Having created the MyBot class, it's not too hard to get it to join a channel

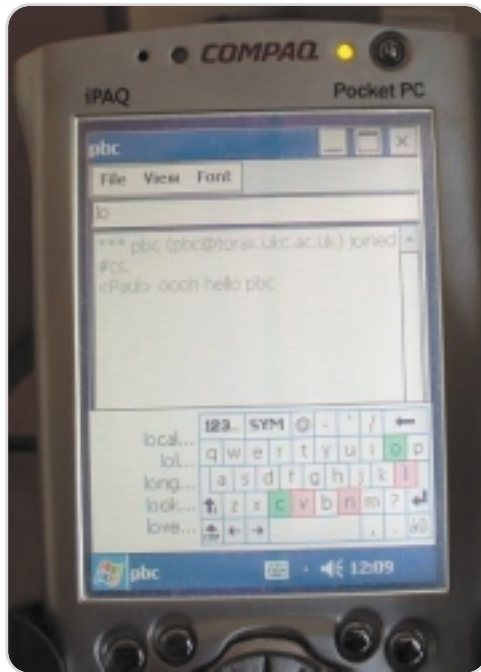


Figure 2 PircBot Client running on an iPAQ

on an IRC server. The preceding code demonstrates how to get the bot to connect to the server irc.freenode.net. The connect method can throw an IOException or IrcException, so you'll need to use a try-catch block here, or allow the exceptions to propagate if appropriate. After the bot has connected to the server, it will join the channel #bots and send a message to all users in that channel. Take note of line 2, where the verbose mode has been activated. This causes all events to and from the server to be logged to the standard output, which is useful for debugging purposes. The destination of this output can be changed by overriding the log method, inherited from the PircBot class.

An "out of the box" feature that's included in PircBot is the ability to create log files by redirecting the default verbose output to a file. Such log files can then be easily processed by tools like plog (Perl IRC Statistics Generator) to generate interesting and often amusing statistics about a channel (see Figure 1).

### Event-Driven Architecture

As the term "bot" is a contraction of "robot," our IRC bot must be independent of direct human control and able to think for itself. IRC bots usually behave semi-autonomously, performing tasks when certain events occur, or responding to commands supplied by authorized users. PircBot allows you to detect events as and when they happen by overriding the appropriate

method. All event-driven methods in PircBot have names that begin with "on". For example, overriding the onDisconnect method enables us to know when our bot has been disconnected from the IRC server so we can perform remedial actions. The following code shows this event-driven method being overridden.

```
public void onDisconnect() {
    try {
        reconnect();
    }
    catch (Exception e) {
        // Could not reconnect.
        System.out.println(e);
    }
}
```

Another example of the event-driven behavior of PircBot shows how to welcome new users to a channel. When a user joins a channel, PircBot calls its onJoin method. This method doesn't do anything unless you explicitly override it in a subclass. The following code shows how you would use the sendMessage method to publicly welcome newcomers to a channel.

```
public void onJoin(String channel,
    String sender,
    String login,
    String hostname) {
    sendMessage(channel,
        "Hi, " + sender + "!");
}
```

To further enhance the functionality of MyBot, we can make it respond to some simple commands from users. Some care is required when parsing messages from other users, as they may include special characters that are used to render colors and other formatting. These spurious characters can be removed by using the helper methods in the Colors class, leaving just the plain text. The Colors class also allows you to format text using a selection of colors and formatting attributes, including bold, underline, and italic, although the appearance of such formatting will depend on the IRC client you are using to view the channel.

The java.util.regex package is ideal for parsing commands from other users for keywords and their arguments. However, PircBot is compatible with JRE 1.1.8, so be careful which other classes you make use of if you intend to develop an applet or client for small PDA devices where it is essential to

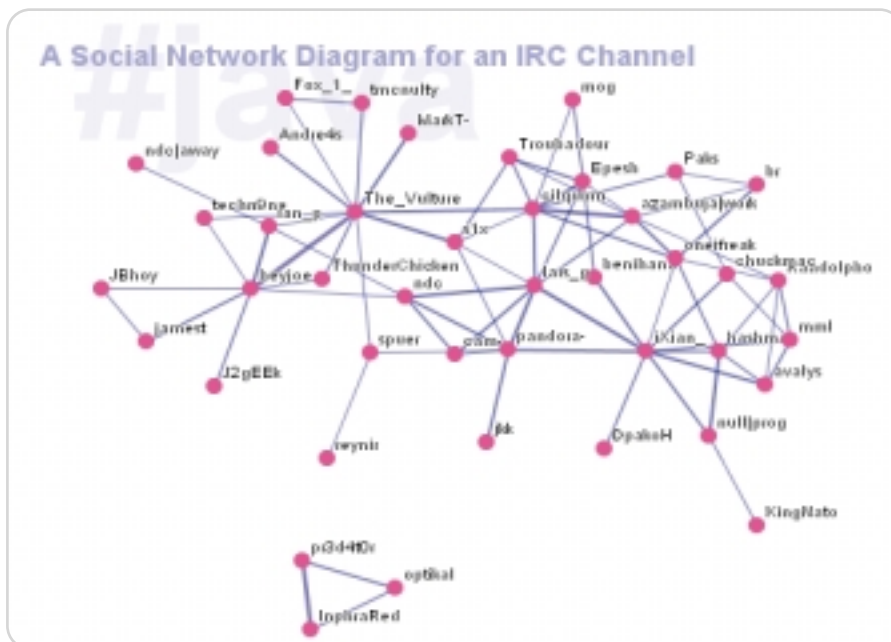


Figure 3 Example output from PieSpy



**WebAppCabaret**<sup>TM</sup>[www.webappcabaret.com](http://www.webappcabaret.com)**J2EE Web Hosting**

Quality Web Hosting at a reasonable price...



# <JAVA WEB HOSTING AND OUTSOURCING>

You have developed the coolest mission-critical application. Now you need to deploy it. Outsource your hosting and infrastructure requirements with WebAppCabaret so you can save time and money and concentrate on other important things. WebAppCabaret is the leading JAVA J2EE Web Hosting Service Provider. From shared hosting to complex multi-dedicated server hosting, WebAppCabaret has the right solution for you. 30 Day Money Back Guarantee and SLA. WebAppCabaret offers the latest Standards based Servlet containers, EJB servers, and JVMs. We provide options such as e-Commerce, EJB 2.x, Failover, and Clustering. Our Tier 1 Data Center ensures the best in availability and performance.

At WebAppCabaret you have the flexibility to choose the LATEST hosting technology best suited for your WEB application or your programming skill. If you are a programmer or consultant, WebAppCabaret has the right hosting solution for your project or client's dynamic web application/services requirements.

## OUTSOURCING:

Do you really need an IT department for your web applications, mail systems, and data backups when we can perform the same functions more efficiently at a fraction of the cost - with competent technical expertise and redundant hosting facilities.

## J2EE HOSTING:

Below is a partial price list of our standard hosting plans. (Reseller accounts also available). For more details please log on to <http://www.webappcabaret.com/jdj.jsp>

### \$39/mo Enterprise

Latest JBoss/Tomcat/Jetty  
Latest JSP/Servlets/EJBs  
Private JVM  
Choice of latest JDKs  
Dedicated IP Address  
NGASI Control Panel  
PHP and Perl  
Web Stats  
900MB Disk  
200MB DB  
MySQL  
PostgreSQL  
Dedicated Apache  
Telnet . SSH . FTP  
5 Domains  
100 Emails  
Web Mail . POP . IMAP  
*more...*

### \$191/mo Dedicated

Managed Dedicated  
Server starts at \$191  
per month for:  
256MB RAM  
Pentium 4  
40GB RAID  
Firewall  
Unlimited Domains  
Unlimited Web Site Hosting  
NGASI Control Panel  
with your own Logo  
Each dedicated  
server configured  
for standalone  
web application  
and web hosting (resellers)  
at no extra charge.  
*more...*

### \$17/mo Professional

Latest Tomcat/Jetty  
Latest JSP/Servlets  
Private JVM  
Choice of latest JDKs  
NGASI Control Panel  
PHP and Perl  
Web Stats  
200MB Disk  
30MB DB  
MySQL  
Telnet . SSH . FTP  
2 Domains  
20 Emails  
Web Mail . POP . IMAP  
*more...*

### \$99/mo Reseller

If you cannot afford a  
Dedicated server for  
reselling web hosting,  
for \$99/mo the Shared  
Reseller plan gives you:

5 Professional Plans  
10 Basic Plans  
NGASI Control Panel  
with your own Logo  
*more...*

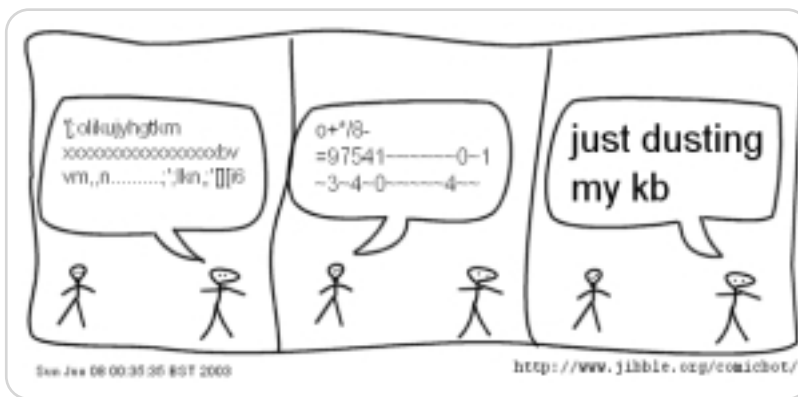


Figure 4 Comic strip produced with ComicBot

restrict yourself to the classes available in Java 1.1.

Most IRC servers impose restrictions on how rapidly data can be sent. The server often disconnects clients if they attempt to send too much data in one go. PircBot provides a solution to this problem by enqueueing all outgoing messages, so single messages are sent as soon as possible, with a minimum delay between each subsequent message. The length of the delay can be set by calling the `setMessageDelay` method.

Sending messages to a user through a network of IRC servers imposes the aforementioned delays; however, another way around this problem is to make use of the DCC (Direct Client to Client) protocol. A DCC request is made via the IRC server as usual, but it asks the recipient to connect directly to your machine to carry out further actions. PircBot supports DCC CHAT, which allows a pair of clients to chat to each other over a direct TCP connection, bypassing the IRC server. DCC SEND is also supported, which allows PircBot to send and receive files.

```
File file = new File("./music.mp3");
DccFileTransfer t = dccSendFile(file,
                                "Dave", 120000);
t.setPacketDelay(100);
```

When PircBot requests to send a file to another client, the recipient can accept the file transfer by connecting directly to the PircBot and downloading the file; so it's important to ensure that there will be no problems posed by NAT or a firewall. The recipient can also choose to resume the download if he or she already has part of the file. The timeout value of 120000 in the above code is the number of milliseconds to wait for the user to accept the request, after which

the server socket will close and the user will lose the chance to download the file.

Each file transfer is represented by a `DccFileTransfer` object. This class allows you to monitor the progress of a file transfer and throttle the speed of the connection if required. Files are sent according to the IRC RFC documentation, that is, in 1KB chunks with acknowledgments. Each `DccFileTransfer` object allows you to include a delay between each "chunk" or "packet." The previous code sample shows a delay of 100 milliseconds being set between each packet, limiting the speed of the transfer to a maximum of 10KB/sec. PircBot can also receive file transfers from other clients. Resuming is supported in both directions, so handling large files is not so problematic on unreliable connections.

Most IRC servers try to establish your identity by contacting the ident server on the machine that you are connecting from. Not everybody runs an ident server, particularly Windows users. A small number of IRC servers will actually refuse to accept connections from clients if they can't find an ident server. PircBot gives you the option of running a "fake" ident server just to appease these types of IRC servers. The ident server is shut down as soon as it has been used.

The PircBot class contains a large number of methods to support and process the IRC protocol as transparently as possible. Some IRC servers implement additional functionality that is not specified in any of the IRC RFC documents. In these circumstances, you'll still be able to make use of the undocumented features by sending raw lines to the server with the `sendRawLine` method (this method also bypasses the outgoing message queue, which is sometimes useful). You can also handle undocu-

mented events received from the server as they'll be sent to the `onUnknown` method if they're not recognized by PircBot as valid commands.

Many people ask (without thinking) whether PircBot supports multiple servers. The answer is yes, of course. This is achieved simply by creating a new instance of PircBot for each server that you wish to connect to. A controller class should be used to maintain a collection of PircBot objects, if they're required to interact with each other. Interesting things can be achieved with multiple server connectivity, such as Steve Jolly's bridgebot. This uses the PircBot framework to bridge infobots on two IRC servers, giving two communities access to each other's shared memory.

### Adding to Existing Applications

There are many situations where it can be beneficial to spend a little time adding PircBot functionality to an existing application. Say you've used Java to implement a program to stream MP3 audio. Multiple clients can connect to it and listen to the music you're playing. Your listeners start to complain that they don't know what track they're listening to and would like more control over what gets played next. What do you do? An easy solution would be to integrate PircBot into your streaming server so it can announce to an IRC channel what track is being played. It could even be used to parse commands from other users to handle requests for the next track. Adding such extra functionality can be done in a matter of minutes.

### PircBot in Education

One of the most appealing aspects of PircBot is that there's so much scope to the applications for which it can be used. This is why it has gained so much interest among undergraduate students who are in the process of learning Java. Students find that PircBot is a good way to apply their learning to an actual application that may even end up doing something useful. There remains the opportunity to explore and understand inheritance, string manipulation, handling exceptions, and so on – but it's a lot more fun this way.

The PircBot Web site has a link to some introductory lecture slides, available in both PDF and Microsoft PowerPoint format. These give a gen-



Diagnose and resolve performance issues  
throughout your distributed J2EE system

## PerformaSure™

Take advantage of Quest PerformaSure's exclusive Tag and Follow technology to diagnose and resolve J2EE performance problems across an entire multi-tier J2EE application. Tag and Follow traces and reconstructs live end-user transactions across the JVMs, web/application servers and databases of your distributed J2EE system. Get to root cause with PerformaSure.

Get detailed diagnostics with rock-bottom overhead features:

- Ultra low-overhead production-grade agents
- Component-level instrumentation and detail dial
- Automatic sampling and filtering, and more.

Part of the **Quest Performance Management Suite for the J2EE Platform.**



Take a tour of PerformaSure today:

<http://java.quest.com/performa-sure/jdj>



Paul Mutton teaches Java to degree students in the UK and it is his language of choice for implementing his research on graph drawing. Paul is a Sun Certified Java Programmer.

pjm2@kent.ac.uk

eral overview of what IRC is and demonstrate how to make a simple IRC bot, as well as provide a few examples to inspire ideas and discussion. Experience has shown that these are very good at encouraging students to be creative and learn more than the minimal set of information they'll need to pass their Java course.

### PircBot Implementations

By providing an IRC framework that's so accessible and easy to use, PircBot has steadily grown in popularity, receiving tens of thousands of downloads. As you would expect, there have been some interesting implementations of PircBot seen along the way.

NewsBot sends messages to all the channels it's in whenever there's a breaking news story. It retrieves the BBC's UK and World RSS feeds every minute and passes on the title of each article as it appears, along with a URL to the full article.

A number of IRC clients have been built using PircBot. TundraIRC was written to fill the gap in suitable IRC clients for Mac OS X. ScreenIRC is another IRC client that implements

PircBot and is designed to be run permanently on a server. A novel feature is that you can then "attach" to this with a separate program with a graphical interface. This allows you to close the GUI without being disconnected from the IRC server and you can safely reattach at a later moment to see what you missed during your absence. PircBot Client (PBC) was the first ever IRC client built using the PircBot framework and uses AWT so that it can run on PDAs with JRE 1.1 installed (see Figure 2).

iscreamBot is a fine example of an IRC bot being put to good use. It's part of the i-scream distributed central monitoring system and is used to relay system alert information to a public channel on an IRC server. rpgBot is another bot designed to help out players of role-play games online and has a simple plugin system to add new commands. The bot is accompanied by some comprehensive documentation.

TrustBot is part of a project designed to build and maintain a trust network on the semantic Web. TrustBot uses PircBot to act as the current interface to the trust network.

It analyzes the network and provides information and inferences about trust between pairs of nodes, ultimately calculating how much one person should trust another. This work was published as "Trust Networks on the Semantic Web" in the proceedings of Cooperative Intelligent Agents, 2003 (<http://mindswap.org/papers/Trust.pdf>).

PieSpy is a bot that got its name by lurking around in #pie, spying on the channel's inhabitants (see Figure 3). It uses PircBot to silently observe a set of channels, monitoring events to infer a social network. A modified spring embedder graph drawing algorithm is implemented in Java to produce an automatic layout, which is used to visualize the network. The implementation of this bot is described in the paper "Inferring and Visualizing Social Networks on IRC" ([www.jibble.org/piespy/](http://www.jibble.org/piespy/)). ImageIO is used to write the visualization as a PNG

file, or high-quality EPS output can be obtained by using the EpsGraphics2D package.

More and more people in the Java scene are starting to keep a blog of their activities. Scot is a bot that can create Weblogs for IRC communities. Scot uses PircBot to interact with the IRC server and stores its data in a MySQL database. PHP provides the Web interface to the database.

ComicBot is another bot that silently observes a channel (see Figure 4). Whenever it sees something that may have been amusing, it quickly generates a cartoon comic strip from one of several templates and places it on the Web. The frequency of output simply depends on how funny the IRC channel is!

Last, but not least, is Monty. This bot uses a modified Markov chain model to learn from what other people say and generate automatic responses. Some entertaining quotes from Monty are available on my Web site. This bot was originally written in Perl, but after stumbling across some limitations of the language, it was ported to Java (a move not regretted!). While this conversion was taking place, it became apparent that some parts of Monty would be useful to other people who want to make IRC bots in Java, so PircBot was conceived. PircBot and Monty were developed in tandem, which is perhaps why so many people find PircBot so easy to use – because I wanted it to be easy for me to use! ☺

### References

- *PircBot*: [www.jibble.org/pircbot.php](http://www.jibble.org/pircbot.php)
- *ComicBot*: [www.jibble.org/comicbot/](http://www.jibble.org/comicbot/)
- *Scot and bridgebot*: [www.elvum.net/bots/](http://www.elvum.net/bots/)
- *pisg*: <http://pisg.sourceforge.net/>
- *NewsBot*: [www.fwit.co.uk/newsbot.php](http://www.fwit.co.uk/newsbot.php)
- *ScreenIRC*: [www.trout-fish.org.uk/screenirc/](http://www.trout-fish.org.uk/screenirc/)
- *TundraIRC*: [www.javaspot.net/tundairc/](http://www.javaspot.net/tundairc/)
- *i-scream*: [www.i-scream.org/](http://www.i-scream.org/)
- *rpgBot*: [cgi-bin.spaceports.com/~xiris/rpgbot.php](http://cgi-bin.spaceports.com/~xiris/rpgbot.php)
- *TrustBot*: [www.mindswap.org/~golbeck/web/trust.shtml](http://www.mindswap.org/~golbeck/web/trust.shtml)
- *Monty IRC bot quotes*: [www.jibble.org/montyquotes.php](http://www.jibble.org/montyquotes.php)
- *EpsGraphics2D package*: [www.jibble.org/epsgraphics/](http://www.jibble.org/epsgraphics/)
- *Paul Mutton's homepage*: [www.jibble.org](http://www.jibble.org)



**Build Incredible Interactive Diagrams**  
**Feature Packed New JGo Release!**

JGo, written entirely in Java, makes it easy for your Java applications to create interactive diagrams of all kinds. Create network editors, schematics, process flow diagrams, visual languages, organization charts, family trees, etc.

