

# Southern Ohio Software Symposium

Cincinnati Marriott Northeast

August 17 - 19, 2007

<http://www.nofluffjuststuff.com/sh/2007-07-cincinnati>

(event schedule as of August 18, 2007)

Fri, Aug. 17, 2007					
	Montgomery	Loveland	Kenwood	Warren	Butler
12:00 - 1:00 PM	REGISTRATION				
1:00 - 1:15 PM	WELCOME				
1:15 - 2:45 PM	10 Ways to Improve Your Code Neal Ford	Groovy: Greasing the Wheels of Java Scott Davis	JavaServer Faces: A Whirlwind Tour David Geary	Gradual Agile: The Secret to Introducing Agile Practices Jared Richardson	Annotation Hammer Venkat Subramaniam
2:45 - 3:15 PM	BREAK				
3:15 - 4:45 PM	Implementing SOA Neal Ford	Groovy and Java: The Integration Story Scott Davis	Killer JavaScript Frameworks: Prototype, Scriptaculous, and Rico David Geary	Distributed Teams: Remote Agility Jared Richardson	Java 6 Features, what's in it for you? Venkat Subramaniam
4:45 - 5:00 PM	BREAK				
5:00 - 6:30 PM	Power Regular Expressions in Java Neal Ford	Real World Grails Scott Davis	Ajaxian Faces David Geary	Shippers Unite! Jared Richardson	OSGi: A Well Kept Secret Venkat Subramaniam
6:30 - 7:15 PM	DINNER				
7:15 - 8:00 PM	Keynote: by Neal Ford				

Sat, Aug. 18, 2007					
	Montgomery	Loveland	Kenwood	Warren	Butler
8:00 - 9:00 AM	BREAKFAST				
9:00 - 10:30 AM	The Zen of REST Scott Davis	RAD JSF with Seam, Facelets, and Ajax4jsf, Part One David Geary	Continuous Integration with Cruise Control Jared Richardson	Drooling with Groovy and Rules Venkat Subramaniam	Metrics-driven Agile Development Neal Ford
10:30 - 11:00 AM	BREAK				
11:00 - 12:30 PM	Mocking Web Services Scott Davis	RAD JSF with Seam, Facelets, and Ajax4jsf, Part Two David Geary	Developing Applications Using in Spring 2.0 Ryan Breidenbach	Building DSLs in Static and Dynamic Languages Neal Ford	Software Development Techniques Jared Richardson
12:30 - 1:30 PM	LUNCH				
1:30 - 3:00 PM	Domain Driven Design Venkat Subramaniam	The Google Web Toolkit, Part One David Geary	Developing Web Applications with Spring MVC Ryan Breidenbach	Introduction to JRuby Neal Ford	Agile Software Testing Strategies Jared Richardson
3:00 - 3:15 PM	BREAK				
3:15 - 4:45 PM	Spring into Groovy Venkat Subramaniam	The Google Web Toolkit, Part Two David Geary	Harnessing the Power of Maven Ryan Breidenbach	Rails for JRuby Neal Ford	Build Teams, Not Products Jared Richardson
4:45 - 5:30 PM	BIRDS OF A FEATHER SESSIONS				

