PRINTABLE SCHEDULE

AI Engineer World's Fair 2026

The largest technical conference for AI engineers

When June 29 – July 2, 2026 **Where** Moscone West, San Francisco, CA **Sessions** 562

Interactive schedule: ai.engineer/worldsfair · Register: app.ai.engineer

CONTENTS

Day 1 — Workshop Day Monday, June 29, 2026

contents ↑

9:00am

SPONSOR Track 1 9:00am-11:00am

From Vibes to Production: Evaluating and Shipping AI Agents That Work 101

Laurie Voss — Head of Developer Relations, Arize AI

Building an AI demo is easy. Knowing whether it actually works — and keeping it working in production — is the hard part. Most teams ship agents on vibes: they try a few prompts, the output looks good, and they push to production with no real way to measure quality or catch regressions. This hands-on workshop walks through the full lifecycle of shipping a real AI agent, using a working financial-analyst agent built on the Claude Agent SDK as the running example. You'll instrument it with tracing, do structured error analysis on its actual outputs, and build a layered evaluation suite — from cheap deterministic code checks to LLM-as-a-judge evaluators with custom rubrics. We'll cover the parts most tutorials skip: why agents fail in ways single LLM calls don't, the eval anti-patterns that quietly mislead you, and how to know whether you can even trust your judge (meta-evaluation). Finally, we'll close the loop: turning eval results into datasets and experiments, running evals online against production traffic, wiring them to monitors and alerts, and feeding failure explanations back to a coding agent to actually fix the underlying problems. You'll leave with a runnable notebook and a repeatable, evaluation-driven workflow you can apply to your own agents the next day.

SPONSOR **Track 2** 9:00am-11:00am

AI on Your Lakehouse: Context Comes in Shapes, Not Queries

Zach Blumenfeld — AI Research Engineer, Neo4j

Your agent can reach your data but still can't use it reliably: vector search and Text2SQL each hand it a slice, but not the view to know what's truly relevant and how to connect the right info. Without that, answers come back confident but wrong, and agent decisions cannot be trusted. The problem isn't caused by a bad model or bad query, but rather a lack of context, and thinking in terms of shapes is what cracks it. In this hands-on session, you'll learn how to build three reusable graph shapes from your lakehouse data using Neo4j, so your agent can navigate and view the right context to answer and act accurately: - Table of Contents (Trees) — navigate what's there - Themes (Communities) — surface patterns nobody named - Connections (Paths & Cycles) — trace how entities, documents, and records relate Portable to BigQuery, Databricks, Snowflake, or anywhere. You'll leave with real, practical techniques and the code to run with your own data and agents.

SPONSOR **Workshops Day 1 · Track 3** 9:00am-11:00am

Cooking with Codex

Charlie Guo — Developer Experience Engineer, OpenAI · **Gabriel Chua** — Developer Experience Engineer, OpenAI

Codex is changing how technical teams ship across the software development lifecycle, from feature implementation to code review and automation. But the real unlock comes when these practices move beyond a single workflow and become shared systems a team can trust. In this hands-on session, you'll use Codex across real development and knowledge-work scenarios: structuring tasks, supervising agentic work, coordinating subagents, using plugins and MCPs, and combining Codex with OpenAI's frontier reasoning, coding, and multimodal models. Bring your laptops and leave with reusable demos and a set of Codex recipes your team can adapt.

SPONSOR **Workshops Day 1 · Track 4** 9:00am-11:00am

The best SDLC is the one you build yourself: Why orchestration changes everything

Shane Wolf — Atlassian · **Andrei Bocan** — Principal Engineer, Atlassian

Industry research shows AI productivity gains have plateaued at 10–15% — because today's tools only optimize the 20% of a developer's day spent writing code. The real bottlenecks are left and right of code: planning, orchestration, review, and operations. We'll also explore the value of AI-powered code reviews - from establishing code standards that AI can seamlessly enforce, to triggering agentic pipelines that autonomously fix issues. Join Atlassian's Shane Wolf and Andrei Bocan for a hands-on deep dive into the AI-native SDLC. In this workshop, we'll move past single-player copilots and show you how Atlassian is turning Jira into an AI-native orchestration layer for the entire software development lifecycle. Then, we'll go further. You'll learn how to build custom automations that chain these capabilities together, transforming your Jira board into an agentic software factory where humans set intent and agents execute.

SPONSOR **Workshops Day 1 · Track 5** 9:00am-11:00am

AI Security Engineer Foundations + Certificate

Javier Garza — Developer Advocate, Snyk

In each of the two sessions, we cover 6 modules and participants receive a certificate of completion at the end. The modules are: OWASP Top 10 for LLM, Addressing Shadow AI, AI Threat Modeling, Securing Agents & MCP, Securing Vibe Coding, & AI Red Teaming

SPONSOR **Workshops Day 1 · Track 6** 9:00am-11:00am

Total Recall: Agent Memory and Harness Engineering

Ignacio Martinez — AI Developer Advocate, Oracle

In this hands-on workshop you'll build a working autonomous agent from the harness up, in a notebook, then see it live in a full working web application and leave with one that can write and run its own automations. You'll implement every surface area yourself: a set of predefined tools, persistent memory through the Oracle AI Agent Memory package, orchestration with LangChain and LangGraph, and LLM access through OCI GenAI Service, composing the full set of Oracle primitives into one harness you understand end to end. Most teams assemble that harness from a dozen disconnected services: one store for vectors, another for state, a separate reranker, a bolt-on memory layer. We take the opposite approach, on a single unified memory core. The organizing principle is optionality by default: you shouldn't have to choose your memory substrate up front. With Oracle AI Database you get file system and database memory in one place, embedding models and rerankers running inside the database kernel, and every retrieval strategy an AI workload needs without leaving the core. And consolidating onto one core is what keeps the whole thing tractable. You know the drill: a production harness has you holding all those moving parts in your head at once, and most of your attention goes to keeping them in sync rather than improving the agent. Pull that sprawl into a single core and the cognitive load drops. You get to think about what the agent does, not where its state lives. That's the difference between controlling your harness and renting its pieces.

SPONSOR Track 7 9:00am-11:00am

Agents That Own Their Inference: Building Production AI Agents on Dedicated GPUs

Du'an Lightfoot — Senior AI Engineer, Akamai Technologies

Every production agent today is renting its intelligence. You're paying per token, sending your customer's data to someone else's servers, and hoping the provider doesn't rate-limit you during your launch. For most teams, that's fine. But for a growing number of teams in regulated industries, with high-volume products, latency-sensitive workloads, or rising token bills, it's starting to look like a liability. In this 120-minute hands-on workshop you'll get a dedicated GPU and build an agent that runs on infrastructure you control. You'll stand up vLLM, point your agent at it, and drive concurrent load through the stack until you can see batching, KV cache pressure, and throughput limits in the metrics. Then you'll optimize the deployment to improve throughput while keeping per-request latency in line. The focus isn't agent frameworks. It's the inference layer underneath them. You'll leave with working code and a real understanding of continuous batching under real concurrency, KV cache tradeoffs, vLLM's metrics, and the bottlenecks that only show up when you operate the inference server yourself.

SPONSOR Workshops Day 1 · Track 8 9:00am-11:00am

Open-Source Inference Engineering for the Agentic Era

Zain Hasan — Staff AI/ML Engineer - DX, Together AI · **Yubo Wang** — LLM Inference, Together AI · **Qingyang Wu** — Staff Research Scientist, Together AI · **Jue Wang** — Senior Staff Researcher, Together AI

Agentic coding workloads demand long contexts, multi-turn conversations, and throughput at a scale that most inference engines weren't built for. TokenSpeed is a new open-source engine purpose-built for this regime, built collaboratively by NVIDIA DevTech, AMD Triton, Qwen Inference, Together AI, and others. In this 2-hour hands-on workshop, Together Inference Research Engineers and a TokenSpeed co-creator will cover TokenSpeed architecture, deploying your first model, optimizing for agentic workloads, kernel and hardware tuning, and throughput/latency trade-offs.

SESSION Track 9 9:00am-11:00am

Advanced workshop: Mastering AI Observability

Doug Guthrie — Solutions Engineer, Braintrust

Your AI is in production, but is it actually good? In this hands-on workshop, you'll learn how to uncover patterns in your production traces using Braintrust Topics, build custom scorers to target real issues, and systematically improve your agent. By the end, you'll have a repeatable eval workflow and trace-backed evidence that your AI is actually doing what you think it is.

SPONSOR Track M 9:00am-10:15am

Get Started with Models in Microsoft Foundry to Build AI Apps

Pamela Fox — Principal Cloud Advocate, Microsoft

In this hands-on lab, you will build a production-ready AI application using Microsoft Foundry, with no fine-tuning or deep machine learning expertise required. You will discover and select models, provision a Foundry project, and connect to a hosted model using the OpenAI SDK. You'll implement a comment moderation workflow, compare model outputs, and package the solution as a hosted agent using Python, ready for real-world integration.

11:05am

SPONSOR Posttraining & Midtraining · Track 1 11:05am-12:05pm

Building self-learning loops for your agent

Fuad Ali — Senior Product Manager, Arize AI

Building an AI demo is easy. Knowing whether it actually works — and keeping it working in production — is the hard part. Most teams ship agents on vibes: they try a few prompts, the output looks good, and they push to production with no real way to measure quality or catch regressions. This hands-on workshop walks through the full lifecycle of shipping a real AI agent, using a working financial-analyst agent built on the Claude Agent SDK as the running example. You'll instrument it with tracing, do structured error analysis on its actual outputs, and build a layered evaluation suite — from cheap deterministic code checks to LLM-as-a-judge evaluators with custom rubrics. We'll cover the parts most tutorials skip: why agents fail in ways single LLM calls don't, the eval anti-patterns that quietly mislead you, and how to know whether you can even trust your judge (meta-evaluation). Finally, we'll close the loop: turning eval results into datasets and experiments, running evals online against production traffic, wiring them to monitors and alerts, and feeding failure explanations back to a coding agent to actually fix the underlying problems. You'll leave with a runnable notebook and a repeatable, evaluation-driven workflow you can apply to your own agents the next day.

SPONSOR Track 2 11:05am-12:05pm

RAG Needs a Map: Using GraphRAG to Retrieve Connected Context

Nyah Macklin — Sr. Developer Advocate, Artificial intelligence, Neo4j

Vector search is good at finding similar text, but real answers often depend on how facts, entities, and documents connect. In this hands-on workshop, you'll build a GraphRAG workflow that uses relationships to retrieve connected context for more grounded AI responses.

WORKSHOP Workshops Day 1 · Track 3 11:05am-12:05pm

How I learned to stop worrying and love the sandbox

Matt Brockman — AI Engineer, E2B

Running sandboxes at scale can get painful. How do you manage a thousand concurrent sandboxes? We'll cover burst traffic, fast sandbox creation under load, resource exhaustion, shared state with volumes, and per-user data isolation. Then you'll trigger each failure, implement fixes, and see the cost impact in real time. You'll leave with hands-on experience debugging sandbox failures and a set of observability and scaling patterns you can start implementing.

WORKSHOP Workshops Day 1 · Track 4 11:05am-12:05pm

The model swap workshop

Pamela Fox — Principal Cloud Advocate, Microsoft · Arun Sekhar — Principal Product Manager for AI Developer Experience, Microsoft

Frontier labs are releasing new models constantly, and it is hard to know when "better" is better enough to justify touching a working system. On top of that, "just swap the model" often turns into real work because providers expose different APIs and different expectations around tools and structured outputs. The model swap workshop is a hands-on bake-off across frontier LLMs. We will run the same scenarios using multiple models (OpenAI, Anthropic, Kimi, and more) and compare results side by side for agentic tool use, structured outputs, and multimodal tasks. Swapping models is not just changing a model name. In this workshop, you will actually do the swaps, including moving between OpenAI-style Responses APIs and Anthropic-style Messages APIs, then see what breaks and what needs to change in your prompts, tool definitions, and JSON strategies. We will finish by running a small eval suite so you can quantify tradeoffs instead of relying on vibes. We will provide the Microsoft Foundry environment for access to the models, no account needed.

WORKSHOP Workshops Day 1 · Track 5 11:05am-12:05pm

Teaching Agents to Search: Building Synthetic Training Pipelines with NVIDIA Data Designer

Dhruv Nathawani — Research Scientist, Nvidia

Modern agentic systems often fail because the right training data simply does not exist. Search agents are a perfect example: if you want a model to browse the web effectively, you need high-quality multi-step trajectories that teach it how to search, refine queries, inspect sources, and recover from dead ends. Those datasets are rarely available off the shelf. In this hands-on workshop, we will show how NVIDIA used Data Designer to build synthetic supervised fine-tuning data for search-capable Nemotron models. Participants will learn how to translate a target capability into a scalable data generation pipeline: defining task structure, generating strong seed examples, producing realistic search trajectories, filtering low-quality generations, and converting traces into training-ready records. Using a real search-agent use case, we will walk through the design decisions behind teaching Nemotron Super to browse the web, including how to create BrowseComp-style tasks, generate tool-use rollouts, and manage the tradeoffs between diversity, correctness, and yield. We will also cover the practical realities of production synthetic data workflows, including validation, dataset curation, and where most pipelines break down. But the goal of this workshop goes beyond search. Participants will leave with a reusable framework for designing any dataset they wish they already had: starting from the behavior they want to teach, mapping that behavior into a data schema, generating examples at scale, and iterating until the dataset is useful for training. By the end of the session, attendees will not only know how to build synthetic data for search agents, but how to design custom datasets for specialized behaviors across reasoning, tool use, and domain-specific applications. Attendees will leave with a practical methodology for synthetic data design, plus hands-on familiarity with NVIDIA Data Designer as an open-source system for rapid experimentation.

WORKSHOP Workshops Day 1 · Track 6 11:05am-12:05pm

Local LLMs and workstation agents: Part 1

Ahmad Osman — Founder & CEO, Osmantic

Have you heard "Buy a GPU," "Opensource AI Must Win," or "Local AI FTW" before? This workshop will be a practical window into that confusing world and a practical map for understanding what different Local AI hardware is actually capable of and which models make sense on each class of machine. Whether you are just getting started or already running models every day, we will demo and work through why a Mac mini, M4 Pro MacBook Pro, M5 Max MacBook Pro, RTX 5070 8GB laptop, Strix Halo box, DGX Spark, and 2x RTX PRO 6000 Blackwell machine should not be configured, benchmarked, or used the same way. What are you trying to run? How much VRAM or Unified Memory do you actually need? When does a small machine make sense? When do you need a real GPU box? When does long context, tensor parallelism, or serving infrastructure start to matter? This should be useful to everyone: people curious about local AI, people buying their first capable machine, people already running models, and people trying to use local inference for scalable agentic workflows. We will close by showing how Codex can automate the boring part: give it my Inference Engine article, the hardware target, and the model of your choice, then ask it to propose the engine, environment, flags, batch settings, KV-cache settings, and benchmark and evaluation plan.

WORKSHOP Workshops Day 1 · Track 7 11:05am-12:05pm

How to Build Quality Gates into Agentic Coding Workflows

Nnenna Ndukwe — Principal Developer Advocate and Software Engineer, Qodo AI

AI coding agents can now generate code at unprecedented speed. But faster code generation creates a new engineering problem: how do we know when agent-written code is actually safe, maintainable, and ready to merge? In this hands-on workshop, attendees will build an agentic coding workflow with enforceable code quality gates across planning, implementation, testing, and code review. By the end of the session, participants will have a working reference pattern for agentic software delivery: an AI-assisted workflow that can inspect a repo, implement a change, run tests, evaluate risk, respond to feedback, and surface what still requires human judgment. This is a technical enablement session for engineers building with AI coding agents, platform teams designing agentic SDLC workflows, and AI engineering leaders thinking about how to scale software quality with AI.

WORKSHOP Workshops Day 1 · Track 8 11:05am-12:05pm

What is an Inference Engine, Anyway?

Charles Frye — Member of Technical Staff, Modal

To run state-of-the-art inference yourself, you must master the inference engine: vLLM, SGLang, TRT-LLM, or your own jawn. The inference engine manages the lifecycle of an inference request, from input to output. In this workshop, we'll examine the architecture of modern high performance inference engines, the key techniques that inference engines need to deliver that performance, and the traces and metrics that inference engines emit.

SESSION Workshops Day 1 · Track 9 11:05am-12:05pm

Agent Speedrun: Idea → Code → Deploy → Observe, Fix → Ship

Elizabeth Fuentes Leone — Developer Advocate, Amazon Web Services · Sandhya Subramani — Senior Developer Advocate, GenAI, Amazon Web Services

One agent. Fully deployed to production before the workshop ends. We'll take you from a blank file to a running production agent using Amazon Bedrock AgentCore and Strands Agents, covering the full lifecycle: ideation, coding the agent loop, deploying to serverless infrastructure, wiring up observability, breaking it intentionally, fixing it with tracing data, and shipping the final version. Bring your laptop and leave with a deployed agent.

SPONSOR Track M 11:05am-12:05pm

From zero to deployed on Azure with AI agents

Gustavo Cordido — Cloud Advocate & AI Content Engineer, Microsoft

What happens when you let AI agents do the building? In this hands-on lab, you'll go from an empty terminal to a deployed app on Azure — with GitHub Copilot CLI and coding agents handling the scaffolding, coding, debugging, and deployment. You'll use the new Azure skills to provision resources and wire up services through natural language, no portal required. This isn't a demo you watch. You'll walk out with a real, working dev workflow you can take straight to your next project.

12:10pm

WORKSHOP **Workshops Day 1 · Track 1** 12:10pm-1:10pm

Evals in AI: A Deep Dive

Tejas Kumar — AI Engineer, IBM

“Our evals pass and our velocity is up, so it works.” It’s the most reassuring sentence in AI engineering and also the most dangerous. Teams are shipping more code than ever while incidents per PR and change-failure rates climb, and the instruments meant to catch this are quietly broken. This talk takes apart both halves of that false comfort. First, why velocity lies: the same AI-driven throughput that lights up your dashboard is what’s eroding quality underneath it. Then we explore four ways offline evals deceive you: LLM-as-judge bias (your grader rewards confident, wordy, wrong answers over terse correct ones), staleness, distribution shift between your golden set and real traffic, and single-score evals that hide which step of an agent actually failed. The centerpiece is a live demo. We’ll wire up an LLM judge on stage and watch it crown a confident, friendly, factually wrong answer. Then we’ll fix it live on stage with a three-line rubric change. Same model, different instrument. From there we’ll build up what to measure instead: traces and spans, production observability, probe-based evaluation, error budgets, and quality leading indicators that sit beside every velocity number. Attendees will leave with a five-line checklist they can apply Monday. No prior eval tooling required. If you’ve ever shipped something agentic and had a nagging feeling the dashboards were too kind, this is for you.

WORKSHOP **Workshops Day 1 · Track 2** 12:10pm-1:10pm

From approval loops to autonomous agents with Docker

John Craft — Solutions Engineer, Docker · **Dan Ndombe** — Staff Developer Success Advocate, Docker

"You've invested in the best models, coding agents, and AI tooling. Now comes the hard part: unlocking autonomous development without creating security headaches, governance gaps, or endless approval loops. In this 90-minute hands-on workshop, you'll learn how to run coding agents in isolated environments built for autonomous work, create a 'golden path' for AI-assisted development across your organization, reduce software supply chain risk with secure, hardened containers, manage multiple agents with the right permissions and guardrails, and scale AI-powered development without slowing developers down."

WORKSHOP **Workshops Day 1 · Track 3** 12:10pm-1:10pm

2 hr deep dive on LLM Inference at Scale — Part 1 of 2

Harshul Jain — Senior Software Engineer - ML/AI, Audible · **Tanmay Sah** — Senior Quantitative Modeler, Zions Bancorporation

Most engineers using LLMs can call an API. Far fewer can explain why their model is slow, why it's running out of memory, or how the inference engines powering every major LLM API actually work. This workshop walks through the full inference stack — from how a transformer generates a single token to serving billions of tokens a day with vLLM, SGLang, TensorRT-LLM, Ray, and KServe/llm-d. 60% explanation with live demos, 40% hands-on exercises. Attendees leave with a running vLLM server they benchmarked themselves. Based on the open-source practitioners handbook being built live at github.com/harshuljain13/llm-inference-at-scale (NOTE: this is a 2 hour workshop that happens over lunch break - you should try to have lunch before or after if attending) compute kindly sponsored by Coreweave/Marimo!

WORKSHOP **Workshops Day 1 · Track 4** 12:10pm-1:10pm

Build the Right Thing: Product Engineering for Software Developers (Part 1)

Kent C. Dodds — Software Engineer and Educator, EpicProduct.engineer

There is nothing quite as demoralizing as finishing a feature and realizing you built the wrong thing. The code is clean. The tests pass. The ticket is closed. And none of it matters. This is happening more often, not less. AI makes it faster and cheaper to implement, which means teams can now waste entire sprints on the wrong idea at unprecedented speed. The bottleneck is no longer "can we build it?" It is "should we build it?" and "are we sure we understand the problem?" This session is a condensed introduction to product engineering for builders: the skills that sit upstream and downstream of implementation. We will not try to cover everything a full-day workshop would. Instead, we will focus on the highest-leverage ideas you can apply on Monday. ### What we'll cover 1. Validate before you build Most wrong builds start with an idea that was never tested. You will learn to separate real user pain from solution-shaped requests, and practice discovery questions that surface past behavior instead of hypothetical enthusiasm. 2. Prioritize what deserves to exist Not every good idea should be built now. Especially in the AI era, "we could build this" is not a reason to build it. We will work through a practical prioritization lens, including the Kano model, to help you distinguish fundamentals from delighters from distractions before your team commits. 3. Own the feature, not just the PR Product engineering does not end at merge. You will leave with a clearer picture of end-to-end feature ownership: staying close to users, setting up simple feedback loops, and improving what you shipped instead of moving on to the next ticket. ### Format This is a 2–3 hour session with Kent C. Dodds. Expect focused teaching, real-world examples, and short interactive exercises and discussion. This is not a full simulation lab or a ticket-closing coding workshop. It is judgment practice for engineers who already know how to ship. ### Who this is for Software engineers (and technical builders generally) who: - Have shipped something polished that nobody wanted - Feel pressure to move fast with AI and want a better filter for what deserves to exist - Want stronger product instincts without becoming a PM - Care about owning outcomes, not just closing tasks Some software engineering experience is assumed. No particular stack is required. PMs and designers often find this valuable too. ### What you'll leave with - Discovery questions for ambiguous work - A prioritization lens you can use before committing to a build - A clearer model for feature ownership and post-ship feedback loops - Language for stakeholder conversations when requirements are unclear

WORKSHOP **Workshops Day 1 · Track 5** 12:10pm-1:10pm

From Zero to Leaderboard: Building an End-to-End AI Agent Evaluation Pipeline

Wolfram Ravenwolf — AI Evangelist, Weights & Biases by CoreWeave

Running one agent eval is easy. Running hundreds — with controlled timeouts, replicated configs, and automated collection across distributed VMs — requires infrastructure that most teams end up building from scratch. In this workshop, we shortcut that process and build a rigorous evaluation pipeline end-to-end. Participants will set up and connect the full evaluation stack: **Layer 1 — The Benchmark Runner.** Configure Harbor to orchestrate parallel agent evaluations on Terminal-Bench 2.0, with W&B Sandboxes providing isolated environments for each task. **Layer 2 — The Collection Pipeline.** Use WolfBench to scan distributed VMs for results, deduplicate across runs, download trajectories, and build a local results archive that survives VM teardown. **Layer 3 — The Analysis Framework.** Compute the five-metric framework (Ceiling / Best / Average / Worst / Solid) across replicated runs. Learn to read the spread: when is a model "better"? When is a score difference just noise? **Layer 4 — The Observability Layer.** Upload full agent conversation traces to W&B Weave for per-turn inspection. See exactly where an agent goes wrong — the command it ran, the output it misread, the moment it started looping. **Layer 5 — The Leaderboard.** Generate interactive HTML charts that show the full performance distribution, not a single bar. We'll work with real data from hundreds of production runs, and participants will leave with a working pipeline they can adapt to their own agents and benchmarks. Laptops required; all tools are open-source.

