

Rocky Mountain Software Symposium

Denver Marriott South @ Park Meadows

November 14 - 16, 2008

<http://www.nofluffjuststuff.com/conference/denver/2008/11/index.html>

(event schedule as of November 13, 2008)

Fri, Nov. 14, 2008					
	Salon A	Salon B	Salon C	Lone Tree	Littleton
12:00 - 1:00 PM	REGISTRATION				
1:00 - 1:15 PM	WELCOME				
1:15 - 2:45 PM	Beginning Drools - Rule Engines in Java Brian Sam-Bodden	Java Memory, Performance and the Garbage Collector Ken Sipe	Evolutionary SOA Neal Ford	Applied AOP Brian Sletten	Groovy, the Blue Pill: Writing Next Generation Java Code in Groovy Scott Davis
2:45 - 3:15 PM	BREAK				
3:15 - 4:45 PM	Advanced Rules Programming with Drools Brian Sam-Bodden	Hacking - The Dark Arts Ken Sipe	Test Driven Design Neal Ford	What's Going On? : Complex Event Processing w/ Esper Brian Sletten	Groovy, The Red Pill: Metaprogramming, the Groovy Way to Blow a Buttoned-Down Java Developer's Mind Scott Davis
4:45 - 5:00 PM	BREAK				
5:00 - 6:30 PM	Professional Java UI development with the Eclipse RPC Brian Sam-Bodden	Security Code Review Ken Sipe	The Productive Programmer: Mechanics Neal Ford	The Semantic Web is Dead! Long Live the Semantic Web! Brian Sletten	Rapid Web Development with Grails and Ajax Scott Davis
6:30 - 7:15 PM	DINNER				
7:15 - 8:00 PM	Keynote: by Ken Sipe				

Sat, Nov. 15, 2008					
	Salon A	Salon B	Salon C	Lone Tree	Littleton
8:00 - 9:00 AM	BREAKFAST				
9:00 - 10:30 AM	Boosting Programmer productivity with Mylyn Brian Sam-Bodden	JavaServer Faces: A Whirlwind Tour David Geary	Spring+JPA+Hibernate: Standards Meeting Productivity for Java Persistence Ken Sipe	Real World Agile Neal Ford	Give it a REST Brian Sletten
10:30 - 11:00 AM	BREAK				
11:00 - 12:30 PM	How to Fail with 100% Code Coverage Stuart Halloway	Facelets David Geary	Spring 2.5 - Spring without XML Ken Sipe	"Design Patterns" in Dynamic Languages Neal Ford	RESTlet for the Weary Brian Sletten
12:30 - 1:30 PM	LUNCH				
1:30 - 3:00 PM	Java.next #1: Common Ground Stuart Halloway	Using Ajax4jsf David Geary	7 Habits of Highly Effective Developers Ken Sipe	Code Metrics & Analysis for Agile Projects Neal Ford	Resource-Oriented Computing w/ NetKernel : Software for the 21st Century Brian Sletten
3:00 - 3:15 PM	BREAK				
3:15 - 4:45 PM	Java.next #3: Dispatch Stuart Halloway	Intro to Seam David Geary	Architecture and Scaling Ken Sipe	Surviving Middle Management David Bock	Viva La Javolution! Brian Sletten
4:45 - 5:30 PM	BIRDS OF A FEATHER SESSION				

Sun, Nov. 16, 2008					
	Salon A	Salon B	Salon C	Lone Tree	Littleton
8:00 - 9:00 AM	BREAKFAST				
9:00 - 10:30 AM	Acceptance Testing Application Behavior Venkat Subramaniam	A Thorough Introduction To Groovy Jeff Brown	Filthy Rich Clients with the Google Web Toolkit, Part I David Geary	Git control of your source Stuart Halloway	Maintaining Project Integrity with JDepend, Macker, PMD, Maven, and other open source tools David Bock
10:30 - 11:00 AM	MORNING BREAK				
11:00 - 12:30 PM	FP on JVM Venkat Subramaniam	Agile Test Driven Development With Groovy Jeff Brown	Filthy Rich Clients with the Google Web Toolkit, Part II David Geary	Introduction to Hibernate Scott Leberknight	The Agile Product Owner David Bock
12:30 - 1:15 PM	LUNCH				
1:15 - 2:15 PM	EXPERT PANEL DISCUSSION				
2:15 - 3:45 PM	Powerful Metaprogramming Techniques With Groovy Jeff Brown	Rich Clients, Rich Data Part I : Linking Brian Sletten	Towards an Evolutionary Design Venkat Subramaniam	Real World Hibernate Tips Scott Leberknight	Estimating vs. Guessing - How Agile Teams Estimate Their Work David Bock
3:45 - 4:00 PM	BREAK				
4:00 - 5:30 PM	Grails - Agile Web 2.0 The Easy Way Jeff Brown	Rich Clients, Rich Data Part II : Consuming Brian Sletten	Caring about your Code Quality Venkat Subramaniam	Google Your Domain Objects With Hibernate Search Scott Leberknight	

