

# Lone Star Software Symposium: Dallas

Crowne Plaza Suites Dallas

June 4 - 6, 2010

The No Fluff Just Stuff Java Symposium Series is proud to announce the return of the Lone Star Software Symposium: Dallas on June 4 - 6, 2010. LSSS2010Dallas will be held at the Crowne Plaza Suites Dallas.

Since 2001, the No Fluff Just Stuff Java Symposium has been regarded as the premier Java/Agility event series anywhere serving over 29,500 attendees with some 190 events. The popularity of the NFJS symposium series can be traced to the following:

- 1). Exceptional Speakers
- 2). Limited Attendance - capped at 250 people
- 3). No Vendors, No Sales Pitches, No Marketecture
- 4). Excellent networking opportunity with speakers and fellow attendees because of small size.
- 5). The Best Value in the Java conferencing space period.

## **Topics for the 2010 NFJS Tour**

Languages on the JVM: Groovy, JRuby, Clojure, Scala  
HTML5

Enterprise Java

Core Java, JVM Internals

No SQL: MongoDB

JSF, GWT

Agility

Groovy, Grails, Gradle

REST, RDFa, Resource Oriented Architectures

GIT Version Control

jQuery, Ajax, Flex, RIA

Mobile Applications - iPhone and Android

More...

## **The Registration Fee Includes:**

3 Day All Access Pass to LSSS2010Dallas

All Meals/Snacks - duration of the symposium

90 Day IntelliJ license compliments of JetBrains

Session Materials

Custom NFJS Binder

Great Giveaways @ NFJS

**Early Bird Registration:** \$850/person good thru 5/10/10 after \$950

**Excellent Group Discounts Available** - bring your entire development team to the show - no travel required!! Rate good thru 5/10/10

## **Registration Fees**

<b>Attendees</b>	<b>Before May. 10, 2010</b>	<b>After May. 10, 2010</b>
5-9	\$750	\$850
10-14	\$725	\$825
15-24	\$700	\$800
25+	\$675	\$775

Go to <http://www.nofluffjuststuff.com/conference/dallas/2010/06/home> and register today!

# Lone Star Software Symposium: Dallas

Crowne Plaza Suites Dallas

June 4 - 6, 2010

**Fri, Jun. 04, 2010**

	Windsor 2-4	Windsor 1	Warwick	Canturbury	Churchill
12:00 - 1:00 PM	REGISTRATION				
1:00 - 1:15 PM	WELCOME				
1:15 - 2:45 PM	Implementing Evolutionary Architecture Neal Ford	Grails - How to Build Enterprise Apps Jeff Brown	The Busy Java Developer's Guide to Collections Ted Neward	Open Source Business Intelligence - Part I Tim Berglund	REST : Information-Driven Architectures for the 21st Century Brian Sletten
2:45 - 3:15 PM	BREAK				
3:15 - 4:45 PM	Emergent Design Neal Ford	GORM Inside And Out Jeff Brown	The Busy Java Developer's Guide to Functional Java Ted Neward	Open Source Business Intelligence - Part II Tim Berglund	RDFa : Weaving Richness and Meaning in the Web Brian Sletten
4:45 - 5:00 PM	BREAK				
5:00 - 6:30 PM	Testing the Entire Stack Neal Ford	Polyglot Web Programming With Grails Jeff Brown	The Busy Java Developer's Guide to Advanced Collections Ted Neward	Practical Agile Database Development Tim Berglund	SPARQL : Querying the Web of Data Brian Sletten
6:30 - 7:15 PM	DINNER				
7:15 - 8:00 PM	Keynote: by Neal Ford				

