

Greater Quebec Software Symposium

Four Points By Sheraton Montreal Centre-ville

April 20 - 22, 2007

<http://www.nofluffjuststuff.com/sh/2007-04-montreal>

Fri, Apr. 20, 2007				
	Mont Blanc	Mont Fuji	Mont St-Helens	Mont St-Michel
12:00 - 1:00 PM	REGISTRATION			
1:00 - 1:15 PM	WELCOME			
1:15 - 2:45 PM	Structuring concurrent applications in JDK 5.0 Brian Goetz	Building a Service Oriented Architecture with ServiceMix Jeff Genender	JavaServer Faces: A Whirlwind Tour David Geary	Groovy: Greasing the Wheels of Java Scott Davis
2:45 - 3:15 PM	BREAK			
3:15 - 4:45 PM	Effective Concurrent Java Brian Goetz	High Availability Java Jeff Genender	Killer JavaScript Frameworks: Prototype, Scriptaculous, and Rico David Geary	Groovy and Java: The Integration Story Scott Davis
4:45 - 5:00 PM	BREAK			
5:00 - 6:30 PM	Java Performance Myths Brian Goetz	Migrating from Tomcat To Geronimo Jeff Genender	Ajaxian Faces David Geary	Real World Grails Scott Davis
6:30 - 7:15 PM	DINNER			
7:15 - 8:00 PM	Keynote: Scott Davis			

Sat, Apr. 21, 2007				
	Mont Blanc	Mont Fuji	Mont St-Helens	Mont St-Michel
8:00 - 9:00 AM	BREAKFAST			
9:00 - 10:30 AM	Spring 2.0: New and Noteworthy Mark Fisher	RAD JSF with Seam, Facelets, and Ajax4jsf, Part One David Geary	Abusing Maven For Fun and Profit : (Near) Zero-Admin Deployments Brian Sletten	The Java Memory Model Brian Goetz
10:30 - 11:00 AM	BREAK			
11:00 - 12:30 PM	Message Driven POJOs Mark Fisher	RAD JSF with Seam, Facelets, and Ajax4jsf, Part Two David Geary	Applied Object-Oriented Metrics Brian Sletten	Squashing bugs with FindBugs Brian Goetz
12:30 - 1:30 PM	LUNCH			
1:30 - 3:00 PM	The Google Web Toolkit, Part One David Geary	Introduction to Spring Security Mark Fisher	Applied AOP Brian Sletten	The Zen of REST Scott Davis
3:00 - 3:15 PM	BREAK			
3:15 - 4:45 PM	The Google Web Toolkit, Part Two David Geary	The Role of Spring in an ESB Mark Fisher	NetKernel : XML Processing for the 21st Century Brian Sletten	Mocking Web Services Scott Davis
4:45 - 5:30 PM	BIRDS OF A FEATHER SESSIONS			

Sun, Apr. 22, 2007				
	Mont Blanc	Mont Fuji	Mont St-Helens	Mont St-Michel
8:00 - 9:00 AM	BREAKFAST			
9:00 - 10:30 AM	Introduction to Object-Relational Mapping with Hibernate Brian Sam-Bodden	Internationalization and Localization in Java David Bock	Give it a REST Brian Sletten	Ajax development with the Yahoo! UI Library and Grails Scott Davis
10:30 - 11:00 AM	BREAK			
11:00 - 12:30 PM	Advanced Object-Relational Mapping with Hibernate Brian Sam-Bodden	Capistrano: Application Deployment and More David Bock	Data Integration : Beyond Cutesy Mashups Brian Sletten	Groovy: Greasing the Wheels of Java Scott Davis
12:30 - 1:15 PM	LUNCH			
1:15 - 2:15 PM	EXPERT PANEL DISCUSSION			
2:15 - 3:45 PM	Introducing Agility to Large Organizations David Bock	Git 'R Done : Scheduling Work With Quartz Brian Sletten	Complex Builds with Ant Brian Sam-Bodden	Groovy and Java: The Integration Story Scott Davis
3:45 - 4:00 PM	BREAK			
4:00 - 5:30 PM	Business Rules Engines in Java and J2EE- An Introduction to the Drools Rules Engine Brian Sam-Bodden	Introducing the Semantic Web Brian Sletten	Maintaining Project Integrity with JDepend, Macker, PMD, Maven, and other open source tools David Bock	Atom: From Blogging to Data Syndication Scott Davis

Greater Quebec Software Symposium

Four Points By Sheraton Montreal Centre-ville

April 20 - 22, 2007

<http://www.nofluffjuststuff.com/sh/2007-04-montreal>

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.

