Strange Loop Program, Sept 21-Sept 22, 2023

Welcome

9/21 9:30 am-9:45 am, Stifel Theatre

Alex Miller

How to build a meaningful career

9/21 9:45 am-10:30 am, Stifel Theatre

The average person will spendÂ *a third*Â of their life at work. Let's do our best to make sure our day to day work aligns with what increases our happiness as opposed to decreasing it. How can you focus on what lifts your spirit while acknowledging the complexities of both people, software, and corporations? Humans are squishy. Join Taylor and Scott as they give you tips on how to create a meaningful career that brings you joy and uplifts those humans around you.

Taylor Poindexter *@engineering_bae - Spotify*

Taylor Poindexter is the co-founder of Black Code Collective, an organization that strives to create a safe space for software engineers to thrive, and she is also an Engineering Manager at Spotify. In her downtime you can find her adventuring, tweeting about tech, creating whiskey videos for Instagram.

Scott Hanselman *@shanselman - Microsoft*

Scott has been a developer for 30 years and has been blogging at https://hanselman.com for 20 years! He works in Open Source on .NET and the Azure Cloud for Microsoft out of his home office in Portland, Oregon. Scott has been podcasting for over 850 episodes of http://hanselminutes.com over 17 years and over 700 episodes of http://www.azurefriday.com. He's written a number of technical books and spoken in person to over one million developers worldwide! He's also on TikTok, which was very likely a huge mistake.

Playing with Engineering

9/21 10:45 am-11:30 am, Stifel Theatre

AnnMarie Thomas *- Playful Learning Lab at Univ of St. Thomas*

AnnMarie is a Professor in the School of Engineering and the Opus College of Business at the University of St. Thomas. She is the founder and director of the Playful Learning Lab, which explores ways to encourage children, of all ages, to embrace playful learning. She and her students created Squishy Circuits. She is the author of Making Makers: Kids, Tools, and the Future of Innovation. She is the executive director of OK Go Sandbox, which she cofounded with the band OK Go.

AnnMarie earned Ph.D. and M.S. degrees in Mechanical Engineering from Caltech, and an S.B. in Ocean Engineering (with a minor in Music) from MIT. Additionally, she completed a professional certificate in Sustainable Design from the Minneapolis College of Art and Design.

AnnMarie co-founded and co-directs the University of St. Thomas Center for Engineering Education (CEE), which offers engineering courses for P- 12 educators and conducts research on engineering at the pre-collegiate level. AnnMarie's teaching covers classes on topics such as Engineering Graphics and Design, Machine Design, Dynamics (with Circus Lab), Toy Design, Technology Prototyping (for Business majors), Environmental Sustainability/Innovation, and courses on Engineering Education for PK-12 teachers.

She served as the Founding Executive Director of the Maker Education Initiative, where her team established the national Maker Corps program and laid the groundwork for this nonprofit.Â

AnnMarie lives in St. Paul, Minnesota, with her husband and two daughters. She is an amateur aerialist, specializing in static and flying trapeze, as well as juggling.

Thursday Lunch

9/21 11:45 am-1:15 pm, Union Station

Lunch buffet will be provided at Union Station in the Midway

The Attacker Has Expensive Radio Equipment, But Your Android Phone Is Resilient

9/21 12:00 pm-12:40 pm, Union Station Grand F

Some of the most dangerous cellular attacks targeting activists, journalists, and the general public have relied on GSM, the 2G cellular protocol. The 2G attack surface is so large that it leaves users vulnerable to everything from real time location tracking attacks, communication interception attacks, and receiving malware delivered over-the-air by anyone nearby.

A few years ago Android set out to work towards disabling all 2G protocols to protect users from this massive attack surface. This talk is (1) part technical deepdive into what it's like to disable a major communications protocol at the radio hardware level in a fragmented ecosystem, (2) an overview of cellular fundamentals (e.g. what your phone is actually doing when it "connects" to a cell tower), and (3) part history lesson about the extensivity of phone radio surveillance and its use by cyber mercenaries all over the world.

Yomna Nasser *@rival_elf - Software Engineer @ Android*

Yomna is a member of the Android Connectivity Security team, and previously worked at EFF on certbot (a popular "Let's Encrypt" client). She is interested in the mysteries of phone systems, both modern and historical.

Why Programming Languages Matter

9/21 12:05 pm-12:45 pm, Union Station Grand DE (PWLConf)

I've spent most of my professional life working on programming languages: studying them, designing them, defining their semantics, comparing features, and even implementing them (mostly poorly). Â Why would an otherwise-sane person do such a thing? Â Because programmingÂ languagesÂ areÂ the ultimate meta-tool: they give you the ability to create from pure thought. Â And yet: aÂ programming language is not just a means for programmers toÂ communicate withÂ computers-it is also a means for programmers to communicate withÂ programmers. Â That is: a good programming language isÂ aÂ social, as well as aÂ technical, enabler.

This talk will take a bumpy ride through the challenges andÂ opportunitiesÂ associated with the six languages that I've played a part in developing over the past forty-five years-most of which you won't have heard of. Â Why do new languages face an uphill struggle for acceptance? Â What areÂ the challenges facing the next generation of languages and language designers? Â WhyÂ shouldÂ you care about any of this? Â BecauseÂ those who do not know the past are doomed to repeat it.

Andrew Black *- gh:apblack - Emeritus Professor, Portland State University*

Andrew Black is an Emeritus Professor of Computer Science at Portland State University. Â His first programmingÂ languageÂ was Algol 60, an accident for which he will always beÂ grateful. Â As a graduate student, he at first refused a request to teach Fortran to undergraduates, on the grounds that it was unfair to subject students to such a language; he relented when his supervisor (Tony Hoare) pointed out that if the students were going to learn Fortran from someone, it would beÂ better if theyÂ learnedÂ it from a teacher who understood its many deficiencies. Â Later, butÂ afterÂ his involvement with the Emerald programming language, he learned, taught and contributed to Smalltalk, an encounter that permanently re-arranged his brain cells.

As well as teaching at four universities,Â Professor Black has worked in the laboratories of four major companies (two of which still exist). Â He has published over 100 research papers, and been involved in the design, development and implementation of six programming languages. For the past ten years he has been the BDFL of the Grace Educational Programming Language, and the major committer to theÂ minigraceÂ implementation.

A JVM threading model for the containerized times

9/21 12:10 pm-12:50 pm, Union Station Grand ABC

The threading model of JVM applications has become a common source of instability and inefficiency in containerized environments. In a company like Nubank, one of the largest fintech, with a microservices architecture comprising over 1500 services, manually tuning the ideal number of threads becomes both daunting and risky.

This presentation introduces a novel approach that addresses these issues by implementing a dynamic control loop and fine-grained load-shedding mechanism. The solution continuously adjusts the number of threads during application execution, utilizing real-time signals such as CPU throttling, CPU usage, memory usage, and runtime pressure. Meanwhile, based on a configurable maximum queuing delay, our load-shedding mechanism ensures that the application remains functional under stress by only rejecting work exceeding its capacity.