**Sat, Jun. 05, 2010**

	Windsor 2-4	Windsor 1	Warwick	Canturbury	Churchill
8:00 - 9:00 AM	BREAKFAST				
9:00 - 10:30 AM	Visualizations for Code Metrics Neal Ford	HTML 5 ... and the Kitchen Sink Brian Sletten	Third time's a charm: What's new in Spring 3.0 Craig Walls	The Modern Enterprise Java Development Environment Brian Sam-Bodden	The Busy Java Developer's Guide to Concurrency (Part 1: Threads) Ted Neward
10:30 - 11:00 AM	BREAK				
11:00 - 12:30 PM	Agile Engineering Practices Neal Ford	Semantic SOA : Meaningful Service Strategies Brian Sletten	Introducing Spring Roo: From Zero to Working Spring Application in Record Time Craig Walls	Tools and Techniques to build Smart Java Applications Brian Sam-Bodden	The Busy Java Developer's Guide to Concurrency (Part 2: Concurrency) Ted Neward
12:30 - 1:30 PM	LUNCH				
1:30 - 3:00 PM	Aspect Oriented Programming With Spring AOP Jeff Brown	What's Brewing in Java Venkat Subramaniam	Pragmatic Architecture Ted Neward	Groovy on the Cloud Brian Sam-Bodden	Decision Making in Software Teams Tim Berglund
3:00 - 3:15 PM	BREAK				
3:15 - 4:45 PM	Compile Time and Runtime Metaprogramming With Groovy Jeff Brown	How to Approach Refactoring Venkat Subramaniam	The Busy Java Developer's Guide to Scala: Basics Ted Neward	Relational Database Alternatives Brian Sam-Bodden	Automated Software Quality Control Tools Pratik Patel
4:45 - 5:45 PM	300 Sessions & Refactoring Workshop with Venkat Subramaniam				

**Sun, Jun. 06, 2010**

	Windsor 2-4	Windsor 1	Warwick	Canturbury	Churchill
8:00 - 9:00 AM	BREAKFAST				
9:00 - 10:30 AM	jQuery: Ajax Made Easy Nathaniel Schutta	Enterprise JPA & Spring 3.0 - Tips and Tricks for JEE Persistence Pratik Patel	Beyond JUnit: Powertools for Test-Driven Development Craig Walls	Cloud computing deep dive for Google App Engine and Amazon EC2 Rohit Bhardwaj	Scala Tricks Venkat Subramaniam
10:30 - 11:00 AM	MORNING BREAK				
11:00 - 12:30 PM	Testing the Web Layer Nathaniel Schutta	Real-world JEE performance troubleshooting & tuning: Tips n' Tricks Pratik Patel	Building RESTful Applications with Spring MVC Craig Walls	Enterprise Cloud testing to make your application foolproof Rohit Bhardwaj	Tackling Concurrency on the JVM Venkat Subramaniam
12:30 - 1:15 PM	LUNCH				
1:15 - 2:15 PM	EXPERT PANEL DISCUSSION				
2:15 - 3:45 PM	JavaScript Beyond the Basics Nathaniel Schutta	Virtualization for development Pratik Patel	Testing with dependencies Venkat Subramaniam	Enterprise Security, Privacy and Data compliance Rohit Bhardwaj	Modular Java: An Introduction to OSGi Craig Walls
3:45 - 4:00 PM	BREAK				
4:00 - 5:30 PM	Hacking Your Brain for Fun and Profit Nathaniel Schutta	Easy mobile development (iPhone, Android, Palm Pre, Blackberry) without native code Pratik Patel	Transforming to Groovy Venkat Subramaniam	SoapUI for testing SOAP and Restful web services Rohit Bhardwaj	Modular Java: Declarative OSGi with Spring Dynamic Modules Craig Walls

# Lone Star Software Symposium: Dallas

Crowne Plaza Suites Dallas

June 4 - 6, 2010

## **Open Source Business Intelligence - Part I by Tim Berglund**

Traditionally, business intelligence tools have been a high-cost part of any enterprise's software inventory. Recently, options have emerged that allow architects to build a credible business intelligence stack out of entirely open-source components. In this brief overview, we will demonstrate ETL, reporting, and analytics tool that can be deployed free or at low cost. Learn how to turn your company's transactional database into a rich data asset with a business-friendly user interface that integrates into your existing software infrastructure.

## **Open Source Business Intelligence - Part II by Tim Berglund**

Once you're familiar with the concepts of data warehousing, star schemas, cubes, and pivot tables, then it's time to dive in and look at how the tools really work. Continuing from the quick demos in Part I, in this talk we'll walk through the process of transforming a transactional database into a star schema, then we'll use an open-source analytics tool to build a "cube" with that schema. Concepts and procedures gently introduced in Part I will be explored more thoroughly, and new tooling will be introduced. **Prerequisite:** *Open Source Business Intelligence Part I (or a solid grasp of BI concepts)*

## **Practical Agile Database Development by Tim Berglund**

Do your team's agile practices extend to the database? Agile methods are fairly well-understood as they apply to code, but these principles are not commonly understood or practiced on the databases that typically accompany enterprise software projects. Learn the tools, techniques, and mindset your team needs to make incremental improvements to the database's design over time with confidence.