Sun, Aug. 19, 2007					
	Montgomery	Loveland	Kenwood	Warren	Butler
8:00 - 9:00 AM	BREAKFAST				
9:00 - 10:30 AM	Debugging and Testing the Web Tier Neal Ford	Beginning Object-Relational Mapping with Hibernate Brian Sam-Bodden	Give it a REST Brian Sletten	get Fit Venkat Subramaniam	Groovy For Java Programmers Jeff Brown
10:30 - 11:00 AM	BREAK				
11:00 - 12:30 PM	Advanced Selenium Neal Ford	10 ways to use Hibernate effectively Brian Sam-Bodden	NetKernel : XML Processing for the 21st Century Brian Sletten	Practices of an Agile Developer Venkat Subramaniam	Test Driven Development With Groovy And Grails Jeff Brown
12:30 - 1:15 PM	LUNCH				
1:15 - 2:15 PM	EXPERT PANEL DISCUSSION				
2:15 - 3:45 PM	Productive Programmer: Acceleration, Focus, and Indirection Neal Ford	Beginning Drools - Rule Engines in Java Brian Sam-Bodden	Introduction To Agile Web Development With Grails Jeff Brown	Applied AOP Brian Sletten	Programming with Mock objects Venkat Subramaniam
3:45 - 4:00 PM	BREAK				
4:00 - 5:30 PM	Productive Programmer: Automation and Canonicity Neal Ford	Professional Java UI development with the Eclipse RPC Brian Sam-Bodden	Advanced View Techniques With Grails Jeff Brown	Introducing the Semantic Web Brian Sletten	FP for Java Programmers Venkat Subramaniam

# Southern Ohio Software Symposium

Cincinnati Marriott Northeast

August 17 - 19, 2007

<http://www.nofluffjuststuff.com/sh/2007-07-cincinnati>  
(event schedule as of August 18, 2007)

## **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.

## **Beginning Drools - Rule Engines in Java by Brian Sam-Bodden**

Software development is expensive, when business rules are hard-coded in your application's source code, changes and additions to those rules translate to wasted time and money. Good object-oriented, component-based approaches can alleviate the burden of keeping up with changes in the business world but they still require that expert knowledge of the changes be passed from the decision makers to the business analysts and finally to programmers that need to implement these changes. Business Rule Engines and Business Rule Languages are based on the basic premise of separation of concerns by empowering business domain experts to express the rules of business in a way that it is directly usable by applications.

## **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.

## **Give it a REST by Brian Sletten**

As developers, we sometimes get to make choices about the technologies we use, sometimes not. We base these decisions on personal experiences, recommendations from others and a general sense of where the industry is going. Web Services have been all the rage for several years now. We have been told time and again that we should be building systems around them; as an industry, we've never been more confused. Perhaps it is time to Give it a REST.

## **NetKernel : XML Processing for the 21st Century by Brian Sletten**

A wise man once said, "XML is like lye. It is very useful, but humans shouldn't touch it." If you've had to incorporate XML into your project by hand, you have probably been burned by getting too close. NetKernel turns this wisdom on its head and encourages you to use XML like the liquid data stream you want it to be. Imagine the simplicity of REST married to the power of Unix pipes. Come see how this open source / commercial product built on a compelling modern architecture can be used to create, manipulate and transform XML.

## **Applied AOP by Brian Sletten**

Most people new to Aspect-Oriented Programming (AOP) are fed up with separation of concerns zealots explaining how great their techniques are at dealing with... logging. Ok, you get it. Logging is a cross-cutting concern that can be appropriately modularized. What else does AOP have to offer? A lot, it turns out. This talk will give an introduction to the motivations of AOP as well as a series of concrete examples drawn from enterprise and client side Java. Come learn how AspectJ-flavored AOP can begin to benefit you immediately either in development or production environments. Learn how to enforce architectural policies, find Swing threading issues, reduce the invasiveness of the Observer design pattern or even improve the reusability of

your domain models. Now that Spring 2.0 provides support for AspectJ, the time has never been better to learn about these new (but backwards compatible) ways of thinking about building software.

### **Introducing the Semantic Web by Brian Sletten**

Just as the world is feeling comfortable with the Web, Tim Berners-Lee et al inform us that what we have seen so far is just the beginning. His original plans at CERN were larger and grander. The Semantic Web is the new vision of machine-processable documents and metadata to improve search, knowledge discovery and data integration and management. While there are many naysayers chiding such grand visions, there are also pragmatic and useful technologies emerging that can be applied today.

