# Lone Star Software Symposium: Austin 2006

Austin Airport Marriott South - Austin, TX

July 07 - 09, 2006

(session listing as of 7/7/2006)

| Friday, July 7, 2006 | | | | | |
|---|---|---|---|---|---|
| | **Bluebonnet** | **Salon A & B** | **Pecan** | **Salon C** | **Salon D** |
| 12:00 - 1:00 PM | REGISTRATION | | | | |
| 1:00 - 1:15 PM | WELCOME | | | | |
| 1:15 - 2:45 PM | Open Source Tools for Agile Development<br>Venkat Subramaniam | Effective Teams: The dirty little secret<br>Bruce Tate | Spring Fundamentals<br>Stuart Halloway | Introduction to Java threads<br>Brian Goetz | JavaServer Faces: A Whirlwind Tour<br>David Geary |
| 2:45 - 3:15 PM | BREAK | | | | |
| 3:15 - 4:45 PM | Groovy for Java Programmers<br>Venkat Subramaniam | Where Agile meets Argyle: New processes in established companies<br>Bruce Tate | Spring Dependency Injection<br>Stuart Halloway | Structuring concurrent applications in JDK 5.0<br>Brian Goetz | JSF: State of the Art<br>David Geary |
| 4:45 - 5:00 PM | BREAK | | | | |
| 5:00 - 6:30 PM | Get Groovier with Grails<br>Venkat Subramaniam | Test First!<br>Bruce Tate | Spring Security with ACEGI<br>Stuart Halloway | The Java Memory Model<br>Brian Goetz | Shale: Turbo-charge your JSF Apps<br>David Geary |
| 6:30 - 7:15 PM | DINNER | | | | |
| 7:15 - 8:30 PM | KEYNOTE: DAVE THOMAS | | | | |

| Saturday, July 8, 2006 | | | | | |
|---|---|---|---|---|---|
| | **Bluebonnet** | **Salon A & B** | **Pecan** | **Salon C** | **Salon D** |
| 8:00 - 9:00 AM | BREAKFAST | | | | |
| 9:00 - 10:30 AM | Java5: The Language, The Libraries, The VM<br>Ted Neward | Ruby for Java Programmers<br>Dave Thomas | Ajax Architecture<br>Stuart Halloway | Improving Java code quality with code auditing tools<br>Brian Goetz | Practices of an Agile Developer<br>Venkat Subramaniam |
| 10:30 - 11:00 AM | BREAK | | | | |
| 11:00 - 12:30 PM | Effective Enterprise Java: State Management<br>Ted Neward | Ruby on Rails<br>Dave Thomas | JavaScript for Ajax Programmers<br>Stuart Halloway | Java Performance Myths<br>Brian Goetz | Real-world Agile Development<br>Neal Ford |
| 12:30 - 1:30 PM | LUNCH | | | | |
| 1:30 - 3:00 PM | Using Ajax with Ruby on Rails<br>Dave Thomas | Hibernate by Example<br>Eitan Suez | Spring AOP<br>Stuart Halloway | Killer Web UIs<br>David Geary | Creating, Telling, and Tracking User Stories<br>David Hussman |
| 3:00 - 3:15 PM | BREAK | | | | |
| 3:15 - 4:45 PM | Testing your Rails Application<br>Dave Thomas | Working with Rules Engines<br>Venkat Subramaniam | Advanced Hibernate<br>Stuart Halloway | Ajaxian Faces<br>David Geary | #Show Me the Numbers# - Agile Planning Tools and Techniques<br>David Hussman |
| 4:45 - 5:30 PM | BIRDS OF A FEATHER SESSIONS | | | | |