The JGo class library provides containers, connectors, links, orthogonal routing with automatic node avoidance and enhanced link options, drag-and-drop, scaling, in-place text editing, tooltips, printing, SVG/XML, nested graphs, lots of new sample nodes and applications, a high performance Auto Layout Option, improved IDE integration, and more...

**New Version 5.0!**

- **FREE Evaluation Kit**
- **No Runtime Fees**
- **Full Source Code**
- **Easy to Learn & Use**
- **Excellent Support**

**www.nwoods.com/go/**  
**800-434-9820**  
 Ask us about SWT!

**Northwoods**  
 SOFTWARE CORPORATION



# ULC

## Rich clients for J2EE

ULC offers rich client GUI components with a server-side programming model. Applications using ULC combine the benefits of browser and desktop applications.

ULC components are lean and based entirely on Java standards. They integrate with any existing J2EE software.

Get this proven library and save time, money, and reduce your risk throughout the software lifecycle.

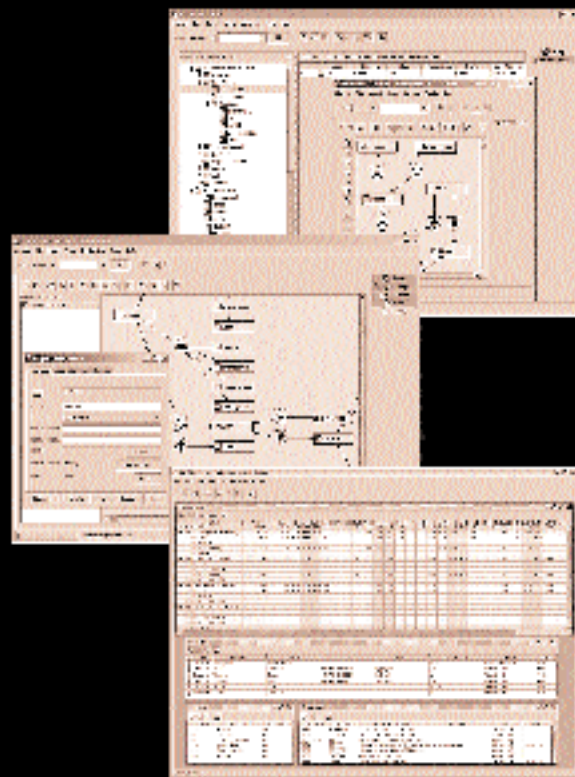
### *Benefits for:*

#### **Development & Maintenance**

- rich GUIs with simple server-side programming model
- integrates with any J2EE software
- great tool support: visual editor, load tester

#### **Deployment & Operation**

- application release on server only
- application independent presentation engine on client
- minimal network traffic: several times less than HTML



**canoo.com**

Canoo Engineering AG

<http://www.canoo.com/ulc/>

Ready to?

 **WebSphere.**

Part of the solution

**Download your free trial today!**

# SWING, THREADS, AND EVENTS

*For a more responsive GUI*

by Marcus S. Zarra

I normally spend a few hours every week reading the discussion forums at the Java Developer Network (<http://forums.java.sun.com>) and find that one of the most common problems is Swing and how it works in a multithreaded environment. I've helped a number of developers in this area when they find that their application stops responding, behaves erratically, or, the most common problem, runs very slowly.

The goal of this article is to explain some of the difficulties developers face when they're new to the Swing API and hopefully help you avoid some of the pitfalls and eliminate potential bugs that are difficult to remove once the application has been completed.

## How Swing Works

Unlike other graphical APIs, Swing runs off a single thread model. This single thread processes both the actual drawing of the GUI as well as any events that are received from the user interface. This means that the developer must pay special attention to the handling of anything related to the GUI as well as how long it takes to process an event.

Failure to take this into account can result in the application feeling sluggish or appearing to be locked up. Even worse, the application could accidentally throw an exception in the event thread, causing the entire application to behave in a strange manner.

## Single Thread

The entire Swing API is designed to run in one thread. If that single thread is blocked or slowed down, the entire GUI will slow down or stop responding. To avoid this, the application needs to do most of its work in a separate thread and interact only with the event thread when it's processing an event or when it needs to update the GUI.





Fortunately, most of the Swing API is already designed to do this as long as the application follows the basic design of the API. It's not difficult to avoid some of the pitfalls, but this becomes more difficult when dealing with events.

When a Swing application receives an event, all the objects that have registered themselves as listeners of that event receive it. For example, when a JButton is clicked, it creates an ActionEvent that's passed to each of the listeners that subscribe to that event. The method `actionPerformed` is then called sequentially on each listener to that event. There's no guarantee that listeners will be called in any order, but it is guaranteed they will all receive it. This situation is illustrated in Figure 1. The JButton fires off an action event to each of its listeners. Listener1 will receive the event and completely handle it before Listener2 or 3 will even know the event occurred. Listener2 will then receive the event and complete its handling of the event before the event is handed off to Listener3.

One way a problem can be created is when a listener behaves poorly and does not return in a reasonable amount of time; for example, when a listener is making a call that takes an unusual amount of time (perhaps loading a file from disk, accessing a database, etc.), and that code is placed either directly or indirectly in the event thread).

For example, Listing 1 has an `actionPerformed` method that will take a long time to return. If we run this example, when the JButton is clicked the GUI won't respond until the wait method has completed. This is because our example is running in the event thread and the entire application is on hold waiting for this code to finish.

### Not Thread-Safe

Another common situation is updating the GUI from a thread other than the Event thread. This situation is exactly the opposite of the one listed earlier. Listing 2 shows a basic situation where we have more than one thread in the application and the wrong thread is attempting to update the GUI. While this code may look perfectly harmless (and in fact, in a simple situation like this, probably is harmless), it's a recipe for disaster.

Since Swing is designed as a single threaded model, the API is not synchronized and therefore not thread-safe. This means we can have a nasty collision in the GUI if we attempt to update the elements from any thread other than the event thread. Doing so can cause data corruption, exceptions to be thrown, and a host of other nasty problems that are extremely difficult to debug. To avoid this problem, always update the GUI via the event thread.

## The Event Thread

If we can't work in the event thread, how does the application do anything? Whenever we have a situation where we know that an event is going to take a long time to be processed, place it in a worker thread.

Listing 3 is a rewrite of Listing 1, so instead of blocking the event thread until the waiter method completes, it constructs an anonymous inner class that extends `java.lang.Thread`, which will call the waiter method for us. It then starts this anonymous thread and returns. In this version of the code, the event will no longer wait until the waiter method has completed, but will return as soon as the thread is initialized and started. This results in a much faster response to the event and the GUI will be "snappier."

If we have a situation where the code that is being called by the event is going to be called frequently from multiple locations, we don't want to rewrite the anonymous class into every location. One way to avoid this duplication of code is to create an inner class that can be instantiated whenever it's needed without duplication. As can be seen in Listing 4, the event instantiates our `WorkerThread` and starts it. It can then instantiate another copy of this thread from any other location in the class as needed.

Another common situation is where we have a piece of code that is going to take a long time to execute and will only be called from a worker thread. In this situation it may be more advantageous to move the entire method into the worker thread as shown in Listing 5. This allows us to completely separate the logic from the GUI. This worker thread can either be in a separate class file or defined as an inner class, as in the example.

Each of these three examples will allow us to return the event thread in a timely manner and process the actual work in a separate thread, thereby allowing the GUI to continue receiving events, etc.

## Multiple Threads

As applications become more complex, they inevitably require multiple threads to run smoothly. This is in direct contradiction to the design of the Swing API. To get around this disparity Sun has built a couple of methods into the AWT and Swing APIs to allow you to use multiple threads with the single thread model. The basic concept behind them is to have all your GUI-related instructions run in the event thread. To do this, you need to insert them into the event queue. The following two methods are available as static methods in the `javax.swing.SwingUtilities` class and the `java.awt.EventQueue`.

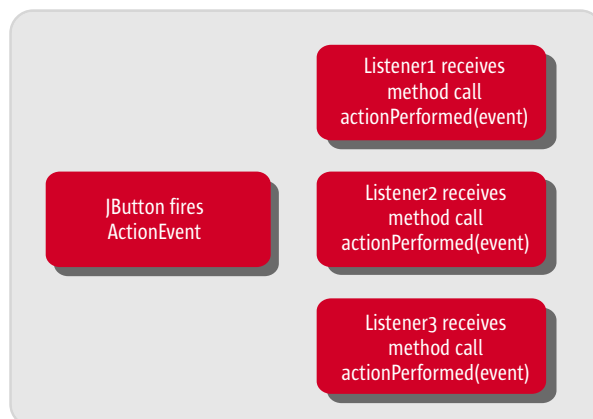


Figure 1 The method `actionPerformed`

`invokeLater(Runnable r)` causes `r.run()` to be executed asynchronously on the AWT event dispatching thread. This thread will be processed in turn once it has reached the top of the event queue. Even if this method is called from the event dispatching thread, it will still be deferred based on the items in the queue.

Listing 6 shows an example of how to use this method. It should be noted that if an exception occurs in the `Runnable` object placed in the event queue, it will cause the event queue thread to fail as opposed to the thread that placed it in the queue. If there is a chance of an exception being thrown, it should be caught and handled as opposed to letting it kill the event thread.

`invokeAndWait(Runnable r)` causes `r.run()` to be executed synchronously on the AWT event dispatching thread. This method call will block until all pending events have been completed and the `run()` method of `r` has been executed.

Listing 7 shows an example of how this method works. As can be seen from the example, it throws two types of exceptions. `InterruptedException` is thrown if the current thread is interrupted. The second exception, `InvocationTargetException`, is a wrapper exception that will be thrown if our `Runnable` throws an exception. The actual exception thrown by the `Runnable` will be contained inside the `InvocationTargetException`.

By utilizing these two methods, it's possible to have a multithreaded application interact either asynchronously or synchronously with the Event Queue and thereby adhere to the single thread design.

## Swing Worker

Sun has developed a utility class that performs the function listed earlier as well as adding some additional features that can be useful. `SwingWorker` is a class you extend off of that allows you to place your time-intensive code in a separate thread that gives you quite a bit of flexibility and control. While this class may not be useful in all situations, it's a lot more useful than constantly re-creating the functionality it provides.

The `SwingWorker` class is not part of the standard Java API and must be downloaded from Sun's Web site separately and compiled. There is a link to it in the references section at the end of the article.

Listing 8 starts off by extending `SwingWorker`. The only method that we must define is `construct()`. As can be seen in the example, we have created a loop inside the `construct` method that will sleep for 100 milliseconds each iteration. Our `JFrame` instantiates a button that we have attached an `ActionListener` to. When the button is pushed, our `ActionListener` creates an instance of our worker class and calls its `start` method. The `start` method executes the `run` method of the parent class `SwingWorker` that, among other things, calls our `construct` method. If you explore the `SwingWorker` class, you'll see that inside the `run` method our `construct` method blocks the thread until it's complete.

The `SwingWorker` class provides us with some additional functionality above and beyond separating our work from the Event Thread. First, the `construct` method returns an object that we can then retrieve from the worker once the thread has completed. If we attempt to retrieve the value before our worker thread has completed, the `get` will block until the thread is done.

When the `SwingWorker` has completed, it will call the finished method that does nothing unless we extend it. The nice thing about the finished method is that it is executed in the event thread as opposed to the worker thread, thus we can





## Searchable Books

Conduct searches across 1,500+ technical books from publishers such as Addison Wesley, Microsoft Press, O'Reilly, Prentice Hall, and many more. Zero in on answers to time-critical questions instantly.

Read one book on your Bookmarks from cover to cover. Or, simply fly to the page you need.

• Browse books by category. Researching any topic is a snap. From XML to database to .Net, you'll find your answer here.

## eLearning Courses

The Isavix High-Impact eLearning Series™ is designed to deliver online training in the most efficient manner possible; each course is approximately one hour in length with no "homework" assignments and can be viewed in Flash and PDF formats. Sample courses include: The Struts Framework, Introduction to Apache ANT, ASP.NET Crash Course, and more.

## Certification Exam Preparation

Whizlabs exam simulators are available for Sun, IBM, BEA, CISCO, Microsoft, Oracle, CompTIA, UAW and Red Hat certifications. These products have been developed by experts who are considered the authorities in their subjects.

From computer professionals to Certification Training centers to College and University students, to hundreds of corporations ... the Whizlabs solutions have helped over 150,000 people get their certifications.

## FREE Resources and Tools

isavix.net is a site that provides eLearning, information sharing, software exchanges, helpful tools, and valuable links for emerging technology (e.g. J2EE, .NET) professionals.

Our free tools and resources include Discussion Forums, Code Exchange, My Secure Files, Quick Hints, My Bookmarks, Download Links, My UML Diagrams, My eLearning, and much more.

## Custom Community Portal For Your Organization

If you like the many benefits isavix.net provides but would want a custom version of our portal available exclusively for your organization, we can do that for you!

Our community can either be installed within your organization's Intranet or provided to you as an ASP solution.

The benefits include an enhanced user experience, professional development, improved team dynamics, effective project management, and better resource allocation. The ROI for an organization is high because developers can save from a few minutes to hours a day since the relevant information is readily available to them, when they need it!

**Corporate and Government solutions  
available ... contact us for details.**



**Isavix Corporation**  
Herndon, VA  
1-866-472-6849  
info@isavix.com



J2ME



J2SE



J2EE



HOME

execute GUI-related code inside it. As can be seen in our example, we show a message dialog letting the user know we have completed the task.

Now what if we want to interrupt our worker before it has completed its task? The `SwingWorker` class also has functionality built in to handle this as well. When we want to stop the worker we call the `SwingWorker.interrupt()` method that will stop the thread dead in its tracks. Note that the finished method will *not* be called if we interrupt the thread.

### Multi-Threaded GUI Construction

This topic is a bit on the controversial side of Swing programming. The reason behind it is simple. It's easy to create a JFC application that behaves erratically by using threads in the construction. However, that being said, it is possible to construct a JFC application by using threads in the construction.

Why would anyone do this? If we have an application that has several panels, a couple of toolboxes, and a few other complex widgets, then the start-up time can be horrific. A perfect example of this situation is one of my favorite editors, `JEdit`. In my opinion this is one of the finest editors out there but it suffers from a common problem: it takes forever to get to a working state. Once the application

When the `JFrame` is ready to actually display it checks and waits for the `JPanels` to finish their own initialization. In each of the `JPanel`'s run methods, the final method call is to decrement the counter. When all the `JPanels` have finished, the counter will be back down to zero and the `JFrame` then knows that it can pack everything and display itself.

While this example is simplistic, it does show us some rules around creating a multithreaded initialization routine:

- **Make sure none of the objects that are threaded have to interact with each other:** Otherwise you may have an object trying to access a resource that has not been initialized yet (i.e., trying to listen to an object that has not been constructed yet).
- **Make certain that all the threads have completed before you validate the GUI:** Calling `setVisible(true)`, `pack()`, and `validate()` will all validate the GUI and line everything up. Calling this prematurely could cause your GUI to behave inappropriately.
- **Launching threads within threads is probably not a good idea:** Unless you are very careful with the threads, this is likely the shortest path to ruin. Avoid doing this unless you are sure about your thread interactions.

“It's easy to create a JFC application that behaves erratically by using threads in the construction. However...it is possible to construct a JFC application by using threads in the construction”

is running it performs beautifully, but be prepared to wait for it to get to that state.

While not every application can benefit from what I am about to describe, it can speed up the initialization of an application and thus give the user the impression of a snappy response. The negatives of this approach are twofold:

- If care is not taken in managing the construction of the GUI, the results can be unpredictable at best.
- Your application could easily consume all of the available CPU on the user's machine until initialization is complete. If we succeed in reducing the time to almost nothing, this is not a major issue; if the initialization still takes a while, it's possible it will cause the user's OS to respond poorly.



**Marcus S. Zarra** is the director of technology at Extraquest Corporation in Denver, Colorado. He began developing professionally in 1985 for school systems and then for legal firms in Phoenix, Arizona. Marcus has been a Java developer since 1996, working in the banking, telecommunications, and insurance industries.

mzarra@extraquest.com

Considering all this, multithreading the initialization routines of a GUI can speed up its start time significantly. Listing 9 shows a stripped down example of a multithreaded GUI initialization.

In this example, the threading is occurring in the `JPanels` that will be going into the `JFrame`. Each of these panels, upon construction, immediately increments the static counter in the `JFrame`. This counter lets the `JFrame` know when all the threads have completed. After incrementing the counter, each `JPanel` then fires up a thread passing itself as the `Runnable` object. Finally, the constructor starts the thread that it just initiated and then returns. This allows the `JFrame` to initialize all four of the `JPanels` very quickly and, in a real-world situation, all of these `JPanels` would be busy initializing themselves while the `JFrame` is off doing other work (initializing listeners, etc.).

### Conclusion

While the examples above cause our application to be more complicated, they also allow the GUI to be more responsive to the user. In the end these tips have to be weighed against the complexity of the GUI you are writing.

If your application has a very simple GUI with a small chance of improper interactions or collisions, chances are you don't have to worry too much. However, as an application gets more complex (as they invariably do), threading becomes more important. Making a habit of watching thread interactions in your GUI can save you from a massive rewrite later when the application has outgrown itself.

