

Great Lakes Software Symposium 2006

Wyndham Northwest Chicago - Itasca, IL

November 17 - 19, 2006

(session listing as of 11/19/2006)

The No Fluff Just Stuff Software Symposium 2006 tour is pleased to announce the Great Lakes Software Symposium coming to Itasca on November 17 - 19, 2006. You will have the opportunity to attend the best technically focused Java / Open Source event anywhere. We make this claim based on the following:

- 1) **Excellent Speakers with unparalleled access**
- 2) **Limited Attendance = 250 Registrants Max**
- 3) **No Vendors, No Sales Pitches, No Marketecture**
- 4) **Unmatched Value - less than 1/3 of the cost of a national conference**
- 5) **Since 2002, we have delivered over 75 conferences throughout North America**

The No Fluff Just Stuff Software Symposium Series caters to individual developers, development teams, project managers, architects and independent consultants. The Great Lakes Software Symposium will offer 5 concurrent sessions over three days with over 55 sessions to choose from. The following topics will be featured:

- 1) Architecture
- 2) Client side Java
- 3) ServerSide Java
- 4) Core Java
- 5) XML / Web Services
- 6) Groovy

Registration Fees

Attendees	Before 10/30/2006	After 10/30/2006
1-4	\$725	\$825
5-9	\$650	\$725
10-14	\$625	\$700
15-24	\$600	\$675
25+	\$575	\$650

The Registration Fee includes the following:

- 1) All Access Pass to the three day symposium
- 2) Handouts from all sessions attended w/binder
- 3) CD with all presentational content @ registration
- 4) Great NFJS Swag
- 5) Opportunity to win an iPod everyday during the symposium raffle

Go to <http://www.nofluffjuststuff.com> for more details.

Questions/Comments: Contact Jay Zimmerman: jzimmerman@nofluffjuststuff.com or (303)469-0486.

Great Lakes Software Symposium 2006

Wyndham Northwest Chicago - Itasca, IL

November 17 - 19, 2006

(session listing as of 11/19/2006)

Introduction to the Spring Framework by Ben Hale

In this day and age, it's pretty hard to be a Java developer and not have heard of the Spring Framework. The question is, what exactly is Spring? Well, you don't have to look any further. This session aims to answer the questions of what Spring is as well as why you might want to use it and what it can do for you.

Spring Web Flow Jumpstart by Ben Hale

One of the new projects under the Spring umbrella is Spring Web Flow. Spring Web Flow is a framework for declaratively modeling web application user interactions. In this session, you'll get a quick primer on the business case for using Spring Web Flow and then we'll jump right in to a live coding session showing its use.

The Spring Experience (in 90 minutes) by Ben Hale

In this interactive session Ben walks you through the experience of building a simple Spring-powered application from the ground up.

Improving Java code quality with code auditing tools by Brian Goetz

Does your program have bugs, despite unit tests, integration tests, and code reviews? You bet. Fortunately, there are some new code auditing tools that can help spot some bugs missed by other approaches.

Introduction to Java threads by Brian Goetz

The Java language included support for threads and concurrency from day 1, but writing correct multithreaded programs is not easy. This session will cover the how and why of using threads in Java.

Java Performance Myths by Brian Goetz

Performance myths about the Java platform abound, from the general "Java is slow", to the more specific "reflection is slow", "allocation is slow", "synchronization is slow", "garbage collection is slow", etc. Many of these myths have their root in fact (in JDK 1.0, everything was slow); today, not only are many of these statements not true, but Java performance has surpassed that of C in many areas, such as memory management.

Structuring concurrent applications in JDK 5.0 by Brian Goetz

JDK 5.0 is a huge step forward in developing concurrent Java classes and applications, providing a rich set of high-level concurrency building blocks.

The Java Memory Model by Brian Goetz

What's the worst thing that can happen when you fail to synchronize in a concurrent Java program? Its probably worse than you think -- modern shared-memory processors can do some pretty weird things when left to their own devices.

Applied Design Patterns by Brian Sletten

Just about every modern software developer has a copy of the Gang of Four "Design Patterns" book sitting on a shelf; many of them have actually read it. The dark secret of the patterns community is that there is often a large gulf between whiteboard simplicity and real-world complexity. Language choice plays a part in the design (and even importance) of patterns. The situation is made even more confusing by the fact that many of the core patterns have now been "voted off the island" for one reason or another. This talk will give a pragmatic overview of the motivations behind design patterns and will focus on applying a handful of the GOF patterns to example scenarios in Java, Ruby and C#. A quick introduction to the role AOP plays in changing the patterns landscape will also be covered.