### **JavaServer Faces: A Whirlwind Tour by David Geary**

JavaServer Faces (JSF) has arrived. The standard Java-based web application framework based on Struts, JSF really took off in 2006. Embraced by developers, vendors, and open-source projects, JSF has hit its stride. If you haven't come up to speed on JSF basics, this is the place to start.

### **Killer JavaScript Frameworks: Prototype, Scriptaculous, and Rico by David Geary**

An introduction to the popular Prototype JavaScript framework, and two frameworks built on top of Prototype: Scriptaculous and Rico.

### **Ajaxian Faces by David Geary**

JavaServer Faces is a perfect platform for implementing Web 2.0 interfaces with Ajax. This session explores how you can use these two potent technologies--JSF and Ajax--together to create applications that look and behave like desktop applications but run in the browser.

### **RAD JSF with Seam, Facelets, and Ajax4jsf, Part One by David Geary**

In this session, see how you can get Ruby On Rails-like productivity on the Java side of the house with this compelling combination of technologies.

### **RAD JSF with Seam, Facelets, and Ajax4jsf, Part Two by David Geary**

A continuation of a 2-session presentation on Seam, Facelets, and Ajax4jsf.

### **The Google Web Toolkit, Part One by David Geary**

Developing highly interactive web applications, for the most part requires knowledge of a wide array of technologies: HTML, CSS, JavaScript, XMLHttpRequest, JSP, JSF, etc. With the Google Web Toolkit (GWT), Google turns that notion of development on its head. Instead, you implement Ajax applications by writing almost entirely in Java. You use an AWT-like API, which the Google compiler compiles to JavaScript that runs on the client.

### **The Google Web Toolkit, Part Two by David Geary**

The second part of a 2-session presentation on the Google Web Toolkit.

### **Gradual Agile: The Secret to Introducing Agile Practices by Jared Richardson**

Agile practices are popular because they work, but getting people to take that first step can be tricky.

### **Distributed Teams: Remote Agility by Jared Richardson**

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

### **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.

### **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.

### **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.

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

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

### **Groovy For Java Programmers by Jeff Brown**

Groovy is an agile dynamic language for the Java platform. Groovy has a Java like syntax along with many features inspired by languages like Python, Ruby and Smalltalk. This session covers a lot of ground including many interactive examples to hilite the powerful language features that make Groovy compelling. A lot of momentum is building in the Groovy and Grails communities right now and this session is aimed at Java developers who want to leverage the power of Groovy.

### **Test Driven Development With Groovy And Grails by Jeff Brown**

The value of Test Driven Development (TDD) has become widely accepted. The practice has extended beyond just XP teams. Good TDD practices yield high quality software and help teams maintain confidence in their software as complexity grows. The dynamic nature of Groovy makes TDD easy and fun. Groovy may be used to unit test not only Groovy code but other code as well. Testing Java code with Groovy is a snap. Learn to use the power of Groovy to test your systems.

### **Introduction To Agile Web Development With Grails by Jeff Brown**

Grails brings the powerful "coding by convention" paradigm to Groovy and Java. Grails is not just another flavor in the pool of web development frameworks for Java. Grails leverages the powerful dynamic features of Groovy while taking advantage of best of breed technologies like Hibernate, Spring, Sitemesh and Quartz to make web application development both fun and easy.

### **Advanced View Techniques With Grails by Jeff Brown**

Grails provides view technologies that offer 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. Grails provides simple mechanisms for leveraging the power of Ajax.

### **10 Ways to Improve Your Code by Neal Ford**

No one writes perfect code, and every developer eventually falls into a slump where they just crank out the same code day after day. This session illustrates 10 different ways to improve your code, covering sacred cows, good citizens, smells, and more.

### **Implementing SOA by Neal Ford**

This talk avoids SOA hype and gets to the meat of the matter: how do you implement a Service-Oriented Architecture, what are the technological pitfalls, how do you test it, and what traps should you avoid. No marketecture: just implementation details.