## **Decision Making in Software Teams by Tim Berglund**

Alistair Cockburn has described software development as a game in which we choose among three moves: invent, decide, and communicate. Most of our time at No Fluff is spent learning how to be better at inventing. Beyond that, we understand the importance of good communication, and take steps to improve in that capacity. Rarely, however, do we acknowledge the role of decision making in the life of software teams, what can cause it to go wrong, and how to improve it.

## **Cloud computing deep dive for Google App Engine and Amazon EC2 by Rohit Bhardwaj**

In this session we will take a deep dive at few cloud computing examples from real world and participants will be able to know how to use cloud computing for Google App Engine, Amazon EC2 and few others.

## **Enterprise Cloud testing to make your application foolproof by Rohit Bhardwaj**

In Cloud testing we cannot use our existing tools which run in isolation. There is a great need for tools that allow the Software developers and Testers to monitor and test the services provided in the cloud. This include finding any problem in current implementation. In the presentation we will explore Cloud Tools and PushToTest tools to automatically test web applications. Attendees will learn different test strategies for cloud testing.

## **Enterprise Security, Privacy and Data compliance by Rohit Bhardwaj**

Data integrity, security, recovery, privacy and regulatory compliance are most important attributes for enterprise implementation. Enterprise customers ask for transparency in how the vendors will provide security programs. Many question need to be asked for any cloud implementation to policy makers, architects, coders and testers. In this presentation we will explore data security and storage, privacy and data compliance issues. We will explore the security management in cloud. Presentation is useful for anyone starting from Executives to developers who are going to implement the enterprise Applications in both private and public cloud.

## **SoapUI for testing SOAP and Restful web services by Rohit Bhardwaj**

SoapUI is an Open Source Web Service Testing Tool for Service Oriented Architecture made for the Software Developers and Testers. Its functionality mainly covers Web Service Inspection, Invoking, Development, Simulation and Mocking, Functional testing, Load and Compliance testing. Productivity enhancement features can be found in the soapUI pro version. SoapUI is the premiere open-source web

service testing tool with over 50000 users worldwide. We will explore few online services like Amazon Web Services. This presentation will show you how to supercharge your web service development and testing efforts: • fast inspection and invocation of web services • validation of contracts, messages and message-exchanges • tool-integrations for all major web service frameworks • powerful functional testing with extensive groovy support • instant simulation/mocking of any web service with groovy support • requirements-driven load-testing **Prerequisite:** none

### **Grails - How to Build Enterprise Apps by Jeff Brown**

Grails represents technology that offers great flexibility and power without the complexity introduced by other Java web application frameworks. Custom tag libraries are a snap. GSP Templates provide a simple mechanism for reusing UI elements. Sitemesh is integrated to help provide a consistent presentation across the entire application. GORM is super powerful ORM. Grails provides simple mechanisms for leveraging the power of Ajax.

### **GORM Inside And Out by Jeff Brown**

GORM is a super powerful ORM tool that makes ORM simple by leveraging the flexibility and expressiveness of a dynamic language like Groovy. With GORM developers get access to all of the power and flexibility of an ORM tool like Hibernate without any of the complexity. **Prerequisite:** Advanced Grails

### **Polyglot Web Programming With Grails by Jeff Brown**

Grails is one of the most flexible and most powerful frameworks on The Java Platform. Grails leverages the flexibility offered by the platform in a way that other web frameworks do not. Grails is a fantastic platform for polyglot web programming. **Prerequisite:** Advanced Grails

### **Aspect Oriented Programming With Spring AOP by Jeff Brown**

Aspect-Oriented Programming (AOP) complements Object-Oriented Programming (OOP) by providing another way of thinking about program structure. The key unit of modularity in OOP is the class, whereas in AOP the unit of modularity is the aspect. Aspects enable the modularization of concerns such as transaction management that cut across multiple types and objects. (Such concerns are often termed crosscutting concerns in AOP literature.)