Rocky Mountain Software Symposium

Denver Marriott South @ Park Meadows

November 14 - 16, 2008

<http://www.nofluffjuststuff.com/conference/denver/2008/11/index.html>
(event schedule as of November 13, 2008)

Beginning Drools - Rule Engines in Java by Brian Sam-Bodden

Drools is an open source pure-Java implementation of a forward chaining rules engine. Drools can be used in a J2SE or J2EE application and allows you to express rules programatically or by building domain specific rule languages. Learn how Business Rules with Drools can make your Java applications more flexible and robust.

Advanced Rules Programming with Drools by Brian Sam-Bodden

In this session you'll learn some of the more advanced features of Drools; a pure-Java Rule Engine. This session will walk through the construction of an advanced Rules application covering such topics as: - Fine control and monitoring of a Working Memory session - Using Decision Tables - Advanced Rule Language Features - Building Domain Specific Languages - Managing your Rules **Prerequisite:** *Beginning Drools*

Professional Java UI development with the Eclipse RPC by Brian Sam-Bodden

Learn how to build featured rich applications using the Eclipse Rich Client Platform. The Eclipse platform is an open tools platform, on top of this platform you can build your own applications (which do not need to be IDE like or IDE related). Yet you can enjoy the benefits of working with a mature and featured rich platform that can greatly reduce the amount of time required to create a professional-looking and robust Java UI application.

Boosting Programmer productivity with Mylyn by Brian Sam-Bodden

Mylyn is a task-focused toolkit for the Eclipse IDE that allows developers to focus on tasks in a way that they never been able to do before. Mylyn eliminates the constant context switching produced by typical ways IDEs are used. No more scrolling/browsing/searching/tagging/sending emails with progress updates... Mylyn provides a new way of working that allows you to focus on specific tasks by reducing information overload. Mylyn also provides a framework for integrating with the most commonly usage task tracking systems and version control systems. In this talk you'll learn how Mylyn can boost your productivity as a Java developer by letting you get the most out of your IDE.

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.

What's Going On? : Complex Event Processing w/ Esper by Brian Sletten

How well do you understand the dynamics of your applications? In our systems, we detect when simple things happen. Customers log in, people buy things, a stock is sold at a particular price, inventory shifts locations... all of these events mean little things, but what about the larger picture? Complex events are particular patterns of simpler events that suggest something deeper is happening. Do you know how you'd discover these bigger picture occurrences? Come hear how the Esper open source software represents a new class of complex event processing (CEP) frameworks that can be added to even high volume, high transaction systems.

The Semantic Web is Dead! Long Live 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 a vision of machine-processable documents and metadata to improve search, knowledge discovery and data integration and management. The only problem is that there is no such thing. There is no Semantic Web, just

the Web we have that is increasingly semantics-enabled. Forget the hype. Come learn how the technologies of this vision are being used today on the Web and in the Enterprise by more people than you might think.

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.

