

# Greater Atlanta Software Symposium

Atlanta Marriott Perimeter Center

May 16 - 18, 2008

<http://www.nofluffjuststuff.com/conference/atlanta/2008/05/index.html>

(event schedule as of May 14, 2008)

Fri, May. 16, 2008					
	Salon A-C	Salon D	Salon E	Monroe/Jackson	Jefferson/Washington
12:00 - 1:00 PM	REGISTRATION				
1:00 - 1:15 PM	WELCOME				
1:15 - 2:45 PM	A Thorough Introduction To Groovy Jeff Brown	7 Habits of Highly Effective Developers Ken Sipe	Evolutionary SOA Neal Ford	Developing Rich Internet Applications Richard Monson-Haefel	Credit Card Software Development: Recognizing and Repaying Technical Debt Jared Richardson
2:45 - 3:15 PM	BREAK				
3:15 - 4:45 PM	Grails - Agile Web 2.0 The Easy Way Jeff Brown	Java Memory, Performance and the Garbage Collector Ken Sipe	Test Driven Design Neal Ford	10 Things Every Software Architect Should Know Richard Monson-Haefel	10 Tips for Getting Your Project Back on Track Jared Richardson
4:45 - 5:00 PM	BREAK				
5:00 - 6:30 PM	Advanced Web Development With Grails Jeff Brown	Architecture and Scaling Ken Sipe	Regular Expressions in Java Neal Ford	Understanding Open Source Licensing Richard Monson-Haefel	Techniques 2008 Jared Richardson
6:30 - 7:15 PM	DINNER				
7:15 - 8:00 PM	Keynote: by Neal Ford				

Sat, May. 17, 2008					
	Salon A-C	Salon D	Salon E	Monroe/Jackson	Jefferson/Washington
8:00 - 9:00 AM	BREAKFAST				
9:00 - 10:30 AM	Designing for Ajax, part 1 Nathaniel Schutta	Simplifying Enterprise Applications with Spring, Part 1 Mark Fisher	Iteration 0 Ken Sipe	Powerful Metaprogramming Techniques With Groovy Jeff Brown	Tactical Continuous Integration with Hudson Andrew Glover
10:30 - 11:00 AM	BREAK				
11:00 - 12:30 PM	Designing for Ajax, part 2 Nathaniel Schutta	Simplifying Enterprise Applications with Spring, Part 2 Mark Fisher	SOAs Challenges Ken Sipe	Agile Test Driven Development With Groovy Jeff Brown	Build Teams, Not Products Jared Richardson
12:30 - 1:30 PM	LUNCH				
1:30 - 3:00 PM	JMX and Spring: Manageability for Spring-based Applications Ken Sipe	Introduction to Hibernate Scott Leberknight	Groovin' builds Gant get any easier Andrew Glover	Dojo: Getting Started Nathaniel Schutta	Agile Software Testing Strategies Jared Richardson
3:00 - 3:15 PM	BREAK				
3:15 - 4:45 PM	Configuring Spring with Annotations Mark Fisher	Real World Hibernate Tips Scott Leberknight	Hacking - The Dark Arts Ken Sipe	JavaScript: the Good, the Bad, and the Ugly Nathaniel Schutta	Easy BDD with Groovy Andrew Glover
4:45 - 5:30 PM	BIRDS OF A FEATHER SESSION				

Sun, May. 18, 2008					
	Salon A-C	Salon D	Salon E	Monroe/Jackson	Jefferson/Washington
8:00 - 9:00 AM	BREAKFAST				
9:00 - 10:30 AM	Code Metrics & Analysis for Agile Projects Neal Ford	Enterprise Integration Patterns with Spring - Part I Mark Fisher	What You Don't Know About Cryptography Roman Hustad	Guerilla Unit Testing Part 1: TestNG with Code Coverage Howard Lewis Ship	Google Your Domain Objects With Hibernate Search Scott Leberknight
10:30 - 11:00 AM	MORNING BREAK				
11:00 - 12:30 PM	Agile Project Management (With Just a Bit About Mingle) Neal Ford	Enterprise Integration Patterns with Spring - Part II Mark Fisher	Web Application Hacking Roman Hustad	Guerilla Unit Testing Part 2: The Weird and Wonderful EasyMock Howard Lewis Ship	The 90-Minute Startup Michael Nygard
12:30 - 1:15 PM	LUNCH				
1:15 - 2:15 PM	EXPERT PANEL DISCUSSION				
2:15 - 3:45 PM	Introduction to JRuby Neal Ford	Basic JPA & JPAQL Pratik Patel	How to Catch Hackers: Security Auditing and Logging Roman Hustad	Introduction to Tapestry 5 Howard Lewis Ship	Failures Come In Flavors (part 1) Michael Nygard
3:45 - 4:00 PM	BREAK				
4:00 - 5:30 PM	Meta-programming JRuby for Fun & Profit Neal Ford	Enterprise JPA - Tips and Tricks for JEE5 Persistence Pratik Patel	How to Do a Security Code Review Roman Hustad	Pragmatic Patterns with Tapestry 5 IoC Howard Lewis Ship	Failures Come In Flavors (part 2) Michael Nygard

