PreCompiler Session 01 - Tuesday 08:00

7 Languages in 7 Hours

Amber Conville

Hands-on experience in seven different languages, ranging from object-oriented to functional, from the cozy and familiar to the eye-squintingly terse and foreign. You'll work through a familiar kata in each of them, showing you how to approach an entirely new language. You'll learn about the strengths and weaknesses of each language out in the real world. You'll also see how each language's quirks can teach us ways to improve the code we write every day.

You may not come out of this pre-compiler an expert in all of these languages, but you'll have learned a lot about how to get started with a new one. You may even discover a new passion! At the very least, the next time a new language comes along, you'll have the tools you need to tackle it, and enough knowledge to help you push past the "what is this syntax even doing" barrier.

Languages: Ruby, Clojure, Haskell, Rust, Scala, Elixir, Go

Bring your Big Data to life using JavaScript and a browser using D3.js

Dan Shultz, Doug Mair

In this session, we will see how to use JavaScript frameworks to bring your big data sets to life in the browser. We'll first walk through the very basics of chart creation using D3, then using a dataset of meteorite strikes on the earth, we will build data visualizations and dashboards to track the meteorite strikes. We will use HTML, CSS, SVG, D3.js and DC.js to build the dashboard.

We will then move into using the Cross Filter JavaScript library, to add cross-chart interactivity and let the users explore the data in ways that we may not have anticipated. Lastly, we will use the latest version of D3.js (V4) and build even more complex visualizations and interactions.

Just bring a laptop with a Chrome browser to participate. All of the technology will be presented using open source frameworks with strong community support.

Commutication: Moving Ideas with Words

damian synadinos

“By words we learn thoughts, and by thoughts we learn life.” – Jean Baptiste Girard

How are concepts conveyed? How are thoughts shared? One way is by using words. But, which words should you use? And how can you tell if your words effectively convey your thoughts? What causes miscommunication, what are its effects, and how can you avoid it? And what are the benefits and limitations of a shared language?

In this workshop, we will move from general to specific as we collaboratively attempt to understand and answer these questions, and many more. Starting with the definition of definition, we will investigate words and meaning, their properties, and their relationship. We will then examine how miscommunication can occur, look at some potential consequences of it, and explore some methods to minimize it. Finally, we will apply these ideas as we attempt to define some common terms and create a “common language”. In the end, attendees will gain new knowledge and tools to help them more effectively transfer ideas with words.

Workshop Takeaways:

- Deeper understanding of the meaning of words and meaning.
- Some causes and effects of miscommunication and practical techniques to avoid it.
- The difficulty of creating a shared language, even in a microcosm.
- Ways to better understand others, and be better understood.

Game Development with the Unity Game Engine

Mike Geig

You want to make video games with the Unity game engine? Let's make video games with the Unity game engine! In this session attendees will build a simple video game from start to finish. Covered is a hands on approach to the construction of game systems from assets. Audience members won't be watching, they'll be doing. The end product will be a complete video game playable on desktop, web, or mobile devices. This is all new content showing you how to use the latest features of the engine. Even if you attended last year there will still be plenty of new things to learn!

Hacking & Hardening Java Web Applications Workshop

Christopher Judd

It seems like everyday there is a new headline about a security breach in a major company's web application. These breaches cause companies to lose their credibility, cost them large sums of money, and those accountable undoubtedly lose their jobs. Security requires you to be proactive. Keep your employer out of the headlines by learning some key security best practices.

This hands-on workshop is designed to teach you how to identify and fix vulnerabilities in Java web applications. Using an existing web application, you will learn ways to scan and test for common vulnerabilities such as hijacking, injection, cross-site scripting, cross-site forgery and more. You will learn best practices around logging, error handling, intrusion detection, authentication and authorization. You will also learn how to improve security in your applications using existing libraries, frameworks and techniques to patch and prevent vulnerabilities.

Introduction to Angular 2 (Part 1)

Jonathan DeJong

Angular 2 is the long awaited sequal to the incredibly popular AngularJS. This pre-compiler is a hands on introduction to Angular 2. This is a hand on, lab based approached to getting up to speed. Each lab consists a demonstration of one or more concepts, followed by a starting point. Solutions to each lab are also provided. We will go over everyting that a developer will need to be productive in Angular 2, including components, directives, pipes, services, dependency injection, as well as overviews of RxJS and TypeScript.

Money Makes Your App Go Round

Noel Rappin

Your customers have money, and you'd like them to give it to you. Payment gateways, such as Stripe, Braintree, and Paypal, make it easy to start charging credit cards and get the money flowing. But charging cards is only the beginning. You need to worry that your app responds gracefully to service failures, since charging a customer for a failed transaction is bad. You need to guard against fraud and security breaches. You need administrative tools that are flexible but secure. You want to test against external services. And you'll run up against the law. Learn from some of my mistakes and build a robust financial application.

In this workshop, attendees will use the Stripe API to start taking credit card payments, but they will also build a robust system around the API call, including data management, failure mitigation, security, and basic reporting.

Nodebots Workshop (Session 1)

Rachel Weil, Brian Sherwin

The NodeBots are back by popular demand! Join Microsoft for a hands-on experience building a robot powered by JavaScript. We'll use the Adafruit Feather HUZZAH microcontroller and the Johnny-Five JavaScript library to build a cloud-connected bot. Then, we'll customize the bot with sensors, lights, and other electronic goodies for a special challenge to be announced. This is a workshop you won't want to miss!

NodeCopters - controlling drones with the power of JavaScript (Session 1)

David Resseguie

Have you ever wanted to command your own spaceship? Demonstrate your air superiority to all your friends? Or maybe you're just looking to take your land-based NodeBots skills from prior CodeMash sessions to the next level. Whatever your reasoning, join us as we form teams of developers to hack our fleet of drones to race, complete challenges, or just show off. Don't know JavaScript? You're welcome to use any programming library that can interact with the Parrot AR drones. But you're on your own and solo space travel can be a lonely place.

Xamarin is Free! - Now What Do We Do?

Jason Awbrey

With Microsoft's acquisition of Xamarin, the interest level from the community spiked. Now included with Visual Studio, Xamarin is more acessible than ever. So what's the next step? How do you get started? What are the pitfalls you should know about? What exactly is Xamarin?

In this workshop we'll talk about what Xamarin is and what it is NOT. We'll look at the different technologies available under the Xamarin umbrella, and most importantly we'll write some code! Bring your laptop and we'll walk through the steps neceesary to build a working Xamarin app, starting from a blank solution.

PreCompiler Session 02 - Tuesday 13:00

7 Languages in 7 Hours (Part 2)

Amber Conville

Hands-on experience in seven different languages, ranging from object-oriented to functional, from the cozy and familiar to the eye-squintingly terse and foreign. You'll work through a familiar kata in each of them, showing you how to approach an entirely new language. You'll learn about the strengths and weaknesses of each language out in the real world. You'll also see how each language's quirks can teach us ways to improve the code we write every day.

You may not come out of this pre-compiler an expert in all of these languages, but you'll have learned a lot about how to get started with a new one. You may even discover a new passion! At the very least, the next time a new language comes along, you'll have the tools you need to tackle it, and enough knowledge to help you push past the "what is this syntax even doing" barrier.

Languages: Ruby, Clojure, Haskell, Rust, Scala, Elixir, Go

Apple Watch Development with Swift

Jeff Kelley

Apple Watch development is pretty different from other mobile development. From the fact that screens are *tiny* to the fact that every watchOS app must be embedded inside of an iOS app, there's a lot to think about before writing a watch app. In this workshop, we'll cover getting started with your watch app, communicating between the Apple Watch and iPhone, making network requests from the watch directly, compiling code and frameworks for both platforms, and more! Bring your Mac, your ideas, and Xcode 8 and we'll create some great watch apps! Although not required, bring an Apple Watch and iPhone if you have them to get your app up and running on a real device!

Behavior Driven Web Application Security

Aaron Bedra

Defending web applications has become an increasingly difficult task. Learn how commonly missed indicators of attack go unnoticed and what you can do to separate the signal from the noise.

This workshop will show you how to analyze the traffic on your web application and turn it into actionable information. With this information we will build an environment capable of responding to the analysis in real time and stopping bad actors before they cause damage. The session will be completely hands on and leave attendees with a working prototype of a fully automated web application defense system.

Game Development with the Unity Game Engine (Part 2)

Mike Geig

You want to make video games with the Unity game engine? Let's make video games with the Unity game engine! In this session attendees will build a simple video game from start to finish. Covered is a hands on approach to the construction of game systems from assets. Audience members won't be watching, they'll be doing. The end product will be a complete video game playable on desktop, web, or mobile devices. This is all new content showing you how to use the latest features of the engine. Even if you attended last year there will still be plenty of new things to learn!

Improv(e) Your Testing! Tips & Tricks from Jester to Tester

damian synadinos

Improvisational comedy—sometimes called improv—is a form of theater in which the performance is created in the moment. Successful improv involves learning and using a variety of skills and techniques which allow performers to quickly adapt to a constantly changing environment and new information. Now reread the previous sentence, but replace the word improv with testing. In many ways, improv is a great analogy for testing. As both an experienced improviser and tester, Damian Synadinos presents some of the many similarities between improv and testing. After setting expectations, Damian takes a closer look at the definitions of improv and testing to better understand their purpose and value. He then thoroughly explains and demonstrates numerous improv tips and tricks with help from the audience, and shows how the very same ideas can be useful in a testing context. Using creative metaphors and critical analysis, old ideas about testing are reframed in novel and notable ways. Whether novice or experienced, you will laugh, learn, and leave with ways to help improv(e) your testing.

Introduction to Angular 2 (Part 2)

Jonathan DeJong

Angular 2 is the long awaited sequal to the incredibly popular AngularJS. This pre-compiler is a hands on introduction to Angular 2. This is a hand on, lab based approached to getting up to speed. Each lab consists a demonstration of one or more concepts, followed by a starting point. Solutions to each lab are also provided. We will go over everyting that a developer will need to be productive in Angular 2, including components, directives, pipes, services, dependency injection, as well as overviews of RxJS and TypeScript.

Nodebots Workshop (Session 2)

Rachel Weil, Brian Sherwin

The NodeBots are back by popular demand! Join Microsoft for a hands-on experience building a robot powered by JavaScript. We'll use the Adafruit Feather HUZZAH microcontroller and the Johnny-Five JavaScript library to build a cloud-connected bot. Then, we'll customize the bot with sensors, lights, and other electronic goodies for a special challenge to be announced. This is a workshop you won't want to miss!

NodeCopters - controlling drones with the power of JavaScript (Session 2)

David Resseguie

Have you ever wanted to command your own spaceship? Demonstrate your air superiority to all your friends? Or maybe you're just looking to take your land-based NodeBots skills from prior CodeMash sessions to the next level. Whatever your reasoning, join us as we form teams of developers to hack our fleet of drones to race, complete challenges, or just show off. Don't know JavaScript? You're welcome to use any programming library that can interact with the Parrot AR drones. But you're on your own and solo space travel can be a lonely place.

Science of Great UI

Mark Miller

Get a big boost on your UI skills. If you believe you're not an artist, that UI is merely subjective, or that great design takes too much effort, then this workshop is for you. Learn the essence with simple, easy-to-retain guidelines. Regardless of whether you're building interfaces for watches, mobile, tablets, web, desktops, elevators, trains, planes, automobiles, HoloLens, or interplanetary spaceships; you'll learn how to reduce visual noise, enhance clarity, widen the path, and design interfaces that are a pleasure to use.

Learn what's really happening inside the brain and how human biology drives important design guidelines. Get a grip on cognitive load, saccades & fixations, peripheral blindness, grouping, granularity, channel limitations, cellular data compression, and symbol recognition. Conduct mind experiments, live in the session, to derive crucial design guidelines for creating better user interfaces.

These guidelines will help you present any information -- any data in any medium -- with clarity. We'll learn how to work with foreground, background, color, size, shape, and emphasis to reduce noise, boost signal, and achieve clarity in your software designs. This workshop is loaded with real-life examples from both the virtual and physical worlds, and includes groundbreaking research and exclusive content.

We'll also explore dark design patterns and the business case behind great usability. We'll learn how to measure the cost of bad UI, and how to present compelling cases for great design to decision makers. Ultimately it's all about making customers satisfied, and this entertaining and information-packed workshop will show you how.

Unleash the computer on your data

Amber McKenzie

Wondering why your magic 8 ball keeps telling you the same things when you ask it about your data? Heard about this newfangled thing called machine learning? Want to get in on that but don't know where to start? This hands-on session will set you on the path from being Miss Cleo to becoming Nate Silver. Attendees will be given an overview of the steps involved in machine learning tasks and will implement several real-world applications, including predictive modeling and natural language processing problems, using Python and scikit-learn.

PreCompiler Session 03 - Wednesday 08:00

3D Modeling for Makers and Game Developers!

Robert Palmer

Just bought a 3D printer and want to design custom parts for your robot army? Are you a developer looking to expand from curly braces and semicolons with Icospheres and UV Maps? Are you a maker ready to move up from browsing Thingiverse and on to creating your own content? This hands-on session will provide you with the tools and techniques to make this happen using Blender, a free, open source software package for 3D art and animation. Topics will include basic 3D modeling, sculpting, texture work, and simple animations. Attendees will leave with the knowledge needed to bring their creations to life!

Behave Yourself, A practical guide to ATDD in python

Doug Morgan

So you can sling some code and automate a web browser, and even work on a team that has a fair bit of UI automation around your "favorite" web application. No doubt you have found yourself spending to much time fixing an inconsistent suite of tests. You might even be asking yourself "are these tests even worth it?". (the answer is **yes!**)

Lets explore where things typically break down in the acceptance testing process. Starting with breaking user stories into acceptance criteria, and how the words the team chooses can affect the software that is built. Then taking a deep dive into best patterns and practices in automating the scenarios with Behave and other python modules. Along the way answering questions such as how should I manage my test data? Should I ever delete tests? And, how do I keep my test pyramid from becoming a test cupcake?

Attendees will walk away from this session armed with information to help them write better acceptance criteria, and tips to keep their test suites clean and less brittle.

Breaking Ground with iOS

Don Miller

This is a beginners class to start developing iPhone or iPod applications using iOS. Just bring your Mac laptop installed with Xcode 8 and you will create your first "Hello World!" application in the first hour of the day.

This session will give you the knowledge to start building iOS applications using Xcode and Swift 3.0. It will also give you the understanding of terminology to help follow along on other online training or presentations at the local CocoaHeads groups. Each topic includes some instruction time and some hands-on lab time. The material presented will also be available after the class has completed.

This introduction class was designed by Don Miller (GroundSpeed™) using his consulting background and teaching experience for his iOS development class at the University of Toledo.

Building Domain-Driven AsP.NET Core Applications (Part 1)

Steve Smith

Writing SOLID, testable ASP.NET Core applications has never been easier, but it may require a change in the way you typically structure your projects and their dependencies. In this hands-on workshop, you'll build a working ASP.NET Core application, complete with unit and integration tests. Along the way you'll learn the basics of Domain-Driven Development, and how to apply them to ASP.NET Core application development. The principles and patterns will apply to developers working with previous versions of ASP.NET MVC, as well.

Presented by one of the primary authors of the official ASP.NET Core documentation (docs.asp.net). This is a hands-on session; please bring your own laptop with tools installed.

CodeMash Capture the Flag Tournament

Michael Woolard, Bill Sempf

There is more to being a hacker than hooded sweatshirts and pasty skin. Hackers pride themselves on being problem solvers, thinking logically through not only what a web applications should do, but everything possible for it to do. We have all sat through the boring Top10 lectures and read the secure coding standards. Now let’s learn what we are protecting our self from. Capture the Flag (CTF) tournaments are a good way of honing your security skills, or discovering skills you never knew you had. Join us for the introduction into CodeMash’s inaugural CTF. We will get you setup with the tools pentesters use, a quick demo and training on how to use those tools, and Hacking-Lab.com will be on hand to discuss how the CTF works and the type of workload you can expect to be able to compete for the grand prize. This will be ran in an open house format, so attend for 4 minutes or 4 hours. Stop by before or after attending a different session, or during a break at a different class. We will get you setup, give you the rundown and you can stay and train/play for as long as you like! More information can be found at https://codemashctf.com/

How Cold is CodeMash...Really?

Brian Sherwin

Attendees will build an Internet connected “Temperature Monitoring Station” that will collect the temperature in the CodeMash sessions throughout the conference. These temperatures will be displayed in the form of a “CodeMash Heat Map” website available to all attendees. During this session, attendees will learn about collecting, securing, communicating and visualizing data from many devices in a close to “real world” scenario.

Leadership 101

Jim Holmes

Have you worked for people you thought were great leaders, but weren't sure where they got that magic from? Have you seen people getting great results from their teams during difficult times and wanted to be able to get similar success with your own teams? Do you want to take on more responsibility in your career, but aren't sure how to get those opportunities? Leadership 101 will help you answer some of those questions!

Join Jim Holmes in this highly interactive workshop where you'll learn some fundamental skills and gather some tools that can help you on your leadership journey. You'll clarify what makes an effective leader, learn a few critical communication skills, and get tips on dealing with difficult people and situations.

You'll leave this workshop with a better understanding of leveraging your strengths and mitigating your weaknesses. You'll also take away approaches for ensuring you're able to best empower your teams to do amazing things.

Nodebots Workshop (Session 3)

Rachel Weil, Brian Sherwin

The NodeBots are back by popular demand! Join Microsoft for a hands-on experience building a robot powered by JavaScript. We’ll use the Adafruit Feather HUZZAH microcontroller and the Johnny-Five JavaScript library to build a cloud-connected bot. Then, we’ll customize the bot with sensors, lights, and other electronic goodies for a special challenge to be announced. This is a workshop you won’t want to miss!

Practical Data Science with R

Matthew Renze

Data science is the practice of transforming data into actionable insight. R is the most popular open-source programming language currently in use by data scientists. In our data-driven economy, this combination of skills is in extremely high demand, commanding significant increases in salary, and is revolutionizing the world as we know it.