Applied REST by Brian Sletten

REST sounds like such a simple thing. But, what is it really? How do you convince your boss to let you try it when she has been sold on the equation SOAP = SOA + P(rofit)? How do you go about building, deploying, publishing and orchestrating web services without the (Un)Holy Trinity of SOAP, WSDL and UDDI?

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.

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.

Effective Teams: The dirty little secret by Bruce Tate

Most conferences will try to tell you that the secret to good software development lies with a process, or a technology, or an architecture. Here's a dirty little secret. You can build working software with an outdated two tier architecture, a waterfall process and COBOL. How? By building a great team.

Java/Ruby Integration with JRuby and ReST by Bruce Tate

You can have rapid web development with Rails without losing access to your critical Java code. With the explosion of the Ruby programming language, more developers will need a strategy for letting Java and Ruby interoperate. This session explores two strategies: JRuby and Rails-based web services.

Three Technologies to Watch by Bruce Tate

The state of the art is progressing rapidly, and dynamic languages are driving the revolution. Find out about these topics that will be central to programming. We'll discuss continuation servers, metaprogramming frameworks and functional languages.

What's New in Spring 2 by Bruce Tate

In this session, we'll review the new features of Spring 2.0. If you've been using Spring 1.x, you'll want to hear about the improvements.

Where Agile meets Argyle: New processes in established companies by Bruce Tate

Agile programming is a collection of core principles and techniques that allow software developers to create lighter, more responsive applications, and to have fun doing it. Many established organizations are either openly or sub-consciously hostile to many of the principles of Agile development.

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.

JSF: State of the Art by David Geary

In 2005, JSF hit its stride, as evidenced from overwhelming support from both vendors and the open-source community. JSF 1.0 had plenty of holes, but open-source projects have arisen to address those needs. This session takes a look at three of those projects:

- Tomahawk (MyFaces component library)
- Facelets
- Seam

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 2005. Embraced by developers, vendors, and open-source projects, JSF has started to hit its stride. If you haven't come up to speed on JSF basics, this is the place to start.

The Google Web Toolkit 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.

Automating Business Value with FIT and FitNesse by David Hussman

The presentation will briefly discuss stories, the origin and authoring of story tests, and a demonstration of how FIT and FitNesse (FIT living within a Wiki) can be used to automate acceptance tests.

Creating, Telling, and Tracking User Stories by David Hussman

The participants of this session will become agile customers and product owners, using personas to create stories for a sample product development.

Losing Battles and Winning Wars: Adopting Agile by David Hussman

Adopting agile is different for each company, but most companies will go through some amount of change during the adoption of agile.

Ready, Set, Agile? by David Hussman

As with many methodologies, moving agile into an organization poses larger challenges. Before jumping in, it helps to ask a few questions before "racing toward agility". This session will provide 3 tactical steps that can help your adoption of agile.

#Show Me the Numbers# - Agile Planning Tools and Techniques by David Hussman

This session will focus on tools and techniques for tracking an agile project plan from creation to project completion.

Extreme Web Caching by Jason Hunter

Web Caching is very important for high traffic, high performance web site but few people know all the professional-level strategies. In this talk I'll share some of the tricks of the trade, including advanced tips from Yahoo's Mike Radwin. We'll start with the basics: using client-side caches, conditional get, and proxies. Then we'll talk about more advanced features: how best to handle personalized content, setting up an image caching server, using a cookie-free domain for static content, and using randomization in URLs for accurate hit metering or sensitive content.

Greasemonkey: Ajax For One by Jason Hunter

Greasemonkey is a Firefox extension with a hook to let you run your own JavaScript after each page loads. People have written hundreds of Greasemonkey scripts, each using Ajax techniques, to do everything from making a static page dynamic to changing a page's style to integrating comparison pricing on Amazon pages and giving Google search results easy keyboard shortcuts. Greasemonkey lets you take back control of the web browsing experience -- and teaches you Ajax while you're at it. In this talk we'll look at about a dozen useful Greasemonkey scripts. Each script provides a benefit in its own right, but like good engineers we'll dissect them to see what makes them tick.

Java Metadata by Jason Hunter

Java's new Metadata facility introduced in J2SE 5.0 defines a way to attach decorations to classes, fields, methods, and even packages that can be extracted by the compiler or runtime tools to provide advanced functionality. Think of metadata as an extended `@deprecated` flag, or think of XDoclet++. In this tutorial session you'll learn how Metadata fits in the Java platform (and how it compares to the C# platform). We'll cover how to use the metadata attributes provided in the core J2SE libraries and how to write your own. We'll also show a bit of what's coming in JSR-181, tasked to define standard metadata attributes for web services.