### Acknowledgment

A special thanks to Jon Svede for all of his hard work reviewing and commenting on the drafts of this article. ☺

### References

- *Using a Swing Worker Thread:* <http://java.sun.com/products/jfc/tsc/articles/threads/threads2.html>
- *Threads and Swing:* <http://java.sun.com/docs/books/tutorial/uiswing/overview/threads.html>
- *High Performance GUIs with the JFC/Swing API:* <http://developer.java.sun.com/developer/community/chat/JavaLive/2002/jl0423.html>



**Listing 1**

```

public void actionPerformed(
    ActionEvent evt) {
    //Calling a method that takes a long
    //time to return
    waiter();
}

private void waiter() {
    //I am a method that takes a long
    //time to return
    try {
        Thread.sleep(60000);
    } catch (InterruptedException e) {
        e.printStackTrace();
    }
}

```

**Listing 2**

```

public ExampleClass extends JFrame
    implements Runnable {
    private JTextField example1;
    private JTextField example2;

    public ExampleClass() {
        getContentPane().setLayout(
            new FlowLayout());
        example1 = new JTextField(
            "Field 1");
        example2 = new JTextField(
            "Field 2");
        getContentPane().add(example1);
        getContentPane().add(example2);
        Thread t = new Thread(this);
        t.start();
    }
}

```

```

}

public void run() {
    //I am not the event thread,
    //I should not be updating the GUI
    example1.setText("Updated");
}
}

```

**Listing 3**

```

public void actionPerformed(
    ActionEvent evt) {
    Thread t = new Thread() {
        public void run() {
            waiter();
        }
    };
    t.start();
}

private void waiter() {
    //I am a method that takes a long
    //time to return
    try {
        Thread.sleep(60000);
    } catch (InterruptedException e) {
        e.printStackTrace();
    }
}

```

**Listing 4**

```

class WorkerThread extends Thread {
    public void run() {
        waiter();
    }
}

```

You can  
use any tool...



But why not  
the right one...

## Struts Studio for Eclipse



Design, build, test, and deliver Struts Web applications, all with **Struts Studio**® – comprehensive, extensible, and easy-to-use. **Struts Studio** is the ideal tool for all your Web development projects.

As an Eclipse plug-in, **Struts Studio**® supercharges the Eclipse IDE into a **WIDE** (Web Integrated Development Environment).

WWW.EXADEL.COM



```

public void actionPerformed(
    ActionEvent evt) {
    WorkerThread wt = new WorkerThread();
    wt.start();
}

private void waiter() {
    //I am a method that takes a long
    //time to return
    try {
        Thread.sleep(60000);
    } catch (InterruptedException e) {
        e.printStackTrace();
    }
}

```

**Listing 5**

```

class WorkerThread extends Thread {
    public void run() {
        //I am a method that takes a long
        //time to return
        try {
            Thread.sleep(60000);
        } catch (InterruptedException e) {
            e.printStackTrace();
        }
    }
}

public void actionPerformed(
    ActionEvent evt) {
    WorkerThread wt = new WorkerThread();
    wt.start();
}

```

**Listing 6**

```

Runnable example = new Runnable() {
    public void run() {
        System.out.println(
            "Example running from " +
            Thread.currentThread());
    }
};
SwingUtilities.invokeLater(example);
System.out.println("This probably will "
    +"be executed before the example "
    +"thread");

```

**Listing 7**

```

Runnable example = new Runnable() {
    public void run() {
        System.out.println(
            "Example running from " +
            Thread.currentThread());
    }
};
try {
    SwingUtilities.
        invokeAndWait(doHelloWorld);
} catch (InterruptedException e) {
    e.printStackTrace();
} catch (InvocationTargetException e) {
    e.printStackTrace();
    //Wrapper for an exception that was
    //thrown from our example
}

```

**Listing 8**

```

public class MyWorker
    extends SwingWorker {
    JFrame parent;

    public MyWorker(JFrame p) {

```

```

        parent = p;
    }

    public Object construct() {
        for (int i = 0; i < 10; i++) {
            try {
                Thread.sleep(100);
            } catch (InterruptedException e) {
                e.printStackTrace();
            }
        }
        return null;
    }

    public void finished() {
        JOptionPane.showMessageDialog(
            parent, "I'm finished!");
    }
}

```

```

public class MyFrame extends JFrame
    implements ActionListener {
    private JButton _pushMe;
    private JButton _stopMe;
    private MyWorker myWorker

    public MyFrame() {
        JPanel buttons = new JPanel(
            new FlowLayout());
        _pushMe = new JButton("Push Me");
        _stopMe = new JButton("Stop Me");
        _pushMe.addActionListener(this);
        _stopMe.addActionListener(this);
        buttons.add(_pushMe);
        buttons.add(_stopMe);
        getContentPane().add(buttons,
            BorderLayout.NORTH);
    }

```

```

    public void actionPerformed(
        ActionEvent evt) {
        if (evt.getSource() == _pushMe) {
            myWorker = new MyWorker();
            myWorker.start();
        } else {
            if (myWorker != null) {
                myWorker.interrupt();
            }
        }
    }
}

```

**Listing 9**

```

public class ThreadedFrame extends
    JFrame {
    private static int initCounter = 0;

    private JTabbedPane _tabbedPane;

    public ThreadedFrame() {
        getContentPane().setLayout(
            new BorderLayout());
        _tabbedPane = new JTabbedPane();
        getContentPane().add(_tabbedPane,
            BorderLayout.CENTER);

        _tabbedPane.addTab("Tab One",
            new TabOne());
        _tabbedPane.addTab("Tab Two",
            new TabTwo());
        _tabbedPane.addTab("Tab Three",
            new TabThree());
        _tabbedPane.addTab("Tab Four",
            new TabFour());

        while (initCounter > 0) {
            try {
                Thread.sleep(50);
            } catch (InterruptedException e) {
                e.printStackTrace();
            }
        }
    }
}

```



```

    }
}

pack();
setVisible(true);
}

public synchronized static final void
incCounter() {
    initCounter++;
}

public synchronized static final void
decCounter() {
    initCounter--;
}
}

public class TabOne extends JPanel
implements Runnable {

    public TabOne() {
        ThreadedFrame.incCounter();
        Thread t = new Thread(this);
        t.start();
    }

    public void run() {
        //Initialize the GUI code here
        ThreadedFrame.decCounter();
    }
}

public class TabTwo extends JPanel
implements Runnable {

    public TabTwo() {
        ThreadedFrame.incCounter();

```

```

        Thread t = new Thread(this);
        t.start();
    }

    public void run() {
        //Initialize the GUI code here
        ThreadedFrame.decCounter();
    }
}

public class TabThree extends JPanel
implements Runnable {

    public TabThree() {
        ThreadedFrame.incCounter();
        Thread t = new Thread(this);
        t.start();
    }

    public void run() {
        //Initialize the GUI code here
        ThreadedFrame.decCounter();
    }
}

public class TabFour extends JPanel
implements Runnable {

    public TabFour() {
        ThreadedFrame.incCounter();
        Thread t = new Thread(this);
        t.start();
    }

    public void run() {
        //Initialize the GUI code here
        ThreadedFrame.decCounter();
    }
}

```

162,019\*



120,031\*



Dr. Dobbs's

75,561\*



MSDN

\*JUNE 2003 IPA AUDIT STATEMENTS  
\*All circulation numbers are publishers' most current own data, six-month average circulation through June 2003.

**SYS-CON**  
MEDIA

# #1 Circulation in the World!

*JDJ is the highest circulation  
i-Technology magazine  
in the world!*

Miles Silverman  
VP Marketing

Carmen Gonzalez  
Senior VP Marketing



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

**JAVA** DEVELOPER'S  
JOURNAL



Where  
**Open Minds**  
Meet



#### Keynote Speakers



**Chris Stone**

*Vice Chairman, Office of the CEO,  
Novell, Inc.*

Wednesday, January 21  
9:15 am – 10:00 am



**Dave Dargo**

*Vice President, Linux Program Office,  
Oracle*

Wednesday, January 21  
11:45 am – 12:30 pm



**Tom Killalea**

*Vice President of Infrastructure,  
Amazon.com*

Wednesday, January 21  
2:45 pm – 3:30 pm



**Sam Greenblatt**

*Senior Vice President and Chief Architect,  
Computer Associates Intl.*

Thursday, January 22  
11:30 am – 12:15 am



**Ross A. Mauri**

*General Manager, e-Business on demand,  
IBM Systems Group*

Thursday, January 22  
1:45 pm – 2:30 pm



**CONFERENCE:** January 20 – 23, 2004

**EXPO:** January 21 – 23, 2004

**THE JAVITS CENTER:** New York, NY

LinuxWorld Conference & Expo is the world's leading and most comprehensive event focusing on Linux and open source solutions. **This January, discover how companies across the globe have achieved higher profits and increased their productivity by utilizing Linux, the fastest growing operating system in the world.**



- Get in-depth coverage on the latest Linux and open source developments and learn about innovative product solutions from the industry's top companies.
- Network with the leaders in the open source movement and discuss with your peers how to best leverage the technology for your organization.
- Participate in LinuxWorld's world-class education program and benefit from interactive training in the all-new Hands-on Labs!
- Attend the 2nd annual Linux Financial Summit and hear how Wall Street firms are leveraging Linux to achieve a lower total cost of ownership. (Sponsored by BEA)
- Discover real world practical solutions and find answers to today's most critical IT issues.

Cornerstone Sponsor

**ORACLE®**

Platinum Sponsors



**IBM**

**intel**



**Novell**



REGISTER ONLINE WITH REFERENCE CODE: A-JDD

**[www.linuxworldexpo.com](http://www.linuxworldexpo.com)**

**IDG**  
WORLD EXPO

Glen Cordrey  
J2ME Editor

## A New Year, a New Future

**T**his is my last column as J2ME editor. A convergence of circumstances has made this an opportune time for me to sign off.

With the January issue of *JDJ* the J2ME section will be merged into the enterprise section. This reflects, in my opinion, a truth that some of the greatest benefits of J2ME will be realized through its symbiosis with J2EE. While the general industry press frequently centers on J2ME's capabilities for game development, or such trifles as Christina Everywhere, I think that it is in the evolution of the mobile workforce that we'll see some of the most significant advantages of J2ME come to the fore.

We're beginning to see how the combination of notebook PCs and WiFi opens up opportunities to work from nontraditional venues. Now extend that model to smaller, more mobile devices supported by 3G networks, and mobile computing becomes pervasive – or, dare I use that overused word, ubiquitous. Current applications such as mobile e-mail, for all of their success, are relatively primitive examples of the benefits to come from mobile computing. Simply being able to remotely access enterprise assets such as engineering schematics, real-time pricing information, and billing records opens up a wealth of opportunity to increase productivity and offer new services. Add the likelihood of innovative applications that leverage enterprise assets in unexpected ways and the benefits of mobile enterprise computing multiply.

What does this have to do with the aforementioned “convergence of circumstances”? Well, *JDJ*'s merging of the J2ME and enterprise sections coincides with a time frame I established for myself earlier in the year – a year-end deadline for either finding enough J2ME work to make it my primary focus, or accepting that the J2ME market isn't mature enough yet for that to happen. New work hasn't come, so I'm concentrating my efforts on J2EE.

Now my opinion about J2ME hasn't become negative – I still think it has great potential and will experience continuing success. It has a powerful edge over its competitors in the quality of the technology and commercial presence, and is a

strong bet to become the predominant technology for developing mobile applications (and may be so already). However, I have come to suspect that the future legions of J2ME programmers forecast by Sun and others won't be quite as massive as claimed. The heavy lifting will continue to be done on the enterprise side.

One reason I think this is so is because, quite simply, the J2ME applications of today can't be very big or complex. Forgive me for stating the obvious, but limited memory, data storage, and processing power significantly limit the number and diversity of potential mobile applications, and one developer can pretty much write an entire application with a few weeks', or at most a few months', effort.

This is changing, of course, and will continue to change. For example, the MIDP 2.0 spec mandates a minimum of 128KB for volatile memory (heap), compared to 32KB for the MIDP 1.0. (With the MIDP 1.0, if you hadn't heard of an obfuscator before, you quickly learned that it was a necessity for compacting almost any MIDlet into the footprint supported by most MIDP devices.) And, of course, continued improvements in the speed, memory capacities, and processing power of the hardware will create opportunities for new applications. But I think that in the near future (the next few years), the real power in J2ME applications will be seen in sophisticated clients for back-end, enterprise systems. The real increase in the number of J2ME programmers will occur, I suspect, as enterprise developers begin to mobilize their applications. Yes, there will be a segment of the Java community that is focused primarily on J2ME development, but I believe that for most Java developers, J2ME development will be an adjunct to their main task, developing enterprise applications, which is where you'll find me.

I'm not disappearing from these pages entirely though. I'll continue as a contributing editor, and will chime in occasionally with my perspective on things J2EE, J2ME, Java, et al. In making that transition, I'd like to thank the *JDJ* staff for giving me the privilege of writing this column, and for their efforts and success in publishing a quality magazine each month. ☺

Building a  
Connected  
MIDlet

46

### A New Year, a New Future

With the January issue of *JDJ*, the J2ME section will be merged into the enterprise section. This reflects, in my opinion, a truth that some of the greatest benefits of J2ME will be realized by its symbiosis with J2EE. While the general industry press frequently centers on J2ME's capabilities for game development, or such trifles as Christina Everywhere, I think that it is in the evolution of the mobile workforce that we'll see some of the most significant advantages of J2ME come to the fore.



Glen Cordrey is a software architect working in the Washington, DC, area. He's been using Java for five years, developing both J2EE and J2ME applications for commercial customers.

glencordrey@sys-con.com



**A LIMITED TIME SAVINGS OFFER FROM SYS-CON MEDIA**

# SUBSCRIBE TODAY TO MULTIPLE MAGAZINES AND SAVE UP TO \$400 AND RECEIVE UP TO 3 FREE CDs!\*

**RECEIVE  
YOUR DIGITAL  
EDITION  
ACCESS CODE  
INSTANTLY  
WITH YOUR PAID  
SUBSCRIPTIONS**



**3-Pack**  
Pick any 3 of our  
magazines and save  
up to **\$275<sup>00</sup>**  
Pay only \$175 for a  
1 year subscription  
plus a **FREE CD**  
• 2 Year – \$299.00  
• Canada/Mexico – \$245.00  
• International – \$315.00

**6-Pack**  
Pick any 6 of our  
magazines and save  
up to **\$350<sup>00</sup>**  
Pay only \$395 for a  
1 year subscription  
plus 2 **FREE CDs**  
• 2 Year – \$669.00  
• Canada/Mexico – \$555.00  
• International – \$710.00

**9-Pack**  
Pick 9 of our  
magazines and save  
up to **\$400<sup>00</sup>**  
Pay only \$495 for a  
1 year subscription  
plus 3 **FREE CDs**  
• 2 Year – \$839.00  
• Canada/Mexico – \$695.00  
• International – \$890.00

**CALL TODAY! 888-303-5282**

**Pick a 3-Pack, a 6-Pack or a 9-Pack**

<input type="checkbox"/> 3-Pack	<input type="checkbox"/> 1YR <input type="checkbox"/> 2YR	<input type="checkbox"/> U.S. <input type="checkbox"/> Can/Mex <input type="checkbox"/> Intl.
<input type="checkbox"/> 6-Pack	<input type="checkbox"/> 1YR <input type="checkbox"/> 2YR	<input type="checkbox"/> U.S. <input type="checkbox"/> Can/Mex <input type="checkbox"/> Intl.
<input type="checkbox"/> 9-Pack	<input type="checkbox"/> 1YR <input type="checkbox"/> 2YR	<input type="checkbox"/> U.S. <input type="checkbox"/> Can/Mex <input type="checkbox"/> Intl.

**TO  
ORDER**

•Choose the Multi-Pack you want to order by checking next to it below. •Check the number of years you want to order. •Indicate your location by checking either U.S., Canada/Mexico or International. •Then choose which magazines you want to include with your Multi-Pack order.

**MX Developer's Journal**

U.S. - Two Years (24) Cover: \$143	You Pay: \$49.99 /	Save: \$167 + FREE \$198 CD
U.S. - One Year (12) Cover: \$72	You Pay: \$29.99 /	Save: \$60
Can/Mex - Two Years (24) \$168	You Pay: \$79.99 /	Save: \$137 + FREE \$198 CD
Can/Mex - One Year (12) \$84	You Pay: \$49.99 /	Save: \$40
Intl - Two Years (24) \$216	You Pay: \$89.99 /	Save: \$127 + FREE \$198 CD
Intl - One Year (12) \$108	You Pay: \$59.99 /	Save: \$30
Digital Edition - One Year (12)	You Pay: \$19.99	

**Linux World Magazine**

U.S. - Two Years (24) Cover: \$143	You Pay: \$79.99 /	Save: \$63 + FREE \$198 CD
U.S. - One Year (12) Cover: \$72	You Pay: \$39.99 /	Save: \$32
Can/Mex - Two Years (24) \$168	You Pay: \$119.99 /	Save: \$48 + FREE \$198 CD
Can/Mex - One Year (12) \$84	You Pay: \$79.99 /	Save: \$4
Intl - Two Years (24) \$216	You Pay: \$176 /	Save: \$40 + FREE \$198 CD
Intl - One Year (12) \$108	You Pay: \$99.99 /	Save: \$8

**Java Developer's Journal**

U.S. - Two Years (24) Cover: \$144	You Pay: \$89 /	Save: \$55 + FREE \$198 CD
U.S. - One Year (12) Cover: \$72	You Pay: \$49.99 /	Save: \$22
Can/Mex - Two Years (24) \$168	You Pay: \$119.99 /	Save: \$48 + FREE \$198 CD
Can/Mex - One Year (12) \$84	You Pay: \$79.99 /	Save: \$4
Intl - Two Years (24) \$216	You Pay: \$176 /	Save: \$40 + FREE \$198 CD
Intl - One Year (12) \$108	You Pay: \$99.99 /	Save: \$8

**Web Services Journal**

U.S. - Two Years (24) Cover: \$168	You Pay: \$99.99 /	Save: \$68 + FREE \$198 CD
U.S. - One Year (12) Cover: \$84	You Pay: \$69.99 /	Save: \$14
Can/Mex - Two Years (24) \$192	You Pay: \$129 /	Save: \$63 + FREE \$198 CD
Can/Mex - One Year (12) \$96	You Pay: \$89.99 /	Save: \$6
Intl - Two Years (24) \$216	You Pay: \$170 /	Save: \$46 + FREE \$198 CD
Intl - One Year (12) \$108	You Pay: \$99.99 /	Save: \$8

**.NET Developer's Journal**

U.S. - Two Years (24) Cover: \$168	You Pay: \$99.99 /	Save: \$68 + FREE \$198 CD
U.S. - One Year (12) Cover: \$84	You Pay: \$69.99 /	Save: \$14
Can/Mex - Two Years (24) \$192	You Pay: \$129 /	Save: \$63 + FREE \$198 CD
Can/Mex - One Year (12) \$96	You Pay: \$89.99 /	Save: \$6
Intl - Two Years (24) \$216	You Pay: \$170 /	Save: \$46 + FREE \$198 CD
Intl - One Year (12) \$108	You Pay: \$99.99 /	Save: \$8

**XML-Journal**

U.S. - Two Years (24) Cover: \$168	You Pay: \$99.99 /	Save: \$68 + FREE \$198 CD
U.S. - One Year (12) Cover: \$84	You Pay: \$69.99 /	Save: \$14
Can/Mex - Two Years (24) \$192	You Pay: \$129 /	Save: \$63 + FREE \$198 CD
Can/Mex - One Year (12) \$96	You Pay: \$89.99 /	Save: \$6
Intl - Two Years (24) \$216	You Pay: \$170 /	Save: \$46 + FREE \$198 CD
Intl - One Year (12) \$108	You Pay: \$99.99 /	Save: \$8