This comprehensive solution has significantly improved the stability and performance of our applications while reducing associated costs.

Luiz Hespanha *@luiz_hespanha - Senior Staff Software Engineer @ Nubank*

Luiz Hespanha is a Systems Performance engineer at Nubank, the most influential Latin America fintech. Working in the area for more *than 20 years, he's been in microservices since before they were called that and has also built useful large-scale systems using RESTful principles and messaging-centric architectures.*

Flavio Brasil *@fbrasisil - Principal Engineer*

Flavio is a seasoned engineer with a specialization in high-performance solutions for functional programming on the JVM. Throughout his career, he has navigated a wide range of technologies, spanning from intricate JIT compilers to sophisticated libraries for functional programming. Driven by his passion for open source, Flavio has contributed to numerous projects such as Quill, a prominent library for seamless database access in Scala, and Finagle, the robust library at the heart of Twitter's scalable platform.

Building Strong and Sustainable Open Source Projects: Lessons from an Epic Road Trip

9/21 12:40 pm-1:20 pm, Union Station Regency AB

This talk draws parallels between an epic road trip and the process of building solid and sustainable open-source communities. Just like the road trip filled with surprises, new friends, and life lessons, building an open-source community can be an unpredictable and rewarding adventure. Key lessons from the road trip experience can be applied to open-source community building, such as having a clear purpose and vision, embracing openness and diversity, being open to surprises, learning from failures, and celebrating successes. The talk further provides practical dos and don'ts for building successful open-source projects. By keeping these lessons in mind, open-source communities can thrive and evolve, fostering innovation and collaboration.

Lawrence Gray *@LarryGray - Kungfu.ai - Director of Engineering*

Dr. Lawrence Gray is an accomplished Machine Learning expert with expertise in Accounting and Geospatial Intelligence. He currently serves as the Director of Engineering at KUNGFU.AI and is a Program Director and instructor of Data Analytics and Data-Driven Decision-Making at the Georgetown University School of Continuing Studies. Dr. Gray earned his Ph.D. in Cellular and Molecular Physiology from Johns Hopkins University, School of Medicine. In addition to his work in machine learning, he dedicates his time to open science initiatives by serving on the board of directors at NumFocus and serves as a core contributor and maintainer for the open-source software package, Yellowbrick. Dr. Gray's research interests focus on helping others cultivate social capital within the data science community.

Without Open Data, there is no Ethical Machine Learning

9/21 12:45 pm-1:25 pm, Union Station Regency C

Machine learning has experienced an explosion of popularity in recent years, and even more so in recent months. In response, companies including but not limited to Reddit, Twitter, and Snapchat have all closed off, restricted access, or made headlines for how "open" they are with data to train large generative models.

Managing large open datasets is often a thankless act - but it's one that we must invest in if we truly want to prioritize a future that includes ethical machine learning. In this talk, we'll explore the role open data plays in machine learning, why it's concerning to see organizations close access to their API, the cost/benefits institutions and individuals run when they open their data, and - most importantly - we'll walk through different resources, projects, and opportunities for individuals to direct us to a better future at this current juncture in open data and machine learning.

Erin Mikail Staples *@erinmikail - developer advocate, community human, tech educator, comedian*

Erin Mikail Staples is a very online individual passionate about facilitating better connections online and off. She's forever thinking about how we can communicate, educate and elevate others through collaborative experiences.

Currently, Erin is a Senior Developer Community Advocate at Label Studio. At Label Studio - she empowers the open source community through education and advocacy efforts. Outside of her day job, Erin is a comedian, graduate technical advisor, content creator, triathlete, avid reader, and dog parent.

Most importantly, she believes in the power of being unabashedly "into things" and works to help friends, strangers, colleagues, community builders, students, and whoever else might cross her path find their thing.

Swift as C++ Successor in FoundationDB

9/21 1:40 pm-2:20 pm, Union Station Grand ABC

Programming languages often prioritize either performance or ergonomics. Swift offers a unique modern type-safe low-ceremony approach taking the best of both worlds that scales from mobile apps to high-performance systems where previously memory-unsafe languages would be used. It also interoperates seamlessly with C and C++.

In this talk, we show how we successfully adopted Swift in FoundationDB, a distributed database struggling to modernize its C++ codebase. Swift's interoperability features allowed the team to incrementally move single functions, or entire types, to Swift. This transition was done without generating any bindings and while preserving the existing semantics.

FoundationDB uses a custom actor runtime, which enables reproducible simulation testing. Again, Swift's flexible concurrency and distributed actor model enabled an incremental side-by-side approach by having Swift concurrency seamlessly execute on the existing scheduling infrastructure.

Konrad Malawski *@ktosopl - Swift Team, Apple*

Konrad works in the Swift team at Apple, where he focuses on foundational server-side Swift libraries and concurrency features of the language. He was part of the design and implementation of Swift's structured concurrency and actor model, as well as the distributed actors language feature and cluster library. He also maintains server-side observability libraries for logging, metrics and distributed tracing.

Previously, he worked on Akka at Lightbend/Typesafe, where he maintained core pieces of the ecosystem, including the clustering, event sourcing, streaming and HTTP libraries. He also made significant contributions to the reactive-streams specification and TCK which later became the foundation of the JDK's java.util.concurrent.Flow types.

Programming Distributed Systems

9/21 1:40 pm-2:20 pm, Union Station Grand DE (PWLConf)

Our interconnected world is increasingly reliant on distributed systems of unprecedented scale, serving applications which must share state across the globe. And, despite decades of research, we're still not sure how to program them!Â In this talk, I'll show how to use ideas from programming languages to make programming at scale easier, without sacrificing performance, correctness, or expressive power in the process.Â We'll see how slight tweaks to modern imperative programming languages can provably eliminate common errors due to replica consistency or concurrency---with little to no programmer effort.Â We'll see how new language designs can unlock new systems designs, yielding both more comprehensible protocols and better performance.Â And we'll conclude by imagining together the role that a new cloud-centric programming language could play in the next generation of distributed programs.

Mae Milano *@mbpmilano - gh:mpmilano - Incoming Assistant Professor, Princeton University*

Mae Milano is an incoming assistant professor at Princeton University and a postdoctoral scholar at UC Berkeley working at the intersection of Programming Languages, Distributed Systems, and Databases.Â Her work has appeared at top-tier academic venues and has attracted the attention of the Swift language team. She is a recipient of the NDSEG Fellowship, has won several awards for her writing and service, and is a founding member of the Computing Connections Fellowship's selection committee (https://computingconnections.org/).

Lessons from building GitHub code search

9/21 1:40 pm-2:20 pm, Union Station Grand F

In this talk, I'll share some lessons we learned building a high-performance code search engine, designed to meet GitHub's large scale. GitHub code search is the world's largest publicly available code search engine, with more than 60 million repositories and over 160 TB of content indexed. To build it, we had to turn the unique content-addressable nature of Git repositories to our advantage. I'll cover the key strategies we used, including using deduplication and repository similarity to reduce indexing workload, full index compaction to remove deleted documents, multiple levels of sharding, and load balancing. Come discover how we turned code search from a frustrating experience to a powerful feature for our users.

