

Greater Oregon Software Symposium

Embassy Suites Hotel - Downtown Portland

April 20 - 22, 2007

<http://www.nofluffjuststuff.com/sh/2007-04-portland>

Fri, Apr. 20, 2007				
	Queen Marie	Chief Poker Jim	Marshall Joffre	Eric Hauser
12:00 - 1:00 PM	REGISTRATION			
1:00 - 1:15 PM	WELCOME			
1:15 - 2:45 PM	Ajax Design and Architecture Glenn Vanderburg	Drizzling with Groovy and Rules Venkat Subramaniam	What's New in Java 6 Jason Hunter	Java NIO Brian Pontarelli
2:45 - 3:15 PM	BREAK			
3:15 - 4:45 PM	JavaScript Exposed: There's a Real Programming Language in There! (Part 1) Glenn Vanderburg	Domain Driven Design Venkat Subramaniam	XQuery By Example: Building an Email Archive System Jason Hunter	The ACEGI Framework Brian Pontarelli
4:45 - 5:00 PM	BREAK			
5:00 - 6:30 PM	JavaScript Exposed: There's a Real Programming Language in There! (Part 2) Glenn Vanderburg	Annotation Hammer Venkat Subramaniam	Web Publishing 2.0 Jason Hunter	SOA Topologies Brian Pontarelli
6:30 - 7:15 PM	DINNER			
7:15 - 8:00 PM	Keynote: Jason Hunter			

Sat, Apr. 21, 2007				
	Queen Marie	Chief Poker Jim	Marshall Joffre	Eric Hauser
8:00 - 9:00 AM	BREAKFAST			
9:00 - 10:30 AM	Forgotten Web Algorithms Jason Hunter	Java Performance Myths Glenn Vanderburg	Groovy for Java Programmers Venkat Subramaniam	Writing Good APIs Brian Pontarelli
10:30 - 11:00 AM	BREAK			
11:00 - 12:30 PM	AOP and JMX: A match made in heaven Ben Hale	Everything Old Is New Again Glenn Vanderburg	Agile Web Development with Grails Venkat Subramaniam	Software Development Techniques Jared Richardson
12:30 - 1:30 PM	LUNCH			
1:30 - 3:00 PM	Pragmatic Unit Testing with TestNG and EasyMock Howard Lewis Ship	Spring 2.0: New and Noteworthy Ben Hale	Enterprise Ajax with Java Greg Murray	Build Teams, Not Products Jared Richardson
3:00 - 3:15 PM	BREAK			
3:15 - 4:45 PM	Spring and Hibernate in the Middle Tier Ben Hale	Making Architecture Work Through Agility Mark Richards	OSGi: A Well Kept Secret Venkat Subramaniam	Subversion: A Quick Start Guide Jared Richardson
4:45 - 5:30 PM	BIRDS OF A FEATHER SESSIONS			

Sun, Apr. 22, 2007				
	Queen Marie	Chief Poker Jim	Marshall Joffre	Eric Hauser
8:00 - 9:00 AM	BREAKFAST			
9:00 - 10:30 AM	Intro to Java Persistence API (JPA) Mark Richards	The Busy Java Developer's Guide to ClassLoaders Ted Neward	Project jMaki - Enabling Web 2.0 Application Developers Greg Murray	Shippers Unite! Jared Richardson
10:30 - 11:00 AM	BREAK			
11:00 - 12:30 PM	Advanced Java Persistence API (JPA) Mark Richards	The Busy Java Developer's Guide to Reflection Ted Neward	JRuby Venkat Subramaniam	Spring Web Flow Jumpstart Ben Hale
12:30 - 1:15 PM	LUNCH			
1:15 - 2:15 PM	EXPERT PANEL DISCUSSION			
2:15 - 3:45 PM	The Busy Java Developer's Guide to Debugging and Monitoring Ted Neward	EJB3 Core Specification (JSR-220) Mark Richards	Learning Tapestry 4 Howard Lewis Ship	Continuous Integration with Cruise Control Jared Richardson
3:45 - 4:00 PM	BREAK			
4:00 - 5:30 PM	The Busy Java Developer's Guide to Java Platform Security Ted Neward	Making The Right Persistence Framework Choice Mark Richards	Tapestry 5 Preview Howard Lewis Ship	Distributed Teams: Remote Agility Jared Richardson

Greater Oregon Software Symposium

Embassy Suites Hotel - Downtown Portland

April 20 - 22, 2007

<http://www.nofluffjuststuff.com/sh/2007-04-portland>

AOP and JMX: A match made in heaven by Ben Hale

You're winding down project and you get that dreaded email from your project manager, "How hard would it be to add some performance monitoring to the system?" Well, after this session, you'll be able to respond, "No problem at all!" It turns out that with a pinch of AOP and a dash JMX, you can introduce amazing management and monitoring capabilities without changing you're mainline code one bit.

