

# Mid South Software Symposium

Memphis Hilton  
April 23 - 25, 2010

The No Fluff Just Stuff Java Symposium Series is proud to announce the return of the Mid South Software Symposium on April 23 - 25, 2010. MSSS2010 will be held at the Memphis Hilton.

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 MSSS2010  
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 3/29/10 after \$950

**Excellent Group Discounts Available** - bring your entire development team to the show - no travel required!! Rate good thru 3/29/10

## **Registration Fees**

Attendees	Before Mar. 29, 2010	After Mar. 29, 2010
5-9	\$750	\$850
10-14	\$725	\$825
15-24	\$700	\$800
25+	\$675	\$775

Go to <http://www.nofluffjuststuff.com/conference/memphis/2010/04/home> and register today!

# Mid South Software Symposium

## -Session Schedule-

(event schedule as of April 21, 2010)

### Friday, Apr. 23

12:00 - 1:00 PM : REGISTRATION

1:00 - 1:15 PM : WELCOME

1:15 - 2:45 PM - Sessions

#### **Session #1 : 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.

#### **Session #2 : 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.

#### **Session #3 : 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.

#### **Session #4 : 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.

2:45 - 3:15 PM : BREAK

3:15 - 4:45 PM - Sessions

#### **Session #5 : 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*

#### **Session #6 : 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.

#### **Session #7 : 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)*

#### **Session #8 : 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.

4:45 - 5:00 PM : BREAK

# Mid South Software Symposium

## -Session Schedule-

(event schedule as of April 21, 2010)

5:00 - 6:30 PM - Sessions

### Session #9 : 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*

### Session #10 : 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*

### Session #11 : 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.

### Session #12 : 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 of 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.

6:30 - 7:15 PM : DINNER

7:15 - 8:00 PM : KEYNOTE: Smithing in the 21st Century by Neal Ford (Southeast BC)

## Saturday, Apr. 24

8:00 - 9:00 AM : BREAKFAST

9:00 - 10:30 AM - Sessions

### Session #13 : 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.

### Session #14 : 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.

### Session #15 : 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.

### Session #16 : SPARQL: Querying the Data Web by Brian Sletten

The human-friendly Web is about nicely-formatted, accessible content for users to browse. There is an emerging Data Web that relies 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 the practical and accessible aspects of employing these ideas on the Web and in the Enterprise. **Prerequisite:** *The Semantic Web: The Future, Now and Rich Web Pages : Publishing Semantic Content with GRDDL and RDFa would both be helpful but are not required*

10:30 - 11:00 AM : BREAK

11:00 - 12:30 PM - Sessions

### Session #17 : 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 sessions 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

# Mid South Software Symposium

## -Session Schedule-

(event schedule as of April 21, 2010)

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.

### **Session #18 : 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)*

### **Session #19 : Enterprise Security API library from OWASP by Ken Sipe**

When it comes to cross cutting software concerns, we expect to have or build a common framework or utility to solve this problem.

This concept is represented well in the Java world with the `loj4j` framework, which abstracts the concern of logging, where it logs and the management of logging. The one cross cutting software concern which seems for most applications to be piecemeal is that of security. Security concerns include certification generation, SSL, protection from SQL Injection, protection from XSS, user authorization and authentication. Each of these separate concerns tend to have their own standards and libraries and leaves it as an exercise for the development team to cobble together a solution which includes multiple needs.... until now... Enterprise Security API library from OWASP.

### **Session #20 : 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*

12:30 - 1:30 PM : LUNCH

1:30 - 3:00 PM - Sessions

### **Session #21 : 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.

### **Session #22 : 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.

### **Session #23 : Enter The Gradle by Ken Sipe**

In the Java build space, first there was ANT, which provided a reliable way to build without an IDE. Then there was Maven, which provided standardization in build life cycles and dependency management. Now... Enter the Gradle, which provides convention over configuration approach to the build process and an approach at building that isn't based XML. **Prerequisite:** *Some Groovy helpful*

### **Session #24 : Code Craft by Nathaniel Schutta**

Despite what some developers think, we spend a lot more of our time reading code, code that was often written by someone that isn't around anymore. How do we deal with this common scenario without resorting to burning our predecessor in effigy? Better, how can we write code in such a way that our successors will heap effusive praise upon us at the mere mention of our name? During this talk, we'll read actual code discussing ways it could be improved. As we work through real examples, we'll explore the importance of patterns, principles like SOLID and SLAP and essential practices like unit testing and continuous integration.

3:00 - 3:15 PM : BREAK

3:15 - 4:45 PM - Sessions

# Mid South Software Symposium

## -Session Schedule-

(event schedule as of April 21, 2010)

### **Session #25 : Groovy and Grails in the Enterprise by Pratik Patel**

Dynamic languages running on the Java Virtual Machine are starting to gain traction for software development, specifically for large enterprise projects. This session explores obstacles to introducing dynamic languages into the enterprise, example applications that can ease the way, and issues surrounding integrating a dynamic language to Java projects. Using several code examples that demonstrate the power of using a dynamic language like Jruby or Groovy, attendees will gain insight into how dynamic languages are making in-roads to the enterprise. This session focuses on non-GUI related usages whereas most people think of dynamic languages for Web development. The target audience for this session is enterprise developers and enterprise architects.