WORKSHOP **Workshops Day 1 · Track 6** 12:10pm-1:10pm

Local LLMs and workstation agents: Part 2

Ahmad Osman — Founder & CEO, Osmantic

From the guy who said "Buy a GPU," "Opensource AI Must Win," and "Local AI FTW": this session shows what you build around the models running locally so agents can actually be effective and efficient when using local models. A local chatbot gives you private text generation. A useful agent needs a system around it: search, scraping, traces, document ingestion, agentic harness integration, and other practical components. The focus of this workshop is setup, not hardware. We will walk through the practical pieces that turn local inference from a model endpoint into the reasoning layer inside a real workflow. The live demo target will be a 2x RTX PRO 6000 Blackwell machine running models locally and using it across different agentic harnesses. The goal is to show how Local AI can be more than private and offline: it can be useful, inspectable, controllable, and built into infrastructure you actually own. Attendees should leave with a practical mental model for building Local AI systems that can read, search, cite, act, and evaluate themselves.

WORKSHOP Workshops Day 1 · Track 7 12:10pm-1:10pm

Beyond RAG: Build a Relational Context Engine from Scratch

Peter Werry — Founding Engineer, Unblocked

In this workshop we'll explore the importance of context engines in modern engineering workflows, and we'll look at why traditional RAG techniques are no longer enough to deliver the context agents need. We'll build a structured query engine that fills the gaps left by RAG, translating natural language into validated database queries over GitHub PR and Issue data. We'll implement schema-aware prompting, identity resolution, query validation, and error-driven retry loops, and you'll walk away with a working query engine for your GitHub repository.

WORKSHOP Track 8 12:10pm-1:10pm

Building AI Agents with Real-Time Web Data

Yohan Raju — Pre-Sales Expert, Bright Data

Your AI agent is only as good as the data it can access — and static training data isn't enough anymore. In this hands-on workshop, you'll learn how to connect AI agents to the live web using Bright Data's MCP (Model Context Protocol) server and scraping APIs, turning any LLM into a real-time web-aware system.

WORKSHOP Workshops Day 1 · Track 9 12:10pm-1:10pm

Research to Reality with Google DeepMind

Paige Bailey — AI Developer Relations Engineering Lead, Google DeepMind

1:15pm

SPONSOR Track 1 1:15pm-2:15pm

Let your agent cook: using skills to evaluate and improve your app

Ankur Duggal — Solutions Architect, Arize AI

SPONSOR Track 2 1:15pm-2:15pm *tentative*

Neo4J L&L

SPONSOR Workshops Day 1 · Track 3 1:15pm-2:15pm

2 hr deep dive on LLM Inference at Scale — Part 2 of 2

Harshul Jain — Senior Software Engineer - ML/AI, Audible · Tanmay Sah — Senior Quantitative Modeler, Zions Bancorporation

Most engineers using LLMs can call an API. Far fewer can explain why their model is slow, why it's running out of memory, or how the inference engines powering every major LLM API actually work. This workshop walks through the full inference stack — from how a transformer generates a single token to serving billions of tokens a day with vLLM, SGLang, TensorRT-LLM, Ray, and KServe/llm-d. 60% explanation with live demos, 40% hands-on exercises. Attendees leave with a running vLLM server they benchmarked themselves. Based on the open-source practitioners handbook being built live at github.com/harshuljain13/llm-inference-at-scale (NOTE: this is a 2 hour workshop that happens over lunch break - you should try to have lunch before or after if attending)

SPONSOR **Workshops Day 1 · Track 4** 1:15pm-2:15pm

Build the Right Thing: Product Engineering for Software Developers — Part 2

Kent C. Dodds — Software Engineer and Educator, EpicProduct.engineer

There is nothing quite as demoralizing as finishing a feature and realizing you built the wrong thing. The code is clean. The tests pass. The ticket is closed. And none of it matters. This is happening more often, not less. AI makes it faster and cheaper to implement, which means teams can now waste entire sprints on the wrong idea at unprecedented speed. The bottleneck is no longer "can we build it?" It is "should we build it?" and "are we sure we understand the problem?" This session is a condensed introduction to product engineering for builders: the skills that sit upstream and downstream of implementation. We will not try to cover everything a full-day workshop would. Instead, we will focus on the highest-leverage ideas you can apply on Monday. ### What we'll cover 1. Validate before you build Most wrong builds start with an idea that was never tested. You will learn to separate real user pain from solution-shaped requests, and practice discovery questions that surface past behavior instead of hypothetical enthusiasm. 2. Prioritize what deserves to exist Not every good idea should be built now. Especially in the AI era, "we could build this" is not a reason to build it. We will work through a practical prioritization lens, including the Kano model, to help you distinguish fundamentals from delighters from distractions before your team commits. 3. Own the feature, not just the PR Product engineering does not end at merge. You will leave with a clearer picture of end-to-end feature ownership: staying close to users, setting up simple feedback loops, and improving what you shipped instead of moving on to the next ticket. ### Format This is a 2–3 hour session with Kent C. Dodds. Expect focused teaching, real-world examples, and short interactive exercises and discussion. This is not a full simulation lab or a ticket-closing coding workshop. It is judgment practice for engineers who already know how to ship. ### Who this is for Software engineers (and technical builders generally) who: - Have shipped something polished that nobody wanted - Feel pressure to move fast with AI and want a better filter for what deserves to exist - Want stronger product instincts without becoming a PM - Care about owning outcomes, not just closing tasks Some software engineering experience is assumed. No particular stack is required. PMs and designers often find this valuable too. ### What you'll leave with - Discovery questions for ambiguous work - A prioritization lens you can use before committing to a build - A clearer model for feature ownership and post-ship feedback loops - Language for stakeholder conversations when requirements are unclear

SPONSOR **Workshops Day 1 · Track 5** 1:15pm-2:15pm

Build a Platform, Unleash an Agent on it.... and Watch it Burn!

Michael Forrester — AI Workforce Transformation, Accenture · **Whitney Lee** — Senior Technical Advocate, Datadog

You get a Kubernetes cluster with an Internal Developer Platform already running: ArgoCD for GitOps, Kyverno for admission control, Falco for runtime detection, Prometheus for observability. Everything is instrumented. Everything is enforced. You also get an AI agent with cluster access. Your job is to get the agent to break something. Deploy a non-compliant workload. Escalate privileges. Modify infrastructure outside Git. Exfiltrate data through an agent response. Some of you will fail because the governance stack catches it. Some of you will succeed because it doesn't. Afterward we regroup and map what got blocked, what slipped through, and why. The 80% that existing CNCF tools already govern becomes obvious. The 20% gap where agent-specific tooling is missing becomes undeniable. You leave with a concrete governance map and the exact list of failure modes your own platform probably isn't covering yet.

SPONSOR **Track 6** 1:15pm-2:15pm

SonarQube + OpenAI: Wiring Your Team for Agentic Development

Killian Carlsen-Phelan — Developer Content Engineer, Sonar

As AI agents take on increasingly complex development tasks, the critical challenge has shifted from generation to verification. A growing body of evidence suggests that as models grow more capable, failures become more frequent and more convincing, making cognitive surrender among human reviewers an acute risk. This talk introduces Sonar's Agent Centric Development Cycle (AC/DC), a three-stage continuous loop of Guide, Verify, and Solve, as the engineering discipline teams need to build now. Teams that embrace AC/DC guide agents within their organizational standards before they write a line of code, verify output in real-time, and solve issues automatically without manual triage. This session will also feature a live demo of the SonarQube OpenAI plugin, showing how a well-guided agent produces code that is faster to verify and cheaper to fix.

SPONSOR **Track 7** 1:15pm-2:15pm

How Reducto parsed the Epstein Files for the Viral JMail Project: The Secret Complexities of Document

Palak Agarwal — Developer Relations Lead, Reducto

Reducto powered the infrastructure behind Jmail, a fully searchable email interface with over 3.5 million scanned government pages built days after the Epstein files release. The site went viral overnight, racking up millions of views across news coverage and social media. In this workshop we'll break down how Reducto's Parse API handled everything from redacted PDFs to handwritten letters to dense financial tables at that scale, then walk through the same pipeline hands-on using the Reducto CLI and MCP. You'll leave with a working setup and a clear mental model for applying document parsing to your own projects.

SPONSOR **Workshops Day 1 · Track 8** 1:15pm-2:15pm

Turning My Obsidian Vault Into a Local AI Engineer

Filip Makraduli — Founding Member of Technical Staff, Superlinked

Personal knowledge bases are messy, but engineering agents need memory: decisions, docs, TODOs, old PRs, architecture notes, incident notes. This talk shows how I made an Obsidian vault usable by an agent using local-first retrieval and small-model inference. The point is not “chat with notes”; it is how to build durable, inspectable agent memory.

SPONSOR **Workshops Day 1 · Track 9** 1:15pm-2:15pm

Continuously improving agents with Langfuse

Lotte Verheyden — AI engineer and developer educator, Langfuse, Clickhouse · **Annabell Schäfer** — Growth Engineer, Clickhouse

Join us for a hands-on Langfuse workshop where we'll show you how to observe, debug, and improve your AI applications, step by step, using a real sample app. Bring your questions and discover how Langfuse can level up your specific use cases!

2:20pm

SPONSOR **Track 1** 2:20pm-4:20pm

From Vibes to Production: Evaluating and Shipping AI Agents That Work 201

Laurie Voss — Head of Developer Relations, Arize AI

Building an AI demo is easy. Knowing whether it actually works — and keeping it working in production — is the hard part. Most teams ship agents on vibes: they try a few prompts, the output looks good, and they push to production with no real way to measure quality or catch regressions. This hands-on workshop walks through the full lifecycle of shipping a real AI agent, using a working financial-analyst agent built on the Claude Agent SDK as the running example. You'll instrument it with tracing, do structured error analysis on its actual outputs, and build a layered evaluation suite — from cheap deterministic code checks to LLM-as-a-judge evaluators with custom rubrics. We'll cover the parts most tutorials skip: why agents fail in ways single LLM calls don't, the eval anti-patterns that quietly mislead you, and how to know whether you can even trust your judge (meta-evaluation). Finally, we'll close the loop: turning eval results into datasets and experiments, running evals online against production traffic, wiring them to monitors and alerts, and feeding failure explanations back to a coding agent to actually fix the underlying problems. You'll leave with a runnable notebook and a repeatable, evaluation-driven workflow you can apply to your own agents the next day.

SPONSOR Track 2 2:20pm-4:20pm

The Data Context Layer: Why Data Engineering Agents Need More Than Code and Databases

Yoni Michael — Co-Founder, typedef · Brandon Callender — Founding Engineer, typedef

Modern AI agents typically understand either code or databases. Code-focused agents reason over files, dependencies, and syntax, while database agents see tables, columns, and query results. This works for software development and basic analytics—but it breaks down for data engineering. In real data environments, agents fail because they lack context: an understanding of how data flows, what it represents, and why it behaves the way it does in production. Introducing the data context layer—a missing third layer that bridges code, data, and business semantics. Without it, agents hallucinate impact, suggest unsafe joins, and struggle with root cause analysis. This presentation will define the data context layer and showcase its use in practice, including end-to-end lineage from sources to reports; semantic metadata such as grain, measures, dimensions and business logic; runtime signals including job executions, failures, and performance patterns; and logical vs. physical modeling distinctions. Attendees will walk away with a greater understanding of: Why the code layer (dbt SQL, manifests, Git history) provides structure but misses grain, aggregation semantics, and join safety Why the data layer (warehouse tables, execution metrics, failures) shows what happened, but not why How the data context layer unifies lineage, semantic metadata, runtime behavior, and business rules The presentation will also cover architecture patterns for building and maintaining a data context layer, including why property graphs are well-suited for contextual reasoning and how agents can query context safely instead of relying on prompt stuffing.

SESSION Workshops Day 1 · Track 3 2:20pm-5:30pm

Special topics in Kernels, RL, Reward Hacking in Agents

Daniel Han — Co-founder, Unsloth

An advanced seminar (good prerequisites: Daniel's 2024 and 2025 hit AIE workshops, but all are welcome!) PLS WATCH: https://www.youtube.com/@aiDotEngineer/search?query=daniel%20han

SPONSOR Workshops Day 1 · Track 4 2:20pm-4:20pm

Burn your flags: How PayPal designs interactive CLI tools for agents

Mark Lummus — Product Lead, PayPal · Navinkumar Patil — Staff Software Engineer, PayPal

The common guidance for designing complex CLI tooling that agents can use is to add a 'non-interactive' mode, where a normally interactive & flow-based command can be executed in a single pass by feeding it a bunch of flags. This is necessary for deterministic automation, but agents aren't scripts; they aren't really constrained in the same way, and they benefit greatly from the same step-by-step contextual workflows that humans do. In this workshop, PayPal goes deep on techniques we've used in our upcoming `paypal` CLI that you can steal to make your complex CLI workflow tool agent-usable — without giving up the guardrails and guidance that interactive CLI tools provide.

SPONSOR Workshops Day 1 · Track 5 2:20pm-4:20pm

AI Security Engineer Foundations + Certificate

Micah Silverman — Director of Developer Relations, Snyk

In each of the two sessions, we cover 6 modules and participants receive a certificate of completion at the end. The modules are: OWASP Top 10 for LLM, Addressing Shadow AI, AI Threat Modeling, Securing Agents & MCP, Securing Vibe Coding, & AI Red Teaming

SPONSOR Track 6 2:20pm-4:20pm

Context Engineering in 2026: Compaction, Memory & Cost

Louis-François Bouchard — CTO & Co-Founder, Towards AI · Samridhi Vaid — Senior Machine Learning Engineer, Towards AI · Omar Solano — AI Engineer, Towards AI

Every long agent session eventually breaks: the assistant that swore it would "never push to main" does exactly that forty turns later. The model didn't get dumber — its context did. This workshop is about engineering the context window so that stops happening, shown with Towards AI's open-source AI tutor, which answers questions for students of our AI-engineering courses. Context engineering is deciding what the model sees on every single call — instructions, history, retrieved course content, memory, and tool outputs — and it's the line between a tutor that holds a coherent session and one that forgets the student's setup halfway through. We'll move in three stages, mirroring how the project actually went. The concepts: the two root problems (a finite window, a stateless model), the full compaction toolkit (truncation, trimming, tool-result clearing, summarization, and offloading to files — and when each actually helps), memory that survives across sessions, skills loaded on demand, and production-grade retrieval (chunking, metadata, course scoping, hybrid search, reranking, and evaluating). We'll cover the tutor's architecture, and the evaluation harness we used to measure every run on Gemini — tokens, cost, latency, and memory probes instead of vibe-checks. At real volume, even Gemini Flash got expensive, so we tested whether open and local models could match the quality for a fraction of the cost and match result quality. Everything is open-source and will be shared during the workshop.

SPONSOR Track 7 2:20pm-4:20pm

Vector Isn't Enough: Hybrid Search & Retrieval for AI Engineers

Jeff Vestal — Senior Principal AI Architect, Elastic

If you build RAG, you reached for vector search first. This lab is about everything that happens after you realize embeddings alone don't cut it in production. You'll write real queries — semantic, lexical, and hybrid — feel exactly where each one fails, and walk out with a production-grade retrieval pipeline and the judgment to know which technique to reach for when. What you'll actually do: 1. Dense vector search, and the mechanism behind it. Run semantic queries over a semantic_text field backed by Jina v5 embeddings — generated server-side, at query time, by the Elastic Inference Service (EIS). No embedding service to stand up, no client-side inference code. We open the hood on how query-time embedding actually works. 2. Break it. Throw adversarial queries at pure vector — exact error codes, version numbers (8.18 vs 9.0), precise config keys — and watch semantic similarity blur the exact match you needed. Then bring in BM25 lexical search to rescue it… and find the queries where keyword search whiffs. Each method is strongest exactly where the other is weakest. 3. Hybrid, properly. Fuse lexical + semantic with Elasticsearch retrievers. Learn the two fusion strategies that matter — Reciprocal Rank Fusion (RRF) and linear combination with score normalization — when to use each, and how to tune them. Optional: cross-encoder reranking with Jina Reranker v2. 4. Why this is the whole game for agents. Wire the hybrid retriever into a RAG flow and prove that retrieval quality, not the model, determines answer quality. Only synthesis truly needs the LLM - retrieve, rank, filter, and document-level security are database work done in milliseconds for a fraction of the cost. The contrarian takeaway: most of your RAG pipeline shouldn't be LLM calls at all.

SESSION Workshops Day 1 · Track 8 2:20pm-4:20pm

Build with Perception Agents

Emile Baizel — Amazon AGI Lab · Shruti Arora — Member of Technical Staff and Customer Engagement, Amazon AGI Lab

Human-agent collaboration is changing, becoming more visual. Models can perceive, point, and verify, but most agents still rely on us typing a paragraph to explain what we're looking at. Meet perception agents: computer use agents that see screens how you see screens. They understand, reason, and verify their own work. They let you point, draw, and describe, just as people collaborate in real life. We call this shared perception, and at AGI Lab we just open-sourced the first two primitives of our perception agent harness: visual verification and visual annotation. In this workshop, you'll get hands-on with both, build one sample use case end-to-end, then take the primitives back to your day-to-day in a mini hackathon. Best ideas win prizes.

SESSION Workshops Day 1 · Track 9 2:20pm-4:20pm

Hands-on AutoResearch: Cracking OpenAI's Parameter Golf

Zhengyao Jiang — CEO & Cofounder, Weco AI · Dixing Xu — Member of Technical Staff, Weco AI · Vayum Arora — Growth, Weco AI · Dhruv Srikanth — Founding Engineer, Weco AI

Heard about autoresearch, or tried it a few times in playground settings? This hands-on tutorial teaches you how to use autoresearch on one of the most serious challenges in ML this year: OpenAI's Parameter Golf. The challenge: train the best language model that fits in just 16MB. We entered our autoresearch agent this past spring, and it outperformed the field of over 1,000 participants. You'll learn how we approached it, then get to do it yourself: kick off an autoresearch agent, watch it improve a tiny language model's training script, steer it when progress stalls, and visualize your results. You'll leave with a working autoresearch setup you can point at your own code. compute kindly sponsored by Modal!

SPONSOR Track M 2:20pm-3:35pm

Observe, optimize and protect your hosted agents in Microsoft Foundry

Pamela Fox — Principal Cloud Advocate, Microsoft

Modern agents fail in ways traditional monitoring can’t catch. In this hands-on lab, learn how Microsoft Foundry Observability helps you move from prototype → production with context-specific evaluation suites (auto-generated evaluators + test datasets) wired into developer workflows via skills/MCP tooling for hosted agents. Then scale quality with continuous evaluation, trace-linked analysis, and adaptive red teaming—and walk away with a sandbox to explore additional features on your own.

4:30pm

SESSION **Workshops Day 1 · Track 1** 4:30pm-5:30pm

The Autonomous Computer: Full-stack Infrastructure for Computer Use Agents

Ang Li — CEO, Simular

Even the world's best computer-use agents cannot repeat their successes at the moment. Agents that write code — emitting structured selector-based actions instead of clicking pixels — break through that ceiling. We'll share two years of experience from Simular's production agent platform, the architectural decisions that mattered (refs over pixels, code as substrate, Simulang DSL), and a live demo: a 30-step unattended Windows workflow, side-by-side with a vision-only baseline. If you're shipping agents to real users, this is the playbook.

SESSION **Track 2** 4:30pm-5:30pm

The Dark Arts of Skill Engineering

Paul Bakaus — Founder, Renaissance Geek, Inc.

Most agent skills are a system prompt and a prayer. They produce safe, median output because that's what LLMs default to. After building 24 design skills across 9 AI platforms, I found the patterns that break through that ceiling, and they're rarely documented or discussed. Make your agents argue: spawn parallel sub-agents that independently evaluate the same work, then force their conflicting opinions into a single result. The output is bolder than any single agent would dare. Build mixture-of-expert skills that route to specialized sub-agents the way frontier models route to specialized networks. Give your skills memory through persistent context files that restore across sessions, so every invocation builds on the last. Wire up skill hooks that auto-activate after execution to validate, transform, or chain into the next skill. Exploit barely documented environment variables and shell expansion to make skills context-aware before they even run. Let's dig into the dark arts of skill engineering to craft ultra powerful skills.

WORKSHOP **Workshops Day 1 · Track 4** 4:30pm-5:30pm

Hill-climbing Skills: How to Improve Agents Without Touching the Model

Shubhankar Srivastava — Founding Sales Engineer, Browserbase

Agent Capability is now highly dependent on the markdown files read at runtime -- skills.This workshop treats skills as a first-class optimization surface. We borrow the concept of autoresearch (from Karpathy) and apply it to the skills your agents already read. You'll see how we at Browserbase did the same for browser agents, enabling our customers to scale the coverage of their browser agents while improving performance(2x faster runs) and optimizing for token spend(upto 10x cheaper).You'll leave with a working http://SKILL.md you generated through an auto-research loop, and a mental model for when skill optimization beats fine-tuning or prompt engineering.

WORKSHOP **Workshops Day 1 · Track 5** 4:30pm-5:30pm

Agent Auth

Bereket Habtemeskel — CEO, Better Auth · Paola Estefania — Staff Engineer, Better Auth

Better Auth has grown to 27k GitHub stars and over 1.5M weekly downloads, becoming a popular choice for developers who want to own their authentication stack. We recently introduced Agent Auth, a protocol designed to support autonomous and delegated agents operating services for an organization or a user. It allows agents to dynamically negotiate capabilities, manage access boundaries, and maintain secure authorization flows. This session will break down the protocol design and demonstrate it live, showing how agents can securely authenticate and operate with dynamic permissions.

WORKSHOP **Workshops Day 1 · Track 6** 4:30pm-5:30pm

The Prime Intellect Stack

Will Brown — Researcher, Prime Intellect

Deep dive into Prime Intellect's open-source ecosystem of post-training tools, including the verifiers and prime-rl libraries, as well as our Lab platform for self-serve training and inference.

WORKSHOP **Workshops Day 1 · Track 7** 4:30pm-5:30pm

Lifestyles of the AI-Native: Voice-coding, agent skills, hooks and scheduled tasks

Nick Nisi — Developer Experience Engineer, WorkOS · **Zack Proser** — AI Engineer, Applied AI, WorkOS

Most engineers are bolting AI onto a workflow that was designed for a pre-AI world. The result is a faster version of the same grind. This talk is about the other path: rebuilding the daily practice of software engineering from the ground up, around what agents are actually good at. Two senior practitioners from WorkOS will walk through how we actually work now as AI-native engineers — not in the aspirational sense, but the literal one. We think out loud and voice-code instead of typing our way to clarity. We package recurring expertise into agent skills so we're not re-explaining context every session. We wire up hooks that fire on the events we care about, and hand off scheduled tasks to agents that run overnight, while we're away from the keyboard, or otherwise off the clock. The throughline is intentional design: deciding what a human should hold onto and what should be delegated, then building the machinery to make that real. Because there are two of us, you'll see more than one set of habits — where our setups converge on the same patterns, and where they diverge based on how each of us thinks and works. The pitch isn't "do more." It's that an AI-native setup, designed deliberately, buys back attention and protects you from the burnout that comes from treating agents as a turbocharger for an old loop. Attendees will leave with a concrete mental model for voice-driven development, a pattern for authoring reusable agent skills, and working examples of hooks and scheduled automations they can adapt the same week.

