

# CENTRAL IOWA SOFTWARE SYMPOSIUM

JUNE 9 - 10, 2023 - DES MOINES, IA

2023 NO FLUFF JUST STUFF TOUR

WORLD-CLASS TRAINING FOR SOFTWARE DEVELOPERS & ARCHITECTS

Software Architecture · Modern Java · Cloud · Docker · Microservices · Kubernetes · Kotlin · Angular · React · Vue.js · JavaScript · Machine Learning · Functional Programming · Web App Security · Spring · Testing

## Attend In-Person or Online

We are excited to come together again for in-person events! However, you will have the option to join us virtually via Zoom.

In addition, attendees will have access to video recordings of all sessions.

## Why Attend the NFJS Tour?

No Fluff Just Stuff is focused on delivering an educational experience, free from vendor bias. NFJS features the best speakers in the industry with knowledge and passion for teaching. Our iterative content is updated not only year-to-year but, week-to-week!

### Learn from the Best

Our speakers are not vendor representatives. They are industry recognized experts. They are published authors, consultants, executives, and open source leaders.

### In-Depth 90-Minute Sessions

Our longer session format, workshops, and multi-part sessions allow speakers to go in-depth and teach the detailed concepts you need to know.

### Agile Practices

Our speakers emphasize and present on topics such as: Test Driven Development, Continuous Integration, Code Quality Measurements, Code Smells, Team Building, and Customer Collaboration.

### Understand Web Security

The web is an increasingly hostile environment for web applications. The NFJS Tour includes security focused sessions and workshops so you will understand best security practices.

### Develop Your Soft Skills

Effective engineers need significant technical depth and breadth and domain knowledge. In addition, there is another perhaps more vital aspect of being an architect - the soft skills. Communication, leadership, persuasion, and more.

### Cloud Architectures

The NFJS Tour explores different cloud computing architectures and how you can take advantage of them.

<https://nofluffjuststuff.com>

# Central Iowa Software Symposium

## -Session Schedule-

(event schedule as of May 17, 2024)

### Friday, Jun. 9

7:30 - 8:15 AM : REGISTRATION/BREAKFAST - BALLROOM FOYER

8:15 - 8:30 AM : WELCOME - BENTON BALLROOM

8:30 - 10:00 AM - Sessions

#### **Session #1 @ POLK : Know your Java? by Venkat Subramaniam**

Many of us have significant experience in Java. Yet, from time to time, we get tripped up by some code that we quite did not expect to behave the way it does.

#### **Session #2 @ DALLAS : Hypermedia and the rest of REST by Michael Carducci**

REST is, undoubtedly one of the most maligned and misunderstood terms in our industry today. So many different things have been called REST, that the world has virtually lost all meaning. Many systems and applications that self-describe as RESTful; usually are not, at least according to REST as defined in Dr. Roy T. Fielding's 2000 Dissertation, Architectural Styles and the Design of Network-based Software Architectures. The wild success of the architecture derived by Dr. Fielding led many to want to emulate it (even when it was inappropriate to do so). As a shorthand, organizations began referring to RESTful systems, which exposed RESTful APIs. Over time RESTful became a buzzword referring to a vague generalization of HTTP/json APIs that typically bear little to no resemblance to the central ideas of REST (and thus elicit few of the benefits). Hypermedia is the central pillar and defining characteristic of the REST architectural style yet it remains almost universally absent.

#### **Session #3 @ HANCOCK : What's New in Spring 6 and Spring Boot 3 by Craig Walls**

In this example-driven session, we'll review several tips and tricks to make the most out of your Spring development experience. You'll see how to apply the best features of Spring and Spring Boot, including the latest and greatest features of Spring Framework 6 and Spring Boot 3.

#### **Session #4 @ SHELBY : Microservices and Domain-Driven Design The strategic parts by Raju Gandhi**

We live in a world of microservices. Yet, what is a microservice? What defines the boundaries of a microservice? How do we define the relationships between microservices? Thankfully domain-driven design gives us the concepts and practices to better design and decompose our services.

#### **Session #5 @ WOODBURY : Machine Learning Data Pipelines by Daniel Hinojosa**

How do we move information realtime and connect machine learning models to make decisions on our business data? This presentation goes through machine learning and Kafka tools that would help achieve that goal.