New Features in Java 5 by Jason Hunter

The new Java 5 release introduces a number of significant Java language enhancements: generics, typesafe enums, autoboxing, an enhanced "for" loop, a static import facility, and a general-purpose metadata facility. This talk gives an overview of the changes and helps you understand what all the funny new syntax means.

Open Source from the Inside by Jason Hunter

Open source isn't about a license, it's about human interaction and individual motivation. I've seen open source from all sides. I've been an individual contributor and a project leader. I've worked on commercial and open source efforts, and have both helped commercial projects go open and designed ways for open projects to absorb commercial codebases. I've been on the front lines in the Apache/Sun negotiations on open source Java that ended on the JavaOne keynote stage with Scott McNealy. In this talk, I'd like to share my favorite stories in and around open source and the lessons they teach us.

Foundations of Ajax by Nathaniel Schutta

Seemingly overnight, Ajax has gone from an obscure acronym to, well, having conferences devoted to it. People are often surprised when they learn that Ajax isn't really anything new - so if it's yesterday's technology, why all the hype? This talk will examine the course of events that led up to the current love affair

with richer client applications. We'll talk examine the technologies that power an Ajax application and discuss how to work with them - and then we'll examine a number of frameworks that will do much of the heavy lifting for us!

Pragmatic Usability (aka, Software Engineer's Guide to Usability) by Nathaniel Schutta

While some companies have the luxury of a full time usability team, most of us have to make do on our own. Sure, it might be easier (and more comfortable) to focus on all the hip back end goodness, but if your user interface makes users yack, your product is doomed.

Advanced Enterprise Debugging Techniques by Neal Ford

This session discusses techniques and tools for debugging enterprise applications (without using `System.out.println()`!)

Clean Up Your Code: 10 Java Coding Tricks, Techniques, and Philosophies by Neal Ford

This session delivers 10 techniques for improving your code, whether you are freshly graduated or a grizzled veteran.

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.

SOA: Next Wave of Enterprise Development or Return of the Son of CORBA? by Neal Ford

Is Service Oriented Architecture the next wave of distributed computing or just the same old crap in a shiny new package? This session provides an overview of what most people agree is the definition of SOA. I talk about SOA, ESB, CORBA, your MOM, and a bunch of other acronyms.

Testing with Selenium by Neal Ford

This session describes the use and workings of Selenium, the open source web user interface testing tool.

The Productive Programmer by Neal Ford

This session shows you how to become a more productive programmer every day by using tools that you didn't know you already had.

Web Application Security Vulnerabilities by Neal Ford

This session highlights common mistakes made by web programmers, stating the problems and avoidance techniques.

Agile Enterprise Architecture: The role of the architect by Pete Behrens

Are you overrunning your architectural runway? Many companies struggle with their ability to retain their architectural integrity when they transition to agile methods. Emergent Architecture (the other EA) can lead to cowboy coding and ad-hoc design decisions that emerge into a poor overall architecture. Enterprise Architecture (EA) has been a tried and true approach to address these architectural needs throughout the organization, yet this approach often leads to a heavy-handed, document-rich, control-oriented culture lacking ability to keep pace with today's dynamic business environment. Attempting to integrate an agile process with an Enterprise Architecture approach can be like mixing oil and water - they just don't work together. This session evaluates alternatives in balancing Agility and EA and proposes an architectural approach to build an Agile Enterprise Architecture into your organization.

Agile Pattern: The Product Increment by Pete Behrens

Agile methods like Scrum and XP expect the team to create a potentially shippable product increment out of complex requirements and unpredictable technology every 1 to 4 weeks. Yet, we have complex and legacy products with deep features which make this extremely difficult. This session will drill into what exactly a potentially shippable product increment is, why it is so hard to create and the Top 10 things we can do to build them more effectively.

Agile Tooling: Team to Enterprise by Pete Behrens

"YAGNI (You Ain't Gonna Need It)" and "Doing the simplest thing possible" are mantras of agile development. A white board, sticky notes, and flip chart paper are by far the best tools for individual teams. However, when coordinating work across 10 - 50 teams across 12 time zones, more tooling is required.

Learn how agile enterprises are leveraging tooling to manage their portfolios, projects and products.