### **Compile Time and Runtime Metaprogramming With Groovy by Jeff Brown**

The dynamic nature of Groovy makes it a fantastic language for building dynamic applications for the Java Platform. The metaprogramming capabilities offered by the language provide everything that an application development team needs to build systems that are far more capable than their all Java counterparts. Taking advantage of Groovy's metaprogramming capabilities brings great new possibilities that would be very difficult or just plain impossible to write with Java alone. Building Domain Specific Languages in Groovy is easy to do once a team has a good understanding of the Metaobject-Protocol (MOP) and the method dispatch mechanisms used by the Groovy runtime environment.

### **Implementing Evolutionary Architecture by Neal Ford**

This talk describes an agile approach to architecture, and merges the current state-of-the-art thinking in both service oriented architectures(SOA) and web-based architectures like HTTP, REST, and hypermedia.

### **Emergent Design by Neal Ford**

Emergent design is a big topic in the agile architecture and design community. This session covers the theory behind emergent design and shows examples of how you can implement this important concept.

**Prerequisite:** understanding of architectural and design concepts

### **Testing the Entire Stack by Neal Ford**

This talk covers testing the entire stack: unit, integration, functional, behavior-driven, databases, user acceptance, mocking & stubbing, and other topics and strategies. **Prerequisite:** Confusion about what to test when and where

### **Keynote: Smithing in the 21st Century by Neal Ford**

Blacksmiths in 1900 and PowerBuilder developers in 1996 have something in common: they thought their job was safe forever. Yet circumstances proved them wrong. One of the nagging concerns for developers is how do you predict the Next Big Thing, preferably before you find yourself dinosaurized. This keynote discusses why people are bad at predicting the future, and why picking the Next Big Thing is hard. Then, it

foolishly does just that: tries to predict the future. I also provide some guidelines on how to polish your crystal ball, giving you tools to help ferret out upcoming trends. Don't get caught by the rising tide of the next major coolness: nothing's sadder than an unemployed farmer watching cars drive by.

### **Visualizations for Code Metrics by Neal Ford**

Judicious use of metrics improves the quality of your code. But interpreting metrics presents a challenge. You have a list of numbers for a project - what does it mean? And what does it tell me about the health of the project overall? This session shows how to produce visualizations for software metrics, making them easier to understand and more valuable. It covers metrics at the individual method level all the way up to the overall architecture of the application. This isn't just a talk about how some tools produce visualizations: this session shows you how to generate your own visualizations, allowing you to customize it to the level in information density that shows real value on your project. I show how to produce projected graphs from dependencies, heat-maps for cyclomatic complexity and code coverage, using XSLT to extract visual information from XML configuration documents, and others. Metrics can't help you if you can't understand them. By creating visualizations, it helps leverage metrics to make your code better.

### **Agile Engineering Practices by Neal Ford**

Most of the time when people talk about agile software development, they talk about project and planning practices and never mention actual development practices. This talk delves into best development practices for agile projects, covering all of its aspects. **Prerequisite:** *Having worked in an organization that values bureaucracy more than individuals*

### **The Busy Java Developer's Guide to Collections by Ted Neward**

For so many Java developers, the `java.util.*` package consists of `List`, `ArrayList`, and maybe `Map` and `HashMap`. But the Collections classes are so much more powerful than many of us are led to believe, and all it requires is a small amount of digging and some simple exploration to begin to "get" the real power of the Collection classes.

### **The Busy Java Developer's Guide to Functional Java by Ted Neward**

Much noise has been made in recent years about functional languages, like Scala or Haskell, and their benefits relative to object-oriented languages, most notably Java. Unfortunately, as wonderful as many of those benefits are, the fact remains that most Java developers will either not want or not be able to adopt those languages for writing day-to-day code. Which leaves us with a basic question: if I can't use these functional languages to write production code, is there any advantage to learning about them? The short answer is yes, for the fundamental premise--"I can't use functional code on my Java project"--is flawed. Java developers can, in fact, make use of functional ideas, and what's better, they don't even have to reinvent them for Java--thanks to the `FunctionalJava` library, many of the core primitives--interfaces that serve as base types for creating function values, for example--already exist, ready to be used.