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

Regular expressions should be an integral part of every developer's toolbox, but most don't realize how important 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: Software "Engineering" & Polyglot Programming by Neal Ford**

About bridges, languages, engineering, polyglot programming, and the near future.

### **Metrics-driven Agile Development by Neal Ford**

Agile software development is a highly disciplined way to build software, and one of the side effects of this discipline is the ability to gather meaningful metrics. This session describes what makes agility perfectly suited to metrics gathering and what kinds of real-world metrics you can generate.

### **Building DSLs in Static and Dynamic Languages by Neal Ford**

This session discusses building Domain Specific Languages and DSL-style code in Java, Groovy, and Ruby. It discusses the different types of DSLs, details on how to implement them in Java, Groovy, and Ruby, and example problem domains where DSLs make sense.

### **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.

### **Rails for JRuby by Neal Ford**

This session explains all the hype surrounding Ruby on Rails, in a context familiar to Java developers. It covers convention over configuration, ActiveRecord, controllers, views, Ajax, scaffolding, testing, and deployment...on the JVM, using JRuby.

### **Debugging and Testing the Web Tier by Neal Ford**

As our applications have spilled from the server across the wire to the web tier, we increasingly must debug and test in the browser. This session covers debugging and testing tools for clients, JavaScript, and Ajax.

### **Advanced Selenium by Neal Ford**

This session discusses advanced Selenium techniques for testing web applications. It discusses techniques for both TestRunner and Remote Control Selenium, including data driven tests, creating branch points, testing Ajax applications, creating flexible tests, integration with continuous integration, and tons more.

### **Productive Programmer: Acceleration, Focus, and Indirection by Neal Ford**

This session discusses how to use the Productive Programmer principles of acceleration, focus, and indirection to become a more productive programmer. This session describes these principles, but the primary focus of this session is demonstration of these principles with real-world examples.

### **Productive Programmer: Automation and Canonicity by Neal Ford**

This session discusses how to use the Productive Programmer principles of automation and canonicity to become a more productive programmer. This session describes these principles, but the primary focus of this session is demonstration of these principles with real-world examples.

### **Developing Applications Using in Spring 2.0 by Ryan Breidenbach**

In 2006, Spring 2.0 was released and Spring became ubiquitous in the enterprise Java space. So if you haven't taken a peek at Spring yet, what are you waiting for?!? If the answer is "This session," you have come to the right place. This session will familiarize you with the benefits Spring provides and why they have made the framework so successful. In addition, we will focus on the efforts that went into Spring 2.0 to make these benefits much more developer friendly. By the end of this session, you should have a grasp of how Spring 2.0 simplifies the utilization of Spring's core features such as bean wiring, AOP, and the simplification of using enterprise Java. If your head is not spinning too much.

### **Developing Web Applications with Spring MVC by Ryan Breidenbach**

Although Spring's rise to ubiquity was driven by its IoC and AOP services, it offers so much more. One of the most powerful of these services is Spring's MVC framework. If you are familiar with other Java web frameworks such as Struts or WebWork, you will recognize that Spring MVC is a similar request-response web framework driven by a Front Controller, in this case the DispatcherServlet. But Spring also provides considerable functionality out of the box. One of the beauties of the Spring MVC is that, like the rest of Spring, all of its components are quite modular. This allows you to mix and match different implementations of the various MVC components as you see fit for your application. In addition, Spring MVC provides rich capabilities for data binding. This allows you to transparently bind form parameters to your application objects without having to create any intermediate objects. Spring's validation features allow you to capture

both binding and application errors. And with Spring 2.0's much improved JSP tag library, both creating forms and displaying validation errors is quite simple.