**WebLogic Developer's Journal**

U.S. - Two Years (24) Cover: \$360	You Pay: \$169.99 /	Save: \$190 + FREE \$198 CD
U.S. - One Year (12) Cover: \$180	You Pay: \$149 /	Save: \$31
Can/Mex - Two Years (24) \$360	You Pay: \$179.99 /	Save: \$180 + FREE \$198 CD
Can/Mex - One Year (12) \$180	You Pay: \$169 /	Save: \$11
Intl - Two Years (24) \$360	You Pay: \$189.99 /	Save: \$170 + FREE \$198 CD
Intl - One Year (12) \$180	You Pay: \$179 /	Save: \$1

**ColdFusion Developer's Journal**

U.S. - Two Years (24) Cover: \$216	You Pay: \$129 /	Save: \$87 + FREE \$198 CD
U.S. - One Year (12) Cover: \$108	You Pay: \$89.99 /	Save: \$18
Can/Mex - Two Years (24) \$240	You Pay: \$159.99 /	Save: \$80 + FREE \$198 CD
Can/Mex - One Year (12) \$120	You Pay: \$99.99 /	Save: \$20
Intl - Two Years (24) \$264	You Pay: \$189 /	Save: \$75 + FREE \$198 CD
Intl - One Year (12) \$132	You Pay: \$129.99 /	Save: \$2

**Wireless Business & Technology**

U.S. - Two Years (24) Cover: \$144	You Pay: \$89 /	Save: \$55 + FREE \$198 CD
U.S. - One Year (12) Cover: \$72	You Pay: \$49.99 /	Save: \$22
Can/Mex - Two Years (24) \$192	You Pay: \$139 /	Save: \$53 + FREE \$198 CD
Can/Mex - One Year (12) \$96	You Pay: \$79.99 /	Save: \$16
Intl - Two Years (24) \$216	You Pay: \$170 /	Save: \$46 + FREE \$198 CD
Intl - One Year (12) \$108	You Pay: \$99.99 /	Save: \$8

**WebSphere Developer's Journal**

U.S. - Two Years (24) Cover: \$360	You Pay: \$169.99 /	Save: \$190 + FREE \$198 CD
U.S. - One Year (12) Cover: \$180	You Pay: \$149 /	Save: \$31
Can/Mex - Two Years (24) \$360	You Pay: \$179.99 /	Save: \$180 + FREE \$198 CD
Can/Mex - One Year (12) \$180	You Pay: \$169 /	Save: \$11
Intl - Two Years (24) \$360	You Pay: \$189.99 /	Save: \$170 + FREE \$198 CD
Intl - One Year (12) \$180	You Pay: \$179 /	Save: \$1

**PowerBuilder Developer's Journal**

U.S. - Two Years (24) Cover: \$360	You Pay: \$169.99 /	Save: \$190 + FREE \$198 CD
U.S. - One Year (12) Cover: \$180	You Pay: \$149 /	Save: \$31
Can/Mex - Two Years (24) \$360	You Pay: \$179.99 /	Save: \$180 + FREE \$198 CD
Can/Mex - One Year (12) \$180	You Pay: \$169 /	Save: \$11
Intl - Two Years (24) \$360	You Pay: \$189.99 /	Save: \$170 + FREE \$198 CD
Intl - One Year (12) \$180	You Pay: \$179 /	Save: \$1

\*WHILE SUPPLIES LAST. OFFER SUBJECT TO CHANGE WITHOUT NOTICE

**Subscribe Online Today [www.sys-con.com/2001/sub.cfm](http://www.sys-con.com/2001/sub.cfm)**

**SYS-CON  
MEDIA**

# Building a Connected MIDlet

...and installing  
on a mobile device

by Alex Bourgeois and Richard Kasperowski

**O**ne of the most powerful aspects of J2ME is connected mobility: you're no longer tied to your desk to accomplish many vital tasks. You can carry everything you need in your shirt pocket, send an e-mail while standing in line at the grocery store, or check the latest stock figures while at a baseball game.

There is nothing new about distributed applications; people have been using them for years and their effectiveness and potential are widely accepted. This is where J2ME comes into the picture – providing mobile devices with a powerful means of running robust connected applications. The possibilities seem endless.

We assume you're already familiar with J2SE or J2EE and the basic architecture of a J2ME MIDlet. If not, you might want to read our article "Building Your First J2ME Application" at [www.altisimo.com/publications/j2me-hello](http://www.altisimo.com/publications/j2me-hello).

This article shows how to build an Internet-enabled mobile application and illustrates the considerations that must be taken into account during design and development. The application we build consists of a J2ME client and a J2EE server. By the end of this article, you'll know how to:

- Get input from a user
- Programmatically establish an Internet connection
- Send data to a remote server
- Read the data returned by the server
- Present the received data to the user

## The Echo Application

We'll first build an "Echo" client and server. Echo is the standard program we write when learning a new networking technology. The client sends a String to the server, the server sends it back to the client, and the client verifies that the server's response is the original String. This little application allows us to test many of the most important features of a new networking system: collecting and displaying information to a user, establishing a connection to a remote system, and transmitting and receiving data along that connection.

We'll build the application iteratively, improving it with each iteration. In the initial iteration, we're primarily interested in having the mobile device establish a connection to a remote server and using that connection to transmit and receive information. As a result, we need both J2EE and J2ME components, each of which is tailored for a specific set of tasks. The aspect

of the application we're most interested in is the J2ME client (EchoClient), which runs on the mobile device.

The J2EE portion of the application (EchoServlet) is a rudimentary servlet that will accept an incoming request, read a string from that request, and then send the same string back to the mobile device. We have the servlet running on our Web site, so you don't need to do anything with J2EE if you don't wish to; the client code you download is configured to connect to our Web site. We only bring it up in the context of a J2ME application to show that the mobile phone needs to connect to something running on a remote server.

The application consists of:

- **J2ME client:** The client side of our application consists of a single class named EchoClient, which is installed on the mobile device. EchoClient initiates and uses a connection to EchoServlet. EchoClient is responsible for creating and displaying all the forms that allow the user to enter and read data that is sent to and received from EchoServlet.
- **J2EE server:** The server is a simple Web application that reads in a String from the client and returns the same String. The server side consists of these classes:
  - EchoServlet:** The servlet class, which reads in the String using readUTF, and then writes it back out using writeUTF.
  - EchoTest:** An automated test client to ensure that the servlet is functioning properly before we begin writing J2ME client code. For a more complex application, we would use JUnit; however, for this example, this simpler test is acceptable.
- **Build tools:** We use Ant and Antenna to compile the .java files, preverify the .class files, and update the version number and .jar size details in the .jad, as well as handle all the other tasks involved with building a J2ME application. We can run our most common tasks using single commands, thereby eliminating typos and unintentional changes to or deletions of files. Ant takes care of platform independence: all we have to do is write build (Unix shell script) and build.bat (Windows batch file) to launch build.xml from either development environment (Linux or Windows).

## Testing the Servlet

We start by building a test of the server. It might seem backward to write a test of the server before we have a working server, but it saves time later. If we know the servlet works, we can be sure any problems we run into during later phases of development originate in the client. EchoTest is a simple custom HTTP client that connects to the EchoServlet, sends a string, and receives the servlet's response. It is useful as a command-



line tool, and contains only a main method, which has the URL hard coded, as well as the String to be sent ("foo"):

```
URL enteredURL = newURL("http://altisimo.com/echo");
String toSend = "foo";
```

EchoTest uses this line of code to establish the connection to the servlet:

```
URLConnection con =
(HttpURLConnection)enteredURL.openConnection();
```

The main functions of EchoTest are:

- Make sure that EchoServlet can both receive and transmit information.
- Print the result to the screen.

The relevant code in EchoTest is shown in Listing 1.

## Coding the Servlet

EchoServlet is a stripped down HttpServlet – all it does is read in a String using readUTF; and then return the same String using writeUTF. readUTF and writeUTF are methods associated with DataInputStream and DataOutputStream, respectively, and are used to read and write Strings across a network connection. readUTF and writeUTF are the methods to use when you're dealing with Strings, and there are specific read and write methods for most of the other data types available in MIDP. You can find a full list of these methods in the MIDP 1.0 API Javadoc.

The code in EchoServlet that does the work is in the doPost method (we delegate our doGet method to doPost, which is a pretty standard practice to simplify maintenance and reduce the number of "moving parts" in the servlet). doPost creates a DataInputStream to read data from the incoming HttpServletRequest, and a DataOutputStream to write data out to the corresponding HttpServletResponse. It collects the data from the client using readUTF, and writes it back out using writeUTF. It's important to set the content type to "application/octet-stream" because we're sending data (a UTF String), not simple text or HTML.

We are sending our response encoded as UTF; there's no readString or writeString method available in DataInputStream or DataOutputStream, so readUTF and writeUTF are the simplest ways to transmit Strings. We need to send the response as an octet-stream because UTF characters don't have a fixed size. They can be 1–3 bytes long, and so must be handled as data. It's also good form to make sure you flush the OutputStream at the end of the method, to be sure that everything you want to send gets out (see Listing 2).

The servlet binary .ear file and EchoTest source code are available for download from [www.altisimo.com/download/echo.ear](http://www.altisimo.com/download/echo.ear) and [www.altisimo.com/download/echo-1.0.zip](http://www.altisimo.com/download/echo-1.0.zip) so you can experiment with them if you wish. Typical usage should look like Figure 1.

## The J2ME Client, Phase I

Now that we know the J2EE portion of the application is functioning correctly, it's time to develop our J2ME client. For this implementation we'll use a single-class solution, EchoClient.



Figure 1 Deploying EchoServlet and running EchoTest

Once again, we won't worry about collecting an input String from the user and will just send "foo" each time, as we did in EchoTest. At this point, we're concerned with making sure that the mobile device can establish the connection and send and receive data. We'll add a way for users to input a String of their choice later.

The main objectives in this section of the implementation are the same as in the development of EchoTest:

- Establish the connection to the Echo server
- Display the results on the screen

Opening a connection using J2ME is slightly different than in J2SE, so we decompose it into three smaller methods (openConnection, openConnectionOutputStream, and openConnectionInputStream) to make the code more readable and to easily allow for the substitution of different methods. The three methods are called when the client code executes outputEcho, which establishes the connection, sends and receives the data, and displays the results to the user. Detailed discussions of each of these methods follow the outputEcho in Listing 3.

The first thing we do in outputEcho is call openConnection, which connects to the URL specified and sets the User-Agent, Content-Type, and Request-Type:

```
String service URL = getAppProperty ("Echo-servlet-URL");
HttpConnection connection = (HttpConnection)
Connector.open(serviceURL);

connection.setRequestProperty("User-Agent",
    System.getProperty("microedition.profiles"));
connection.setRequestProperty("Content-Type",
    "application/octet-stream");
connection.setRequestMethod(HttpConnection.POST);

return connection;
```

Next, outputEcho calls openConnectionOutputStream, which consists of a single line of code that returns the HttpConnection's DataOutputStream:

```
return connection.openDataOutputStream();
```

After that, outputEcho gets the HttpConnection's inputStream by calling openConnectionInputStream. There are some checks in openConnectionInputStream that were not required when opening the DataOutputStream. openConnectionInputStream's first task is to get an HTTP status code from the connection. The status code comes from the server (status codes are defined in RFC 2616). If the HTTP status code indicates that the URL will be redirected or has been redirected, then we cache the new URL to improve the performance of subsequent requests. If the status code indicates that our URL is valid, we return a DataInputStream; otherwise we print a message to system.out to tell the user that the DataInputStream could not be opened. In a truly robust program, this condition should bring up an error screen on the device to alert the user to the failure, but for now we'll be content with an error appearing in the emulator console (see Listing 4).

Once a successful connection has been established and the data has been both written from and received by the MIDlet, we construct a Form to display the result to the user. There's no way to simply print to the display, as we would have been able to do to System.out or System.err in J2SE, so we must use a Form. The Form contains only the String that was returned from the server. To build the Form, we first instantiate a new Form object with the title "Echo Output" and then append the returned String. Once we have placed the data in the Form, we add an Exit Command, which we created earlier as a class-scope variable, and assign a

CommandListener, in this case the EchoClient class. Then we use `display.setCurrent` to display the new Form to the user.

```
Form outputForm = new Form("Echo Output");
outputForm.append(fromServer);

outputForm.addCommand(EXIT);
outputForm.setCommandListener(this);

display.setCurrent(outputForm);
```

## The .jad File

J2ME applications are traditionally installed via a .jad file, a descriptor file that the mobile device uses to ensure that it has enough room to store and run the application. The .jad file for our Echo client looks like this:

```
MIDlet-1: Echo,,com.altisimo.j2me.echo.client.EchoMIDlet
MIDlet-Name: Echo
MIDlet-Version: 1.87
MIDlet-Vendor: Altisimo Computing
MIDlet-Jar-URL: echo.jar
MIDlet-Jar-Size: 26096
Echo-Servlet-URL: http://altisimo.com/echo
```



Figure 2 Iteration results

The first six lines of the file are required in every .jad file, and are for the most part self-explanatory. The first line holds three pieces of information: (1) the name of the first MIDlet, (2) the icon to display in association with the MIDlet (in this case there is no icon, so it's left blank), and (3) which class to run in the .jar file to start the MIDlet. If there were more than one application in this MIDlet suite, we would include an additional line for each one, beginning with MIDlet-2, etc. It's important to ensure that the MIDlet-JAR-Size exactly matches the size of the .jar file, or devices will not be able to successfully download the application.

The last line in the .jad file, Echo-Servlet-URL, is one that we added. Like a J2EE deployment descriptor, we can use the .jad file to set some application properties outside of the code. A major advantage to putting the URL in the .jad file is that you're able to change the target of the application without having to recompile and repackage everything.

When users run the client, there's a short



Figure 3 Input UI Form

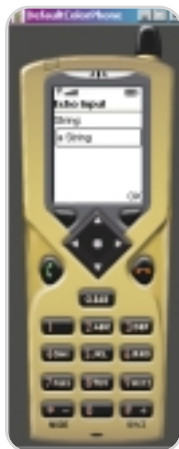


Figure 4 User has input a string



Figure 5 Show output UI

pause as it talks to the Echo server, and then they see a screen that looks like Figure 2. This figure shows that the Echo server sent the string "foo" back to the client. So far, so good.

## The J2ME Client, Phase II

Now that we have successfully established and used a connection to a remote server, it's time to add some functionality for the users. First, we'll allow users to enter the String that they want to be echoed by the server. This is similar to what would be done if we were building a Swing-based GUI, although stripped down significantly. Once again, we use a Form to gather the information from the user. To get the user's input, we create a TextField and append it to the Form. A nice feature of TextField is that the label for the field is directly associated with the TextField, so they'll be rendered next to each other on the client device.

```
Form inputForm = new Form("Echo Input");
inputField = new TextField("String:", "", 10, TextField.ANY);
inputForm.append(inputField);

inputForm.addCommand(OK);
inputForm.setCommandListener(this);

display.setCurrent(inputForm);
```

To get the String that the user entered, we call `getString` on the TextField. As a result, we change the first line of `echoOutput` to read as follows:

```
String fromUser = inputField.getString();
```

The input gathering and output screens look like Figures 3–5.

This is the only other change we need to make to allow the user to input a String to be echoed. To be thorough, we really should validate the String to make sure that it's not null or empty, since that will cause a `RuntimeException`, but we'll put this off until later.

## OTA Provisioning

Now that we have a working Echo client, we need to let our users install it. The easiest way to distribute an application is to make the .jad and .jar files available for download on a Web site. More often than not, a new user is going to download your application directly to a handheld device, not to a desktop PC, so you need to make your .jad and .jar files available on a site that is accessible to the device's very limited Web browser. Most devices have small displays, slow network connections, and limited HTML browsers, so make the page simple. By the time a user decides to download an application, they're interested in just that, not pretty pictures and other fluff.

This whole process of finding, downloading, and installing applications is called over the air (OTA) provisioning. Provisioning is jargon from the telecom world and is fairly synonymous with distribution or deployment; however, since the majority of the time we use mobile phones to accomplish this task, it seems fitting to use that industry's term.

You need to make one back-end change to your Web server for OTA provisioning to work correctly: you must update your Web server's configuration to serve up the .jad and .jar files as the correct file types. Optionally, you can use an OTA provisioning servlet that sets the MIME types for you, but we found it easier to just go in and make the change manually. We added a few lines to our Apache installation's `conf/httpd.conf` file:

```
# For J2ME OTA provisioning:
AddType text/vnd.sun.j2me.app-descriptor jad
AddType application/java-archive jar
```





Figure 6 OTA provisioning Web page

Once the .jad and .jar files are available and the changes have been made to your Web server configuration, all a user has to do is point the browser to the location of your .jad file and follow the prompts the device provides. A few clicks later and the application should be up and running on the user's device. Listing 5 provides a simple OTA provisioning Web page and Figure 6 provides a rendering of how it might look on your handheld browser.