# Greater Atlanta Software Symposium

Atlanta Marriott Perimeter Center

May 16 - 18, 2008

<http://www.nofluffjuststuff.com/conference/atlanta/2008/05/index.html>  
(event schedule as of May 14, 2008)

## **Tactical Continuous Integration with Hudson by Andrew Glover**

This session will walk attendees through a series of iterations on a fictional Java project where an automated build system is created that facilitates compilation, testing, inspection, and deployment. This build system is then plugged into the Hudson CI server and as features are coded using Agile techniques like developer testing, attendees will ultimately see firsthand how a Continuous Integration process reduces risk and improves software quality.

## **Groovin' builds Gant get any easier by Andrew Glover**

There's no question that Ant is the de facto standard for building Java applications; however, even its creator has acknowledged an inherent limitation with Ant's expressiveness due to its reliance on XML. Recently, the popularity of Ruby and the Rails framework has brought to focus Ruby's de facto build platform: Rake. Rake's expressiveness comes from its reliance on Ruby itself to define a DSL for software assembly. While Rake's ultimate focus is Ruby, there are a number of interesting projects that utilize expressive DSLs for building Java including Gant, which uses Groovy as a DSL format and builds upon Ant's existing cornucopia of tasks.

## **Easy BDD with Groovy by Andrew Glover**

Behavior-driven development, or BDD, has attracted a lot of attention via RSpec in the Ruby community, but BDD's roots stem from JBehave, a Java based framework modeled off of the xUnit paradigm. But JBehave isn't the only framework available for Java developers-- with the advent of Groovy, new options are available for embracing BDD in the spirit of RSpec's innovative behavior based DSL.

## **Guerilla Unit Testing Part 1: TestNG with Code Coverage by Howard Lewis Ship**

Part one (of two) covers the TestNG unit testing framework, and shows how it integrates with tools such as Emma or Cobertura (for code coverage) and Selenium (for integration testing).

## **Guerilla Unit Testing Part 2: The Weird and Wonderful EasyMock by Howard Lewis Ship**

In part two (of two) we go in depth on EasyMock, the weird and wonderful tool for creating mock objects on the fly. We'll do a good bit of live coding as we examine how to use, tame and extend this powerful tool.

## **Introduction to Tapestry 5 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.

## **Pragmatic Patterns with Tapestry 5 IoC by Howard Lewis Ship**

Everyone likes the Gang of Four design patterns, but it's not always clear just how to make use of them in your day to day coding efforts. Hidden inside Tapestry 5 is an Inversion of Control (IoC) container that is structured around several common patterns (Chain of Command, Strategy, Facade and Filter Chain will be covered). This isn't academic navel-gazing ... this is about leveraging the common patterns so that you can write code you can easily test, and about creating frameworks and toolkits that can be easily extended. We'll see how Tapestry uses these patterns, and go from there into how you can apply the same techniques to your own projects, resulting in better, cleaner, more testable code.

## **Credit Card Software Development: Recognizing and Repaying Technical Debt by Jared Richardson**

Technical debt has long been recognized in technical circles for years, but convincing your manager to budget time to repay "technical debt" has always been problematic. Let's couch the term technical debt concept in language more familiar to our managers: credit card debt.

## **10 Tips for Getting Your Project Back on Track by Jared Richardson**

Software projects fail over and over for many of the same reasons. We'll look at some of the more avoidable problems and some solid ways to fix them, or avoid them in the first place.

### **Techniques 2008 by Jared Richardson**

There are a number of great techniques you can use across technologies and projects. Come hear some of my favorites and contribute a few of your own. We'll discuss topics from DRY to creating a zone defense for your product.