Spring 2.0: New and Noteworthy by Ben Hale

Spring 2.0 has marked a major advance in the Spring Framework. While still maintaining backwards compatibility, this release adds quite a few new features. What are those features and how do they add value? Come by and see.

Spring and Hibernate in the Middle Tier by Ben Hale

To today's JEE developer, there are two indispensable tools for creating applications; Spring and Hibernate. Together these two frameworks comprise one of the most powerful and often used stacks in the industry. While it is possible to do amazing things it's not always obvious how best to use them to maximize value. This session aims to correct that.

Spring Web Flow Jumpstart by Ben Hale

Have you ever developed a web application with a long user action based on form input? Did you curse the Java community for their inability to address this very common application type? Well, attend this session about Spring Web Flow and you'll curse no more.

Java NIO by Brian Pontarelli

The Java NIO packages that were added in JDK 1.4 and these packages allow Java applications to perform true non-blocking IO operations. This presentation will cover the basics of the standard IO packages, which date back to the beginning of Java, and some of the shortcomings they have. This will be followed by coverage of the newer NIO packages and how they address these issues.

The ACEGI Framework by Brian Pontarelli

The ACEGI framework is a comprehensive security library built on top of the popular Spring Framework. This talk will cover the basics of using the ACEGI framework within a Java web application.

SOA Topologies by Brian Pontarelli

This talk will cover many of the different types of SOA topologies from EJBs and WebServices all the way to message queues and tuple spaces. SOA has many different meanings but it never dictates a single implementation and this talk covers many of the most common implementations of a service oriented architecture.

Writing Good APIs by Brian Pontarelli

Writing APIs is fairly easy but writing an API that is usable and lives longer than a few days is hard. This talk discusses methodologies, tips and tricks for writing good APIs.

Ajax Design and Architecture by Glenn Vanderburg

Ajax applications have unique design and architectural challenges and opportunities. This presentation will show you how to take advantage of the Ajax's strengths, and work around its quirks.

JavaScript Exposed: There's a Real Programming Language in There! (Part 1) by Glenn Vanderburg

With the sudden importance of Ajax, it's time to take JavaScript seriously. That means learning it the right way: looking at the fundamentals of the language and surveying its strengths and weaknesses, instead of just copying other people's poorly written examples.

JavaScript Exposed: There's a Real Programming Language in There! (Part 2) by Glenn Vanderburg

Building on part 1, this talk dives deep into JavaScript's object model. We'll see how it differs from more mainstream object-oriented languages, and why. We'll explore how to hide some of those differences, as well as the reasons you might not want to. Additionally, we'll cover useful tools for JavaScript testing, debugging, and profiling.

Java Performance Myths by Glenn Vanderburg

Performance myths about the Java platform abound, from the general "Java is slow", to the more specific "reflection is slow", "allocation is slow", "synchronization is slow", "garbage collection is slow", etc. Many of these myths have their root in fact (in JDK 1.0, everything was slow); today, not only are many of these statements not true, but Java performance has surpassed that of C in many areas, such as memory management.

Everything Old Is New Again by Glenn Vanderburg

The early years of computers -- the '50s and '60s -- were characterized by furious exploration of a huge variety of different ideas. Since then many of the hot topics of those days have moved to the fringe, largely ignored by the mainstream of software development. But some of them are being rediscovered, and a lot of what we think of as "new developments" are really just some old ideas returning to center stage.

Enterprise Ajax with Java by Greg Murray

Ajax and Java are the perfect combination for creating Web 2.0 applications. This session will many of the key issues that Java developers may experience when developing Ajax clients and services. Topics that we will cover various topics including security, performance, usability, debugging, and working with JavaScript Object Notation (JSON). This session will conclude with a discussion of both Java-centric and JavaScript-centric approaches (i.e, light client-side logic versus heavy client-side logic) along with the tradeoffs to consider with both of these approaches.

