

Lone Star Software Symposium: Austin

Marriott Austin Airport Hotel

Jun. 29 - Jul. 01, 2007

<http://www.nofluffjuststuff.com/sh/2007-06-austin>

(event schedule as of June 28, 2007)

| Fri, Jun. 29, 2007 | | | | | | |
|--------------------|---|---|---|---|--|--|
| | Lone Star A&B | Lone Star C | Lone Star D | Bluebonnet | Pecan A | Pecan B |
| 12:00 - 1:00 PM | REGISTRATION | | | | | |
| 1:00 - 1:15 PM | WELCOME | | | | | |
| 1:15 - 2:45 PM | Ajax Architecture Stuart Halloway | Structuring concurrent applications in JDK 5.0 Brian Goetz | Agile Requirements with User Stories Pete Behrens | Groovy: Greasing the Wheels of Java Scott Davis | Annotation Hammer Venkat Subramaniam | Implementing SOA Neal Ford |
| 2:45 - 3:15 PM | BREAK | | | | | |
| 3:15 - 4:45 PM | JavaScript for Ajax Programmers Stuart Halloway | Effective Concurrent Java Brian Goetz | Agile Release Estimating, Planning and Tracking Pete Behrens | Groovy and Java: The Integration Story Scott Davis | Java 6 Features, what's in it for you? Venkat Subramaniam | Power Regular Expressions in Java Neal Ford |
| 4:45 - 5:00 PM | BREAK | | | | | |
| 5:00 - 6:30 PM | Prototype: Ajax and JavaScript++ Stuart Halloway | 10 Ways to Improve Your Code Neal Ford | A Scrum Experience Pete Behrens | Real World Grails Scott Davis | OSGI: A Well Kept Secret Venkat Subramaniam | Squashing bugs with FindBugs Brian Goetz |
| 6:30 - 7:15 PM | DINNER | | | | | |
| 7:15 - 8:00 PM | Keynote: by Scott Davis | | | | | |

| Sat, Jun. 30, 2007 | | | | | | |
|--------------------|---|--|---|---|--|--|
| | Lone Star A&B | Lone Star C | Lone Star D | Bluebonnet | Pecan A | Pecan B |
| 8:00 - 9:00 AM | BREAKFAST | | | | | |
| 9:00 - 10:30 AM | Debugging and Testing the Web Tier Neal Ford | Drooling with Groovy and Rules Venkat Subramaniam | Developing Applications Using in Spring 2.0 Ryan Breidenbach | Java Performance Myths Brian Goetz | Ruby for Java programmers Bruce Tate | Agile Enterprise Architecture: The role of the architect Pete Behrens |
| 10:30 - 11:00 AM | BREAK | | | | | |
| 11:00 - 12:30 PM | Advanced Selenium Neal Ford | Spring into Groovy Venkat Subramaniam | The Zen of REST Scott Davis | The Java Memory Model Brian Goetz | Rails for Java Programmers Bruce Tate | Agile Tooling: Team to Enterprise Pete Behrens |
| 12:30 - 1:30 PM | LUNCH | | | | | |
| 1:30 - 3:00 PM | Effective Hibernate Scott Leberknight | Domain Driven Design: Simplifying Enterprise Architecture Ramnivas Laddad | Mocking Web Services Scott Davis | Introduction To Agile Web Development With Grails Jeff Brown | Building ChangingThePresent: Agility in Action Bruce Tate | Data Integration : Beyond Cutesy Mashups Brian Sletten |
| 3:00 - 3:15 PM | BREAK | | | | | |
| 3:15 - 4:45 PM | Advanced Hibernate Scott Leberknight | Leveraging annotations with AOP Ramnivas Laddad | Harnessing the Power of Maven Ryan Breidenbach | Advanced View Techniques With Grails Jeff Brown | Metrics-driven Agile Development Neal Ford | JavaServer Faces: A Whirlwind Tour David Geary |
| 4:45 - 5:30 PM | BIRDS OF A FEATHER SESSIONS | | | | | |