In this workshop, we’ll learn about the practice of data science, the R programming language, and how they can be used to answer day-to-day questions about your business. In addition, we’ll learn how to transform and clean our data, create and interpret descriptive statistics, data visualizations, and statistical models. We’ll also learn how to handle Big Data, make predictions using machine learning algorithms, and deploy R to production.

Prerequisites: Please bring your own Windows laptop and complete Lab 0 at the URL below to install all of the necessary software before the workshop begins.

All course materials can be found at the following URL:
http://www.matthewrenze.com/workshops/practical-data-science-with-r/

React Everywhere: Three Platforms with One Framework (Part 1)

Leonard Smith

React is more than just the hottest client side framework. React is also a viable option on the server and mobile, and you can share more code between all these platforms than you probably think!

In the morning we'll learn about React and its ecosystem and write a web application. We'll also learn how to reuse React components on both the server and client. This will be a great introduction if you're new to React.

In part two, we'll add mobile as an additional client using React Native. We'll see how we can structure our project to reuse as many bits as possible in a clean and maintainable way.

By the end of the session you'll have a working application that can be deployed to the web and to your phone from the same code base!

PreCompiler Session 04 - Wednesday 13:00

Building an Effective Interview Framework

Cassandra Faris, Jared Faris

Growing your team is about more than just hiring the people with the strongest technical skills. The interview process is about making sure the people that join your team fit into the larger picture. In this workshop, a software development manager and a technical recruiter will teach you how to identify the needs of your specific team. They will help you create a unique interviewing framework that considers your team's technical and non-technical needs. This framework will compare those needs with the strengths and weaknesses of potential team members. Then, you will apply your framework to common scenarios to help make more informed and effective interviewing decisions. Finally, we will discuss how to take this framework back into the real world.

Building Domain-Driven AsP.NET Core Applications (Part 2)

Steve Smith

Writing SOLID, testable ASP.NET Core applications has never been easier, but it may require a change in the way you typically structure your projects and their dependencies. In this hands-on workshop, you'll build a working ASP.NET Core application, complete with unit and integration tests. Along the way you'll learn the basics of Domain-Driven Development, and how to apply them to ASP.NET Core application development. The principles and patterns will apply to developers working with previous versions of ASP.NET MVC, as well.

Presented by one of the primary authors of the official ASP.NET Core documentation (docs.asp.net). This is a hands-on session; please bring your own laptop with tools installed.

Building Serverless Applications in AWS Workshop

Christopher Judd, Jarred Olson

Tired of trying to manage and maintain servers? Never have a large enough operations team? Don't have a budget for running lots of server? Don't want to pay for servers siting idle? Afraid you might become so popular that you won't be able to scale fast enough? Don't worry, it is possible to alleviate these issues by moving to a serverless architecture that utilizes microservices hosted in the cloud. This type of architecture can support all different types of clients including web, mobile and IoT.

During this hands-on workshop, you will build a serverless application utilizing AWS services such as Lambda, API Gateway, S3 and a datastore.

Functional Programming 101

Nathan Dotz

As an advocate of functional programming, I'm regularly asked "I've tried to learn functional programming a number of times, but how do I apply this to my everyday work?". One of the hardest parts about learning a new technical skill or concept can be finding ways to apply those skills in our daily practice, especially when constrained by time or money to achieve a specific goal. Sometimes, one of the best ways to learn is to break away from what we're used to and learn in a totally new environment, the better to bring those new experiences and ideas back to the environments we're used to once we return.

In this workshop, we'll start by exploring two functional programming languages that these days are relegated largely to academic study, highlighting the influences they've had on modern functional programming paradigms and reinforcing these ideas with exercises to be solved as a group along the way. First, we'll cover ML, a language whose ideology remains largely unchanged for over 40 years and considered the common ancestor of modern functional languages like Haskell, OCaml and F#. Next, we'll cover Racket, a LISP derived from Scheme which shares similarities and probably inspiration with a number of currently popular languages, perhaps most obviously, Clojure and perhaps less obviously, Javascript. Lastly, we're back to where you work: I'll present exercises that help to reinforce the ideas learned from ML and Racket to be solved either in the language of your choosing, or Javascript (the de facto language of examples and solutions du jour).

Join me for an enlightening foray into some languages largely forgotten or ignored by the industry that are certain to help you bring new ideas about functional programming to your daily practice, whatever it may be.

Git Hands-On Workshop

Keith Dahlby, Cori Drew

No matter how long you've been using Git, it becomes much easier to use once you approach it with the understanding that it's fundamentally different from a centralized VCS, and grok just enough of its internals to wipe away those scary "will it break? what just happened?" feelings.

Act I of this precompiler will focus on how Git works under the hood, from content to configuration. To develop a mental model of how Git works and why, we'll introduce its core concepts with "The Git Parable" by Tom Preston-Warner. Around that core model we will build a more complete understanding of key local and remote Git interactions.

Act II will introduce katas for advanced techniques that will earn you a spot in your friends' and coworkers' git.txt. Katas will include history inspection and manipulation, disaster recovery, and power tools like bisect and filter-branch.

Hands-on Techno-Fashion

Kimberly Clavin, Jonathan Stevens

A morphing of technology, convenience and aesthetics results in a wide range of possibilities in the world of wearables and techno-fashion. According to a recent ABI Research report, wearable health and fitness devices are expected to hit 169.5 million in five years, a huge jump from the nearly 21 million devices sold in 2013. This hands on workshop will immerse participants into the world of programmable e-textiles through a simple development board. Join in collaborating to invent and create your own techno-fashion project. Dream up a GPS dog collar or make a dazzling tiara. The technology is as easy as connecting a battery, copy and paste the code. Leave with resources on how to make your idea a reality.

Nodebots Workshop (Session 4)

Rachel Weil, Brian Sherwin

The NodeBots are back by popular demand! Join Microsoft for a hands-on experience building a robot powered by JavaScript. We'll use the Adafruit Feather HUZZAH microcontroller and the Johnny-Five JavaScript library to build a cloud-connected bot. Then, we'll customize the bot with sensors, lights, and other electronic goodies for a special challenge to be announced. This is a workshop you won't want to miss!

Practical Data Science with R (Part 2)

Matthew Renze

Data science is the practice of transforming data into actionable insight. R is the most popular open-source programming language currently in use by data scientists. In our data-driven economy, this combination of skills is in extremely high demand, commanding significant increases in salary, and is revolutionizing the world as we know it.

In this workshop, we'll learn about the practice of data science, the R programming language, and how they can be used to answer day-to-day questions about your business. In addition, we'll learn how to transform and clean our data, create and interpret descriptive statistics, data visualizations, and statistical models. We'll also learn how to handle Big Data, make predictions using machine learning algorithms, and deploy R to production.

Prerequisites: Please bring your own Windows laptop and complete Lab 0 at the URL below to install all of the necessary software before the workshop begins.

All course materials can be found at the following URL:
http://www.matthewrenze.com/workshops/practical-data-science-with-r/

React Everywhere: Three Platforms with One Framework (Part 2)

Leonard Smith

React is more than just the hottest client side framework. React is also a viable option on the server and mobile, and you can share more code between all these platforms than you probably think!

In the morning we'll learn about React and its ecosystem and write a web application. We'll also learn how to reuse React components on both the server and client. This will be a great introduction if you're new to React.

In part two, we’ll add mobile as an additional client using React Native. We’ll see how we can structure our project to reuse as many bits as possible in a clean and maintainable way.

By the end of the session you'll have a working application that can be deployed to the web and to your phone from the same code base!

Shiny, Let's Be Bad Guys: Exploiting and Mitigating the Top 10 Web App Vulnerabilities

Mike Pirnat

The Internet is a dangerous place, filled with evildoers out to attack your code for fun or profit, so it's not enough to just ship your awesome new web app--you have to take the security of your application, your users, and your data seriously. You'll get into the mindset of the bad guys as we discuss, exploit, and mitigate the most common web app security flaws in a controlled environment.

We'll discuss each kind of the most prevalent security flaws at the theoretical level; then using a specially-crafted, deliberately vulnerable app, individuals or pairs will carry out exploits against these flaws, and we'll discuss strategies for mitigating each type of attack in several popular Python frameworks.

We'll be using the OWASP Top 10 as our topic roadmap, addressing issues such as:

- Injection Attacks
- Broken Authentication & Session Management
- Cross-Site Scripting (XSS)
- Insecure Direct Object References
- Security Misconfiguration
- Sensitive Data Exposure
- Missing Function-Level Access Control
- Cross-Site Request Forgery (CSRF)
- Using Components with Known Vulnerabilities
- Unvalidated Redirects and Forwards

You'll want to set your brain to "devious" mode; you'll also need a laptop with Python 2.7 or 3.3 (or a buddy you can pair with). Having pip and virtualenv will be useful too, as will having Git installed to pull down the code we'll be working with.

Attendees **do not need prior security experience**; this tutorial is aimed at **intermediate web developers** who are interested in gaining hands-on experience with **simple forms of the most common attacks**. Attendees should have some experience with **Python, Javascript, and SQL**, and may benefit from at least a passing familiarity with Django (eg, previously attending a Django tutorial or working through the online tutorial).

Main Session 01 - Thursday 08:00

.NET on Linux: Zero to Hero In Minutes

Don Schenck

.NET Core 1.0 is the new .NET.
ASPNET Core 1.0 is the new ASP.NET.
.NET on Linux is the new hotness.

In only minutes, you'll learn how to install .NET on Linux and create your first program. You'll learn about: the different parts of .NET Core 1.0; package managers and NuGet; the new command line interface (CoreCLI); and a great editor that is not Visual Studio. By the end of this presentation, you will know how to create C# programs that runs on Linux. No smoke, no mirrors, no tricks; it just works.

We'll see how to create a basic MVC program, and then we'll run that program in a Linux container.

Finally, we'll debug the program, running in Linux, *from within Visual Studio running on Windows*.

You'll also learn why this is important to you and your organization.

Finally, you'll leave with all the tools needed to start right away, including a Developer's Copy of Red Hat Enterprise Linux (RHEL) that you can start using on your PC or Mac today.

5 Shades of Gray - Presenting with Clarity

Mark Miller

Learn how to how to present information effectively. This fast-paced session covers information relevance, emphasis, proximity, information in parallel, smallest effective difference, contrast, size, color, text, fonts, borders, and signal & noise. We'll show how existing user interfaces can be improved. If you don't have a graphic artist on your team, this session will show you how to fake it, and if you do, this session will show you how to communicate with them persuasively. This session shows how you can make a big difference with small changes.

Automation for the Win!

Randy Syring

A great development workflow will make your good developers great, and your great developers exceptional and automation is a cornerstone of a great development workflow. Every task that can be done by a machine should be. Continuous Integration (CI) is the practice of testing each change done to your codebase automatically and as early as possible.

Learn how to increase the quality of your code and improve team productivity by leveraging a CI pipeline to run tests, lint, and measure code coverage. Then, integrate all that knowledge right into a GitHub pull request for easy team review & verification before deployment.

Clouds & Containers: Hit the High Points and Give it to Me Straight, What's the Difference & Why Should I Care?

Mark Heckler

As developers, we hear a non-stop stream of technical-but-marketing messages for containers, orchestration tools, and cloud services. There is extensive overlap in these areas with regard to both means and ends, and it's time to clear the fog and get to the bottom of things. This talk will give a quick overview from a hardcore developer's perspective of the following topics:

- How can I use containers to develop better software?
- What are orchestration tools? Do I need to consider/use them?
- How do cloud/PaaS options compare? What are the tradeoffs?
- What is the difference?
- Why should I care? (Or should I?)

In this session, the presenter discusses several of these technologies, compares them, and *deploys real applications to them LIVE* to demonstrate subtle differences and tradeoffs each choice imposes upon developers, for better or worse. Come to this session to level up on containers, clouds, and developing real production software, regardless of where or how you deploy it.

Components and More: Effective Angular 2 Testing Strategies

Ken Dale

Angular 2 is released! With it comes new opportunities for brilliantly authored and well-tested applications! While tests don't come free, they're well worth the time and effort -- especially for larger or long-lived codebases.

We'll explore the full gamut of Angular 2 testing: What to test, how to test, classifications of tests, and the implications and tradeoffs for each of these decisions. Soon you'll be ready to `npm test` Angular 2 projects with confidence -- no Angular 1.x experience required!

Data-centric Encryption in Practice

Wolfgang Goerlich

They make it sound easy. Identify data assets. Assess threats. Protect it. Encrypt it. In theory, sure. In practice, not so much. Besides, what does a real attack look like? This session explores the problem, punch and counter-punch, by demonstrating attack techniques and encryption coding practices. We will start with common use cases, such as data warehousing, payment systems, Big Data analytics, and more. We will then discuss the threats and vulnerabilities, perform a basic threat modeling and risk assessment, and show how criminals punch through the security. Using secure development patterns and tools, we'll demonstrate how to block and counter-punch the criminals. Heavy on the demos and light on the theory, this talk provides practical guidance on effective encryption.

Fat Controller CQRS Diet

Derek Comartin

Do your controllers need to go on a diet?

Fat controllers can quickly lead to tight coupling by the abundance validation, business logic, and data access.

Thin out your controllers by only using your web framework for what it's good at: HTTP, Routing, and Serialization.

Attendees will learn to how to organize code by feature by leverage CQRS and the Mediator pattern to decouple your core business capabilities from your web framework.

How to Have the Best Dates Ever! (On date, time, and time zones in programming.)

Matt Johnson

"Do dates make you nervous?" "OMG Yes, especially when doing math across time zones!" The Internet is full of jokes about dates, times, and time zones, but why? It turns out that time is a complex domain that most programmers don't fully understand. In this session, a maintainer of the open source libraries Noda Time (.net) and Moment.js will walk through the nuances and gotchas.

Several conceptual fundamentals and best practices will be dished out, and many fallacies will be dispelled. Participants will walk away from this session understanding the date and time features of JavaScript, .NET, SQL Server, Python, and other platforms, and with actionable knowledge to apply to their projects. Prevent date bugs, and learn how to have the best dates ever!

How to Win Friends and Influence People With Data Visualization

Amanda Cinnamon

Data visualization and infographics are hot topics in today's information-saturated environment, yet the dialogue tends to focus on tools and methodology. Rarely does anyone take the time to understand what makes a graphic powerful. High quality visual elements are those that are easy to read and quickly convey a message. This session discusses the principles of good data visualization, including the concepts of cognitive load, preattentives, and data-ink ratio. See examples of how thoughtful selection of visual elements can transform a lack-luster plot into a meaningful image that will help you tell a compelling story and leave a lasting impression.

Implementing binary protocols with Elixir

Ole Michaelis

In this talk you'll learn how to implement a binary protocol using Elixir. We look at binary pattern matching, de- and encoding of primitives, binary data frames and extracting tests from RFCs. We will also take a look at implementing HPACK as part of HTTP/2, so you learn some of that along the way.

Looks Like Rain Again: Secure Development in the Cloud

Bill Sempf

There is much talk about security in the cloud these days, and it is understandable why. With IAAS out of direct supervision of our admins, they, and the executives, are understandably nervous. As developers, though, we know the forecast - the weather is pretty much the same when developing cloud based applications as it is when writing on-premise apps. During our time together, we'll go over those few practices that differ, and remind ourselves of those we should always be following.

Sowing the seeds of STEM through Techno-Fashion

Kimberly Clavin

Technology is both a rising force in our everyday lives, and one of the fastest growing job sectors. How can we get more young people engaged in the field while making sure that companies have a diverse workforce? To answer that question various partners in Central Ohio came together to offer an immersive experience in techno-fashion, for girls. The highly desired and attended event offered stations exploring techno-fashion topics such as data analytics, prototyping, patterns, electronic hardware and coding. Tony Wagner stated in his book Creating Innovators that a pattern of “play to passion to purpose” will encourage innovative thinking by nurturing curiosity and inquisitiveness of young people. In order to attract and grow more women in technology we must create supportive environments that sparks a passion. Participants of this talk will learn about how to empower more girls into technology as well as gain insight on how to run a similar event in their area.

Stand in the Gap: Have a Mentor, Be a Mentor

Joel Mason

We talk about mentoring a lot in the software industry, but how much do any of us really know about how to mentor? If we're honest with ourselves, we are probably making it up as we go. This talk will hopefully move beyond that and move towards having a purpose to our mentoring relationships. We will take a look at how to start a mentoring relationship, setting a purpose for the mentorship and ultimately how to make the most out of it. Whether you are a mentor, a mentee or even someone who is just interested in possibly doing this someday then you will benefit from this talk.

Truths Universally Acknowledged: Swift Design Patterns as Jane Austen Heroes

Anne Cahalan

Have you ever looked at your code and realized that a certain design pattern was charging in like a romantic hero, sweeping away the confusion and bringing order to chaos? Or perhaps you've encountered a pattern that you utterly hated...until it's virtues slowly grew on you as you realized that the alternative was a complete disaster? Let's imagine Ada Lovelace reading Jane Austen, and compare some of my favorite design patterns with some of my favorite Jane Austen heroes.

What are Observables and why should I care?

Randall Koutnik

Learn about the next generation of asynchronous abstraction: Observables. Born of a cross between the Observer & Iterator patterns, observables are being used from Angular 2 to the data teams at Netflix & Microsoft. Start at a high level, learning why observables stand out among the many asynchronous abstractions available to the developer. Then, learn how to conceptualize event streams as simple observable flows when tackling typeaheads on the frontend. Finally, enter the world of machine learning and see how observables can be used for stream processing. Walk out with the confidence to harness the power of observables to straighten out your asynchronous apocalypse.

Main Session 02 - Thursday 09:15

Automated Testing: Beyond The Basics

Jim Holmes

You've been through some initial training or bootstrapping. Now you're three to six months down the road and you're finding a lot of time is being spent chasing intermittent test failures. You're spending a lot of time maintaining your codebase. You're spending a lot of time dealing with frustration and trust in the automated tests. What's you're not doing is adding a lot of value around new feature work.

