

Greater Toronto Software Symposium 2006

Four Points by Sheraton Toronto Airport - Mississauga, Ontario

October 20 - 22, 2006

(session listing as of 10/20/2006)

The No Fluff Just Stuff Software Symposium 2006 tour is pleased to announce the Greater Toronto Software Symposium coming to Mississauga on October 20 - 22, 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 Greater Toronto 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) XML / Web Services
- 3) ServerSide Java
- 4) Groovy
- 5) Core Java

Registration Fees

Attendees	Before 10/2/2006	After 10/2/2006
1-4	\$775	\$875
5-9	\$700	\$775
10-14	\$675	\$750
15-24	\$650	\$725
25+	\$625	\$700

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: zimmerman@nofluffjuststuff.com or (303)469-0486.

Greater Toronto Software Symposium 2006

Four Points by Sheraton Toronto Airport - Mississauga, Ontario

October 20 - 22, 2006

(session listing as of 10/20/2006)

Introduction to TestNG, the next generation testing framework for developers by Andrew Glover

No one will argue that JUnit has positively affected the quality of thousands of Java applications around the world. JUnit's simplicity and ease of use ushered in a whole new era of code quality; however, as many developers have found, its simplicity has also limited its use. TestNG was designed from the ground up to overcome some of JUnit's limitations; moreover, TestNG's features make it a great tool to complement your JUnit tests.

Practically Groovy: Real World Groovy for Thrill Seekers by Andrew Glover

The key to incorporating any tool into your development practice is knowing when to use it and when to leave it in the box. Dynamic languages can be an extremely powerful addition to your toolkit, but only when applied properly to appropriate scenarios.

Unit Testing Java Objects with Groovy by Andrew Glover

What makes Groovy particularly appealing with respect to other scripting platforms is its seamless integration with the Java platform. Because it's based on the Java language (unlike other alternate languages for the JRE, which tend to be based on earlier predecessors), Groovy presents an incredibly short learning curve for the Java developer. And once that learning curve has straightened out, Groovy can offer an unparalleled rapid development platform.

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.

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 $SOAP = SOA + P(\text{rofit})$? How do you go about building, deploying, publishing and orchestrating web services without the (Un)Holy Trinity of SOAP, WSDL and UDDI?

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.

"Bottom 10" Reasons that Agile Teams Fail by Clinton Begin

Agile methodologies such as eXtreme Programming and SCRUM are hot topics today -- and they are also hot targets. When things go wrong on an agile project, it's far too convenient to blame the methodology.

Ruby Rebuttal: A case for Java by Clinton Begin

By jumping from Java to Ruby, are we throwing the baby out with the bathwater?

iBATIS and the Enterprise Database by Clinton Begin

This presentation will focus on database challenges that object oriented developers often face in an enterprise environment. Object Relational Mapping (ORM) tools are not ideal for all databases, and therefore alternatives such as iBATIS must be sought.

New and Exciting in Spring Framework 2.0 by Colin Sampaleanu

Spring Framework is the leading full-stack Java/Java EE application framework. Now Spring 2.0 is here, with three attributes which capture what users can expect in this major new release: Simple, Powerful, and Proven.

Spring Web Flow by Colin Sampaleanu

Spring Web Flow has created a lot of excitement in recent months by bringing the ability to model a "flow" or "conversation" in a declarative fashion, when creating Java web applications.

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.

Shale: Turbo-charge your JSF Apps by David Geary

JavaServer Faces is a well designed user interface framework, but it lacks a number of features you might otherwise expect out of the box; for example, JSF does not explicitly provide support for client-side validation. So, from the folks that brought you Struts, comes Shale, a collection of useful enhancements to JSF. A top-level Apache Software Foundation project, Shale adds some really cool features to vanilla JSF, including:

- Web flow: script dialog flow
- Remote Method Calls: easily call JavaBean methods from JavaScript
- Tapestry-like views: code views in pure HTML
- Use Apache Commons Validator validators on the client or server, or both
- JSF testing framework: mocks for easy JSF testing

 There's a lot of cool stuff in Shale that makes JSF a much more compelling proposition. Come see what it's all about.

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 organizations 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.

Agile Metrics and Diagnostics: Knowing When and What to Measure by Deborah Hartmann

Some Agile teams are still using Waterfall metrics, others aren't measuring anything at all. There's little help to be had: few hard-and-fast Agile Metrics have emerged as effective measures. This might seem surprising, but perhaps the temporary and contextual nature of agile metrics explains the emergence of a more heuristic approach. The best teams are creating metrics on demand, in support of their goals, applying their understanding of Agile principles and values to create useful measurement tools on the fly. We propose to call these local, contextual measurement tools diagnostics to distinguish them from more long-term metrics used to measure contribution to overall organizational goals. Since Team metrics or diagnostics are highly contextual, they need to be evaluated carefully to know if they are useful. We will present both long term metrics used by the business, and potential diagnostic measures used by the team. We will present a tool for evaluating diagnostic metrics for Agile teams, and use it to evaluate a few team metrics proposed by participants.