RESTlet for the Weary by Brian Sletten

If you have started to take a look at REST as way of exposing web services or managing information spaces, you may be frustrated by the support offered by legacy containers. There is no direct support for REST concepts in the J2EE specs (yet). XML-based configurations are so 1990's. Come learn about Restlets, a little API that has caught the attention of many in the RESTafarian community. **Prerequisite:** *Give it a REST (unless you are very comfortable with REST)*

Resource-Oriented Computing w/ NetKernel : Software for the 21st Century by Brian Sletten

Imagine the simplicity of REST married to the power of Unix pipes with the benefits of a loosely-coupled, logically-layered architecture. If that is hard to imagine, it may be because the architectures available to you today are convoluted accretions of mismatched technologies, languages, abstractions and data models. NetKernel is a disruptive technology that changes the game. It has been quietly gaining mind share in the past several years; people who are exposed to it don't want to go back to the tired and blue conventions of J2EE and .NET. Not only does it make building the kinds of systems you are building today easier, it does it more efficiently, with less code and a far more scalable runway to allow you to take advantage of the emerging multi-core, multi-CPU hardware that is coming our way. Come see how this open source / commercial product can change the way you think about building software.

Viva La Javolution! by Brian Sletten

You're a good Java programmer. You understand the JDK libraries and how to use them. The problem is that many fundamental APIs don't take the bigger performance picture in mind. Garbage collection can end up killing your app if you aren't careful. Concurrency problems and contention can keep your well-intentioned software from leveraging modern hardware architecture that support multi-core and multi-cpu systems. Who knew that simply using the standard library code the way it was designed was opening you up for performance problems in your apps? Don't worry, Javolution has your back.

Rich Clients, Rich Data Part I : Linking by Brian Sletten

You hear a lot of talk about rich clients, but the richness they purport to provide is predicated on having access to rich data as well as a rich user interaction style. Without the right levels of abstraction, it is hard to address and link all of the data we have to care about these days. Additionally, the web sites that do support the notion of linking require you to publish your data into TheirSpace. Forget that. You want to be able to link publicly available information to sensitive information in YourSpace. Ever since we started doing relational joins, we've looked for ways to tie data together. The problem is, the relational model is a bit tired and doesn't move at the speed of the Net. We need schemes that integrate relational data, web pages, XML files, RSS feeds and various other sources of information.

Rich Clients, Rich Data Part II : Consuming by Brian Sletten

You hear a lot of talk about rich clients, but the richness they purport to provide is predicated on having access to rich data as well as a rich user interaction style. Without the right levels of abstraction, it is hard to address and link all of the data we have to care about these days. Additionally, the web sites that do support the notion of linking require you to publish your data into TheirSpace. Forget that. You want to be able to link publicly available information to sensitive information in YourSpace. Ever since we started doing relational joins, we've looked for ways to tie data together. The problem is, the relational model is a bit tired and doesn't move at the speed of the Net. We need schemes that integrate relational data, web pages, XML files, RSS feeds and various other sources of information.

Surviving Middle Management by David Bock

Most good developers eventually have the opportunity to be managers. Whether they call you the "project manager", "Technical Lead", "Lead Developer", or some other classic middle-management title, you become

the 'goto' guy between management and developers. You're the guy who is expected to keep the project in-line, track a schedule, and occasionally answer the question "How's it going?", and perhaps still contribute at a technical level. So how do you do that?

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.

The Agile Product Owner by David Bock

Agile software development isn't just about the development team or managers... the customer has an active role too. The customer should be prioritizing the stories in each release, potentially working onsite in constant contact with the development team, and even participating in daily status meetings. Done well, the customer's presence has a positive influence on the development iteration. Done poorly, the customer detracts from the team's focus. So how do you be the customer of an agile team? How do you teach someone to be that customer?

Estimating vs. Guessing - How Agile Teams Estimate Their Work by David Bock

Estimating is regarded as little more than 'educated guessing', but so much can hang on the quality of those estimates. With good estimates we can set clear expectations for project delivery, but with bad estimates we can run over schedule and over budget, or worse. We often estimate when we know the least about the work that needs to get done - so how can we make the best of what is potentially a bad situation?