Domain Driven Design with AOP and DI by Ramnivas Laddad

Domain Driven Design (DDD) suggests dealing with complex software system using a domain model and preserving the model in implementation. Since domain model entities have rich behavior, so should their software implementation artifacts. A direct mapping between domain model and software artifacts create simple-to-understand, inexpensive-to-implement, and easy-to-evolve systems. While the idea behind DDD isn't new and the value is easily understood, many implementations do not adhere to its principles. This disconnection may be due to many obstacles in implementing it. Combining Dependency Injection (DI) with a full-fledged aspect-oriented programming (AOP) system such as AspectJ help overcome many obstacles.

Spring AOP in Depth by Ramnivas Laddad

Support for aspect-oriented programming is an important part of the Spring framework. It is the AOP support that allows keeping implementation of functionality such as transaction management and security out of your POJOs. While many developers only use aspects provided with Spring, once you understand how it all works, you can make a better use of those aspects, extend them, and write brand new aspects.

Testing Strategies for Web Applications by Ramnivas Laddad

Ever wondered if you can automate testing of your web application, but couldn't produce a satisfactory solution? If so, this is the session for you! Attend this session to understand the alternatives you have for unit and functional testing of web applications.

The State of AOP by Ramnivas Laddad

A lot is happening in the field of Aspect-oriented programming (AOP). AspectJ and AspectWerkz, the two leading AOP implementations, have merged, bringing in their respective strengths. The merged version, AspectJ 5, adds many new features aimed at simplifying writing and deploying aspects. The new features include an annotation-based and XML-based syntax to define aspects, support for new Java 5 concepts, and load-time weaving. The tools support for AOP continues to improve, as well. Further, the most popular IOC framework, Spring, enables integrating aspects written in AspectJ. There is also serious discussion and preliminary work going on to support AOP right into the VM itself. All in all, there is a lot to learn about the changes in the exciting field of AOP. This session is designed to help you get up to date with all these changes.

Easing into Agile by Scott Davis

How do you get started with an Agile development methodology? Everyone has been talking about eXtreme Programming for years, but how do you get it introduced to your team? Many times, you're not simply transitioning from one methodology to another -- you're introducing a methodology for the first time. Adding structure to a previously unstructured endeavor. Adding a touch of discipline where programmers once roamed free.

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.

Holistic Testing by Scott Davis

Mark Twain once said, "Everyone talks about the weather, but nobody does anything about it." Do you feel the same way about Unit Testing? Are you actively testing your code, or are you just thinking about testing your code... some day... once you get some more free time...

Real World 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 are driving the AJAX revolution. You'll see examples of RESTful, SOAP, and JSON web services, as well as the strengths and weaknesses of each.

Ajax, Flash, and Java - Choosing The Right Rich Client Technology for Your Next Project by Scott Delap

Today's users are beginning to demand richer and richer application experiences. Plain html pages simply don't cut it anymore. Applications like Google Maps (Ajax) and Yahoo Maps (Flash) show how the UI experience can be pushed to the next level. As an IT manager, how do you decide which route to take however? Should you use Ajax because it is the new "it" technology. Is Flash a viable option with its 95%+

browser availability? Perhaps Java deployed through web start is really the best choice in contrast to what the buzz would lead you to believe. This presentation takes a look at these three core rich client technologies from both deployment/user experience and ease of development perspectives.

Creating Polished Swing Applications by Scott Delap

Too often, Swing applications are slow, ugly, and hard-to-maintain. It turns out that it doesn't have to be this way. Swing can be used to create highly-responsive, beautiful applications that are very maintainable. If this isn't consistent with your own experience, don't feel bad; it's not very obvious how to make Swing sing.

Introducing the Eclipse Rich Client Platform by Scott Delap

Rich client application development using Java can be intimidating giving the vast flexibility in application design and structure. It also can be frustrating to create the large number of support services (persistence, menus, event and job frameworks) that a large scale rich client applications needs. The Eclipse Rich Client Platform is one project attempting to solve these issues by providing a core infrastructure that not only provides the day to day services a rich client application developer needs, but also providing a suggested path to guide you down the road of designing your application. This presentation introduces both the Eclipse RCP and the tools provided by the Eclipse IDE that assist developers in writing RCP apps.

Effective Enterprise Java: State Management by Ted Neward

Managing state--both transient state (like your shopping cart) and your durable state (like your order placements, your inventory management forms, and so on)--is tricky in an enterprise application. In this talk, we'll examine some of the trickiness, both high-level and low-.