Sound familiar? It's a pattern that's common across nearly every team that is new to building significant automation suites. Jim Holmes helps you find a few approaches that may save your sanity—all based on years of experience and hard knocks.

This isn't a WebDriver 101 session; we won't be covering basics like waits, find logic, or basic page objects. Instead, we'll dive in to solving problems using software craftsmanship principles, custom-designed APIs, and approaches like Selenium Grid to help you scale out your test suites via parallel execution. You'll leave the session with practical tips to apply to your own testing.

3M Requests per second - The New ASP.NET Core on Linux

Jeffrey Fritz

In this talk, we will explore the speedy new ASP.NET Core framework for building applications on Linux. We'll look at how Microsoft has a complete open source stack for developing and delivering web applications on Linux and other operating systems. Attendees in this session will learn about how ASP.NET Core is built for performance and how easy it is to get installed and working with Docker or Apache

A math-free introduction to Neural Networks

Randall Koutnik

You've probably heard of artificial neural networks - mysterious brains that power titans like Google & Netflix. They sound fascinating - machines teaching themselves? You may even have tried to build your own but were stymied by scads of equations and other strange symbols. Fear no more, fellow human! This talk will give you the tools and vocabulary you need to apply machine learning to your day job and side projects.

Beyond the language: the importance of algorithms in programming

Simone Carletti

Mastering a programming language is not enough to write efficient code. Programmers often try to squeeze a programming language as much as they can in order to achieve the best performance, without realizing that the solution could be found in a more efficient algorithm or data structure.

The talk is meant to inspire you to learn more about algorithm design by demonstrating, with real world examples, how a properly designed algorithm may drastically increase the efficiency of your code, regardless of the programming language you are using.

Chipping away at the monolith with Go.

Aaron Salvo

Google's Go programming language is taking the cloud by storm. Go provides built-in concurrency, has a rich HTTP server package, and compiles down to a single statically-linked binary; making it perfect for creating horizontally scalable microservices. Attendees will see an end-to-end case study of using Go and the Go Kit library to breakdown a large legacy system into a modern software as a service.

Docker and Kubernetes Recipes

Arun Gupta

So you are a developer but how comfortable are you and your team taking Docker from development to production? Are you hearing developers say, “But it works on my machine!” when code breaks in production? And if you are, how many hours are then spent standing up an accurate test environment to research and fix the bug that caused the problem?

Docker provides PODA (Package Once Deploy Anywhere) and helps you reduce the impedance mismatch between dev, test, and production environment and simplifies application deployment.

This workshop/session explains how to package, deploy, and scale applications using Docker. It will also cover orchestration frameworks like Docker Engine in Swarm Mode and Kubernetes.

Outline:

Docker and Kubernetes 101 Running first application with Docker and Kubernetes Package your application with Docker and Kubernetes Sharing your application using Docker Hub Package your multi-container application with Docker and Kubernetes Deploy your application using Maven Deploy your application using Docker for AWS and Kubernetes on AWS Develop/deploy your application using NetBeans, Eclipse and IntelliJ

Hypermedia APIs: The rest of REST

Chris Marinos

Let's face it, the term "REST" has become too popular for its own good. If you think that REST translates to "send JSON using HTTP", then it's time for a hypermedia intervention! In this talk, you'll learn about the origins and core tenants of RESTful design, where most APIs go astray, and how hypermedia APIs bridge the gap. You'll also learn how to take a normal JSON+HTTP API and enrich it with hypermedia to reach REST enlightenment. You'll walk away with a better understanding of REST architecture and plenty of new concepts to help you build awesome, easy to use web APIs.

Identity and Access Management - What is IT IAM doing?

Penny Vanlerberg

The ability to know WHO has access to WHAT in any environment is a vital tool in this day of cyber espionage and hacking warfare. Especially considering all those contractors and non-employees granted God like access to your computers and systems. Who is watching them to make sure access is revoked upon completion of their contractual engagement? What ARE they doing with the keys to your kingdom?

This session will provide the basic building blocks required to stand up an Identity and Access Management (IAM) solution by defining IAM terminology, describing how IAM fits into the security realm, presenting best practices defined by the IAM community, and sharing compelling war stories from active duty on the battlefield. (I don't understand?! Why doesn't a Taco Bell receipt count as a signed security agreement?)

Come see what IT IAM is doing and protect those keys to your kingdom!

Integrating React into a legacy web app

Matthew LaForest

You know the web app, the one started 6 years ago, before all of these cool new JS technologies came along. Creaking with a 3 year old copy of jQuery loaded down with 30 some plugins of code that is almost impenetrable.

You would like to modernize it but you don't even know where to start. You have a working site, and know the pitfalls of the dreaded "rewrite". How do you integrate the new tooling along side what is already there and allow yourself to build something knew. Learn how you can integrate React seamlessly into a legacy web application. Have it slowly take on more responsibility as you let it become your new codebase.

Leadership Journey: From Software Developer to Leader

Michael Eaton

You've spent years working on your skills as a software developer. You measure your days by the number of commits you've made and look forward to the daily pairing sessions with your team mates. Then, one day, a leadership position opens up on your team and you start thinking about it. Should I or shouldn't I? What if I go for it and end up hating it or I'm just not that good at it? What if I love it? How will my day change?

In this interactive session, we'll explore "why leadership", answer some hard questions and talk about one developer's path from writing code to leading people and the challenges that have been experienced.

Multi-cloud Continuous Delivery with Spinnaker

Andrew Glover

Spinnaker is an open source, multi-cloud Continuous Delivery platform jointly developed by a number of companies including Netflix, Google, and Microsoft. The platform was built with flexibility in mind and consequently supports strong integrations with AWS, GCP, Kubernetes, Azure, CloudFoundry, and OpenStack. Companies around the globe, large and small, are leveraging Spinnaker and its core concept of immutable infrastructure as means for rapidly and reliably deploying software assets. In this joint talk, Netflix, Google, and Microsoft will present Spinnaker's core feature of orchestrated deployments to public clouds, the best practices associated with multi-cloud delivery, and the myriad lessons learned for scalable and reliable global deployments to any cloud.

Practical MVP and MVVM Patterns for your iOS Apps

Priya Rajagopal

Is the Model View Controller (MVC) design pattern not working too well for your iOS apps? Despite your best efforts, does the dreaded "Massive View Controller" creep up on you making your code unmaintainable and unreusable? Then this talk is for you! In this session, I'll discuss how you can better architect your iOS apps by adopting two popular alternatives to the MVC pattern - the Model-View-View Model (MVVM) and the Model-View-Presenter (MVP) . I'll provide an overview of these patterns and we will spend the bulk of the session walking through code that demonstrates these patterns in practice. Whether you are just starting off building iOS apps or you have been doing it for some time, this talk can benefit you.

Pragmatic Functional Programming With Erlang

John Daily

For many developers, the thought of learning functional programming is intimidating. Parentheses, lambda calculus, type theory, and you can't even modify a variable!

In this talk we'll look at code that does useful work without requiring a Ph.D. in greybeard. Talk to software on another computer without opening a socket. Iterate over your data without setting up an iterator class hierarchy. Disassemble a network protocol header with this one cool line of code!

We'll talk about ideas that are being adopted by languages you probably already use. No functional programming experience required, just a willingness to see what can happen when you think outside the OO box.

Problem solving recipes learned from supporting Spark

Justin Pihony

Due to Spark, writing big data applications has never been easier…at least until they stop being easy! At Lightbend we've helped our customers out of a number of hidden Spark pitfalls. Some crop up often; the ever-persistent OutOfMemoryError, the confusing NoSuchMethodError, shuffle and partition management, etc. Others occur less frequently; an obscure configuration affecting SQL broadcasts, struggles with speculating, a failing stream recovery due to RDD joins, S3 file reading leading to hangs, etc. All are intriguing! In this session we will provide insights into their origins and show how you can avoid making the same mistakes. Whether you are a seasoned Spark developer or a novice, you should learn some new tips and tricks that could save you hours or even days of debugging.

Taming the JavaScript Dragon with TypeScript

Dustin Ewers

From humble beginnings, JavaScript has gone from a hastily thrown together language for web pages to a hastily thrown together language that runs everywhere. If you can do it with code, you can probably do it in JavaScript. It's the Swiss Army Knife of programming languages.

Unfortunately, JavaScript wasn't designed for the large scale programs we use it in today. We've had to rely on design patterns, willpower, and luck to mitigate JavaScript's failings. However, there is a better way. TypeScript adds features to JavaScript that make it usable on large scale projects. In this talk, we'll explore TypeScript and how it can be used to tame large scale JavaScript projects. We'll cover some of the important features of TypeScript and how to get started using it in your own projects. After this talk, you'll be ready to slay the JavaScript Dragon.

The Whole World is Burning: Here is your bucket of water

Charles Yost

Information Security is a galvanizing term. It carries much power, and therefore, much responsibility. It can be hard to obtain buy-in for long term security needs from Management. And with all the vulerabilities cropping up every day, the pressure to "be secure" can be overwhelming. How can a lone developer make a difference?

This talk is all about the second half of its title. Common security concerns will be reviewed, then addressed with a focus on what can be done when faced with them or how to avoid them in the first place.

The whole world is burning. But you can make a difference. Here is your bucket of water.

Main Session 03 - Thursday 10:30

A Technical Tour of Real-World Web Application Vulnerabilities

Justin Collins

You've seen the headlines: *"TWITTER HACKED!"* and then it turns out someone had a weak password like "dadada". Or else the details are so hazy it's impossible to tell what actually happened. As a result, security issues become vague problems that happen to *other* companies.

But there is good news! Thanks to the growing popularity of bug bounties and public disclosure, it is easy to take a peek into real security vulnerabilities at well-known companies and learn from them. This tour includes stops at Facebook, Twitter, United Airlines, Domino's, Instagram, and more! This tour covers all relevant technical details, no fluff. Please fasten your seatbelt and keep hands, feet, and head inside the bus at all times.

An Introduction to Xamarin.Forms

Jason Farrell

With Google now claiming an 80%+ worldwide marketshare percentage and Apple maintaining a 60%+ share of mobile profits it is no longer an option to support one or the other; good mobile apps support both platforms and leverage those platforms unique idioms appropriately. But even with the code sharing abilities offered by Xamarin we still end up doing a lot of work with UIs. For this, Xamarin has created Xamarin.Forms, a unique tool that allows developers to create a single UI definition that gets interpreted for each supported platform. This enables developers to quickly write applications in as many as 4 platforms simultaneously. In this talk, I will showcase how this works including a full discussion of custom renders and styles, and where Forms makes sense and where it does not. I will also cover FreshMVVM which is fast becoming a favorite among developers looking to leverage MVVM in their Forms applications.

Asset Template Guides: Creating Responsive Images

Kevin Mack

Inspired by traditional print design, Asset Template Guides are a solution to predict reflow of content in a responsive environment, while giving visual designers more creative freedom when designing for responsive sites, and improving communication between developer and designers. Through discussion and a demo, this presentation provides an introduction to Asset Template Guides and the process for teams to utilize them.

Bash on Ubuntu on Windows

Paul DeCarlo

Windows 10 now provides developers with a familiar Bash environment. This environment will allow users to:

1. Run native Linux binaries including grep, sed, and awk
2. Navigate a new Linux based file system using these commands
3. Run bash shell scripts which rely on supported command line utilities

Windows accomplishes this through the Windows Subsystem for Linux which allows Ubuntu user-mode binaries provided by Canonical to run on Windows 10. This means that the command line utilities are the same as those that run within a native Ubuntu environment.

In this session we will showcase scripting, code editing / compilation, and execution of X11 apps compiled for Linux using a local X11 server from within the Bash on Ubuntu on Windows environment. We will then discuss the implications of these features as they relate to existing developer workflows.

This will include a demonstration showcasing compilation of various programs using node.js, python, c++, asp.net, ruby and even a port of the original first person shooter Quake. We will also include a demonstration showing how to build and deploy a Ruby based web application from within Visual Studio Code using Bash on Ubuntu on Windows as an integrated terminal.

Finally, we will show how to obtain the latest bits for Bash on Ubuntu on Windows that are shipped in the upcoming Windows 10 Anniversary update.

Birth of a Language: Elixir

Anthony Eden

How often do you have the opportunity to see the birth of a new programming language? What mysteries does this process hold within? Come along on a journey, through the magic of git, into the earliest days in the development of the Elixir programming language. Watch history unfold as José TDDs this new language and learn a bit about lexers, grammars, and parser development along the way.

Declarative testing for JavaScript applications

Aaron Salvo

Testing is an important part of software development, but testing is not without its share of challenges. Smaller teams find it difficult to divert resources away from writing application code to write test code. It is often impractical to dedicate a single engineer to testing; even on a large team. Test writing then becomes the responsibility of all team members, and this leads to a maintenance problem with test code varying wildly from one component to the next.

This talk will discuss solving these problems by creating a standard test framework that is driven by simply declaring the structure of the component being developed. Attendees will see a real-world example using Jasmine, Karma, and simple JSON to test an AngularJS application. They will also see how to automate test creation using Yeoman generators. Attendees will be able to reduce the time spent writing test code, improve their team’s testing discipline, and make their tests easier to maintain going forward.

Electron: Desktop Development for Web Developers

Chris Woodruff

Imagine your Manager asks you to develop a new desktop application. You're so used to web development that you don't know how to get started. What should you do? Dig out your old VB 6 books or learn a new technology? Better yet, keep those web skills and use Electron.

Electron is the cross-platform application shell GitHub originally built for the Atom editor. You’ll leave this session with the starting skills to set up your development environment, generate your starter project, and build some business logic. Attendees will also start building a killer UX with the Photon control kit. Chris Woodruff will be your guide to leveraging your valuable skills for the desktop.

Failure is not an Option?

Benjamin Bykowski

Failure is an axiom of life but for as much good as it creates, it gets a bad rap. The real travesty isn't our failure, but that too often we don’t make use of it, learn from it, or lean into it. Some of the biggest disasters in history have come from the misapplication of failure, while some of the best ideas and biggest innovations have failure at the heart of their inception. In this talk we'll look at failure at both its best and worst, and answer some important questions along the way. Is it possible to fail without resorting to blame and shame? How does cognitive bias prime us to fail? Are there both good and bad ways to fail? (*spoiler alert* YES!) Is failure a better teacher than success? With historical examples and real-life stories of failure in design and development, we'll look at the anatomy of failure and discuss proven techniques for failing fast and approaches to handling failure to take back to your team and organization. At least one attendee will go home with a copy of Fail Better: Design Smart Mistakes and Succeed Sooner.

Full Stack ASP.NET MVC Performance Tuning

Dustin Ewers

> "Performance is a feature" - Jeff Atwood

Application performance is often under-appreciated. At least until the user complaints start rolling in. Spending time tweaking your application can be tough to justify to management, but performance is the ultimate feature. Even small increases in performance can drastically improve adoption of your application. Performance impacts the bottom line like a meteorite impacts the Earth.

In this talk, we're going to explore how to improve the performance of ASP.NET applications from top to bottom. We'll start by looking at ways to find the bottlenecks. Then we'll look at each layer in the stack and fix common performance issues. You'll leave with a plethora of new tools to tune up your own applications. After this talk, you'll be able to make your apps faster than Usain Bolt with cheetahs for shoes.

Life After Nil

Sam Jones

Ruby is late to the type system party. Let’s give Ruby something smart to say when it gets there. You’ll take a journey with Haskell’s type system. Along the way, you’ll learn how types can let you forget about nil, declaratively model your domain, and allow your compiler to drive your design.

How can ANYONE be productive in a language without if/else, while, or even classes? Let me show you! Haskell is proof that sometimes constraints can be liberating. There are murmurs that a type system will be coming to Ruby. Before that happens, you should get informed about what is hot in the current type system market. Haskell is known for it’s type system, but instead of describing it with dense language, let’s take a journey through code examples. Throughout our trip, we will avoid scary buzzwords like “monad” and “algebraic data type” because, honestly, what good is a formal definition when you don’t understand the power behind the concept.

We will compare solutions in Ruby to solutions in Haskell, and each stop on our trip will introduce a new mind blowing paradigm brought to you by Haskell’s type system. You will learn about type systems through a non-threatening story, and you’ll understand the value a type system can bring to your code. Key stops on our route will be “forget about nil”, “declaratively model your domain”, and “allow your compiler to drive your design”.

Patterns of Effective Test Setup

Seth Petry-Johnson

Writing clean, effective, and manageable tests begins with the "fixture", the set of data used in the test. If you've ever struggled with the "arrange" part of a test, or if you've ever looked at someone else's "arrange" and struggled to understand the context it establishes, then you've suffered the pains of poor fixture setup.

In this session you'll see a collection of patterns and techniques that will help you write smaller, more expressive tests that are easier to read, understand and maintain. We'll talk about patterns for constructing test data for unit tests, patterns for saving that data in the database for integration tests, and some common anti-patterns that you may not realize you're following. Code samples will be in C# and NUnit but the core concepts will be presented in a language-agnostic way and are applicable in many contexts.

"Clean setup begets clean tests". Let me show you how.

Preparing for This Augmented Life
Heather Wilde

Hololens. Rift. Gear. Pokemon Go.

Like 3D televisions, these things are starting to enter our everyday life. While still years away, people are being primed for the idea of what an augmented life could look like by popular media.

As companies start creating the early hardware that will support both Virtual and Augmented Reality in the mainstream, we must ask ourselves:

What are we attempting to accomplish?

What goals are we trying to set?

And above all -

What will the world we are creating look like?

TechHappy: Hacking Positive Communities
Lisa Anderson

Community: "A group of people living in the same place or having a particular characteristic in common."

Are you a leader or member of a community that could benefit from increased collaboration and more positive morale?

Whether you are a manager, community leader, or parent, you have the ability to "hack" your way to leading a more engaged, happy, balanced, and results-oriented group.