Effective Concurrent Java by Brian Goetz

The Java programming language has turned a generation of applications programmers into concurrent programmers through its direct support of multithreading. However, the Java concurrency primitives are just that: primitive. From them you can build many concurrency utilities, but doing so takes great care as concurrent programming poses many traps for the unwary.

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.

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.

Squashing bugs with FindBugs 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 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.

Advanced Object-Relational Mapping with Hibernate by Brian Sam-Bodden

Hibernate is rapidly becoming the tool of choice when it comes to Object-Relational Mapping in Java. For simple applications with fairly simple object models and database schemas, using Hibernate is fairly straight forward. Unfortunately for most of us real applications have complex object-models that need to be wired to sometimes ancient and convoluted database schemas.

Complex Builds with Ant by Brian Sam-Bodden

Ant has revolutionized the way we build applications in Java and it has become a de facto standard in the Java world. As applications grow in complexity some developers are finding themselves dealing with ever growing and complex builds. Complex builds have to deal with Multiple Operating System, multiple Application Servers, multiple APIs and multiple stages of development.

Business Rules Engines in Java and J2EE- An Introduction to the Drools Rules Engine 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.

Abusing Maven For Fun and Profit : (Near) Zero-Admin Deployments by Brian Sletten

Ok, I can't promise you profit, but hopefully you'll have fun. Maven 2 introduces a number of new features (including that performance feature) that make it a swell project management tool for development. Come hear about how we can abuse Maven to manage distributed deployment scenarios before the Modules JSR is done.

Applied Object-Oriented Metrics by Brian Sletten

Object-oriented code metrics are a little like Artificial Intelligence: those who did it twenty years ago roll their eyes at the thought and prophesy the same ultimate failure at applicability now. Those who grew up with Java are approaching the topic with new eyes and are finding useful ways of incorporating metrics into their projects. Come hear about tools and ways to measure properties of software, how they might be beneficial and where you are likely to go astray with this approach.

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.

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.

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.

Data Integration : Beyond Cutesy Mashups by Brian Sletten

Ever since we started doing relational joins, we've looked for ways to tie data together. The web has given us no end of new data sources to integrate but it seems like the best we can come up with is locating Starbucks on Google Maps. The problem with browser-based mashups is that they don't survive the session, we have no way of referring to the results in future queries and ultimately we don't maintain ownership or control of the process. We want control of our data and our mashup results. We want ever more ways to view, explore and requery them in multi-faceted ways. We want our data integration strategies to be less Vanilla Ice "Under Pressure/Ice-Ice Baby" and more Nine Inch Nails "The Hand that Feeds" (trust me, it makes sense).

Git 'R Done : Scheduling Work With Quartz by Brian Sletten

Software engineers are usually familiar with the notion of scheduled tasks and cron jobs at the OS level. Quartz is a relatively new open source Java API for scheduling jobs in your applications or Enterprise.

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.

Internationalization and Localization in Java by David Bock

Internationalization and Localization in Java is easy, right? Everyone knows you just store your strings in some resource bundles, set the locale, wave your hands a little bit, and your application is good-to-go.

Right? Maybe not... Java provides some great utilities to get started, but leaves you needing more when it comes to things like screen layout, cultural sensitivities, semantic differences in translation, use of color and iconography, and other issues.

Capistrano: Application Deployment and More by David Bock

Capistrano (formerly Switchtower) is a tool originally written to help automate application deployment for Ruby on Rails. It does this well, but it has grown up into a tool capable of much, much more. It can be used for deploying Java applications, updating server configurations across an enterprise, administering networks, backing up files, and all sorts of other activities. Any activity you might do from the command line, you can now do simultaneously across large numbers of machines, with all machines succeeding (or rolling back in case of failure) together.

Introducing Agility to Large Organizations by David Bock

For several years, I was a member of a team of people caught in the middle of a 200+ person software development company, with senior management wanting "buzzword compliant process improvement" such as CMMI, and engineers wanting more "agile" solutions (and people on both sides confusing Agile with ad-hoc). We were responsible for sorting it all out. Reconciling this was a herculean effort, and can be a source of lessons learned for your own process improvement efforts. Are you trying to be more agile in your organization? Are you expecting it to be harder than it needs to be because of political and bureaucratic forces beyond your control? Do you have to "educate" your senior management to protect them from buzzwords? Come learn from my successes... and mistakes.