10:00 - 10:30 AM : BREAK: BALLROOM FOYER

10:30 - 12:00 PM - Sessions

#### **Session #6 @ POLK : Non-Language Changes in Java by Venkat Subramaniam**

Java has been evolving at a rapid pace. Some of the most noticed changes are the features in the language. However, there are other interesting and significant changes in Java, not related to the language but in the JDK and the ecosystem.

#### **Session #7 @ DALLAS : The Linked Data Revolution is here, are you ready? by Michael Carducci**

The web is arguably the single most impactful revolution in human history (to date). By agreeing on a simple set of standards, we have collectively unlocked all the world's information. Documents can be discovered, retrieved, published, and shared so easily we don't even think about it. Data, on the other hand, is a different story. Our data remains stuck in the 1980s. Locked in silos, each with a different format, interface, and conventions that must be interpreted by a human, parsed, mapped, and converted. Data is at the heart of many problems we solve today, and we produce data exponentially faster than we can consume it.

#### **Session #8 @ HANCOCK : Spring into K8s: Deploying Spring Application in Kubernetes by Craig Walls**

Although Java originally promised write once, run anywhere, it failed to fully deliver on that promise. As developers, we can develop, test, and build our applications into WAR or executable JAR files and then toss them over the wall to a Java application server and Java runtime that we have no control over, giving us zero confidence that the application will behave the same as when we tested it. Containers fulfill the write-once, run anywhere promise that Java wasn't able to, by packaging the runtime and even the operating system along with our application, giving greater control and confidence that the application will function the same anywhere it is run. Additionally, containers afford several other benefits, including easy scaling, efficiency in terms of resource utilization, and security by isolating containers from their host system and from other containers. While deploying Spring applications in containers has always been possible, Spring Boot makes it easier to containerize our applications and run them in container architectures such as Kubernetes. Spring Boot's support for containerization includes two options: Creating containers based on buildpacks or using layers as a means of modularizing and reducing the size of our application deployments. Moreover, new components in the Spring ecosystem can make your Spring applications Kubernetes-savvy so that they can take advantage of what a containerized architecture has to offer.

# Central Iowa Software Symposium

## -Session Schedule-

(event schedule as of May 17, 2024)

### **Session #9 @ SHELBY : Domain-Driven Design - Where Rubber meets the road by Raju Gandhi**

In this session we will go further with DDD, considering aspects like aggregates, domain events, factories and repositories, and consider how DDD can be represented in architecture, be that hexagonal, REST or RPC.

### **Session #10 @ WOODBURY : Next Gen Testing Tools for Java by Daniel Hinojosa**

We have been using JUnit and doing TDD for years, but you can take testing further. In this session, we will discuss some tools you absolutely need for testing your code outside of the regular stack you currently use.

12:00 - 1:00 PM : LUNCH: BENTON BALLROOM

1:00 - 2:30 PM - Sessions

### **Session #11 @ POLK : Asynchronous Programming in Java using Virtual Threads by Venkat Subramaniam**

Threads have been part of Java since the beginning. But, the new virtual threads, introduced as prerelease in Java 19, are different in how they're implemented and how we can benefit from them.

### **Session #12 @ DALLAS : The Linked Data Revolution II - Tools, patterns, and practices by Michael Carducci**

Part one of this series introduces the ideas, motivations, and applications of linked data along with historical context. This more technical session dives deeper into the tech stack and available tooling.

### **Session #13 @ HANCOCK : Reactive Spring APIs by Craig Walls**

In this example-driven presentation, we'll focus on how to build reactive APIs in Spring. We'll start with Spring WebFlux, a reactive reimagining of the popular Spring MVC framework for HTTP-based APIs. Then we'll have a look at RSocket, an intriguing new communication protocol that is reactive by design.

### **Session #14 @ SHELBY : Measuring your architecture by Raju Gandhi**

It's not just architecture's evolutionary architecture. But to evolve your architecture, you need to measure it. And how does that work exactly? How does one measure something as abstract as architecture? In this session we'll discuss various strategies for measuring your architecture. We'll see how you know if your software architecture is working for you, and how to know which metrics to keep an eye on. We'll also see the benefits of measuring your architecture.