### **Build Teams, Not Products by Jared Richardson**

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

### **Agile Software Testing Strategies by Jared Richardson**

Creating and maintaining a solid automated test suite is critical to an Agile strategy, but often we're just told to "Do it." In this talk we'll look at several pragmatic strategies for creating and building your suite.

### **A Thorough Introduction To Groovy by Jeff Brown**

Groovy is an agile dynamic language for the Java platform. The language and its libraries bring many things to the table to ease the process of building applications for the Java platform. This session provides a detailed run through Groovy with lots of code samples to drive home the power of the language.

### **Grails - Agile Web 2.0 The Easy Way by Jeff Brown**

Grails is a full stack MVC framework for building web applications for the Java platform. Grails makes web application development both fun and easy. This session covers all of the fundamentals of building web applications with Grails.

### **Advanced Web Development With Grails by Jeff Brown**

Grails makes web application development both fun and easy. This session dives beyond the basics to cover advanced details of Grails that bring the really exciting features to your applications. **Prerequisite:** *Grails - Agile Web 2.0 The Easy Way*

### **Powerful Metaprogramming Techniques With Groovy by Jeff Brown**

Metaprogramming is a key component in building truly dynamic and flexible applications with Groovy. Groovy's metaprogramming capabilities bring great new possibilities to the table that would be very difficult or just plain impossible to write with Java alone. This session will demystify a lot of the magic that seems to be going on inside of a Groovy application. **Prerequisite:** *A Thorough Introduction To Groovy*

### **Agile Test Driven Development With Groovy by Jeff Brown**

Dynamic languages bring a lot of interesting elements to the table for teams interested in doing Test Driven Development (TDD). Groovy lends itself very well to TDD and this session demonstrates many features of the language and its libraries that help teams build more testable systems and build better tests.

### **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.

### **Java Memory, Performance and the Garbage Collector by Ken Sipe**

You are using Java, whew!!! No need to worry about memory, the garbage collector will handle that. Those who have had a memory issue in Java are not so naive any more. Often memory utilization and heap sizes are an after thought and are not recognized until the application is in production, often caused by application uptime, production request volume or production sets of data. When the OutOfMemory Error occurs, often the science of development seems to brake down and knobs are turned. First the (-mx) maximum heap space gets adjusted... More is better right. The next OutOfMemory, heads start scratching, code reviews start in earnest, and Google gets several new hits. Did you know that it is possible to get an OutOfMemory error without running out of heap space?

### **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?

### **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 criteria 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.

### **SOAs Challenges by Ken Sipe**

SOA... Is it hype? What's real... and what's not? What is the right abstraction level?

### **JMX and Spring: Manageability for Spring-based Applications by Ken Sipe**

This session describes management of Java resources using the Java Management Extensions JMX API. JMX provides a unified framework to instrument Java systems with monitoring and management capabilities.

### **Hacking - The Dark Arts by Ken Sipe**

A live Hacking demonstration exposing the tools and techniques used by Hackers.

### **Simplifying Enterprise Applications with Spring, Part 1 by Mark Fisher**

Developing enterprise applications isn't easy. You not only have to worry about constantly evolving business logic, but also need to address infrastructure concerns ranging from transaction management and security to manageability and integration with diverse external applications. Spring, the most popular lightweight enterprise application framework, comes to the rescue by simplifying the common needs of enterprise applications. This session (part 1 of 2) presents the core concepts of the Spring Framework.

### **Simplifying Enterprise Applications with Spring, Part 2 by Mark Fisher**

This session (part 2 of 2) will cover advanced concepts in the Spring framework. While the core concepts in the first session will get you started with Spring, the advanced concepts in this session will help you be more effective at developing Spring-based applications.

### **Configuring Spring with Annotations by Mark Fisher**

In this session, we will take a deep-dive into annotation-based dependency injection with Spring 2.5. You will learn how to combine annotation and XML formats, how to customize component scanning, and how to leverage Java 6 annotations within a Spring application. Since there is no "one size fits all" solution to application configuration, we will wrap up the discussion with general guidelines to consider when employing this approach.

### **Enterprise Integration Patterns with Spring - Part I by Mark Fisher**

In the first-part of this two-part workshop, Mark will focus on the essentials of Enterprise Integration with Spring. First, he will take a whirlwind tour of Spring's enterprise integration support libraries. Next, he will discuss the "big picture" of an event-driven architecture based on messaging with an overview of key enterprise integration patterns. Attendees will leave with a clear understanding of Spring's integration capabilities and an appreciation for the benefits of message-driven architecture, ready to put that into practice in Part II.