Luke Francl *@lof - GitHub*

Luke Francl works on code search at GitHub. He's excited to build software that makes developers happier and more productive. Prior to joining GitHub, he worked at a search-as-a-service startup, as a freelance developer, and spent a lot of time doing XML sit-ups in the Java world. He lives in San Francisco with his family.

Turning GraalVM from Research to Product

9/21 1:40 pm-2:20 pm, Union Station Regency AB

GraalVM is a new runtime system that provides instant startup and low footprint for Java applications via ahead-of-time compilation. It also has polyglot capabilities that allows to execute and interoperate with many languages including JavaScript, Python, Ruby, WebAssembly, and native languages. This session discusses lessons learned while turning the vague research ideas behind GraalVM into a released product over the timespan of more than a decade, how we experienced our entry to the developer community, and also major technical questions where we now think differently.

Thomas Wuerthinger @thomaswue - Vice President of GraalVM Research & Development

Thomas Wuerthinger is a Vice President at Oracle Labs and the GraalVM founder and project lead. Previously, he worked on the Crankshaft optimizing compiler of V8 at Google, and the Maxine research virtual machine at Sun Microsystems. He received a PhD degree from JKU Linz for his research about dynamic code evolution.

Diffusion as Art Therapy

9/21 1:40 pm-2:20 pm, Union Station Regency C

The capacity to imagine and visualize plays a crucial role in human psychology. Fantasies, dreams, and visions use image as their core language. Imagery plays a fundamental role in psychological healing, via expressive arts, sand play, and other therapeutic modalities.

-In this talk we'll look at how diffusion models like Stable Diffusion and Midjourney can create new possibilities for therapeutic work with trauma and the unconscious.

-We'll start with a guided scientific tour of the nervous system. Then we'll zoom in to look at the role imagery can play in therapeutic work specifically for working with a "freezy" nervous system, when other pathways like language and sensation are offline. Finally we'll conclude with a case study of using diffusion models in therapeutic practice. ",AI/Machine Learning,Michael,Nagle,"Hi!

-I am a researcher and product designer and educator. In my 20s I ran an alternative summer camp and homeschooling center called Camp Kaleidoscope. In my 30s I was a research scientist for the Communications Design Group, working on the precursor to Dynamicland, and later led research at Coda.

-I'm newly passionate about diffusion models, and deeply interested in how they can create novel forms of creative and expressive interaction both for kids and adults!

Michael Nagle @nagle5000 - co-founder at Moebius Labs

Hi!

-I am a researcher and product designer and educator. In my 20s I ran an alternative summer camp and homeschooling center called Camp Kaleidoscope. In my 30s I was a research scientist for the Communications Design Group, working on the precursor to Dynamicland, and later led research at Coda.

-I'm newly passionate about diffusion models, and deeply interested in how they can create novel forms of creative and expressive interaction both for kids and adults!

Is my Large Language Model a Strange Loop?

9/21 1:40 pm-2:20 pm, Union Station Regency C

Large Language Models aren't just the new hotness - they also offer fascinating glimpses of an alternative approach to software which isÂ evolvedÂ instead of written.Â We'll explore emergent representations and composition of concepts from the inside outwards; and then turn around and consider some emergent behaviors and feedback loops from the outside of the training process.Â Whether you're curious about superposition, induction, self-evaluation or self-correction... by the end of the talk you'll have more questions than you started with!

***Zac Hatfield-Dodds** - Assurance Team Lead @ Anthropic*

Zac (zhd.dev) is an Australian software engineer and researcher working in San Francisco, on AI safety research by day and on property-based testing by night.Â He co-maintains open-source projects such as Hypothesis, Trio, and Pytest; has been elected a Fellow of the Python Software Foundation; and swears that he'll finish writing his dissertation someday. If he's not at a computer, Zac is probably some combination of camping in a national park, reading a book or three, and munching on dark chocolate.

Oatmeal is Cheap: A Fundamental Theorem for Procedural Generators

9/21 2:30 pm-3:10 pm, Union Station Grand ABC

By studying the Kolmogorov complexity of procedural generators, we end up with a fundamental theorem for programs that can generate things. There's a deep link between the knowledge we encode in a generator and the complexity of what it can output. In the end, this result helps us understand the pragmatic decisions designers have to take when developing games like Minecraft, Borderlands or Tea Garden. The talk is illustrated with interactive examples that the audience can play with.

YounÃ¨s Rabii *@pyrofoux - Game Designer + Generative AI Researcher*

YounÃ¨s Rabii is a PhD researcher in game AI and the award-winning designer of games such as SUPER IS HOT, Tea Garden and Neurocracy. Their work has been previously exhibited in the French embassies and international festivals. Their current research is focused on building AI techniques that can help designers create new forms of play and storytelling. YounÃ¨s' work has two main goals: (1) resist the systemic oppression within the AI and game industries; (2) create games that can surprise players and designers alike.

Concatenative programming and stack-based languages

9/21 2:30 pm-3:10 pm, Union Station Grand DE (PWLConf)

In this talk we'll exploreÂ *stack-based*Â programming languages, in which your program operates directly (and only!) on a stack of values. It might seem daunting at first to program without variable names, but the simplicity of stack-based languages makes them interesting to reason about mathematically, and also fun to tinker with! We'll look at how stack-based languages areÂ *concatenative*, letting you break apart your program into arbitrary pieces without affecting its meaning. We'll compare them with combinatory logic, and see how small we can make our language while still being Turing-complete. And we'll show how they make good low-level (but still readable!) assembly languages, by examining a Uxn program and running it on a variety of interesting hardware.

References:

- Henry Baker. "Linear logic and permutation stacks - The Forth shall be first". SIGARCH Computer Architecture News 22:1, March 1994.
- Jeremy Gibbons. "Continuation-Passing Style, Defunctionalization, Accumulations, and Associativity". The Art, Science, and Engineering of Programming, 2022.
- Brent Kirby. "The theory of concatenative combinators".
- Slava Pestov, Daniel Ehrenberg, Joe Groff. "Factor: A Dynamic Stack-based Programming Language". ACM SIGPLAN Notices 45:12. December 2010.
- Smullyan, Raymond. To Mock a Mockingbird. Alfred A. Knopf: New York, 1985.
- Manfred von Thun. "The mathematical foundations of Joy".

Douglas Creager *@dcreager - Walland Heavy Research*

I've been thinking about, and using, programming languages for a very long time. By day, I manage GitHub's Semantic Code team, figuring out how to understand and analyze ~every programming language under the sun. By night, I balance family life with hacking on my own language, like any good PL enthusiast would!

IPVM: Seamless Services for an Open World

9/21 2:30 pm-3:10 pm, Union Station Grand F

The advent of TCP/IP and the web produced an explosion of innovation by radically lowering the barrier to entry to networking. Thanks to recent technical and social innovation, we now have the building blocks for the next generation of open services: location-free verifiable data and computation. This talk presents UCAN Invocation and the Interplanetary VM (IPVM). IPVM code runs anywhere (even offline), respects data privacy, and services interoperate seamlessly without pre-negotiation.