WORKSHOP **Workshops Day 1 · Track 8** 4:30pm-5:30pm

The Art and Science of Loopcraft with Pi (and friends)

Joel Hooks — Co-founder; software developer and developer-education entrepreneur, badass.dev / egghead.io

This workshop helps agentic coding practitioners stop treating agents like pretend coworkers and start designing reliable, compounding loops. Using Pi as the concrete demo surface, Joel Hooks will show how loop state, handoffs, review, memory, and operator control become visible, while keeping the ideas portable to Claude, Codex, Cursor, and similar coding agents. Practitioners should leave able to identify loops inside their agent workflows, diagnose when failures need gates/evidence versus orchestration/memory/leverage, and understand how model-shaped lifecycles differ from traditional human SDLC rituals.

WORKSHOP **Workshops Day 1 · Track 9** 4:30pm-5:30pm

Evolution of agentic surfaces

Gagan Bhat — Member of Technical Staff, Anthropic · **Isabella Kai He** — Member of Technical Staff, Anthropic

Getting an agent into production takes more than a good prompt: it needs somewhere to run code, credentials it can't leak, sessions that survive interruption, and infrastructure that scales. This talk traces how Anthropic's agentic surfaces evolved from the raw API to Claude Managed Agents, and what our Applied AI team has learned about harness design along the way.

5:00pm

WORKSHOP **Expo Stage 2 · Expo Stage 2 NW** 5:00pm-6:00pm

Human Connection in the Age of AI

Joyce Zhang — Dating Coach for Tech Founders, Joyce Consulting Group · **Carole Robin, Ph.D.** — Co-Founder, Leaders in Tech

Building AI safely requires both technical skills and interpersonal skills. A live demo of connection tools from Stanford's "Touchy Feely" course, then hands-on practice. Co-hosted with Leaders in Tech.

6:00pm

SESSION **Expo Stage 3 · Expo Stage 3 SW** 6:00pm-6:15pm

Expo Welcome Speech

Sonar · **Extend AI** — Extend AI

6:15pm

SESSION **Expo Stage 3 · Expo Stage 3 SW** 6:15pm-7:15pm

Runway AI Film Festival

Runway's annual AI Festival — a celebration of creatives experimenting at the forefront of art and technology across film, design, new media, fashion, advertising, and gaming, with a screening of finalist AI films. https://aif.runwayml.com/

Day 2 — Session Day 1 Tuesday, June 30, 2026

contents ↑

9:00am

KEYNOTE **Software Factories · Main Stage** 9:00am-9:05am

The Highest Loop

swyx — Curator, Latent Space / AI Engineer

We celebrate the third birthday of the AI Engineer post.

9:05am

KEYNOTE **Software Factories · Main Stage** 9:05am-9:25am

On AI and Knowledge

Pablo Castro — Distinguished Engineer and CVP, Microsoft

9:25am

KEYNOTE **Software Factories · Main Stage** 9:25am-9:45am

The Golden Age of AI Engineering

Alexander Embiricos — Head of Enterprise Product, OpenAI · Romain Huet — Head of Developer Experience, OpenAI

TBD

9:45am

KEYNOTE **Software Factories · Main Stage** 9:45am-10:05am

GLM-5.2: Frontier Intelligence, Open Weights.

Zixuan Li — Head of Z.ai, Z.ai

10:05am

KEYNOTE **Software Factories · Main Stage** 10:05am-10:25am

Thom Wolf keynote

Thom Wolf — Co-founder and CSO, Hugging Face · Olive Song — RL Lead, MiniMax

10:25am

KEYNOTE **Software Factories · Main Stage** 10:25am-10:30am

Security Track intro

Manoj Nair — CTO & Chief Innovation Officer, Snyk

10:45am

SESSION **Software Factories · Main Stage** 10:45am-11:05am

Getting the most out of Codex

Jason Liu — Developer Experience, OpenAI, OpenAI

SESSION **Claws & Personal Agents · Track 1** 10:45am-11:05am

Security Firewall for Agents

Ryan Dahl — CEO, Deno

Why personal agents that run untrusted LLM code need a sandboxed OS/runtime model, not just a compute sandbox.

SPONSOR **Vision & OCR · Track 2** 10:45am-11:05am

The State of Vision

Joseph Nelson — Cofounder, CEO, Roboflow

SESSION **Search & Retrieval · Track 3** 10:45am-11:05am

Pinecone 2.0

Edo Liberty — Founder and Chief Scientist, Pinecone

Autonomous agents are smart but don’t know your business or your objectives. That’s why most agents in the enterprise remain stuck in retrieval loops, burning millions of tokens on processing raw documents A shift from traditional retrieval systems + agents (aka RAG) to purpose-built knowledge engines is underway. I'll talk about why moving reasoning upstream and compiling raw enterprise data into specialized, task-specific context artifacts is critical to unlocking reliable agentic workflows. And I'll show you how offloading knowledge management to a dedicated layer enables engineering teams to achieve up to a 90% reduction in token consumption while drastically improving task completion rates, speed, and accuracy.

SESSION **Workshops Day 2 · Track 4** 10:45am-11:05am

Claude Managed Agents Workshop (Part 1)

Priyanka Phatak — Member of Technical Staff, Anthropic · Gabriel Cemaj — Member of the Technical Staff, Anthropic

Build an agent with Claude Managed Agents

SPONSOR **Security · Track 5** 10:45am-11:05am

Through the AI Fog: The architectural decision the next 24 months of agentic security depends on.

Manoj Nair — CTO & Chief Innovation Officer, Snyk

SESSION **Voice & Realtime AI · Track 6** 10:45am-11:05am

The New Primitives: Building AI-Native Software

Kwindla Kramer — CEO, Daily

In the future, every piece of software with a human-facing surface will be built from new, LLM-centric primitives. (Just like every piece of software today has networking, threads/async routines, UI on top of some flavor of Model/View/Controller abstractions, etc.) We're just starting to invent these new primitives. The list, though, will definitely include: 1. Subagents - multiple inference loops, multiple models, async tool calls 2. Very long context - memory + episodic human interactions over a long period of time, structured data input (not just output), progressive skills/context loading, graceful compaction & summarization 3. dynamic user interface generation / user interfaces driven by LLM inference 4. conversational voice input

SESSION **LLM Recsys · Track 7** 10:45am-11:05am

Tokens In, Engagement Out: Training LLM-Recommenders

Devansh Tandon — Principal Product Manager, Meta

SESSION **Forward Deployed Engineering · Track 8** 10:45am-11:05am

How Forward Deployed Engineering is done at Factory

Eno Reyes — CTO & Co-Founder, Factory

SESSION **Data Quality · Track 9** 10:45am-11:05am

Data Quality is the Compute Multiplier

Ari Morcos — Co-founder, CEO, DatologyAI

Better data quality is the highest-leverage and most underinvested part of building a model: it produces a better model for the same compute, whether you're mid-training on an open base or pre-training from scratch. This session is a practical look at data curation, covering what data quality actually means, the stages of a modern curation pipeline (cleaning, filtering, deduplication, synthetic data generation, algorithmic mixing, and multi-stage composition), and which steps matter most in practice. It draws on DatologyAI's frontier data research and customer results, including Thomson Reuters' mid-training gains on proprietary legal domain data and Arcee's Trinity model reaching the open frontier on public data alone. You'll leave with a concrete sense of where better data quality pays off and how data curation is shaping the future of model training.

SPONSOR Track M 10:45am-11:05am

Build agents fast with GitHub Copilot (from idea to working app)

Idan Gazit — Head of GitHub Next, GitHub

See how developers go from prompt to a working agent using GitHub Copilot and real workflows. We'll walk through generating code, iterating quickly, and keeping velocity inside your existing dev loop.

SESSION Agentic Commerce · Leadership 1 10:45am-11:05am

Inside the AI economy: What Stripe's data reveals

Nilofer Rajpurkar — Product Lead, Agent and Developer Experience, Stripe

Stripe powers 78% of the Forbes AI 50, giving Stripe index-level visibility into the AI economy. AI companies are growing faster, selling globally by default, and monetizing earlier. See the data behind the growth: how AI has collapsed the cost of launching, how the fastest-growing companies are adapting their pricing, and the role agents are starting to play in commerce.

SESSION Claws & Personal Agents · Leadership 2 10:45am-11:05am

Governance Is the Real Bottleneck to AI ROI

David Hsu — CEO, Retool

As AI systems move from generating content to taking Claw-based agents action inside production systems, governance (not model quality) becomes the limiting factor. David will break down why visibility, guardrails, approvals, and rollback matter more than raw intelligence, and how companies can enable AI adoption without creating security and compliance disasters.

SESSION Expo Stage NE · Expo Stage 1 NE 10:45am-11:05am

Every AI company is accidentally building a bank.

Dor Sasson — Stigg

You're logging usage, billing later, hoping agents behave. They don't. Here's the architecture that fixes it before the invoice hits.

SESSION Expo Stage 2 NW 10:45am-11:05am

The Enterprise Agentic Gap: When Developer-Level AI Tools Hit Millions of Lines

Dan Adler — Sourcegraph

Agentic coding tools have transformed individual developer workflows but owning a large codebase with millions of interdependent lines across multiple code hosts is a different problem entirely. Off-the-shelf AI coding tools weren't built for it, and at scale, they break down in ways that aren't obvious until you're already in trouble. This talk covers the failure modes you'll hit when applying developer-level agentic tools to enterprise-scale migrations, and how Sourcegraph's agentic migrations solution was built to solve what others couldn't.

SESSION Expo Stage 3 SW 10:45am-11:05am

How PayPal Enterprise Payments handles agent-initiated payments across ChatGPT and Google AI Mode

Sam Parsons — Senior Staff Software Engineer and Tech Lead, PayPal Braintree

PayPal Enterprise Payments has shipped integrations across the major agentic surfaces in the last six months each with human-in-the-loop confirmation and full transaction attribution back to the originating AI platform. We'll tour all three paths: ACP for ChatGPT apps (delegated payment tokens via complete_checkout, allowance validation, facilitator_details attribution), UCP with Google Pay for Google AI Mode (server-side tokenizationSpecification, parsing androidPayCards for the single-use token), and a preview of MCP Apps inline checkout, where the payment surface renders in-chat and card data never enters the LLM context. For each path we'll cover where PayPal Enterprise Payments fits, what the shopper and merchant each see, and the tradeoffs between them. You leave with working code and the docs to evaluate which path fits your stack.

SESSION Expo Stage 4 SE 10:45am-11:05am

Agentic Search for Coding Agents

Jakub Hojsan — Exa

11:10am

SESSION **Software Factories · Main Stage** 11:10am-11:30am

Rise of the Software Factory

Tereza Tížková — Growth, Factory

The Stanford HAI 2024 AI Index reports a 30x productivity gap between AI leaders and laggards. The differentiator is not company culture, prompting technique or model selection, but the infrastructure. Organizations capturing outsized value from AI agents have machine-readable codebases, deterministic internal APIs, CI/CD pipelines with agent-addressable hooks, and permission models granular enough to scope exactly what an agent can touch. I believe the “agents as employees” framing is most useful if you operationalize it. An employee has persistent identity, episodic and semantic memory, scoped permissions that don’t get renegotiated every task, an audit trail, and a defined escalation path when things go wrong. Persistent computer use (with a stable execution environment that survives across steps) was the real inflection point that is making this possible. Some interesting production problems remain under-explored. How do you give an agent persistent identity across pull requests? How do you recover from partial failure mid-task without discarding completed work? How do you enforce code ownership policies when the author is a model? How do you bound token spend when pipelines spin up sub-agents recursively? This talk defines agent readiness as a concrete infrastructure checklist: structured codebases, deterministic APIs, per-agent scoped credentials, atomic and idempotent operations, structured execution traces, and explicit thresholds for when the agent stops and a human takes over. It presents research results in practice, and what are the steps organizations need to take to be fully agent-ready.

SESSION **Claws & Personal Agents · Track 1** 11:10am-11:30am

Your Agent Didn’t Fail. Your Harness Did.

Vinoth Govindarajan — Member of Technical Staff, OpenAI

AI agents do not fail only because the model is wrong. Many production failures happen in the harness around the model: state is not persisted, two runs mutate the same session, a tool call never returns, an approval loses scope, or an internal success never becomes user-visible proof. This talk uses OpenClaw as a public case study to examine real harness failure modes and extract a reusable production model for AI engineers. We will look at how events enter an agent system, how session state is rehydrated, why single-writer lanes and throttles matter, and why tool execution needs scoped approvals and auditable receipts. The core idea is simple: a model proposes, the harness commits, and the receipt proves it. Attendees will leave with a practical 'run receipt' audit they can apply to their own agents: what woke it up, which state did it inherit, what authority did it use, what executed, and what evidence survived.

SPONSOR **Vision & OCR · Track 2** 11:10am-11:30am

Building the Document Context Layer for AI Agents

Jerry Liu — CEO, LlamaIndex

AI agents are the new knowledge workers, but knowledge work depends on unstructured enterprise context. ~90% of that data lives in the form of document containers - from human-native (PDFs, Word, Pptx) to emerging agent-native formats (HTML, MD). Doing RAG in 2026 involves generalized agent harnesses with tools, MCPs, and skills. In this world, every company building agents needs a Document Context Layer, the bridge between their unstructured docs and the agents trying to reason over them. This talk covers what that layer looks like in practice: from document understanding, retrieval, and workflows, to areas yet to be explored — agent-native formats, versioning, editing, permissions, and longer-running agents.

SESSION **Search & Retrieval · Track 3** 11:10am-11:30am

The unreasonable effectiveness of BM25 for agentic search

Jo Kristian Bergum — CEO, Hornet.dev

GPT-5 is shockingly good at search, and that changes the "BM25 as a baseline" story. Using GPT-5 search trajectories from BrowseComp-Plus, I'll show how default BM25 parameters and evaluation harnesses can make lexical retrieval look weak, while real agent queries often play directly to BM25's strengths. Much like grep became a core retrieval primitive for coding agents, BM25 is re-emerging as a powerful primitive for agentic search.

SESSION **Workshops Day 2 · Track 4** 11:10am-11:30am

Claude Managed Agents workshop (Part 2)

Priyanka Phatak — Member of Technical Staff, Anthropic · **Gabriel Cemaj** — Member of the Technical Staff, Anthropic

Build an agent with Claude Managed Agents

SPONSOR Security · Track 5 11:10am-11:30am

Your LLM Stack Is a 2008 Database With Better Marketing: Why ML Security Is Dominated by Misconfiguration, Not Missing Features

Lovina Dmello — Senior Software Developer, NVIDIA

ShadowRay exposed over a billion dollars of data through a missing authentication check. It wasn't a zero-day. It wasn't a clever new attack class. It was a default config someone never flipped off. That story is not the exception in production ML, it's the rule. We synthesized 139 peer-reviewed papers on production ML security across access control, runtime security, infrastructure, and operations. Five findings stood out, and one of them upends how most teams think about ML security: - Misconfiguration, not missing features, is the dominant failure mode. The mechanisms exist. Teams aren't using them, or are using them wrong. - Adversarial defenses impose 15–30% inference overhead, which is why almost no production system actually runs them. - ML-specific security tooling lags general DevOps tooling by years. - Security, data-science, and ops teams operate in expertise silos that create persistent gaps no single team can see. - LLM and multi-tenant GPU threats are evolving faster than defenses (prompt injection, RAG poisoning, GPU side channels). This talk walks through the four-pillar defense-in-depth framework, the six-category threat taxonomy that maps each attack to its primary and secondary defenses, and a four-level security maturity model that matches overhead budgets to deployment contexts. You leave knowing where your stack actually sits and which 3 misconfigurations account for most of the risk.

SESSION Voice & Realtime AI · Track 6 11:10am-11:30am

Speech-to-Speech Model Research at Google DeepMind

Valeria Wu Fon — Product Manager, Google DeepMind · Tom Ouyang — Principal Engineer, Google DeepMind

Most voice interfaces today are built as a 3-way cascade system (ASR/LLM/TTS). While functional, this cascaded approach introduces latency bottlenecks, strips away non-verbal nuance, and limits emotion-aware, multi-turn dialogue. Today, we are witnessing a profound shift toward native speech-to-speech models that process audio natively from end to end. In this session, we'll explore the exciting paradigm at Google DeepMind to train speech-to-speech models for real-time voice agents. We will cover the high-level product and research challenges of building voice agents that feel truly conversational, optimizing for fluid turn-taking and low latency while maintaining enterprise-grade intelligence.

SESSION LLM Recsys · Track 7 11:10am-11:30am

Spotify LLM Recsys

Jacqueline Wood — Staff Machine Learning Engineer, Spotify · Yves Raimond — SVP/GM, AI & Personalization, Spotify

SESSION Forward Deployed Engineering · Track 8 11:10am-11:30am

How Forward Deployed Engineering is done at Cursor

Pauline Brunet — VP, Forward Deployed Engineering, Cursor

SESSION Data Quality · Track 9 11:10am-11:30am

The Messy Reality of Scale: Synthetic Data and Pre-Training at Poolside

Robert McHardy — Pre-training Lead, poolside · Marah Abdin — Team Lead - Synthetic Data, poolside

TBD — focus on data quality considerations for LLM pretraining and code generation.

SESSION AI-Native Enterprises · Leadership 1 11:10am-11:30am

Building the engine while flying the plane — launching the Figma MCP server

Jesse Lumarie — Software Engineer, Figma

What does it actually take to go from a vague idea to a production-ready AI system that people depend on? In this talk, I'll walk through the real story of building Figma's MCP server as a founding engineer whilst the MCP spec evolved—starting from early prototypes, through dead ends and architectural pivots, to launching both the initial product, creating new tools and eventually a fully remote server.

SESSION **AI Architects: Show my Workflow · Leadership 2** 11:10am-11:30am

Your Agent Evolved. Your Evals Didn't.

Ameya Bhatawdekar — VP, Field CTO, Braintrust

Knowing which generation your agent is in, which failure modes your current evals are blind to, and what to build next is the difference between shipping with confidence and flying blind. Agent architectures have evolved through six generations; prompt, chain, ReAct loop, workflow graph, modern agent loop, AI harness. And each one quietly breaks the eval strategy of the generation before it. A prompt-quality rubric won't catch a bad tool call; a trace scorer won't catch memory poisoning. Using a single SRE incident response agent threaded through every generation, this talk shows exactly where each architecture outgrows its evals and what you need to close the gap.

SESSION **Expo Stage 1 NE** 11:10am-11:30am

Give your coding agents the power of turbogrep!

Owen Halpert — GTM, turbopuffer

Coding agents can grep the filesystem, but sometimes semantic search is more useful for finding the right files, especially on large codebases. Claude Code and Codex, unlike Cursor, do not use semantic search for code retrieval. There are good reasons for this, but Cursor has consistently demonstrated that semantic retrieval can materially improve code search to improve answer accuracy, increase code retention, and reduce token usage. In this session, we'll share a coding agent plugin for semantic codebase search alongside other modalities (BM25, regex/globbing/grep, filtering), and demonstrate how an agent can choose the right tool for the job. We'll share benchmark-style results that compare answer quality and token consumption with and without semantic retrieval across a small set of representative tasks.

SESSION **Expo Stage 2 · Expo Stage 2 NW** 11:10am-11:30am

Actionable Knowledge For Agents With Context Graphs

Will Lyon — Product Manager, Neo4j

SESSION **Expo Stage 3 SW** 11:10am-11:30am

Frontier models for the hard parts, open weights for the rest

Kimchi is an open-source coding agent that orchestrates multiple AI models—including open-weight models like Kimi K2.7 and MiniMax M3 alongside commercial frontier models—to intelligently route each task to the best model for the job. Powered by Ferment, Kimchi evaluates every step, automatically reworking or escalating tasks when needed to maintain quality while minimizing the use of expensive frontier models. The result is high-quality code generation at approximately 2.5x lower cost than relying on commercial models alone—all with the transparency and flexibility of open source.

SESSION **Expo Stage 4 SE** 11:10am-11:30am

Agents, codebases, and teams: what it actually takes to ship together

Aditya Khandelwal — MTS, Amazon AGI Lab

Using a coding agent solo is one thing. Getting a whole team to trust agent-written code, agent-run reviews, and long-running agent work is another. That's where most teams stall. This talk is about what it actually takes to get there: how to shape a codebase so agents can work in it safely, how to earn a skeptical team's trust instead of mandating it, and the failure modes that only show up once agents are part of the daily workflow.

11:40am

SESSION **Software Factories · Main Stage** 11:40am-12:00pm

Orchestras, not Factories

Charlie Holtz — CEO, Conductor

Everything is Conductor now! I want to tell the story of how we came up with the original interface, what I think everyone (including us) is getting wrong and what's coming next.

SESSION **Claws & Personal Agents · Track 1** 11:40am-12:00pm

Everyone Gets A Software Company

Benjamin Guo — Cofounder, Zo Computer · **Rob Cheung** — Co-founder, Zo Computer

SPONSOR **Vision & OCR · Track 2** 11:40am-12:00pm

Skill issue: stop deploying vision language models, use them with Skills to build e2e vision apps on edge

Merve Noyan — MLE, Hugging Face

With the boom of vision language models barrier of entry to build vision apps are much lower so developers tend to use them right away. However, these models are very large and inefficient in production. In this talk, I will go through combining vision language models with Skills to build end-to-end vision apps from training to deployment using HF Skills, on top of showing the state-of-the-art in small computer vision/multimodal models.

SESSION **Search & Retrieval · Track 3** 11:40am-12:00pm

The Search Engine for the Agentic Web

Will Bryk — CEO, Exa

Every search API claiming to be "built for AI" is actually Google with a wrapper. That's a problem, because AI agents don't search like humans. A human waits 1 second for a result. An agent making 50 sequential searches at 1 second each creates a 50-second lag. That kills the product. And latency is just one dimension: agents need semantic precision, structured outputs, and a range that spans sub-200ms real-time retrieval all the way to multi-step deep research. No human-facing search engine was ever designed to do that. Will Bryk, CEO of Exa, shares what he learned building a search engine from scratch for AI. He'll cover the architectural decisions behind Exa's latency spectrum, what real usage patterns look like across companies like Cursor, Notion, HubSpot, and Lovable, and why the benchmarks the field relies on today are dangerously inadequate for evaluating agentic search. The bigger argument: search is becoming the most critical primitive in AI infrastructure, and almost no one is building it right.

SESSION **Workshops Day 2 · Track 4** 11:40am-12:00pm

Claude Managed Agents workshop (Part 3)

Priyanka Phatak — Member of Technical Staff, Anthropic · Gabriel Cemaj — Member of the Technical Staff, Anthropic

Build an agent with Claude Managed Agents

SPONSOR **Security · Track 5** 11:40am-12:00pm

We Gave an Agent Production Code Access and Then Tried to Sleep at Night

Moritz Johner — Staff Engineer, Form3

We let an agent touch production code to fix CVEs. That is either automation or a supply chain incident, depending on how honest your architecture is. PatchPilot started simple: find vulnerable dependencies, patch them, open a PR, let CI prove the fix, move on. Then reality showed up. The agent needed repository access, CI logs, credentials, and a Docker socket. Without that, it was useless. With it, every security reviewer in the room had a point. This is the production case study: what we gave the agent, what we refused, what infosec pushed back on, and where they were right. We will cover scoped permissions, constrained PRs, audit trails, approval gates, CI evidence, credential boundaries, and the gap between "it generated a patch" and "we can defend this change." Agentic remediation is not just developer productivity. It is a new participant in your software supply chain.