Learn how bringing positive energy, reinforncement, and expressing gratitude can cultivate a culture of harmony, balance, and help your team achieve desired outcomes faster and more efficiently.

That Time I Built a Thinking Robot
Seth Juarez

I decided to build a thinking robot in an effort to become allied with our future robot overloads early in the history of their inevitable takeover. This session will detail the process I followed in working with hardware and software to get the robot to solve specific tasks using machine learning. Attendees will learn the basics of IoT and machine learning so they too can form their own early alliance.

Why Clock Speeds Won't Be Getting Much Faster, or: EEs Are Amazing

Dan Wiebe

Concurrent software is a whole lot harder to get right than single-threaded software, even with high-powered concurrency primitives, isn't it? But we're forced into it whether we like it or not because our electrical-engineering brethren on the hardware side of the industry have decided, seemingly in unison, to stop increasing our clock speeds and start increasing our core counts instead.

What gives? Why would they do that to us? What did we ever do to them?

Come find out what gives. Does concurrent software challenge you? You may gain new appreciation and wonder for the challenges faced by processor designers and the amazing degree to which they have met, surmounted, or circumvented those challenges.

Main Session 04 - Thursday 11:45

5 Tips for Cultivating Emotional Intelligence in the Workplace

Christina Aldan

Learning to manage our state of mind in the workplace is an acquired skill. While stress in the workplace in unavoidable, it is possible to cultivate Emotional Intelligence (EQ) to manage our state of mind. Practicing EQ helps us identify and eliminate stressors in our lives. Awareness of self and awareness of others strengthens personal and professional relationships. When we understand the motivations of ourselves and the perspectives of others we form deeper connections. In this presentation, learn five tips for cultivating Emotional Intelligence in the workplace.

A Dance With Intelligent Dragon Drones

Jennifer Marsman

Drones are increasingly used in various commercial and consumer scenarios – from agriculture drones (providing farmers with crop and irrigation patterns) to consumer drones (that follow you around as you engage in action sports), to drone racing. Drones are outfitted with a large number of sensors (cameras, accelerometers, gyros, etc.), and can continuously stream these signals in real time for analysis.

This talk introduces the landscape of the various drone technologies that are currently available, and shows you how to acquire and analyze the real-time signals from the drones to design intelligent applications. We will demonstrate how to leverage machine learning models that perform real-time facial detection along with predictions of age, gender, emotion, and object recognition using the signals acquired from the drones. You will walk away understanding the basics of how to develop applications that utilize and visualize these real-time insights.

This talk is targeted at data scientists, students, researchers, and IT professionals who have an interest in building intelligent applications using drones and machine learning. It will be a fun and exciting exploration as we demonstrate a drone with the power of recognizing faces, ages, genders, emotions, and objects. You will learn how to leverage these same machine learning models to imbue intelligence into drones or other applications.

A Synchronicity – Asynchronous Programming in Android

John Pendexter

Mobile application responsiveness can make or break user experience. It is dramatically apparent when an application hangs for fractions of a second. On mobile devices that traditionally have less memory than their desktop counter parts, it is important to write code in such a way that resource intensive operations do not impact the experience. This often means asynchronous programming. The Android platform has numerous ways of writing asynchronous code. This session will explore many of the paradigms for handling asynchronous code on Android, including async tasks, adapters, services, loaders, queues, and events. Pros and cons and appropriate situations for these approaches will be discussed. Developers attending this session will leave with the tools to load data asynchronously and provide the smoothest UI experience for users of their Android applications.

Accessible Design - You have the Power - Use It!

Dylan Barrell

"Accessibility is usability" is the mantra of many in the accessibility profession. The truth is that up to 80% of accessibility isues can be avoided by integrating accessibility into the design process.

Designers have the power to open up the Web to people of all abilities - but to do so, they must grab the accessibility flag, hold it high, hold it proudly and be the change they want to see.

Most organizations tack accessibility onto the back end of their development process as a testing concern or as somethings their developers just need to take care of. This means that developer have to make up for the gaps that designers leave. They have to decide what the different elements in a design comp are. They have to determine the interaction model for non-touch/-mouse devices. They have to guess at the best order for reading content on the page, etc., etc.

In this session Dylan will cover an overview of all the design concerns that impact accessibility, how to think about accessibility during the design process and how to communicate this to developers and QA so that they can be successful in designing with accessibility in mind.

Auto Layout on iOS - How Many Screen Sizes Are There Now?

Brandon Campbell

Remember the good old days of iPhoneOS 2 where all you had to worry about was laying out your user interface for one screen size? They're long gone now. iPhones alone come in 4 different sizes and don't even get me started on iPad multitasking.

Well Auto Layout is here to help. In this session we'll explore the brief, sometimes embarrassing, history of iOS layout and explore using Auto Layout to create dynamic interfaces that respond to any screen size in Interface Builder or in code. We'll also look at some advanced layout techniques that will allow you to spend less time worrying about how your interface will respond to size changes and more time on what really makes your app work.

Containers for Windows Developers

Michael Collier

Over the last few years, Docker has popularized Linux containers, but Windows developers have been left out in the cold This changes with Windows Server 2016. At this session we'll cover the basics of the new Windows Server container model and how to develop applications/services that can take advantage of this exciting new advancement, for cloud, for on-prem, and for the future.

Gulping All the Things.

Chris DeMars

Lost in the forest of front-end build systems out there? Check out Gulp! Gulp makes it super easy to minify, concatenate, and optimize your assets in an easy piping build stream. There are only a few commands to learn, so writing complex tasks (like you would in another system) isn't a thing in Gulp. In this talk, I will walk through what Gulp is and create a working gulpfile that can be used across endless front end projects that minifies, concatenates, and optimizes assets, as well as live reloads the browser using browser-sync.

JavaScript Robotics? Not a Bad Idea

Brian Genisio

JavaScript is seeing its renaissance right now. All the cool kids are coding JavaScript everywhere they can. But robotics? Really? Doesn’t that sound like a bad idea? I’d like to show you the opposite: why JavaScript robotics is NOT a bad idea! It turns out that the same things that make JavaScript great for the web also make JavaScript great for robotics. Enter Johnny-Five: a JavaScript robotics platform with beautiful abstractions. We’ll discover the power of Johnny-Five and explore the possibilities of controlling low-cost electronics to manipulate the real world with a much more accessible language than C/C++.

Let's Write a Lambda Calculus in Haskell

Michael Gilliland

Predating computer programming itself, the lambda calculus has been called "the simplest programming language" in existence. Are you curious to learn more? Or do you simply want to see what this Haskell thing is all about? Then this is the talk for you!

Michael Gilliland will live-code an interpreter for a simple lambda calculus. Not only that but it will be built in language based on the lambda calculus: Haskell. No libraries will be used, so this session will serve as a great Haskell pool for you to dip your toes in.

You'll leave this session with a better understanding (1) of how interpreters work, (2) what the lambda calculus is and (3) what basic Haskell code looks like.

Security Automation in your Continuous Integration Pipeline

Jimmy Byrd

Developers use unit tests and acceptances tests in continuous integration (CI) to find bugs early and often in a repeatable way. Security is an important part of any software development life cycle. So why not add security analysis tools to this pipeline? This talk will cover adding and using OWASP/pipeline, a framework made for running security analysis tools in CI.

Simplify Your Life with CQRS

Joel Mason

Command Query Responsibility Separation doesn’t only simplify your code—it simplifies your thought process in all kinds of areas. In this session we will look through the benefits of CQRS for data persistence, performance, supportability and testing. We will also look at the way that these concepts allow a team to organize around a design and the clarity that CQRS can help provide.

The Evolution of Memory and Resource Management

Jeff Walker

The management of memory and resources like files and network connections has slowly evolved over time. First there was static memory layout, then manual memory management and eventually smart pointers. Today most developers work in a garbage collected language. This largely mitigates issues of memory management, but provides no help with other resources. In 2015, the Rust language baked in a new approach that unifies memory and resource management. Could this approach be the future of memory management? Come learn about each approach and their trade-offs through code examples. Seeing Rust's borrow checker will change how you think about resource management in other languages. Then, take a peek at some languages that may bring this new approach to a broader audience.

Unconventional Automation – Using test artifacts to reveal new testing capabilities

Matt Perrin

Test automation artifacts have the potential to create unique testing opportunities against their systems under test. In this session, a conceptual analysis of a large-scale test suite's results beyond the traditional pass/fail metric will unveil new automation capabilities. By programmatically examining test result data, attendees will learn about efficiencies discovered including exploratory UI testing with WebDriver, test suite consolidation, and test error classification within existing automation solutions.

Details: The test results generated by repeated execution of test automation scripts are a valuable and often ignored resource of information about a system under test. During this session, I will share approaches discovered while leading a large scale, multi-state test automation team looking to increase their test coverage. The ideas presented utilize both test results and client/server error tracking to show how new, adaptable automation can be generated that allows for more overall testing to be accomplished. * The session will show concepts where by utilizing web element information logged within a test result can create programmatically generated variations of a test that act as exploratory UI testing within WebDriver. * The session will show concepts where test results are used to collate and organize nearly identical tests together into testing super-sets; an approach we have labeled as Signature Based Automation. * The session will show concepts where test results can be programmatically inspected for related data features and failure points in order to provide test maintenance recommendations.

We Gonna Do What They Say Can't Be Done, Part I: Unit-Testing iOS

Dan Wiebe

Some folks say that iOS apps aren't really susceptible to automated testing at all, and that that's okay since they tend to have pretty short lifetimes anyway. Others say that part of an app's operation can be tested with functional UI tests, but beyond that it's really a lost cause. Still others can unit-test the middle layers, but code that involves the UI or the hardware is really untestable.

All those folks are wrong.

There are indeed limits to what automated tests can reach in an iOS app, but those limits are nowhere near where you've been told they are. Come see a non-trivial iOS app that shows 100% unit-test coverage, including the UI and the accelerometer, and learn a few tricks of the trade that will enable you to take similar advantage in your own code.

Who Are You & What Can You Do? Understanding Authentication and Authorization with Federated Identity Services

Kevin Cody

Authentication and authorization are two critical components to any highly secure and easily usable application. But it’s easy to get lost in acronym soup. Worse, between misconfigurations and lack of appropriate threat modeling, federated identity services can add substantial risk to a previously secure system.

Get details on how to effectively comprehend and avoid the security pitfalls in utilizing SAML, OAuth, OpenID, FIDO, Assertions, and more. No matter what you’re using – Java or .Net, Python or Ruby, JavaScript or the programming flavor de jour – this topic has direct bearing on anyone building or utilizing modern applications.

Main Session 05 - Thursday 13:00

.NET on Linux: Entity Framework Core 1.0

Don Schenck

The more things change, the more they stay the same.

Or do they?

Entity Framework (EF) Core 1.0 is a new start for EF. Small changes can trip up even the most skilled developer, and you don't want to waste time tripping over your own feet when you could be coding.

Well take heart, because this session will quickly bring you up to speed on EF Core 1.0. As an added bonus, we'll do it all in Linux, just to show how cross-platform .NET really has become. Well use a SQLite database too, because open source.

You'll leave with knowledge, a list of resources to guide you, and a zero-cost Developer's copy of Red Hat Enterprise Linux.

An End-to-End Overview of IoT

Vince Fabro

Interested in learning more about the Internet of Things (IoT) but not sure how it fits into your business? Want to avoid some common pitfalls on your company’s first IoT project? Don’t we all, and IoT projects are challenging! In this session we’ll discuss some practical, real-world production IoT solutions across multiple industries, and we’ll focus on many lessons learned and best practices along the way. This is not a narrowly focused deep dive into a single aspect of IoT such as device hardware or analytics; rather, we’ll take a balanced, end-to-end approach, discussing all the key aspects of IoT solutions: from planning an IoT concept, to the devices, to connectivity and processing, to advanced analytics and presentation/action. Finally, we’ll take a quick look at the architecture of some IoT solutions, as well as some options to help you jump start your own IoT solution. If you're looking to leverage IoT within your organization, this presentation will help you understand the many benefits and unique challenges associated with Internet of Things solutions.

An Introduction to building websites with Aurelia and Asp.Net Core

Eric Potter

In this session, we will look at the tools and techniques needed to get started building modern web applications with Aurelia for the front end and Asp.Net core for the backend.

We will create a new Aurelia front end with the Yeoman generator. We will create a WebAPI project in Visual Studio. And then we will look at making them play nicely together.

Applied Experience Design

Scott Sullivan

When you get down to it, UX isn't insanely complicated, you just have to know how to figure things out. It's a learning process that begins with a hypothesis and evolves with experimentation to verify your assumptions in the real world.

In this talk we'll cover: * What's UX and what's not UX * "Design Thinking" * The end-to-end UX design process * Research * Tools * Getting a job doing UX * How to suck at design

Building "serverless" software with AWS Lambda

Jonathan Knapp

I was asked to build a fuzzytext search interface for information stored in the SEC's massive Edgar database which holds all of the electronic documents filed with the SEC. By leveraging managed ElasticSearch, S3 for document storage, and the asynchronous job processing power of AWS Lambda I was able to build a solution that required absolutely no ongoing server maintenance for my client.

In this talk I'll explain how I was able to:

- parse gigabytes of info without IP activity restrictions
- provide an easy way to scale or disable the application
- continuously monitor parsing activity and application health

You will learn about the different services I utilized with their strengths and weaknesses as well as alternative services like Iron.io which allows you to write code in many different languages. I'll also talk about different ways async processing can be applied to other situations such as managing contact forms for static websites as FormKeep does.

Digital Payments in a Mobile World

Parag Joshi

Apple Pay and Android Pay are now here as are several other methods to pay electronically rather than using conventional means such as check or cash. How do we integrate these in our app and what are our options?

In this session, we will look at at an Android app that uses Android Pay to process payments using Stripe. We will take a look at the Stripe backend and also the android pay development do's and do not's.

We will also look at an iOS app that uses Square to accept credit card payments. No doubt, if you have walked into a store or rented a car, you have seen the Square terminals. We will take a look at the Square reader and walk through the code to interact with a Square app.

You will leave this session with an understanding of what it takes to integrate Stripe and Square into your apps.

ES6 Patterns in the Wild

Joe Morgan

Syntax changes are fun, but how can you best use them? Don't read contrived foo-bar examples. Open source code is a rich source of emerging ES6 patterns and best practices. How is React using the Map operator? How does Khan Academy leverage generators? How does Redux use. . .pretty much everything else? Spread operators. Object destructuring. Template Literals. Arrow Functions. It's all out there. The best part of learning through open source is that when new syntax is announced you'll have a backlog of projects that will likely try it out increasing your personal catalog of patterns.

Have Your Best Season Yet: Becoming a (Microsoft) MVP

Lisa Anderson

It's time to have your best "season" yet! If you are a technical expert with an active role as a community thought leader (speaker, blogger, forum contributor, author, GitHub contributor, mentor, etc), join Lisa Anderson, Microsoft Community Program Manager, to learn how you can be recognized and awarded for your outstanding community contributions. Learn how following your passion and deepening your technical area of focus can help you create a wide-sweeping impact and how you can be awarded as an MVP! The MVP community is comprised of over 4,000 MVPs worldwide, and offers you a rich and diverse network of talented tech experts who can help you learn new skills and advance your career. As an MVP, you also gain insider access to Microsoft product development teams, and a plethora of other benefits. Come join the discussion and learn how to reap the rewards of your best season yet!

How to disclose a security vulnerability

Carol (Nichols || Goulding)

It's 10pm, do *you* know where your PGP private keys are? You may be thinking that you're not a security professional; you won't be the one to discover a security problem. Think again! You know what a security problem looks like: you've probably coded up a few yourself! Learn from your past mistakes-- and mine-- and prepare your software disaster kit. Hear my story about the security problem in an open source project that I found and reported, and along the way I'll walk you through the things I wish I had known how to do before I got all worked up. We'll go over the simplest way to encrypt your problem report using someone's public key, how to generate a keypair for yourself so that the people you reported to can send a secure reply, and how to distribute your public key *now* so people can be sure of your identity. We'll also discuss the many possible meanings of "responsible" in this situation, and look at some case studies of disclosures that did not go very smoothly for one or more of the users, the reporter, or the vendor. Studies show you're 11.4 times more likely to need to report a security vulnerability than to fend off a zombie apocalypse: be ready.

How to Hire Programmers You'll Want as Teammates

Scott Drake

Few things will ruin a great team faster than a bad hire, but there are few resources to help you hire better. In fact, most advice for hiring programmers is bad advice because it is applicable to very few teams and organizations. In this session, you'll learn a hiring playbook that covers both technical ability and team fit. It includes the three types of programmers and how to decide which is best for your unique needs; the eight factors that reveal if programmers will play well together; how to talk about your opportunity so it attracts the right people; and how to perform interviews that are fun and informative. This session will help you stop winging it and make your next teammate a great choice.

Mastering GitHub

Keith Dahlby

GitHub is often used as a basic Git host, but its platform has so much more to offer. From simple and powerful issues and pull requests, to advanced features for power users and integrators, it's a tool worth knowing well in its own right. This session will review everything you need to know to master collaboration with GitHub, from best practices for GitHub Issues and how it represents basic Git concepts, to hidden features and the tools enabling its developer ecosystem.

Process Oriented Programming with Elixir

Chris Nelson

With object oriented programming, the core language abstraction is the Object. With functional programming, the core abstraction is the function. Although Elixir can be and is often described as a functional language, the concept of the Process is so central to Elixir and the Erlang OTP platform that it is worth considering what impact it has on application design.

This session will explore processes in Elixir: what they are and how they let us design systems. We'll look at the original vision for objects as expressed by Alan Kay. Next, we'll look at how Elixir processes work and see how they compare. Finally, we'll take an example application and see how processes can be applied to design.

Releasing the monolith on a daily basis

Vincent Kok

Struggling to get software released on a daily basis? Stressed about how to apply the same techniques that make companies successful with continuous deployment? Learn from the experience of Atlassian's Confluence development team on its journey from releasing once a week to every day.