| Sun, Jul. 01, 2007 | | | | | | |
|--------------------|--|---|--|--|---|--|
| | Lone Star A&B | Lone Star C | Lone Star D | Bluebonnet | Pecan A | Pecan B |
| 8:00 - 9:00 AM | BREAKFAST | | | | | |
| 9:00 - 10:30 AM | Enterprise Applications with Spring: Part 1 Ramnivas Laddad | The Busy Java Developer's Guide to ClassLoaders Ted Neward | What's New in Java 6 Jason Hunter | RAD JSF with Seam, Facelets, and Ajax4jsf, Part One David Geary | Pragmatic Extreme Programming Neal Ford | Git 'R Done : Scheduling Work With Quartz Brian Sletten |
| 10:30 - 11:00 AM | BREAK | | | | | |
| 11:00 - 12:30 PM | Enterprise Applications with Spring: Part 2 Ramnivas Laddad | The Busy Java Developer's Guide to Debugging and Monitoring Ted Neward | Web Publishing 2.0 Jason Hunter | RAD JSF with Seam, Facelets, and Ajax4jsf, Part Two David Geary | Building DSLs in Static and Dynamic Languages Neal Ford | Give it a REST Brian Sletten |
| 12:30 - 1:15 PM | LUNCH | | | | | |
| 1:15 - 2:15 PM | EXPERT PANEL DISCUSSION | | | | | |
| 2:15 - 3:45 PM | The Google Web Toolkit, Part One David Geary | The Busy Java Developer's Guide to Reflection Ted Neward | Productive Programmer: Acceleration, Focus, and Indirection Neal Ford | Spring/Hibernate Integration Patterns, Idioms, and Pitfalls Scott Leberknight | XQuery By Example: Building an Email Archive System Jason Hunter | NetKernel : XML Processing for the 21st Century Brian Sletten |
| 3:45 - 4:00 PM | BREAK | | | | | |
| 4:00 - 5:30 PM | The Google Web Toolkit, Part Two David Geary | The Busy Java Developer's Guide to Java Platform Security Ted Neward | Productive Programmer: Automation and Canonicity Neal Ford | AspectJ for Spring Developers Ramnivas Laddad | Forgotten Web Algorithms Jason Hunter | Introducing the Semantic Web Brian Sletten |

Lone Star Software Symposium: Austin

Marriott Austin Airport Hotel

Jun. 29 - Jul. 01, 2007

<http://www.nofluffjuststuff.com/sh/2007-06-austin>

(event schedule as of June 28, 2007)

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.

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.

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.

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.

Give it a REST by Brian Sletten

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

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

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

/ commercial product built on a compelling modern architecture can be used to create, manipulate and transform XML.

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.

Ruby for Java programmers by Bruce Tate

With the explosion of Ruby on Rails and the Java community interest in features like closures and continuations, the Ruby programming language is an excellent one for all developers to know. As the JRuby virtual machine picks up steam, Ruby becomes a must language to understand. The best way to learn Ruby is to see it in action.

Rails for Java Programmers by Bruce Tate

The productivity of Ruby on Rails cannot be denied, but the explosion of Ruby on Rails left many developers, with hard commitments to Java deployment platforms, out in the cold. The continued evolution of JRuby can change that. JRuby is a Ruby implementation on the Java virtual machine. And yes, it runs Rails. In this session, you will learn Rails as it was meant to be learned, within the context of building a live site, from scratch.

Building ChangingThePresent: Agility in Action by Bruce Tate

ChangingThePresent is the increasingly popular charity donations portal that lets you give donation gifts instead of another pair of fuzzy slippers. The site is built and maintained under unusual circumstances. The team is distributed, with no more than two developers in any one place. The team uses agile techniques such as automated testing, heavy customer involvement, and a SCRUM-like release plan to deliver the core features.

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.

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.

What's New in Java 6 by Jason Hunter

The Java 6 (Mustang) release should make your life easier, for a change. It doesn't alter the core language like Java 5 did. It doesn't pack in so many sub-JSRs that you'll be overwhelmed by the amount you have to learn. Instead Java 6 adds several handy things that honestly should have been added before. Among the improvements we'll cover in this fast-paced class: * A new Console class * A real Compiler API * A GIF writer * Pluggable Locale data * Access to disk partition size data * Array reallocation * Low-level floating

point functions * Reflective access to parameter names * Access to network interface details * Pluggable annotation processing * Improved class file format * Streaming XML with StAX * A new Scripting interface

Web Publishing 2.0 by Jason Hunter

If we're moving toward Web 2.0, what does that mean for online publishing? In this talk I'll answer that question. Based on my experience as Principal Technologist at Mark Logic working with dozens of the largest online publishers, I'll present a vision for how the Web 2.0 concepts like personalization, collective intelligence, the long tail, and the importance of "owning the data" can and should reshape the face of online publishing -- and how XML, XQuery, and XML-aware text search act as the key enablers. I'll also introduce new Web Publishing 2.0 concepts like "Sweat the content" and "Give answers not links".

XQuery By Example: Building an Email Archive System by Jason Hunter

The classic searchable email archive system is cluged together -- a frankenstein monster combining a relational database with a search engine, with Java just barely able to keep the two together. In this talk we'll demonstrate how email is more content than data, how it's better encoded in XML rather than relational tables, and how Java can convert emails to XML and drive an XQuery backend to produce a simpler and more scalable email archive system.