SESSION **Voice & Realtime AI · Track 6** 11:40am-12:00pm

Voice Agents Can Just Do Things

Charlie Guo — Developer Experience Engineer, OpenAI

Too many voice AI integrations still treat speech as fancier chat: audio in, audio out. But we're at a point where speech can be a control plane for software, and most developers are unaware that voice has become a capability overhang. Current realtime models can understand intent, call tools, speak while work is underway, recover from corrections, and decide what the user actually needs to hear. As a result, we're seeing three practical patterns emerge: voice-to-action, systems-to-voice, and voice-to-voice. We'll show how each pattern changes the architecture, where Realtime 2's reasoning and tool-calling matter, and why chained STT / LLM / TTS systems start to break down as the interaction patterns become richer.

SESSION **LLM Recsys · Track 7** 11:40am-12:00pm

LLM Recsys at DoorDash

Raghav Saboo — Staff Machine Learning Engineer, Tech Lead Search & Personalization, DoorDash, Inc.

SESSION **Forward Deployed Engineering · Track 8** 11:40am-12:00pm

AI tools for Forward Deployed Engineering

Vasuman Moza — Founder & CEO, Varick Agents

SESSION Data Quality · Track 9 11:40am-12:00pm

Rethinking Environments for Long Horizon Work

Rayan Garg — CEO, Theta Software

As autonomous agents push towards longer-horizon tasks, a number of challenges emerge in measuring and improving frontier model capabilities. In this talk, we discuss how long-horizon tasks are defined and measured, how RL environments and verifiers have to scale for more complex and open-ended tasks, and how we navigate these problems at Theta.

SPONSOR Track M 11:40am-12:00pm

Use Copilot across CLI, dev, and cloud workflows to move faster end-to-end

Pamela Fox — Principal Cloud Advocate, Microsoft

Copilot isn't just for writing code. Learn how to use it across CLI and cloud workflows to scaffold apps, debug faster, and automate repetitive steps across your entire dev lifecycle.

SESSION AI-Native Enterprises · Leadership 1 11:40am-12:00pm

Agentic SDLC at Uber: Building Blocks for Uber's Software Factory

Uday Kiran Medisetty — Distinguished Engineer, Uber · Adam Huda — Sr Engineering Leader for AI Dev Tools, Uber

99% of Uber engineers are using AI every month, 70% of PRs are attributed to AI, and 15% of PRs are now done entirely by autonomous agents. In this session, we go behind the scenes to show you exactly what it takes to get there — starting with the foundational building blocks: the model gateway, MCP infrastructure, agent skills, knowledge systems, and cloud developer environments that make agentic engineering possible at scale. Then, once those foundations are in place, we show you how to assemble them into a fully agentic SDLC. We'll walk through every stage — from research and spec writing, to autonomous code generation, to verifying and validating that code before it ships, to monitoring what happens after it lands, and continuously improving it over time. With tooling example demos throughout. Whether you're just starting your agentic journey or already running agents in production, you'll leave with a concrete blueprint for what this looks like end to end.

SESSION AI Architects: Show my Workflow · Leadership 2 11:40am-12:00pm

The Last Human Code Review: Building Trust in AI-Generated Code

Itamar Friedman — Co-Founder & CEO, Qodo

By the end of 2026, asking a human to review every pull request will be as optional as asking one to run every unit test manually. The tooling will be ready. The question is whether organizations are. In this talk, Itamar Friedman, CEO of Qodo, explains why we are approaching the end of line-by-line human code review as a default requirement and explores what has to be true for teams to get there. The barrier was never agentic AI capability. It was trust. And trust in automated review does not come from smarter models or faster feedback loops. It comes from systems that provide a trustworthy, concise and personalized proof-of-validation report. These systems are built on how engineering teams at specific organizations write their code: their own rules and standards, their PR history, their architecture decisions, their tribal knowledge that lives in comments and conversations and gets lost when engineers leave. Itamar will walk through the shift from PR-by-PR review toward continuous, context-based code review and governance, and share a practical approach to making human code review optional. If your team is shipping AI-generated code faster than humans can read it, join us for the discussion.

SESSION Expo Stage 2 NW 11:40am-12:00pm

Agentic vs. Vector Search: An Eval-Driven Approach to Coding Agent Performance

Jess Wang

Evals let you replace gut feelings with quantifiable decisions. This talk breaks the basic concepts of evals, including the four core components: datasets, tasks, scoring, and experiments. Then, to solidify the concept, we'll walk through a real eval comparing agentic search versus vector search for coding agents. We'll also cover practical challenges like tracing Claude Code subprocess calls and why a single eval run is never enough. You'll leave with a concrete framework for building evals that actually inform your ship decisions.

SESSION Expo Stage 3 SW 11:40am-12:00pm

Agents Don't Have Coworkers, They Have Hostages

Gabriel Martinez — Engineering Manager, G2i

Modern coding workflows are rife with vibe slop. As organizations scale, proper roles and governance systems must be well-defined to ensure a high standard of quality. How do world-class teams scale quality in a world full of slop?

SESSION **Expo Stage 4 SE** 11:40am-12:00pm

Would your AI agent get the job? A performance review framework for enterprise agents

Andreea Pleşea — Co-Founder and COO, Druid AI · **Dan Bălăceanu** — Chief Product Officer and Co-Founder, DRUID AI

There are dozens of ways to build an enterprise AI agent: agentic frameworks, direct LLM APIs, conversational AI platforms, vertical SaaS. They all claim to do the job. But how do you actually compare them on the same task, with the same data, against the same KPIs? This session presents a vendor-agnostic evaluation framework that treats AI agents the way enterprises treat new hires: set the role, define success criteria, run candidates through identical scenarios, and measure outcomes. The architecture uses any LLM to track positive and negative drift across agents against weighted goals, monitoring everything from hallucination rates and token consumption to user sentiment and conversation quality. Inputs are standardized. Outputs are both quantitative (accuracy, cost, hours saved) and qualitative (tone, clarity). The methodology supports continuous evaluation, not just pre-deployment benchmarks, but ongoing performance reviews that can compare agent work against human baselines. Walk away with a concrete, repeatable process for answering the only question that matters: which agent actually does the job?

12:05pm

SESSION **Software Factories · Main Stage** 12:05pm-12:25pm

What we learned by analyzing 1M AI-generated PRs

Daksh Gupta — co-founder/CEO, Greptile

We analyzed >1M end-to-end AI generated PRs reviewed by Greptile to understand what types of bugs they tend to create and some strategies on mitigating them. For instance, did you know that Claude Code is nearly 3X more likely than Codex to introduce auth bypass vulnerabilities?

SESSION **Claws & Personal Agents · Track 1** 12:05pm-12:25pm

Tethered: Our Agents Are Us

Shu Fang — Software Engineer, Two Sigma Investments

Personal AI assistants have dominated the zeitgeist of late with the advent of OpenClaw. However, letting an agent run as you remotely with access to your full suite of tools terrifies us in the technical community. How then did we get comfortable with enabling this functionality firmwide at a 70 billion dollar hedge fund? This talk will go over the underlying architecture, controls, and UX that enables every employee at Two Sigma to have a remote AI Assistant that acts as us in full. With access to our entire set of internal tools. Notably, this isn't just for engineers. Every single employee gets a remote agent that assumes their identity and can take broad action on their behalf. And we're ok with it.

SPONSOR **Vision & OCR · Track 2** 12:05pm-12:25pm

Modality Misalignment and Originality Attribution in Short-Form Video: A Multi-Agent Approach at Platform Scale

Aditya Gautam — Machine Learning Lead, Meta

Short-form video presents a class of content understanding problems that are qualitatively different from text or single-modality media. Audio, visual, and text signals within the same piece of content frequently diverge, sometimes incidentally and sometimes deliberately, creating a modality misalignment that defeats systems designed around any single signal. At the same time, the resharing dynamics of short-form video platforms create originality attribution chains that degrade quickly and are poorly captured by metadata alone. Addressing both problems at platform scale, reliably and under real latency and cost constraints, is the challenge this talk is built around. The core of the talk is the multi-agent architecture developed to address this, published at ACM WSDM 2025, and the reasoning behind its design. Each agent in the system is specialized for a distinct aspect of the problem: understanding what a piece of content is actually communicating across modalities, identifying where those modalities diverge meaningfully, and tracing originality through the resharing graph to surface attribution that platform metadata misses. We will cover the design principles behind this decomposition, the tradeoffs between specialization and complexity, the evaluation framework built to measure performance in a setting where ground truth is genuinely ambiguous, and the practical optimizations that made the system viable at scale. We will also be honest about the limitations: where the multi-agent approach added overhead that simpler baselines handled adequately, and what the boundaries of the system's reliability actually look like in production conditions. The broader takeaway is a set of principles for approaching multimodal content understanding problems where the signals are misaligned by nature rather than by exception. Attendees will leave with a framework for thinking about agent decomposition across a complex multimodal problem, a grounded understanding of how originality attribution degrades at scale and what it takes to recover it, and practical lessons about building evaluation and optimization pipelines for systems where the problem itself resists clean benchmarking.

SESSION Search & Retrieval · Track 3 12:05pm-12:25pm

Rebuilding the web for agents

Liad Yosef — Co-creator, MCP Apps

AI apps are the new browsers. And the web is not ready. For thirty years we built the web for human eyes, benchmarked by tools like Lighthouse: humans measuring human behavior. That era is ending. Bot traffic has overtaken human traffic, and we can't hand-write a benchmark for what comes next - every best practice goes stale the moment models improve. Your next customer isn't a human with a credit card - it's an agent with a protocol, and it would rather not see your interface at all. That shift moves the UX question from how a human experiences your product to how an agent does, and how a human experiences that agent. Already, some services report their MCP traffic outpacing their web UI. The agent is rapidly becoming the main surface, and it always takes the path of least friction. Claude Code might consistently prefer PostHog over Mixpanel simply because PostHog *has the better agentic surface* - and Mixpanel loses customers without a human ever weighing in. Meanwhile the agentic web protocol stack keeps multiplying, a new one seemingly every week. The harder problem isn't discovery - it's operability: whether the web can actually be run once an agent arrives, and what is the ideal stack for that. Should we lean into headless protocols, or ones like WebMCP that treat the UI as the source of truth? Does a site need to implement every new spec just to support every kind of agent? So we stopped guessing and watched real agents work the whole journey: finding, understanding, authenticating, acting, handing back to a human. The findings go against the last year of agent-readiness advice. Agents ignore the files we built for them, reaching for docs and homepages instead - and whatever they reach, they trust and act on. But when those files are linked properly, their usage jumps 4x. The format isn't the key for the agentic web. Reachability is. The web will never be completely headless. Some moments still demand a human: choosing a seat, comparing options, casually exploring. And agents aren't uniform - some want full headless access, others spin up a browser to fill the gaps, but that's a friction point, not a free fallback. So the web is going nearly headless, always with a human eye at the end. This talk maps the entire agent web landscape based on findings from real agent journeys research: * Which protocols earn their place and which are noise. * Why "agent-ready" and "accessible" are the same engineering problem. * How MCP Apps close the last mile - and when headful protocols like WebMCP step in. * How to build for agent-readiness that survives the next model - not a checklist that's stale in a month. The gap between ready and not is about to separate the relevant from the invisible.

SESSION Workshops Day 2 · Track 4 12:05pm-12:25pm

Claude Managed Agents workshop (Part 4)

Priyanka Phatak — Member of Technical Staff, Anthropic · Gabriel Cemaj — Member of the Technical Staff, Anthropic

Build an agent with Claude Managed Agents

SPONSOR Security · Track 5 12:05pm-12:25pm

Agentic Development Security

Ezra Tanzer — Director, Product Management, Snyk

SESSION Claws & Personal Agents · Track 6 12:05pm-12:25pm

Your Voice Agent is Just a Walkie-Talkie

Neil Zeghidour — Co-founder & CEO, Gradium

Everyone says cascaded voice pipelines are dead and native speech models are the future. Yet production environments are still dominated by STT-LLM-TTS stacks. Reconciling the natural flow of native audio with the elite reasoning of a cascaded agent remains an unsolved systems problem. This talk dissects the brutal technical trade-offs behind that counterintuitive reality. We will break down why your voice agent is still stuck behaving like a walkie-talkie and map out the specific technical roadmap required to build full-duplex AI that actually works.

SESSION LLM Recsys · Track 7 12:05pm-12:25pm

Open Q&A: LLM Recsys

Devansh Tandon — Principal Product Manager, Meta

SESSION Forward Deployed Engineering · Track 8 12:05pm-12:25pm

How Forward Deployed Engineering is done at Cognition

Jia Wu — Deployed Engineering Lead, Cognition AI

SESSION Posttraining & Midtraining · Track 9 12:05pm-12:25pm

Bugcrowd posttraining talk

David Brumley — Chief AI and Science Officer, Bugcrowd, Inc

SESSION AI-Native Enterprises · Leadership 1 12:05pm-12:25pm

Scaling Code Quality: Building uReview, Uber's Multi-Agent Code Review Engine

Will Bond — Staff Software Engineer, Uber · **Ameya Ketkar** — Software Engineer, Uber Technology Inc.

At Uber scale, human-only code reviews create massive bottlenecks, while generic AI tools overwhelm developers with noisy, hallucinated spam. This session explores the architecture behind uReview, Uber's multi-agent AI code review engine designed strictly for high-precision feedback. Attendees will learn how we moved beyond monolithic prompts to build a modular pipeline featuring deep contextual ingestion, specialized domain agents, and a Generator-Verifier grader system. By enforcing strict confidence scoring and semantic deduplication, uReview filters out AI noise, shifting the focus from comment quantity to high-signal actionability and significantly reducing Pull Request cycle times. Talk Outline I. The Code Review Crisis at Uber Scale (0–3 mins) Establish the critical tension between engineering velocity and code quality, highlighting why standard AI implementations fail in massive monorepo environments. 1. The Monorepo Bottleneck: At Uber, thousands of engineers commit code daily. Relying solely on human reviewers creates a massive operational bottleneck, leading to reviewer fatigue, extended Pull Request cycle times, and inevitable missed vulnerabilities. 2. The Developer Spam Problem: Generic LLM integrations fail because they prioritize comment quantity over actionable quality. If an AI posts ten hallucinated suggestions on a diff, developers will simply mute the tool. AI must reduce cognitive load, not add to it. 3. The Signal-to-Noise Mandate: Defining the North Star for uReview. The goal is not to replace human reviewers, but to build an AI system that respects developer time by delivering high-precision, strictly verified code feedback. II. The uReview Architecture: A Modular Agentic Pipeline (3–10 mins) Detail the transition from a monolithic prompt approach to uReview's sophisticated, multi-stage agentic workflow designed for enterprise codebases. 1. Deep Contextual Ingestion: A standard git diff is not enough. We discuss how uReview fetches extended context, integrating with our build systems to analyze surrounding functions, upstream dependencies, and class hierarchies before generating a single token. 2. Specialized Domain Assistants: Instead of a generalist model, uReview deploys independent AI agents. We route code to narrow, specialized agents—such as a Go Concurrency Analyzer, a Java Memory Leak Detector, or a Security Vulnerability Scanner—to ensure precise, domain-specific insights. 3. Hybrid Intelligence: Probabilistic LLMs cannot operate in a vacuum. We detail how uReview integrates deterministic tools, like Bazel dependency graphs and static linters, to ground AI suggestions in objective codebase realities. III. Engineering the Trust Layer (10–17 mins) Dive into the verification phase. This is the core engineering that filters out AI noise and ensures uReview maintains developer trust. 1. The Generator-Verifier Pattern: Implementing a Grader Model architecture. A primary agent generates code suggestions, but a secondary, high-reasoning model audits those suggestions against strict coding guidelines to catch hallucinations before they reach the PR. 2. Confidence Scoring and Suppression: We assign a numerical confidence score to every generated comment. If a comment falls below our calibrated threshold, uReview silently drops it. We explore the engineering behind suppressing low-confidence outputs to prevent tooling spam. 3. Semantic Deduplication: Technical strategies for merging overlapping warnings. If a deterministic static analysis tool and an LLM agent flag the same null pointer exception, uReview merges them into a single, concise developer instruction. IV. Operationalizing uReview at Scale (17–20 mins) Conclude by discussing the long-term governance, feedback loops, and measurable impact of running an AI review engine in production. 1. The Telemetry Feedback Loop: We embedded Useful and Not Useful rating buttons directly into the developer UI on every uReview comment. We discuss how this telemetry flows back into a curated data lake, driving continuous Reinforcement Learning from Human Feedback and prompt refinement. 2. Shifting Success Metrics: Why organizations must abandon vanity metrics like total comments posted. We measure uReview's success through Actionability Rate (the percentage of AI comments accepted as commits) and the reduction in Mean Time To Merge.

SESSION AI Architects: Show my Workflow · Leadership 2 12:05pm-12:25pm

Prototyping as Leadership: How a CTO Ships with AI Agents

Hursh Agrawal — Co-Founder & CTO, The Browser Company

I am a CTO and co-founder with a toddler, 15+ recurring meetings a week, 7 direct reports, and right now—7 open pull requests across two repos. Most engineering leaders eventually hit a wall where this kind of calendar tetris forces them to stop shipping code and start communicating solely through roadmaps. But what if AI agents didn't just act as coding assistants, but fundamentally restructured how executives use fragmented time to prototype the future? In this talk, I will share the exact multi-model workflows I use to plan with one model, implement with another, and build asynchronous play-and-feedback loops that fit perfectly between meetings. You will learn how to navigate code reviews for agent-assisted executive PRs, and leverage AI to shift your leadership style from telling your team what to build to showing them functional prototypes.

SESSION **Expo Stage 1 NE** 12:05pm-12:25pm

Your Agent Is Lying to You About Whether It Worked

Dat Ngo — AI Architect, Arize AI

Every span is green, every tool call returned cleanly, and the agent still regenerated the same plan 27 times before giving up invisible to any outcome metric, obvious in the trajectory. We pull up a real trace where the outcome looks healthy and the path is a disaster, then show Signal, our agent, surfacing it automatically: sweeping the project, ranking it above the noise, and linking straight to the offending trace with debugging evidence attached. The live version of the trajectory-over-outcomes argument, with a one-click path from "something's wrong" to "here's exactly where."

SESSION **Expo Stage 2 NW** 12:05pm-12:25pm

Why building building agent quality platforms is hard.

Hossein Niazmandi — Solutions, Braintrust

An eval platform is not just a test runner. You are building shared definitions of good, reliable data pipelines, labeling workflows, versioning, and trust in results across many teams and model changes. This session breaks down the hidden complexity, the common failure modes, and the design principles that make evals credible and usable in day-to-day engineering.

SESSION **Expo Stage 3 SW** 12:05pm-12:25pm

Can LLMs write fast multi-GPU kernels? We built a benchmark to find out.

Simran Arora — Computer Science PhD Student, Stanford University

LLMs have gotten surprisingly good at writing GPU kernels, but almost all the benchmarks measuring that progress are single-GPU. In production, communication is the bottleneck: all-reduce alone accounts for over 20% of inference latency on Llama-3.3-70B, and that gap keeps widening as compute scales faster than interconnect bandwidth. ParallelKernelBench (PKB) offers a benchmark and evaluation framework for multi-GPU kernel generation and includes 87 problems from real codebases where the task is replacing PyTorch + NCCL with a CUDA kernel that moves data directly over NVLink. We tested GPT-5.5, Gemini 3 Pro, Opus 4.7, and other frontier coding models. Under a third of problems solved were correctly, and fewer than a quarter of those beat the naive baseline. We'll cover why they fail, what the patterns look like, and a few cases where models produced kernels faster than anything publicly available, including one for NVIDIA NeMo-RL's GRPO training loop, which has no prior optimized public reference. The benchmark is open source and we want to see what you can do!

SESSION **Expo Stage 4 SE** 12:05pm-12:25pm

Self-Improving Agents That Teach the Company Back

Rafal Wilinski — Founding Engineer, Runlayer

Agents forget too much. A run might solve a customer escalation, debug a deployment, or figure out the review pattern for a tricky code path, then the knowledge disappears into a transcript. At Runlayer, we started treating that knowledge as a product surface. Skills are reviewable, editable instructions that agents can load over MCP. An agent can start with a task, learn something useful while doing the work, and draft or update a private skill from that run. That skill loads into future runs for the same agent, stays inspectable by humans, and can eventually graduate into a team or org-level skill. The flywheel gets more interesting once a skill becomes useful beyond the agent that created it. A learned skill can move from one agent's private memory into shared organizational knowledge, then become available through the Runlayer plugin inside Claude Code, ChatGPT, and other AI clients employees already use. The agent does the work, captures the playbook, and the company gets better at that work everywhere agents are used. This talk walks through the architecture and product choices behind self-improving skills: post-run distillation, skill mutation tools, private-by-default scoping, runtime loading, UI inspection, promotion into shared skills, and the safety boundary between this agent learned something and everyone should now use it. The goal is an agent that leaves behind a better handbook for the next person, the next run, and eventually the whole organization.

12:30pm

12:30pm-1:30pm **Fireside Chat: Gergely Orosz × Simon Eskildsen**

1:30pm

SESSION **Software Factories · Main Stage** 1:30pm-1:50pm

Get Out of the Model's Way

Kevin Hou — Engineering Lead @ Antigravity, Google DeepMind

From autocomplete to chat to agents to agent orchestration...how do you build a product that scales with intelligence? What core primitives enable agents to operate at the technical (and non-technical) frontier? How can you best squeeze every ounce of capability out of your agentic dev tools? I'll answer all these questions and break down how Google Antigravity creates dynamic agent teams to solve complex tasks like building an OS-Kernal and automating research workflows.

SESSION **Claws & Personal Agents · Track 1** 1:30pm-1:50pm

Agents' next frontier: agent-to-agent and network effects

Jean-Denis Greze — Co-Founder & CEO, Town

MCP v. CLI was about how agents talk to tools. That’s not settled (but we’re camp MCP… mostly). Almost nothing has settled how agents talk to each other - and that's where the next wave of value (and network effects and virality) lives. At Town we run a personal AI agent in production inside real people's inboxes, calendars, and Slack, and we've built agent-to-agent (A2A) on our platform: 1:1 A2A messaging, agents that carry a short bio of one another, HITL when sensitive data is shared or write actions are involved, and early tests around 1:N A2A. I’ll talk about the why, the opportunity, and the production architecture underneath. Audience takeaway: a concrete mental model for building multi-agent systems on top of the data and surfaces users already live in, plus our learnings on early failure modes to avoid.

SPONSOR **Vision & OCR · Track 2** 1:30pm-1:50pm

From Ingestion to Agents: How Leading AI Teams Build on Document Intelligence

Adit Abraham — CEO and cofounder, Reducto

The agents of tomorrow are only as good as the context they reason on — yet most real-world data lives in messy, unstructured documents. In this session, we reveal the patterns that separate AI teams shipping reliable, production-grade agents from those stuck debugging pipelines. Drawing on patterns we've seen from AI-native startups to Fortune 10 enterprises, we'll cover what it takes to transform complex documents into clean, accurate context at scale across legal, finance, healthcare and more. From ingestion architecture to agent-ready outputs, walk away with the strategies top teams use to turn document chaos into competitive advantage.

SESSION **Search & Retrieval · Track 3** 1:30pm-1:50pm

If we want them to do Knowledge Work, we need to design Knowledge Agents

Benjamin Clavié — Member of Technical Staff, Mixedbread Inc.