| Sunday, July 9, 2006 | | | | | |
|---|---|---|---|---|---|
| | **Bluebonnet** | **Salon A & B** | **Pecan** | **Salon C** | **Salon D** |
| 8:00 - 9:00 AM | BREAKFAST | | | | |
| 9:00 - 10:30 AM | Performance Monitoring in J2EE Applications<br>Ramnivas Laddad | Guerrilla Web Techniques<br>Scott Davis | Pragmatic XML Services<br>Ted Neward | SOA: Next Wave of Enterprise Development or Return of the Son of CORBA?<br>Neal Ford | Losing Battles and Winning Wars: Adopting Agile<br>David Hussman |
| 10:30 - 11:00 AM | BREAK | | | | |
| 11:00 - 12:30 PM | The State of AOP<br>Ramnivas Laddad | Real World Web Services<br>Scott Davis | Extend the Customization Possibilities of Your Java App with Scripts<br>Ted Neward | Clean Up Your Code: 10 Java Coding Tricks, Techniques, and Philosophies<br>Neal Ford | Stepping Toward Agile<br>David Hussman |
| 12:30 - 1:15 PM | LUNCH | | | | |
| 1:15 - 2:00 PM | EXPERT PANEL DISCUSSION | | | | |
| 2:00 - 3:30 PM | XML Data Binding with JiBX<br>Eitan Suez | What's New in Spring 2<br>Bruce Tate | Testing with Selenium<br>Neal Ford | Domain Driven Design with AOP and DI<br>Ramnivas Laddad | Easing into Agile<br>Scott Davis |
| 3:30 - 3:45 PM | BREAK | | | | |
| 3:45 - 5:15 PM | Extreme Agility with jMatter<br>Eitan Suez | Three Technologies to Watch<br>Bruce Tate | The Productive Programmer<br>Neal Ford | Enterprise AOP with AspectJ<br>Ramnivas Laddad | Holistic Testing<br>Scott Davis |

# Lone Star Software Symposium: Austin 2006

Austin Airport Marriott South - Austin, TX
July 07 - 09, 2006
(session listing as of 7/7/2006)

**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.

**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.

**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.

**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.

**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.

**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 langauges.

**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-conciously hostile to many of the principles of Agile development.

**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 archtiecture, a waterfall process and COBOL. How? By building a great team.

**Test First! by Bruce Tate**
Test-first development should be a cornerstone techniques for all developers. Whether you use test-first development for all of the code you write (as suggested by Agile experts such as Bob Martin), or you just apply it to certain problems, every developer should know how. In this session, you'll see us develop a simple application using TDD, and you'll also learn why the technique is so effective.

**Testing your Rails Application by Dave Thomas**
The Ruby on Rails framework has unit and functional testing baked right in. In this talk we'll see how easy it is to get started with testing in Rails, and we'll explore jut how deep the testing support goes.

**Ruby on Rails by Dave Thomas**
The Ruby on Rails framework has exploded onto the scene over the last few months. Propelled by some genuine benefits, and fueled by a whole lot of controversy, Rails seems here to stay. So, is it a Java killer?

(No.) Is it a great way to develop certain classes of web application? (Yes.) Does it really deliver the 10-fold increase in developer productivity that some have claimed? (It depends...)

**Ruby for Java Programmers by Dave Thomas**
Ruby recently enjoyed its tenth birthday. Instead of cake and candles, the community celebrated by releasing a wave of new libraries and frameworks that make Ruby programming even easier. This talk features some of the best of these, as we explore Ruby.

**Using Ajax with Ruby on Rails by Dave Thomas**
Ajax is becoming a requirement for new applications: it creates richer user experiences and more dynamic applications. However, doing Ajax by hand is difficult and error prone. The good news is that if you use Rails, you don't have to do Ajax the hard way.

**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.

**#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.

**Stepping Toward Agile by David Hussman**
As with many methodologies, moving agile into an organizations poses larger challenges. This session will provide 3 tactical steps that can help your adoption of agile.

**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: <ul><li>Tomahawk (MyFaces component library)<li> Facelets<li>Seam</ul>

**Killer Web UIs by David Geary**
User interfaces are usually the most turbulent aspect of an application during development. Constant tinkering with the UI means constant changes to your code, so as a UI developer, you want to minimize the scope and effects of those code changes. Open-source Java provides two powerful software packages that help you manage UI complexity: Tiles and Sitemesh. Tiles composes webpages from discrete regions of your user interface known as tiles. A tile contains a JSP page for layout and one or more JSP pages for content. Sitemesh decorates webpages with decorators that can be associated with URL patterns. Once you set up your decorators, you can decorate pages that match a decorator's URL pattern.

**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.

**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.

**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 sample product development.

**Shale: Turbo-charge your JSF Apps by David Geary**
Struts is the most popular Java-based Web application framework today, but that's rapidly changing. There's a newcomer on the block, a leaner, meaner, better-designed framework loosely based on Struts that's poised to dethrone Struts as the reigning king of Java-based web application frameworks. That framework, of course, is JavaServer Faces. Craig McClanahan, the father of Struts and the co-spec lead for JSF 1.0, has proposed reinventing Struts for Struts 2.0 as a set of services for JSF applications. That new framework, which has no direct ties to Struts as we know it, is called Shale.