Maintaining Project Integrity with JDepend, Macker, PMD, Maven, and other open source tools by David Bock

How many times have you started a new project only to find that several months into it, you have a big ball of code you have to plod through to try to get anything done? How many times have you been the "new guy" on an established project where it seems like the code grew more like weeds and brambles than a well-tended garden? With a few good structural guidelines and several tools to help analyze the code, we can keep our project from turning into that big ball of mud, and we can salvage a project that is already headed down that path.

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.

Building a Service Oriented Architecture with ServiceMix by Jeff Genender

There is a big buzz and a lot of hype about Service Oriented Architecture (SOA) and the Enterprise Service Bus (ESB). A solid SOA architecture allows you to use standards-based, pluggable, and loosely coupled services to integrate and communicate the critical business logic from disparate systems. This essentially allows a common ground for different systems to interact with each other. But there is more to a good SOA implementation than simply integrating services. This is where orchestration and other key concepts become important, and an ESB helps fill in this gap.

High Availability Java by Jeff Genender

Clustering and high availability is a critical topic in Java Enterprise circles. There are many strategies to clustering but not every strategy is right for every application. Knowing which strategy for which application is critical in weighing scalability vs performance. This session discusses clustering topologies such as full replication, distributed clustering, master/slave, and other strategies.

Migrating from Tomcat To Geronimo by Jeff Genender

Have you thought about moving your web applications to an application server, but were afraid to try? Geronimo may be the application server for you! Since Geronimo allows you to take on an application server a little at a time by building the stack that is right for you, migrating your Tomcat applications to Geronimo is a great first step.

Migrating from Tomcat To Geronimo by Jeff Genender

Have you thought about moving your web applications to an application server, but were afraid to try? Geronimo may be the application server for you! Since Geronimo allows you to take on an application server a little at a time by building the stack that is right for you, migrating your Tomcat applications to Geronimo is a great first step.

Spring 2.0: New and Noteworthy by Mark Fisher

Spring 2.0 has marked a major advance in the Spring Framework. While still maintaining backwards compatibility, this release adds quite a few new features. What are those features and how do they add value? Come by and see. In this session we'll provide a practical tour of what's new in Spring 2.0. Spring 1.x users who are looking to upgrade to Spring 2.0 will love this session. If you're not using Spring already, this talk will give a great overview of the things you're missing out by not using Spring 2.0.

Message Driven POJOs by Mark Fisher

One of the most exciting new features of Spring 2.0 is its support for Message-Driven POJOs. It is now possible to receive JMS messages asynchronously and delegate the handling of those messages to simple objects. If your POJO has a return value, it will automatically be sent to a reply destination. Spring's messaging containers support configurable pooling of concurrent consumers and offer full integration with Spring's transaction management.

Introduction to Spring Security by Mark Fisher

Spring Security (a.k.a. Acegi) enables self-contained, consistent solutions for securing your applications. The interceptor-based approach is non-invasive even when extended to accommodate domain-specific requirements. The two main security processes (authentication and authorization) are decoupled in order to provide flexibility across a wide variety of providers and strategies.

The Role of Spring in an ESB by Mark Fisher

An Enterprise Service Bus (ESB) brings flow-related concepts such as transformation and routing to a Service-Oriented Architecture. An ESB can also provide an abstraction for endpoints. This promotes flexibility in the transport layer and enables pluggability of POJO services.

GIS for Web Developers by Scott Davis

Based on the book GIS for Web Developers, this talk demonstrates how you can build your own Google Maps in-house using nothing but open source software. We also discuss integrating free, public domain data from sources like the US Census Bureau and the USGS. If you're looking for real-world examples of AJAX in use, you'll find it here. If you're looking for real-world examples of web services in use, you'll find it here.

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.

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.

Ajax development with the Yahoo! UI Library and Grails by Scott Davis

Yahoo! is a company that eats its own dog food. They open sourced the Ajax code that drives many of their own websites, including their eponymous homepage, Yahoo! Mail, and Yahoo! News. Come see first hand how the various pieces of the library work together as a seamless whole.

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.

Atom: From Blogging to Data Syndication by Scott Davis

You've probably read a blog or two, but have you ever looked at the underlying protocol that makes the blogosphere work? RSS kick-started the phenomenon, but Atom is the IETF specification that codifies it. Atom as a blogging dialect is interesting, but Atom as a data syndication format is something that is on the rise. Google recently deprecated its SOAP API and is aggressively moving towards Atom as its preferred way to interact with its services.