It's tempting to assume that just like agents revolutionised coding, they will revolutionize other areas: legal, finance, advertising, and even medicine. All of those have in common that they are fundamentally knowledge work. And thankfully, humans have spent thousands of years searching for the best possible workflows for knowledge work. And yet, we seem to be disregarding all of these learnings, forcing every knowledge task into the shape that worked for coding. Today, we're going to talk about the history of knowledge work and how tools were co-designed to support it to understand how we should be building Knowledge Agents, themselves co-designed with their Knowledge Tools. This is key to avoiding falling into a "good enough" local optimum: think about legal clerking, a core part of the legal industry where information gathering and reasoning is performed to support the work of senior lawyers. The practice of clerking follows its own code, rules and best practices, which could not have feasibly emerged from studying software engineering: and similarly, there is no reason to believe knowledge agents could emerge from coding agents.

SESSION **Workshops Day 2 · Track 4** 1:30pm-1:50pm

Everybody Gets a Digital Clone! (Part 1 of 3)

Neil Zeghidour — Co-founder & CEO, Gradium

Walk out of this workshop with a deployed digital clone that makes your phone calls for you. We will skip the theory and immediately get our hands dirty wiring together OpenClaw, Twilio, and Gradium to build an autonomous voice agent on a live cellular network. You will tackle the hardest parts of real-time telephony: routing audio streams, handling human interruption, and killing latency. In 60 minutes, your AI will be ready to call restaurants for the daily special, book appointments, and actively negotiate on your behalf.

SPONSOR Security · Track 5 1:30pm-1:50pm

Using LLMs to Secure Source Code

Eugene Yan — Member of Technical Staff, Anthropic

Models are now finding and fixing real vulnerabilities at scale. Drawing on Anthropic's work with security teams, this talk walks a six-step workflow — threat model, sandbox, discover, verify, triage, patch — through one running example, shows where orgs actually bottleneck, and gives you a copy-paste path to your first scan.

SESSION Voice & Realtime AI · Track 6 1:30pm-1:50pm

Tolan: Voice-First AI Companion

Paula Dozsa — iOS Engineer, Tolan

SESSION LLM Recsys · Track 7 1:30pm-1:50pm

From approval loops to autonomous agents with Docker pt1

John Craft — Solutions Engineer, Docker

You've invested in the best models, coding agents, and AI tooling. Now comes the hard part: unlocking autonomous development without creating security headaches, governance gaps, or endless approval loops. In this 90-minute hands-on workshop, you'll learn how to run coding agents in isolated environments built for autonomous work, create a 'golden path' for AI-assisted development across your organization, reduce software supply chain risk with secure, hardened containers, manage multiple agents with the right permissions and guardrails, and scale AI-powered development without slowing developers down.

SESSION Forward Deployed Engineering · Track 8 1:30pm-1:50pm

The Dirty Secret of Forward Deployed Engineering

Natalie Meurer — Head of Agent Engineering, Sierra

Since its origins at Palantir, the term "Forward Deployed Engineer" has described wildly different jobs, yet today it's one of the fastest-growing roles in AI. What happened? And what does that reveal about the future of engineering? Join Nat Meurer, Head of Agent Engineering at Sierra, for a historical tour of one of tech's most misunderstood roles, and why its biggest contradiction may explain where the industry is headed next.

SESSION Data Quality · Track 9 1:30pm-1:50pm

The Base Model is Dead

Varun Singh — Pre-Training Lead, Arcee AI

It's a common belief that large language models are trained to be a good model of human web-text, and thus base models are "mirrors" of what we see on the internet. Historically, this was largely true, but no modern base model truly reflects the internet in the way that GPT-3 once did. Instruction data along with synthetic reasoning traces are moving earlier and earlier into the training pipeline, and "mid-training" has emerged as a new stage to accommodate longer datapoints that more concretely resemble downstream capabilities. As a result, pre-training no longer has the goal of creating a linguistic prior, but instead has the additional goals of baking in behavior and more atomic skills into the trained "base" model. Between this shift in what a base model is and the blurring of the lines between the different stages of model training, it's an open question as to what the best approach is here (at least outside the walls of the big labs). But I believe that the role we view the base model playing will continue to shift as we're pulled forward through new phases of model capabilities.

SPONSOR Track M 1:30pm-1:50pm

Modernize CI/CD using agent-assisted workflows that reduce manual debugging

Salil Subbakrishna — GitHub

AI agents are reshaping CI/CD. See how workflows become adaptive—understanding failures, fixing issues, and accelerating releases without constant manual intervention.

SESSION **Software Factories · Leadership 1** 1:30pm-1:50pm

Spin at the Gate Until Green: The Engineering Primitives Behind Self-Driving Codebases

Andrew Orobator — Senior Software Engineer, Reddit

Most AI-assisted development fails the same way: the AI produces plausible output, the human can't tell if it's right, so they check manually, find the problem, re-prompt, and repeat. This loop doesn't scale. There's a different approach. If you can express correctness as a binary — does it compile, do the tests pass, does the lint check clear — you can remove the human from that loop entirely. The AI submits. The gate checks. If red, it adjusts and resubmits. Spin at the gate until green. This talk covers the engineering primitives that make this possible: personas (consistent behavior at the agent level), skills (composable, reusable prompt modules), worklogs (accountability across sessions), postmortems (turning failures into constraints), and spec-driven development (making the target explicit enough for a machine to hit it). The culmination is a flag lifecycle agent — triggered by a cron job, cleaning up stale feature flags, verified by compile + test + lint, no human in the loop. Not hypothetical. Working prototype, proven in practice. I co-authored a ten-part series on this methodology with Claude. The series was built using the workflow described in this talk. If you don't trust the theory, the fact that this talk exists is the proof.

SESSION **AI Architects: Show my Workflow · Leadership 2** 1:30pm-1:50pm

Serving 2 Million Models Without Melting: Scaling the Hugging Face Hub

Arek Borucki — Machine Learning Platform & Database Engineer, Hugging Face

Hugging Face hosts over 2 million public models, 500,000+ datasets, and serves 13 million users across 50,000+ organizations, including over 30% of the Fortune 500. That growth didn't come with a manual.In this talk, we'll pull back the curtain on the infrastructure decisions that kept the Hub fast and reliable as traffic grew by orders of magnitude. We'll dive into why we chose MongoDB Atlas as our core data layer, how its document model maps naturally to the messy reality of ML model metadata, and what it took to keep p99 latency low when every request hits a catalog of millions. We'll also cover the trade-offs we faced, the things that broke along the way, and what "lean operations" actually means when your platform serves a third of the Fortune 500. Expect real architecture decisions, real numbers, and lessons you can take back to your own stack.

SESSION **Expo Stage 1 NE** 1:30pm-1:50pm

Every Agent, Everywhere, All at Once

Vlad Luzin — Founder, Band.ai

Coding agents are deaf to anything outside their own session, and a LangGraph or CrewAI one has no idea the others exist. Different vendors, different frameworks, different machines none of them share a way to work together. This demo fixes that live: the Claude Code on your laptop, Codex on your colleague's, a LangGraph agent you're running locally, and the OpenClaw on your Mac Studio at home collaborating on the same goal, going back and forth, full-duplex, across every vendor, framework, and machine line at once.

SESSION **Expo Stage 3 SW** 1:30pm-1:50pm

Designing Evals That Earn User Trust

Felipe Blanes — Amazon

Most teams measure their agent against a benchmark, ship it, and hope. But when your agent serves real users, a benchmark won't tell you if it's actually working. This session is about building an eval suite that captures what success looks like in production, runs against real user workflows, and feeds back into product decisions. Here's the flywheel we use in practice: start with what success looks like from the user's perspective, instrument production workflows to capture those signals, diagnose where the agent falls short, and feed those insights into the next thing you build. You'll see how it shaped concrete product bets, turning eval results from a report card into a discovery tool.

SESSION **Expo Stage 4 · Expo Stage 4 SE** 1:30pm-1:50pm

Stop prompting

Greg Pstrucha — Sentry

In this talk I dive into usage of tooling, type systems and frameworks to enforce guardrails and limit slop produced by AI agents inside large codebases.

1:55pm

SESSION **Software Factories · Main Stage** 1:55pm-2:15pm

Self-Improving software factories: The new open source model"

Zach Lloyd — Founder and CEO, Warp

Alt titles: Agent orchestration with message passing / Agent orchestration for every model / Warp's approach to agent orchestration With models getting more capable, we've quickly scaled from single agent problems to multi-agent problems – How can agents delegate tasks to accomplish ever-larger goals? You may have heard of "agent swarms" or "agent teams" in this arena, but they come with drawbacks: model lock-in, complex UX, or both. We want to share how we've tackled orchestration with our model-agnostic platform, Oz. Our approach has some unique goals: - Support any model, *and* any harness (`claude`, `codex`, etc) - Delegate across local instances *and* across isolated cloud sandboxes - Provide a UX that requires zero tmux or TUI knowledge to use We'll explore how we implemented message passing across harnesses, how we handle agent sandboxing with Docker containerization + serverless deploys, and how we designed these primitives to make a system that works with any agent. You'll walk away with a clear outline of how to build agent orchestration well. Plus, we invite you to try our Oz orchestration platform and tell us what you think. Talk format: Primarily a tech demo and code walkthrough. We'll show multiple examples of tasks that are best served by delegation, and show both local and cloud-based runs. We'll also walk through the design of our message passing implementation at a high level to show how it works.

SESSION **Claws & Personal Agents · Track 1** 1:55pm-2:15pm

Claude for long-horizon tasks

Lance Martin — Member of Technical Staff, Anthropic

Claude is capable of long horizon tasks. In this talk, we'll share lessons learned about building agent harnesses for reliable and secure long-horizon work. This include decoupling the brain and hands, self-verification, self-learning, and design for evolving agent harnesses.

SPONSOR **Vision & OCR · Track 2** 1:55pm-2:15pm

The Best Models Still Reason Like Toddlers

Andrew Dai — Co-founder and CEO, Elorian

Frontier AI models score 80–90% on standard benchmarks like RKGI, yet when tested on visual tasks any 3-year-old handles effortlessly (like counting objects in an image), those same models fall to pieces. I watched this gap widen firsthand during my 14 years at Google Brain and DeepMind, where I co-led development on GLaM, PaLM 2, and Gemini. The problem is that most models hit high RKGI scores not through genuine visual understanding, but by coding – a workaround that scores well and reveals little. Strip that away and you're left with systems that struggle to solve a simple crossword puzzle, identify what's the same or different across two images, or navigate a basic 3D view. These tasks are essential to achieve human-level reasoning capability. And the current benchmark ecosystem wasn't built to evaluate for it, leaving us with top scoring models that can't even follow along with Count Von Count. In this talk I'll dig into why the current eval landscape systematically overstates capability, the structural reasons it does so, and how we got here from the viewpoint of someone who was inside a leading frontier lab. I'll close with what I believe a more rigorous, consensus-driven eval framework needs to look like, and why the field needs to build one before the next generation of visual systems ships into the real world. Fixing visual reasoning starts with fixing how we measure it. For engineers building on top of these models today, whether that's document understanding, robotic perception, medical imaging, or any system where visual perception context matters, the cost of getting this wrong is already showing up in production.

SESSION Search & Retrieval · Track 3 1:55pm-2:15pm

Your Agreements Are a Database You Can't Query. We're Fixing That

Hiral Shah — Senior Director of Product, AI Applications, Docusign · **Sean Sodha** — Senior Product Manager, NVIDIA

Agreements power every enterprise business, but the most critical data — pricing schedules, SLA obligations, rate cards — is often trapped in tables that traditional extraction tools destroy. This session shows what changes when you can actually extract that data accurately at scale and make it searchable. We'll walk through the before and after: Before: Contract tables require manual review. Rate cards are buried. SLA terms are scattered across exhibits. Procurement teams spend hours piecing together pricing structures — and searching for specific terms means opening every document. After: Tables are automatically extracted, structured, and queryable. Operations teams can surface SLA notification requirements on demand. Legal can answer "what hourly rate did we agree to?" in seconds. Docusign will share what we've achieved evaluating NVIDIA Nemotron Parse for our document processing pipeline, including how we tested against real enterprise contracts (not synthetic benchmarks), why we're serving the model via vLLM, and what it takes to turn extracted table data into searchable, retrievable agreement intelligence. NVIDIA will cover the architecture behind Nemotron Parse and where the model is heading — including how NeMo Retriever's embedding and reranking models connect extracted data to search and RAG-based applications. Attendees will leave with a realistic view of where vision-language models excel at document understanding, where the gaps remain, and how to think about building searchable contract intelligence into their own systems.

SESSION Workshops Day 2 · Track 4 1:55pm-2:15pm

Everybody Gets a Digital Clone! (Part 2 of 3)

Neil Zeghidour — Co-founder & CEO, Gradium

Walk out of this workshop with a deployed digital clone that makes your phone calls for you. We will skip the theory and immediately get our hands dirty wiring together OpenClaw, Twilio, and Gradium to build an autonomous voice agent on a live cellular network. You will tackle the hardest parts of real-time telephony: routing audio streams, handling human interruption, and killing latency. In 60 minutes, your AI will be ready to call restaurants for the daily special, book appointments, and actively negotiate on your behalf.

SPONSOR Security · Track 5 1:55pm-2:15pm

Dual-Surface Architecture: Serving Humans and Agents from the Same Tool Layer

Ethan (Jung Min) Cha — AI Development Lead, The Carlyle Group

Every enterprise AI talk right now is about capability. Almost none are about containment. That's the gap this talk fills, because it's where regulated deployments actually die. The Deterministic Harness is the set of rigid rails around a model: schemas, data contracts, tool boundaries, and audit paths. These rails are what turn a probabilistic model into a deployable enterprise asset. The idea isn't new. Aviation wraps pilots in envelope protection. Nuclear wraps reactors in passive safety. Banking wraps algorithmic trading in transaction limits. Every regulated industry figured out the same thing eventually: high-variance systems only become deployable when wrapped in low-variance containment. Enterprise AI is catching up, not inventing. I'll walk through the single governed MCP and API server we built at Carlyle, and the architectural decisions behind it. You'll leave with four things: 1. A phased rollout model where each phase earns the next. Moving from locked-down reads to trusted writes isn't risk mitigation. It's trust compounding. Each phase generates the observability that underwrites the autonomy granted in the next one. Skip a phase and you don't save time. You destroy the evidence base that would have justified the next step. 2. One contract, two surfaces. A single data layer that serves both the human UI and the agent. The institution then has exactly one answer to any question either might ask. When the agent and the UI disagree, users lose trust in both. 3. An intent based feedback loop that captures what LLM providers structurally cannot. The gap between what users tried to accomplish and what the system actually delivered is invisible to Anthropic, OpenAI, and Google. Only the harness owner sees it. We close that loop back into the governed server, and it compounds into differentiation that model providers cannot replicate from where they sit. 4. The failure modes we hit and what we'd redesign. A pre mortem folks will inherit for free, from two regulated industries where a wrong answer has a named owner.

SESSION **Voice & Realtime AI · Track 6** 1:55pm-2:15pm

5 Voice Agent Failure Modes You'll Hit in Week One

Venky B — Founder & CEO, Plivo · **Vyas A** — Head of Product, Plivo

Building a voice agent that demos well is easy now. The hard part starts the second a real person calls it. Most voice agents today are basically a chatbot with a microphone bolted on, they listen, then think, then talk, one side at a time, like a walkie talkie. Real conversations don't work that way. People pause in the middle of a thought, they say "um" and "uh", they talk over you, they change their mind halfway through. The agent has to work out when you're actually done talking, when it should stop talking, and when you've said something it cannot afford to get wrong, like your phone number or email. None of this shows up when you test with text. All of it shows up in week one. This talk is the five failures that hit every team in that first week, the ones we see again and again. For each case we will walk though examples and best practices for what actually breaks and what to do about it. If you're about to put a voice agent in front of real callers, or you already did and it's quietly falling apart, this is the talk that saves you the weeks everyone else burns figuring it out

SESSION **LLM Recsys · Track 7** 1:55pm-2:15pm

From approval loops to autonomous agents with Docker pt2

John Craft — Solutions Engineer, Docker

You've invested in the best models, coding agents, and AI tooling. Now comes the hard part: unlocking autonomous development without creating security headaches, governance gaps, or endless approval loops. In this 90-minute hands-on workshop, you'll learn how to run coding agents in isolated environments built for autonomous work, create a 'golden path' for AI-assisted development across your organization, reduce software supply chain risk with secure, hardened containers, manage multiple agents with the right permissions and guardrails, and scale AI-powered development without slowing developers down.

SESSION **Forward Deployed Engineering · Track 8** 1:55pm-2:15pm

How Forward Deployed Engineering is done at Decagon

Sunny Rekhi — FDE CTO, Decagon

SESSION **Data Quality · Track 9** 1:55pm-2:15pm

Ending AI Slop

Thais Castello Branco — Founder & CEO, Taste Labs

SESSION **AI-Native Enterprises · Leadership 1** 1:55pm-2:15pm

AI Evals Platform for Cross-Functional Teams at Scale

Nachiket Paranjape — Software Engineer, DoorDash · **Swaroop Chitlur Haridas** — DoorDash

DoorDash's Evals Platform is designed for more than just engineers. It brings human review, automated judges, and online experimentation into a single calibration loop so engineering, product managers, and strategy and operations teams can all contribute to improving AI quality. Engineers can instrument, trace, and evaluate agent behavior, while cross-functional teams can review outputs, curate trusted examples, and provide structured feedback that improves how automated judges behave over time. By combining experimentation, fully customized annotation workflows, calibration, and analytics in one system, the platform turns AI quality from a fragmented technical exercise into a shared operating model for continuously improving agent performance and making rollout decisions with confidence. While vendor platforms offer pieces of this workflow, we needed something broader: a unified system that lets engineers, product managers, and Strategy & Ops all participate directly in improving AI quality. Our goal is not just to run evals, but to enable cross-functional teams to review outputs, calibrate judges, run experiments, and make rollout decisions without being blocked on engineering. That requirement, along with tighter integration into our internal workflows and operating model, is why we are building this platform in-house.

SESSION AI Architects: Show my Workflow · Leadership 2 1:55pm-2:15pm

IT Admin for the AI Workforce: Why Your AI Agents Will Need Their Own IT Department

Sarthak Aggarwal — Co-founder, Decawork

Every enterprise will soon run two workforces - human and AI. Humans already have IT departments managing their identities, access, incidents, and compliance. Who manages all that for your fleet of 10,000 AI agents? Nobody. Yet. At Decawork AI, we started by building autonomous IT resolution for human employees - a dual-agent system where the agent that thinks can't act and the agent that acts can't improvise. We're live in production across multiple enterprises - autonomously resolving incidents across identity systems, security platforms, endpoint infrastructure, and collaboration stacks. But here's what we discovered: the patterns for managing human IT - identity lifecycle, access governance, incident resolution, audit logging - are the exact same patterns you'll need to manage AI agent fleets at scale. The next massive infrastructure layer isn't AI agents doing work. It's AI agents managing other AI agents. This talk covers the architecture, the production war stories, and the thesis: IT Admin for the AI workforce is an inevitability, and we're building it now.

SESSION Expo Stage 1 NE 1:55pm-2:15pm

Who Approved That MCP Server? Governing the Tool Layer

Jim Clark — Principal Software Engineer, Docker

Your developers are installing MCP servers faster than security can review them. An unvetted server is a direct line to your data. This talk shows how the Docker MCP Gateway puts every server and tool behind one org-managed catalog: vetted, signed, default-deny on anything unapproved, governed by the same policy engine as network and filesystem. Walk away with a hands-on demo: stand up a catalog, block an unvetted server, and watch policy enforce at the runtime.

SESSION Expo Stage 2 NW 1:55pm-2:15pm

Voice Agents Are Mostly Invisible. Here's How to See Them.

Fuad Ali — Senior Product Manager, Arize AI

Voice agents are one of the fastest-growing and hardest-to-debug categories: the failures live in latency, turn-taking, transcription drift, and tone none of which show up in a text log. We demo Voice traces and Session views, following a real voice session span by span, and Voice evals for scoring what text-only observability can't reach. A short, differentiated session on a problem most of the room is about to hit and few tools address.

SESSION Expo Stage 3 SW 1:55pm-2:15pm

what we learned by analyzing 1M AI generated PRs

Background coding agents are quickly moving from novelty to real-world software development workflows. Based on Greptile's analysis of millions of pull requests across 65,000 organizations, this talk explores how often end-to-end AI-generated Pr's are being used and how their quality compares to human-written code. The data shows detectable agent-generated Pr's grew from under 1% in February 2025 to 27.6% in April 2026, with early quality signals like revert rates and code churn suggesting these agents may already be competitive in serious codebases.

SESSION Expo Stage 4 SE 1:55pm-2:15pm

Deploying browser agents at scale

Derek Meegan — Software Engineer, Browserbase

Not every browser agent trajectory is the same, and treating them like they are is how teams quietly burn budget on agents that never ship. This talk walks through the two trajectory types behind every browser agent, the cost/performance/maintainability tradeoffs that decide whether they hold up, and the concrete patterns for evaluating, hardening, and iterating on them.

2:25pm

SESSION Software Factories · Main Stage 2:25pm-2:45pm

We're the bottleneck, but we don't have to be

Ido Salomon — Co-Creator, MCP Apps

As agents improve at doing real work, humans become the real bottleneck. Luckily, the skills we need to work with agents aren't entirely new, they've just been hiding in unexpected places. Drawing lessons from AgentCraft's Warcraft-inspired UI for coordinating multiple agents, this talk explores how gamification can raise the ceiling for sophisticated AI orchestration while lowering the floor for everyday developers. Ido will show how visual state, spatial metaphors, and autonomy can make multi-agent systems more approachable, inspectable, and fun to use.

SESSION Claws & Personal Agents · Track 1 2:25pm-2:45pm

From coding to Knowledge work agents

Karan Vaidya — Co-founder, Composio

MCP, skills, Cli - so much noise - what's the best way for agents to communicate

SPONSOR Vision & OCR · Track 2 2:25pm-2:45pm

You're Not Thinking Big Enough: Rebuilding Food Systems from First Principles with AI Agents

Cody Menefee — Success Engineer, Firecrawl

Most of the AI world is still thinking too small. We're building SaaS wrappers and GTM agents while real-world systems are still run through fragmented knowledge, delayed feedback, and human guesswork. In this talk, I'll show how I'm building an outdoor agentic system for pasture-raised livestock operations using LLMs, a Firecrawl-curated knowledge base, drone and satellite imagery, and geo collars to monitor pasture, guide animal movement, and support better decisions across cattle, sheep, poultry, and more. I'll cover the architecture, retrieval and grounding, human approval loops, and what broke first: hallucinated confidence, weak environmental grounding, sparse evals, and the gap between a smart answer and a safe action. It's a case study in building agents for the physical world, and a broader argument that AI's real upside is in rethinking real-world systems from first principles.

SESSION Search & Retrieval · Track 3 2:25pm-2:45pm

How to Connect AI to Billions of Legal Documents

Simon Eskildsen — CEO and co-founder, turbopuffer · Jacob Lauritzen — CTO, Legora

Legora's foundational engineering challenge is connecting frontier LLMs to billions of legal documents so the models can efficiently solve end-to-end legal workflows without burning extra tokens. We'll share the retrieval architecture we built with turbopuffer that achieves: 1. Strict data isolation across millions of legal cases in a very security-conscious domain 2. Predictable search performance (<100ms p90 latency) on large contexts 3. High retrieval quality (95%+ recall@10) with fewer agent loops We'll retrospect on two architectures that failed to achieve all 3 (and why), and the key design factors that make the current solution work at our scale. Practical takeaways include: - How to evaluate per-tenant vs shared-index retrieval under strict data isolation - How to efficiently index and retrieve context to maximize relevance per input token - How to build a highly intelligent AI application when your inference budget is constrained

SESSION Workshops Day 2 · Track 4 2:25pm-2:45pm

Everybody Gets a Digital Clone! (Part 3 of 3)

Neil Zeghidour — Co-founder & CEO, Gradium

Walk out of this workshop with a deployed digital clone that makes your phone calls for you. We will skip the theory and immediately get our hands dirty wiring together OpenClaw, Twilio, and Gradium to build an autonomous voice agent on a live cellular network. You will tackle the hardest parts of real-time telephony: routing audio streams, handling human interruption, and killing latency. In 60 minutes, your AI will be ready to call restaurants for the daily special, book appointments, and actively negotiate on your behalf.