Forgotten Web Algorithms by Jason Hunter

In this talk I'll explain -- without any needless math or boring proofs -- several fun algorithms of interest to back-end web programmers. Each algorithm was selected because it's really practical, really interesting, or both. The algorithms aren't always the same but can include: public key cryptography, credit card checksum validation, TCP Slow Start, two's complement, priority queues, the XOR swap, and the Google MapReduce function for massively distributed calculation.

Introduction To Agile Web Development With Grails by Jeff Brown

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

Advanced View Techniques With Grails by Jeff Brown

Grails provides view technologies that offer great flexibility and power without the complexity introduced by other Java web application frameworks. Custom tag libraries are a snap. GSP Templates provide a simple mechanism for reusing UI elements. Sitemesh is integrated to help provide a consistent presentation across the entire application. Grails provides simple mechanisms for leveraging the power of Ajax.

Implementing SOA by Neal Ford

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

Power Regular Expressions in Java by Neal Ford

Regular expressions should be an integral part of every developer's toolbox, but most don't realize how important it is. Regular expressions have existed for decades, but many developers don't understand how to take full advantage of this powerful mechanism, either through command line tools and editors or in their development.

10 Ways to Improve Your Code by Neal Ford

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

Debugging and Testing the Web Tier by Neal Ford

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

Advanced Selenium by Neal Ford

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

Metrics-driven Agile Development by Neal Ford

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

Pragmatic Extreme Programming by Neal Ford

This session talks about how to actually get XP done in the real world (and what to tell your boss). This session includes artifacts (like project tracking sheets) from real XP projects.

Building DSLs in Static and Dynamic Languages by Neal Ford

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

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

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

Productive Programmer: Automation and Canonicity by Neal Ford

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

Agile Requirements with User Stories by Pete Behrens

User Stories, a key practice from Extreme Programming, provide a right-sized solution to more efficiently identify, track and implement product requirements. Learn how identify, write and decompose "good" user stories that drive agile behavior and business value.

Agile Release Estimating, Planning and Tracking by Pete Behrens

Business leaders and stakeholders require accountability and accuracy in our software release projections and yet, as an industry, we have failed. However, many of these same leaders are not convinced that agile is any more than an excuse to avoid projections at all. While it is true that agility provides the framework to support change, it doesn't mean you can't provide accurate projections. In fact, a well-executed agile process actually provides more accurate results with less time investment than traditional methods. This session will demonstrate these agile project management techniques to manage 6-12 month projects.

A Scrum Experience by Pete Behrens

Scrum is a very easy agile framework to understand, but is very difficult in practice. Why is that? For one, Scrum requires compressing an entire software lifecycle into very short time increments of 2-4 weeks in length. It requires cross-functional team commitment, discipline, communication, and collaboration to accomplish their goals. These changes are difficult and often expose organizational and environmental issues that must be addressed for the team to be successful. This session brings focus to the Scrum heartbeat - the sprint. After a brief introduction of the Scrum framework and a focus on the sprint, we will be taking an experiential hands-on journey through a full sprint with your newly formed team.

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 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: Simplifying Enterprise Architecture by Ramnivas Laddad

Domain-Driven Design (DDD) recommends dealing with complex software system using a domain model and preserving the model through implementation. A direct mapping between domain model and software artifacts creates simple to understand, inexpensive to implement, and easy to evolve systems. The DDD approach suggests ways to distill domain knowledge into a model and offers patterns to design and implement that model.

Leveraging annotations with AOP by Ramnivas Laddad

Specifying metadata using annotations has gained huge popularity since its introduction in Java 5. However, the story on consuming annotations isn't as clear. Reading and processing annotation is still a complex process often requiring you to understand byte-code manipulation tools and their low-level API. As a result, most developers shy away from using custom annotations, limiting their usages of annotations only those prescribed by frameworks. The result is missed opportunities for programming simplification. In this session, we explore how AOP can make it a simple task to consume annotation in a powerful manner.

Enterprise Applications with Spring: Part 1 by Ramnivas Laddad

This session (part 1 of the two-part session) shows the core concepts in the Spring Framework -- the most popular lightweight container that recently crossed 1 million downloads.

Enterprise Applications with Spring: Part 2 by Ramnivas Laddad

This session (the second part of the 2-part session) will cover advanced concepts in the Spring framework. While the core concepts in the first session will get you started with Spring, the advanced concepts in this session will make you effective at developing Spring-based applications.

AspectJ for Spring Developers by Ramnivas Laddad

If you are keeping up-to-date with all the cool features in Spring 2.0, you have surely heard about much improved integration with AspectJ. AspectJ is for real. Come to this session to understand the core concepts of this wonderful technology and how to use it to create even simpler Spring-based applications.