JavaServer Faces: A Whirlwind Tour by David Geary

In April 2005, annual growth rates for jobs in JavaServer Faces, Struts, and Ruby on Rails were all at about 0%. Today, Struts' growth rate still hovers around 0%, but JSF and Rails have taken off. At the end of 2007, both JSF and Rails were growing at a rate of between 400-500% annually (according to indeed.com). JSF has passed the adoption tipping point, and is now the Java-based framework of choice, as is evidenced by its ecosystem. From vendors such as MyEclipse and RedHat to open source projects such as Seam, Facelets, and Ajax4JSF, JSF is where the action is. Come see why JSF is so popular. In this code- and demo-intensive session, I'll show you the fundamentals of JSF. **Prerequisite:** *Some knowledge of Java-based web applications, such as Struts, is a plus, but is not required. If you have a significant experience with JSF, you probably already know most of what's covered in this session.*

Facelets by David Geary

Facelets is a combination of Tiles and Tapestry, and it's the hottest JSF-related open source project on the planet. It's popularity is well deserved, and in fact, much of what is in Facelets today will make its way into the JSF 2.0 spec due out in 2008. So not only can you come to this session and see some really cool demos that you can put to use in the real world, but you'll also be learning JSF 2.0 before it's even been defined! How's that for a ROI? **Prerequisite:** *Some knowledge of JSF is essential. If you're familiar with a templating framework, such as Velocity or Tiles, that's a plus, but not required.*

Using Ajax4jsf by David Geary

Ajax4jsf makes it very easy to add Ajax to your JSF applications. Come to this presentation to see how.

Intro to Seam by David Geary

Have you ever stopped to think that you need to learn two frameworks to develop a non-trivial, database-backed, web application? Struts and iBatis; JSF and Hibernate; Tapestry and EJB3.0.

Filthy Rich Clients with the Google Web Toolkit, Part I by David Geary

The Google Web Toolkit (GWT) is truly a revolutionary framework that lets you develop Ajaxified web applications without knowing anything about Ajax or JavaScript. But the GWT goes way beyond basic Ajax by letting you implement desktop-like applications that run in the ubiquitous browser.

Filthy Rich Clients with the Google Web Toolkit, Part II by David Geary

In the second part of this talk, you will learn how to extend the GWT by implementing custom widgets, including a scrolling viewport and a drag and drop framework. After discussing custom widgets, you will see how to integrate database access into your GWT applications, and how to deploy your GWT applications to external servers.

A Thorough Introduction To Groovy by Jeff Brown

Groovy is an agile dynamic language for the Java platform. The language and its libraries bring many things to the table to ease the process of building applications for the Java platform. This session provides a detailed run through Groovy with lots of code samples to drive home the power of the language.

Agile Test Driven Development With Groovy by Jeff Brown

Dynamic languages bring a lot of interesting elements to the table for teams interested in doing Test Driven Development (TDD). Groovy lends itself very well to TDD and this session demonstrates many features of the language and its libraries that help teams build more testable systems and build better tests.

Powerful Metaprogramming Techniques With Groovy by Jeff Brown

Metaprogramming is a key component in building truly dynamic and flexible applications with Groovy. Groovy's metaprogramming capabilities bring great new possibilities to the table that would be very difficult or just plain impossible to write with Java alone. This session will demystify a lot of the magic that seems to be going on inside of a Groovy application. *Prerequisite: A Thorough Introduction To Groovy*

Grails - Agile Web 2.0 The Easy Way by Jeff Brown

Grails is a full stack MVC framework for building web applications for the Java platform. Grails makes web application development both fun and easy. This session covers all of the fundamentals of building web applications with Grails.

Java Memory, Performance and the Garbage Collector by Ken Sipe

You are using Java, whew!!! No need to worry about memory, the garbage collector will handle that. Those who have had a memory issue in Java are not so naive any more. Often memory utilization and heap sizes are an after thought and are not recognized until the application is in production, often caused by application uptime, production request volume or production sets of data. When the OutOfMemory Error occurs, often the science of development seems to brake down and knobs are turned. First the (-mx) maximum heap space gets adjusted... More is better right. The next OutOfMemory, heads start scratching, code reviews start in earnest, and Google gets several new hits. Did you know that it is possible to get an OutOfMemory error without running out of heap space?

Hacking - The Dark Arts by Ken Sipe

A live Hacking demonstration exposing the tools and techniques used by Hackers.

Security Code Review by Ken Sipe

Security concerns abound... According to Gartner 75% of all attacks are at the web application tier. There has never been a more urgent time to understand the security concerns and how to apply solutions to our web applications.