Verifiable computation opens the door to content addressed invocations, results, and workflows. This greatly lowers the complexity over traditional architectures (e.g. LAMP). Not only is this easier to reason about, it also (paradoxically) has better than linear scaling: the more it's used, the faster it is! As anyone can participate, it doesn't depend on the handful of data centres provided by large orgs: anyone can provide capacity (including in underserved regions like Africa and South America).

Brooklyn Zelenka *@expede - CTO @ Fission Codes*

Brooklyn the co-founder and CTO of Fission, where her team is creating next-gen tools that empower developers to easily build innovative, decentralized applications. She is the author of numerous libraries including Witchcraft, and founded the Vancouver Functional Programming Meetup. Her belief in open standards has lead to work spanning distributed VMs (Ethereum, IPVM), authorization (UCAN), data privacy (Webnative FS & Rhizome DB), varsig, multiformats, and others.

Babashka: a meta-circular Clojure interpreter for the command line

9/21 2:30 pm-3:10 pm, Union Station Regency AB

Babashka is a Clojure interpreter for cross platform scripting. It is available as a single binary that starts instantly. It makes Clojure a viable replacement for writing bash scripts. Babashka comes with a handful of libraries out of the box (JSON, command line parsing, etc.) and supports loading libraries from the Clojure ecosystem. The interpreter is written in a meta-circular approach, akin to Structure and Interpretation of Computer Programs. It is compiled to a single binary using GraalVM native-image which is the reason it starts fast, but also uses less memory than a JVM, Clojure's original runtime. As such, babashka brings together many exciting technologies to broaden the reach of Clojure even more. This talk explores the high level use cases of babashka, its impact on the Clojure community, its history, technical implementation details and the author's approach to open source development.

Michiel Borkent *@borkdude - Clojure OSS developer*

Michiel Borkent, also known as @borkdude on the web, is an open source software developer who loves Clojure. He is the author of clj-kondo, babashka, SCI, nbb and other tools that you might have heard of. When he is not coding, he likes to go for a walk outdoors or eat vegetables!

Risks and Opportunities of AI in Incident Management

9/21 2:30 pm-3:10 pm, Union Station Regency C

Large Language Models provide a powerful "sidekick" in resolving incidents. Our talk opens by exploring what LLMs can do when things go wrong: from parsing your codebase to debug, to ad-hoc testing scripts, to brainstorming solutions with engineers. These features ought to be considered by orgs of all sizes. Not only do they reduce the time sink of incidents, they open up that time for feature development, compounding advantages.

LLMs aren't perfect, and their common failure modes are critical when applied to incident response. We'll cover some of these failures and how they'd look in the context of incident response, including hallucination, misprioritization, and black boxing.

This isn't the end of the world. Orgs just need to weigh these risks versus the speed and convenience of LLM incident response. To mitigate the risk, orgs need to invest in people. We'll show how the resilience, adaptability, and knowledge of your incident response teams can compensate for the risks of LLMs.

Emily Arnott *@emilyarnott8 - Community Manager at Blameless*

Emily is the community manager at Blameless, an incident workflow solution. She loves seeking out the cutting edge in how companies stay online.

Thursday Snack

9/21 3:10 pm-3:45 pm, Union Station

Snack will be provided at Union Station in the MIdway

Software & The Game of Go

9/21 3:45 pm-4:25 pm, Union Station Grand ABC

Familiarity with the ancient game of Go (Weiqi/Baduk) has noticeably increased outside of Korean, Japan and China thanks to the historic match between Lee Sedol and DeepMind's AlphaGo in 2016. However, this talk isn't about contemporary machine learning techniques (though we may touch upon those topics), rather we explore the congruences between the act of writing software and learning how to play this beautiful game. As Yu Ji writes in the preface to the Xuanxuan Qijing published in mid-1300s (you read that right):

"There is nothing it [Go] does not encompass in the rise and fall of the Way of the World, and in the waxing and waning of Man's affairs."

David Nolen *@swannodette - Vouch*

David Nolen maintains ClojureScript. He works at Vouch. He lives in New York City.

Inside the Wizard Research Engine

9/21 3:45 pm-4:25 pm, Union Station Grand DE (PWLConf)

WebAssembly offers the hope of a fast, portable low-level target that competes with native code performance, but without many of the drawbacks of native code. Born in the Web, Wasm now runs in many contexts, such as embedded systems, cloud and edge deployments, and blockchains. With new features coming to Wasm that add garbage-collected data, first-class functions, and stack switching, it offers the tantalizing opportunity to retarget many programming languages for greater safety and portability. Yet these new features offer new challenges, and some of the same challenges facing managed runtimes. In this talk, I will outline my work on a new Wasm engine, Wizard, designed for instrumentation, dynamic analysis, and experimentation in VM design. In particular, I'll cover its innovative in-place interpreter and the instrumentation capabilities that unlocks.

Ben L. Titzer *@TitzerBL - gh:titzer - Principal Researcher, Carnegie Mellon University*

Ben L.Â TitzerÂ is a Principal Researcher at Carnegie Mellon University. A former member of the V8 team at Google, he co-founded the WebAssembly project, led the team that built the implementation in V8, and led the initial design of V8's TurboFan optimizing compiler. Prior to that he was a researcher at Sun Labs and contributed to the Maxine Java-in-Java VM. Currently, he is working on a new Wasm research engine and several WebAssembly-related research projects. He is the designer and main implementer of the Virgil programming language.

Building Koi Pond: Simulating Millions of Slack Clients

9/21 3:45 pm-4:25 pm, Union Station Grand F

During the summer of 2020, Slack's performance infrastructure team was tasked with building a load testing solution that could simulate hundreds of thousands of users in a single team ahead of a big customer launch. What followed was two months of prototyping: figuring out the right balance between simulating high-fidelity usage, efficient usage computing resources, and speed of execution. Join Maude for a roller coaster ride of a story and a thrilling live demo of what our load testing systems can do!

Maude Lemaire *@QcMaude - Sr. Staff Software Engineer @ Slack*

Maude is a Senior Staff Engineer at Slack where she is a founder and technical lead for the backend performance infrastructure team. She's responsible for large-scale load test tooling, performance regression monitoring, and successfully onboarding the world's largest companies to Slack. Over the past six years, she's helped the product scale from just 60,000 users per team to over 2 million. When she doesn't have her nose in a flamegraph, you can find Maude building strong, empathetic engineering cultures.

In October 2020, Maude published "Refactoring at Scale" with O'Reilly Media, a blueprint for how technical leaders can successfully navigate large, complex refactors.

Comedy Writing With Small Generative Models

9/21 3:45 pm-4:25 pm, Union Station Regency AB

A technical and practical review of generative text models and their use in comedy projects. Starting with a writer's account of learning to code by reverse engineering a phone's predictive text, I'll track the progression of sequential text models from ngrams to LSTMs to transformers and explore how the properties of these models translate into the feel of their output. Finally I'll ask: Is there still a place for small language models in a time of ever larger ones? And I'll answer: yes!