### **The Busy Java Developer's Guide to Advanced Collections by Ted Neward**

Once you've learned the core Collections classes, you're done, right? You know everything there is to know about Collections, and you can "check that off" your list of Java packages you have to learn and know, right? **Prerequisite:** *Busy Java Developer's Guide to Collections*

### **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)*

### **Pragmatic Architecture by Ted Neward**

Building an application is not the straightforward exercise it used to be. Decisions regarding which architectural approaches to take (n-tier, client/server), which user interface approaches to take (Smart/rich client, thin client, Ajax), even how to communicate between processes (Web services, distributed objects,

REST)... it's enough to drive the most dedicated designer nuts. This talk discusses the goals of an application architecture and why developers should concern themselves with architecture in the first place. Then, it dives into the meat of the various architectural considerations available; the pros and cons of JavaWebStart, ClickOnce, SWT, Swing, JavaFX, GWT, Ajax, RMI, JAX-WS, , JMS, MSMQ, transactional processing, and more.

### **The Busy Java Developer's Guide to Scala: Basics 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. Sporting the usual object-oriented concepts as classes and inheritance, Scala also offers a number of powerful functional features, such as algebraic data types, immutable objects by default, pattern matching, closures, anonymous functions and currying, and more. Combined with some deep support for XML generation and consumption, Scala offers Java programmers an opportunity to write powerful programs with concise syntax for a new decade of Java programming.

### **Automated Software Quality Control Tools by Pratik Patel**

This session is aimed at helping developers get started with automating the collection of software quality metrics. We'll cover continuous integration, automated code metrics gathering, and analysis of these metrics. The ability to be able to detect problems early, and also to write higher quality code early, helps avoid bugs and headache down the line. We'll cover some best practices around using and putting in tools to help achieve higher quality.

### **Enterprise JPA & Spring 3.0 - Tips and Tricks for JEE Persistence by Pratik Patel**

As with many technologies, the basics are easy. The hard part comes when the developer needs to do sophisticated integration, development, and testing as part of an enterprise application. A large enterprise application requires the developer to think of issues that affect the development, scalability and robustness of the application. This presentation will cover the advanced topics described below with a focus on the new persistence features in Spring 3.0 and JPA 2.0.

### **Real-world JEE performance troubleshooting & tuning: Tips n' Tricks by Pratik Patel**

Performance tuning any application is a black art that can consume much time. Fortunately, Java has many tools that can aid in this effort. There also are a number of basic tips that can help to analyze and fix performance problems. The Java memory model is usually something that you don't need to tune, but for high performance applications it is necessary to tweak. While there are a number of advanced things that can be done to performance tune an application, we'll discover that the simple, basic things are all that are usually needed to make your apps fly.

### **Virtualization for development by Pratik Patel**

We've all heard about virtualization for deploying applications. How about leveraging virtualization for development? In this session, we'll look at some time saving tips and build a virtual VM for development and testing.

### **Easy mobile development (iPhone, Android, Palm Pre, Blackberry) without native code by Pratik Patel**

So you have a great idea for an iPhone app, you've tried learning Objective-C, but it's just too hard. What about those other new platforms like Palm Pre and Android? Who wants to write the same app three times? Four times if you count Blackberry! Fear not, there is a much easier way for you to develop on the iPhone. Using a development style called "hybrid mobile applications" you can write apps for iPhone and other platforms using stuff you already know: HTML, CSS and Javascript. In this course, we'll go over the basics for hybrid development

### **The Modern Enterprise Java Development Environment by Brian Sam-Bodden**

In this session we'll look at what a modern Java/Java EE environment could and should look like. This session is a survey of the software infrastructure that needs to be in place to create a productive and successful development environment. Version control, continuous integration, metrics and static code analysis and more. Learn what needs to be setup early, what can wait, and what can be sourced out and how to deal with privacy and security issues.

### **Tools and Techniques to build Smart Java Applications by Brian Sam-Bodden**

In this session we will explore the Java tools, techniques and algorithms that enable us to filter, classify, relate and discover patterns in our data that might not immediately be obvious. With the emergence of social networking applications a great deal of data and hidden connections that can be leveraged to build better and smarter applications.

### **Groovy on the Cloud by Brian Sam-Bodden**

In this session we'll talk about how the combination of cloud computing a flexible, lightweight dynamic language like Groovy and a few architectural and design principles can be used to create highly scalable and maintainable web applications. In this session we'll look into EC2 and other cloud offerings and what Groovy offers to tap into the power of commodity cloud computing platforms.