### **Session #26 : MongoDB by Ted Neward**

MongoDB is designed for problems without heavy transactional requirements that aren't easily solved by traditional RDBMSs, including problems which require the database to span many servers. Like other document-oriented database systems such as CouchDB, MongoDB is not a relational database management system. The database manages collections of JSON-like documents which are stored in a binary format referred to as BSON.

### **Session #27 : Spring 3 into REST by Ken Sipe**

REST as an architectural approach is greatly simplified through the selection of framework or tool to help with the tedious and repetitive template style that it demands. Until recently, some of the best approaches were through frameworks that required the use of languages other than Java, such as Rails or Grails. In the Java space, the choices were limited. The newly released Spring 3 changes that. One of the most significant changes in Spring 3 is its support for REST, which includes client as well as server support.

**Prerequisite:** Java 5

### **Session #28 : 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.

4:45 - 5:45 PM : BIRDS OF A FEATHER SESSION

## **Sunday, Apr. 25**

8:00 - 9:00 AM : BREAKFAST

9:00 - 10:30 AM - Sessions

### **Session #29 : 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?

### **Session #30 : 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 its 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.

### **Session #31 : 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.

### **Session #32 : Introduction to NetKernel : Software for the 21st Century by Brian Sletten**

Imagine the simplicity of REST married to the power of Unix pipes with the benefits of a loosely-coupled, logically-layered architecture. If that is hard to imagine, it may be because the architectures available to you today are convoluted accretions of mismatched technologies, languages, abstractions and data models. NetKernel is a disruptive technology that changes the game. It has been quietly gaining mind share in the past several years; people who are exposed to it don't want to go back to the tired and blue conventions of J2EE and .NET. Not only does it make building the kinds of systems you are building today easier, it does it more efficiently, with less code and a far more scalable runway to allow you to take advantage of the emerging multi-core, multi-CPU hardware that is coming our way. Come see how this open source / commercial product can change the way you think about building software.

# Mid South Software Symposium

## -Session Schedule-

(event schedule as of April 21, 2010)

10:30 - 11:00 AM : MORNING BREAK

11:00 - 12:30 PM - Sessions

### **Session #33 : Debugging your Production JVM by Ken Sipe**

So your server is having issues? memory? Connections? Limited response? Is the first solution to bounce the server? Perhaps change some VM flags or add some logging? In today's Java 6 world, with its superior runtime monitoring and management capabilities the reasons to bounce the server have been greatly reduced.

### **Session #34 : Agile UI by Nathaniel Schutta**

Some developers assume that agility and usability are mutually exclusive - in reality, they are extremely complimentary; if you squint, you might have a hard time telling the difference between agile practices and good user interface design. This usability talk is aimed squarely at developers giving you the tools you need to develop UIs that won't make your users yack. We'll discuss the importance of observation, personas, paper prototyping, usability testing and the importance of good moderators. In addition, we'll map the various aspects of user interface design to a typical agile iteration.

### **Session #35 : 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.

### **Session #36 : Tracer Bullet Development with OSGi by Matt Stine**

Tracer Bullet Development (TBD) is a technique that allows you to prove out the proposed architecture of your system by firing a "tracer bullet" through a vertical slice of your system that exercises all of its horizontal components. It has multiple benefits, including encapsulation, decoupled code, parallel code development, and more. OSGi is a specification for a dynamic module system for Java with multiple open source implementations. It allows you to modularize your system into "bundles" which essentially firewall their own classloader space. Objects running within a bundle can only see types that they explicitly import and only expose types that they explicitly export. They interact with other bundles by exposing and consuming services which are registered under a public interface. It seems that Tracer Bullet Development and OSGi are a match made in heaven!

12:30 - 1:15 PM : LUNCH

1:15 - 2:15 PM : EXPERT PANEL DISCUSSION

2:15 - 3:45 PM - Sessions

### **Session #37 : Architecture and Scaling by Ken Sipe**

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

### **Session #38 : 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.

### **Session #39 : 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.

### **Session #40 : Polyglot OSGi by Matt Stine**

One of the greatest benefits of OSGi is its firewall-esque encapsulation of implementation details. The only traffic that gets in or out is the traffic that you explicitly specify; otherwise, all bets are off. The aspiring polyglot can bring in the right tool for the right job by hiding it behind OSGi services as an "implementation detail," provided that only Java language types are exported.

3:45 - 4:00 PM : BREAK

4:00 - 5:30 PM - Sessions

### **Session #41 : So you want to be an Architect by Ken Sipe**

This session is a quick look at all aspects of being a corporate software architect. Whether you are a developer looking to move into the role of architect, needing to have an understanding of what is expected or already in the role of software architect looking for new and interesting ideas, this session is for you.

### **Session #42 : Hacking Your Brain for Fun and Profit by Nathaniel Schutta**

The single most important tool in any developer's 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

# Mid South Software Symposium

## -Session Schedule-

(event schedule as of April 21, 2010)

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.

### **Session #43 : 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

### **Session #44 : The Agile Guerilla by Matt Stine**

So you discovered agile software development this weekend. You've finally found the tools that you're going to use to fix your team. Do you rush in to work Monday morning with a slide deck in one hand and a baseball bat in the other, ready to bludgeon the first person who checks in untested code? How do you think that's going to work out for you? I can tell you from personal experience that it doesn't play out too well. There is a better way.