### **Session #15 @ WOODBURY : Kubernetes Immersion by Daniel Hinojosa**

This is an introductory workshop to get started with Kubernetes. This assumes that you, the attendee, are fresh to this technology. This workshop begins with defining terms, deploying Kubernetes objects, tracking the health of your deployments, and exposing your application to the outside world. Our end goal is to have you deploy a full application on the internet.

2:30 - 2:45 PM : BREAK: BALLROOM FOYER

2:45 - 4:15 PM - Sessions

### **Session #16 @ POLK : Structured Concurrency in Java by Venkat Subramaniam**

A prerelease feature in Java, structure concurrency is going to change how we do concurrent programming.

### **Session #17 @ DALLAS : VDD: Value Driven Development - 10 Golden Rules for incremental Greatness by Michael Carducci**

On the NFJS tour, there are questions that seem to come up again and again. One common example is "How do we determine which new tools and technologies we should focus our energy on learning?" another is "How do we stop management from forcing us to cut corners on every release so we can create better and more maintainable code?" which, after awhile becomes "How can we best convince management we need to rewrite the business application?" There is a single meta-answer to all these questions and many others.

### **Session #18 @ HANCOCK : Reactive Spring Persistence by Craig Walls**

In this example-driven presentation, we'll focus on working with reactive data persistence. We'll start by seeing how to create reactive repositories for relational databases with Spring Data R2DBC. Then we'll explore non-relational reactive persistence for MongoDB and Cassandra.

### **Session #19 @ SHELBY : Advanced Git by Raju Gandhi**

You have been using Git for a while. You know how to stage and commit your work, create and delete branches and collaborate with your team members using remotes. But Git often leaves you confused "ever committed to your work to the wrong branch? Even worse, ever accidentally delete a branch that you needed to keep around? And what is God's good name is "Detached HEAD state"? Why tag commits, when we have branches? Is there a better work-flow than just using merges?

# Central Iowa Software Symposium

## -Session Schedule-

(event schedule as of May 17, 2024)

What's the difference between a merge and a rebase? The answer to all of these questions, and more, lies in the constitution of a commit, and the directed acyclic graph (DAG) that Git uses to manage your history. This, right here, is the key to understanding everything in Git.

### **Session #20 @ WOODBURY : Kubernetes Immersion (continued) by Daniel Hinojosa**

This is an introductory workshop to get started with Kubernetes. This assumes that you, the attendee, are fresh to this technology. This workshop begins with defining terms, deploying Kubernetes objects, tracking the health of your deployments, and exposing your application to the outside world. Our end goal is to have you deploy a full application on the internet.

4:15 - 4:30 PM : BREAK: BALLROOM FOYER

4:30 - 6:00 PM - Sessions

### **Session #21 @ POLK : Pattern Matching in Java by Venkat Subramaniam**

The switch feature of Java has gone through an amazing transformation. In this presentation will start with switch as a statement, transform from there to an expression, and into a full blown pattern matching syntax.

### **Session #22 @ DALLAS : Personal Knowledge Management - Second Brain Methods and Madness by Michael Carducci**

We are knowledge workers and ultimately, we must own our growth and learning. [Personal Knowledge Management]([https://en.wikipedia.org/wiki/Personal\\_knowledge\\_management](https://en.wikipedia.org/wiki/Personal_knowledge_management)) is a process of collecting information that one uses to gather, classify, store, search, retrieve and share knowledge in their daily activities and the way in which these processes support work activities. Despite taking notes, bookmarking web content, and highlighting passages in books; often we struggle to recall or rediscover these many insights we pick up daily in our work and life. This session introduces a tool and some process recommendations to never again lose discoveries and knowledge resources.

### **Session #23 @ HANCOCK : Spring Graph QL by Craig Walls**

In this example-driven session, we're going to look at how to implement GraphQL in Spring. You'll learn how Spring for GraphQL builds upon GraphQL Java, recognize the use-cases that are best suited for GraphQL, and how to build a GraphQL API in Spring.