The talk begins with the team's build and deployment process, providing insights into dealing with particularly large builds and tests and deployment complexities. Next, the speaker explores, in detail, the cultural and technical problems that prevented the team from making that transition quickly, including: slow builds, flaky tests, a lack of automation, the wrong mindset and dealing with release blockers, to name a few.

The talk concludes with a discussion of the strategies the team has implemented to resolve these problems, including: reducing complexity, defining ownership, setting and monitoring time limits and establishing a “culture of green.” Learn how you, too, can make continuous delivery happen in a real, (and not so perfect!), engineering organization.

Scale Your Node Application, Skip the Infrastructure

Matthew Williams

You don't have to look far to find yet another article on scaling your NodeJS app to handle large numbers of users. But the techniques covered usually involve becoming an expert in deploying hardware or a guaranteed minimum outlay of cash to handle your expected load. But what if there were a way to scale almost infinitely without having to worry about the infrastructure to run any of it. Using platforms like AWS Lambda, Azure Functions, and IBM OpenWhisk, you can focus on providing scalable functionality in your NodeJS application without having to think about any of infrastructure details.

In this session, Matt Williams will show you how to get started building a complex Node application on AWS Lambda from scratch. Starting with the standard CLI, we move to other frameworks like Node Lambda and Serverless to add more functionality to serve your users. We will consider some key architectural decisions that affect how the application is designed. And all the way along we look at ways to monitor the application to help find the bottlenecks. By the end of the session, you will be eager to start working on your next application on AWS Lambda

Threat Modeling for Secure Software Design

Robert Hurlbut

Threat modeling is a way of thinking about what could go wrong and how to prevent it. Instinctively, we all think this way in regards to our own personal security and safety. When it comes to building software, some software shops either skip the important step of threat modeling in secure software design or, they have tried threat modeling before but haven't quite figured out how to connect the threat models to real world software development and its priorities. In this session, you will learn practical strategies in using threat modeling in secure software design and how to apply risk management in dealing with the threats.

Main Session 06 - Thursday 15:30

Are You Ready for Chaos? Horizontal Scaling in a Briefcase

Matthew Groves

For many applications, it's better to "scale out" by combining lots of computers into a cluster than to "scale up" by packing more and more resources into a single box. This is called distributed computing, and it creates some fascinating engineering challenges.

Some background (like the CAP theorum) will be discussed .This session will also demonstrate distributed computing using a self-contained, fully-functioning computing cluster literally packaged in a briefcase. You'll see how the nodes communicate, how the cluster behaves when chaos strikes and nodes fail, and how this architecture impacts software design.

The "briefcase cluster" is running Couchbase, which will be used for demonstrations, but these principles apply to a variety of distributed computing systems.

Build a JavaScript Dev Environment in 1 hour

Cory House

There’s a staggering number of decisions to make when setting up a JavaScript development environment. In this rapid fire one-hour session, let’s cut through the noise and build a powerful, rapid feedback development experience. After less than an hour of work, we’ll run a single line of code to lint, bundle, minify, run tests, open the browser, and display our app. We’ll check a single command line to see feedback on all of this goodness. And every time we hit save, this wild circus will fire off automatically, hot reload our app, and report the status. If you’re writing JavaScript with plain old script tags, get ready to be floored. The future of JavaScript has arrived and I have great news: The pain is over.

Building Self-Defending Applications with AppSensor

John Melton

AppSensor is an open-source OWASP project that enables building self-defending applications with attacker detection and automated response capabilities. There are many security protections available to applications today. AppSensor builds on these by providing a mechanism that allows architects and developers to build into their applications a way to detect events and attacks, then automatically respond to them.

Not only can this set of capabilities stop and/or reduce the impact of an attack, it gives you valuable visibility and security intelligence about the operational state of your applications. The self-protection model benefits all types of applications. In particular, it has gained traction with developers operating in the cloud and on DevOps teams. The increased visibility and speed of response become critical at scale, and the need for this capability becomes evident. This presentation will cover AppSensor concepts, features, use cases, integrations, deployment models, and some lessons learned from deploying in the field.

Career Growth Questions You're Afraid to Ask

Cassandra Faris

As a developer, you’re constantly bombarded with job opportunities. Some even sound like super exciting things you’d like to do! How do you know if they’re actually right for you? Even if you’re not actively seeking a job, this is an important question that all developers need to answer. Drawing on real-life examples, this session will walk you through the career change and opportunity evaluation process. It will cover topics including benefits questions, intellectual property concerns, and assessing company culture. Most importantly, it will help you identify your career growth priorities and whether that seemingly shiny new opportunity meets them.

Catapulting Yourself Into Windows IoT Core

Ian Lee

Have you ever wanted to build an Internet connected catapult or maybe just a door that is unlocked via facial recognition? If so, then you need to know how to build hardware. One of the best things to come from Microsoft in recent years is Windows IoT Core. It's a Windows operating system that runs on the credit card size Raspberry Pi computer. With it you can build apps that can read sensors and control hardware as well as connect all that information to the Internet. In this session I'll give you all the knowledge you need to get you started in the exciting world of hardware development using the Raspberry Pi + Windows IoT Core by building an Internet connected catapult. You may want to bring a hard hat!

DDD for beginners

Rob Allen

Domain Driven Design focusses on modelling the domain logic. This talk looks at the components of the model layer of your web application and the options you have. How are you supposed to organise your models in an MVC application? What goes where? What is a service class, a mapper or an entity? We'll discuss the terminology and take a look at what Domain Driven Design is and see how you use service layer to provide the business logic for your application and hide your persistence code from your controllers. By the end of this session you will be equipped to create excellent, maintainable models in your projects.

Deep Dive Into ES6 Generators

Jonathan Mills

Generators provide a powerful and complex new tool for use in your JavaScript development. From managing asynchronous code to building sequential processes, generators provide a framework for cleanly structuring your applications' external processes. This session will dive deep into the world of generators. Escape the challenges presented by complicated promises and nested callbacks, and utilize generators to build code that is both functional and maintainable.

Don't write tests. Generate them!

Chris Keathley

Automated test suites are invaluable. They provide protection against regressions and can serve as a design tool when building new apis. But, despite this protection bugs still slip through. We could try to write more tests but attempting to cover every edge case is an untenable problem. Luckily, we can use property based testing to generate edge cases for us.

Originally developed in Haskell, property tests have spread to many other languages. In this talk we’ll discuss the basics of property testing, demonstrate how we can determine properties for our system, and look at real world examples of property tests using elixir.

Fragments: The solution to (and cause of) all of Android's problems

Michael Yotive

Introduced in Android 3.0, Fragments have become the standard to which developers can encapsulate portions of their user interfaces within an activity. Since being added, there has been controversy as to their usefulness and confusion as to where they fit within the Android lifecycle.

In this talk, we will explore the history of Fragments and the improvements made for Android 7.0. Also, we will look at the arguments against fragments and how the android developer community created their own frameworks for fragment-less architecture.

Overthrow the Trichromatocracy: Making Color-blind Friendly UIs

Scott Favre

Think your snazzy new UI is fantastic? Did you test it with color-blind users? No? Then you're probably in for a surprise for how much information you are trying to convey in a way that is not accessible to your users who have colorblindness or color sight deficiencies.

We will discuss the details of what it means to be color blind, some clinical details of the condition, and what you can do as a programmer or designer to improve the usability of your software for the 1 in 12 of your users who have a color deficiency.

Python 3: It's Time

Charles Yost

Being indecisive? Still using Python 2 for personal or professional projects? If so, there is good news. It is time to move to Python 3. Signifigant benefits come from using Python 3 over Python 2, and will be covered in depth. In fact, there has never been a better time than now to start using Python 3. Stuck with a legacy Python 2 project? Ways to make Python 2 code compatible with Python 3 will be discussed. And maintaining environments allowing side-by-side development in Python 2 and 3 will be included as well. Get ready, it's time for Python 3.

Taking a byte of Java Bytecode

Magnus Stahre

Java, Scala, JRuby, Jython, Clojure. What do all these languages have in common? They are all powered by Java Bytecode. With the current ecosystem of languages running on the JVM, it is becoming important to learn more about the least common denominator.

This talk gives an introduction to Java Bytecode, with practical examples and an implementation of a very simple JVM based language.

The rise of polyglot at Netflix

Mike McGarr

A quick scan through Netlfix's Github repository will inform you that Netflix has built their core cloud platform on Java. Our who cloud deployment platform is built on and for JVM applications, and it has served us well for years. But as Netflix evolves and grows, platforms like Node.js and Python have grown not only numbers, but are also being used in increasingly important systems. We need to start thinking about building tools to support a polyglot world at Netflix.

In this talk, Mike McGarr (Manager, Netflix Developer Productivity) will talk about the various tools and approaches we are employing to provide first class support for a variety of languages and platforms at Netflix. Mike will share some of the challenges of supporting a polyglot codebase as well as lessons learned for enterprises embarking this journey.

Toward a Better Front-end Architecture: Elm

Jeremy Fairbank

Amidst the overwhelming cacophony of competing JavaScript frameworks, Elm is a promising voice. Elm is a relatively new language that compiles to JavaScript. Elm is a functional language that encourages describing UI, state, and events in a declarative manner, while performing as good as or better than the current JavaScript framework hotness. With type safety and pure programming constructs, Elm promises code that has fewer bugs and is easier to reason about. In this talk, dive into Elm, exploring its syntax and more importantly its architecture. Learn about unidirectional data flow, modular and composable UI components, update functions, and commands and tasks for side effects. Ultimately, discover how functional programming with Elm's architecture can solve real problems. Leave this talk equipped to start utilizing Elm to construct non-trivial apps with more maintainable code and better determinism.

Zero Downtime Upgrades of IaaS Databases

Ross Smith

Upgrading relational databases without downtime is a challenge under the best of circumstances and with the limited control hosted solutions give it's even worse. Often time there's no way to get a deterministic backup, and promoting read replicas incur downtime. All hope is not lost, in this talk we will discuss a reliable strategy for upgrading IaaS databases with no downtime and a minimum of acrobatics. The technique uses an intermediary read replica to provide a consistent place to start replication on a new database.

Main Session 07 - Thursday 16:45

"What's the Worst That Could Happen?" - Practical Security for Mobile Developers

Jeff Kelley

It seems like every day, there's a new security vulnerability in the news. Whether it's an OS update that you need to install *right now*, a hack that makes you change your social media passwords, or something much more sinister from a government organization, these stories can be alarming for the experienced developer and layperson alike.

In this talk, we'll cover two main topics: first, what the actual threats are to your data, how seriously you should take them, and some simple steps you can take today to make your devices more secure. Second, we'll cover what this means for the apps you create. From storing your users' data securely on-device to the open-source tools you choose, we'll look at how you can help keep your users safe and prevent your app from being tomorrow's security theater blog post.

A Primer to Developing for VR

Gabbie Gibson

VR is the new hotness in interacting with the digital world. The GearVR, Oculus Rift, and Vive have created an amazing landscape for interacting with 3D worlds - a landscape that will soon be a billion dollar industry. Just like in the early days of mobile, the opportunity to craft exciting new products and games is wide-open.

In this talk, I will give you the tools to start developing for VR. I'll also cover how to create intuitive interactions, and present tips and tricks for dealing with motion-sickness and locomotion.

Agile Swift

Godfrey Nolan

Step by step introduction to get unit testing, UI testing, mocking and continuous integration up and running for your Swift projects. The session will cover mostly iOS projects but we'll also look at running Swift on the Linux platform.

Big Data = Big Testing

Rob Sinkko

When Progressive Insurance decided to move its 15 billion miles (80 round trips to the Sun) of Snapshot driver data into a Big Data solution, one of the key questions was "How do we ensure it all copies over accurately?" This was no "copy this table over to that table" kind of project. It involved using a brand new architecture, taking 20 tables and compacting them into 7 tables, recomputing and adding several fields, ensuring accuracy of all the data for the analysts, and building tests that could be run quickly and repeatedly. Manual testing would be impossible at such a scale so an automated testing solution was developed. Attendees will learn in this session: how this daunting task was approached, why certain testing decisions were made, the pitfalls and successes of the testing effort, and how to mitigate testing risks for their future Big Data projects.

Bringing Up Our Future - On Mentoring Junior Developers

Sarah Dutkiewicz

While it is one thing to learn from videos or blog posts, it is a different experience when you have someone to turn to that can answer the questions you have when you have them. It makes a world of difference to be able to ask someone "What do they mean when they say 'inheritance vs. interfaces'?" When you have someone who forces you to think about what it is you want in life and what steps you may need to take to get there... someone who can make introductions and offer direction... it makes your career and life a little less scary and a lot more manageable. In this session, we will look at some practices that are used by great mentors in the field and learn lessons on how we can be great role models and mentors for our future - junior developers!

Caching Made “Bootiful”!

Viktor Gamov

Needing more performance from your Java applications? Is latency causing you stress? Repetitive loading of the data in applications, burning CPU time, taxing I/O / disk access? Too many applications caching the same data sets pushing the limits of your data management and application architecture? If so, take a look at JCache! This code-driven session demonstrates how to integrate Hazelcast Distributed Caches into your Spring or Java EE applications. Adding a bunch of annotations to a method can achieve an orders-of-magnitude speed improvement in applications with high-latency.

Can I Build a 12-Factor App in .Net?

Eric Kepes

The 12 factors (12factor.net) for building a modern application look pretty daunting. Some of them even go against the way we build in .Net. In this session, we will look at why these factors are important to all applications. We will see why you should take them seriously. We will see how to apply them to a .Net application to ease deployment and support of you applications.

DevOps Has Always Been About Security

Warner Moore

Do you need to secure your leading technology product and business? Are you struggling with aligning your security program to agile and devops? Organizations can be compliant and secure while remaining innovative. This presentation explores innovative processes and practices for evolving traditional security controls. With these techniques, you can securely enable the speed of change needed to disrupt in today's market place while meeting compliance and audit requirements. Warner Moore will explore alternative methods using real world examples for addressing security controls such as separation of duties, change control, and release management using automation and modern practices.

Dialyzer: Optimistic Type Checking for Erlang and Elixir

Jason Voegele

Static typing versus dynamic typing is an age-old debate amongst computer scientists and programmers, and the fact that we still argue about it suggests that there is no single right answer for all circumstances. But what if we could have the best of both worlds by combining the safety guarantees of static type systems and the freedom and flexibility of dynamic type systems?

In this talk, I will present an introduction to an optimistic, gradual type system as implemented by the Dialyzer tool for Erlang and Elixir. I will highlight the differences and trade-offs between static and dynamic typing, and present optimistic, gradual typing as a good compromise.

Human Readable Markup with Semantic UI

Nate Taylor

Is CSS and layout a mystery to you? When you read HTML are you able to understand what will be displayed without constantly looking up classes or documentation? If you’re like me, these are problems you run into when doing the view layer of your application. CSS classes become meaningless combinations of letters that don’t really seem to express intent.

However, there’s a library to help with that. Semantic UI is a framework that aims to create rich UIs “using human-friendly HTML.” Gone are the days of col-md-3. Instead, our HTML will have classes like “three column wide”. But it’s more than that. Semantic UI has a wide variety of elements, collections, and views that can make any application pop, all while following sane patterns.

This talk will examine several key UI components, and talk about the underlying principles of Semantic UI. After leaving the talk, you’ll be able to install Semantic UI (hint: it’s really simple) and get up and running on your next project.

Learn the Lingo of Languages

Amanda Cinnamon

Are you intimidated by conversations where terms like “dynamic typing" and "imperative languages” are thrown around? Do you consider yourself a rockstar coder, but lack confidence when it comes to behind the scenes operations of compilers, interpreters and programming languages? This session decodes the jargon used to describe programming language principles and practices. You will learn how to intelligently compare and contrast languages, a power that you can wield to choose the best language for your project, level up your interview game, or simply dominate the language superiority flame wars. The choice is yours.

React on Rails

Josh Graber

ReactJS is an excellent tool for building rich experiences for the web, but integrating it with Ruby on Rails presents some unique challenges.

This session will explore three approaches with increasing degrees of complexity for using React with your Rails application.

The first demonstration will use the “react-rails” and optionally “sprockets es6” gems to write simple ReactJS components and distribute them with Rails applications.

Next you'll see how to use webpack with babel to write pure ES2015 modules for ReactJS components to enhance Rails applications.

The final example is a full scale front end React application built on Redux with UI routing. This application will interact with a Rails JSON API.

We will discuss the strengths and challenges to each approach. All approaches will be discussed and demonstrated with code samples.

Real-time Server Telemetry: The downfall of logging and rise of data pipelines

Stephen Shary

Logging in modern applications is a feast or famine. Every scrap of information is needed when triaging production issues, but who can find relevant information among all the DEBUG and PERFORMANCE lines? Logging to files and STDOUT has become incompatible with cloud-native applications and distributed systems (and most other things). Modern telemetry and APM solutions need to tackle the issue of finding needles in haystacks as well as view the system from 50,000 feet with real-time data. Hear the story of our experiences at Kroger creating an open-source-based system including Elastic Search that handles over 900GBs of data each day. See how we visualize our current production status and discover problematic applications, servers and systems.

Stranger Streams: How to RxAndroid

Michael Yotive

Functional Reactive Programming and the RxJava/RxAndroid libraries are a much talked about subject these days. But for someone new to these concepts, things may seem a bit strange.

In this talk, I will introduce you to the reactive extensions and showcase how they can make complex tasks on Android easier.

Testing at the boundaries: using Consumer Driven Contracts to keep your microservices in sync.

Andrew Fitzgerald

As the number of microservices in a system grows, it becomes much more difficult to stand up an entire system to test service to service interactions. You can mock out dependencies to test single services in isolation, but how can you be sure that your mocks stay up to date as the system evolves?

