

Western Canada Java Software Symposium

Radisson Hotel Calgary Airport

September 26 - 28, 2008

<http://www.nofluffjuststuff.com/conference/calgary/2008/09/index.html>
(event schedule as of September 27, 2008)

Fri, Sep. 26, 2008				
	Rocky Mountain A	Rocky Mountain B	Selkirk A	Selkirk B
12:00 - 1:00 PM	REGISTRATION			
1:00 - 1:15 PM	WELCOME			
1:15 - 2:45 PM	Groovy, the Blue Pill: Writing Next Generation Java Code in Groovy Scott Davis	Enterprise Messaging Using JMS (Part 1) Mark Richards	The Busy Java Developer's Guide to ClassLoaders Ted Neward	Towards an Evolutionary Design Venkat Subramaniam
2:45 - 3:15 PM	BREAK			
3:15 - 4:45 PM	Groovy, The Red Pill: Metaprogramming, the Groovy Way to Blow a Buttoned-Down Java Developer's Mind Scott Davis	Enterprise Messaging With JMS (Part 2) Mark Richards	The Busy Java Developer's Guide to Concurrency (Part 1: Threads) Ted Neward	Tools to facilitate Agile Development Venkat Subramaniam
4:45 - 5:00 PM	BREAK			
5:00 - 6:30 PM	Rapid Web Development with Grails and Ajax Scott Davis	SOA Unplugged Mark Richards	The Busy Java Developer's Guide to Concurrency (Part 2: Concurrency) Ted Neward	Acceptance Testing Application Behavior Venkat Subramaniam
6:30 - 7:15 PM	DINNER			
7:15 - 8:00 PM	Keynote: by Venkat Subramaniam			

Sat, Sep. 27, 2008				
	Rocky Mountain A	Rocky Mountain B	Selkirk A	Selkirk B
8:00 - 9:00 AM	BREAKFAST			
9:00 - 10:30 AM	EJB3 Core Specification (JSR-220) Mark Richards	Architecture and Scaling Ken Sipe	The Busy Java Developer's Guide to Hacking (on) the JDK Ted Neward	Practices of an Agile Developer Venkat Subramaniam
10:30 - 11:00 AM	BREAK			
11:00 - 12:30 PM	Java Persistence: Approaching the Silver Bullet Mark Richards	7 Habits of Highly Effective Developers Ken Sipe	The Busy Developer's Guide to Scala Ted Neward	BDD in Java and Groovy Venkat Subramaniam
12:30 - 1:30 PM	LUNCH			
1:30 - 3:00 PM	Professional Java UI development with the Eclipse RPC Brian Sam-Bodden	Spring+JPA+Hibernate: Standards Meeting Productivity for Java Persistence Ken Sipe	JavaServer Faces: A Whirlwind Tour David Geary	Estimating vs. Guessing - How Agile Teams Estimate Their Work David Bock
3:00 - 3:15 PM	BREAK			
3:15 - 4:45 PM	Boosting Programmer productivity with Mylyn Brian Sam-Bodden	Spring 2.5 - Spring without XML Ken Sipe	Facelets David Geary	The Agile Product Owner David Bock
4:45 - 5:30 PM	BIRDS OF A FEATHER SESSIONS			

Sun, Sep. 28, 2008				
	Rocky Mountain A	Rocky Mountain B	Selkirk A	Selkirk B
8:00 - 9:00 AM	BREAKFAST			
9:00 - 10:30 AM	Beginning Object-Relational Mapping with Hibernate Brian Sam-Bodden	JAX-RS-enabled Brian Maso	Using Ajax4jsf David Geary	Caring about your Code Quality Venkat Subramaniam
10:30 - 11:00 AM	BREAK			
11:00 - 12:30 PM	10 ways to use Hibernate effectively Brian Sam-Bodden	REST-enable Your Domain Model Brian Maso	Intro to Seam David Geary	Iteration 0 Ken Sipe
12:30 - 1:15 PM	LUNCH			
1:15 - 2:15 PM	EXPERT PANEL DISCUSSION			
2:15 - 3:45 PM	Real World Test Driven Design Scott Davis	Filthy Rich Clients with the Google Web Toolkit, Part I David Geary	Better Web Service Modeling and Specification Brian Maso	Maintaining Project Integrity with JDepend, Macker, PMD, Maven, and other open source tools David Bock
3:45 - 4:00 PM	BREAK			
4:00 - 5:30 PM	GIS for Web Developers: Adding Where to Your Application Scott Davis	Filthy Rich Clients with the Google Web Toolkit, Part II David Geary	Alternatives to Java Brian Maso	Surviving Middle Management David Bock

Western Canada Java Software Symposium

Radisson Hotel Calgary Airport

September 26 - 28, 2008

<http://www.nofluffjuststuff.com/conference/calgary/2008/09/index.html>

(event schedule as of September 27, 2008)

JAX-RS-enabled by Brian Maso

Introduction to The Java API for RESTful Services (JAX-RS). RESTful Java web services are a pretty radical departure from what you are probably familiar with. JAX-RS avoids the "Java method == service operation" typical in all the popular web service stacks, opting instead for a much more comfortable way of making information services available over HTTP. For the busy developer who wants a fast, practical introduction to RESTful services and the JAX-RS API in particular.