SPONSOR Security · Track 5 2:25pm-2:45pm

Agentic Security: Permissions, Provenance, and the Agent Supply Chain

Steve Yegge — Icon, Gas Town

As AI agents move from demos into production engineering workflows, the security boundary shifts from code alone to the permissions, tools, prompts, dependencies, credentials, and orchestration layers that agents can touch. This talk frames agentic security broadly: least-privilege agent permissions, sandboxing and capability design, provenance for agent-generated changes, risks in agent/tool/package supply chains, and practical patterns for keeping autonomous coding and operational agents auditable and containable.

SESSION **Voice & Realtime AI · Track 6** 2:25pm-2:45pm

I Monitored Crime Audio. Voice Agents Scare Me More.

Sumanyu Sharma — Founder/CEO, Hamming AI

Bad voice-agent calls are starting to look less like QA bugs and more like incident scenes. I learned that instinct at Citizen, where noisy radio, ambiguous speech, fast-moving incidents, and real-time alerts became information people might actually act on. That work was stressful for obvious reasons. Voice agents scare me more. Not because they sound creepy. Because they sound good enough that people trust them. And now they are connected to calendars, CRMs, EHRs, reservation systems, refunds, transfers, account data, and support workflows. At Hamming, we monitor more than 10,000 voice agents and have analyzed millions of calls. The weird thing you learn at that scale is that production voice agents do not usually fail like demos. They fail quietly. The agent sounds natural, but misses a two-word answer. It handles the happy path, but loses the plot when the caller interrupts. It says the address was updated, but no tool call happened. It supports six languages, but gets worse at the switch point between two of them. This talk is about treating every bad voice-agent call like an incident scene. The evidence is there if you collect it: transcript, waveform, latency waterfall, interruption points, ASR uncertainty, tool trace, system-of-record state, and post-call outcome. At Tesla, I learned that autonomous systems need release gates and regression loops before they hit the real world. At Citizen, I learned that messy audio becomes safety-critical when people act on it. Voice agents need both instincts. The takeaway is a voice-agent forensics loop. What did the caller say? What did the agent think happened? What did the tool actually do? What does the system of record say? And how do we turn that weird production failure into a regression test before it happens 10,000 more times?

SESSION **LLM Recsys · Track 7** 2:25pm-2:45pm

From approval loops to autonomous agents with Docker pt3

John Craft — Solutions Engineer, Docker

You've invested in the best models, coding agents, and AI tooling. Now comes the hard part: unlocking autonomous development without creating security headaches, governance gaps, or endless approval loops. In this 90-minute hands-on workshop, you'll learn how to run coding agents in isolated environments built for autonomous work, create a 'golden path' for AI-assisted development across your organization, reduce software supply chain risk with secure, hardened containers, manage multiple agents with the right permissions and guardrails, and scale AI-powered development without slowing developers down.

SESSION **Forward Deployed Engineering · Track 8** 2:25pm-2:45pm

How Forward Deployed Engineering is done at Ramp

Leo Mehr — Director of Engineering, Ramp

SESSION **Data Quality · Track 9** 2:25pm-2:45pm

Scaling to Long-Horizons: Algorithms, Environments, Compute

Ross Taylor — CEO, General Reasoning · Chengxi Taylor — Co-founder & President, General Reasoning Inc.

What does it take to scale language models to year long tasks? In this talk we'll cover the algorithm, environment and compute considerations for scaling language models to long horizons. We'll cover the latest reinforcement learning approaches, how to build hard, high-fidelity long-horizon environments, and how to build scalable infrastructure for these tasks.

SPONSOR **Track M** 2:25pm-2:45pm

Using AI tools to teach old apps new tricks

Maria Bledsoe — General Manager, Product Marketing, Microsoft

Becoming AI-ready starts with modernizing your legacy systems and technical debt — and keeping them modernized. We'll show how you can use agentic AI to take on the hardest parts of modernization: analyzing large codebases, mapping dependencies, planning upgrades, refactoring safely, while doing it all at scale with enterprise controls. With GitHub Copilot modernization capabilities, you can move from legacy complexity to modernized apps in days, not months.

SESSION **AI-Native Enterprises · Leadership 1** 2:25pm-2:45pm

Productionizing LLM Gateways: Architecture, Tradeoffs, and Hard Lessons from the Trenches

Kanish Manuja — Principal Software Engineer, Twilio Inc.

As organizations scale their use of large language models, the biggest challenge is no longer prompting, it's productionizing. This session dives deep into building and operating an LLM gateway that sits between applications and model providers, handling routing, observability, cost control, reliability, and safety at scale. Drawing from real world experience, this talk breaks down the architecture of a production LLM gateway, including model abstraction layers, request orchestration, fallback strategies, caching, rate limiting, and evaluation pipelines. We'll explore hard tradeoffs such as latency vs. cost, quality vs. determinism, and vendor lock-in vs. flexibility. Attendees will leave with concrete design patterns, failure modes to avoid, and a mental model for turning LLM experiments into resilient, scalable systems.

SESSION **AI Architects: Show my Workflow · Leadership 2** 2:25pm-2:45pm

The Era of Compound Engineering

Kieran Klaassen — GM of Cora / Compound Engineering, Every/Cora

Most codebases get harder to work with every year. Yours doesn't have to. **Compound Engineering** is a philosophy where each unit of work – every bug fix, every feature, every code review – makes the next one easier. This talk is about how that shift changes everything: from how fast you ship to how many engineers you actually need. --- At Every, we run five products with single-person engineering teams. That's not a headcount accident – it's a system. When I built [Cora](https://cora.computer), I wanted to find out how much one engineer could do with the right AI workflows. The answer became the **Compound Engineering** philosophy, now with 17k stars on GitHub. Traditional codebases accumulate complexity. Compound codebases accumulate capability. Bug fixes eliminate entire *categories* of future bugs. Patterns become tools. Over time, the codebase gets easier to understand, easier to modify, and easier to trust. **You'll walk away with:** - The mental model behind compound engineering - Concrete patterns for making every PR compound - How to scale output without scaling headcount

SESSION **Expo Stage 1 NE** 2:25pm-2:45pm

Beyond Golden Signals: Monitoring in the Age of GenAI

Marina Petzel

The four golden signals (Latency, Errors, Traffic, Saturation) have been the foundation of application monitoring for years, and it still matters, but for GenAI applications, these signals alone leave significant blind spots. A request can return 200 OK with low latency while the response hallucinates, leaks PII, or costs much more than expected. This talk will walk you through what changes when you're monitoring non-deterministic, token-priced, prompt-injectable systems. We'll cover three additional monitoring dimensions: Cost (token attribution, model-mix tracking, wasted spend on failed requests), Safety (prompt injection detection, PII scanning, jailbreak attempts), and Quality (hallucination rate, relevance scoring, user satisfaction) and show why each one is necessary alongside your existing instrumentation.

SESSION **Expo Stage 2 · Expo Stage 2 NW** 2:25pm-2:45pm

Build agents fast with GitHub Copilot (from idea to working app)

Idan Gazit — Head of GitHub Next, GitHub

SESSION **Expo Stage 3 SW** 2:25pm-2:45pm

Building agents is trivial now, context is the next frontier

Jeff Ng — Engineer, Unblocked

Standing up an agent used to be the hard part. A new class of cloud-agent frameworks has made it almost trivial: in an afternoon you can ship a fleet that reasons, plans, and calls any API you point it at. So why do so many of them fail the moment they touch real work? Because a capable agent still doesn't know the organization it operates in: its decisions, history, incidents, and how a particular team actually operates. That knowledge isn't in the model or the API, and no amount of construction adds it. This talk exposes the missing component, then shows how to build it live on a real workflow — the same move that helps a coding agent helps a support or operations one. Construction is solved. The missing context, tacit and tribal knowledge is the bottleneck that's left, and it sits upstream of everything verification attempts to catch after the fact.

SESSION **Expo Stage 4 SE** 2:25pm-2:45pm

Continuous Engineering: Software Development for the Age of Agents

AI has changed everything about how we write code. But the hard parts of building software have gotten even harder: aligning your team, maintaining architectural integrity, and worst of all, reviewing the oceans of agent-driven code. The tools and processes we rely on git pull requests; code review were built for emailing patch files. We need a new paradigm. In this talk, we're going to explore Continuous Engineering, a new approach to software development that treats the agent thread as the core unit of collaboration. Branches should be as cheap as ideas, code should carry the context of the conversation that generated it, and the work should be available to your colleagues (and their agents) as it happens. We'll walk through what this looks like in practice, and what we're building to make it possible.

2:50pm

SESSION **Software Factories · Main Stage** 2:50pm-3:10pm

Notion's Token Town

Sarah Sachs — Eng Lead, AI, Notion

SESSION Claws & Personal Agents · Track 1 2:50pm-3:10pm

Your company brain will leak secrets. Here's how we stopped it for big banks and ourselves.

Tanmai Gopal — CEO/cofounder, PromptQL

Everyone wants a shared "company brain", one single AI that knows everything the org knows. But it's nearly impossible to build one, because the moment AI scrapes everyone's data into one place, a single wrong answer to the wrong person is a breach. The downside of modifying a above-my-pay-grade shared skill, or leaking confidential information to the wrong colleague is catastrophic. Ergo, company brain projects can only ever ship to the few people who already had access to everything, or stay hobbled with strictly public information (eg: River at Shopify). We've been building one for the last year and have successfully deployed for Fortune 100 banks, for distributed-operations orgs with global scale, and for ourselves as a 70-person AI-native startup. I'll leave you with a blueprint covering how we solved the following problems: 1. Permissions for shared data and tools 2. A shared context layer (skills, knowledge, semantic layer) with its own access control 3. Scoping the blast radius of wrong context 4. Auto-learning without auto-leaking If your company brain effort has been blocked by security, compliance, or just a healthy fear of the intern asking the AI a question and getting back the exec comp table, this is the talk.

SPONSOR Vision & OCR · Track 2 2:50pm-3:10pm

From VLM/VLA's to Embodied Agents

Armen Aghajanyan — Co-Founder & CEO, Perceptron AI

SESSION Search & Retrieval · Track 3 2:50pm-3:10pm

Where RL Will Take Search

Maximilian-David Rumpf — CEO, SID.ai · Lotte Seifert — Founder, SID AI

Search is having its Bitter Lesson moment. By turning search into an RL problem, we can finally scale search quality with compute! RL is extremely sample efficient when compared to classical search training objectives and we see no ceiling to how far we can scale this new paradigm. We cover the training of SID-1, the first RL-trained search model, and how search will look like post-RL.

SESSION Workshops Day 2 · Track 4 2:50pm-3:10pm

Setting Yourself Up for Success — Part 1

Jason Liu — Developer Experience, OpenAI, OpenAI

I will walk you through the process of understanding how Codex works as a general tool to control your computer (setting up your memory vault/ assistant threads, prompting it to talk to other threads, and exploring computer use), how to think about things like long running work streams, and preparing yourself to start thinking in loops.

SPONSOR Security · Track 5 2:50pm-3:10pm

It's 10pm. Do You Know Where Your Agents Are?

Kim Maida — Founding GTM Engineer, Keycard

Agents right now can sign legal contracts, run untethered, manage your dating profile, conduct financial transactions, and push code to production. Most agents have long-lived API keys and are dangerously overprivileged even when they're not making requests. In this talk, I'll demo how to solve the problem with the right access at the right time. You'll walk away knowing how to control agent access whether you're running coding agents from the CLI, building MCP servers, or connecting agents to third-party APIs.

SESSION **Voice & Realtime AI · Track 6** 2:50pm-3:10pm

Realtime Voice Agents with Frontier Intelligence

Bohan Li — Staff Software Engineer, EliseAi

Dive into how the EliseAI voice agent harness orchestrates multiple models with jagged capability profiles to achieve realtime latency without sacrificing intelligence. Reduces p90 effective latency overhead of ASR, TTS, and tool calling to sub 200ms, unlocking frontier models like GPT 5.5 for voice. ### ASR: Eager Speculative Transcription We introduce speculative transcription by pairing local Whisper or Parakeet fine-tunes for speed with API models like Scribe, Nova, or Gemini Flash for accuracy. A local content match classifier operates at sub 10ms latency, allowing us to immediately trigger the downstream pipeline from the fast local transcription and dynamically replace text with the more accurate transcription if significant differences occur. This process runs on a eager 100ms VAD delay, securely releasing the generated response audio only after a fixed silence threshold has passed. ### LLM: Async background tool injection To eliminate expensive tool calling round trips, we implement system leveraging async background tool injection where the primary model makes no direct tool calls. Instead, local fine-tuned tool-calling models continuously observe the realtime transcription stream in the background. "Fake" tool call traces are then injected into the primary LLM's context, which primes it for immediate, one-shot response generation. ### TTS: Prefix caching and infilling Many Agent responses start with the same set of 3-6 words. We can cache this audio, releasing it immediately while we infill the remaining response audio conditioned on this prefix to preserve speech prosody. With this approach, a relatively small cache can achieve a 90% hit rate across a wide range of voices, languages and model providers.

SESSION **LLM Recsys · Track 7** 2:50pm-3:10pm

From approval loops to autonomous agents with Docker pt4

John Craft — Solutions Engineer, Docker

You've invested in the best models, coding agents, and AI tooling. Now comes the hard part: unlocking autonomous development without creating security headaches, governance gaps, or endless approval loops. In this 90-minute hands-on workshop, you'll learn how to run coding agents in isolated environments built for autonomous work, create a 'golden path' for AI-assisted development across your organization, reduce software supply chain risk with secure, hardened containers, manage multiple agents with the right permissions and guardrails, and scale AI-powered development without slowing developers down.

SESSION **Forward Deployed Engineering · Track 8** 2:50pm-3:10pm

Forward Deployed Engineering 101

Kevin Bai — Member of Technical Staff, Anthropic

SESSION **AI Architects: Show my Workflow · Track 9** 2:50pm-3:10pm

When Will The Benchmaxxing Plague End?

Nick Heiner — VP of RL Environments, Surge AI

Model releases are heralded by a flourish of trumpets, a chorus of weeping angels, and often, inflated benchmark claims. Why do benchmarks so often not reflect real-world value? Is it intrinsic to the science of benchmarking, or just the consequence of our current practices? Is LM Arena a cancer on AI?

SESSION **AI-Native Enterprises · Leadership 1** 2:50pm-3:10pm

From AI-Assisted to AI-Native: Building a Frontier Development Team

Clare Liguori — Senior Principal Engineer, Amazon Web Services

When features that took two weeks now ship in an afternoon, the bottleneck shifts from writing code to making decisions. Frontier teams have discovered this firsthand, achieving 3-10x productivity gains by fundamentally rethinking how developers work with AI agents. This talk covers the practices that separate frontier teams from those who merely "sprinkle" AI on their existing workflows: running agents asynchronously for hours, investing in comprehensive agent steering files, enabling local integration testing for agent self-correction, and automating everything from coding to operations to documentation. You'll learn how teams at Amazon slowed down to speed up, the temporary productivity dips they accepted, and the organizational changes required to sustain this velocity.

SESSION **AI Architects: Show my Workflow · Leadership 2** 2:50pm-3:10pm

How I automate my own job at Hugging Face using agents

Niels Rogge — Machine Learning Engineer, Hugging Face

This talk will showcase how I automated a large part of my own job at Hugging Face. This involves both open (GLM-5.1) and closed-source models (Claude, Gemini), the Claude Agents SDK, serverless infra like Modal and Hugging Face Jobs. I will also discuss how I use agentic coding tools like Cursor and Codex to implement AI agents which automate my job, and how everything is connected to the internal Slack of Hugging Face.

SESSION **Expo Stage 1 NE** 2:50pm-3:10pm

6 Pillars of an Agentic Harness That Fixes Production Incidents

Varun Krovvidi

A model delights us when any plausible answer works, but a production incident has one right answer, and the model alone can't reliably reach it. Getting there depends less on the model and more on the orchestration, context, and judgment built around it. That work is harness engineering, and it is the new frontier. This session breaks down the six pillars of an agentic harness required to fix production incidents: model orchestration, context, reasoning, actions, learning, and evals. Join Resolve AI to walk through what each one does, why a better model doesn't make any of them go away, and how they compose to find the root cause of a live incident across massive context, under a clock, with real revenue on the line.

SESSION **Expo Stage 2 NW** 2:50pm-3:10pm

Video Discovery for Agentic World-Model Training

Rafael Levi — DevRel, Bright Data

Physical AI had its “Attention Is All You Need” moment with the rise of Vision-Language-Action models. The next bottleneck is data: not just more video, but the ability to find the exact real-world moments that teach models how the world works: gravity, motion, causality, human behavior, and object interactions. This session explores a new approach: discovering specific scenes from the vastness of the web. We’ll show how teams can search for moments like objects falling, people interacting with environments, or actions unfolding over time, then collect and structure only the relevant clips for training and evaluation. Attendees will learn how scene-level discovery changes multimodal data pipelines, reducing wasted collection, processing, storage, and review, while making it easier to build targeted datasets for VLA systems, robotics, physical AI, and agentic world models.

SESSION **Expo Stage 4 SE** 2:50pm-3:10pm

Self-Driving Production: AI Wrote your Code. AI Should Fix It, Too

Self-driving production is the next frontier of autonomous software development but getting there is a journey. In this session, we ll show how enterprises are progressing from manual operations and AI copilots toward closed-loop, autonomous production systems with Traversal.

3:20pm

SESSION **Software Factories · Main Stage** 3:20pm-3:40pm

fighting slop with slop

Vaibhav Gupta — CEO, Boundary

We haven't done a code review in two years. The last time I read every line of code in a PR was about six months ago. And we build a programming language with a runtime meant to replace V8. This is real engineering: compiler internals, runtime behavior, type systems, codegen, concurrency semantics, and FFIs across multiple languages. The thing that makes this possible is a technique we call "fight slop with slop" - every line of code is analyzed in depth by a sprawling toolchain of custom visualizers, linters, test snapshots and a whole bunch more. While the core language VM code has super high standards, a lot of these meta-tools are mostly vibe-coded. I'll dive deep into all the tactical things we've built, and how to adopt "fight slop with slop" in your own team

SESSION **Claws & Personal Agents · Track 1** 3:20pm-3:40pm

Every Harness Will Become A Claw

Sam Bhagwat — Founder/CEO, Mastra

Most of the Harness discussion is just a reprise of Context Engineering from last summer. But it's not 2025 anymore. We live in a Claude Code world, and the best way to think about a harness is Context engineering + Coding Agents = Harness. Harnesses are a magical DX because of specific features like planning mode, parallel subagents, skills, background tasks etc. But it doesn't stop there. People are shoving their harnesses in a box, making them listen to external events, giving them channels (the ability to ping its users), and a heartbeat. They are making them into Claws. And actually, harnesses _want_ to become claws, so they can take up more share of mind, suit collaboration workflows, and be available afk. I propose "Steinberger's law", a spinoff of Zawinski's law: every harness will expand until it becomes a Claw

SPONSOR **Vision & OCR · Track 2** 3:20pm-3:40pm

From Scratch to SOTA: Training a 3B State-Space Vision Model for 1.4 Billion People

Krishna Prasad Srinivasan — Head of Vision Models, Sarvam

India has 22 official languages. Across those languages live over a billion people whose knowledge is locked inside scanned images in scripts that most frontier models perform poorly. The problem is dire - until now, there wasn't even a comprehensive benchmark to measure Indic OCR performance, let alone training data at scale. When Sarvam AI set out to solve this, we had to build the infrastructure before the model, creating the first ground-truth benchmark for Indic document intelligence. In this talk, Krishna Srinivasan, who led the Vision Models team to build India's first sovereign VLM from scratch, will walk through the end-to-end engineering lifecycle. We will cover: (a) Architecture: Why we chose a 3B-parameter state-space architecture over transformer baselines to handle high-resolution visual inputs with minimal memory overhead and faster inference. (b) Training Pipeline: The exact recipe we used: starting with text-only pre-training, moving to continual pre-training with text and images, followed by SFT. Finally, we'll cover the advances we made in implementing large-scale RL with Verifiable Rewards for visual tasks in just 3 days using deterministic character-level reward signals. (c) Compute Efficiency: How we trained a frontier-competitive multimodal model with extreme capital efficiency, optimizing distributed training and GPU cluster management to punch far above our compute class. (d) Agentic Workflows: How this model powers Sarvam Akshar, a first-of-its-kind agentic document intelligence workbench featuring visual grounding and automated proofreading loops. The results speak for themselves: Sarvam Vision achieves best-in-class global scores (84.3% on olmOCR-Bench, 93.28% on OmniDocBench) and dominates Indic OCR. Attendees will learn the blueprint for compute-efficient multimodal training, and deploying state-space VLMs for population-scale enterprise workloads.

SESSION **Search & Retrieval · Track 3** 3:20pm-3:40pm

Stop Chunking Like It's 2022

Yuval Belfer — Sr. Developer Advocate, AI21 · **Niv Granot** — Tech Group Lead, AI21 Labs

Every RAG system bets everything on a single chunk size. 500 tokens? 800? Pick wrong, and half your queries fail before they start. But here's what nobody tells you: all the picks are wrong; there is no single chunk size that works for all queries. We ran oracle experiments across meeting transcripts, story chapters, and TV scripts. The result? Queries disagree violently on what chunk size works best - sometimes by 40 percentage points. Your "tuned" chunk size isn't a compromise; it's systematic underperformance. In this talk, we'll expose why fixed chunking fails and show you a dead-simple fix: index at multiple chunk sizes, aggregate at retrieval time using Reciprocal Rank Fusion. No retraining. No LLM overhead. Just 1-37% better recall across benchmarks by letting queries vote with their ranks instead of forcing them into one-size-fits-all boxes. Walk away knowing exactly when your chunk size is sabotaging you - and how to stop leaving 20-40% of your retrieval performance on the table.

SESSION **Workshops Day 2 · Track 4** 3:20pm-3:40pm

Setting Yourself Up for Success — Part 2

Jason Liu — Developer Experience, OpenAI, OpenAI

I will walk you through the process of understanding how Codex works as a general tool to control your computer, how to think about things like long running work streams, and preparing yourself to start thinking in loops.

SPONSOR **Security · Track 5** 3:20pm-3:40pm

AI's Jurassic Park Period

Aaron Stanley — CISO, dbt Labs

Early in my career, I accidentally and unrecoverably changed data I was collecting for a federal investigation. Twenty years later, with the help of AI and a career's worth of experience as a security leader, I intentionally did the same thing. Make no mistake, what my agent and I did together was dangerous. It was only because I had enough subject matter expertise in both the functional and risk issues that I could navigate it safely. We are in AI's Jurassic Park period: no matter how clearly we define the rules, models will search for paths to completion. And they are very good at making those paths look safe, reasonable, and correct even when they violate policy or basic intuition. Designing the right control set is about allowing for the right expertise to be injected at the right time in the co-creation process so we can move quickly and safely into the next evolution.