In this session we will explore the concept of Consumer Driven Contracts. You will learn how you can: * formalize the interactions between two services into a contract * keep your contracts up to date as a side-effect running your tests * test services in isolation by generating mocks from contracts * safely evolve the interfaces of services while maintaining compatability * bonus: use CDC to test the interactions between a web app and the server

Main Session 08 - Friday 08:30

Automating Security in Building Software

Warner Moore

Everyone's concerned about security from your peers to your board, but what does that mean to building software? This presentation will explore techniques for embedding security into your Software Development Lifecycle using automation and aligning to your existing practices for building software. Better yet, many of these automation techniques align to DevOps culture and practices. Building secure software doesn't mean slowing down delivery or adding meaningless paperwork – it can complement your favorite ways to build software!

Creating a responsive application using Reactive Extensions

tamir dresher

Reactive applications are designed to handle asynchronous events in a way that maximizes responsiveness, resiliency, and elasticity. Reactive Extensions (Rx) is a library that abstracts away the sources of events and provides tools to handle them in a reactive way. With Rx, filtering events, composing event sources, transforming events, and dealing with errors all become much simpler than with traditional tools and paradigms. Reactive Extensions requires a change of mindset: you'll learn to think about your application as a message hub that knows how to react to messages. After this session, you'll better understand what Rx is, and you'll have a starting point from which to effectively use it in your application.

Docker Container Lifecycles – Problem or Opportunity?

Baruch Sadogursky

Docker is hot. However, as Docker container use spreads into more mature production pipelines, there can be issues about control of Docker images to ensure they are production-ready. Is a promotion-based model appropriate to control and track the flow of Docker images from development to production? We will demonstrate how to implement a promotion model for docker images, and then show how to distribute them to any kind of consumer, being it a customer or a data center.

Drops of Jupyter in Your Hair

Brian Sherwin

Engineers and Scientists love their notebooks. Data Scientists love using their Jupyter notebooks. When you are creating documents, wouldn't it be great to include live code, real-time visualizations as well as text describing what's going on along the way. This isn't just about documentation that tells how to run the code…these notebooks can run live code, modify data, call web services and more. In this presentation, we’ll examine how to create, execute and build living notebooks that can bring your code and data to life.

Horizontally Scaling Node.js and WebSockets

James Simpson

Underneath every breakout website or app is a horizontally scaling back-end, but how do we get from a single process Node.js server to a highly-available, auto-scaling system? In this talk, we’ll take a high level look at a full production stack before getting our hands dirty with the secret sauce: Node.js, WebSockets and Redis. Through a live coding demo, you’ll learn how to take a single-server app and scale it infinitely. Walk away with a better conceptual understanding of high-scale web systems and practical tools to start implementing these techniques in your own projects today.

In the Loop - Automated Testing with Hardware

Ben Rogers

Hardware-in-the-Loop [HIL] testing is the gold standard for establishing firmware reliability on your real hardware. Historically, this has been done with big, custom racks of equipment at great expense.

This session will cover the basics of small-device testing: test rigs, firmware updates, logic analyzers, and protocol simulators. Then we'll move on to automation, scripting, and continuous integration.

In short, you'll learn some new best practices for releasing world-class firmware on your devices.

Introducing Managed Effects to Enterprise Front-ends: What, Why, and How

Kofi Gumbs

Effect management sounds like an abstract paradigm with few tangible benefits. Thus, many dismiss it out of hand. That is a mistake.

Managed effects are natural extensions of common design patterns. It emphasizes separation of business logic and implementation details on a whole new level.

This may sound like a big change, but it is easy to adopt effect management in incremental steps: no re-writes required!

Using Elm as an example, the speaker will explore the observable benefits that managed effects can have on your project. You will walk away with a new appreciation for the paradigm and the knowledge required to start implementing it today.

Make Donuts Great Again: The Tale of the Broken Build

Jay Harris

It's a dog-eat-dog world. We celebrate individual speed, strength, and intelligence, embodied within our workplace as more hours, less vacation, and my-code-is-better-than-yours. But software development is not an individual sport. If a compile fails, we all suffer. If a deadline is missed, we all feel the pain. This model is broken, and we all need to fix it. To build projects together, we must start building projects together. No more individual scores. Celebrate teamwork, collaboration, and co-ownership. Let's build, united, as We.

Making a Successful Component Library: Digital Brand Style Guide

Kevin Mack

Creating a component library is not a new piece of the development and design process but with the growth of responsive and contextual factors it has become a requirement for complex systems. Digital Style Guides bring alignment of teams and solve complexities by providing quality tested code that is fully reusable across all projects. Within this presentation, we will discuss the variations of different style guides, toolkits, and frameworks that exist today and identify the right solution for your team to create a successful component library.

Mashing Up QA and Security

Craig Stuntz

Security is domain specific quality assurance, but developers, testers, and security professionals often don't work together. When this type of disconnect exists between big groups of people who are very good at their jobs, there is usually a mostly untapped potential for learning. I've been exploring this landscape by writing an open source fuzzer aimed at discovering new test cases (not just crashes!) using binary rewriting of managed executables and genetic modification of a test corpus, implemented in F# and using Mono.Cecil. I'll contrast the fundamentals of each discipline, demonstrate tools used by experts on both sides of the security and QA fence, and challenge the audience to find new ways to mix them up. Expect to see lots of code and leave with ideas for making entire communities better, not just your own team!

MongoDB Aggregation like a Champ!

Nuri Halperin

Got lots of data? Need to run some reports? You don't need add-ons. You don't need map-Reduce magic. You don't need to install or pay for extra platforms. All you need is MongoDB, a shell prompt and some aggregation framework know-how! This session is all about the aggregation framework in MongoDB. It explains the pipeline architecture, major operators and how to put it all together.

Also covered will be some new operators in MongoDB 3.2 , and what to do about the pesky lack joins!

If you are still exporting data just to do dig into your data or do aggregation, there's a much better way!

Picture This: Automated Visual Regression Testing

Jeff Koenig

For years people thrown around the idea of visual regression testing, however, it typically has been limited to only a single browser and for static web pages. Imagine a world where when you make a code change you are able to detect unexpected distortions in your web application across all supported environments with minimal effort!

In this talk I will discuss how I built a framework to do Visual Regression Testing using Capybara, Compatriot and BrowserStack. I'll discuss what I learned and how you can get started doing it today.

The Millennial's Guide to Freedom™ - Tales of the #remoteworking #millenial #vanlifer.

Antoine Meunier

Quarter-life crisis is a bitch. For us, millenials, it seems to be the fear of leading the same boring life as our parents – sorry boomers.

So, when bits and bytes convert into a source of income and remote working becomes part of one's life, there is little choice but to jump head first into freedom by taking on the road leaving everything behind.

Right?…… RIIIIIIGHT?

Millennial or not, take a seat on this wild ride through the good, the bad, and the fugly of going for the #vanlife and working remotely.

This is the (cautionary) tale on the remote working movement and work-life balance as we know it – as told by a digital nomad who gave up the posh lifestyle and the trendy appartment for a 1972 Volkswagen bus and a pair of wet socks.

The Rust Language: Memory, Ownership and Lifetimes

Jeff Walker

Rust is a new systems programming language that provides functional programming features with "zero-cost abstractions" and brings an innovative approach to resource and memory management. Memory is managed through rules of ownership, borrowing and reference lifetimes enforced by the borrow checker at compile time. This provides safe, deterministic resource management and "runs blazingly fast, prevents segfaults, and guarantees thread safety." Learn the rules and a structured way of thinking about resource management that applies to code in any language. You'll also learn when you should consider using Rust for a project. Finally, take a peek at some languages that may bring this new approach to a broader audience.

Using iBeacons in iOS with Swift

Matt Nedrich

GPS is great for outdoor location sensing. Unfortunately, it falls short indoors. iBeacons pick up where GPS leaves off by providing a means for indoor location sensing using Bluetooth low energy (BLE).

In this talk we'll talk about what iBeacons are, how they work, and when you can use them. We'll also introduce the programming model to interact with them in iOS using Swift and the Apple CoreLocation framework. Live demo included!

Main Session 09 - Friday 09:45

Abusing C# More

Jon Skeet

What language could be complete without some horrible abuse? If you can't do terrible, evil things with it, how could you ever create works of great art?

Of course, anyone can write plain *bad* code. The trick for really *evil* code is to make it attractive; to make it seduce you with it utility, brevity and general glamour. Pierce that attractive exterior though, and the horrors are revealed, twisting language features into shapes they were never intended to take.

As C# has evolves as a language, as its feature surface expands, new crevices can be found, containing as-yet unseen terrors.

Anatomy of a Distributed Denial of Service attack

Aaron Kalin

Ever been to a website or tried to use an internet service that was suddenly not working? It may have been the victim of a Distributed Denial of Service attack or DDoS as they are called. They happen almost daily in varying shapes and sizes, but you don't always get the details of how and why. You'll hear the story of such an attack from someone who was a victim along with detail of how it happened, why it happened, and what they are doing to protect themselves the next time it happens so you can too.

Become a Remote Working Pro

Michael Eaton

There is no doubt about it, working remotely can be an amazing experience - no commute and wearing pants is optional! Of course, it's not all rainbows and unicorns since there are challenges to overcome and of course work to be done. This interactive session will answer questions like, "how can I convince my boss to let me work remotely?", "how can I make sure they don't forget about me once I start working remotely?" and many more. It will equip you with the tools and techniques for being a successful remote team member.

Building for the PHP Command Line Interface

Steve Grunwell

Executing PHP from the command line enables us to interact with our applications in new and interesting ways: from performing site maintenance to scaffolding new projects, CLI tools like WP-CLI, Artisan, and Drush make it easy to interface with our code without ever opening a browser.

Attendees will be introduced to popular PHP CLI tools and their default capabilities. We'll discuss characteristics of good CLI scripts, strong use-cases for writing custom commands, then write several CLI programs across different platforms.

From Zero to the Actor Model

tamir dresher

The Actor Model that was introduced in 1973 looks at the world as a set of actors that holds a state and a behavior. This idea is very similar to OOP that we all know and love, but actors are not limited to a single machine. Instead, actors can be distributed to multiple processes and machine, and create a network of processors that encapsulate the problem-domain. This model has significant implications on the way we can write large scale distributed systems and can make very hard problems, very easy to solve. In this session you will get to know Akka.Net library that makes the Actor model a piece of cake.

Functional Reactive Programming with JavaScript

Jonathan DeJong

What is functional programming? What is reactive programming? Taken together, these two programming practice can help break the cycle of unmaintainable spaghetti code in large applications by producing predictable, testable, composable code. In this talk, we'll cover the fundamentals of functional and reactive programming in JavaScript. We'll look the reactive extensions project (RxJS), as well as Redux, and see how these can be utilized to manage complex intereactions and produce predicatable systems. We'll also look at how these practices can be used to make life easier in major front end frameworks, including Angular 2, React, and Vue.

Getting Started with the Particles: Photon & Electron

Kristen Smith

From www.particle.io : A tiny, reprogrammable Wi-Fi development kit for prototyping and scaling your Internet of Things product. The Photon is a $19 tiny Wi-Fi development kit for creating connected projects and products for the Internet of Things. It's easy to use, it's powerful, and it's connected to the cloud.

In this session, we'll build a few simple programs to show off some of the basic functionality of this versatile (and adorable) device. From a blinking "Hello, World!" LED to a sensor-driven "When to Water Your Houseplant", we'll explore some of the countless project possibilities the Photon enables!

Then we'll check out the Electron! Where the Photon gets your Things connected via Wi-Fi, this particle gets them connected via 3G. We'll expand on what we learned from the Photon to build a 3G-enabled DIY GPS tracker using the Electron!

How to write Java web apps like a JS hipster

Stephen Shary

It's easy to find a web-developer position at an enterprise company and spend multiple years writing code in a language that was bleeding edge 10 years ago. Passionate developers go home and continually spend time learning new languages and writing in new styles that allow them to stay up-to-date. Not everyone is up for that lifestyle and the daily grind can be miserable when you're limited on technology choices. This session focuses on the changes that have been made in the past couple of years to allow Java to go through a "renewal" so that developers now have the ability to write code that is so similar to modern Javascript server applications. It will demonstrate how Java8, Spring Boot, AsyncIO allow developers to write code in a modern way that is concise, fast, scalable and just as "cool" as the startup in the loft downtown.

Migrating the Monolithic to the Microscopic

Michael Collier

Many people are talking about adopting microservices and moving away from the monolithic application. But how do you go about moving to a microservices based architecture? How do you host your microservices reliably on Azure Service Fabric? In this session we’ll provide a brief review of microservices and Service Fabric principals. We'll then quickly move into a through discussion of field proven strategies for transforming development and IT teams to utilizes these technologies. We'll review real-world customer examples of how to best handle deployments, monitoring, securing, and generally managing services with Azure Service Fabric.

Programmers don't have to suck at UX and UI

Sebastian Hermida

As a programmer, you apply patterns and principles to your code every day. You manage abstractions, refactor your codebase, and write unit tests.

However, most of us are lost as soon as we have to touch the UX and UI side of our application. We hesitate to make changes. We find that our UI design sometimes looks "odd". Things don't feel natural.

Let's fix that! In this session, we are going to look at the programming models and concepts that you already know and apply them to the UX and UI world.

Push it (Push it Real Good)

Lyndsey Padget

Git. It can be intimidating if you're accustomed to other kinds of source control management. Even if you're already using it and comfortable with the basics, situations can arise where you wish you understood it better. Developers often just want to write code and tell everyone else to take a hike, but the reality is that most of us work on teams where the feature-based code we write must be integrated, tested, and ultimately released. This session will cover the most critical git concepts, basic and advanced, in a completely visualized way. At the same time, you'll pick up git terminal commands to help you understand (or even eliminate) a git GUI you already use. Go beyond the basics to learn how to get yourself out of a git pickle, practical release management strategies, and more.

Testing your Swift Code

Matt Nedrich

Swift has experienced fantastic adoption over the past few years as developers transition away from Objective-C. Unfortunately, some of the tried and true Objective-C testing strategies don't easily apply to Swift. In this talk we'll take a deep dive into ways to test your Swift code. We'll cover unit tests, UI tests, and introduce some helpful testing libraries. We'll also talk about how you can take your tests to the next level by setting up a continuous integration and automatic deployment workflow.

Tips and Tricks for Testing Lambda Expressions in Android

David Carver

This session will address some tips and tricks to help bring some sanity to unit testing when dealing with Lambda Expressions. Lambdas provide some great benefits to making implementation of listeners and interfaces easier. They provide a much more compact way to represent Anonymous inline overrides, but they also run into the same problems when writing unit tests. As RXJava is being used more and more to make asynchronous code easier for Android, it is making it more difficult to unit test as well due. Forcing more integration testing and less unit testing. Causing in many causes for tests to no be written.

We will look at some refactoring techniques that can be used, look at some situations where using a lambda may not be the best way to accomplish things. We'll take a look at tools like Butterknife that can accomplish similar things but in a more straight forward way to allow unit testing to occur. In addition we'll look at some ways to test those RXJava Lambda expressions. These patterns come from experience on a large retail project that has over 1000 tests and heavily leverages RXJava, Retrofit, Butterknife, Dagger, and many other square related frameworks.

By the end, the attendees will have some tools they can take back, and apply to their own projects to help make unit testing easier when dealing with Lambda expressions.

Using the GPU to Write an Elegant AI Algorithm using CUDA

David Hauck

During this talk we will create a simple AI algorithm using CUDA on an NVIDA GPU. With Big Data becoming such a large part of business, people are starting to turn to different methods to crunch data. One of these methods is using Graphics Processing Units to do the work on a massive scale. This talk will be using CUDA, a platform created by NVIDIA to create programs that run on GPUs. Specifically, the class will show creating a basic CUDA AI program using the k-means clustering algorithm. Writing a program to run on thousands of threads at a time can be vastly different from traditional programming, so come prepared to think outside of the box!

Write Better JavaScript with TDD

James Bender

JavaScript has become one of the most popular programming languages. Many of us who have been working with statically typed, OOP-orient languages like C# or Java have started working more and more in JavaScript. But while most of these developers who been using TDD in their former language come to JavaScript, they often leave this practice behind. And it’s understandable why developers who are familiar with unit testing in C# or Java may struggle in JavaScript. The language is different, the tooling is different, and all those cool frameworks that we use to build our applications can add complexity to the situation. The good news is that unit testing your JavaScript is easy, and most frameworks have testability already built in! In this session, you'll see how to make sure your JavaScript is easy to test. You'll be introduced to the most popular testing tools that help you write tests, even for JavaScript that uses frameworks like jQuery Angular and React. And you'll see how TDD can help you start writing better JavaScript.

Main Session 10 - Friday 11:00

An Experience Report from Building a Large React.js/Redux Application

Jeremy Miller

I have been building and refining a large React.js based application over the past couple years for the open source Storyteller 3 project (http://storyteller.github.io). From crude beginnings with early React.js and a homegrown Flux-like architecture to the latest version and Redux, I've learned a lot about structuring and testing reactive clients based on React.js's ideas of uni-directional flow and component composition. In this talk I'll show how I've been able to use Redux to drastically simplify state management across various parts of the application. I'll also talk about the technical challenges I faced while trying to integrate the React.js UI with a rapid stream of events coming from the server through Web Sockets. I'll talk about composing state changes to the underlying Redux store with Immutable.js and the efforts I had to go through to eliminate unnecessary screen component refreshes in order to make a responsive application. Finally, I'd like to show how the integration with Redux can make your application much more efficient for automated testing of your client.

Canopy: Easy Automated UI Testing for the Web

Eric Potter

Automated UI tests are a great way to quickly validate full stack behavior in your application. They complement unit tests by testing your entire system. Canopy is a powerful and intuitive open source tool for writing automated tests for web applications. It provides access to the power of the Selenium WebDriver through a DSL that is easy to learn and use. The DSL is built on top of F#, giving you the full power of that language and its ecosystem. In this session, we will look at how to get started, even if you have no F# experience.

I'll cover how to create your first tests, how to make your tests rubust, and how to build up a test suite. I'll also show you how to make use of some of F#'s more powerful features.