REST-enable Your Domain Model by Brian Maso

Expose your domain model with a REST API. Its extremely easy using JAX-RS annotations and a standard JAX-RS implementation. In this extended demo talk, we'll take an example domain model and see how to decompose it in to a RESTful API. An implementation is easy to build with JAX-RS + a domain object serializer. You'll take away the skills and tools to impress your friends and colleagues.

Better Web Service Modeling and Specification by Brian Maso

This one's for all the "architects" out there designing and specing services, and those who have to work with them. Whether you are building it or consuming it, the most painful thing about sharing a service is sharing an understanding of what the service does. This presentation teaches you how to dispel ambiguities, techno-mumbo-jumbo, and reliance on institutional knowledge that bogs down service development and testing today using the 5 essential parts of an interface specification known as Operation-State Modeling (OSM).

Alternatives to Java by Brian Maso

Alternative languages for the Java VM are hot: Groovy, JRuby, Scala, Clojure, and more. What is driving the interest in these alternative languages? Are these new, fancy-dancey languages just a fad; or maybe are you missing out by not knowing more about them? If you've been reluctant, or don't see where Java and these new languages fit in the "big picture", start your alternative explorations in this talk.

Professional Java UI development with the Eclipse RPC by Brian Sam-Bodden

Learn how to build featured rich applications using the Eclipse Rich Client Platform. The Eclipse platform is an open tools platform, on top of this platform you can build your own applications (which do not need to be IDE like or IDE related). Yet you can enjoy the benefits of working with a mature and featured rich platform that can greatly reduce the amount of time required to create a professional-looking and robust Java UI application.

Boosting Programmer productivity with Mylyn by Brian Sam-Bodden

Mylyn is a task-focused toolkit for the Eclipse IDE that allows developers to focus on tasks in a way that they never been able to do before. Mylyn eliminates the constant context switching produced by typical ways IDEs are used. No more scrolling/browsing/searching/tagging/sending emails with progress updates... Mylyn provides a new way of working that allows you to focus on specific tasks by reducing information overload. Mylyn also provides a framework for integrating with the most commonly usage task tracking systems and version control systems. In this talk you'll learn how Mylyn can boost your productivity as a Java developer by letting you get the most out of your IDE.

Beginning Object-Relational Mapping with Hibernate by Brian Sam-Bodden

Hibernate is an open source Object-Relational Mapping Framework that mostly automates the tedious and time-consuming task of persisting Java objects to a relational database. Hibernate is quickly becoming the preferred way for enterprise developers to overcome the object-relational impedance mismatch and a good alternative to the coarse-grained Entity EJBs, low-level raw JDBC, and by-committee specifications like JDO. Learn what your choices in the ORM arena, what to look for in an ORM tool, and how to get started with Hibernate for your next J2SE or J2EE project.

10 ways to use Hibernate effectively by Brian Sam-Bodden

Learn 10 tried and true ways to improve the way you use Hibernate today. In this session you would learn about a collection of 10 tips, tricks, practices and tools that will make you more effective at designing, implementing, testing and tuning your application's Hibernate-powered object-relational layer.

Estimating vs. Guessing - How Agile Teams Estimate Their Work by David Bock

Estimating is regarded as little more than 'educated guessing', but so much can hang on the quality of those estimates. With good estimates we can set clear expectations for project delivery, but with bad estimates we can run over schedule and over budget, or worse. We often estimate when we know the least about the work that needs to get done - so how can we make the best of what is potentially a bad situation?

The Agile Product Owner by David Bock

Agile software development isn't just about the development team or managers... the customer has an active role too. The customer should be prioritizing the stories in each release, potentially working onsite in constant contact with the development team, and even participating in daily status meetings. Done well, the customer's presence has a positive influence on the development iteration. Done poorly, the customer detracts from the team's focus. So how do you be the customer of an agile team? How do you teach someone to be that customer?