That's it; you now have a connected MIDlet and all the information needed to install that MIDlet on a mobile device.

of features we still need to implement, which we'll address in another article:

- **Architecture:** Our client is complex enough that we should be using multiple classes to handle the various tasks. We will rework the code to follow the architecture of Sun's J2ME blueprint.
- **Progress indicator:** We should tell users where we are in the communication process between the mobile device and the servlet. In the current implementation, they see a blank screen during the network round-trip and have no idea why it takes so long. We need to set users' expectations so they don't get angry at the application.
- **Stop button:** We should allow users to interrupt the connection. Users currently have no way to stop the network round-trip to check their calendar or do anything else on the handheld device.
- **Input validation:** The J2EE component of our application will throw an Exception if the String it receives is null or empty. We can resolve this issue by adding a preliminary client-side test before we send the data to the servlet.
- **Exception handling:** There are a variety of other issues that can arise during the execution of our application. We'll explore how to identify and handle many of these issues. ☹

## What's Left

We have a working connected MIDlet, but this J2ME client is far from complete. In its current state, our client only works in the best case scenario and does not give the user the option to cancel a transaction once it has begun. There are a number

### Listing 1

```
OutputStream outStream = con.getOutputStream();
DataOutputStream dataOutStream = new
DataOutputStream(outStream);

dataOutStream.writeUTF(toSend);

InputStream inStream = con.getInputStream();
DataInputStream dataInStream = new
DataInputStream(inStream);

//get the response from the server
fromServer = dataInStream.readUTF();
System.out.println("Reply from Servlet: " + fromServer);
```

### Listing 2

```
InputStream in = req.getInputStream();
OutputStream out = res.getOutputStream();
DataInputStream dataIn = new DataInputStream(in);
DataOutputStream dataOut = new DataOutputStream(out);

//read in the value passed in
String passedIn = dataIn.readUTF();

//use this value to send back
String toReturn = passedIn;

res.setContentType("application/octet-stream");

//write it out to the response
dataOut.writeUTF(toReturn);
dataOut.flush();
```

### Listing 3

```
private void outputEcho() {
    ...
    String fromUser = "foo";

    HttpURLConnection con = openConnection();

    DataOutputStream dataOutStream =
        openConnectionOutputStream(con);
    dataOutStream.writeUTF(fromUser);

    DataInputStream dataInStream =
        openConnectionInputStream(con);

    //get the response from the server
    String fromServer;
    fromServer = dataInStream.readUTF();

    //close all the connection pieces and free resources
    closeConnection(con, dataOutStream, dataInStream);
    //display the results to the user
    Form outputForm = new Form("Echo Output");
    outputForm.append(fromServer);
```

```
outputForm.addCommand(EXIT);
outputForm.setCommandListener(this);

display.setCurrent(outputForm);
...
}
```

### Listing 4

```
int responseCode = connection.getResponseCode();

if (responseCode == HttpURLConnection.HTTP_OK ||
    responseCode == HttpURLConnection.HTTP_CREATED) {
    if (responseCode == HttpURLConnection.HTTP_CREATED) {
        // If the server forwarded to another URL, use that
        URL
        // next time.
        serviceURL = connection.getHeaderField("Location");
    }

    DataInputStream inputStream =
        connection.openDataInputStream();

    return inputStream;
} else {
    System.out.println("openConnectionInputStream failed!!!");
}
```

### Listing 5

```
<!DOCTYPE html PUBLIC "-//W3C//DTD HTML 4.01
Transitional//EN">
<html>
<head>
    <meta http-equiv="content-type"
        content="text/html; charset=ISO-8859-1">
    <title>Altisimo Computing</title>
</head>
<body>
    <h1>Altisimo Computing</h1>
    <p>Altisimo is your source for J2ME applications,
        consulting, and training.<br>
        Call us at 888-868-4306.<br></p>

    <p>Downloads:<br></p>
    <ul>
        <li><a href="hello.jad">Hello World</a>
            (license:
            <a href="LICENSE-GPL-MULTIPART">GPL</a>)</li>
        <li><a href="echo.jad">Echo</a>
            (license: <a href="LICENSE-GPL-MULTIPART">GPL</a>,
            <a href="LICENSE-Sun">Sun</a>)</li>
    </ul>
</body>
</html>
```



**Alex Bourgeois** is a cofounder of Altisimo Computing, a J2ME consulting and training company. He has been programming in Java since 1997 and has developed numerous applications for private and commercial use. Alex also cochairs the New England Java Users Group J2ME Special Interest Group.



**Richard Kasperowski**, a cofounder of Altisimo Computing, has been programming in Java since 1997 and has developed numerous applications for private and commercial use. Richard is also the cochair of the New England Java Users Group J2ME Special Interest Group.

jdj@altisimo.com

# 4<sup>TH</sup> Annual International Deve



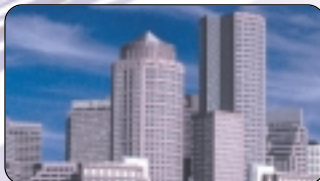
DEVELOPMENT TECHNOLOGIES EXCHANGE

**February 24-26, 2004**  
**Hynes Convention Center, Boston, MA**

**ARCHITECTING JAVA, .NET, WEB SERVICES, MX, XML, AND  
OPEN SOURCE**



Hynes Convention Center  
Boston, MA



## PROVIDING DEVELOPERS THE LATEST IN:

- Application Integration
- Desktop Java
- Mobility
- ASP.NET
- VS.NET
- JavaServer Faces
- SOA
- Interoperability
- Rich Internet Applications

Preregister for **FREE** full-day tutorials featuring the hottest Java, .NET, XML, Web Services, and Security issues. (ADVANCE REGISTRATION REQUIRED)

### **FREE** Exhibition Floor (ADVANCE REGISTRATION REQUIRED)

Meet with industry-leading companies showcasing the newest products and services.

- Product Demonstrations Theater
- Opening Night Reception
- Internet Café
- Case Study Forum

Register By  
December 19, 2003  
**SAVE** Up  
To  
**\$400**



**MX**

**WebSphere**

**XML**

## For more information

visit [www.sys-con.com/edge](http://www.sys-con.com/edge)

call 201 802-3069

e-mail [events@sys-con.com](mailto:events@sys-con.com)

Over 100 participating companies will display and demonstrate over 300 developer products and solutions.

Over 3,000 C-Level Executives, Systems Integrators, System Architects, Developers, and Project Managers will attend the conference and expo.

Over 100 sessions – training, certifications, seminars, case studies, and panel discussions – promise to deliver real-world benefits, the industry pulse, and proven strategies.

## Full-Day Tutorials Targeting

- Java
- .NET
- XML
- Web Services
- Open Source
- Security

Conference program available online!  
[www.sys-con.com/edge](http://www.sys-con.com/edge)

PLATINUM SPONSOR: **macromedia**

GOLD SPONSORS:

**Microsoft**



Computer Associates

SILVER SPONSOR:

**Novell**

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

**LINUXWORLD**

**LINUXWORLD.COM**

**JAVA DEVELOPER JOURNAL**

**WebSphere**

**JavaWorld**

**WebServices**

**XML JOURNAL**

**SD Times**

**eaIndustry**

**PerfectXML**

**NET JOURNAL**

**asp.netPRO**

**WebSphere**

**WebLogic**

**wireless**

**CE Advisor**

**ColdFusion**

**ASP**

**HSP**

**PowerBuilder Journal**

**MX**



# Developer Conference & Expo

## Attend a **FREE** One Day Security Tutorial Presented By



### Strategies for Web Services Security Success

Are you a developer, software architect, IT operator, or security administrator deploying or planning to deploy XML Web services? If so, this technical seminar is designed to arm you with the practical information and best practices you need to securely deploy XML Web services in your environment. Many questions will be addressed, such as: Why do Web services need special security? What standards are being created and what do they address? How do I leverage my existing environment to secure Web services? What are the different architectural and technical approaches to solving the problem? How can I centrally manage security in a decentralized environment?

### Course Highlights/Benefits

- Overview of XML Web services security: Why is it important?
- Discussion of various standards (WS-security, SAML, XML-Enc, XML-Sig, XKMS)
- Architectural considerations
- Malicious Web services attacks
- Strategies for securing XML Web services today and in the future
- "Nuts and bolts" demonstrations of security solutions

As part of the tutorial, we will show you how to secure your entire XML environment without adding any additional code.

Seating is limited.

Full-conference attendees will receive priority seating for all tutorials; all other seating is on a first-come, first-served basis.

## Attend a **FREE** One Day Tutorial Presented By

The Microsoft logo, consisting of the word "Microsoft" in a bold, sans-serif font.

### .NET—The Smart Client Perspective

For the business world, one of the most exciting promises of the information age is the opportunity to provide employees with access to information and the tools to act on that information whenever and wherever they need to. To a certain extent, that promise has already been achieved. Today, most businesspeople work on PCs that provide access to information, applications, and resources far beyond the boundaries of their local machine.

There are limits, however. Today's Internet model for information and application distribution assumes access to a network connection, but ubiquitous Web connectivity still lies in the future. And some computing tasks require robust functionality that can only be provided efficiently by "rich" client applications that reside on the local computer.

A challenge arises when your organization requires both the flexibility and immediacy that comes with online access to data and applications, and the full functionality of traditional client software.

The answer: smart client software.

This day-long tutorial focuses on developing and deploying smart client applications.

Seating is limited. Full conference attendees will receive priority seating for all tutorials, all other seating is on a first-come, first-served basis.

VISIT [www.sys-con.com/edge](http://www.sys-con.com/edge) FOR DETAILS and REGISTER TODAY

The program, including topics and times, is subject to change. Please refer to [www.sys-con.com](http://www.sys-con.com) for all updates.



# JAVA SESSIONS

## ■ Aspect-Oriented Programming and Java

This session introduces Aspect-Oriented Programming (AOP) and how it applies to enterprise Java application development, with an emphasis on applications for service-oriented architectures such as Web services. AOP has become a major topic in the future of enterprise Java development. This session will present a conceptual road map, including tangible examples of how AOP works, and provide an understanding of both the potential and challenges of applying AOP in a J2EE context.

## ■ Squeezing Java

Java is a very powerful language; while it offers the developer a rich array of tools, the fundamentals mustn't be overlooked. Improving your code at the core layer will result in great improvements in efficiency and produce (hopefully) fewer bugs. We'll look at the do's and don'ts of programming and learn lots of hints and tips that will accelerate your Java coding.

## ■ Enterprise Architecture and Open Source

Use of open source software within the enterprise is gaining momentum. The vast majority of organizations are using some form of open source software in production environments, including Linux, Apache, and JBoss. The enterprise architecture, however, needs to incorporate the best thinking of the industry; this includes not only using open source but contributing to it.

The model in which open source software gets developed has practices that could assist an organization in becoming agile in their software development practices and allow them to develop software faster, with cheaper costs and better quality. In this session, you will learn:

- Two models of development: the cathedral and the bazaar
- The value proposition of using open source
- Harnessing the power of the mob: the value proposition of contributing to open source
- Making the build versus buy decision: additional thoughts

## ■ J2EE v1.4

Day-to-day work with deadlines makes it difficult to keep abreast of the rapidly evolving landscape of J2EE, especially given the numerous constituent J2EE technologies. J2EE v1.4 is chockful of new services that affect and benefit a wide range of enterprise development tasks. This talk will extract core material from the speaker's new J2EE Developer's Handbook and describe what's embodied in J2EE v1.4. In particular, the new Web services features provided by J2EE v1.4 will be highlighted. The talk will also briefly address those services missing from the current J2EE standards but still needed when building enterprise applications.

## ■ Apache Axis

Apache Axis is the popular SOAP engine that includes everything you need to start producing Web services. Discover just what Axis is, and how you can utilize the power of this free engine to kick start your Web services.

## ■ Empowering Java and RSS for Blogging

One of the fastest-growing areas over the last few years is the blogging community. The ease with which you can post and publish information has enabled everyone to become their own publisher. One of the powers of blogs has been the syndication of data via the RSS (XML) protocol. Discover how you can easily produce and consume RSS feeds within your Java applications for wider appeal and hook into JavaBlogs, for example.

## ■ JUnit/Ant

A defined and easily repeatable process is one of the most necessary but often least-used aspects of good software development. A defined build process ensures that your project's software is built, deployed, and tested identically each time. Without this type of control and predictability, valuable time is often lost chasing down bugs that don't exist or rejecting solutions that were only partially implemented.



A critical measure of the success of software can be found in whether or not it executes successfully. Equally important, however, is whether or not that software does what it was intended to do. JUnit is an open source testing framework that provides a simple means for developers to define their intentions of how their software should work. JUnit then provides test runners that process your intentions and verify that your code performs as intended. The result is software that not only works, but works in the correct way.

Apache's Ant is a powerful scripting tool that enables developers to define and execute routine software development tasks using the simplicity and extensibility of XML. Ant provides a comprehensive mechanism for managing software development projects, including compilation, deployment, testing, and execution. In addition, it is compatible with any IDE or operating system.

## ■ Next Phase in the Evolution of J2EE

J2EE has been making major inroads in the enterprise space for a number of years. However, it is only with the 1.4 release that we have had uniform and easy access to Web services. Discover how to leverage the new features of J2EE 1.4 and why this release is a significant milestone in the evolution of J2EE.

## ■ Simplifying J2EE Applications

J2EE is a large, complex specification for server-side, Web-enabled application development. Over the past few years, the presenter has led many teams through the J2EE jungle, trying to steer them away from the hype and keep them focused on delivering rock-solid, end-user applications. This tutorial will discuss a variety of tips, tricks, and lessons that he has learned so you and your teams can develop J2EE applications better, faster, and simpler than before.



## WEB SERVICES SESSIONS

### ■ Exploring the Dark Side

The growing use of services-oriented architectures puts pressure on application developers relying on Web services for key features of their applications. Performance, scalability, and reliability of these components affect the ability of applications to meet service-level agreements, yet can't easily be analyzed as a part of the application when developers have a problem. In fact, the Web service may be on a different software platform than the rest of the application. This session describes how developers can shed light on memory use in Web services written in either .NET or Java, even if they didn't write the code and do use another platform.

### ■ Web Services Progress Report

Web services have been the buzz for the last couple of years. There have been many technical and some market success stories but the concept remains confusing. New "standards" are proposed on a regular basis, but they overlap one another and seem to form rifts along the same fault lines as previous industry politico-strategic controversies. A group of people from the W3C Web Services Architecture working group have been arguing that many of the ideas coming from the Web services community are antithetical to the principles of the Web itself and are unlikely to ever work on an Internet scale. This presentation provides a progress report on the effort to distinguish Web services architectural principles from the marketing agenda of individual companies.

### ■ ID, Please. The Case for Giving Web Services an Identity

Without identity management, Web services can be consumed by anyone. The challenge for Web services developers is to provide appropriate access based on the user's identity. As identity management moves into the forefront of technology, directory services will evolve from simple LDAP repositories used for authentication and storage to robust engines that provide identity integration, access management, and policy enforcement. This presentation will discuss how identity management and directory services provide a robust solution for Web services authentication, authorization, and single sign-on.

### ■ Web Services Orchestration, Management, and Security: Will They Play Together?

Web services orchestration, management, and security are among the principal challenges facing implementers of service-oriented architectures today. There is still much confusion in the IT community about the standards themselves, which are at various stages of maturity. Their relevance to enterprise IT and how they might someday be able to effectively work together is often unclear. This session provides an overview of standards in these three critical areas; and more important, how each affects the other. Attendees will then gain practical knowledge and a deeper understanding of future trends and the need to address certain real-world issues in order to create a more cost-effective and agile IT infrastructure.

### ■ Service-Oriented Integration: Making the Right Choices To Support The Next-Generation of Integration

Applications are increasingly being developed "built-to-integrate," pro-

viding the ability to easily expose key functionality through commonly defined interfaces. Gartner calls this concept SODA, or service-oriented development of applications. When applied to the ever-present integration challenge, SODA represents a transition to service-oriented integration.

But making the right architectural decisions is absolutely vital to ensuring success with service-oriented integration projects – whether applications were built to integrate or not.

This presentation will examine the leading choices for supporting service-oriented integration: enterprise service buses, integration brokers, and application suite platforms.

### ■ Government Real-Time Fraud Detection Using Web Services

Government agencies are faced with increasing amounts of data and are challenged to make sense of, and act on, that data in real time. Failure to interpret and execute on data can result in security threats and, potentially, loss of life. Government agencies are increasingly investing in Web services solutions to address their need for real-time access to information.

The Canadian Passport Office is an example of a government agency leveraging Web services to exchange information in real time to combat terrorism and other illicit uses of fraudulently obtained passports. They selected IT consulting firm Pentelar and Sybase, Inc., technologies to electronically authenticate identity document data through the use of Web services and ebXML.

This session will discuss this pilot project and highlight the ebXML capabilities that enable the Canadian Passport Office to address real-time information exchange.

### ■ WS-CAF: Standardized Web Services Transactions and Composite Applications

The Web Services Composite Application Framework is a collection of three specifications – Web Service Context (WS-CTX), Web Service Coordination Framework (WS-CF), and Web Service Transaction Management (WS-TXM) – designed to solve problems that arise when multiple Web services are used in combination ("composite applications") to support information sharing and transaction processing. As coauthor of the specification, we will discuss how WS-CAF addresses the underlying issues of Web service context propagation and transaction management to expand the scope, usability, and reliability of Web services for business process automation.

### ■ Securing Web Services: What Can Be Done Today?

Security is listed as one of the main barriers to the adoption of Web services today. With the proliferation of security standards, there is a lot of confusion over which ones are mature enough to use and how they fit together. This session will present the current and emerging security standards for Web services and show how they can fit together architecturally to address various security concerns.

4

# XML SESSIONS

## ■ Universal Business Language

Web service technologies promise to revolutionize electronic business, but global interoperability of business processes cannot occur without the semantic standardization of the messages exchanged in business transactions. This session will describe the OASIS UBL project to create standard XML Schemas for basic business documents, explore the relationship of UBL-based business to traditional EDI, and note the explosive potential of standard markup combined with reliable XML messaging.

## ■ Real Best Practices for XML Web Services Management and Security

Companies deploying Web services in a meaningful way are increasingly finding they need to address Web services management and security early in the architectural phase. Basic Web services connections are easy to do, but managing the security, performance, scalability, and inevitable changes to the production environment requires some knowledge, expertise, and planning. This session cuts through the hype and outlines real-world mistakes many companies make when deploying Web services and the real best practices from companies that have successfully captured the value of XML Web services. It provides practical advice on how to successfully manage and secure your XML Web services environment.

## ■ SOA Foundation Components: Building an XML Content Router

One of the fundamental components for any burgeoning SOA will be an XML content router. This session explores the concepts, patterns, and open source software available that facilitate building an XML content routing system. The system can be exposed as a Web service or simply as a stand-alone J2EE component for use in your enterprise. The "restaurant" pattern is introduced as the principal design pattern for building the service, and this pattern's applicability to building generic services is discussed. Applying the router as an XML data integration tool is also discussed, as well as its potential for acting as a service orchestrator.

## ■ What's New in XSLT 2.0?

XSLT 2.0, which may achieve W3C Recommendation by conference time, offers unparalleled power in conjunction with XPath 2.0 for transforming XML documents. In this engaging, example-rich session, Steve Heckler demonstrates the most important new features of XSLT 2.0, including Sequences, new data types and XML Schema support, regular expressions, multiple document output, grouping, new control-flow operators, and much more. Current and future support for XSLT 2.0 on the Java and .NET platforms will also be discussed. Most examples will use Saxon, but .NET examples will be included if .NET supports XSLT 2.0 by conference time.

## ■ Using XML Schemas Effectively in WSDL Design

Developers building Web services today are beginning to see the value of using the document-style approach over RPC. Recent experi-

ence shows that to take full advantage of document-style Web services requires a strong knowledge of XML Schemas and related XML standards. This presentation presents a number of important tips and techniques for properly using XML Schemas in the design of a Web services interface (WSDL), including XML-based development tools, binding considerations between XML and underlying objects, WSDL reusability through XML Schemas, and XML Schema naming best practices.

## ■ XML: A Manager's Guide

As more and more IT projects utilize XML and its derivatives as fundamental technologies, it is key for today's manager to be aware of the various ingredients of XML. The objective of the session is to provide an essential introduction around XML from a manager's perspective. From core XML processing; transformation; metadata definition and schemas; applications in Web, wireless, and speech applications; Web services; industry-standard vocabularies; and more, this session provides a comprehensive review of the various technologies related to XML.

## ■ Using Rules to Clean Up XML

Garbage in, garbage out – it's an axiom that applies to many aspects of enterprise development, but none more so than building reliable and robust Web applications and integration projects with XML. Since its inception, XML has been seen as the cure-all for problems related to Web applications and integration projects. However, poorly written XML can slow down an integration project, or worse, cause the integration project to collapse. The key to successfully using XML in an integration project is to first understand the inefficiencies that may cause poorly written XML, and then apply a rule-based system that establishes policies to follow.