Project jMaki - Enabling Web 2.0 Application Developers by Greg Murray

Creating a breathtaking user interface requires that you and your team are skilled in JavaScript, Cascading Style Sheets (CSS), and Ajax interactions. This session will discuss how you can use Project jMaki (or just the ideas behind it) to create JavaScript-centric user interfaces that are developer and designer friendly. Project jMaki is an open-source client/server framework hosted on java.net that is focused on creating JavaScript-centric web applications. jMaki promotes a clean separation of JavaScript, CSS, and template code following the web design approach widely known as "unobtrusive JavaScript". This session will include a technical discussion of the architecture and features of jMaki including JavaScript toolkit interoperability, widget design, using layouts, wiring JavaScript functionality together, and service/data integration.

Pragmatic Unit Testing with TestNG and EasyMock by Howard Lewis Ship

You've heard about unit testing but were daunted when it came time to put the pedal to the metal. That's because JUnit is just one tool and there's others you need to learn about, including the wonderful and wierd EasyMock and the easy and powerful TestNG.

Learning Tapestry 4 by Howard Lewis Ship

An introduction to the Apache Tapestry web application framework, which will explain the concepts and features of the framework with some simple applications. We'll discuss Tapestry forms, request cycle, component object model. The use of several important components (including the powerful Table data grid) will be featured.

Tapestry 5 Preview by Howard Lewis Ship

Tapestry 5 is a complete rewrite of Tapestry from the ground up. It takes everything good about Tapestry and cranks the volume up to eleven, while removing the frustrating parts of using Tapestry. This session takes the wraps off this new and innovative technology, showing off important new features such as live class reloading (the ability to change your Java classes and continue using the application without interruption or redeployment), the simplified coding model, and the total lack of XML. This session is of interest to those already using Tapestry 4, and those new to Tapestry and ready to jump on the bandwagon.

Software Development Techniques by Jared Richardson

Throughout our software careers we learn habits from our coworkers, from books we've read, and occasionally, from conferences we attend. Much of our competence comes from the tips and tricks we pick up as we go.

Build Teams, Not Products by Jared Richardson

A great team builds great software, but how do you build a great team?

Subversion: A Quick Start Guide by Jared Richardson

Subversion is a free source code management system that's very powerful.

Shippers Unite! by Jared Richardson

An overview of the Agile software approach from the book Ship It! A Practical Guide to Successful Software Projects.

Continuous Integration with Cruise Control by Jared Richardson

Continuous Integration is increasingly recognized as a vital practice in an Agile software shop. Traditionally it's been difficult to set up and administer. Today, that's no longer the case.

Distributed Teams: Remote Agility by Jared Richardson

How do you keep a team scattered across time zones in sync?

What's New in Java 6 by Jason Hunter

The Java 6 (Mustang) release should make your life easier, for a change. It doesn't alter the core language like Java 5 did. It doesn't pack in so many sub-JSRs that you'll be overwhelmed by the amount you have to learn. Instead Java 6 adds several handy things that honestly should have been added before. Among the improvements we'll cover in this fast-paced class: * A new Console class * A real Compiler API * A GIF writer * Pluggable Locale data * Access to disk partition size data * Array reallocation * Low-level floating point functions * Reflective access to parameter names * Access to network interface details * Pluggable annotation processing * Improved class file format * Streaming XML with StAX * A new Scripting interface

XQuery By Example: Building an Email Archive System by Jason Hunter

The classic searchable email archive system is cluged together -- a frankenstein monster combining a relational database with a search engine, with Java just barely able to keep the two together. In this talk we'll demonstrate how email is more content than data, how it's better encoded in XML rather than relational tables, and how Java can convert emails to XML and drive an XQuery backend to produce a simpler and more scalable email archive system.

Web Publishing 2.0 by Jason Hunter

If we're moving toward Web 2.0, what does that mean for online publishing? In this talk I'll answer that question. Based on my experience as Principal Technologist at Mark Logic working with dozens of the largest online publishers, I'll present a vision for how the Web 2.0 concepts like personalization, collective intelligence, the long tail, and the importance of "owning the data" can and should reshape the face of online publishing -- and how XML, XQuery, and XML-aware text search act as the key enablers. I'll also introduce new Web Publishing 2.0 concepts like "Sweat the content" and "Give answers not links".