### **Session #24 @ SHELBY : Advanced Git (continued) by Raju Gandhi**

You have been using Git for a while. You know how to stage and commit your work, create and delete branches and collaborate with your team members using remotes. But Git often leaves you confused &#8212; ever committed to your work to the wrong branch? Even worse, ever accidentally delete a branch that you needed to keep around? And what is God's good name is "Detached HEAD state"? Why tag commits, when we have branches? Is there a better work-flow than just using merges? What's the difference between a merge and a rebase? The answer to all of these questions, and more, lies in the constitution of a commit, and the directed acyclic graph (DAG) that Git uses to manage your history. This, right here, is the key to understanding everything in Git.

### **Session #25 @ WOODBURY : Do more with JReleaser by Daniel Hinojosa**

Hey. Remember that time when we used to create jar or war files and we used to just ssh into a box and deploy on a single box? Well, it was simpler but also maybe that wasn't that great of an idea. Time has certainly moved on, and our releases have become very advanced with very technical CI/CD pipelines, docker or debian packages, multi-purpose testing, producing signatures, perform security scans, perform releases, and then when you're done tell the whole world about it. Whew! This presentation introduces JReleaser, a release platform for Java that does a multitude of chores for you.

6:00 - 7:00 PM : DINNER: BENTON BALLROOM

7:00 - 8:00 PM : Keynote: Sufficiently Advanced Technology, Software & Soothsaying - Michael Carducci

## **Saturday, Jun. 10**

7:30 - 8:15 AM : BREAKFAST: BALLROOM

8:15 - 9:45 AM - Sessions

### **Session #26 @ POLK : Designing Microservices: From Architecting to Data Modeling by Venkat Subramaniam**

How does designing Microservices differ from designing more traditional applications? What is a better way to learn than to take a problem, analyze the requirements, explore the design options, apply the concepts of bounded context, and arrive at the architecture and design of Microservices to realize the requirements?

# Central Iowa Software Symposium

## -Session Schedule-

(event schedule as of May 17, 2024)

### **Session #27 @ DALLAS : The Influential Engineer - Overcoming resistance to change by Michael Carducci**

By the end of this conference you will have learned many new tools and technologies. The easy part is done, now for the hard part: getting the rest of the team-and management-on board with the new ideas. Easier said than done. Whether you want to effect culture change in your organization, lead the transition toward a new technology, or are simply asking for better tools; you must first understand that having a "good idea" is just the beginning. How can you dramatically increase your odds of success? You will learn 12 concrete strategies to build consensus within your team as well as 6 technique to dramatically increase the odds that the other person will say "Yes" to your requests.

### **Session #28 @ HANCOCK : Knowledge Graphs: Architecture and Information by Brian Sletten**

Knowledge graphs are a rapidly emerging concept for machine-processable models of complex and dynamic domains. They represent the intersection of Web architecture and information. If your organization wants to resolve its most pernicious data integration problems or facilitate machine learning initiatives, knowledge graphs are likely to be part of your future.

### **Session #29 @ SHELBY : A Vue perspective - Web Apps with Vue.js &#8212;&#160;Part I by Raju Gandhi**

In this session we will build a full application using Vue.js. We will start by discussing how you can start working with Vue, all the way to seeing what it takes to build an app with Vue, including state management and routing. **\*\*Note\*\*:** We'll be covering Vue version 3

### **Session #30 @ WOODBURY : Kafka Fundamentals by Daniel Hinojosa**

Kafka is a "must know." It is the data backplane of the modern microservice architecture. It's now being used as the first persistence layer of microservices and for most data aggregation jobs. As such, Kafka has become an essential product in the microservice and big data world.

9:45 - 10:00 AM : BREAK: BALLROOM FOYER

10:00 - 11:30 AM - Sessions

### **Session #31 @ POLK : Designing Microservices: From Architecting to Data Modeling (continued) by Venkat Subramaniam**

How does designing Microservices differ from designing more traditional applications? What is a better way to learn than to take a problem, analyze the requirements, explore the design options, apply the concepts of bounded context, and arrive at the architecture and design of Microservices to realize the requirements?

### **Session #32 @ DALLAS : Influential Engineer Part 2 - Persuasion Patterns by Michael Carducci**

In Part 1, you learned the core principles of influence and persuasion. How to we take this back to the office and apply what we've learned?