### **Harnessing the Power of Maven by Ryan Breidenbach**

2006 appeared to be the year that Maven achieved the momentum it needed to overtake Ant as the build tool of choice for Java developers. A lot of that has to do with the vastly improved Maven 2. But a lot of it has to do with the simplicity, organization and power that Maven brings to projects. The session will bring developers new to Maven with everything it has to offer. This includes creating your very first Maven project, learning the significance of the POM file, how to let Maven and its repositories manage your dependencies, and how to let Maven report of the health of your own projects. And for the Ant users in the audience, you will get to see a side-by-side comparison of the two build tools' build philosophies.

### **Groovy: Greasing the Wheels of Java by Scott Davis**

This is the year of the dynamic scripting language. Ruby (and Rails) has won the hearts and minds of many independent software developers. JavaScript is experiencing a renaissance thanks to the wild success of AJAX and websites like Google Maps. And Groovy (JSR-241) brings the same level of excitement and "scripting goodness" to the Java platform.

### **Groovy and Java: The Integration Story by Scott Davis**

I'm attracted to Groovy because of its spirit of inclusiveness. Because it extends my platform of choice, not replaces it -- include a single JAR in your classpath and you are Groovy-enabled. Because it offers full bidirectional integration with Java. Because it offers a nearly flat learning curve for experienced Java developers. Come see how you can use Groovy to augment your existing Java codebase.

### **Real World Grails by Scott Davis**

Scott Davis is the Editor in Chief of aboutGroovy.com. The website, in addition to being, umm, about Groovy, is implemented in Grails. This talk shows you how to get started with Grails, but also talks about the experience of using it in a live, production web site.

### **The Zen of REST by Scott Davis**

Google quietly deprecated their SOAP search API at the end of 2006. While this doesn't mean that you should abandon SOAP, it does reflect a growing trend towards simpler dialects of web services. Google joins a number of popular websites (Yahoo, Flickr, YouTube, del.icio.us) that offer all of the benefits of web services without all of the complexity of SOAP.

### **Mocking Web Services by Scott Davis**

In this talk, we'll survey the web services exposed by leading websites (Google, Yahoo, Amazon, eBay) and discuss how they can be easily mocked up for testing purposes and to aid offline development. You'll see working examples of RESTful, SOAP, and JSON web services, as well as strategies for unit and functional testing your asynchronous, service-oriented architecture.

### **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.

### **Java 6 Features, what's in it for you? by Venkat Subramaniam**

What benefit do new Java 6 features offer you. Are there issues with using these features. The objective of this presentation is not simply to introduce you to the features, but to the effective use of these as well.

### **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.

### **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.

### **Spring into Groovy by Venkat Subramaniam**

What do you get when you mix an agile, object-oriented, dynamic language with a lightweight, flexible, and extensible framework? You get a Groovier Spring. Spring allows you to develop using Groovy as much as Java. Groovy brings some neat concepts to the Java Platform that is hard to realize directly through the Java language. Using these capabilities can lead to elegant and easier Spring development.

### **get Fit by Venkat Subramaniam**

Unit testing tells you, the programmer, that your code (and the change) meets your expectations. How do you know if you are meeting your customers' expectations? Agile development is all about feedback and doing what's relevant to the customers, isn't it? Framework for Integration testing or Fit helps you to automate tests for customer expectations.

### **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 speaker will share with you practices in a number of areas including coding, developer attitude, debugging, and feedback. The discussions are based on the book with the same title as the talk.

### **Programming with Mock objects by Venkat Subramaniam**

You are convinced that Test Driven Development is good for you and your project. You realize the benefits it has to offer. What's holding you back? All the code and components that your code so heavily depends on is most likely making you wonder if TDD is really for you. We will start out by looking at dependency and dependency inversion. Then we will discuss how mock objects can help separate our code from its dependencies.

### **FP for Java Programmers by Venkat Subramaniam**

Most interest around Functional Programming (FP) has been academic until recently. Recent commercial languages are beginning to exploit FP features. Knowing more about FP will not only help us make better use of these features, but to exploit those. In this session we will take a close look at FP.