### **Enterprise Integration Patterns with Spring - Part II by Mark Fisher**

Building on Part I, Part II of this workshop will demo a series of messaging systems built on Spring. The samples will exercise event-driven scenarios involving distributed architectures with messaging and remoting. Each sample will highlight a variety of important enterprise integration patterns.

### **The 90-Minute Startup by Michael Nygard**

Cloud computing is taking the world by storm. Amazon's Web Services, EC2, and S3 provide completely virtual infrastructure, letting startup and existing companies create sites and web applications faster than ever before. In this session, Michael will use cloud computing to create and deploy a fully-functional web site. You will learn how to create and run your own virtual infrastructure in the clouds.

### **Failures Come In Flavors (part 1) by Michael Nygard**

The typical JEE application does not reach the fabled "five nines" of availability. Far from it. It's more like "double eights". Come see why enterprise applications and web sites are only serving users 88% of the time instead of 99.999%. Part 1 of 2

### **Failures Come In Flavors (part 2) by Michael Nygard**

What can we do about the dismal uptime of typical applications? We are asked to provide "five nines", but only reach 88%, on average. Come learn how to prevent the Stability Antipatterns from biting you. Apply these Stability Patterns to contain damage, recover from shocks, and survive disasters. Part 2 of 2

### **Designing for Ajax, part 1 by Nathaniel Schutta**

So you've convinced the boss that your new web application just has to have Ajax...but now what? With dozens of libraries making even the most blinkish of interactions trivial, how do you decide where to sprinkle the magic Ajax dust? This talk will give a plain old boring "web 1.0" an Ajax facelift with a focus on improving the user experience providing you with a game plan for introducing Ajax to your world.

### **Designing for Ajax, part 2 by Nathaniel Schutta**

We'll pick up where Part 1 left off working in even more advanced approaches such as offline support with Google Gears.

### **Dojo: Getting Started by Nathaniel Schutta**

So you want to do some Ajax and you've rightly concluded that you don't want to build your own library. After some thought, you've settled on using Dojo - but you're not sure how to get going. This talk will introduce Dojo and discuss several ways that Ajax can improve your new or existing application.

### **JavaScript: the Good, the Bad, and the Ugly by Nathaniel Schutta**

Thanks to Ajax, JavaScript is cool again and developers are taking a second look at this much maligned language. This session will give you an overview of this misunderstood language as well as opening your eyes to some of the excellent tools available to ease the pain of developing in this dynamic language.

### **Evolutionary SOA by Neal Ford**

This session demonstrates that "Agility" and "SOA" complement each other quite well. Just because SOA is buzz-word compliant doesn't mean that you should throw good practices out the window. This session demonstrates how you can apply the principles of agility to building highly complex distributed enterprises.

### **Test Driven Design by Neal Ford**

Most developers think that "TDD" stands for Test-driven Development. But it really should stand for "Test-driven Design". Rigorously using TDD makes your code much better in multiple ways.

### **Regular Expressions in Java by Neal Ford**

Regular expressions should be an integral part of every developer's toolbox, but most don't realize what an important topic it is. Regular expressions have existed for decades, but many developers don't understand how to take full advantage of this powerful mechanism, either through command line tools and editors or in their development.

### **Keynote: Ancient Philosophers & Blowhard Jamborees by Neal Ford**

It turns out that ancient philosophers knew a lot about software -- did you know that Plato defined object-oriented programming? This keynote applies old lessons to new problems and old problems to new lessons. It describes why SOA is so hard, and why people in your company make bone-headed decisions. What other keynote includes Rube Goldberg, Aristotle, Dave Thomas, and Chindia?

### **Code Metrics & Analysis for Agile Projects by Neal Ford**

What does code + methodology have to do with one another? Everything! Agile projects focus on delivering working code, and tools exist to allow you to verify some quality metrics for your code. This session is a survey of tools and metrics that allow you to determine the quality of your code and strategies to "wire it" into your agile project.

### **Agile Project Management (With Just a Bit About Mingle) by Neal Ford**

You can read books about Agile projects, but you must consult real-world experience to really understand the dynamics of agile project management. This session discusses agile management topics including estimation, project tracking, and useful metrics (and how to obtain them). And just a little about Mingle, the agile project tracking tool from ThoughtWorks.

### **Introduction to JRuby by Neal Ford**

This session describes JRuby, the 100% pure-Java implementation of the Ruby programming language. It covers the basics of programming with JRuby and examples of how to integrate it into existing Java projects.