Soft Skills and Organizational Dynamics by Ken Sipe

As you rise through the organization from developer to architect or manager, there are different skills which must be honed in order to maximize your influence. As evident by the open source community, technical leadership is more dependent on influence then it is on authority. This session will focus on skills development, dynamics in a corporate environment and community building.

Spring+JPA+Hibernate: Standards Meeting Productivity for Java Persistence by Ken Sipe

Well the standards created EntityBeans.... yea. and the community created Hibernate. Fortunately the standards body learned some lessons and created JPA. JPA requires a vendor implementation and none make a better choice then Hibernate. Combined with Spring this trio is a powerhouse when it comes to developer productivity on applications requiring persistence.

Spring 2.5 - Spring without XML by Ken Sipe

Spring 2.5 is brand spanking new, with a number of fantastic features. With growth of large and complex Spring applications which struggle with xml manageability and with the added pressure of Guice and SEAM there is a push for less XML, with solution leaning towards annotations. Spring 2.5 adds to the toolset provided in Spring 2.0 to provide a development environment where XML is greatly reduced... or eliminated if you so choose.

7 Habits of Highly Effective Developers by Ken Sipe

Thoughts lead to words, words lead to action, actions lead to habits. In this session we'll sharpen the development saw in the process of understanding what makes a hyper-productive programmer. The focus will consist of developer habits and development processes.

Architecture and Scaling by Ken Sipe

Scale... what is scale... how do you applications which are scalable. How do you know if the application scales?

Evolutionary SOA by Neal Ford

This session demonstrates that "Agility" and "SOA" complement each other quite well. Just because SOA is buzz-word compliant doesn't mean that you should throw good practices out the window. This session demonstrates how you can apply the principles of agility to building highly complex distributed enterprises.

Test Driven Design by Neal Ford

Most developers think that "TDD" stands for Test-driven Development. But it really should stand for "Test-driven Design". Rigorously using TDD makes your code much better in multiple ways.

The Productive Programmer: Mechanics by Neal Ford

Developers from the 1980s would be shocked at how inefficiently developers use their computers because of the advent of graphical operating systems. This talk describes how to reclaim productivity afforded by intelligent use of command lines and other ways of accelerating your interaction with the computer and bending computers to do your bidding. Stop working so hard for your computer!

Real World Agile by Neal Ford

There's the perfect world, and then there's the world you have to live in. Lots of organizations would like to reap the benefits of Agile development techniques but don't know how to get started. This session discusses the key benefits you can derive from Agile software development so that you can decide for yourself how many agile techniques will work within your organization.

"Design Patterns" in Dynamic Languages by Neal Ford

The Gang of Four book should have been entitled "Palliatives for Statically Typed Languages", because the recipes it provides are cumbersome solutions to the problems it poses. Using powerful languages makes the solutions in the GoF book look hopelessly complicated. This session shows how to solve the same problems concisely, elegantly, and with far fewer lines of code using the facilities of dynamic languages.

Code Metrics & Analysis for Agile Projects by Neal Ford

What does code + methodology have to do with one another? Everything! Agile projects focus on delivering working code, and tools exist to allow you to verify some quality metrics for your code. This session is a survey of tools and metrics that allow you to determine the quality of your code and strategies to "wire it" into your agile project.

Groovy, the Blue Pill: Writing Next Generation Java Code in Groovy by Scott Davis

There are wild-eyed radicals out there telling you that Java is dead, statically-typed languages are passe, and your skills are hopelessly out-of-date. Those extremists are the same ones who don't bat an eye at throwing out years of experience to learn a new language from scratch, pushing aside a familiar IDE for a new one, and deploying to a whole new set of production servers with little regard to legacy integration. While this "burn the boats" approach to software development might sound exciting to some folks, it's giving your manager the cold shakes right now. What if I told you that there was a way that you could integrate seamlessly with your legacy Java code, continue to use your trusty IDE and stable production servers, and

yet take advantage of many of the exciting new dynamic language features that those fanatics keep prattling on about? You'd probably say, "Groovy!" I would, too...