### **Session #33 @ HANCOCK : IPFS : Architecture and Decentralization by Brian Sletten**

Decentralization and Content-based addressing represent a significant advancement in the development of stable, scalable, censorship-resistant systems. They require a remarkable amount of architectural thinking to work effectively. The Interplanetary File System (IPFS) is an umbrella project covering a cornucopia of extremely well designed layers that will prop up and extend the Web in many new directions. Come here about a future that looks a little bit like combining the Web with Git, Bittorrent, Self-certifying file systems, Distributed Hash Tables and more.

### **Session #34 @ SHELBY : A Vue perspective - Web Apps with Vue.js &#8212;&#160;Part II by Raju Gandhi**

In this session we will build a full application using Vue.js. We will start by discussing how you can start working with Vue, all the way to seeing what it takes to build an app with Vue, including state management and routing. **\*\*Note\*\*:** We'll be covering Vue version 3

### **Session #35 @ WOODBURY : Kafka Fundamentals (continued) by Daniel Hinojosa**

Kafka is a "must know." It is the data backplane of the modern microservice architecture. It's now being used as the first persistence layer of microservices and for most data aggregation jobs. As such, Kafka has become an essential product in the microservice and big data world.

11:30 - 12:15 PM : EXPERT PANEL DISCUSSION: BENTON BALLROOM

12:15 - 1:00 PM : LUNCH: BENTON BALLROOM

1:00 - 2:30 PM - Sessions

### **Session #36 @ POLK : Refactoring Code: An Incremental and Purpose Driven Approach by Venkat Subramaniam**

Continuous refactoring is critical to succeeding in projects and is an important part of sustainable agile development.

# Central Iowa Software Symposium

## -Session Schedule-

(event schedule as of May 17, 2024)

### **Session #37 @ DALLAS : An Architect's Approach to API Strategies by Michael Carducci**

Integration, once a luxury, is now a necessity. Doing this well, however, continues to be elusive. Early attempts to build better distributed systems such as DCOM, CORBA, and SOAP were widely regarded as failures. Today the focus is on REST, RPC, and graphql style APIs. Which is best? The go-to answer for architects is, of course, "it depends."

### **Session #38 @ HANCOCK : Machine Learning Workshop by Brian Sletten**

Machine Learning is all the rage, but many developers have no idea what it is, what they can expect from it or how to start to get into this huge and rapidly-changing field. The ideas draw from the fields of Artificial Intelligence, Numerical Analysis, Statistics and more. These days, you'll generally have to be a CUDA-wielding Python developer to boot. This workshop will gently introduce you to the ideas and tools, show you several working examples and help you build a plan to for diving deeper into this exciting new field.

### **Session #39 @ SHELBY : Container Security Fundamentals Part I by Raju Gandhi**

Containers are everywhere. Of course, a large part of the appeal of containers is the ease with which you can get started. However, productionizing containers is a wholly different beast. From orchestration to scheduling, containers offer significantly different challenges than VMs. In particular, in terms of security. Securing and hardening VMs is \_very\_ different than that for containers. In this two-part session, we will see what securing containers involves.

### **Session #40 @ WOODBURY : Java Design Patterns Deep Dive by Daniel Hinojosa**

Since 1994, the original Gang of Four Design Patterns book, "Design Patterns: Elements of Reusable Object-Oriented Software" has helped developers recognize common patterns in development. The book was originally written in C++, but there have been books that translate the original design patterns into their preferred language. One feature of "The Gang of Four Design Patterns" that has particularly stuck with me has been testability for the most part. With the exception of singleton, all patterns are unit testable. Design Patterns are also our common developer language. When a developer says "Let's use the Decorator Pattern" we know what is meant.

2:30 - 2:45 PM : BREAK: BALLROOM FOYER

2:45 - 4:15 PM - Sessions

### **Session #41 @ POLK : Refactoring Code: An Incremental and Purpose Driven Approach (continued) by Venkat Subramaniam**

Continuous refactoring is critical to succeeding in projects and is an important part of sustainable agile development.

### **Session #42 @ DALLAS : Finding Signal in the Noise: The art of Execution by Michael Carducci**

In tech teams it's a constant firefight. We react. Then we react to the reaction... the cycle continues. In all this noise, in all this chaos, how do we move forward. How do we remain proactive?