Jamie Brew *@jamieabrew - Founder, Botnik Studios*

Jamie Brew is a founder of the A.I. comedy company Botnik, widely known as one of the "Big Four" tech giants. Previously, he was a writer and editor at The Onion and ClickHole.

Designing Fully Rationalized Commercial Buildings with Generative Software

9/21 3:45 pm-4:25 pm, Union Station Regency C

Designing a building is a complicated task with dozens, sometimes hundreds of participants. Architects explore the form, mechanical and structural engineers design the building systems, interior architects create spatial test fits, building code experts ensure everything adheres to code, and still others try to predict how the building will function for its end users (is the space functional? enjoyable?). In this talk I'll share how we've built generative software that designs fully rationalized commercial buildings, and measures how they will perform with respect to important metrics like flexibility, carbon footprint, end-user experience, and cost.

Jen Carlile *@jencarlile - CTO, Outer Labs*

The intersection of technology and the built environment is a place that Jen has considered home for over a decade. Currently the CTO of Outer Labs, a company she co-founded in 2018 to help real estate owners scale through better technology, Jen brings her extensive background as a software engineer to assist in the development of custom technology solutions targeting the AEC space (Architecture, Engineering, & Construction). Prior to her work at Outer Labs, Jen co-founded Flux Data, a cloud platform spin-off from Google X intended to lower the cost and improve the quality of buildings across the globe.

Supporting Data Journalism through Compilers for Visual Inputs

9/21 4:35 pm-5:15 pm, Union Station Grand ABC

Traditionally, we view compilers as translational machines between two symbolic representations-a source language and a target language. This model excels for textual inputs, but what if we want to transform a visual input, such as a data graphic or thematic map, into a program? This question looms large in today's newsrooms, where data journalists often need to turn story sketches, SVG graphics, Figma files, and more into interactive programs.

In this talk, we'll discuss our research on compilers for visual inputs and explore their application in the newsroom. We'll highlight two projects-reviz and cartokit-that allow users to (1) reverse engineer data visualizations from the DOM into parameterized programs and (2) develop interactive thematic maps through direct manipulation. Along the way, we'll discuss the unique compilation challenges that come with treating visual inputs as source programs. We'll also share lessons learned from our time embedding with data journalists.

Parker Ziegler *@parker_ziegler - Ph.D. Student, University of California, Berkeley*

Parker Ziegler is a Ph.D. student in Computer Science at the University of California, Berkeley, where he is advised by Professor Sarah E. Chasins. His research focuses on developing novel programming languages and programming interfaces for data journalism, with an emphasis on geospatial analysis and cartographic design. His research has been published in the ACM CHI Conference on Human Factors in Computing Systems and the Journal of the American Association of Geographers. Prior to beginning his Ph.D., Parker was a software engineer at Formidable working on next-generation open-source tools and libraries in the TypeScript, React, GraphQL, and Node.js ecosystems. He is also a StrangeLoop alum, having spoken about his work in the ReasonML community at StrangeLoop 2019.

Formal semantics for multi-language programs

9/21 4:35 pm-5:15 pm, Union Station Grand DE (PWLConf)

Multi-language programs are ubiquitous and language designers have long been designing programming languages to support interoperability. We've had platforms such as .NET, JVM, and COM that facilitate interoperability, and languages such as Scala, F#,Â SML.NET, and many more that treat it as a central design feature. In a 2007 paper, Matthews and Findler pointed out that most multi-language research was focused almost exclusively on how to implement interoperability efficiently and not on the quite subtle semantics of these features. They presented a multi-language semantics framework that gives language designers a methodology for taking two languages, adding interoperability boundaries between them, and giving an operational semantics to those boundaries. I'll describe the mechanics of multi-language semantics, how they support reasoning about the behavior of mixed-language programs, and discuss the impact that this tool has had in the last 15 years on research on compiler correctness, secure compilation, and the design and verification of safe FFIs.

Amal Ahmed *- Professor, Northeastern University*

Amal Ahmed is a Professor at the Khoury College of Computer Sciences at Northeastern University. Her interests include type systems and semantics, compiler verification, language interoperability, and secure compilation. Her earlier work showed how to scale the logical relations proof method to realistic languages, leading to wide use of the technique, including for soundness of advanced type systems such as Rust's, correctness of compiler transformations, and verification of fine-grained concurrent data structures. Her current work includes design and verification of safe FFIs and richer ABIs, and development of RichWasm, a richly typed WebAssembly that supports safe, shared-memory, inter-language linking. Over the last decade, she has been a frequent speaker and less-frequent organizer of the Oregon Programming Languages Summer School (OPLSS) and the Programming Languages Mentoring Workshop (PLMW), which seek to broaden participation in PL research.Â

Designing Dope Distributed Systems for Outer Space with High-Fidelity Simulation

9/21 4:35 pm-5:15 pm, Union Station Grand F

This talk will recount the use of deterministic, high-fidelity simulation to develop the guidance, navigation, and control software for VISORS, a state of the art solar science mission, launching 2024, that will demand tight coordination between multiple spacecraft to image the sun with unprecedented resolution. VISORS will push the boundaries of what software can do in space, and to create it, our team at the Stanford University Space Rendezvous Laboratory had to push the boundaries of how we simulate software on the ground.

Sharing lessons learned from VISORS, this talk will detail techniques for event-driven high-fidelity simulation and interface design to find defects quickly, simplify debugging, and move faster when developing any distributed software system. Beyond simulating real-world systems like spacecraft, aircraft, robots, and autonomous vehicles, these techniques can also be used to simulate databases, service-oriented architectures, and other concurrent applications.

Toby Bell *@tobyyybell - Stanford University, Space Rendezvous Laboratory*

Toby Bell is a PhD student in Aeronautics & Astronautics advised by Simone D'Amico in the Space Rendezvous Laboratory at Stanford University. His research focuses on distributed space systems, software simulation, and spacecraft flight software optimization. He received his MS in Computer Science from Stanford University and previously worked as a flight software and simulation engineer at Astranis Space Technologies and a Starlink process engineer at SpaceX. His passions include choral singing, social dancing, and building compilers.

The level of human involvement behind Remote Desktop Protocol brute-force attacks

9/21 4:35 pm-5:15 pm, Union Station Regency AB

We launched high-interaction honeypots on the Internet from which we collected and analyzed over 3.4 million connections attempts that supplied hashed credentials over a period of 3 months. We used that information to measure the extent of human involvement and the level of sophistication behind brute-force attacks on Remote Desktop Protocol. Different time patterns in attacks are presented, and five different clusters of attackers' strategies are revealed. Risk mitigations are discussed.