GraphQL: What it is and why you should care

Chris Nelson

GraphQL is an open specification created by Facebook for exchanging data between a client and server. It turns REST on its head by letting the client ask for exactly the data it needs rather than the server sending whatever it thinks is best. In this talk, attendees will learn about what GraphQL is and the basic syntax. We’ll see how it works on the client and the server, and how it can be used to solve some of the most frustrating problems in modern web app development. We’ll also see how powerful features like schema introspection allow you to do things in web development we’ve never been able to do before. Expect to come away with a good understanding of where and how to harness GraphQL on your next project.

Microservices: Lessons from the Trenches

Gregory Beamer

Microservices are all the rage. They are the silver bullet of architectural styles. But what does it take to implement them and make them work? What are the foundations to build using this architectural style.

In this session, you will learn about microservices from a pragmatic standpoint, based on about 2 years of experience in consulting on the architectural style. Rather than look at the purist approach, as outlined by Martin Fowler, and others, you will learn what works and what doesn't, based on experience in the field.

Session includes * Foundational topics necessary to implement microservices * Basics on the architectural style as they apply to real world problems. * Necessary It and shifts to implement microservices in the Enterprise

More Money with Less Effort: Validate App Ideas Before Coding

Scott Showalter

Building apps is time consuming, without any guarantee of success. Countless apps, littering the App Store, Google Play and beyond, were born out of a developer's personal pain, yet have made little or no revenue. Before you build that "killer" app, you can indeed know beyond the shadow of a doubt that there's actual demand for that app—that people will actually buy, use and love it—but how? Validate! Join Scott for an amusing, fast-paced workshop that teaches you several techniques that you can use right away to prove your ideas have worth, or otherwise lead you to insight that helps you transform them into incredibly desirable products by building only the parts people would actually use. Learn to discover your early adopters and first paying customers, even before you launch the app, divine whether or not your idea solves a problem people care about, what to avoid when engaging in user research, the three key questions that will lead to early money and early adoption, tools to develop codeless prototypes and ways to scale these mobile app validation concepts to desktop software or Web apps/websites

Pair Testing: The Secret Sauce of Agile Testing

Jess Lancaster

Pair Testing, say what? It’s like Pair Programming, only you’re testing! And it’s the secret sauce of Agile testing. At TechSmith our testers not only use pair testing to make better software we make better teams along the way. In this session we’ll explore why pairing works, the many benefits of pair testing, why you as agile team member need to be pairing, and how to get started. I will also share my team’s pair testing mistakes and what we did to improve. Oh, and I’ll make sure you get my pair testing recipe for making this secret sauce as soon as you're back at work. Whether you’re a tester, developer, manager, or other team member, I'll show you how pairing will supercharge your team’s testing efforts!

Personas for Gamers: Character Sheets for your Users

Jonathon Baugh

Imagine that you are a big hulking barbarian and you need to cast a spell to save the poisoned king from death. Not going to happen. That is the situation we are presenting our users with when we build software for the wrong persona.

This session will be a hands-on workshop for anyone in the tech industry exploring how and why we can use personas in a meaningful way. To make it more tangible, we'll frame the concept of users as characters in an RPG. We'll center the lessons learned around a fictitious project, making it real and will include take-home materials. At then end of the workshop you'll have experienced how to effectively bring a human element to your project through personas.

Programmer Bedtime Stories

Greg Bulmash

Enjoy some algorithms as allegory…

Hear the cautionary tale of "The Recursed Army," the importance of clearly defined terms in "the F.arm(er) and the GigaCow," the observer pattern helps a king navigate palace intrigue and royal succession in "King Floyd and the 17 Princes," and variable scope creates the locked room mystery "A Dame, Two Spells, and a Scope."

Quick! Check your Properties (and Write Better Software)

Paulmichael Blasucci

Love it or Loathe it, testing is an unavoidable aspect of developing high-quality software. But there's a world of difference between well-tested code and simply writing tests. Very often, the difference boils down to how you assess the "correctness" of your code. By focusing on software as a mapping from inputs to outputs, property-based testing offers a powerful methodology for testing the logical invariants a piece of code should always uphold, regardless of implementation details.

This session presents a review of FsCheck, a powerful library for doing property-based testing and sophisticated data generation in the CLR ecosystem. In addition to the basic features of the library, attendees will learn about identifying common invariant patterns and building specifications from collections of properties. Time will also be spent exploring how FsCheck can be used to refine domain models and integrate with other popular testing tools. Finally, real-world examples will highlight both the strengths and weakness of this approach to software craftsmanship. While no prior knowledge is assumed, familiarity with the CLR (.NET or Mono) will be helpful.

S.O.L.I.D. Elixir

Dave Shah

S.O.L.I.D. - for roughly a decade, many have learned and sought to apply these principles in the context of Object Oriented Programming. As Functional languages like Elixir become more mainstream, one might wonder if these concepts are still applicable and, if so, seek out concrete examples that demonstrate their merit in this new functional world. Come prepared to tour each of the S.O.L.I.D. design principles and to examine how S.O.L.I.D. principles not only apply to Object Oriented Programming, but also to Functional Programming in Elixir.

The Secret to using HoloLens Spatial Mapping in Unity

Chad Carter

One of the most exciting aspects of the HoloLens is the device's ability to do spatial mapping. This is the ability the device has to scan the environment and understand it as a 3D mesh. You can then write code to handle working inside of this environment.

The game Fragments uses the data to either position actors sitting down on your couch, or standing in the room. The game Young Conker places the level objectives and obstacles based on the environment that was scanned.

In this talk, Chad Carter, a HoloLens and Unity 3D consultant and instructor, will show you how to use Unity and HoloLens together to work with the spatial mapping data.

You will learn, step by step, how to setup the development environment for HoloLens, how to deploy Unity code to the HoloLens, how to import the scanned meshed from the HoloLens back into Unity so that the development iteration cycle can be shortened.

Thinking Like a Hacker

Chris Maddalena

When it comes to security and vulnerabilities, it can be difficult to understand how vulnerabilities are found and how different vulnerabilities can be strung together. Whether you want to be a penetration tester, move into application security, or just understand how this all works, the first essential step is thinking like an attacker. This is an often repeated idea in the secuirty industry: to excel at defending, you must understand the attacks, and vice versa. Switching into an offensive mindset takes effort and practice.

This switch requires breaking old habits and ideas. Instead of testing if an application will accept the intended input, you need to learn to twist your usual thinking and look for ways errors and different functions can be abused. This can be difficult to do when you are accustomed to only thinking about what your intended user will do with the application. This presentation will discuss basic concepts used by security researchers (e.g. fuzzing) and how penetration testers, and less friendly attackers, will attempt to break an application for their own designs. The presentation will include demos of a couple of offensive tools and stories detailing how attackers were able to map the design of an application and abuse it.

Tune SQL Server Like a Guru - The Big Three!

Kevin Boles

EVERY client I engage with is taught to RUN 3 SCRIPTS IMMEDIATELY, WITHOUT THOUGHT AND WITHOUT DELAY as soon as a performance problem is noted. I am exceptionally good at performance tuning and forensics, but I (nor anyone else out there) cannot get at the root cause of a problem without information about the occurrence. These scripts will give you said information (which is very often transient in nature)! I could be putting myself out of work teaching stuff like this! :-D

Twelve Factor Revisited

Matt Williams

Originally devised by Heroku, a twelve factor application is a methodology for building software-as-a-service applications.

At first glance, it seems geared toward application development. However, like an iceberg, there is far more hiding beneath the surface.

12 Factor will change the way that you build applications and how you think about systems. The first step, though, is to recognize that we have a problem -- traditional methods are unmanageable. They do not easily scale. Dependencies are hard to isolate. Concurrency is hard. Logging and monitoring are an afterthought. Backing services are closely coupled. Servers and applications follow the 4-H model -- we raise them with loving, devoted care until it's time to slaughter them.

12 Factor addresses these issues and many more.

This session discusses the 12 Factors and why they are important to DevOps.

Twelve factor is not just for developers any more.

Your First Open Source Project - Get Started Right

Jeffrey Fritz

Have you thought about starting an open-source project? Are you interested in getting involved by contributing to a project? Is your business thinking about releasing some code to the open-source community? In this session, we will look at how to get started contributing to a project and what you need to look out for. We will take a look at starting a new project and how to get free resources to support your efforts. We'll look at some common pitfalls and tips that you need to know to get on-track. This session will be lead by one of Microsoft's program managers who leads several open-source projects and collaborates with the .NET Foundation. Attendees will receive all the information they need to start a project, help a project, or release some code to the community successfully.

Main Session 11 - Friday 12:15

A Crash Course on Building Microservice-based Architectures

Shawn Wallace

Driven by popular DevOps stories, many large and successful companies are implementing and singing the praises of microservice-based architectures. Is it time for your organization to get on board too?

In this session, we'll discuss: * Why implementing microservice-based architectures is something to consider * How this approach is different from "traditional" service-oriented architectures * How microserice-based applications are designed and implemented * Benefits and possible pitfalls to work around * Tooling tricks to give you a head start to "predictable" success

We'll also talk about topics such as state management, data persistence, transaction support and the role of the enterprise so you can better decide if you should stay with your current process or go forward with a microservice-based architecture.

An Applied Introduction to R

Gary Short

In this session I'll will provide you with a comprehensive introduction to the R statistical programming language, via the medium of analysing the sports trading exchanges. We'll cover all the basics of the language as well as how to visualise results. By the end of this session you will have an appreciation of the language and the confidence to go forward and explore data in your favourite domain.

ASP.NET Core Identity Management

Ondrej Balas

Injecting custom code into authentication and authorization in ASP.NET has always been a chore. ASP.NET Identity is a library built to replace both ASP.NET Membership and Simple Membership, making it much easier to implement custom authentication and authorization without the need to rewrite core components. In this session I will go deep into the abstractions that ASP.NET Identity builds atop of, and show how to take advantage of these hook points to implement a custom membership system.

Automating the Gaps of Unit Testing Mobile Apps

Geoffrey Goetz

Apple has provided an orchestra of instruments for developers to use to fine tune their apps and create the best user experience possible. At compile time, within the simulator, on the device and in the field. Apple offers some solid advice to offer developers on how to make their apps the best they could possibly be. This session will look at the series of tools available from using static analyzers, to knowing when to use the simulator, and when using the device is required to collect the data necessary to understand just how an application is performing. It will also show how to gather information from testers that are testing the app live in out in real world scenarios rather then just in the team room.

Building failure resistant systems with circuit breakers

Stuart Ingram

Murphy's law is universal and constant, if something can go wrong it will go wrong, which is especially true of distributed heterogeneous systems. Failures can take many forms from a complete service breakdown to a single latent service causing a cascading catastrophic failure for your users or even intermittent service failures. This talk will discuss how to build resilient, highly available systems utilizing circuit breaker and bulkhead design patterns that help provide service and user guarantees regardless of service QoS breakdowns. See how visualizing the telemetry around service interactions, latency and failures can provide valuable early insights into growing problems before they affect your customers. Learn how Netflix, one of the largest examples of a distributed system, implements these patterns at scale and how you can apply them to your infrastructure big or small. Failure is now an option when implemented the right way!

Design Before Code: Thinking About Accessibility from the Ground Up

Caitlin Geier

In web development, accessibility (i.e. making sure people with disabilities can use your site) is often thought of as a technical challenge which can be overcome by teaching developers to use proper semantic elements and well-placed ARIA attributes. In practice, making a site or application accessible usually involves a developer hacking away at the code (and the design) until all of the automated tests pass. If you start with a design that takes the needs of people with disabilities into account, much less time and effort will be needed to make the finished product accessible. This session will include an overview of accessible design principles and will give some practical tips for designers (and for the developers who work with them).

Don't Write Secure Code; Design Secure Systems

Seth Petry-Johnson

The best way to build secure systems is to stop writing security-related code on a daily basis.

Developers have their hands full with complex systems, confusing business rules, technical edge cases, responsive UIs, etc. Security requirements, when they even exist, are repetitive to implement, hard to test, and often get crowded out by other demands. When developers handle security on a feature-by-feature basis, the result is a wildly inconsistent mess of security holes.

In this session developers and architects will learn real-world techniques for designing security into the application framework itself, rather than leaving it up to individual features. You'll see how to implement access control in your data access layer, declaratively handle permission checks with Attributes and Annotations, automate security testing with static analysis, and more. Come learn how secure software *design* can dramatically reduce the day-to-day burden of secure *coding*.

Code samples will be in C# and ASP.NET MVC, but the focus will be on techniques and concepts that easily generalize to other platforms.

How HTTP/2 Fits Into Your Workflow

Byron Delpinal

The HTTP/2 spec has been out for almost two years now, yet adoption has been (predictably) slow. With more and more companies putting a high priority on performance, it's time to take a serious stand against high load times and rid ourselves of jank once and for all. We'll start with a brief history of the HTTP protocol, fall into the HTTP/2 spec itself to figure out what's new, and finally learn how to enable this protocol on our own websites. This talk will cover browser support, server support, and CDN support, as well as how to cleanly overcome HTTP/1.x performance hacks like spritesheets and file concatenation.

How to Build The Internet

Andre D. Henry

What if you had to rebuild the Internet? Do we know its origins? How it got to be what it is? As professionals dealing with modern computing the Internet as all around us. We should understand what the Internet is and how it works.

In this talk I will discuss Internet architecture, protocls and software. We will also talk about what the average developer should know about Internet underpinnings and tools that you can use to diagnose Internet problems. We will also discuss how to write software thats well behaved on the Internet.

Incredibly Strange Programming Languages

Craig Stuntz

If you've ever suspected that “all programming languages are pretty much the same; they just have different syntax,” well, you will never suspect that again! Covering languages from the unusually powerful (Idris) to the illuminated (قلب) to the profoundly limited (BlooP), and all points in between, these languages will help you think differently about approaches to software problems you face in your day job. Of course we’ll have a lot of fun, but these languages are no joke. The practical benefit of an impractical language is the power to find new approaches to common problems.

Over-Achiever, Under-Believer: How to Match Your Confidence to Your Competence

Maureen Zappala

You’re talented, accomplished and successful. You are a technical expert or a high level manager. People think you have it all together. Yet you still think to yourself “Oh my gosh! I don’t know what I’m doing! And everyone is about to figure it out! They’re on to me!” Many smart and successful people suffer from the Impostor Syndrome: the persistent feeling of not measuring up to their resume. They are an overachiever, but an under-believer. It’s a cognitive distortion that is especially prevalent in technical and creative fields. If this is you, you are not alone. It’s estimated that 70% of people suffer from this. But there’s help! There are strategies that can silence that impostor voice so you can embrace your success and be confident to move further ahead without fear of being unmasked as a fake. You can be released from the clench of the counterfeit. In this session, learn 4 powerful techniques that can lead you to true confidence and the freedom to flourish. You can grab greater opportunities, have bigger dreams and extend a wider influence. You won’t have to feel like a fraud anymore. Objective 1: Understand the causes, symptoms and likely “victims” of the Impostor Syndrome. Objective 2: Learn a 3 step strategy to turn down the volume of the Impostor voice in your head. Objective 3: Explore how engaging in community can further diminish the Impostor syndrome Objective 4: Learn a methodical process of documenting successes and accomplishments so you can match confidence to competence.

Reverse Engineering a Bluetooth Lightbulb

Jesse Phelps

I recently made build lights for the company I work for and my home office. They integrate with TeamCity and indicate when a build is running and success/failure of all the tests. In this session, we will reverse engineer a bluetooth light bulb's protocol, learn how to have an Intel Edison communicate with the bulb, and by the end you too will know how to make your own build lights!

Please note that this talk will be highly technical. We will be discussing low level details of bluetooth communication, protocol analysis with Wireshark, sniffing bluetooth packets, etc.

Safety-enabled apps: using Boston crime data to make Pokemon Go a safer space

Raj Singh

Governments are increasingly making their data available freely in machine-readable format, which is great. A drawback, however, is that government data is organized and produced in a way that’s most useful to government operations, and it's not offered in an app-developer or data scientist friendly way. I talk about a project to harvest crime data daily from cities across the US and make it available for web apps and analytics as part of a larger effort to deliver ready-to-use open data sets on the cloud. The importance of adding safety data to apps is underscored by recent news that thieves have been "luring" Pokemon Go players into secluded areas. More broadly, I believe user-facing apps and back-end analytics will increasingly leverage 3rd party contextual data sources to refine and improve their capabilities.

So you really want to IoT your car?

Lwin & Min Maung

All vehicles made or imported to the US after 1997 are required to have an ODBII(On Board Diagnostics 2) connector exposed to users. Auto manufacturers have exposed data from vehicles this way. As technology progressed, auto manufacturers switched from mechnical feedback and controls to programmable Bus system.

So, in order for you to switch on the radio in the car, your mechnical or touch screen communicates via CAN Bus to achieve such a task. The same goes for turning on lights, horns, all the way to stopping (and if available self driving and parking vehicles with little or no assistance from users).

We will use IoT microcontroller to tap into CAN Bus and communicate with a Ford Explorer. We will then stream the data from the vehicle to Azure IoT Hub (service bus) and use stream analytics to push data to be viewed or process and then sent back to the vehicle.

Let me show you the future of data analysis from vehicles on the cloud and controlling vehicles from the cloud.

TDD vs. ATDD - What, Why, Which, When & Where?

Daniel Davis

Join in for a discussion around Test Driven Development (TDD) vs. Acceptance Test Driven Development (ATDD) and start to explore what the differences between them are. Through the use of high level abstracted concepts and then actual real code examples, you'll gather some insight into why each is used, see some of the advantages and disadvantages of both, and have a better understanding of which should be used when and where. By the end of the session you should be well along the path to TDD vs. ATDD enlightenment.