### **Relational Database Alternatives by Brian Sam-Bodden**

In our industry the database is king and we we say database, more often than not we mean "relational database". This talk is a survey of the many alternatives to a relational database and the situations in which they can (and sometimes must) be used. In this talk we'll discuss systems like MongoDB, CouchDB, Neo4J, Cassandra and others and how can they be used with your existing Java infrastructure

### **jQuery: Ajax Made Easy by Nathaniel Schutta**

Sure, Ajax might not be the hardest thing you'll have to do on your current project, but that doesn't mean we can't use a little help here and there. While there are a plethora of excellent choices in the Ajax library space, jQuery is fast becoming one of the most popular. In this talk, we'll see why. In addition to it's outstanding support for CSS selectors, dirt simple DOM manipulation, event handling and animations, jQuery also supports a rich ecosystem of plugins that provide an abundance of top notch widgets. Using various examples, this talk will help you understand what jQuery can do so you can see if it's right for your next project.

### **Testing the Web Layer by Nathaniel Schutta**

While your project might have nearly 100% code coverage on the server tier, many projects ignore testing the web layer. With more and more code being pushed to the browser, a lack of tests for the client code begs for trouble.

### **JavaScript Beyond the Basics by Nathaniel Schutta**

JavaScript is one of the most widely used languages around and yet its also one of the most misunderstood. With Ajaxified UIs becoming the norm, this humble language is once again at the forefront.

### **Hacking Your Brain for Fun and Profit by Nathaniel Schutta**

The single most important tool in any developers toolbox isn't a fancy IDE or some spiffy new language - it's our brain. Despite ever faster processors with multiple cores and expanding amounts of RAM, we haven't yet created a computer to rival the ultra lightweight one we carry around in our skulls - in this session we'll learn how to make the most of it. We'll talk about why multitasking is a myth, the difference between the left and the right side of your brain, the importance of flow and why exercise is good for more than just your waist line.

### **REST : Information-Driven Architectures for the 21st Century by Brian Sletten**

There is a shift going on in the Enterprise. While still used and useful, the promises of the SOAP/WSDL/UDDI Service-Oriented Architecture (SOA) stack have failed to live up to their promise. A new vision of linked information is enveloping online and Enterprise users. The REST architectural style is squarely behind this thinking as a way of achieving low-cost, flexible integration, increased data security, greater scalability and long-term migration strategies. If you have dismissed REST as a toy or are unfamiliar with it, you owe it to yourself to see what is so interesting about this way of doing things.

### **RDFa : Weaving Richness and Meaning in the Web by Brian Sletten**

The human web is reasonably well in hand by now. We are getting pretty good at building systems that people find valuable and entertaining. We have not spent as much time concerned about our software friends. There is a ton a rich content available on the web that is too difficult to extract in automated ways using just XHTML, the meta tag and microformats. This talk will introduce you to some emerging

technologies from the Semantic Web camp to enrich your web pages with useful information for both automated extraction and improved browsing experiences.

### **SPARQL : Querying the Web of Data by Brian Sletten**

The human-friendly Web is about nicely-formatted, accessible content for users to browse. There are emerging Data Webs (both public and private) that rely on technologies from the Semantic Web stack to link increasingly rich connections between various data sources. SPARQL and RDF are the main tools for expressing and using this connectivity. This talk will introduce you to one of these topics and the practical and accessible aspects of employing them on the Web and in the Enterprise. Getting people to come to consensus on common models and schemas is usually the hardest part of any data integration strategies. These technologies help lower the bar on both the technical and social costs of stepping up your integration strategies.

### **HTML 5 ... and the Kitchen Sink by Brian Sletten**

HTML 5 is an adventurous and confusing prospect that will help change the Web as we know it. It is being finalized as a standard but won't be fully supported by most browsers for quite some time. Companies like Apple and Google have already committed to it as the future of Web application development, however. There are a huge number of new features, updates and gotchas coming at us (including the proverbial kitchen sink!) so it is time to get prepared. This talk will walk you through the new bits and try to put it all into perspective.