### EXPO HALL

**TUESDAY, FEBRUARY 24,**  
11:00AM-4:00PM, Welcome Reception 4:00PM

**WEDNESDAY FEBRUARY 25,**  
11:00AM-4:00PM

## ■ XForms: Simplifying the Development of Transactional Web Forms

XForms is a W3C specification that specifies a declarative language for solving a common requirement for advanced user interaction, data validation, and XML processing. XForms is designed to be integrated into XHTML, but is not restricted to being a part of that language alone. It can be integrated into any suitable markup language. This session will give an introduction to XForms and explain how XForms fits in the client tier of the J2EE application architecture. In addition, it will cover the benefits of XForms and why it is a perfect fit for interacting with J2EE and Web services. A demonstration of XForms in a J2EE environment, using an XForms-compliant browser and a sample application, will further illustrate the advantages.



## .NET SESSIONS

### ■ .NET Compact Framework Performance Tips and Tricks

Learn the techniques that can be used to increase the responsiveness of user interface and network operations for users of applications built on the .NET Compact Framework. Look under the covers at advances and changes in the "Whidbey" release to significantly improve performance. Get a general overview of how the .NET Compact Framework works under the hood at runtime, with specific focus on performance implications. Then we will cover general user interface tips to increase performance. Explore how asynchronous infrastructure, such as threading, in the .NET Compact Framework can be leveraged to optimize both user interface and network operations. Learn about the architectural guidelines for creating applications that perform well under frequently changing network conditions.



### ■ Best Practices and Techniques for Building Secure ASP.NET Applications

When the enterprise depends on your application, careful attention to security is essential. This session provides specific recommendations to follow when developing secure ASP.NET Web applications and services, and focuses on the details of configuring IIS for security. Understand how to use authentication, authorization, threat modeling, configuration settings, and secure database access to create secure systems and learn common coding techniques for storing secrets, error handling, data validation, and code access security.

### ■ Using the Enterprise Instrumentation Framework

The Microsoft .NET Framework 1.1 and Windows Server 2003 offer a number of new features to help developers instrument their code. In this session we'll learn about the challenges facing application management in today's distributed world. We will examine the new unified instrumentation API in the Enterprise Instrumentation Framework (EIF), including the new Windows Event Trace available in Windows Server 2003, configurable at-source event filtering, and how request-based event tracing using EIF allows you to put a request context around the trace messages that map to a business process flow in your application. We will also discuss the benefits of using EIF in your application for both the developer and the application administrator.

### ■ .NET Framework: Exploring What's New in the Base Class Library for "Whidbey"

The base classes serve as the essential libraries for any developer. Continued evolution of the base classes provides numerous benefits,

including the ability to develop more reliable, faster solutions, easier-to-write code, and more solutions entirely in managed code. Take a look at the many features that are a part of that evolution, including features in IO, event-logging, and various features in System.Collections.Generic classes and interfaces.

### ■ Microsoft Office 2003: A Solutions Platform

For all developers who would like to integrate custom business solutions with Microsoft Office products, this session will introduce you to the expanded developer features that have been included in the newest version of Microsoft Office. Come explore new XML-based programmability in everything from Word 2003 and Excel 2003 to FrontPage 2003 and SharePoint. Build powerful, modular solutions with Web services. Learn about InfoPath 2003 support for XML standards. Discover how to use the Microsoft Visual Studio Tools for the Microsoft Office System to automate and extend Microsoft Office Word 2003 and Microsoft Office Excel 2003 using Visual Basic .NET and Visual C# .NET. More than ever, Office has a solution for you.

### ■ BizTalk Server 2004 Technical Drilldown

Biztalk Server 2004 is designed to enhance Enterprise Application Integration (EAI), Business Process Automation (BPA), and Information Worker Integration. Join us for a technical drilldown into the new features and toolsets available.

### ■ Moving your Architecture to .NET

This session's emphasis is on how to migrate existing business components from VB6 COM objects to VB.NET assembly components. We'll spend time discovering how to best move different tiers of a multi-tiered application from COM to .NET, as well as effective strategies on how to wrap existing COM components for interoperability. We'll also examine best practices for moving your application from a COM-based architecture to a .NET-based architecture.

## Who Should Attend

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



**VISIT [www.sys-con.com/edge](http://www.sys-con.com/edge) FOR DETAILS and REGISTER TODAY**

The program, including topics and times, is subject to change. Please refer to [www.sys-con.com](http://www.sys-con.com) for all updates.

## MX SESSIONS

### ■ Enterprise Infrastructure for Rich Internet Applications

Learn how Macromedia's technology initiative "Flex" fits seamlessly into today's new service-oriented architectures (SOA). We'll cover design patterns for rich clients, accessing Web services, and securing your Flex application.

### ■ ColdFusion Components from the Ground Up

ColdFusion Components (CFCs) are considered to be the most important enhancement to the CFML language since it was created some eight years ago. CFCs combine the power of objects with the simplicity of CFML, and in this session, you'll discover that not only are they incredibly powerful, they're also remarkably easy to master.

### ■ Code-Based Rich Internet Applications

Learn how to use Macromedia's technology initiative "Flex" to create rich Internet applications. This session will cover using components, layouts, and managers to build user interfaces, as well as using Flex's XML-based language to create and manipulate client-side data models.

### ■ Tips and Tricks for Writing and Using CFCs

ColdFusion Components are simple to write and simpler to use. But that simplicity hides a series of powerful features and technologies that you can (and should) take advantage of. In this session, you'll learn how to use (and how to not use) inheritance, "super," persistence, constructors, and more.

### ■ Leveraging Web Services

Web services technology is changing the way we think about designing and building applications. Come and learn what all the fuss is about, find out exactly which problems Web services solve, see Web services created and used, and even discover how Web services expose the world of .NET.

### ■ Building an RIA with Macromedia Flash and ColdFusion Web Services

Learn about the quickest and easiest way to build rich Internet applications using Macromedia Flash with connections to Web services built in ColdFusion.

### ■ ColdFusion Components

ColdFusion Components combine the power of objects with the simplicity of CFML. This is the way object-based development was intended to be, and in this session, you'll learn about this combination first hand. Starting with a simple data-driven application, you'll gradually convert it into a highly scalable and manageable multitier application, and in the process, will be amazed at just how easy ColdFusion makes this process.

### ■ Rapidly Build Web Services Applications with ColdFusion and Studio MX

The last year has shown that Web services are not just another passing fad and their promise of platform-independent distributed applications has been realized. Compared to other application server platforms, ColdFusion makes creating Web services easy. This session covers how to create a ColdFusion Component (CFC) in Dreamweaver, as well as how to expose that CFC as a Web service by just toggling one attribute of the CFC. That's right: in ColdFusion, it is just that easy.

### ■ Using Macromedia Flash with Web Services

Web services, a technology that allows developers to execute remote procedures, is emerging as a revolutionary tool for Web application development. Macromedia Flash MX 2004 Professional is a powerful tool for building applications that consume Web services built in any technology, including Macromedia ColdFusion, Java, ASP.NET, and PHP. In this session you will explore the visionary computing model that Web services represent as you use Macromedia Flash components to develop a Web service-based application. You will learn how to discover Web services, work with data and UI components, perform data binding, examine security issues, and aggregate multiple Web services into a cutting-edge Web service consumer.

### ■ Using Web Services with ColdFusion

You're a ColdFusion user and want to be able to take advantage of Web services? You're in luck. No other language or platform makes Web services consumption as painless as ColdFusion does. In this hands-on session, you'll experience Web services for yourself by building a complete application around some of the most popular Web services available today.



**Sheraton Boston Hotel**  
Official Hotel of Edge 2004 East



**CONFERENCE: Feb. 24 – 26, 2004 EXPO: Feb. 24 – 25, 2004**

**Hynes Convention Center, Boston, MA**

## THREE WAYS TO REGISTER FOR CONFERENCE

- 1) On the Web:** Credit Cards or "Bill Me." Please make checks payable to SYS-CON Events.
- 2) By Fax:** Credit Cards or "Bill Me" 201-782-9651
- 3) By Mail:** 135 Chestnut Ridge Road, Montvale, New Jersey 07645, Attention: Registration

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

Please complete sections 1, 2, 3 and 4

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

First Name \_\_\_\_\_ Last Name \_\_\_\_\_  
 Title \_\_\_\_\_  
 Company \_\_\_\_\_  
 Street \_\_\_\_\_  
 Mail Stop \_\_\_\_\_  
 City \_\_\_\_\_  
 State \_\_\_\_\_ Zip \_\_\_\_\_ Country \_\_\_\_\_  
 Phone \_\_\_\_\_  
 Fax \_\_\_\_\_ E-Mail \_\_\_\_\_

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

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

Check # \_\_\_\_\_ Amount of Check \$ \_\_\_\_\_

Charge my ☐ Visa ☐ MasterCard ☐ American Express ☐ Discover

Name on card \_\_\_\_\_

Card # \_\_\_\_\_ Exp. Date \_\_\_\_\_

Signature \_\_\_\_\_

Billing Address (if different from mailing address) \_\_\_\_\_

## 3 PLEASE INDICATE

### YOUR CONFERENCE CHOICE

Total Registration fee \$ \_\_\_\_\_

	Before 12/19/04	Before 1/24/04	Before 2/21/04	Onsite
<input type="checkbox"/> <b>GP: Gold Passport Three-Day Conference</b> Good for all three days of the conference program, including keynotes, all conference sessions & one tutorial	\$1,395.00	\$1,495.00	\$1,695.00	\$1,795.00
<input type="checkbox"/> <b>3D: Three Day Conference</b> Good for all three days of the conference program, including keynotes, all conference sessions (excludes tutorials).	\$1,295.00	\$1,395.00	\$1,495.00	\$1,695.00
<input type="checkbox"/> <b>2D: Two Day Conference</b> Good for two days of the conference program, including keynotes, all conference sessions (excludes tutorials).	\$995.00	\$1,195.00	\$1,295.00	\$1,395.00
<input type="checkbox"/> <b>1D: One Day Conference</b> Good for one day of the conference program, including keynotes, all conference sessions (excludes tutorials).	\$595.00	\$695.00	\$695.00	\$795.00
<input type="checkbox"/> <b>VIP PASS FREE with preregistration</b> Select one: <input type="checkbox"/> FREE - The Smart Client Perspective (Feb. 24) <input type="checkbox"/> FREE - Strategies for Web Services Security Success (Feb. 25)	FREE	FREE	FREE	\$99.00
<input type="checkbox"/> <b>EO: Expo Hall Only</b> Includes keynotes, expo presentations, and workshops.	FREE	FREE	FREE	\$50.00

### SPECIAL DISCOUNTS AVAILABLE

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

4

### A. Your Job Title

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

### B. Business/Industry

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

### C. Total number of employees at your location and entire organization (check all that apply):

	Location	Company
10,000 or more	01 <input type="checkbox"/>	01 <input type="checkbox"/>
5,000 – 9,999	02 <input type="checkbox"/>	02 <input type="checkbox"/>
1,000 – 4,999	03 <input type="checkbox"/>	03 <input type="checkbox"/>
500 – 999	04 <input type="checkbox"/>	04 <input type="checkbox"/>
100 – 499	05 <input type="checkbox"/>	05 <input type="checkbox"/>
100 or less	06 <input type="checkbox"/>	06 <input type="checkbox"/>

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

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

### E. What is your company's gross annual revenue?

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

### F. Do you recommend, specify, evaluate, approve or purchase wireless products or services for your organization?

01 ☐ Yes 02 ☐ No

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

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

SYS-CON Events, Inc., and SYS-CON Media make no warranties regarding content, speakers, or attendance. The opinions of speakers, exhibitors, and sponsors do not reflect the opinion of SYS-CON Events and SYS-CON Media, and no endorsement of speakers, exhibitors, companies, products, or sponsors is implied.



If you require special assistance covered under the Americans with Disabilities Act, please call 201-802-3058 by February 10, 2004.

**CANCELLATIONS, SUBSTITUTIONS, REFUNDS**  
 Fax written request to SYS-CON Registration 201-782-9651. Requests for refunds received prior to January 16, 2004 will be honored less a 10% handling charge; requests received after January 16, 2004 and before February 6,

2004, will be honored less a 20% handling charge. No requests for refunds will be honored after February 6, 2004. Requests for substitutions must be made in writing prior to February 20, 2004. No one under 18 is permitted to attend. No warranties are made regarding the content of sessions or materials.

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

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

# Teamstudio Analyzer for Java

Reviewed by  
Maxim Chpakov

**W**hat is every Java developer's nightmare? Maintaining code, even if he or she has written it. Code is often chaotic and incomprehensible, mostly due to nonuniform coding styles.

For decades, premier software vendors realized that uniformity in projects cannot be assured without additional inspections during development. Teamstudio is one of these software vendors. The company has made a name for itself with its software tools for Lotus Notes and Domino; now it also develops software tools for Java. Teamstudio Analyzer for Java automatically inspects your code and provides control over a uniform coding style within your projects.

## Product Description

Teamstudio Analyzer for Java is a plug-in that supports most of the popular Java IDEs. It inspects Java code based on 192 predefined rules, and produces a number of reports with coding style violations. The tool can also auto-fix problems according to predefined patterns. The 192 predefined rules are Java and J2EE best-practice patterns. The rules are configurable and settings can be exported to, or imported from, a file. Users can configure the existing rules to their needs and also write their own rules, e.g., project specific rules, as desired.

## Installing and Using Teamstudio Analyzer for Java

Teamstudio Analyzer for Java is available for download from [www.teamstudio.com](http://www.teamstudio.com). You'll need to register (at no cost) in order to gain access to the download page. Next, you can obtain an evaluation copy from Teamstudio's Web site ([www.teamstudio.com/javaj](http://www.teamstudio.com/javaj)). My personal experience with this was very good. Their customer service is excellent and the contacts are pleasant. Teamstudio always answered by e-mail within two hours.

Installation is very simple. Start the downloaded setup file and follow the wizard. After a few simple questions, you're equipped with the tool in less than one minute. Teamstudio Analyzer

for Java version 2.1 currently supports the following IDEs:

- Eclipse release 2.0.x
- JBuilder releases 7, 8, and 9
- Oracle9i JDeveloper release 9.0.3
- NetBeans release 3.4.x and 3.5.x
- Sun ONE Studio 4
- WebSphere Studio Application Developer (WSAD) for Windows release 5.x

You can also use this tool with Apache Ant (1.4.-1.5.3), binding it to its build script. I used Analyzer for Java within Eclipse 2.0, JBuilder 6, WSAD, and Ant.

Teamstudio has also written some good, detailed installation documentation.

Teamstudio Analyzer for Java is ready to use immediately after installation. It's user friendly so you don't need any special knowledge about best practices, coding patterns, or how to carry out any complicated configuration steps. The 192 predefined and totally preconfigured rules encompass both Java coding patterns and best practices, and Java language and J2EE specifications. Based on my experience, this totally suffices for most J2EE and Java projects. However, if you still require additional rules, Teamstudio gives you full control over user-defined rules. You can extend the `com.teamstudio.analyzer.user.JavaUserRule` class and fill it with your own func-

## Teamstudio, Inc.

900 Cummings Center  
Suite 326T  
Beverly, MA 01915  
**Web:** [www.teamstudio.com](http://www.teamstudio.com)  
**E-mail:** [sendinfo@teamstudio.com](mailto:sendinfo@teamstudio.com)  
**Phone:** 800.632.9787 (U.S.)  
**Phone:** +44 (0) 1487 772200 (Europe)

## Specifications

**Platforms:** Windows 98, 2000, XP, NT 4.0, and ME with JDK 1.3.x or 1.4.x and one of the supported IDEs, and Linux and Solaris as supported by your IDE

**Pricing:** \$295 offer includes one year of maintenance. Additional years available.

## Test Platform

1.6 GHz Intel Pentium IV, 21GB Disk, 768 MB RAM, Windows XP w/o Service Pack, Sun JDK 1.4.1, Eclipse 2.0.

tionality. You can also use all features of Analyzer within your rules, e.g., you can build your own GUI for configuration.

Teamstudio Analyzer for Java possesses extensive configuration possibilities; Figure 1 provides an example. You can combine your personal set of rules for code inspection from different rules or rule categories. Rules are configurable over a GUI.

The settings can be stored in a file and/or imported from a file. This makes teamwork simple because, once done, the settings can be problem free, spread over as many team members as desired.

Of course, there's detailed help at every stage of installation, configuration,

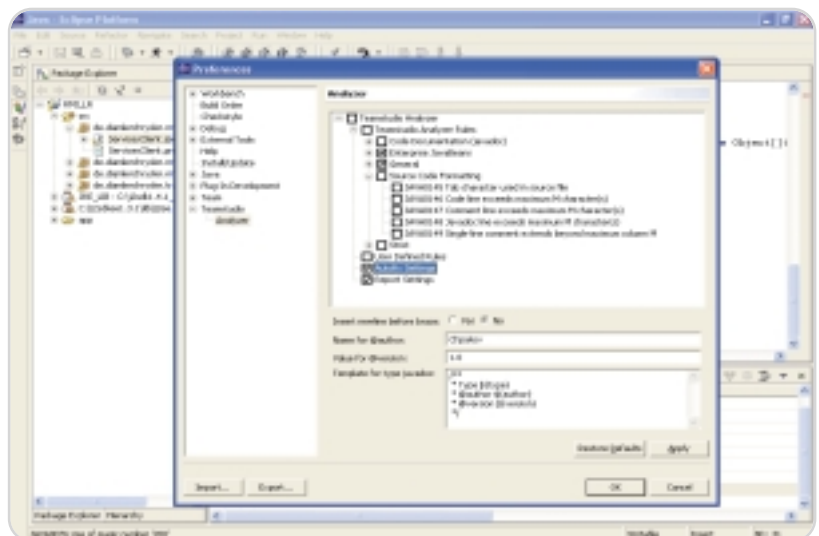


Figure 1 Configuration sample

Maxim Chpakov is a senior software engineer with DaimlerChrysler Technology, Services & Solutions, and is a valued member of the DC-TSS quality control team. Working for many years as a software developer, Maxim believes that one of the important preconditions for the success of every project is ensuring a uniform coding style.

[maxim.chpakov@daimlerchrysler.com](mailto:maxim.chpakov@daimlerchrysler.com)



and use. Teamstudio provides good documentation, which is available as a PDF file or as context-help within the IDE.

Once analysis is complete, the tool builds a report with messages about the found violations of the selected rules (see Figure 2). This report appears as warning messages in the message UI of the IDE, so that with a mouse-click you can immediately jump to the relevant place in the code where a violation of a rule has been asserted.

The reports can also be exported into an HTML or XML file. HTML reports are very intelligible and deliver all necessary information, including statistics and a complete list of violations.