SESSION Voice & Realtime AI · Track 6 3:20pm-3:40pm

"My name is... my name is...": A Linguistic Map for Building and Debugging Voice Agents

Midam Kim — ML Engineer, ServiceNow

Every voice AI engineer has heard it: a caller repeating their name three times, getting more frustrated with each attempt. The logs look clean. Confidence scores look fine. Linguistics can help solving the mystery. By the end of this talk, you'll have a diagnostic framework for the failures that slip past standard metrics, a way to turn "the agent just didn't get it" into concrete, debuggable failure modes. The framework maps three levels of linguistic structure (sounds, words, and interactions) against the two dimensions every voice agent engineer already works in: what we hear (speech recognition) and what we speak (speech synthesis). That 3×2 grid surfaces problems your current tooling can't see, including: 1. Why your user cannot make your system understand their name 2. Why a single well-intentioned vocabulary hint can cause catastrophic drops in a non-English language 3. Why a transcript that's "cumulatively correct" can still ruin the user experience Drawing on examples from production multilingual voice AI work, I'll show where linguistic expertise connects to the engineering decisions you're already making and where it reveals failure modes that confidence scores will never warn you about. Who this is for: Voice AI engineers, ML practitioners on Voice AI pipelines, and anyone who's watched clean logs while their agent quietly fails real users.

SESSION LLM Recsys · Track 7 3:20pm-3:40pm

From approval loops to autonomous agents with Docker pt5

John Craft — Solutions Engineer, Docker

You've invested in the best models, coding agents, and AI tooling. Now comes the hard part: unlocking autonomous development without creating security headaches, governance gaps, or endless approval loops. In this 90-minute hands-on workshop, you'll learn how to run coding agents in isolated environments built for autonomous work, create a 'golden path' for AI-assisted development across your organization, reduce software supply chain risk with secure, hardened containers, manage multiple agents with the right permissions and guardrails, and scale AI-powered development without slowing developers down.

SESSION Forward Deployed Engineering · Track 8 3:20pm-3:40pm

How Forward Deployed Engineering is done at Kepler

Vinoo Ganesh — CEO & Co-Founder, Kepler

SESSION Data Quality · Track 9 3:20pm-3:40pm

Building Worlds for Models

Nicolai Ouporov — CEO, Fleet

Hold for Fleet AI. Company focuses on simulated environments / training gyms for AI agents and fits the posttraining / RL environments theme.

SPONSOR Track M 3:20pm-3:40pm

Surviving Your Own Velocity: How VS Code Ships Weekly with 40 People

Harald Kirschner — Principal Product Manager, Microsoft

A ~40-person team ships VS Code weekly to millions of users. Models got good enough to lean on, and leaning in is exactly what broke our process. This talk is the part most AI talks skip: what you have to rebuild after agents start working. We had to scale three things at once: how fast we ship, how we hold quality, and how fast we learn, and each one we fixed revealed the next. I'll walk through the harnesses, evals, and self-healing systems that keep velocity from becoming regression, and the patterns you can steal.

SESSION AI-Native Enterprises · Leadership 1 3:20pm-3:40pm

How to Get Your Org to Adopt Coding Agents (Without Shipping Garbage)

Eyal Blum — Software Engineer, Figma

AI coding agents promise 10x. On complex, production work inside a real org, the honest number is 2-5x — and getting there requires a journey most teams aren't prepared for. At Figma, we ship AI products to millions of users, but internally our engineering org is spread across three stages of adoption. The honeymoon, where AI is magic. The crash, where AI writes bad code and your best engineers are stuck protecting the quality bar. And the real skill — 2-5x with disciplined development practices and proper investment. This talk covers why adoption is uneven, what the trust curve looks like from the inside, and what leaders can do about it: guide teams to align on plans before generating code, set honest expectations, invest in the fundamentals that make codebases agent-friendly, and create space for skeptics without judgment. You'll leave with a framework for driving adoption more organically without mandating it — and without shipping garbage.

SESSION AI Architects: Show my Workflow · Leadership 2 3:20pm-3:40pm

Your Fine-Tuned Model Is Tech Debt: A 50x ROI House of Cards

Dan Bjornn — Senior Data Scientist, Lease End

We built an AI application on top of fine-tuned models that generated $12M in revenue at 50x ROI. It was fast, cheap, and impressively accurate. Then it started having problems. Small errors accumulated. The model misread intent and nuance, handling conversations wrong. But retraining was too costly to justify for each fix, so known bugs piled up until we hit critical mass. Each retraining cycle took a week end-to-end, most of it spent curating data and validating our classification pipeline. And fixes caused whack-a-mole regressions across intents that required multiple iterations per cycle. Over time, the model became increasingly rigid. Each retraining was harder than the last. Then our team started using Claude Code, and we realized context management was the real lever, not model specialization. We rebuilt on frontier models using well-crafted system prompts and progressive context management, feeding the agent only what it needs when it needs it. Adjustments that used to require a week-long retraining cycle now take a small context change. Fine-tuning should be a last resort, not a first instinct. The cases where it's the right call are far fewer than they used to be. Before you fine-tune, ask: can I solve this with better context instead?

SESSION Expo Stage 3 · Expo Stage 1 NE 3:20pm-3:40pm

Can Your Agent Hear You Now?

Thor 雷神 Schaeff — Member of the Technical Staff (DevX) at Google DeepMind, Google DeepMind

SESSION Expo Stage 2 NW 3:20pm-3:40pm

From Context to Memory: Your Agents Need a Real Memory Layer

Anders Swanson — Developer Evangelist, Oracle

Most agents don't really have memory. They have a context window, a pile of temporary files, maybe an AGENTS.md, and a retrieval step that attempts to build state from whatever the model can still see. You've seen the flashy demos, but these systems fall apart when an agent needs to recover from failure, revisit prior work, and observe if failures are less frequent over time. This talk explores agent memory as a systems problem. Effective memory isn't just storing data: it's an evolving knowledge layer with write filtering, consolidation, reflection, and forgetting. Agents need persistence, and they also need structure. Raw logs and Markdown scratchpads aren't enough. A real memory layer weights recency, combines retrieval techniques, and correlates episodic memories. Serious agent memory is inherently multi-model. The best systems use full-text search, semantic retrieval, graph relationships, and structured state to reconstruct context with far more precision than filesystem grep alone. This is where databases become essential as the foundation for real memory. Memory shapes how agents behave, adapt, and improve over time.

SESSION Expo Stage 3 SW 3:20pm-3:40pm

Running a 20T-Token Data Pipeline: Infrastructure Lessons from Production

Bogdan Gaza — Co-Founder & CTO, DatologyAI

The problem. Curation algorithms tend to get the spotlight: model-based quality filtering, embedding-based deduplication, synthetic generation at scale, target distribution matching. The engineering behind them, the systems that actually run those algorithms reliably on petabytes of data and thousands of GPUs, usually gets overlooked. This session is about the engineering. What we built. The infrastructure behind two production data curation pipelines, on two very different shapes of workload: Arcee Trinity-Large-Thinking three model generations in nine months, with the curated corpus scaling from 8T to 10T to 20T tokens. Trinity-Large's 20T-token corpus included 8T+ synthetic tokens generated on clusters peaking at 2,048 H100 GPUs. Each generation incorporated deeper curation and broader domain coverage; the pipeline ran end-to-end multiple times, not once. Thomson Reuters legal 100B tokens of mid-training output, generated from TR's proprietary legal corpus, delivered as a deployment artifact and plugged into their existing SFT and DPO post-training. Different operational profile entirely: smaller scale, sensitive data, customer-environment integration. What you'll learn about. The metadata bottleneck. At trillion-token scale, fetching metadata from object storage across millions of files becomes the dominant source of idle time. We offload metadata management to Spark and use a lightweight file-level distribution scheme to drive idle time to near zero. Fault tolerance at multi-week scale. Long-running GPU inference jobs fail. We use one-to-one partition mapping between Spark and Ray jobs to get idempotent, resumable execution. A node failure no longer means reprocessing the dataset. Heterogeneous workload scheduling. Curation pipelines mix CPU-heavy preprocessing (Spark) with GPU-heavy inference (Ray + vLLM). An in-house scheduler routes each job type to isolated node pools, preventing resource fragmentation and ensuring critical training jobs aren't blocked by upstream CPU work. Inference tuning across models. vLLM defaults aren't right for every model. Tuning batch size, speculative decoding, and n-gram sampling per-model yields up to 40% throughput improvement, without over-engineering. Pipeline reproducibility. Treating a curated training corpus as a versioned deployment artifact rather than a one-off output. What that enables when a customer wants to run mid-training against a pre-trained base. For engineers building or operating large-scale data pipelines for ML training

SESSION **Expo Stage 4 SE** 3:20pm-3:40pm

From raw documents to AI-ready data

Leo Platzer — Founder, Deasy Labs / Collibra

Starting from a real document corpus full of overlapping, look-alike files, we walk through what it takes to make retrieval on those files reliable, from deduplicating to enriching with metadata. Watch how each step reshapes the vector space, and what happens to the answers that come back.

3:45pm

SESSION **Software Factories · Main Stage** 3:45pm-4:05pm

Loop Engineering from first principles

Kyle Mistele — CTO, HumanLayer

Code is free, software is infinite, and agents can do it all - that's the promise of the lights-off software factory, where humans interact only with tickets & specifications, and nobody reads the code, let alone writes it. We ran our own for six months, and we have the scars to prove it - bad code compounded, and agents created problems that agents couldn't solve - until we had to throw it all away. But this is a survivor's guide, not an obituary. In this talk, we'll share the challenges we encountered, what we liked, what we hated, what we're still doing, what we stopped doing, and what we started doing afterwards.

SESSION **Software Factories · Track 1** 3:45pm-4:05pm

Gadgets: Personal app vibe coding that is actually safe

Kenton Varda — Principal Engineer, Cloudflare

We are entering the end game of Kenton's 15-year master plan. The architect of Cloudflare Workers, Durable Objects, Cap'n Proto, and Sandstorm.io, and the guy who coined the term "Code Mode", will demo Gadgets, an AI productivity suite which ties all these ideas together. We've all heard that the future is micro-apps customized for every niche, but how do we actually make that usable, how do we make it scale, and most importantly, how do we make it safe for even non-developers to use? Kenton will show how Gadgets solves these problems, including a sandbox design that makes it essentially impossible for apps to have vulnerabilities at all.

SESSION **Workshops Day 2 · Track 4** 3:45pm-4:05pm

Setting Yourself Up for Success — Part 3

Jason Liu — Developer Experience, OpenAI, OpenAI

I will walk you through the process of understanding how Codex works as a general tool to control your computer, how to think about things like long running work streams, and preparing yourself to start thinking in loops.

SPONSOR **Security · Track 5** 3:45pm-4:05pm

Secure Cloud Compute

Ethan Sutin — Co-founder, Bee (acq. Amazon)

SESSION **Voice & Realtime AI · Track 6** 3:45pm-4:05pm

Act, Confirm, or Stop? Smarter behavior for AI assistants, wearables & robots

Amit Desai — Director, Voice & Assistant AI, Roku

Voice is our favorite way to command AI assistants and robots — and it is error-prone. The industry's reflex is to chase accuracy, but accuracy is only one knob: we can control system behavior in other ways to increase user satisfaction. This talk shifts the lens from accuracy to user outcomes. Give the AI agent more than one move: besides acting, let it stop, reject, confirm, clarify, or disambiguate. The question stops being "how often are we right?" and becomes "what does each outcome cost the user?" Bad outcomes are not equally bad to users — so price them relatively, then have the AI system minimize that user cost. Call it OUCH: Outcome User Cost Heuristic; we optimize system behavior to minimize the OUCH. Same accuracy, lower user cost, greater user adoption. We will walk through practical AI assistant examples illustrating this approach, then show how the same framework extends across AI environments — smart speakers, TVs, glasses, embodied AI, robots, wearables, and vehicles — by repricing outcomes and swapping the confirmation UI. Why this matters now: the cost of voice-command errors is escalating as we move into AI assistants and embodied AI, where wrong actions can be more expensive and dangerous. Mainstream voice adoption will not come from chasing accuracy alone; we need systems to price in the cost of being wrong.

SESSION **Data Quality · Track 9** 3:45pm-4:05pm

Data and Environment Curation for Post-training LLMs

Mahesh Sathiamoorthy — CEO, Bespoke Labs

Hold for Bespoke Labs. Company works on data curation, eval tooling, and reinforcement-learning environment curation for agent development.

SESSION **AI Architects: Show my Workflow · Leadership 2** 3:45pm-4:05pm

Unlock Agent Autonomy: The Runtime for AI-Native Systems

Tushar Jain — EVP of Engineering, Docker

The way software gets built in 2026 doesn't look like it did in 2024. The actors changed. Agents read and write entire codebases. Subagents spawn to chase down a flaky test, refactor a module, or triage an incident. But this shift doesn't stop at the SDLC. Agents increasingly invoke tools, interact with enterprise systems, install dependencies, call APIs, and orchestrate workflows across local machines, CI systems, cloud infrastructure, and organizational boundaries. The teams leaning into this shift are moving faster, and the gap is widening by the quarter. But few have the confidence to let agents operate autonomously across those environments. Not because the model capability isn't there. Trust isn't. Agents can pull a poisoned dependency, invoke an untrusted tool, wipe a database, leak sensitive data, or access systems they shouldn't. Prompt-level instructions won't close that gap, the unlock has to happen one layer down, at the runtime layer itself. Docker spent the last decade making it safe to ship software by getting the runtime right: isolation, network policy, trusted base images, and credentials. Agents are the next workload, and the same principles apply. Tushar Jain, EVP of Engineering at Docker, walks through what the runtime layer for AI-native systems looks like in practice: hardened runtime foundations, sandboxes that constrain what agents can touch, and governance controls that limit what agents can introduce, access, and execute across local, CI, cloud, and enterprise environments. The pattern is the same on every vector: reduce the surface area of what the agent gets to decide, so the parts that matter aren't left to a prompt. Attendees leave with a clearer framework for giving agents more autonomy safely. Engineers see how agentic applications can operate across tools and infrastructure. Security leaders get a runtime model that maps to controls they already understand. Platform teams get a way to scale agent execution without standing up a new runtime for every team.

SESSION **Expo Stage 1 · Expo Stage 1 NE** 3:45pm-4:05pm

How We Built the Airbyte Agent MCP Server and CLI

Pedro Lopez — Senior Software Engineer, Airbyte

Agents need a reliable way to reach live business data. At Airbyte we built two interfaces for that, and this session is how. Cam built much of that surface. He covers the MCP server that exposes hundreds of sources through one endpoint with managed auth, and the CLI that's designed for agent harnesses rather than humans, with embedded help, packaged agent skills, and no credentials passed over the command line. Expect the real engineering: why a CLI turned out to fit autonomous agents better than the API or SDK, how auth works across the layers, and the tradeoffs the team made along the way. Come if you're building agent tooling or thinking about how to expose your own systems to agents cleanly.

SESSION **Expo Stage 2 NW** 3:45pm-4:05pm

From Chatbots to Agents: How Reducto builds for Agent Experience to Enable Real Work

Abhi Arya — Product, Software, Infra, and Applied AI, Reducto

Many agent demos work. Most agent systems in production don't. The gap usually isn't the model or the tools. It's everything in between: how context gets structured, how multi-step tasks stay on track, how you handle the edge cases that only show up when real scenarios from real customers hit your pipeline. At https://reducto.ai/, we've spent the last couple of months building agent-first workflows for some of the most document-heavy industries out there. We've hit most of the failure modes you're probably hitting too. This talk shares what we've learned, from how to think about Agent Experience (AX) as a design layer, to the specific decisions that make complex workflows actually reliable in production. You'll walk away with tactical approaches to structuring context, model guidance, designing recoverable workflows, and building the feedback loops that let your system improve over time without a full rebuild.

SESSION Expo Stage 3 SW 3:45pm-4:05pm

Towards Reliable Financial Agents: How a 4B Model Outsmarted a 235B Giant

Charlie Dickens

Large generalist models have excellent reasoning but this does not necessarily imply specialized knowledge and tool calling capabilities. They can still hallucinate column names, ignore constraints, and generate SQL that returns nonsensical results. The problem isn't intelligence it's reliability and specialization. In this talk we'll show how a 4B model was fine-tuned to outperform a 235B model on real financial analysis tasks. The key was not adding more reasoning ability, but enforcing tool discipline. Using synthetic data generation and reinforcement learning with the open-source rLLM framework, the model learned to explore schemas, validate outputs, and retry failures instead of hallucinating confident nonsense. One key result: tool-use fundamentals generalize. Training on simple tool interactions transferred to much harder, multi-step financial tasks. If you're building LLM systems that interact with databases, APIs, or internal tools, this talk focuses on the behaviors that actually matter and how to teach them without frontier-scale compute.

SESSION Expo Stage 4 SE 3:45pm-4:05pm

AI Enablement at Automattic: How a Remote Company Builds AI Fluency

Em Shreve

Automattic is a remote company. About 600 of us will step away from regular work this year for an immersive AI program. That's a little over a third of the company. This talk walks through a field report of what we built and why: the curriculum, the cohort design, and what we've learned about making AI fluency work across a distributed organization.

4:30pm

KEYNOTE Software Factories · Main Stage 4:30pm-4:50pm

Harness Engineering is not Enough: Why Software Factories Fail

Dex Horthy — Co-Founder, HumanLayer

4:50pm

KEYNOTE Harness Engineering · Main Stage 4:50pm-5:10pm

In Code They Act, In Proof We Trust

Erik Meijer — Research Scholar, Leibniz Labs

AI agents today execute on blind trust, and the failure modes are already in the headlines: a dealership chatbot agreeing to sell a $76,000 Chevy Tahoe for $1, a coding agent wiping a production database during a code freeze, an "agent skill" quietly installing a keylogger on a developer's machine. These are not edge cases. They are the predictable consequence of allowing agents to act without any mechanical guarantee of correctness or safety. Execution is irreversible. You cannot unsend a message, unwire a payment, or un-delete a database. In that regime, permitting an unsafe action costs far more than withholding a safe one, and thus the economically rational choice is to refuse to let agents act on unchecked intent alone. Automind is an agent harness that enforces this discipline by construction. Before any action runs, the agent must submit its execution plan together with a machine-checkable proof of safety and correctness, written in Universalis, a literate logic programming language designed to be read by humans and verified by machines. A small, auditable checker decides whether the plan is allowed to execute. By left-shifting the trust boundary, we no longer have to trust the agent's proposal, or even its proof; only the checker. Policy compliance becomes a static property, established before the first side effect. We can finally demand formal proofs, not vibes, from the agents we deploy.

5:10pm

KEYNOTE Software Factories · Main Stage 5:10pm-5:30pm

Recursive Model Improvement

Lee Robinson — ML, Model Behavior, Cursor

Day 3 — Session Day 2 Wednesday, July 1, 2026

contents ↑

9:05am

KEYNOTE **Autoresearch · Main Stage** 9:05am-9:25am

Field Guide to Fable

Thariq Shihipar — Claude Code, Anthropic

https://x.com/trq212/status/2027463795355095314

9:25am

KEYNOTE **Software Factories · Main Stage** 9:25am-9:45am

In the Land of AI Agents, the Verifiers Are King

Tariq Shaukat — Chief Executive Officer, Sonar

As AI agents take on increasingly complex development tasks, the critical challenge has shifted from generation to verification. Hallucination is not a temporary bug. Evidence suggests that as models grow more capable, failures become more frequent and more convincing, making cognitive surrender among human reviewers an acute risk. This talk introduces a three-stage discipline for responsible agentic development, Guide, Verify, Solve, and argues that rigorous verification infrastructure is both a safety requirement and a competitive advantage. Counterintuitively, code quality matters more in an agentic world: clean, low-complexity codebases make agents faster, cheaper, and more reliable, while technical debt compounds at machine speed.

9:45am

KEYNOTE **Autoresearch · Main Stage** 9:45am-10:05am

Perception Agents

Antje Barth — Member of Technical Staff, Amazon AGI Lab

Human-agent collaboration is changing, becoming more visual. The agents most teams ship today still wait for us to type a paragraph to explain what we're looking at. They cannot see a screen, navigate a UI that changes, or recover when an application throws an unexpected modal. That is the architectural gap between agents that demo well and agents that work alongside real teams in real software. Perception agents close it. They see and use computers the way people do, reason about what they see, and act with clicks and keystrokes.

10:05am

KEYNOTE **Autoresearch · Main Stage** 10:05am-10:25am

Research to Reality with Google DeepMind

Benoit Schillings — VP of Technology, Google DeepMind

TBD. Expected focus areas include generative AI for code, deep thinking algorithms, and the future of pre-training and transformer models for Gemini.

10:25am

KEYNOTE **Autoresearch · Main Stage** 10:25am-10:30am

Evals Track Intro

Laurie Voss — Head of Developer Relations, Arize AI · **Aparna Dhinakaran** — CPO, Arize

10:45am

SESSION **Autoresearch · Main Stage** 10:45am-11:05am

First Steps Toward Automated AI Research

Richard Socher — CEO & Co-Founder, You.com / Recursive Superintelligence

SESSION **Sandbox & Platform Engineering · Track 1** 10:45am-11:05am

Don’t build agents, build environments

Adam Azzam — Member of Product Staff, Modal

We’ve largely settled on what a coding agent is: a model working in a loop, calling tools. As a result, the hard part has moved. It’s no longer the agent loop, it’s the environment around it. This talk is about the real challenges of building fast-booting, reliable, reproducible environments for coding agents at scale.

SPONSOR **Robotics & World Models · Track 2** 10:45am-11:05am

Building the simulation infrastructure for practical world model use

Christopher Manning — Distinguished Member of Technical Staff, Moonlake AI

What is the most important capability for world model applications and the pursuit of embodied AI? We believe it is not a question of having the most beautiful pixels but the ability to reason about causality in multimodal environments. At Moonlake, we are working on building action-conditioned multimodal world models which provide spatial and physical state consistency over long time periods. We believe that building and training on synthetic worlds provides the data and compute efficient path to truly useful world models. We are building the simulation infrastructure platform for companies that need to build and manage worlds (assets, scenes, digital twins) at scale, including robotics/autonomy teams, digital factory operators, and game authors. Our product today primarily finds applicability in simulation and the operationalization of digital twins. Simulation can include training robotics, world models for AGI research, autonomous vehicles, or content creation for media and entertainment. Operationalization of digital twins involves the reconstruction of scans into reusable assets, e.g., turning image and point-cloud scans into sim ready assets for digital factory Integration projects. We are building toward a future where AI systems do not just generate worlds, but understand how they work. Moonlake learns from each workflow: The more workflows, failures, and human interventions that Moonlake sees, the better it becomes at reconstructing, validating, and preparing complex simulation worlds. The session will include discussion and demos.

SESSION **Memory & Continual Learning · Track 3** 10:45am-11:05am

Beyond Static Intelligence: Evaluating Continual Learning

Parth Asawa — CS PhD student, UC Berkeley

Continual learning, the ability of AI systems to improve through sequential experience, has attracted substantial interest, but no high-quality benchmark exists to evaluate it. We introduce Continual Learning Bench (CL-Bench), the first difficult, expert-validated benchmark designed to measure whether LLM-based systems genuinely improve with experience. CL-Bench spans six diverse domains (software engineering, signal processing, disease outbreak forecasting, database querying, strategic game-playing, and demand forecasting), each validated by domain experts and designed so that tasks share a learnable latent structure (codebase layout, disease outbreak dynamics, opponent strategies) that a stateful system can discover online but a stateless one cannot. We evaluate frontier models across several agent architectures, from naive in-context learning (ICL) to dedicated memory systems, introducing a gain metric to isolate learning from prior capabilities. We find that these systems leave headroom for improved continual learning: agents frequently overfit to immediate observations or fail to reuse knowledge across instances, and dedicated memory systems do not fix this---in fact, naive ICL outperforms systems dedicated to memory management. CL-Bench is the first benchmark to evaluate continual learning across diverse real-world domains with expert-validated tasks and isolate online learning from underlying model capability, showing a need for better continual learning systems.