AndrÃ©anne Bergeron *@AndreanBergeron - Cybersecurity Researcher, GoSecure*

Andréanne Bergeron has a Ph.D. in criminology from the University of Montreal and works as a cybersecurity researcher at GoSecure. Acting as the social and data scientist of the team, she is interested in online attackers' behaviors. She is an experienced presenter with over 38 academic conferences and is now focusing on the infosec field.â€¯She has presented at BSides Montreal, NorthSec, CypherCon and Human Factor in Cybercrime amongst others.â€¯

Astronaut Horse: Stable Diffusion Artist Collaborations

9/21 4:35 pm-5:15 pm, Union Station Regency C

Open source generative AI tools have the potential to help us fundamentally redefine our relationship to the creative process. This applies to visual artists as much as it does to creative writers, musicians and software developers.

Starting at the Recurse Center in late 2022 a group of developers, researchers and fine artists began collaborating to develop a series of tools taking advantage of Stable Diffusion by way of resources made available by fast.ai and Hugging Face, which has generously supported us with an ongoing Community GPU grant.

This talk would share the tools built, artworks generated and lessons learned from the project so far including select images generated by fine-tuned artist models and physical artworks made in response by collaborating artists.

http://www.astronaut.horse

Tom Betthauser *- Full-Stack Developer / Visual Artist*

I'm a full-stack developer and former studio art / art history instructor originally from San Francisco, currently living and working in Sacramento CA. My studio practice is focused on painting, drawing, sculpting and building software tools to enrich my creative work. I attended the Recurse Center in 2022 and the Yale School of Art in 2012.

Strange Loop Party

9/21 7:00 pm-9:30 pm, City Museum

We're taking over the City Museum for the Strange Loop party! The City Museum is a warehouse packed full of explorable art made from repurposed materials. The museum is fractal in nature and difficult to describe. It features all manner of slides, tunnels, and beautiful weirdness. They have a strict "No Maps!" policy - it's up to you to find your own path.

- Date: Thu Sept 21
- Time: 7-9:30 pm
- Shuttles: available continuously between Union Station (20th Street) and the City Museum from 6:45-10 pm
- Drinks: Available for purchase
- Food: Not available, please eat before!

Using data-driven metrics to anticipate and prevent security incidents

9/22 9:30 am-10:10 am, Union Station Grand ABC

Working on an incident response team means you deal with fire after fire. At GitHub, we developed a team whose primary focus was not to respond to fires but to anticipate them. This talk will cover how the new PSIRT Engineering team was able to prevent future incidents from happening and reduce pressure on the incident response team by analyzing data driven metrics, digging into specific product area pain points and performing variant analysis of historic incidents.

Caitlin Buckshaw *@caitlin_v3 - GitHub, Product Security Engineer*

Caitlin Buckshaw is a Product Security Engineer at GitHub. With over a decade of experience in the IT/Security domain, she has channeled her skills into product security and incident response in recent years. Her mission is to employ a data-driven approach, along with an emphasis on education and her ardor for security, to delve deeper into the nuances of incidents and develop comprehensive measures to mitigate their impact and prevent future happenings. In her non-security time, she enjoys walks along the Carolina beach and gardening.

Experimentation: putting research papers into prod

9/22 9:30 am-10:10 am, Union Station Grand DE (PWLConf)

Innovation happens when we apply concepts from research papers and put them into production. Taking complex ideas and applying them to a real world production environment requires one key step: online evaluation. Theory and offline evaluations can only take you so far, you have to evaluate ideas in an online controlled setting. The better your experimentation practices are, the better you will be at evaluating research in production settings.Â

In this talk, we will zoom in on experimentation strategies from popular research papers that in theory, should really unlock innovation and the speed of experimenting but in practice, are hard to apply, why that's the case and what you can do to put research paper concepts into production.

Leemay Nassery *@LeemayNassery - Engineering leader at Spotify*

Leemay Nassery is an engineering leader at Spotify. She is also the author of Practical A/B Testing.

New algorithms for collaborative text editing

9/22 9:30 am-10:10 am, Union Station Grand F

Real-time collaboration made apps like Google Docs and Figma popular, and many other apps have also been adding similar features. However, the algorithms that enable collaboration are surprisingly tricky, even for seemingly simple data types like text. In this talk I will share insights from our recent research on collaborative text editing, answering questions such as: How can we combine ideas from version control, such as branches and pull requests, with real-time sharing? And how can we move from cloud-centric apps to a more decentralised, local-first model of software without sacrificing features or performance?

Martin Kleppmann *@martinkl - Distributed systems researcher*

Martin Kleppmann is a researcher in distributed systems and security protocols at the Technical University of Munich and at Ink & Switch. He received his PhD from the University of Cambridge. In a previous life he was a Silicon Valley software engineer and entrepreneur, cofounding and selling two startups and working on large-scale data infrastructure at LinkedIn. He is the author of the best-selling O'Reilly bookÂ Designing Data-Intensive Applications.

Playable Quotes for Game Boy Games

9/22 9:30 am-10:10 am, Union Station Regency AB

It's easy to quote text, music, and movies. But what about software?

This session covers a working prototype of a system that can be used to make quotes of Game Boy games.

JoÃ«l FranuÅ¡iÄ‡ *- Artist/Programmer*

JoÃ«l Franusic is an artist and programmer from Alameda, California. His art and work focus on his mission to make all software, from all time, instantly available for use by any programer.

Adam Smith *@rndmcnlly - Assistant Professor of Computational Media, UC Santa Cruz*

Adam is a livelong recreational programmer and career academic. His Design Reasoning Lab aims applies artificial intelligence methods to the development of tools for creating, maintaining, interpreting, archiving, and transmitting meaningful moments of interaction with media like videogames and mobile apps.

Ectype - bringing type safety (and more!) to vanilla JavaScript

9/22 9:30 am-10:10 am, Union Station Regency C

More than a library but not quite a language, Ectype showcases a novel approach to web languages: rather than creating a completely new syntax and transforming that code into JavaScript, Ectype reshapes JavaScript itself into a cleaner, more deliberate language.

Ectype uses a novel pairing of a library and type-checker that work in tandem to bring new language features like static type-safety, runtime type introspection, and more, directly to vanilla JavaScript. Declare types and validate foreign data, all from the familiar syntax of JavaScript, backed by the guarantees of static type-checking.

At the same time, Ectype allows JavaScript to be a *scripting* language again: because files written with Ectype are all valid JavaScript, they can be immediately run in Node, Deno, Bun, the browser, and anywhere else JavaScript will run - all without needing to be compiled or bundled first.

Holly Wu *- Recreational Language Enthusiast*

Holly has been doing full-stack web dev in one form or another for over a decade, but it wasn't until 2017 that she got bit by the programming language design bug and disappeared down the rabbit hole of trying to make JavaScript better. She emerges from that rabbit hole before you now, covered in peculiar flowers and luminous fungi, to present to you the discoveries from her journey.

Computational Physics, Beyond the Glass

9/22 10:20 am-11:00 am, Union Station Grand ABC

Seymour Papert's book "Mindstorms" is a foundational document in the Constructionist theory of education. Learners learn best from the world by building things in it and physically testing its bounds, and teachers in this tradition don't teach "at" learners, but create rich "microworlds" full of insight for them to explore.