**Hibernate by Example by Eitan Suez**
This talk covers the core of the Hibernate Object/Relational Mapping framework by example; that is: in a hands-on manner.

**Extreme Agility with jMatter by Eitan Suez**
The jMatter framework is a modern implementation of the Naked Objects Architectural Pattern using Swing, Hibernate, and deployed with Java WebStart. This open-source framework produces 2-tier workgroup apps (Swing front-ends that talk to rdbms back-ends) intended to be used in a LAN or VPN environment. Developers using a Naked Objects style framework focus on building a behaviourally complete domain model and leave everything else (UI, persistence, etc) to the framework. By focusing on the domain model only, jMatter claims to offer 10x productivity for building Swing workgroup apps. Come discover jMatter in a hands-on presentation where we'll be developing a live application and hold discussions about this new empowering style of producing business applications.

**XML Data Binding with JiBX by Eitan Suez**
JiBX is an open source XML data binding API for Java. JiBX is younger than most other APIs in this space (Castor XML, BEA XMLBeans, JAXB). JiBX's philosophy on data binding is that: [a] databinding should be fast, and [b] databinding frameworks should allow for the divergence and evolution of your codebase from its xml representation. JiBX excels on both counts and consequently is a practical tool for the purpose of data binding. In this session, Eitan will be covering all aspects of Dennis Sosnoski's JiBX framework.

**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.

**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.

**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.

**Testing with Selenium by Neal Ford**
This session describes the use and workings of Selenium, the open source web user interface testing tool.

**Real-world Agile Development by Neal Ford**
Lots of developers want to use Agile development technique but don't know where to start. This session discusses how to get started with Agility, the key benefits you can expect, and the pitfalls to avoid.

**Enterprise AOP with AspectJ by Ramnivas Laddad**
Enterprise application development is a gold mine for applications of AOP. There are many crosscutting concerns found in a typical enterprise application, ranging from well-known security and transaction management to application- and technology-specific concerns. Using AOP leads to implementations that are easy to understand and easy to change.

**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.

**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.

**Performance Monitoring in J2EE Applications by Ramnivas Laddad**
J2EE has become the main new platform for enterprise application deployment. Good performance is an important business requirement. Supporting this requirement needs application profiling during the development phases and performance monitoring after application deployment. Come to this session to understand challenges and choices in monitoring J2EE applications.

**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.

**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 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.

**Spring AOP by Stuart Halloway**
Learn to use Spring AOP, aspect injection. and AspectJ integration

**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.

**Advanced Hibernate by Stuart Halloway**
Hibernate is easy to get started with, but can sometimes be hard to make efficient or secure. In fact, the default settings for Hibernate createapplications that will run slowly, cause unwanted round trips to the database, and may be more restrictive and/or permissive from a security standpointthan you would otherwise want.

**Spring Dependency Injection by Stuart Halloway**
Dependency Injection (DI) is the cornerstone of Spring. The core concept is quite simple, but (surprise!) actual practice can become complex. To take full advantage of Spring DI, you need to understand not only the basics on configuration, but also the container lifecycle model and the various hooks provided by the framework.

**Spring Security with ACEGI by Stuart Halloway**
Spring offers developers a simpler, more robust method for configuring applications. These benefits extend

to security through the ACEGI framework. ACEGI makes the otherwise daunting task of securing your application logical and straightforward. More importantly, through its support for single sign-on provision through Yale's CAS system and its ability to provide instance-level authorization, Spring extends the common security model of most J2EE apps beyond what they are traditionally capable of.

**Spring Fundamentals by Stuart Halloway**
The Spring framework is one of the fastest growing open source frameworks. New job postings are gaining rapidly, and many customers are adopting Spring instead of heavier alternatives. In this session, we#ll introduce Spring. You#ll see how Spring can give you much of the power of EJB, without the complexity or pain.

**Extend the Customization Possibilities of Your Java App with Scripts by Ted Neward**
Ever wished you could just put parts of your program in end-users' hands and let them build the infinite little changes they want? Ever thought about how you might make your application more robust by writing less code, not more? Embed a scripting engine into your application--complete with the safeguards necessary to ensure that users can't do anything they shouldn't be able to--and release yourself from the Principle of Perpetual Enslavement.

**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 enteprrise application. In this talk, we'll examine some of the trickiness, both high-level and low-.

**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.

**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.

**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 variying 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.

**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.

**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.