Trends in Enterprise Java by Floyd Marinescu

There are a number of trends occurring that are rapidly changing the way we think about and develop Enterprise Java applications. Trends such as - Annotations, Dependency Injection, and AOP - Web 2.0 turning the internet into an application platform - Domain Driven Design - SOA - Open Source in emerging economies - the rise of scripting languages ... and others are changing the nature of software development in Java.

EJB 3 Part 1: Core Spec and Spring Comparison by Mark Richards

The new EJB 3 specification (JSR-220) offers some great improvements over the prior EJB specs in terms of development simplicity and new features. In this session we will explore in detail some of the new features of the core EJB 3 specification. Included in this session will be defining and accessing session beans, JTA transaction management, declarative security, and interceptors. During the session I will demonstrate the new features of EJB 3 through interactive coding examples. We will then look at how the EJB 3 specification differs from the Spring Framework, where each is useful, and speculate as to what will happen in the future with these two frameworks. This session is part one of a two-part EJB 3 session (part two covers the new Java Persistence API).

EJB 3 Part 2: Java Persistence API (JPA) by Mark Richards

In addition to providing a simplified API, the new EJB 3 specification (JSR-220) defines a standard ORM Java Persistence API (JPA) that replaces those nasty Entity Beans that were part of the EJB 2.x specification. As you will see in this session, JPA bears a striking resemblance to popular ORM solutions like Hibernate and Toplink. In this session we will explore in detail the new Java Persistence API offered by JSR-220. We will start by discussing the overall design and architecture of the JPA and how the major components within JPA interact. We will then look at defining mapping objects (entities) and how to use the EntityManager to manage these entities. Through interactive coding examples we will see how to use the JPA for simple queries, complex queries, and finally stored procedures. This session is part two of a two-part EJB 3 session.

Techniques in Architecture Agility by Mark Richards

As companies continue to change the way they do business, so must the IT systems that support the business. Changes due to regulatory requirements, competitive advantage, mergers, acquisitions, and industry trends require flexible IT systems to meet the demands of the business. Software Architects must therefore make their architectures more agile to meet the flexible demands of today's business. In this session we will explore some of the challenges facing Software Architecture and discuss several concrete techniques for applying agility to both the architecture process and the technical architecture itself. Through real-world examples provided in this session you will learn how to apply various agile techniques to improve your architectures and overcome some of the challenges facing software architecture in today's ever-changing market.

Understanding the Role of an ESB by Mark Richards

The Enterprise Service Bus is an integral part of any Service-Oriented Architecture. It is the glue that binds the business services to the client applications. There are many ESB third-party products and solutions in the marketplace, but in most cases these products only serve to further confuse us in terms of what an ESB is, particularly when you consider that an ESB is really an architectural component that has many different implementations. In this session we will take a detailed, product-agnostic look at the role of an ESB and the capabilities an ESB must provide. Through this session you will learn what an ESB is, the role of an ESB, what capabilities it provides, and the various ways an ESB can be implemented. We will also take a close look at the Java Business Integration (JBI) specification (JSR-208) and see what impact it will have with the ESB world. With the information from this session you will learn how to determine your own specific requirements for an ESB and then match those requirements to the product space rather than having the tail wag the dog!

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.

Guerrilla Web Techniques by Scott Davis

Frameworks? We don't need no stinkin' web frameworks. OK, so maybe that's overstating the case. Web frameworks do plenty of good things, but sometimes they can also be golden handcuffs. Too many web developers fall into the trap of thinking, "If it can't be done by my web framework, then it simply can't be done."

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.

Rolling Your Own Google Maps, part I by Scott Davis

The release of Google Maps was a "Wizard of Oz / Technicolor" moment for web developers everywhere. It didn't just change the way we look at mapping sites; it forever changed the way we look at all web sites. It put AJAX on the map, both figuratively and literally.

Testing the Web Tier by Scott Davis

Hopefully your test plan involves more than, "Well, it compiled..." JUnit is fast becoming a required part of the modern Java developer's toolkit. Unit testing your Java classes is a great start, but your test plan shouldn't stop there. This talk will introduce several additional testing tools for the web developer -- HttpUnit, Canoo WebTest, and JMeter. These tools allow you to test a live website with no changes to the production code. Even better, you can test sites that have been implemented in technologies other than Java.

Java5: The Language, The Libraries, The VM by Ted Neward

Java5 introduced a whole slew of new features, including annotations (JSR 175), new language features (the enhanced for loop, generics, static imports, and more), new library support (java.lang.instrument, among others), and some interesting enhancements to the virtual machine itself.

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-.

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.

Pragmatic XML Services by Ted Neward

There's a lot of talk about web services, and most of it falls into one of two categories: lots of low-level talk about vendor-specific tools and extensions, or lots of high-level talk that never shows you a line of code. XML services aren't that hard, and in this talk, we'll see how, why and when to do one.

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.

Open Source Tools for Agile Development by Venkat Subramaniam

As a Java developer, you have taken the time to learn the basics of the language and relevant parts of its rich API. However, you need more than that to develop serious industrial strength applications. In this presentation, the speaker will introduce you to a number of open source tools which you can use to improve your application quality and your development process.

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?

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.