Maintaining Project Integrity with JDepend, Macker, PMD, Maven, and other open source tools by David Bock

How many times have you started a new project only to find that several months into it, you have a big ball of code you have to plod through to try to get anything done? How many times have you been the 'new guy' on an established project where it seems like the code grew more like weeds and brambles than a well-tended garden? With a few good structural guidelines and several tools to help analyze the code, we can keep our project from turning into that big ball of mud, and we can salvage a project that is already headed down that path.

Surviving Middle Management by David Bock

Most good developers eventually have the opportunity to be managers. Whether they call you the "project manager", "Technical Lead", "Lead Developer", or some other classic middle-management title, you become the 'goto' guy between management and developers. You're the guy who is expected to keep the project in-line, track a schedule, and occasionally answer the question "How's it going?", and perhaps still contribute at a technical level. So how do you do that?

JavaServer Faces: A Whirlwind Tour by David Geary

In April 2005, annual growth rates for jobs in JavaServer Faces, Struts, and Ruby on Rails were all at about 0%. Today, Struts' growth rate still hovers around 0%, but JSF and Rails have taken off. At the end of 2007, both JSF and Rails were growing at a rate of between 400-500% annually (according to indeed.com). JSF has passed the adoption tipping point, and is now the Java-based framework of choice, as is evidenced by its ecosystem. From vendors such as MyEclipse and RedHat to open source projects such as Seam, Facelets, and Ajax4JSF, JSF is where the action is. Come see why JSF is so popular. In this code- and demo-intensive session, I'll show you the fundamentals of JSF. **Prerequisite:** *Some knowledge of Java-based web applications, such as Struts, is a plus, but is not required. If you have a significant experience with JSF, you probably already know most of what's covered in this session.*

Facelets by David Geary

Facelets is a combination of Tiles and Tapestry, and it's the hottest JSF-related open source project on the planet. It's popularity is well deserved, and in fact, much of what is in Facelets today will make its way into the JSF 2.0 spec due out in 2008. So not only can you come to this session and see some really cool demos that you can put to use in the real world, but you'll also be learning JSF 2.0 before it's even been defined! How's that for a ROI? **Prerequisite:** *Some knowledge of JSF is essential. If you're familiar with a templating framework, such as Velocity or Tiles, that's a plus, but not required.*

Using Ajax4jsf by David Geary

Ajax4jsf makes it very easy to add Ajax to your JSF applications. Come to this presentation to see how.

Intro to Seam by David Geary

Have you ever stopped to think that you need to learn two frameworks to develop a non-trivial, database-backed, web application? Struts and iBatis; JSF and Hibernate; Tapestry and EJB3.0.

Filthy Rich Clients with the Google Web Toolkit, Part I by David Geary

The Google Web Toolkit (GWT) is truly a revolutionary framework that lets you develop Ajaxified web applications without knowing anything about Ajax or JavaScript. But the GWT goes way beyond basic Ajax by letting you implement desktop-like applications that run in the ubiquitous browser.

Filthy Rich Clients with the Google Web Toolkit, Part II by David Geary

In the second part of this talk, you will learn how to extend the GWT by implementing custom widgets, including a scrolling viewport and a drag and drop framework. After discussing custom widgets, you will see how to integrate database access into your GWT applications, and how to deploy your GWT applications to external servers.

Architecture and Scaling by Ken Sipe

Scale... what is scale... how do you applications which are scalable. How do you know if the application scales?

7 Habits of Highly Effective Developers by Ken Sipe

Thoughts lead to words, words lead to action, actions lead to habits. In this session we'll sharpen the development saw in the process of understanding what makes a hyper-productive programmer. The focus will consist of developer habits and development processes.

Spring+JPA+Hibernate: Standards Meeting Productivity for Java Persistence by Ken Sipe

Well the standards created EntityBeans.... yea. and the community created Hibernate. Fortunately the standards body learned some lessons and created JPA. JPA requires a vendor implementation and none make a better choice then Hibernate. Combined with Spring this trio is a powerhouse when it comes to developer productivity on applications requiring persistence.

Spring 2.5 - Spring without XML by Ken Sipe

Spring 2.5 is brand spanking new, with a number of fantastic features. With growth of large and complex Spring applications which struggle with xml manageability and with the added pressure of Guice and SEAM there is a push for less XML, with solution leaning towards annotations. Spring 2.5 adds to the toolset provided in Spring 2.0 to provide a development environment where XML is greatly reduced... or eliminated if you so choose.

Iteration 0 by Ken Sipe

The success of an Agile / SCRUM project is a successful start. The first interaction is often referred to as iteration 0. Other iterations have a set of stories with clear acceptance certain which establishes the velocity of the team and its effort. What then is accomplished in iteration 0? How do we get an Agile process started.