This talk will discuss my work using the Emmy computer algebra system and its powerful physics and differential geometry engine to animate microworlds instantiated beyond the laptop's glass. We'll explore a tabletop interface built for kids to explore gravity, and robots tricked by Lisp into acting like orbiting bodies. These tools represent an attempt to smuggle Papert's view of education and learning into modern classrooms, libraries and living rooms.

Sam Ritchie *- Researcher, Mentat Collective*

Sam is a researcher at the Mentat Collective, currently working on a series of interactive, multiplayer computational textbooks for exploring mathematical physics and other forms of modeled reality. He has worked as an engineer at (Google) X, Stripe, Twitter, and founded Paddleguru and Racehub. Sam is most well known in the software world as the author of Summingbird, Algebird, and SICMUtils, and as the maintainer of Cascalog. He has a secret identity as a mountain athlete and amateur aircraft mechanic, and lives with wife Jenna and daughter Juno in Boulder, Colorado.

War Time Proofs and Futuristic Programs

9/22 10:20 am-11:00 am, Union Station Grand DE (PWLConf)

Historians remind us that we cannot predict the future if we don't understand the past. And the past sometimes has new explanations, if we look for them. I want to show you a particular instance of the Curry Howard correspondence, based on the work of Kurt GÃ¶del in the 40s that I claim is an example of this phenomenon of new explanations from the past. GÃ¶del wanted to prove the consistency of arithmetic by restricted means. In the forties he came up with the ingenious idea of interpreting any formula of arithmetic as a formula of the shape

```
<mi>âˆƒ</mi>
<mi>u</mi>
<mo lspace="0.2222em" rspace="0.2222em">:</mo>
<mi>U</mi>
<mi>âˆ€</mi>
<mi>x</mi>
<mo lspace="0.2222em" rspace="0.2222em">:</mo>
<mi>X</mi>
<mi>.</mi>
```

```
<msub>
  <mi>A</mi>
  <mi>D</mi>
</msub>
<mo form="prefix" stretchy="false">(</mo>
<mi>u</mi>
<mo separator="true">,</mo>
<mi>x</mi>
<mo form="postfix" stretchy="false">)</mo>
```

where

```
<mi>A</mi>
<mi>D</mi>
```

is quantifier-free. This was his famous Dialectica interpretation, published in 1958. Some 30 years after following Hyland's suggestion, I showed how that GÃ¶del idea corresponds to a kind of linear logic type theory. This was interesting then because Linear Logic was just beginning and people didn't know how good for Computer Science it was going to be. Now, 35 years later, Linear Logic is used in many advanced programming languages and type systems, and more applications are expected. But we're still explaining and extending the tricks of the old master, as I hope to show you.

Valeria de Paiva *@valeriadepaiva - gh:vcvpaiva - AI principal scientist at theÂ Topos Institute*

Valeria de Paiva is a mathematician and AI principal scientist at the Topos Institute, working on natural language processing, logical inference, and all kinds of semantics, especially Dialectica categories. Before Topos, Valeria worked at top industry NLP labs, including Samsung Research America, Nuance, Deem, Cuil, and for many years at Xerox PARC. Before that, she was a professor at the University of Birmingham, UK. She is very keen on making sure that women are not too short-changed in their professional lives. For that, she maintains the "Women in Logic" website, Facebook group, and blog, helps the ACM-W Scholarship program, and started the international workshop Women in Logic, now in its 7th year.

The Economics of Programming Languages

9/22 10:20 am-11:00 am, Union Station Grand F

In the mythology of open source, programming languages are created by people who seemingly have no direct economic function. They are just really good at compilers (somehow) and have a house to live in (somehow) and have a lifetime to devote to creating a useful programming language (somehow!)

We will examine specific organizations that create programming languages. Where do the salaries for compiler engineers come from? How does Go end up with 5 engineers and Dart end up with 30? Who signs off on these expenses and why? Does this put any boundaries on language design or development practices? And how do the economics work for people outside of the major tech corporations?

My goal is to give the talk I needed to hear 10 years ago when I was just starting on Elm. By clearly delineating the many variations of corporate funding and independent funding, I hope users will come away with a better foundation for evaluating and comparing programming languages.

Evan Czaplicki *@evancz - Creator of Elm*

Evan is the creator and developer of the Elm programming language. He lives in Denmark, working alongside his wife at elm.studio to keep Elm independent and interesting.

Didn't Chrome Already Have a Root Store?

9/22 10:20 am-11:00 am, Union Station Regency AB

After years of operating as "flair" on top of platform root stores and APIs, Chrome now operates an official root program and root store. I helped launch the root store, and this talk explains why Chrome finally did this, how the Web PKI has changed for the better since Chrome was first launched, and how the Chrome Root Program helps us build a better Web PKI in the future. We face new challenges now, including rolling out post-quantum security, and this necessitates a change in how we think about PKI in Chrome---a continued focus on security, but a new focus on simplicity and automation. You'll walk away learning how certification authorities (CAs) get included in root stores, how to run a modern and compliant CA, and why we think we can improve security by cutting requirements.

David Adrian *@davidcadrian - Product Manager, Chrome Security, Google*

I am currently a product manager at Google, adding value to Chrome Security. Despite this, I identify both as an engineer and security researcher. At Google, I work on all things HTTPS and PKI in Chrome.

Prior to Google, I cofounded Censys, an attack-surface management security startup. I defended my PhD at the University of Michigan in 2018. My research was based on using Internet-wide scanning to measure how cryptography is used on the Internet. It led to the discovery of the Logjam and DROWN attacks on TLS, and the creation of Censys. I'm also a core contributor to the ZMap open-source project.

In 2021, I accidentally started the "Security. Cryptography. Whatever." podcast with Deirdre Connolly and Thomas Ptacek.

An approach to computing and sustainability inspired from permaculture

9/22 10:20 am-11:00 am, Union Station Regency C

In this session, I will explore some playful low-power, sometimes analog, computation systems and esoteric programming languages, designed to work offline, on salvaged devices, advised from spending the past 7 years sailing around the Pacific Ocean.

No seafaring experience required.

Devine Lu Linvega *- Hundred Rabbits, Crew*

Devine Lu Linvega is a designer and musician living aboard a sailboat somewhere on the foggy shores of the Pacific ocean. Devine has been developing and teaching livecoding environments all the while fending off the rising tide of noxious modern software and operating systems.

Friday Lunch

9/22 11:00 am-1:00 pm, Union Station

Food trucks will be provided at Union Station on 20th Street

Can a Programming Language Reason About Systems?

9/22 11:20 am-12:00 pm, Union Station Grand F

We have lots of languages that apply logic to verifying, simulating, or generating systems, but they all use the syntax of mathematical proofs which many programmers find difficult to learn. Fault is a new language developed to help people model systems and reason about their behavior and limitations. These systems can be technical, they can be organizational, regulatory or they can be hybrids. This talk will cover the basic styles of logic programming-model checking, simulation, symbolic execution-and how Fault converts a programmer friendly syntax into logic friendly models. Watch Fault do things like give advice about 5,000 page export regulations, tell you when your micro services architecture is going to start breaking down, and solve sudoku puzzles.