Web Publishing 2.0 by Jason Hunter

If we're moving toward Web 2.0, what does that mean for online publishing? In this talk I'll answer that question. Based on my experience as Principal Technologist at Mark Logic working with dozens of the largest online publishers, I'll present a vision for how the Web 2.0 concepts like personalization, collective intelligence, the long tail, and the importance of "owning the data" can and should reshape the face of online publishing -- and how XML, XQuery, and XML-aware text search act as the key enablers. I'll also introduce new Web Publishing 2.0 concepts like "Sweat the content" and "Give answers not links".

Forgotten Web Algorithms by Jason Hunter

In this talk I'll explain -- without any needless math or boring proofs -- several fun algorithms of interest to back-end web programmers. Each algorithm was selected because it's really practical, really interesting, or both. The algorithms aren't always the same but can include: public key cryptography, credit card checksum validation, TCP Slow Start, two's complement, priority queues, the XOR swap, and the Google MapReduce function for massively distributed calculation.

Making Architecture Work Through Agility by Mark Richards

As companies continue to change the way they do business, so must the IT systems that support the business. Changes due to regulatory requirements, competitive advantage, mergers, acquisitions, and industry trends require flexible IT systems to meet the demands of the business. Software Architects must therefore make their architectures more agile to meet the flexible demands of today's business. Through real-world examples and scenarios we will explore some of the challenges facing Software Architecture and discuss several concrete techniques for applying agility to both the architecture process and the technical architecture itself. We will also look at various architecture refactoring techniques, and discuss the pros and cons of each. By attending this session you will learn how to apply various agile techniques to improve your architectures and overcome some of the challenges facing software architecture in today's ever-changing market.

Intro to Java Persistence API (JPA) by Mark Richards

In addition to providing a simplified API, the new EJB3 specification (JSR-220) defines a standard ORM Java Persistence API (JPA) that is rapidly gaining in popularity. As you will see in this session, JPA bears a striking resemblance to popular ORM solutions like Hibernate and Toplink. In this session we will explore in detail the new Java Persistence API offered by JSR-220. We will start by discussing the overall design and architecture of the JPA and how the major components within JPA interact. We will then look at defining mapping objects (entities) and how to use the EntityManager to manage these entities. Through interactive coding examples we will investigate the pros and cons of detached entities and merging, how to map and use entity relationships (1-1, 1-N, N-1, and N-N), discuss Lazy Loading, and finally see how to use XML mappings rather than annotations. More advanced features of JPA will be covered in a separate session.

Advanced Java Persistence API (JPA) by Mark Richards

This session picks up where the Intro to JPA session left off and covers some of the more advanced topics in the Java Persistence API. Some of the topics covered in this session include switching persistence providers, versioning, compound keys, entity inheritance, handling both simple and complex stored procedures, and finally JPA Query Language (JPQL). Some knowledge of JPA is recommended for this session as I will not be covering the basics of JPA (that is covered in a separate Intro to JPA session). Through a combination of slides and interactive coding I will demonstrate these advanced topics using both Hibernate and Toplink JPA.

EJB3 Core Specification (JSR-220) by Mark Richards

The new EJB 3 specification (JSR-220) offers some great improvements over the prior EJB specs in terms of development simplicity and new features. In this session we will discuss why EJB is still important, and explore in detail some of the new features of the core EJB 3 specification. Included in this session will be defining and accessing session beans, JTA transaction management, declarative security, and interceptors. For those of you who still like to code in XML, I will also discuss and show how we can use XML rather than annotations within EJB3. During the session I will demonstrate the new features of EJB 3 through interactive coding examples. Note: this session does not cover the new Java Persistence API (JPA) - that topic is covered in separate JPA sessions.

Making The Right Persistence Framework Choice by Mark Richards

Java Persistence has come a long way in the past 4 years. We have many viable options available now, including JPA, Hibernate, iBATIS, Toplink, and OpenJPA. With so many options available now it is difficult to know when to use which framework. In this session we will focus on native Hibernate, JPA, and the iBATIS framework, and discuss the main strengths and weaknesses of each approach and what the decision criteria is for using each of these frameworks. Knowing that it is not a one-size-fits-all situation when it comes to Java Persistence, through interactive coding we will take a look at how to use iBATIS together with native Hibernate or JPA and when this makes sense. By attending this session you will gain the knowledge necessary to make informed decisions about which Java Persistence Framework to choose for your current or next Java-based application.