Enterprise Messaging Using JMS (Part 1) by Mark Richards

The chances are good that at some point in your career you will need to use messaging to pass information between applications, subsystems, or external systems, particularly with service-oriented architecture on the rise. The Java Messaging Service (JMS) allows Java applications to implement messaging using a standard API, thereby removing the dependency on any particular messaging provider. In Part 1 of this session we will take a look at some of the basics of messaging, including sending and receiving messages, message types, and request/reply messaging. I will begin the session by going over the basics of messaging and the JMS API. Then, through interactive coding using OpenJMS I will demonstrate how to connect to JMS providers, send messages, receive messages, and use message properties. Please note that this is a two part session.

Enterprise Messaging With JMS (Part 2) by Mark Richards

In Part 1 of the JMS session I covered messaging models, messaging basics, the JMS API, and point-to-point messaging. In this interactive code-intensive session I will cover some additional JMS topics such as browsing queues, load balancing, publishing and subscribing to messages within the pub/sub model, durable and non-durable subscribers, message selectors, and message filtering. I will also discuss and demonstrate message prioritization, persistent and non-persistent messages, and finally message

expiration (expiry). Note that this is Part 2 of a two-part JMS session. **Prerequisite:** *Enterprise Messaging With JMS (Part 1) or some knowledge of JMS*

SOA Unplugged by Mark Richards

Awareness about Service Oriented Architecture (SOA) has grown significantly in the past several years. Unfortunately, along with that growth has come a significant amount of confusion about what SOA really is. SOA has become such a ubiquitous buzzword that it now has many faces and means different things to different people. CIO's, managers, vendors, business users, architects, and developers all see SOA differently which creates a sea of confusion about what is and isn't SOA. In this highly interactive and thought provoking session we will look beyond the hype and marketure of SOA and explore SOA from an architecture and development point of view - in other words, SOA as an architecture pattern. During this session we will look at SOA use cases, services, integration, implementation, guiding architecture principles of SOA, and attempt to answer the following question: What is and isn't SOA?

EJB3 Core Specification (JSR-220) by Mark Richards

EJB3 (JSR-220) offers some great improvements over the prior EJB specs in terms of development simplicity and new features. In this session we will explore in detail some of the new features of the core EJB 3 specification. Included in this session will be a hands-on discussion and demonstration of session beans, dependency injection, interceptors (aop), and Message-Driven Beans (MDB). For the interceptors discussion I will be showing how to define interceptors for enabling a method trace, mocking objects, and sending JMS message notifications to be later picked up by the MDBs I will be creating. 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) - only the core specification.

Java Persistence: Approaching the Silver Bullet by Mark Richards

Java Persistence has come along way since the days of straight JDBC coding and custom framework development. We have at our disposal several outstanding open source frameworks such as Hibernate, Toplink, iBatis, and OpenJPA (just to name a few), and we now have a promising and emerging standards-based solution called Java Persistence API (JPA). However, all too often we find in the Java persistence space that it is a world of one-size-does-not-fit-all. We continually struggle with traditional ORM solutions like Hibernate when it comes to reporting queries, complex queries, complex relationships, and stored procedures, and we also struggle with managing the enormous amount of SQL required for solutions such as iBATIS or JDBC-based frameworks. In this coding-intensive session we will take a detailed look at identifying and overcoming the challenges we face when using frameworks such as Hibernate, iBATIS, and JPA, and how to combine the various persistence frameworks to create an effective Java persistence solution that approaches (but of course does not reach) the silver bullet.

Groovy, the Blue Pill: Writing Next Generation Java Code in Groovy by Scott Davis

There are wild-eyed radicals out there telling you that Java is dead, statically-typed languages are passe, and your skills are hopelessly out-of-date. Those extremists are the same ones who don't bat an eye at throwing out years of experience to learn a new language from scratch, pushing aside a familiar IDE for a new one, and deploying to a whole new set of production servers with little regard to legacy integration. While this "burn the boats" approach to software development might sound exciting to some folks, it's giving your manager the cold shakes right now. What if I told you that there was a way that you could integrate seamlessly with your legacy Java code, continue to use your trusty IDE and stable production servers, and yet take advantage of many of the exciting new dynamic language features that those fanatics keep prattling on about? You'd probably say, "Groovy!" I would, too...