Groovy, The Red Pill: Metaprogramming, the Groovy Way to Blow a Buttoned-Down Java Developer's Mind by Scott Davis

This talk focuses on the ways that Groovy can turn a traditional Java developer's world-view upside down. We'll start by talking about how you can thumb your nose at The Man by leaving out many of the main syntactic hallmarks of Java: semicolons, parentheses, return statements, type declarations (aka Duck-typing), and the ever-present try/catch block. Then we'll look at features like operator overloading and method pointers that Groovy welcomes back into the language with open arms.

Rapid Web Development with Grails and Ajax by Scott Davis

Grails is a Java- and Groovy-based web framework that is built for speed. First-time developers are amazed at how quickly you can get a page-centric MVC web site up and running thanks to the scaffolding and convention over configuration that Grails provides. Advanced web developers are often pleasantly surprised at how easy it is to break out of that coarse-grained navigation model using the native Ajax support baked into the framework.

Introduction to Hibernate by Scott Leberknight

This session introduces the Hibernate Object/Relational Mapping (ORM) framework, showing the basics of persisting Java objects to relational databases. No prior knowledge of Hibernate or ORM is assumed.

Real World Hibernate Tips by Scott Leberknight

Hibernate is a very powerful object/relational mapping framework. With the vast amount of power also comes the responsibility to choose which features of Hibernate to use and how to use them, as well as things to avoid. We'll look at some real world Hibernate tips and tricks in this session.

Google Your Domain Objects With Hibernate Search by Scott Leberknight

Hibernate is one of the pre-eminent object/relational mapping technologies, but the Hibernate Search project adds full-text search capabilities to an already extremely capable tool to allow you to Google your domain objects.

How to Fail with 100% Code Coverage by Stuart Halloway

Over the last few years, we have taken dozens of projects to 100% coverage, and there are still plenty of things that can go wrong. We will look at examples the various problems, and show how to prevent them from infecting your project.

Java.next #1: Common Ground by Stuart Halloway

In this talk, we will explore and compare four of the most interesting new JVM languages: Clojure, Groovy, JRuby, and Scala. Each of these languages aims to greatly simplify writing code for the JVM, and all of them succeed in this mission. However, these languages have very different design goals. We will explore these differences, and help you decide when and where these languages might fit into your development toolkit. For more information see <http://blog.thinkrelevance.com/2008/8/4/java-next-common-ground>.

Java.next #3: Dispatch by Stuart Halloway

Dispatch takes many forms. Single dispatch, switch statements, pattern matching, and multiple dispatch all meet similar needs: Selecting runtime behavior in response to varying runtime conditions. Flexible dispatch is a key element of Java.next. All of the Java.next languages support dispatch strategies that are far more flexible than Java's single dispatch. In this talk (Part 3 of the Java.next series), I will explore how the Java.next languages (Clojure, Groovy, JRuby, and Scala) support dispatch.

Git control of your source by Stuart Halloway

Git is not the next step in evolution of centralized source control, following in the footsteps of cvs, svn, etc. These tools provide centralized history of deltas, where git provides distributed history of trees of content. In this talk, you will see the advantages of the git approach: Incredible speed. Local, disconnected operation.

Source control workflow customized to your team. Centralized, distributed, or layered, you can build it with git. Cheap and easy branching, tagging, and merging. Editing and refactoring your commits.

Acceptance Testing Application Behavior by Venkat Subramaniam

How do you ensure your applications meet the expectations of your key customers? In this session we will explore using the FIT tool and Behavior Driven Design tools to do exactly this.

FP on JVM by Venkat Subramaniam

Functional Programming Languages (FPLs) have been around for a long time. A lot of features that we get excited about in dynamic languages are common place in FPLs. FPLs are gaining importance due to various changes in our industry. What's exciting is that you can use them on the JVM. In this presentation we will dig into the details of what makes FPLs so interesting and look at ways to use them on the JVM?in your Java projects.

Towards an Evolutionary Design by Venkat Subramaniam

A good design is critical for success with agile development. That does not mean a big up-front design. The design has to be evolutionary. However, the design you evolve must be extensible and maintainable. After all, you can't be agile if your design sucks.

Caring about your Code Quality by Venkat Subramaniam

We all have seen our share of bad code. We certainly have come across some good code as well. What are the characteristics of good code? How can we identify those? What practices can promote us to write and maintain more of those good quality code. This presentation will focus on this topic that has a major impact on our ability to be agile and succeed.