### **Semantic SOA : Meaningful Service Strategies by Brian Sletten**

The goal for web services was always to reduce our burden by increasing the potential for reuse of business functionality. Somehow, we got lost along the way in a morass of confusing, unfulfilling and downright broken technologies. While we are interested in pursuing REST-based systems for managing information, we need some strategies for tying it all together sensibly. If we abandon WSDL, SOAP and UDDI, what do we replace them with? This talk will walk you through combining resource-oriented strategies with technologies from the Semantic Web to describe, find, and bind to services in dynamic, flexible and extensible ways. We will start to blur the distinction between data, documents, services and focus on information and how it is connected to what we already know. **Prerequisite:** *The Semantic Web: The Future Now, Give it a REST and SPARQL : Querying the Data Web would all be helpful talks to have attended*

### **What's Brewing in Java by Venkat Subramaniam**

Java has come a long way, and yet there is so much that's happening in this space. In this presentation we will take a look at the exciting additions and changes coming up in the next version of Java. **Prerequisite:** *Good programming knowledge of Java*

### **How to Approach Refactoring by Venkat Subramaniam**

You can't be agile if your code sucks. You know that you have to constantly refactor your code and design. But the questions is how? In this presentation, instead of looking at a laundry list of refactoring techniques, we will instead look at how to effectively approach refactoring and along the way discuss some core principles to look for.

### **Scala Tricks by Venkat Subramaniam**

Scala is a very powerful hybrid functional pure object oriented language on the JVM. Scala is known for its conciseness and expressiveness. In this presentation we will look at some common tasks you do everyday in developing applications and see how they manifest in Scala.

### **Tackling Concurrency on the JVM by Venkat Subramaniam**

In this presentation we will take a quick walk through the issues with concurrency and how the solutions provided in Scala and Clojure help address those.

### **Testing with dependencies by Venkat Subramaniam**

Testing is a key ingredient to the success of a project. However, testing becomes awfully hard when your application deals with dependencies and that is often the reality.

### **Transforming to Groovy by Venkat Subramaniam**

Groovy is a elegant, dynamic, agile, OO language. I like to program in Groovy because it is fun and the code is concise and highly expressive. Writing code in a language is hardly about using its syntax, however. It

is about using the right idioms. Come to this section to pick up some nice Groovy idioms. **Prerequisite:** *Some knowledge of Groovy is helpful but not required.*

### **Third time's a charm: What's new in Spring 3.0 by Craig Walls**

In this session, I'll lead a guided tour through the latest that Spring has to offer. Whether you're a Spring veteran or a Spring newbie, there will be something new for nearly everyone.

### **Introducing Spring Roo: From Zero to Working Spring Application in Record Time by Craig Walls**

In this example-driven session we'll see how to swiftly develop Spring applications using Spring Roo. We'll start with an empty directory and quickly work our way up to a fully functioning web application. You'll see how Roo handles a lot of heavy-lifting that you'd normally have to do yourself when working with Spring. And we'll stop at a few scenic points along the way to see how Roo accomplishes some of its magic.

### **Beyond JUnit: Powertools for Test-Driven Development by Craig Walls**

Writing tests is more than just writing JUnit test cases and hoping that they'll pass when your project is built. If you want assurance that your code is sound and provides the desired functionality, then you'll want to test it from every angle and run those tests as frequently as possible.

### **Building RESTful Applications with Spring MVC by Craig Walls**

In this session, we'll look at the latest features of Spring 3.0's MVC framework, with an emphasis on building RESTful web applications. You'll see how Spring 3.0's new `@PathVariable` annotation can be used to develop controllers that respond to resource-oriented RESTful URLs and how `ContentNegotiatingViewResolver` can turn a user-facing web application into a powerful web-based API. We'll explore other ways Spring 3.0 supports working with REST, including how to write REST clients.

### **Modular Java: An Introduction to OSGi by Craig Walls**

Contrary to what you may have heard, OSGi is neither complex, nor heavyweight. In this session, I'll show you how OSGi can actually simplify application development rather than complicate it. We'll look at the benefits of modularity, the fundamentals of OSGi, and see how to develop basic OSGi bundles. We'll also see how a few gadgets in the OSGi toolbox can ease the development of OSGi bundles.

### **Modular Java: Declarative OSGi with Spring Dynamic Modules by Craig Walls**

This session will introduce you to Spring-DM. You'll learn how working with OSGi services can be as easy and as natural as declaring a in Spring. In addition, we'll look at how to use Spring-DM's web extender to develop modular web applications in OSGi. And we'll see how Spring-DM became part of the OSGi specification as Blueprint Services. **Prerequisite:** *Modular Java: An introduction to OSGi*