Groovy, The Red Pill: Metaprogramming, the Groovy Way to Blow a Buttoned-Down Java Developer's Mind by Scott Davis

This talk focuses on the ways that Groovy can turn a traditional Java developer's world-view upside down. We'll start by talking about how you can thumb your nose at The Man by leaving out many of the main syntactic hallmarks of Java: semicolons, parentheses, return statements, type declarations (aka Duck-typing), and the ever-present try/catch block. Then we'll look at features like operator overloading and method pointers that Groovy welcomes back into the language with open arms.

Rapid Web Development with Grails and Ajax by Scott Davis

Grails is a Java- and Groovy-based web framework that is built for speed. First-time developers are amazed at how quickly you can get a page-centric MVC web site up and running thanks to the scaffolding and

convention over configuration that Grails provides. Advanced web developers are often pleasantly surprised at how easy it is to break out of that coarse-grained navigation model using the native Ajax support baked into the framework.

Real World Test Driven Design by Scott Davis

Everyone has their favorite excuses for not writing unit tests: "It takes too much time", "It's not my job", "But it compiles!" In this presentation we talk about the importance of testing, and how the act of writing your own unit tests leads to better architected code.

GIS for Web Developers: Adding Where to Your Application by Scott Davis

Based on the book GIS for Web Developers, this talk demonstrates how you can build your own Google Maps in-house using nothing but open source software. The Portland, Oregon Transit Authority recently migrated from a proprietary web mapping solution to the suite of 100% free and open source software discussed in this book. We look at Java-based clients, Java-based servers, and everything in between. We also discuss integrating free, public domain data from sources like the US Census Bureau and the USGS. If you're looking for real-world examples of AJAX in use, you'll find it here. If you're looking for real-world examples of web services in use, you'll find it here.

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 Concurrency (Part 1: Threads) by Ted Neward

Java's threading capabilities took a serious turn for the better with the release of Java5, thanks to the incorporation of the java.util.concurrent packages, a set of pre-built components for thread pooling and execution, synchronization, and more.

The Busy Java Developer's Guide to Concurrency (Part 2: Concurrency) by Ted Neward

Java's threading capabilities have been a part of the Java platform since its inception, yet for many Java developers, using Threads still remain a dark and mysterious art, and synchronization beyond the use of the "synchronized" keyword is almost unknown. *Prerequisite: The Busy Java Developer's Guide to Concurrency (Part 1: Threads)*

The Busy Java Developer's Guide to Hacking (on) the JDK by Ted Neward

Ever since its 1.1 release, the Java Virtual Machine steadily becomes a more and more "hackable" (configurable, pluggable, customizable, choose your own adjective here) platform for Java developers, yet few, if any, Java developers take advantage of it. Time to take the kid gloves off, crack open the platform, and see what's there. Time to play.

The Busy Developer's Guide to Scala by Ted Neward

Scala is a new programming language incorporating the most important concepts of object-oriented and functional languages and running on top of the Java Virtual Machine as standard "dot-class" files.

Towards an Evolutionary Design by Venkat Subramaniam

A good design is critical for success with agile development. That does not mean a big up-front design. The design has to be evolutionary. However, the design you evolve must be extensible and maintainable. After all, you can't be agile if your design sucks.

Tools to facilitate Agile Development by Venkat Subramaniam

The first item in the Agile Manifesto reads that we must prefer "people and interaction over process and tools." Given a choice between average people with superior tools and superior people with average tools, you are likely to achieve greater success with the latter. However, it is important to be continuous and not

be episodic?so you want to get continuous feedback about the state, health, and quality of your code and application. Tools can help us a great deal to realize this and make us productive.

Acceptance Testing Application Behavior by Venkat Subramaniam

How do you ensure your applications meet the expectations of your key customers? In this session we will explore using the FIT tool and Behavior Driven Design tools to do exactly this.

Keynote: Here's your sign, Fallacies, and Responsibilities by Venkat Subramaniam

Keynote topic on Fallacies in software development and developer responsibilities.

Practices of an Agile Developer by Venkat Subramaniam

You have worked on software projects with varying degree of success. What were the reasons for the success of your last project? What were the reasons for those that failed? A number of issues contribute to project success - some non-technical in nature. In this presentation the speakers will share with you practices in a number of areas including coding, developer attitude, debugging, and feedback. The discussions are based on the 2007 Jolt productivity award winning book with the same title as the talk.

BDD in Java and Groovy by Venkat Subramaniam

In this presentation we will take a look at what BDD is and look at tools to create them in Java and Groovy.

Caring about your Code Quality by Venkat Subramaniam

We all have seen our share of bad code. We certainly have come across some good code as well. What are the characteristics of good code? How can we identify those? What practices can promote us to write and maintain more of those good quality code. This presentation will focus on this topic that has a major impact on our ability to be agile and succeed.