The Busy Java Developer's Guide to ClassLoaders by Ted Neward

If you've ever gotten a ClassCastException and just knew the runtime was wrong about it, or found yourself copying .jar files all over your production server just to get your code to run, then you probably find the Java ClassLoader mechanism to be deep, dark, mysterious, and incomprehensible. Take a deep breath, and relax--ClassLoaders aren't as bad as they seem at first, once you understand a few basic rules regarding their operation, and have a bit more tools in your belt to diagnose ClassLoader problems. And once you've got that, and hear about ClassLoaders' ability to run multiple versions of the same code at the same time, and to provide isolation barriers inside your application, or even compile code on the fly from source form, you might just find that you like ClassLoaders after all... maybe.

The Busy Java Developer's Guide to Reflection by Ted Neward

If you've never used Reflection (java.lang.reflect), you don't know what you're missing. In this presentation, we'll take a code-first, soup-to-nuts look at the Java Reflection APIs, from how to examine the class metadata that Reflection provides, to using annotations to enhance that metadata with your own information, even through the use of Java Dynamic Proxies to create flexible object "interceptors" that can layer services in front of ordinary method calls with nothing more complicated and an interface and a factory.

The Busy Java Developer's Guide to Debugging and Monitoring by Ted Neward

Bugs? We all know your code has no bugs, but someday, you're going to find yourself tracking down a bug in somebody else's code, and that's when it's going to be helpful to make use of the wealth of tools that the Java Standard Platform makes available to you--tools that your IDE may not know exist, tools that you can make use of even within a production environment.

The Busy Java Developer's Guide to Java Platform Security by Ted Neward

Permissions, policy, SecurityExceptions, oh my! The Java platform is a rich and powerful platform, complete with a rich and powerful security mechanism, but sometimes understanding it and how it works can be daunting and intimidating, and leave developers with the basic impression that it's mysterious and dark and incomprehensible. Nothing could be further from the truth, and in this presentation, we'll take a pragmatic, code-first look at the Java security platform, including Permissions, the SecurityManager and its successor, AccessController, the Policy class and policy file syntax, JAAS, and more.

Drizzling with Groovy and Rules by Venkat Subramaniam

Rule based programming allows us to develop applications using declarative rules. These can simplify development in applications where such rules based knowledge is used for decision making.

Domain Driven Design by Venkat Subramaniam

Domain Driven Design (DDD) is an approach that places emphasis on the domain model and carrying it into implementation. DDD is mostly repackaging of fundamental OO Design. It brings new emphasis to what we should be already doing, but often find it hard and confusing given the realities and complexities of our real world. In this presentation we will take a close look at what DDD is and how to use it for agile development. We will discuss several design options, and also look at some examples of good modeling and layering.

Annotation Hammer by Venkat Subramaniam

Annotation is an interesting feature in Java. However, like any features, there are good uses and bad uses. When should you use Annotation? This presentation will answer that question for you.

Groovy for Java Programmers by Venkat Subramaniam

Object-oriented scripting languages, or agile dynamic languages, as some like to call those, are gaining programmers' attention. Groovy bring this excitement to the Java platform with its ability to generate byte code. You can use Groovy instead of Java for some parts of your application. By learning it, you can switch between the languages where you consider fit.

Agile Web Development with Grails by Venkat Subramaniam

Agile development is all about developing code and seeking feedback from your users to make sure you're developing what's relevant. When they suggest changes, those must be affordable and reliable. Grails, along with its facility to develop test driven, is a killer combination for rapidly developing web applications. In this ZePo (Zero PowerPoint) presentation, we will take a test driven approach to developing a small but fully functional web application in Grails. We will cover the fundamental features of Grails along with utilizing other capabilities like Ajax. At the end of this presentation, you not only be confident, but eager to roll your own web application using Grails.

OSGi: A Well Kept Secret by Venkat Subramaniam

In this presentation we will introduce OSGi and discuss how it can help modularize and version your enterprise Java applications.

JRuby by Venkat Subramaniam

Object-oriented scripting languages, or agile dynamic languages, as some like to call those, are gaining programmers' attention. Several dynamic languages are on the JVM. Groovy and JRuby are two languages that are drawing developers' interest. Sun has shown support for these two, and especially JRuby by hiring the core developers.