### **Meta-programming JRuby for Fun & Profit by Neal Ford**

Ruby is the revenge of the Smalltalkers. Not since Smalltalk has a language had such powerful meta-programming facilities. While this may seem like a minor feature, it turns out that surgical meta-programming allows solutions to problems that are clearer, more concise, more maintainable, and take orders of magnitudes fewer lines of code.

### **Basic JPA & JPAQL by Pratik Patel**

Doing basic Object-to-Relational Mapping is fun and easy with JPA. Annotate your persistent classes, define a couple of configuration parameters, and you're off and running. This session starts with a basic object model and adds persistence using annotations. Learn how to do mappings for your object model for simple and complex relationships. Also learn how to map Java5 constructs like Enumerations. Unit testing with JPA can be tricky. Where do you use mock objects? How can I structure my unit tests to exercise my DAO's effectively? How do I unit test JPAQL? Do I need to enhance or can I use a LoadTimeWeaver in my unit tests? This presentation will show, using live code examples, how to effectively unit test JPA components so developers can have confidence in the code they build using JPA.

### **Enterprise JPA - Tips and Tricks for JEE5 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. A large enterprise application often will have several sub-projects that each contain their own JPA persistence unit. This opens up a number of questions around how to organize the persistence units and how the code between sub-projects should interoperate. Developers will gain insight into these issues and will see a couple of solutions using live code examples.

### **Developing Rich Internet Applications by Richard Monson-Haefel**

With literally hundreds of RIA products (e.g., Adobe Flash, Nexaweb, Backbase) and open source Ajax projects (e.g. Dojo, GWT, Prototype) to choose from. Picking the right RIA technology for the job requires months of research. Richard Monson-Haefel has been researching and writing about RIA alternatives for two years and has already done the research so you don't have to.

### **10 Things Every Software Architect Should Know by Richard Monson-Haefel**

An effective software architect understands that every application is different and requires unique choices regarding programming language, middleware, integration, data access, user interface design, etc. Richard Monson-Haefel has distilled knowledge from his own experience and from personal interviews with the World's best software architects to define 10 principles every software architect should know in order to be effective.

### **Understanding Open Source Licensing by Richard Monson-Haefel**

What does GPL, LGPL, MIT, Apache licenses, copy left, and dual licensing mean? Richard Monson-Haefel explains both the legal and technical implications of the major open source licenses in plain English. He explains when and how you can use open source in the enterprise and in the development of software products and how to protect your organization from abusing open source licensing.

### **What You Don't Know About Cryptography by Roman Hustad**

This session provides a gentle introduction to cryptography then covers the many subtle mistakes that even experienced developers make when writing cryptographic code.

### **Web Application Hacking by Roman Hustad**

See the hacker's toolbox in action as various web applications are ripped open by exploiting simple software bugs. Common problems such as Cross-Site Scripting (XSS) and SQL Injection will be demonstrated and explained, along with more subtle vulnerabilities including privilege escalation, data tampering, and Cross-Site Request Forgery.

#### **How to Catch Hackers: Security Auditing and Logging by Roman Hustad**

This session examines the code that developers must write in order to enable the detection of malicious activity and preservation of evidence after a security breach.

#### **How to Do a Security Code Review by Roman Hustad**

This session is a hand-on exercise in Java code review that will cover both manual and automated techniques. If you envision code review as a line-by-line slog through thousands of programs, you will be surprised to learn some effective techniques that reduce the tedium and increase your enjoyment of this activity (well, maybe not the enjoyment part). Familiar methods such as pair programming and peer reviews are a great place to start and will immediately increase the security of your code base.

#### **Introduction to Hibernate by Scott Leberknight**

This session introduces the Hibernate Object/Relational Mapping (ORM) framework, showing the basics of persisting Java objects to relational databases. No prior knowledge of Hibernate or ORM is assumed.

#### **Real World Hibernate Tips by Scott Leberknight**

Hibernate is a very powerful object/relational mapping framework. With the vast amount of power also comes the responsibility to choose which features of Hibernate to use and how to use them, as well as things to avoid. We'll look at some real world Hibernate tips and tricks in this session.

#### **Google Your Domain Objects With Hibernate Search by Scott Leberknight**

Hibernate is one of the pre-eminent object/relational mapping technologies, but the Hibernate Search project adds full-text search capabilities to an already extremely capable tool to allow you to Google your domain objects.