Essential db4o by Ted Neward

Tired of writing object-table mappings? For years, Java developers have wrestled with the problems of storing objects into relational format and retrieving them back again; for all that Hibernate and JDO and other O/R tools make it easier (though not easy) to do, isn't there another way? In this presentation, we'll explore an alternative approach, real object persistence, using the db4o toolkit (www.db4objects.com).

Java Annotations: From Definition to Consumption by Ted Neward

Want to get the soup-to-nuts story on Java annotations? In this presentation, we'll first talk about what annotations provide to the Java language. After setting ourselves a conceptual basis to operate from, we'll look at the language definition for Java annotations, from how to use them to how to define them. Finally, we'll take a look at the other side of annotations, consuming them at source-level (using "apt", the annotation processing tool), class-level (using a bytecode toolkit such as BCEL), and at runtime (using enhancements to the Reflection API made in Java5).

Java6: Exploring Mustang by Ted Neward

Mustang, the forthcoming Java6 release, is just around the corner, and even if you're not looking to adopt the new platform right away, it's important to know what's there so you can start to plan for it. In this presentation, we'll go over the major new features of the Java6 platform, including the new integrated XML services capabilities (JAX-WS and JAXB), dynamic/scripting language support (javax.script), new JVM "attach" capabilities, new annotations supported by the javac compiler, and more.

The Busy Java Developer's Guide to ClassLoaders by Ted Neward

If you've ever gotten a ClassCastException and just knew the runtime was wrong about it, or found yourself copying .jar files all over your production server just to get your code to run, then you probably find the Java ClassLoader mechanism to be deep, dark, mysterious, and incomprehensible. Take a deep breath, and relax--ClassLoaders aren't as bad as they seem at first, once you understand a few basic rules regarding their operation, and have a bit more tools in your belt to diagnose ClassLoader problems. And once you've got that, and hear about ClassLoaders' ability to run multiple versions of the same code at the same time, and to provide isolation barriers inside your application, or even compile code on the fly from source form, you might just find that you like ClassLoaders after all... maybe.

Java/EE Web Services and SOA @ Work: Architecture & Development by Tom Marrs

Have you tried to deploy J2EE Web Services and thrown up your hands in frustration at the lack of tool support? Do you want to know how to develop and deploy Java EE-compliant Web Services so that they work every time? Would you like to see how to develop/deploy Web Services in Spring with XFire? Are you wondering if SOA is just hype and fluff? Do you think SOA is just marketing's re-packaging of Web Services? Would you like to know how Web Services and SOA (Service-Oriented Architecture) fit together? If so, then this talk is for you.

Java/J2EE Architecture @ Work: EJB 3 vs Spring and Hibernate by Tom Marrs

You've used EJB in the past and been disappointed - it was too heavy and difficult to use. Like Bruce Tate, maybe you've gone from "Bitter" to "Better, Faster, Lighter". With EJB 3 shipping in early 2006, maybe it's time to take another look. We'll compare EJB 3 with alternative frameworks - Spring and Hibernate - to see if EJB 3 has closed the gap.

Get Groovier with Grails by Venkat Subramaniam

Inspired by the Ruby on Rails project, Grails brings the ease of web development and "convention over configuration" to the Java platform. We will learn how to create web applications using Grails, how to integrate it with Hibernate, and how to Ajax it, all using the built in features of Grails. This section assumes that you are familiar with Groovy or you have attended the #Groovy for Java Programmers# session. The session will be example driven with live coding where we will build a web application from scratch.

Groovy for Java Programmers by Venkat Subramaniam

Object-oriented scripting languages, or agile dynamic languages, as some like to call those, are gaining programmers' attention. Groovy bring this excitement to the Java platform with its ability to generate byte code. You can use Groovy instead of Java for some parts of your application. By learning it, you can switch between the languages where you consider fit.

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.

Refactoring your code - a key step in agility by Venkat Subramaniam

Refactoring is one of the core practices in Agile Software Development. Refactoring is based on some core principles that apply to more than writing good code. But, what's refactoring? Why should you do it? How do you go about doing that? What tools are available to successfully refactor your App?

Spring into Unit Testing by Venkat Subramaniam

Test Driving Development is a valuable technique that has several benefits. However, unit testing is hard when your code has dependencies-this often becomes a quick excuse to give up on TDD. This example driven Zero Powerpoint (ZePo) presentation will help you spring into unit test your Spring applications. We will look at techniques to realize good unit testing, and focus on effective use of mock objects and frameworks to help you toss mock objects for your spring application.

Working with Rules Engines 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.