Developing Applications Using in Spring 2.0 by Ryan Breidenbach

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

Harnessing the Power of Maven by Ryan Breidenbach

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

Groovy: Greasing the Wheels of Java by Scott Davis

This is the year of the dynamic scripting language. Ruby (and Rails) has won the hearts and minds of many independent software developers. JavaScript is experiencing a renaissance thanks to the wild success

of AJAX and websites like Google Maps. And Groovy (JSR-241) brings the same level of excitement and "scripting goodness" to the Java platform.

Groovy and Java: The Integration Story by Scott Davis

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

Real World Grails by Scott Davis

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

KEYNOTE: No, I Won't Tell You Which Web Framework to Use: or The Truth (with Jokes) by Scott Davis

"Which framework should I use?" is the question most often heard on the No Fluff, Just Stuff tour. It's well worth asking. Unfortunately, there is no simple answer. After years on the tour, most speakers have crafted a response that would make any Washington politician proud -- long on style, but essentially, "Well, it depends..."

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.

Effective Hibernate by Scott Leberknight

Hibernate seems simple on the surface yet when you go beyond very simple use cases it can become much more complex. Intended for beginner to intermediate-level Hibernate developers, come see how to put Hibernate to effective use on your projects.

Advanced Hibernate by Scott Leberknight

This session covers advanced Hibernate topics beyond simple object persistence including session management, object locking, detachment and versioning, lazy loading performance issues and query tuning, advanced O/R mapping support, legacy database considerations, and the Hibernate cache architecture.

Spring/Hibernate Integration Patterns, Idioms, and Pitfalls by Scott Leberknight

Using Spring's Hibernate integration significantly simplifies applications that use Hibernate for data persistence by removing tedious and repetitive infrastructural code that you need to write. Intended for developers familiar with Spring/Hibernate integration basics, who want to learn additional idioms and solutions to common problems.

Ajax Architecture by Stuart Halloway

Ajax applications have unique architectural challenges and opportunities. This presentation will show you how to take advantage of the Ajax's strengths, and work around its quirks.

JavaScript for Ajax Programmers by Stuart Halloway

This presentation covers JavaScript from the perspective of an Ajax programmer. We assume that you may be using an Ajax toolkit, but still need to be able to read, modify, and test the JavaScript code in your

application. You will learn the common idioms of JavaScript by looking at working code from the Ajax toolkits themselves.

Prototype: Ajax and JavaScript++ by Stuart Halloway

Learn to simplify Ajax development with Prototype through a series of real-world examples. Along the way, learn to code in Prototype's modern JavaScript style, taking advantage of Prototype's extensions to JavaScript's object model

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.

The Busy Java Developer's Guide to Debugging and Monitoring by Ted Neward

Bugs? We all know your code has no bugs, but someday, you're going to find yourself tracking down a bug in somebody else's code, and that's when it's going to be helpful to make use of the wealth of tools that the Java Standard Platform makes available to you--tools that your IDE may not know exist, tools that you can make use of even within a production environment.

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

If you've never used Reflection (`java.lang.reflect`), you don't know what you're missing. In this presentation, we'll take a code-first, soup-to-nuts look at the Java Reflection APIs, from how to examine the class metadata that Reflection provides, to using annotations to enhance that metadata with your own information, even through the use of Java Dynamic Proxies to create flexible object "interceptors" that can layer services in front of ordinary method calls with nothing more complicated than an interface and a factory.

The Busy Java Developer's Guide to Java Platform Security by Ted Neward

Permissions, policy, `SecurityExceptions`, oh my! The Java platform is a rich and powerful platform, complete with a rich and powerful security mechanism, but sometimes understanding it and how it works can be daunting and intimidating, and leave developers with the basic impression that it's mysterious and dark and incomprehensible. Nothing could be further from the truth, and in this presentation, we'll take a pragmatic, code-first look at the Java security platform, including `Permissions`, the `SecurityManager` and its successor, `AccessController`, the `Policy` class and policy file syntax, JAAS, and more.

Annotation Hammer by Venkat Subramaniam

Annotation is an interesting feature in Java. However, like any features, there are good uses and bad uses. When should you use Annotation? This presentation will answer that question for you.

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

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

OSGi: A Well Kept Secret by Venkat Subramaniam

In this presentation we will introduce OSGi and discuss how it can help modularize and version your enterprise Java applications.

Drooling with Groovy and Rules by Venkat Subramaniam

Rule based programming allows us to develop applications using declarative rules. These can simplify development in applications where such rules based knowledge is used for decision making.

Spring into Groovy by Venkat Subramaniam

What do you get when you mix an agile, object-oriented, dynamic language with a lightweight, flexible, and extensible framework? You get a Groovier Spring. Spring allows you to develop using Groovy as much as

Java. Groovy brings some neat concepts to the Java Platform that is hard to realize directly through the Java language. Using these capabilities can lead to elegant and easier Spring development.