An additional feature of Analyzer for Java is an autofix mechanism. If you let the tool run over the code of a middle-sized project (about 200–500 classes) for the first time, you normally get thousands of warning messages. Of course, usually only a few of these violations are critical to your project and the remaining you'd probably like to automatically correct. You can do this with Analyzer for Java. For some predefined rules you can let the tool automatically correct the violated code using the autofix function, so that they correspond to the given patterns. The autofix function does have some deficiencies. First, it's not enabled for all

of Analyzer's rules. User-defined rules cannot be automatically fixed and the configuration possibilities of autofix patterns are very limited at the present time.

The tool is well suited for any Java developer because it doesn't demand any special knowledge, but rather mediates it. If a programmer constantly develops with the support of such a tool, his or her coding style will become better. Teamstudio wrote a very intelligible and detailed description (also with code examples) of all predefined rules, so that no rule remains unexplained. This tool is very effective if used in the programming process. Even the most experienced Java gurus can learn some patterns, not to mention the beginners who will use best practices from the start, in spite of their inexperience.

### Summary

We used Teamstudio Analyzer for Java for different reasons during development and we were always absolutely satisfied with it. The quality of this tool is excellent. Analyzer for Java is an ideal product for both quality checks and an inspection of the finished code. If you develop with Java, check out Teamstudio Analyzer for Java. I'm sure that the advantages you'll get from this tool are definitely worth the time and the money you'll invest in it. ☺

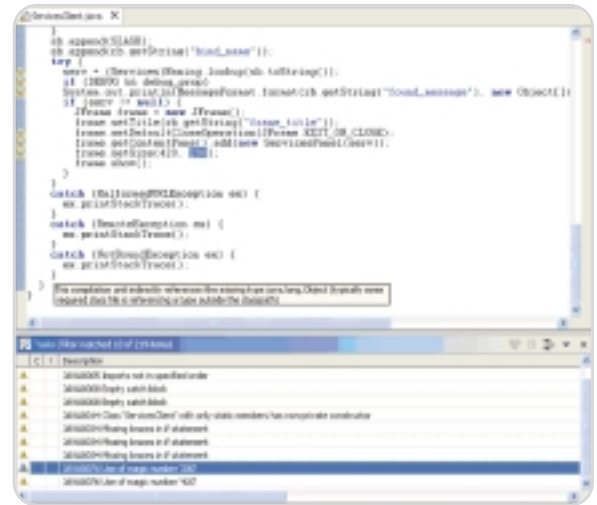


Figure 2 Working with Analyzer

### Snapshot

**Target Audience:** Java programmers, managers

**Level:** Beginner to advanced

**Pros:**

- Low price
- Good documentation
- Feature-rich product
- User friendly
- Good quality product

**Cons:**

- No significant cons

www.linuxworld.com



**The Leading Magazine  
for Enterprise and  
IT Management**

## LinuxWorld Magazine

There is no escaping the penetration of Linux into the corporate world. Traditional models are being turned on their head as the open-for-everyone Linux bandwagon rolls forward.

Linux is an operating system that is traditionally held in the highest esteem by the hardcore or geek developers of the world. With its roots firmly seeded in the open-source model, Linux is very much born from the "if it's broke, then fix it yourself" attitude.

Major corporations including IBM, Oracle, Sun, and Dell have all committed significant resources and money to ensure their strategy for the future involves Linux. Linux has arrived at the boardroom.

Yet until now, no title has existed that explicitly addresses this new hunger for information from the corporate arena. *LinuxWorld Magazine* is aimed squarely at providing this group with the knowledge and background necessary to make decisions to utilize the Linux operating system.

Look for all the strategic information required to better inform the community on how powerful an alternative Linux can be. *LinuxWorld Magazine* does not feature low-level code snippets but focuses instead on the higher logistical level, providing advice on hardware, to software, through to the recruiting of trained personnel required to successfully deploy a Linux-based solution. Each month presents a different focus, allowing a detailed analysis of all the components that make up the greater Linux landscape.

**Regular features include:**

*Advice on Linux Infrastructure*

*Detailed Software Reviews*

*Migration Advice*

*Hardware Advice*

*CEO Guest Editorials*

*Recruiting/Certification Advice*

*Latest News That Matters*

*Case Studies*



LINUXWORLD® IS THE REGISTERED TRADEMARK OF INTERNATIONAL DATA GROUP, INC.



SYS-CON MEDIA IS THE REGISTERED TRADEMARK OF INTERNATIONAL DATA GROUP, INC.

**SAVE 30%  
OFF!**  
REGULAR ANNUAL COVER PRICE \$71.76  
**YOU PAY ONLY  
\$49.99**  
12 ISSUES/YR

\*OFFER SUBJECT TO CHANGE WITHOUT NOTICE

**SUBSCRIBE  
TODAY!**

WWW.SYS-CON.COM

OR CALL

1-888-303-5282

**FOR ADVERTISING INFORMATION:  
CALL 201 802.3020 OR  
VISIT WWW.SYS-CON.COM**

The World's Leading IT-Technology Publisher

# XML Tunneling Technology

(XTT) Version 3.0 by InsiTech Group, Inc.

Reviewed by  
Robert Carten

**A**t WinWin Solutions we build custom business applications for insurance, health care, and financial vertical markets. Our client demands for B2B integration over the Internet led us to look for a platform that extends our complex transaction-oriented solutions to the Internet without the penalty of a heavy client interface or having to implement complicated work flows in HTML. We needed simple, effective tools that could extend our apps without damaging them.

The InsiTech Group met our needs with its XML Tunneling Technology (XTT) version 3.0 – a tool and framework for developing ultra-thin, rich-GUI, distributed Java applications.

Currently, there are many technologies that can be used to build thin-client Internet-based B2B applications including Struts, Droplets, Canoo, etc. While some of these tools facilitate the delivery of a rich GUI interface in a thin-client distributed environment, we believe they fall short of the benefits offered by XTT's fully extensible architecture. XTT delivers a pure Java Swing thin-client application that uses XML to marshal data back and forth from the app server to the client.

Unlike Droplets and Thinlets, which utilize a Java server-side programming model, XTT allows you to distribute your business logic on the client or app server. XTT even allows you to run the application while not connected to the Internet.

When we found XTT we were excited because out of the box it provided the components necessary to handle most business problems. In fact, XTT features include pessimistic record locking, strong event processing, data streaming, built-in verifiers and formatters, as well as an intuitive VB-like RAD environment.

## What Is XTT?

The XTT Framework is a thin-client GUI application development tool for building enterprise class applications in heavy transaction-oriented market sectors such as health care, financial,

insurance, and manufacturing. XTT provides wizards and templates, enabling you to quickly build thin-client Java forms.

## Business Benefits

We found that the XTT framework enabled us to stay focused on adding business value rather than infrastructure coding. In fact, the point-and-click RAD environment allowed us to rapidly prototype concepts in the iterative early phases of a project and deliver real working forms for the client to test and approve.

XTT wizards let you create database-enabled forms simply through a point-and-click interface. Code generated by these wizards can be further modified or customized in Sun ONE Studio, NetBeans, or JBuilder IDEs.

XTT provides strong support for event control. XTT routes the data-access events, like preread and postread, to a single method. From this

## InsiTech Group, Inc.

1157 Highland Ave., Suite 209  
Cheshire, CT 06410

**Phone:** (203) 699-0029

**Fax:** (203) 699-0270

**Web:** [www.insitechgroup.com](http://www.insitechgroup.com)

## Specifications

**Platforms:** Any platform with JDK 1.4.x, Sun ONE Studio 5, NetBeans 3.5, JBuilder 8

**Pricing:** Community Edition – free. Developer version – \$499 per client connection

## Test Platform

**Client:** Toshiba Satellite Pro 700MHz, 300MB RAM, 10GB, Win2k

**Server:** Apache Tomcat on Win2k

**IDE:** NetBeans 3.5, JDK 1.4.1

method you can easily tune client- or server-side responses to deliver what you need. The event model is simple and effective. The whole team knows where to put or find the business logic.

XTT really does provide a rich user interface in a very thin client framework. In fact, the distributed component of the framework that manages communication on the client is only 216KB. This JAR is downloaded once when the application is accessed for the first time.

Robert Carten is a senior developer at WinWin Solutions, Inc., in Westwood, NJ. WinWin specializes in delivering custom applications using software from Revelation Technologies Inc. Bob has over 20 years' experience in the software world. A software consultant, he has worked with many of the major database products on Windows, Unix, and IBM platforms.

[bob@winwinsol.com](mailto:bob@winwinsol.com)

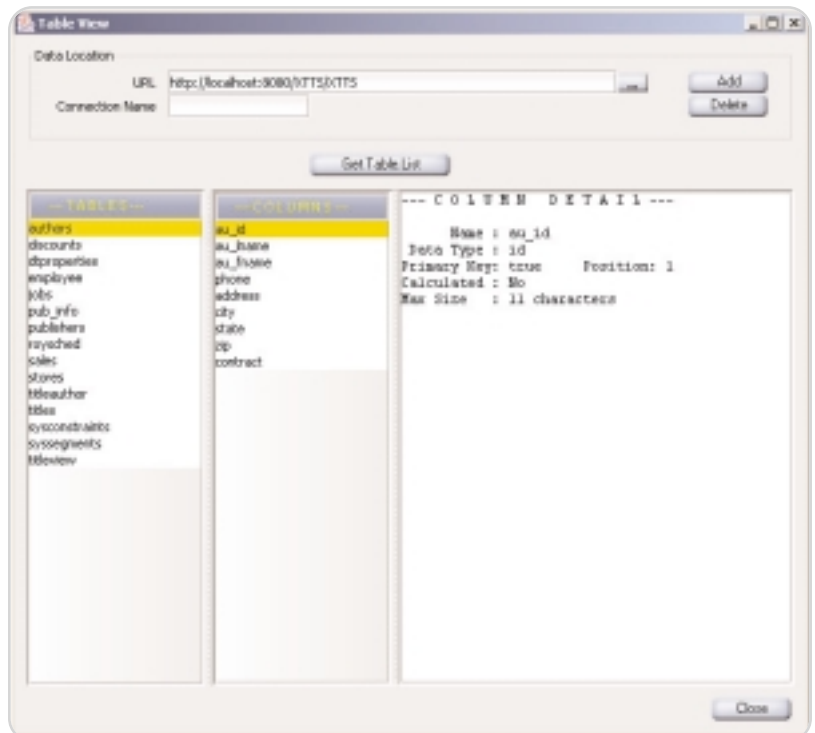


Figure 1 Accessing the data



## Creating a Form in XTT

To create a form, simply open the Form Wizard, point to the appropriate data source, select the base tables, create relationships, and generate the form. Once the form is generated, use the IDE to compile and execute it. You can then use the drag-and-drop capabilities of the IDE to refine, add business logic, or further customize a form if needed.

The process of developing a form follows these three steps:

1. TableView Wizard enables you to view the database metadata remotely without having to use the LAN console for the database (see Figure 1).

2. Form Wizard enables you to select tables, create relationships between the data, and specify the manner in which the data will be rendered (e.g., Text field, grid, pop-up, checkbox, etc.) (see Figure 2).
3. Generate and execute the form in the IDE (see Figure 3).

Here we have built an Authors Entry form that accesses three different tables and presents information to the user in the form of text fields, grids, and pop-ups. We didn't write any code by hand, rather we leveraged XTT and the IDE to generate it for us. This ability to rapidly prototype real working solutions was very important to us.

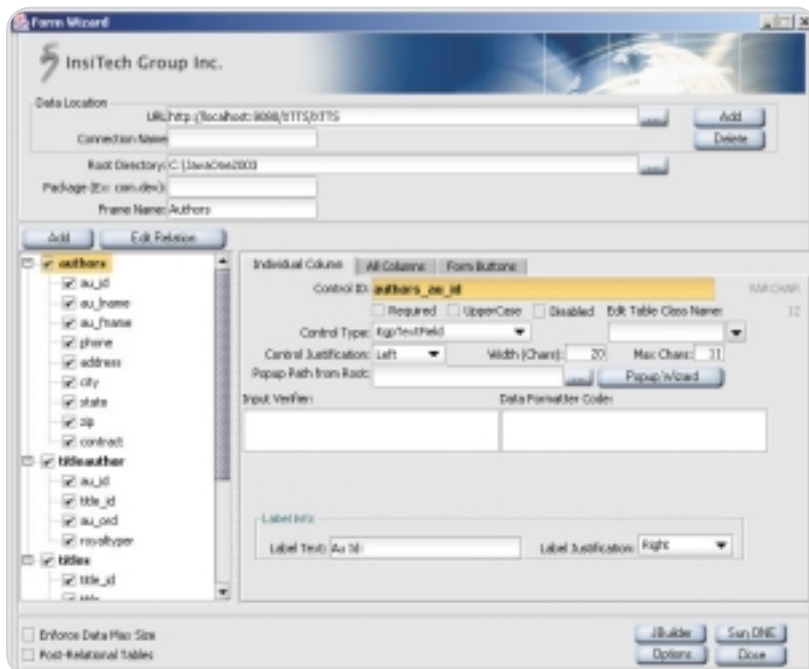


Figure 2 Creating forms using wizards

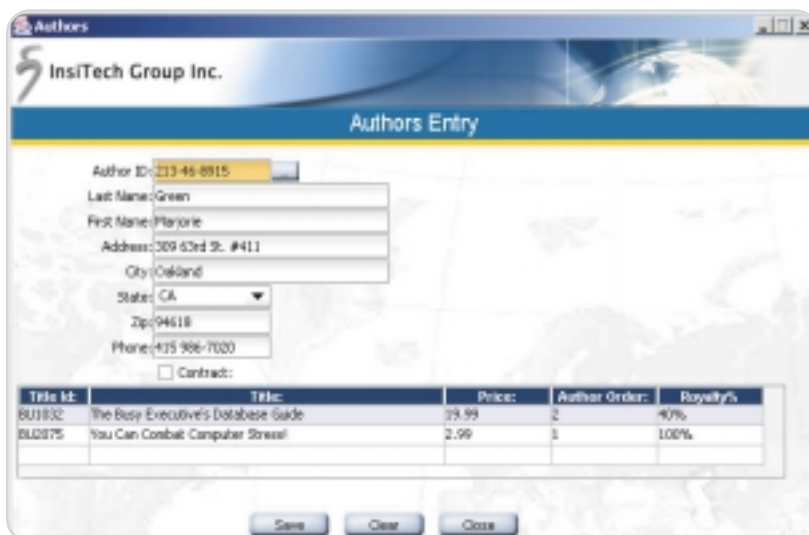


Figure 3 Generated Java form

## Cons

As with any software product there are some trade-offs. Most notably, XTT only works with IDEs that support visual rendering of JavaBeans such as Sun ONE Studio, NetBeans, and JBuilder – no support for Eclipse as of yet. Also, XTT does not support the use of UML modeling tools.

In addition, the wizards are Microsoft VB-like in that they're one way and don't support the saving and reopening of half-completed forms. However, to overcome this you can simply rerun the wizard and create a temporary form and then copy/paste the fields to the existing form in the IDE.

Another minor issue we encountered was that the XML/Java templates must be edited by hand since there are no wizards as yet to facilitate these edits.

## Summary

In our business, we frequently outgrow "quick success" tools – often the fun of the demo turns to real frustration as you realize that simplifications that enabled quick success constrain a full solution. However, XTT is a small, focused, well-engineered set of Java objects/tools that enables Java developers to focus on solving business problems rather than complicated infrastructure coding. XTT is reliable and feature rich – but most of all it offers a transparent solution that does not limit your ability to integrate with other Java APIs (i.e., Web services).

XTT turns your Java IDE into a RAD powerhouse so you can deliver what clients want, when they want it – in the manner you know best. If you need to develop transaction-oriented rich client interfaces, especially for distributed applications, you definitely need to take a look at XTT. ☺

## Snapshot

**Target Audience:** Java, Visual Basic, and PowerBuilder developers

**Level:** Beginner to advanced

### Pros:

- Easy to use, feature-rich product
- Wizards and template-based development
- Strong event processing support
- Easy integration with existing Java APIs
- Good quality product

### Cons:

- No support for Eclipse
- Wizards are one way
- XML/Java templates must be hand edited



Onno Kluyt

# From Within the Java Community Process Program

From elections to final ballots

**W**elcome to the December edition of the JCP column! Every month you can read about the Java Community Process: newly submitted JSRs, new draft specs, Java APIs that were finalized, and other news from the JCP. This month I'll cover how the elections are progressing, and a new JSR from Sun, two JSRs in Community Review, two in Public Review, and three on the Final Approval ballot.

## The Executive Committee Elections, the Open Election Stage

This time there are two seats on each EC open for election. When you look at the candidates on the election Web site at <http://jcpelection2003.org>, you'll see that things look a little different compared to previous years. For the SE/EE EC there are nine candidates running for the two available seats. While that number is not remarkable, the fact that eight of the nine are individual members is remarkable. Doug Lea, the sitting EC member, is running as well as Larry Cable, Gustavo Alvarado, Dong Chen, Dave Marquard, Selvan Rajan, Huy Nguyen, and Richard Monson-Haefel. The one corporate member is Sonic Software. For the ME EC, five candidates are running for the two available seats. Selvan Rajan is also running for a seat on this EC and he is joined by Ericsson, Esmertec, PalmSource, and Intel. The elections closed on November 17 and the results were announced on the 18th (after this magazine had gone to press).

In last month's ratification vote all candidates were approved by the JCP membership. I congratulate IBM, HP, Fujitsu, Oracle, Motorola, Siemens, and Matsushita on their reelection and welcome Vodafone to the ME EC.

## A New JSR for the J2EE Environment

JSR 233, J2EE Mobile Device Management and Monitoring Specification, is currently in the JSR Review stage. In short, this JSR proposes

a facility on top of the J2EE platform to remotely manage and monitor the software on a mobile device. Two examples of areas where there is a need for this functionality are enterprises that enable their workforce to interact with enterprise information from mobile devices, and carriers providing customer support for software services on mobile devices. The JSR will have a couple of relationships with existing JSRs. JSR 232 defines a management environment on the device but not the server infrastructure. JSR 160 does define a management and monitoring structure but it depends on JSR 3, both of which are optional for J2EE environments; however, mobile devices generally do not support either one. Finally, JSR 233 aims to reuse APIs from the J2EE client provisioning JSR (#124), which completed last month.

## Two JSRs in Community Review

JSR 170, which is led by Day Software based in Switzerland, is working to define a specification for access to content repositories. Its expert group is one of the larger ones in the JCP program with 30 JCP members participating. The JSR provides an implementation-independent API to access common services that content repositories provide, such as version control, fine-grained access control, full-text search, and content categorization.

Submitted by Nokia in April of this year, JSR 212 is now in Community Review. It defines the Java API for SAMS Messaging and is destined to work in both J2SE and J2EE environments. The specification supports both short (SMS) and multimedia (MMS) messages.