Marianne Bellotti *@bellmar - Rebellion Defense*

Marianne Bellotti is the author of Kill It With Fire, Building Safety Critical Systems, and Hiring Engineers, as well as a popular column on technology and system safety. She has built data infrastructure for the United Nations to help humanitarian organizations share crisis data and tackled some of the oldest and most complicated computer systems in the world as part of United States Digital Service. At Auth0 she ran Platform Services, a portfolio that included shared services, untrusted code execution, and developer tools. Currently she runs engineering teams at Rebellion Defense. She can be found on most social networks under the handle bellmar.

From Geometry to Algebra and Back Again: 4000 Years of Papers

9/22 11:25 am-12:05 pm, Union Station Grand DE (PWLConf)

A whirlwind tour of the history of our system of vector algebra - including applications from art, science, and programming - leading to a plea that we trade it in for something even better: Geometric Algebra.

Jack Rusher *@jackrusher - Applied Science Studio*

Jack Rusher's long career as a computer scientist includes time at Bell Labs/AT&T Research and a number of successful startups. His current work focuses on the deep relationship between art and technology.

Cursorless: A spoken language for editing code

9/22 11:30 am-12:10 pm, Union Station Grand ABC

If you could design a spoken language from scratch for editing code, how would it look? What would be your nouns? Would they be tokens? Functions? Lines? What would be your verbs, your adjectives, and your adverbs?

Cursorless is one answer to these questions: a spoken language designed for maximally efficient code editing by voice. Cursorless leverages the tree-sitter real-time parser to enable high-level, "smart" code manipulations while retaining the flexibility to use "dumb" primitives like tokens, lines, delimiter pairs, and regexes when necessary.

Learn how a handful of simple abstractions - actions, modifiers, marks, and scopes - empower Cursorless users to create powerful and concise command chains that would leave even the most seasoned vim user drooling on their keyboard.

Pokey Rule *@PokeyRule - Creator of Cursorless*

Pokey Rule is the creator and lead maintainer of Cursorless. He releases all of his code under the MIT license and relies on donations from users to sustain the project. Prior to working on Cursorless, Pokey managed a machine learning team at Globality. He studied programming languages and human-computer interaction at Stanford University.

Unmasking the Godfather - Reverse Engineering the Latest Android Banking Trojan

9/22 12:00 pm-12:40 pm, Union Station Regency AB

Banking malware has wreaked havoc on millions of Android users over the last few years, employing advanced stealth techniques to evade detection. During this presentation, I will reverse engineer the Java code of the pervasive Godfather Android banking malware demonstrating how to recognize its modern devious tactics. The Godfather malware surfaced towards the end of 2022 as another family dedicated to stealing banking data and funds from victims. Initial infection spawns from abuse of core Android system services and accomplish their theft by spying on infected devices, exfiltrating data, or performing remote device commands. Some banking trojans even generate fake HTML overlays designed to trick users into willingly entering credentials. But don't worry, I'm not just here to scare you! By the end of this talk, you will thoroughly understand the Godfather's deceptive functionality and likely think twice before granting an application extra permissions.

Laurie Kirk *@lauriewired - Reverse Engineer at Microsoft*

I currently work as a Security Researcher at Microsoft in incident response. I specialize in cross-platform malware analysis with a focus on mobile threats. I also run a YouTube channel (@LaurieWired) that covers all sorts of in-depth Malware Analysis, Reverse-Engineering, Exploitation and security topics.

https://www.youtube.com/@lauriewired

I worked as a Software Engineer for an aerospace company before finding my current calling in Cyber Security. I received my Bachelor's Degree from Florida State University in Computer Science with a minor in Math. There, I dove into learning computer architecture, algorithms, and my favorite, low-level programming.

Birdsong as code

9/22 12:05 pm-12:45 pm, Union Station Regency C

Most people intuitively understand birdsong as a form of music, yet the practical and artistic pressures on bird musicians are very different to those on human artists. In this talk, we will transcribe birdsong into code and use it to investigate how birds' musical structures differ from our own. In doing so, we will take advantage our ability to create novel and context-specific abstractions in code, which is not possible in traditional music notation.

Chris Ford *@ctford - Director de TecnologÃa, Thoughtworks EspaÃ±a*

Chris has coded both correct and incorrect programs for many years, although he prefers the former. He has also coded both correct and incorrect musical compositions for many years, though he doesn't really mind which. As well as a longstanding interest in functional programming, types and continuous delivery, Chris is always interested in learning better ways to write programs that do what they are supposed to do. He is Director de TecnologÃa for Thoughtworks EspaÃ±a based in Barcelona.

A Long Strange Loop

9/22 1:40 pm-2:20 pm, Stifel Theatre

A look back at the history and highlights of Strange Loop

Alex Miller *- gh:puredanger - Strange Loop LLC*

Alex Miller is the creator of the Strange Loop, Clojure/west, and Lambda Jam conferences. Since 2013 he has been part of the Clojure language team, currently working at Nubank, the largest commercial user of Clojure. He is the co-author of Clojure Applied, Programming Clojure 3rd edition, and the upcoming Clojure Brain Teasers.

How to Make Hard Things Easy

9/22 2:30 pm-3:15 pm, Stifel Theatre

Julia Evans *@b0rk - gh:jvns - Wizard Zines*

Julia Evans is a software engineer and writer based in Montreal who loves investigating weird bugs and helping people learn how computers work. She runs Wizard Zines, where she writes & publishes tiny books that teach computing fundamentals. She also helps organize !!Con (a conference about the joy, excitement, and surprise of computer programming), and previously worked as a software engineer at Stripe.

Drawing Comics at Work

9/22 3:30 pm-4:15 pm, Stifel Theatre

Randall Munroe *@xkcd - xkcd*

Randall MunroeÂ is the author ofÂ New York TimesÂ bestsellersÂ What If? 2,Â How To,Â What If?, andÂ Thing Explainer; the science question-and-answer blogÂ What If?;Â and the popular web comicÂ xkcd.

A former NASA roboticist, he left the agency in 2006 to draw comics on the internet full time.

He lives in Massachusetts.

Closing

9/22 4:15 pm-4:45 pm, Stifel Theatre

Alex Miller

Closing Reception and Signing

9/22 5:00 pm-6:30 pm, Stifel Theatre

After the conference, join us for a closing reception as we toast the end of Strange Loop!

Also, Randall Munroe (xkcd) will be doing a signing. You can pre-register to receive his latest book "What If? 2" at the conference (or copies will also be available on-site).

Randall Munroe *@xkcd - xkcd*

Randall MunroeÂ is the author ofÂ New York TimesÂ bestsellersÂ What If? 2,Â How To,Â What If?, andÂ Thing Explainer; the science question-and-answer blogÂ What If?;Â and the popular web comicÂ xkcd.

A former NASA roboticist, he left the agency in 2006 to draw comics on the internet full time.

He lives in Massachusetts.