Main Session 12 - Friday 14:45

.NET Standard for Mere Mortals

Jeffrey Fritz

.NET Standard is a buzzword that Microsoft started using with the Build event in 2016. What is .NET Standard, and how does this framework effect how I'm building applications today? Come learn from one of the .NET team members about how .NET Standard makes writing code that works on any operating system with any project model a snap. You'll be able to build packages that work on your iPhone, web server, and docker containers when you finish this session.

98 And 3/4 Percent Guaranteed

Aaron Bedra

These days, having a build pipeline is common. We build pipelines for testing, packaging, containerizing, and all sorts of things that help us ensure that things won't fall to pieces when we press the big red deploy button. The problem is that we often forget to add security checks to our build process.

Join Aaron as he walks through the various tools and techniques we can add to our build. You will learn how to approach static analysis, dependency analysis, container analysis, and specialized security test suites in an automated fashion that will provide actionable feedback before it's too late.

A new storm is brewing: Spring Data Flow Server for Kubernetes

David Lucas

What do you get when you combine Spring Integrations, Data, and Kubernetes? You get an awesome data microservices tool that orchestrates long running streams and short lived tasks via JVM processes. This session will introduce the Spring Data Flow Server for Kubernetes, its awesome UI dashboard and how to create streams and tasks using the UI designer. It creates an environment where creating microservices is easier by providing a layer over top of Kubernetes and Docker. This demo will show how to setup and create some example streams and tasks.

Block__Element--Magic: CSS Modularity for the masses.

Chris DeMars

Have you ever faced a code base worked on by more than a dozen developers, blankly staring at thousands of lines of CSS, not knowing where to begin? A solution to this is B.E.M., or Block Element Modifier. B.E.M. is a CSS architecture. It's a methodology. It's a naming convention. It is based off of Object Oriented CSS. This talk will focus on basic B.E.M. CSS architecture and how it can be the solution towards modularity and writing clean, well maintained code within a large organization.

CSS is hard, we all know that. There has to be a more effective way to facilitate the modularity and flexibility of our code. When tasked with creating new components for a Web project and inheriting over 8,000 lines of code, where do you start? When you are told to keep it modular as possible, what solutions do you have at your disposal? During this time is when you break in to your bag-of-tricks with one of the many modular CSS architectures out there.

B.E.M. is a great method to use to keep code very flexible and modular from component to component, and page to page. The greatest thing about using B.E.M. is the reusability of the code and being able to maintain the code in small pieces opposed to a large blocks with excessive declarations and generic naming.

One of the greatest takeaways from using B.E.M. is the naming of styles. With B.E.M., you can be as specific as you want to describe exactly what that style is being applied too. Naming the style according to where it is being applied and what it is being used for helps other developers who inherit your code as well as your future self.

Functional Web Programming using Elm

Spencer Schneidenbach

How can you bring true functional programming into the web world AND transpile down to JavaScript so you can run it anywhere? No, it's not a pipe dream - it's Elm. Elm describes itself as a direct competitor to React, but it’s so much more – it’s a language as well as a set of tools for building scalable web applications in a concise, maintainable way.

In this session, Spencer will explore the Elm programming language and how its functional roots will help you create web apps that scale. Attendees will learn about creating Elm components using the model-update-view pattern, how to use commands to perform HTTP requests, and how subscriptions can help create event-based reactive applications.

Getting Started with Functional Programming in F#

Reid Evans

Functional programming is one of those things that's only good for math and statistics and stuff right??? Wrong! In this session we'll start from scratch and code a multi tier application using functional techniques. Attendees will see F#'s Type Providers which are used to statically type check our data access, and asynchronous data access to help the app scale. This is a live coding talk designed to show unique features of the F# language and how it is directly applicable to solving business problems.

Leadership is NOT Rocket Science!

Maureen Zappala

Great technical experts do not automatically make great leaders and managers. Unfortunately, far too many people are promoted into a leadership position simply because of technical competence and superiority. But that often sets them up for failure. They don’t have the right “soft” skills to lead well. The result is that everyone loses. The leader is discouraged. The crew is frustrated. Upper management is forced to accept lower morale and productivity. Leadership is not rocket science, but it’s just as complex because it involves people, not equations. And people are complex. In this session, Maureen will uncover a unique twist on leadership that can transform a technical expert into a managerial superstar. When leaders discover the “Five Fuel Sources”, which are the five key capacities that every employee needs to have filled, they will be able to lead with authority, passion and confidence. And, their teams will begin to explode with loyalty and camaraderie. You can become the leader that people will want to follow.

Machines Learning Human Biases: How Does It Happen? Can We Unteach Them?

Devney Hamilton

Machine learning techniques are based on the assumption that the future will resemble the past, with some allowance for uncertainty and yet-unobserved events. That assumption has held up well in machine learning applications like advertising, weather, and self driving cars. But what about applications that predict a person’s future actions and use that prediction to make a big decision about that person's life? What if we train our machine learning systems on data containing human biases we do not want to reinforce in the future? This talk first draws on the use of machine learning classification techniques in criminal sentencing to explore questions about how machine learning systems learn human biases. It will then look to promising work at MIT that demonstrates removing unwanted biases from Google’s word2vec, a vector space representing word meanings. Many machine learning techniques, including the classification used in the sentencing case, start with building a vector space to mathematically represent our observations of the past. If we can, as MIT researchers have, ‘de-bias’ our vector spaces, perhaps we can use machine learning to create the future we want.

Native iOS and Android Development with C# and Xamarin

James Montemagno

Xamarin helps C# developers become native iOS, Android, and Windows mobile app developers overnight. In this session, you'll learn how to leverage your existing .NET and C# skills to create iOS and Android mobile apps in Visual Studio with Xamarin. Besides letting you write your iOS and Android apps in C#, Xamarin lets you reuse existing .NET libraries and share your business logic across iOS, Android, and Windows apps. This session will also give you the tools to determine how much existing C# code can go mobile to iOS and Android, plus help you determine the architecture necessary to support maximum code sharing and reuse. You'll also receive some guidance and best practices for handling fragmentation across and within each device platform.

This session will cover the Xamarin platform and how to create native iOS, Android, and Windows apps in C#. Moreover, it will really focus on the code with several live coding adventures throughout the entire session. When you leave, you'll have the knowledge to create your first iOS & Android in C# with Xamarin in Visual Studio.

Unicorns May Poop Rainbows, But Must Still Obey Design Laws

Scott Showalter

A humorous spin on a serious, often unexciting topic: design laws and principles that govern not just UI design, but product strategy, UX design and information architecture alike. While researching some of the covered laws on your own (including Fitt's Law, Hick's Law, Miller's Law and several other important laws and principles) can be enough to make you dangerous, this session reveals examples of real danger in the wild and aims to bring you up to speed quickly on the biggest things to watch out for when designing or developing products. Examples show both "what to do" and "what not to do", with the ultimate goal of making you so good, it'd be dangerous for clients not to work with you, for companies not to hire you, and for team members not to rely on you for such intimate knowledge of "the law" so you at least know what you're doing if you ever decide to break them.

Upgrading A Robotic Exoskeleton - Iron Man Suit Mark 2

Scott Preston

Last year I brought version 1 of my Iron Man suit, The Mark 1. It was steel & aluminum weighing over 200lb. It was powered by an air compressor and 4 air cylinders. It even had an arc reactor light. But despite how awesome it was it was hard to walk, didn't have built-in biometric sensors and it required 3 people to assist me in and out of the suit. Needless to say it was time for an upgrade.

This years talk will feature a new suit, The Mark 2. It will be powered differently, will be more maneuverable, have biometric sensors, and be even more awesome. This suit will also be self-fitting, so I can walk into the suit and have it form around me. It will also feature a built in Raspberry PI 3 integrated with the sensors and controlling the suit actuators via JavaScript.

I will also bring a fully intact Mark 1 suit and give a brief synopsis of last year's talk before I talk about all the fun I had building the Mark 2.

What's in the (SQL) box?! Let's try and avoid some deadly sins.

Andrew Edwards

What happens when you want to update your system; where do you get the time for that amidst pushing out new features, fixing bugs, etc?

This session will explore what happened when my company decided to switch from MSSQL to PostgreSQL and move the 10,000 functions and stored procedures over. This talk isn't about the T-SQL or PostgreSQL languages, nor their nuances (although both may be mentioned in the talk), but rather how myself and my small team went about converting and updating this huge codebase.

Zero Defects and Less Pizza

Jeff Morgan

Is it really possible to have zero defects? The short answer is yes - if you are willing to change the way you collaborate, code, and test your application. Allowing defects in your software is an admission that it is okay to have a low quality product. This creates the mindset that believes quality is not important which is propagated throughout the code that is produced. Teams that have zero defects can deliver at a consistent pace and often more rapidly over the long term than teams that log their defects.

Defects cost much more than is obvious. Between documenting, inventorying, prioritizing, reproducing, fixing, retesting and all the meetings and pizzas to get to releasable software, it is a huge amount of waste that many organizations believe is unavoidable. And then there comes the "hardening phase"!

Join Cheezy as he brings more than a dozen years of hands-on experience coaching teams on agile engineering and team practices that drive quality higher. He will show what needs to change in your organization in order to have teams that truly focus on quality, deliver defect free software, and deliver more value to their customers. From source code management, to advanced development and testing techniques, to dev-ops and build pipelines, this talk is full of ideas and real life practices that you can take back to your work and utilize right away. By implementing these practices you can also help Cheezy with his lifetime goal of reducing the number of pizza parties (staying late at work) and increasing the number of beer parties (celebrations for a great release).

Main Session 13 - Friday 16:00

A History of F#: From Euclid to Type Providers

Rachel Reese

Have you ever wondered where your favorite feature came from? Was it influenced by a feature in another language? How are the different programming languages even related? I spent a couple months researching the history of some programming languages, and wanted to share that with you. In this talk, I cover the history of the ML family from approximately the dawn of time, eventually focusing on F# specifically.

Adding ES6 to Your Developer Toolbox

Jeff Strauss

Web developers constantly look for the latest and greatest ways to hone their craft, but changes come fast. From jQuery to Angular to Ember to React, CoffeeScript to TypeScript, it seems there is always something new. But ES6 is something different. With ES6 we are seeing the evolution of core JavaScript. It includes syntactic improvements and great new features never before seen in client-side code. Linters and transpilers for ES6 are readily available and easy to use. There is no need to wait; learn how to leverage the power of "the new JavaScript" in your applications today!

Bug Bounties and the OWASP Top 10: Messy Vulns and Real Lessons

Justice Cassel

Much of the existing application security & secure development curriculum show security issues in a vacuum, or in the simplest example setting. On the other hand, public bug bounty reports inherently show bugs in real world context. Sometimes that context is unbelievably trivial, other times it is intricate and pointedly specific to the vulnerable site. Both of these extremes provide important nuances that help developers and testers understand how to identify and remediate security issues.

This walking tour of common vulnerabilities, as well as more pragmatic “dirty” hacks, bridges the theory/practice divide with illustrative examples drawn from real-world bug bounty programs to help you see your code as attackers do. Finally, you’ll see some examples of how others remediate (often badly) when faced with serious, public facing vulnerabilities and get a better appreciation for how defense-in-depth buys you time to do things right.

Building An Audio Microscope: Exploring the Mircrocosm Inside a Snapshot of Sound

Krista Campbell

Every day we take pictures left and right with our phones and other devices. We don't think twice about our ability to capture a moment in time that we can later see whenever we like. Zooming in to look at details of a picture, reversing it, changing its color, contrast, etc. are all functions we take for granted. *Why isnt such functionality commonplace for audio as well?*

Sure, we can sample, loop, process, and transform audio samples, but *we can't zoom in to an instance of sound the same way *we can with an image because audio data is fundamentally different than video data.

The reason is that audio data is movement.

It captures the movement of the surface of a microphone, which can later move the surface of a speaker, creating sound waves which move the surface of our ears. For us to hear it, it has to move. It cannot be sitting still like a picture.

Even though audio data is drastically different than video data, if we manipulate it differently, knowing how our brain puts it all together, *it IS possible to take a microscopic snapshot of sound* which we can then stretch out, zoom into, and explore.

In this session we will use a graphical audio processing programming environment to show you how to build a very special kind of sampler: an audio microscope. We will use that microscope to explore the micrcosm of sound inside a single note.

Continuous Delivery Anti-patterns

Jeff Morgan

It wasn't long ago that everybody thought "Continuous Deliver" was only for tech companies. It is amazing how much things have changed over the past two years. More and more enterprises are exploring what it will take to achieve Continuous Delivery and Continuous Deployment for their core customer facing applications. The challenge for these companies is that many of the practices and processes they have put in place over the years stand in the way of achieving this goal. From rigorous engineering practices, to a very different view on product ownership, to organizational restructuring, to streamline the software development pipeline, Continuous Delivery .i.e. the ability to reliably release software at any time requires a significant shift in the mindset.

Join Cheezy as points out several Continuous Deliver anti-patterns and how to avoid or eliminate these patterns within your organization in order to align your development value stream, operations, release management, and product owners. If you want to know what it takes to achieve Continuous Deliver then this is one talk that you will not want to miss.

Effective Deployment of Service Oriented Architectures with Consul & Terraform

James Nugent

As organizations adopt service-oriented architectures, the boundaries between the development teams responsible for individual services and operations teams responsible for shared infrastructure start to blur.

In this talk we'll look at how layered Terraform configuration can enable service teams to deploy autonomously, making use of shared resources provisioned by other teams.

We'll also look at how each service instance can self-bootstrap into Consul, integrating into an nginx-based load balancer and API gateway.

Fiery Reactions: A Mobile ReactJS Ecosystem with Firebase

Nick Stewart

Facebook has given us React Native, a javascript platform for building native applications with native look and feel. Google has given us Firebase, a backend system that provides authentication, data storage, hosting, analytics, notifications, and more. And, of course, Apple gave us the iPhone.

In this talk, we'll use these technologies from these tech giants together to rapidly build a robust mobile application and deploy it to a device. We'll see how much faster it is to avoid building out a backend entirely, and we'll see the advantages of a multi-platform tool over native code in mobile development.

Goldilocks Prototyping: Finding the fidelity that’s just right

Brian Greene

Prototypes can serve a number of valuable purposes during the process of creating a website or mobile app. Finding the right fidelity takes into account a number of different considerations that can quickly evolve over the course of a project. A prototype can communicate a number of different design considerations to a wide variety of audiences.

In this session, you’ll learn how to choose the right fidelity of prototype to help you gain alignment and garner valuable feedback. A number of different prototyping tools and techniques will be covered so you are better equipped to choose the right tool, at the right fidelity, at the right time for your next project.

How to Build the Virtual DOM

Tommy Graves

For years front-end developers have bemoaned having to use the DOM; it's slow, inconsistent, and difficult to use. While libraries of the past provided better ways to interface with the DOM, recent libraries like React and Vue have instead implemented an entirely new DOM: the virtual DOM. Implementations of the virtual DOM allow developers to create and maintain UI components using entirely declarative interfaces that completely abstract away the real DOM. In this session, attendees will unlock the power of the virtual DOM through a step-by-step tutorial on the creation of a virtual DOM implementation in under 200 lines of code. In doing so, they will gain a precise understanding of the advantages the virtual DOM provides to speed, maintainability, and testability.

Leading Hackers and Hacking Leaders

Jonathan Popham

A master stops punches before they are thrown.

Can you conquer conflict before it conquers you?

It doesn't matter if you are a ranger at Burning Man, if you are running a hackerspace, if you are working for a startup, or if you need someone to drop their fists before they drop you -- you will face social challenges and how you answer those challenges will determine your future comfort or maybe even your present safety.

Thriving in any situation where a community is involved requires certain skills. This talk covers skills that will help you navigate personal and professional relationships. How you manage the community of people around you, whether they be volunteers, employees, peers, bosses, spouses, or family, will determine your success and happiness.

This talk will cover topics such as building social capital, trust, body language, positive conflict resolution, and community relationships.

Multi-region Delivery Netflix style

Andrew Glover

Netflix rapidly deploys services across multiple AWS accounts and regions over 4,000 times a day. We’ve learned myriad lessons in regards to reliability and efficiency; what’s more, we’ve built sophisticated tooling which has facilitated our expanding global footprint. In this session, you’ll learn about how Netflix confidently delivers services on a global scale and how you can do the same with a number of best practices combined with freely available open source software.

Four things attendees will learn: * Best practices for multi-region software delivery, including taking advantage of metrics for global deployment windows * Coordinating deployment pipelines * Fault tolerance strategies * A peek into the future of Delivery at Netflix

Rules as an Architectural Pattern For Development

Steve Swing

Rule-based architectural patterns produce solutions that are highly performant, composable, loosely coupled, flexible, and can be thoroughly tested. A rule-based approach to application logic integrates well with other architectural patterns such as big data analysis output, streaming data pipelines for real-time processing, or message-oriented processing. Rule-based architectures complement microservices architectures too. Large unwieldy opaque rule engines of the past are not required. Nor are rule-specialized languages and tooling. Developers don't need specialized training when rules are implemented with the language of the project, available collections API, and business domain language. This session describes an internal architectural pattern to implement complex business logic to deliver quality software that endures. The concepts can be implemented in any language.

Sunlight & Air: 10 Lessons for Growing Junior Developers

Erika Carlson

How do we become software developers with the knowledge and skill to do our work well? Training junior developers is an essential piece of building a successful team, but it's a complex and challenging task. The lessons shared in this talk are drawn from experiences training, coaching, and mentoring over 50 junior developers in the past 2 years, and successes, failures, challenges, and rewards will be discussed. Attendees will learn strategies for identifying potential, assessing team fit, guiding technical growth, and coaching new developers in soft skills and interpersonal development.

Your DBA Says "It Depends". Now What?

Maggie Pint

Do you feel like most interactions with your DBA/IT team result in statement "It Depends"? Do you wonder what metrics your DBA is evaluating to make assertions about the system? Do you want to know key tips for performance tuning applications? A former DBA turned full-stack web developer will take you through some of the metrics and tools that DBAs use to evaluate performance so that you can more easily communicate with your DBA or troubleshoot your SQL server personally.