### **Session #43 @ HANCOCK : Machine Learning Workshop (continued) by Brian Sletten**

Machine Learning is all the rage, but many developers have no idea what it is, what they can expect from it or how to start to get into this huge and rapidly-changing field. The ideas draw from the fields of Artificial Intelligence, Numerical Analysis, Statistics and more. These days, you'll generally have to be a CUDA-wielding Python developer to boot. This workshop will gently introduce you to the ideas and tools, show you several working examples and help you build a plan to for diving deeper into this exciting new field.

### **Session #44 @ SHELBY : Container Security Fundamentals Part II by Raju Gandhi**

Containers are everywhere. Of course, a large part of the appeal of containers is the ease with which you can get started. However, productionizing containers is a wholly different beast. From orchestration to scheduling, containers offer significantly different challenges than VMs. In particular, in terms of security. Securing and hardening VMs is \_very\_ different than that for containers. In this two-part session, we will see what securing containers involves.

### **Session #45 @ WOODBURY : Java Design Patterns Deep Dive (continued) by Daniel Hinojosa**

Since 1994, the original Gang of Four Design Patterns book, "Design Patterns: Elements of Reusable Object-Oriented Software" has helped developers recognize common patterns in development. The book was originally written in C++, but there have been books that translate the original design patterns into their preferred language. One feature of "The Gang of Four Design Patterns" that has particularly stuck with me has been testability for the most part. With the exception of singleton, all patterns are unit testable. Design Patterns are also our common developer language. When a developer says "Let's use the Decorator Pattern" we know what is meant.

4:15 - 4:30 PM : BREAK: BALLROOM FOYER

4:30 - 6:00 PM - Sessions

# Central Iowa Software Symposium

## -Session Schedule-

(event schedule as of May 17, 2024)

### **Session #46 @ POLK : Let's Have Some Fun with Game Of Life: And Learn to Think Functionally Along the Way by Venkat Subramaniam**

Game of Life is an intriguing game. At first look it looks simple, but as you look closer, it appears to be quite complex. How can we implement this game with different constraints, what are the constraints? Is it possible to use functional programming for this, to honor immutability? You see, it is intriguing.

### **Session #47 @ DALLAS : Mob Mentality - Collaborative coding and problem solving by Michael Carducci**

Mob Programming is a style of programming in which the entire team sits together and works on a single task at a time. Teams that have worked this way have found that many of the problems that plague normal development just melted away, possibly because communication and learning increases. Teams also find that the quality of their code increases. They find their capacity to create increases. However, the best part of all this is that teams end up being happier and more cohesive.

### **Session #48 @ HANCOCK : Modern Software Systems : Succeeding at IT in the 21st Century by Brian Sletten**

In the last 30 years, our industry has been upended by advancements that unlock previously unimaginable capabilities. It still seems like there is far too much failure and not enough success in IT systems though. To be successful in the 21st Century, you will need to understand where we are and where we are going. It is a complex amalgamation of developments in hardware, computer languages, architectures and how we manage information. Very few people understand all of the pieces and how they connect.

### **Session #49 @ SHELBY : On being an effective developer by Raju Gandhi**

As developers we not only operate in different contexts, but also often have these different contexts interplay as part of our work. Each of the tools that we use ; version control systems like Git (along with collaborative tools like Github/Gitlab), IDE's like Eclipse/IntelliJ, build systems like Gradle, Ci/Cd tooling like Jenkins, IaaS tools like Ansible, the command line ; all introduce context. To be effective developers we need to know when to operate in a certain context, combine or tease apart how these contexts interplay. Can you improve your release announcements if format your commit messages consistently? You bet! How should your build tool interact with your version control system? What does naming your files have to do with how you use your IDE?

### **Session #50 @ WOODBURY : IntelliJ Dojo by Daniel Hinojosa**

Remember in the Matrix, when Neo said "I know Kung Fu", and then Morpheus said "Show me", well we will be doing that except with IntelliJ. In this dojo, we will be using all the wonderful keymappings that are available in IntelliJ and we will make you a lean mean coding machine!