## Two JSRs in Public Review

The first JSR we'll cover in this section is JSR 147, Workspace Versioning and Configuration Management, led by IBM. While this JSR was submitted a while back, in the last months it has gained

remarkable momentum by going through Community Review in late October and early November and subsequently entering Public Review. Collaborative authoring tools for software development, document management, and so on have a need for services to manage versioning and configuration of files and other resources. This JSR provides a client-side API and builds on several protocols (DeltaV, WebDAV, HTTP) to deliver this functionality.

The second JSR I want to cover is the J2ME-based Security and Trust Services JSR, or JSR 177, of which Sun is the Spec Lead. The collection of APIs defined here is aimed toward CLDC- as well as CDC-enabled devices.

## Three JSRs Facing the Final Ballot

In the "almost final" category we have three efforts this month. Two are technology related and the third is process oriented. First off is the PDA Profile JSR led jointly by PalmSource and IBM. JSR 75 defines Java APIs for several features that PDA devices typically have, such as personal information management and file system access. The J2ME CLDC platform is the base requirement for this profile. Next is JSR 169, the JDBC optional package for the CDC+Foundation Profile. The JSR defines equivalent functionality to the java.sql package while taking into account the constraints of CDC-based devices. And last, the Program Office's own JSR, number 215, which specifies what JCP 2.6 will look like. After the JSR goes final, the Program Office needs to implement a couple of changes to the Web application that runs JCP.org in order to roll out the new version of the JCP, which we hope to do by late January, early February.

That's it for this month. I'm very interested in your feedback. Please e-mail me with your comments, questions, and suggestions. ☺

Onno Kluyt is the director of the JCP Program Management Office, Sun Microsystems.

[onno@jcp.org](mailto:onno@jcp.org)



Advertiser	URL	Phone	Page
Altova	www.altova.com/upgrade		19
BEA Systems	dev2dev.bea.com/cerebrum		2-3
Borland Software Corporation	go.borland.com/j6		11
Canoo Engineering AG	www.canoo.com/ulc/		33
Computer Associates	ca.com/lifecycle		4
Edge East 2004	www.sys-con.com	201-802-3069	50-57
Exadel	www.exadel.com		39
IBM Rational	ibm.com/rational/seamless		17
ILOG	jviews-info-kit.ilog.com	1-800-for-ILOG	25
InetSoft Technology Corp.	www.inetsoft.com	888-216-2353	27
Infragistics, Inc.	www.infragistics.com	800-231-8588	14-15
iSavix	http://isavix.net	703-689-3190	37
LinuxWorld Conference & Expo	www.linuxworld.com		42-43
Mercury Interactive	www.mercuryinteractive.com/optimizej2ee		13
Northwoods Software Corp.	www.nwoods.com/go	800-434-9820	32
Oak Grove Systems	www.oakgrovesystems.com/jdj	818-440-1234	23
Parasoft Corporation	www.parasoft.com/jdj12	888-305-0041	21
Quest Software, Inc.	http://java.quest.com/jclass/jdj		9
Quest Software, Inc.	http://java.quest.com/performasure/jdj		31
Quest Software, Inc.	http://java.quest.com/jprobe/jdj		Cover IV
ReportingEngines	www.reportingengines.com/info/trial7.jsp	888-884-8665	22
SAP	sap.com/netweaver	800-880-1727	7
Software FX	www.softwarefx.com	800-392-4278	Cover III
WebAppCabaret	www.webappcabaret.com/jdj.jsp		29

**General Conditions:** The Publisher reserves the right to refuse any advertising not meeting the standards that are set to protect the high editorial quality of **Java Developer's Journal**. All advertising is subject to approval by the Publisher. The Publisher assumes no liability for any costs or damages incurred if for any reason the Publisher fails to publish an advertisement. In no event shall the Publisher be liable for any costs or damages in excess of the cost of the advertisement as a result of a mistake in the advertisement or for any other reason. The Advertiser is fully responsible for all financial liability and terms of the contract executed by the agents or agencies who are acting on behalf of the Advertiser. Conditions set in this document (except the rates) are subject to change by the Publisher without notice. No conditions other than those set forth in this "General Conditions Document" shall be binding upon the Publisher. Advertisers (and their agencies) are fully responsible for the content of their advertisements printed in **Java Developer's Journal**. Advertisements are to be printed at the discretion of the Publisher. This discretion includes the positioning of the advertisement, except for "preferred positions" described in the rate table. Cancellations and changes to advertisements must be made in writing before the closing date. "Publisher" in this "General Conditions Document" refers to SYS-CON Publications, Inc.

This index is provided as an additional service to our readers. The publisher does not assume any liability for errors or omissions.

# Next Month

Don't miss the new release of *JDJ*!

## JDJ v3.0

To reflect the changes with Java, JDJ v 3.0 will include several new sections:

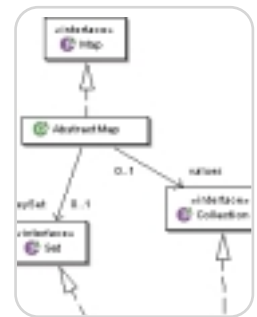
- Java Enterprise (including the mobile enterprise)
- Desktop Java
- Core Java
- Java Gaming

...as well as Java success stories, interviews with Java luminaries, and more!

## The Future of Java

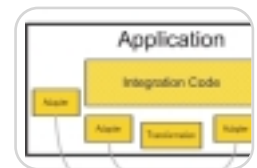
### *An Interview with Alfred Chuang*

Alfred Chuang, CEO of BEA, talks with **JDJ** about Java and the industry, what sets BEA apart from others in the field, what the future holds for Java, and more.



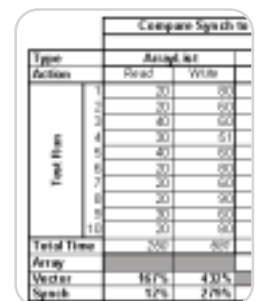
## Java Collections

The spotlight is on J2EE and as a result many programmers are ignoring the foundation of the JDK. The collections API is one of the most useful parts of the JDK. This article will review the collections architecture and point out some of the useful features of the collections API (sorting and searching).



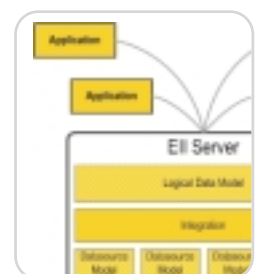
## Rich Desktop Clients

This article will focus on the knowledge, techniques, and patterns that are needed to design and implement a polished desktop application in Java and Swing (and/or SWT), and deploy it to end users. Creating a clean, solid application framework, without creating code that isn't needed, is one of the most important parts of a rich client, which is why this article isn't just about Swing, it's about the code that sits above Swing tying everything together.



## Which EII Solution Is Right for You?

Enterprise Information Integration (EII) represents a new category of software that enables disparate data silos to be integrated into a single virtual database for applications. This approach gives developers a powerful tool for simplifying data integration and building flexible applications. If you haven't heard of EII yet, you will soon as the industry rallies around this concept and more EII projects reach deployment.



# Industry News

## HelpBreeze JavaHelp 3.0 Offers a Solution for Creating Online Help

(Sunnyvale, CA) – Solutionsoft has announced the release of HelpBreeze JavaHelp Edition 3.0, a major upgrade to its authoring tool for Sun's JavaHelp platform. HelpBreeze 3.0 offers full support for the recently released JavaHelp 2.0 software, and also includes dozens of new features and improvements throughout the product.

HelpBreeze provides a complete visual environment for creating and maintaining JavaHelp systems. The software includes the tools necessary for creating, editing, and testing the online help system. Help authors are completely insulated from the details of the HTML and XML files required to implement the help system.

[www.solutionsoft.com/javahelp.htm](http://www.solutionsoft.com/javahelp.htm)

## Sun to Open a Wal-Mart Compliant RFID Test Center

(Bentonville, AK) – Sun Microsystems has announced that it will open an RFID test center where Wal-Mart suppliers can test their RFID solutions to guarantee compliance with the Wal-Mart standard.

Wal-Mart will use the Electronic Product Code (EPC) compliant RFID technology for identifying, tracking, and tracing deliveries and inventory. The Sun RFID test center will be located in Dallas, Texas, and will be open to Wal-Mart's suppliers. It will be powered by the Solaris Operating System using Sun Java Enterprise Software and the Sun standards-based implementation of Savant with additional value around

self-healing and provisioning of EPC readers.

[www.sun.com](http://www.sun.com)

## Sun Teams with AMD

(Las Vegas) – Sun Microsystems has announced that it has teamed with AMD to deliver AMD Opteron processor-based x86 systems to customers in the next calendar year. Sun will add the processor-based x86 to its current line of entry-level systems, giving customers greater platform choice with price-performance. Further, Sun and AMD will work together to optimize the Solaris Operating System for AMD Opteron 64-bit processors, which will enable customers to take advantage of the Unix platform and binary compatibility between 32- and 64-bit environments.

[www.sun.com](http://www.sun.com)

[www.amd.com](http://www.amd.com)

## esmertec on Acquisition Trail with New Funds

(London) – Java applications developer esmertec has raised 23 million Euros (about \$20 million) in third-round financing to expand its JVM offerings in the mobile phone business and increase its presence in the embedded Java sector.

The additional funds will also be used to expand esmertec's global reach, with much more focus on Asia and Japan where the majority of Java-based development for mobile phones, home multimedia devices, and "smart machines" is taking place.

[www.esmertec.com](http://www.esmertec.com)

## Eclipse Poised for Growth

Eclipse, the open-source development platform, marks its second anniversary this month, and IBM, its main sponsor, is putting more money, manpower, and research into making the Java-based integrated development environment more attractive to developers and more of a competitor to Microsoft Corp.'s Visual Studio .NET. IBM and its Eclipse partners are leveraging Eclipse as the foundation for new application development technology on the Java side.

## Plumtree, Documentum, BEA, and Sun Join Forces

(San Francisco, Pleasanton, San Jose, and Santa Clara, CA) – Enterprise software vendors Plumtree Software, Documentum, BEA Systems, Inc., and Sun Microsystems, Inc., have announced an open source site for organizations to share portlets developed according to the new JCP JSR 168 and WSRP OASIS standards. Available at <http://sourceforge.net/projects/portlet-opensource/>, the site is hosted by SourceForge, an independent organization that hosts a variety of Java technology and Linux open-source initiatives. Plumtree, Documentum, BEA, and Sun will provide an initial library of standards-based portlets as well as ongoing feedback, suggestions, and best practices for successful JSR 168 and WSRP portlet development. The site is open to all organizations. The open source site, known as the Portlet Open-Source Trading site, or POST, will help companies learn from their industry peers and share best practices for developing standards-based portlets.

## SeeBeyond Releases ICAN Suite 5.0

(Monrovia, CA) – SeeBeyond has announced the latest release of its integration broker platform, Integrated Composite Application Network (ICAN) 5.0 suite.

ICAN 5.0 is a fully integrated, J2EE-certified integration suite that provides open standards-based technologies to integrate people, processes, and systems across the extended enterprise.

[www.seebeyond.com](http://www.seebeyond.com)

## BEA Offers Preview Release of Streaming API for Java

(San Jose, CA) – BEA Systems, Inc., an application infrastructure company, has announced the public availability of a preview release of the Streaming API for Java (StAX), a new Java API designed to improve developer productivity and performance by making it easier to incorporate Extensible Markup Language (XML) into Java.

StAX is designed to overcome many of the disadvantages of former methods by providing the efficiency of streaming APIs and the control of tree-based APIs. This new method represents a next generation of APIs – pull parsing. Unlike SAX and DOM, StAX is bidirectional, and can allow programs to both read existing XML documents and create new ones.

<http://dev2dev.bea.com/technologies/stax/index.jsp>





# SYS-CON MEDIA

**304,187 of the World's Foremost IT Professionals**

**DIRECT MAIL, EMAIL OR MULTI-CHANNEL**

**Target CTOs, CIOs and CXO-level IT professionals and developers who subscribe to SYS-CON Media's industry leading publications**

## Java Developer's Journal...

The leading publication aimed specifically at corporate and independent Java development professionals



## PowerBuilder Developer's Journal...

The only PowerBuilder resource for corporate and independent enterprise client/server and web developers



## ColdFusion Developer's Journal...

The only publication dedicated to ColdFusion web development



## LinuxWorld Magazine...

The premier monthly resource of Linux news for executives with key buying influences



## WebLogic Developer's Journal...

The official magazine for BEA WebLogic application server software developers, IT management & users

## Web Services Journal...

The only Web Services magazine for CIOs, tech, marketing & product managers, VAs/ISVs, enterprise app architects & developers



## .NET Developer's Journal...

The must read technology publication for Windows developers & CXO management professionals

## XML Journal...

The world's #1 Leading XML resource for CEOs, CTOs, technology solution architects, product managers, programmers and developers



## WebSphere Developer's Journal...

The premier publication for those who design, build, customize, deploy, or administer IBM's WebSphere suite of Web Services software



## Wireless Business & Technology...

The wireless magazine for key corporate & engineering managers, and other executives who purchase communications products/services

Recommended for a variety of offers including Java, Internet, enterprise computing, e-business applications, training, software, hardware, data back up and storage, business applications, subscriptions, financial services, high ticket gifts and much more

**NOW AVAILABLE!**

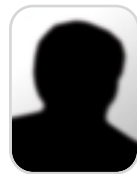
The SYS-CON Media Database  
304,187 postal addresses

For email information:  
contact Frank at 845-731-3832  
frank.xipalla@epostdirect.com  
epostdirect.com 800-409-4443 fax 845-620-9035

For postal information:  
contact Kevin at 845-731-2684  
kevin.callapy@edithroman.com  
edithroman.com 800-223-2194 fax 845-620-9035







Henry Roswell

# Dude— Where's My Language?

**A**fter Sun reported significant losses, a \$1 billion tax charge, and layoffs, many people began to predict a slow death for the company. Several analysts were already writing obituaries for Sun's executive team, while newsgroup pundits were swapping ideas about what they'd do in Scott McNealy's shoes. What Scott actually did was play the "Do or Die" card.

## Renationalize the Java Trademark

Java means different things to different people. To most of the readers of *JDJ* it is the language we all love and cherish. Sun vigorously defended Java's ability to create cross-platform portable code by hauling Microsoft into court for trying to contaminate the language. More recently, they turned their guns on IBM for creating SWT. Protection of the Java trademark was always a concern for its business partners who had to negotiate Sun's restrictive licensing terms ([www.sun.com/policies/trade\\_marks](http://www.sun.com/policies/trade_marks)). Famously, JBoss was forced to withdraw its J2EE compatibility claims because it hadn't purchased and run the TCK from Sun. By monopolizing the trademark, however, Sun can now do whatever it wants with its crown jewel.

## Rename Failing Products

Java's trademark policy is a tricky beast for anyone who has to name a

product using it. Few have dared to call their product the "Java XXX", and instead create names such as "XXX for Java technology" or more imaginatively "JXxx". Sun, however, has no such restrictions, and promptly renamed Mad Hatter, Java Desktop; Orion, Java Enterprise Server; and Project Rave, Java Studio. At first there was a flurry of outrage at this in the forums, with most complaints from people who felt the Java brand had been diluted. However, apart from the loss of some of the core principles, such as Write Once, Run Anywhere, most everyone recognized that it made good business sense for Sun to use a successful trademark to promote its product line.

## Use Java to Sell Sun's Products

Given the success of the Java brand, why not use this to leverage sales in the renamed Sun product suite? Other companies have profited from Java in the tools and app server space, so surely Sun can reclaim market share by selling its product stack around the Java adjective? If you visit [www.java.com](http://www.java.com), you'll find it contains links to Java Studio, Java Enterprise Server, and Java Desktop; not that many links to WebLogic, WebSphere, Borland, or other JCP participants' products. The java.com site is the one people visit when they click the "Java get it now button" launched at JavaOne to help

get Java installed on Microsoft desktops, and endorsed by companies such as Dell, HP, and Gateway. Not sure whether using java.com as a portal to Sun's commercial products is in the spirit of the Java trademark, but what can anyone do about it except look on in envy?

## Go for the Kill

The final move: erase the competition. The next step was to make Java System Application Server 8.0 free to download. Giving away the \$2,000 app server stole a lot of thunder from free products like JBoss and Geronimo. Free J2EE servers have dented but haven't really hurt IBM or BEA, both of whom have reinvented themselves; however, if the Java Web sites and Java technology supremos draw people with free copies of Sun's app server and other parts of the relaunched product line, perhaps this will change. I leave my closing words to Jonathan Schwartz of Sun quoted at [http://zdnet.com.com/2100-1104\\_2-5107201.html](http://zdnet.com.com/2100-1104_2-5107201.html): "We are embracing and monetizing Java in ways that we haven't historically." Interesting choice of words from the marketing VP of a company trying to challenge Microsoft. Perhaps the Redmond motto of "Embrace – engulf – extinguish" has become the new mantra in Santa Clara? ☺

Henry Roswell is a veteran consultant who would like to think he's seen it all, but is constantly amazed by new events everyday.

[henry@sys-con.com](mailto:henry@sys-con.com)

Java Dudes



# A Fresh Brew of Chart FX...



## With a Robust Blend of Form and Function.

10 years of creating quality developer charting solutions gives us an edge over others in the industry. So, although we're the new kid on this block, we've been in the neighborhood for a long time. The maturity of Chart FX and our support will prove it. Chart FX for Java is a 100% Java

component that uses JSP technology to produce charts in a variety of formats: PNG, JPEG, SVG and FLASH. Developed using JDK 1.4, it supports J2EE 1.3 and J2SE 1.4. Chart FX for Java is available as a Server-side Bean and an Enterprise Java Bean (EJB) that runs on most popular Java Application Servers, with features like borders, gradients, alpha blending, anti-aliasing and transparency.

Additionally, it will produce native Windows output in the form of a ActiveX or .NET component. Easy data population from JDBC, XML, Text Files, API and other popular data sources. Includes a Designer, which is a stand-alone tool to set the visual attributes of the chart, and a Resource Center containing the Programmers Guide, the Javadoc API and hundreds of samples. For more information or to download a trial version, visit [www.softwarefx.com](http://www.softwarefx.com).



  
**Chart FX**  
for JAVA

[www.softwarefx.com](http://www.softwarefx.com)

In the US: (800) 392-4278 • In the UK: +44 (0) 117 905 8733 • In Germany: 0800-24 27 839



As Java development evolves, so does JProbe®

## JProbe®

Find the cause of J2EE code performance, memory and threading problems faster than ever before with Quest JProbe. New investigative features for finding memory problems combined with dramatic performance improvements mean even quicker resolution of problems in your application, servlet, JSP and EJB code.

JProbe Suite

JProbe Profiler

JProbe Memory Debugger

JProbe Threadalyzer

JProbe Coverage

Part of the **Quest Performance Management Suite for the J2EE Platform**



For more info and a free eval, visit:

<http://java.quest.com/jprobe/jdj>