SESSION **Workshops Day 2 · Track 4** 10:45am-11:05am

Build realtime multimodal agents with Gemini Live

Thor 雷神 Schaeff — Member of the Technical Staff (DevX) at Google DeepMind, Google DeepMind

The Gemini Live API is incredible versatile when it comes to building realtime AI experiences. From live translation across 2000 different language pairs to building realtime multimodal agents that can work across text, audio, and vision. This workshop gets you from zero to fully conversational agent in a matter of hours.

SPONSOR **Evals · Track 5** 10:45am-11:05am

Vending-Bench: Long-Horizon Agent Evals for a Simulated Vending Business

Lukas Petersson — Co-Founder, Andon Labs

Long-horizon agent evals via a simulated vending machine business, testing negotiation, pricing, and supplier management over 365 days.

SESSION **Design Engineering · Track 6** 10:45am-11:05am

Understanding is the new bottleneck

Geoffrey Litt — Design Engineer, Notion

Autonomous loops are hot, but the reality is that most agentic tasks still require human judgement. And to guide your agents well, it's not enough to just verify correctness -- you actually need to understand the work they're doing. In this talk, I'll share some techniques for staying in the loop and efficiently developing understanding, combining old ideas from education and cognitive science with modern agent capabilities. You'll walk away with some practical tips for moving faster with agents by understanding more, not less.

SESSION **Computer Use · Track 7** 10:45am-11:05am

Computer-use models will agentify the web, not APIs

Dhruv Batra — Co-founder & Chief Scientist, Yutori

We are rushing towards a world where every single digital surface (email, calendar, messaging, ..., every desktop app, every phone app, every web app) that was previously meant for humans is now managed by AI agents. Of course, there are technical challenges to be solved: - Model context windows haven't increased in 2 years. And the digital world is OOMs bigger (the ultimate "big world hypothesis") anyway, so how does one architect this? - A large part of the digital world (most of the web) does not have APIs available and requires agents to act like humans (consume pixels, output keyboard/mouse actions). - Human preferences and the digital world change, and require agents to maintain a dynamic memory and continually learn. But even if we could solve these problems, what does this world look like? - The digital world, particularly the web, was built for human consumption (and is often hostile to bots). - For a while to come, we will be sharing the digital roadways with these digital robots. - What does end-to-end encryption and privacy mean when the other "end" of the communication is an AI agent? The Yutori team has spent the last year building the world's best computer use model (slightly better than Opus 4.6 and GPT 5.4 while being 2x faster and 4-5x cheaper on browser use tasks), converted the web into a webhook with Scouts (agents that monitor the web 24/7 for anything you care about), and are now releasing Yutori agent that expands from the open web to your most common digital surfaces. This talk will be grounded in Yutori's learning from what it takes to build agents that are always on, taking us one step closer to the world where every digital surface is their playground.

SESSION **Context Engineering · Track 8** 10:45am-11:05am

Build-Time vs. Run-Time: Why Your Dev Tools Will Fail in Production

Averi Kitsch — Staff Software Engineer, Google · **Prerna Kakkar** — Senior Software Engineer, Google

A dangerous pattern is evolving in the ecosystem: developers are deploying "Build-Time" tools into "Run-Time" environments. In this session, we will introduce a critical distinction for the MCP ecosystem: the difference between Build-Time Agents (Developer Assistants like Gemini Code Assist) and Run-Time Agents (End-user applications like a Customer Support bot). Drawing from our experience building the MCP Toolbox, we will demonstrate why the "Atomic" tools that make Build-Time agents powerful become catastrophic liabilities for Run-Time agents. We will provide a framework for transitioning your architecture across three key axes: Design: Moving from flexible, atomic primitives to "Composite Workflows" that encapsulate business logic. Security: Shifting from "Developer Identity" (trusted) to "Workload Identity" (zero-trust), where the agent is treated as an untrusted user. Reliability: Why production agents need "Agent-Readable" errors (natural language guidance) rather than the stack traces that developers rely on. Attendees will leave with a clear rubric for evaluating whether their tools are truly "Production Ready" or just "Prototype Ready."

SESSION **Posttraining & Midtraining · Track 9** 10:45am-11:05am

What's next after RLHF?

Diogo Almeida — CEO, TypeSafe AI

RLHF was a massive commercial success: roughly 100% of LLM usage is through RLHF'd models - but it was in many ways also a research failure. Let's talk about how it conquered the world, how it defied its creators expectations, why AI is in the bimodal state it's in (is it a bubble or a machine god?), and how to make AI actually transform the economy.

SPONSOR **Track M** 10:45am-11:05am

From framework to runtime: running agents with Foundry Agent Service

Tina Manghnani — Product Manager, Microsoft · **Keiji Kanazawa** — Principal Product Manager, Microsoft

See how agents move from frameworks into production systems. Learn how Foundry Agent Service provides hosted execution, scaling, and lifecycle management—combining models, tools, and orchestration into a production-ready runtime.

SESSION **AI-Native Enterprises · Leadership 1** 10:45am-11:05am

How do you diffuse AI into the real world?

Varun Shenoy — Cofounder, Long Lake

Most AI conversations are still about models, benchmarks, and demos. We want to talk about what it actually takes to make AI work inside real companies. The gap between impressive demos and production value is where most enterprise AI efforts die. We've all seen burned budgets, cynical teams, and tools that never leave the pilot phase. We've spent the last two years closing that gap across the American services economy, and we'll share a bit of our playbook. This talk walks through three layers of what real AI deployment looks like, drawn from Long Lake's live operating environments: Measure: How we built domain-specific evals and workflows to improve performance on real HOA management tasks, not synthetic benchmarks, but metrics tied to actual business outcomes. Embed: How we put AI directly inside tools like Revit, meeting users where they already work instead of asking them to change how they operate. Scale: The enablement playbooks and operating techniques we use to help teams of property managers, payroll specialists, and more adopt AI in their day-to-day jobs. The broader theme is vertical superintelligence: not just better models, but systems built around proprietary data, workflow context, domain tools, human enablement, and continual learning. This talk is for builders and operators who care less about benchmark theater and more about how to deliver measurable outcomes, deal with change management, and teach non-technical workforces to use AI effectively in production beyond just Claude Code / Cowork.

SESSION **AI Architects: Tokenmaxxing · Leadership 2** 10:45am-11:05am

The Z/L Continuum: Should AI Engineers Still Read Code?

Alex Volkov — AI Evangelist & Host of ThursdAI, W&B from CoreWeave

At AI Engineer Europe, two of the best speakers gave directly opposite advice. Zechner: slow the f*** down, read every line your model writes. Lopopolo: code is a liability, you don't even open the IDE anymore. Both got applause. The room walked out confused. On the train back I sketched the Z/L Continuum on a napkin — a five-stop spectrum from "read the clanker code" to "what IDE?" — and the whole week clicked into place. In this talk I'll walk through the Continuum, introduce FOMAT (Fear of Missing Agent Time — coined backstage by Michael Richman), and make four arguments: the Continuum is real, your stop is per-task not per-person, model capability bends everything toward L, and FOMAT is a filter problem, not an agent problem. You'll leave with a vocabulary for the argument every AI engineer is having right now. Audience takeaways A shared vocabulary (Z, L, the five stops) for the debate splitting AI engineering teams FOMAT — name the fear so you can manage it A per-task framework for choosing where on the Continuum to operate Why capability drift makes "I'll never let it cook" a losing position over time Speaker: Alex Volkov · ThursdAI · @altryne

SESSION **Expo Stage 2 · Expo Stage 2 NW** 10:45am-11:05am

AI Engineering & Governance 2026 Trends

Wallon Walusayi — Qodo

AI Engineering & Governance 2026 Trends

SESSION **Expo Stage 3 · Expo Stage 3 SW** 10:45am-11:05am

Why AI Didn't Actually Make You Ship Faster

Gabriel Spencer-Harper — CEO, Meticulous

AI generates code faster than humans can review and verify it, and most engineering teams adopting codegen have hit the same wall: verification. In this session, Gabriel (CEO of Meticulous) breaks down why assertion-based testing has a structural ceiling that AI codegen has made impossible to ignore, what exhaustive verification actually requires technically (behavior capture, determinism, and backend isolation), and why the teams solving this now are the ones who will ship at the speed AI enables. The talk includes case studies from LaunchDarkly, which saw an 80% reduction in major frontend incidents after rollout, and Notion, which deployed verification infrastructure across every engineer on every PR to confidently adopt AI-generated code at scale.

SESSION **Expo Stage 4 · Expo Stage 4 SE** 10:45am-11:05am

Redesigning how software gets built

TBD — Sonar — Sonar

AI is already transforming how software is built, but most organizations are still treating it as a productivity tool rather than a governance challenge. The real question isn't whether to adopt AI-assisted development; it's whether your operating model is designed to control what comes out of it. This session reframes the AI development conversation around three practitioner horizons: organizations that are proficient with the status quo, those capturing velocity today, and those building toward the next frontier, where AI agents operate with genuine autonomy at scale. The gap between these horizons isn't model capability. It's operating model maturity. Most organizations are still applying AI to isolated steps in the development process. The real value only arrives when you redesign the system end-to-end: how work flows, how decisions are made, and how teams interact with AI as a core contributor. That transition requires something most teams haven't built: a governance layer that is accurate, consistent, repeatable, transparent, and auditable. This talk explores what that governance layer looks like in practice, including how to instrument controls at the point of generation, enforce standards without slowing agents down, and build the organizational confidence to let agents operate at scale without losing visibility or accountability. The companies getting the most out of agentic development aren't the ones with the best models. They're the ones with the strongest foundations. True governance isn't a gate at the end of the pipeline. In an agentic world, it's the architecture the pipeline runs on.

11:00am

SESSION **CTO Circle · Leadership Lounge** 11:00am-12:00pm

Tokenomics: From AI Spend to AI Value

Martin Harrysson — Senior Partner, McKinsey & Company · Matt Linderman — Partner, Technology Practice, McKinsey & Company · Prakhar Dixit — Partner, McKinsey

Facilitated, peer-to-peer, under the Chatham House Rule — not recorded. As enterprise AI adoption accelerates, token spend is scaling faster than value realization. We address i) how to make decisions amid unclear cost and value dynamics, ii) how to shift from token-level to workflow-level analysis, and iii) how to manage downstream behavior implications on AI usage.

11:10am

SESSION **Autoresearch · Main Stage** 11:10am-11:30am

Autoresearch for Dense Retrieval: Test-Time Compute with Frozen Embedding Models

Han Xiao — VP, AI, Elastic

Test-time compute is widely believed to benefit only large reasoning models. We show it also helps small embedding models. Since modern embedding models are distilled from LLM backbones, a frozen encoder should benefit from extra inference compute without retraining. Using an agentic program-search loop spanning 144 generations, we explore 144 candidate programs over a frozen encoder API. The search produces twelve Pareto-optimal programs spanning cost ratios of c=1.2 to 14.7 over the single-pass baseline. The programs are structurally diverse: the search independently rediscovers Rocchio pseudo-relevance feedback, ColBERT-style MaxSim at sentence granularity, reciprocal rank fusion, and the Fisher linear discriminant, all without trainable parameters or external models. Every frontier program improves nDCG@10 over the frozen baseline across all 14 MMTEB retrieval tasks spanning legal, financial, long-document, and general domains.

SESSION **Sandbox & Platform Engineering · Track 1** 11:10am-11:30am

Letting the Interns Loose — How We Accelerated AI Adoption.

Shashank Goyal — Head of Provider Ecosystem, OpenRouter

SPONSOR Robotics & World Models · Track 2 11:10am-11:30am

Building the simulation infrastructure for practical world model use (Part 2)

Christopher Manning — Distinguished Member of Technical Staff, Moonlake AI

What is the most important capability for world model applications and the pursuit of embodied AI? We believe it is not a question of having the most beautiful pixels but the ability to reason about causality in multimodal environments. At Moonlake, we are working on building action-conditioned multimodal world models which provide spatial and physical state consistency over long time periods. We believe that building and training on synthetic worlds provides the data and compute efficient path to truly useful world models. We are building the simulation infrastructure platform for companies that need to build and manage worlds (assets, scenes, digital twins) at scale, including robotics/autonomy teams, digital factory operators, and game authors. Our product today primarily finds applicability in simulation and the operationalization of digital twins. Simulation can include training robotics, world models for AGI research, autonomous vehicles, or content creation for media and entertainment. Operationalization of digital twins involves the reconstruction of scans into reusable assets, e.g., turning image and point-cloud scans into sim ready assets for digital factory Integration projects. We are building toward a future where AI systems do not just generate worlds, but understand how they work. Moonlake learns from each workflow: The more workflows, failures, and human interventions that Moonlake sees, the better it becomes at reconstructing, validating, and preparing complex simulation worlds. The session will include discussion and demos.

SESSION Memory & Continual Learning · Track 3 11:10am-11:30am

Scaling up Continual Learning

Ronak Malde — Co-Founder and CEO, Trajectory

Trajectory (stealth) is a research and product lab building the platform for continual learning, where frontier models are continuously trained as they interact with the real world. We are a team of ex-Deepmind, OpenAI, Meta superintelligence, Apple, and raised 15M from Conviction. The Fair will be after we have launched to the world. We will be walking through the primitives of continual learning, and how we can scale fast by leveraging these tools.

SESSION Workshops Day 2 · Track 4 11:10am-11:30am

Build realtime multimodal agents with Gemini Live (continued 2)

Thor 雷神 Schaeff — Member of the Technical Staff (DevX) at Google DeepMind, Google DeepMind

The Gemini Live API is incredible versatile when it comes to building realtime AI experiences. From live translation across 2000 different language pairs to building realtime multimodal agents that can work across text, audio, and vision. This workshop gets you from zero to fully conversational agent in a matter of hours.

SPONSOR Evals · Track 5 11:10am-11:30am

From Signal to PR: Anatomy of a Self-Improving Agent

Jason Lopatecki — CEO, Arize

What if your observability platform didn't just tell you something was wrong, but told you why, and opened a PR with the fix? We'll walk through how we built Autopilot at Arize: an autonomous investigation agent that triggers on monitor alerts or schedules, pulls traces into a working filesystem, runs root-cause analysis, and produces actionable assets: a PR with prompt or code changes ready for review. We'll cover the architecture decisions (cloud agents vs. sandboxed containers, AI harness + skills), why traces-on-a-filesystem is the key unlock for agent-driven debugging, and how we dogfooded the system on our own agent, Alyx, before shipping it to customers. You'll leave with a concrete picture of what "observability that fixes itself" looks like in practice, and where and why the human stays in the loop.

SESSION Design Engineering · Track 6 11:10am-11:30am

The Spatial Harness: Bringing Agents to the Canvas

Max Drake — Product Engineer, tldraw

What if chat is the wrong interface for managing agents? What if we're holding ourselves back by squeezing our thoughts and the way we work to into a one-dimensional, single-threaded interface? At a high level, this talk aims to present the work we've done at tldraw to build a spatial harness, or a way to allow agents to work on a canvas and collaborate with users and each other natively. This work represents important steps towards building better agent + canvas experiences, a product category we've seen explode in the recent months (Paper, Replit Agent 4, Google Stitch, etc). It's also not something I've really seen talked about elsewhere. See: - Multi-agent collaboration on the canvas (fairies.tldraw.com) - We've also recently brought code mode (https://blog.cloudflare.com/code-mode-mcp/) to the tldraw desktop app and MCP app.

SESSION **Computer Use · Track 7** 11:10am-11:30am

Computer Use at the Edge of the Statistical Precipice

Pierluca D'Oro — Founder, Programma Labs

Evaluating Computer Use Agents (CUAs) on interactive environments is fraught with methodological pitfalls that the field has yet to systematically address. We show that a 1MB replay script that blindly executes a recorded action sequence without ever observing the screen outperforms frontier models on prominent static benchmarks, and prove that its expected success rate is exactly equal to the source agent's pass@k in deterministic environments. We trace this and other failures to two root causes: non-principled environment design (static, unsandboxed, or unreliably verified environments) and non-principled evaluation methodology (naive aggregation and misuse of pass@k for stateful UI interactions). To address the first, we propose PRISM, five design principles for CUA environments and instantiate them in DigiWorld, a benchmark of 15 realistic sandboxed mobile applications able to evaluate agents in over 3.2 million verified unique configurations. To address the second, we develop an aggregation framework that correctly accounts for the nested structure of CUA benchmarks. All together, we show that principled environment design and rigorous evaluation methodology are not optional refinements but prerequisites for meaningful CUA research.

SESSION **Context Engineering · Track 8** 11:10am-11:30am

It’s Tokens All The Way Down: How RLMs are Different

Kevin Madura — Director, Advanced Technology, AlixPartners

Recursive Language Models represent an intuitive but distinctively important approach to how LLMs handle context. The practical implications are bigger than they first appear. Tasks that would traditionally require careful prompt engineering, custom agent scaffolding, or multi-step orchestration collapse into surprisingly simple, composable programs. In this talk, we’ll cover what makes an RLM distinct from a coding agent, explore where the abstraction shines and where it breaks down, and walk through concrete use cases that are informed by real-world situations at scale. We’ll see side-by-side comparisons to understand trade-offs in complexity, performance, time, and token usage.

SESSION **Posttraining & Midtraining · Track 9** 11:10am-11:30am

State of Data

Sean Cai — CEO, Independent / State of Data

SESSION **AI-Native Enterprises · Leadership 1** 11:10am-11:30am

How to avoid disaster when vibe-coding a billing engine

Andrew Garvin — Cofounder of Metronome, Stripe

This talk covers what that infrastructure looks like in practice: which primitives matter, where the human checkpoints belong, and what changes when your billing system needs to be legible to machines instead of configured by humans clicking through a UI. When building AI products, billing and pricing should be directly tied to the products themselves. They're in the hot path. Every token, every agent action, every inference is a billable moment, and if your entitlement checks aren't keeping up, a single runaway agent can rack up thousands of dollars in seconds with no one to send the bill to. Get metering wrong and you're either eating costs or overcharging customers. Get ledger consistency wrong and your invoices don't add up. Get tax wrong across 47 jurisdictions and you find out from a regulator, not a user. Here's the thing, though — agents are legitimately good at billing strategy. They can pick pricing models, configure plans, run simulations, and iterate on packaging way faster than a human team could. You want them doing that work. But proration, multi-currency, revenue recognition, tax — this stuff took the industry years to get right, and it's unforgiving when you get it wrong. The question then becomes not whether agents should be making billing changes, it's what they should be operating on when they do. Agents need tight, composable building blocks where the correctness is already baked in, human-in-the-loop checkpoints before anything irreversible goes out the door, and sandbox environments where they can experiment freely without torching production. That's the architecture that lets you move fast on pricing without waking up to broken invoices. Target audience: Engineers and technical founders building AI products that charge for usage — whether that's per-token, per-action, or per-seat with consumption overages. If you've ever hard-coded a pricing tier, duct-taped metering onto an existing system, or wondered how your billing setup is going to survive your next pricing change, this talk is for you. Audience takeaways: - A clear understanding of why billing for AI products sits in the hot path — and what specifically goes wrong when metering, entitlements, or ledger consistency can't keep up. - A practical architecture for making billing agent-operable: composable primitives with correctness baked in, human-in-the-loop checkpoints on irreversible actions, and sandbox environments for safe experimentation. - A framework for deciding where agents should be empowered to move fast on billing strategy and where guardrails need to be non-negotiable.

SESSION **AI Architects: Tokenmaxxing · Leadership 2** 11:10am-11:30am

Is Orchestration the Future?

Vlad Luzin — Founder, Band.ai

ChatGPT, Claude Code, OpenClaw — three inflection points that reshaped the industry in two years, each pointing the same way: the next step is many agents, not one. Which raises the question nobody's answered well yet — how do many agents actually work together? Today's answer is orchestration, and it's genuinely good — until you need stateful peers holding a single conversation together, which none of them are built to do. So we'll make a different case: that the next inflection point is a collaboration layer that lets separate agent systems share one conversation as stateful peers, whatever they're built on. We'll show that this is the inflection point the last three were leading to with a demo and a real enterprise use case.

SESSION **Expo Stage 1 NE** 11:10am-11:30am

Harnessing Agents: The Durable Runtime for Dynamic Workflows

Viren Baraiya — Co-Founder and CTO, Orkes

Agents increasingly generate and revise workflows at runtime instead of following control flow written in advance. That breaks a common assumption of durable execution: that the workflow graph is known when the system is deployed. How do you safely run and recover a program that did not exist until an agent created it? This talk shows how Conductor provide a durable harness for dynamic workflows. Connecting existing agent frameworks to Conductor without requiring developers to rewrite their agent logic. Conductor executes the generated plan as an inspectable workflow with durability, parallelism, retries, human approvals, MCP tool calls and policy enforcement. We will demonstrate an agent creating a workflow, executing part of it, and replanning the remainder as conditions change while preserving completed work and using idempotency and saga compensation to manage side effects safely. The agent owns the plan. The harness owns the guarantees.

SESSION **Expo Stage 2 · Expo Stage 2 NW** 11:10am-11:30am

AI-Assisted Engineering: 5 Trends We're Seeing From 500+ Organizations

Justin Reock — Deputy CTO, DX

AI is reshaping how engineers work but what does that actually look like at scale? Drawing on data and patterns from more than 500 organizations, we break down the five most significant trends emerging in AI-assisted engineering today. This fast-paced theater session cuts through the hype to deliver concrete, evidence-based insights that engineering leaders can act on immediately. Key takeaways: Discover the top 5 AI-assisted engineering trends observed across 500+ organizations Understand how leading teams are integrating AI into their engineering workflows Leave with actionable strategies to apply at your organization

SESSION **Expo Stage 3 · Expo Stage 3 SW** 11:10am-11:30am

The Death of Keyword Search and the Rise of Agent-Readable Catalogs

Nixon Dinh — PayPal

As search shifts from classic keyword matching to more conversational experiences, product data quality becomes critical to LLM-powered retrieval. At PayPal, we tested how enriching traditional catalog data could help AI systems better find, understand, and rank products across large-scale commerce catalogs. We built a RAG-based AI judge to compare enrichment approaches and identify five patterns that consistently improved AI discovery results.In this talk, we'll share the evaluation framework, key lessons, and a practical approach for preparing enterprise data for conversational and agentic search.

SESSION **Expo Stage 4 · Expo Stage 4 SE** 11:10am-11:30am

FDE Playbook: Build an AI Support Agent and Give It a Voice

Matt Lawler — Forward Deployed Engineer Lead, AssemblyAI

Bio: Matt Lawler leads FDE for Onboarding at AssemblyAI, helping teams ship speech-to-text and voice AI to production, from model selection and architecture through deployment and scale. Description: Most support bots can read. Joey can talk back. In this session, AssemblyAI's Forward Deployed Engineer Lead, Matt Lawler, shares how his team built Joey, an AI support agent that increased end-to-end resolution rates from 10% to 75%. He'll walk through the architecture, key lessons learned, and how the team extended Joey into a fully voice-enabled agent.

11:40am

SESSION **Memory & Continual Learning · Main Stage** 11:40am-12:00pm

Memory Harnesses for Long-Running Research Agents

Stefania Druga — Research Scientist, Sakana.ai

At Sakana AI we build agents that run for hundreds of turns to read literature, run experiments, and draft papers. The model rarely breaks. The harness around it is the weak point: the agent contradicts a decision it made 80 turns ago, redoes finished work, or drifts from the question it started on. This is the binding-constraint thesis. For long-horizon tasks, reliability is set as much by the harness as by the model as clearly instantiated in autoresearch recent efforts. This is a field guide to the harness's memory layer. I'll trace a real research agent through its lifecycle, show exactly where context rot and drift set in, and cover the patterns that hold over 100+ turns: three-tier memory, progressive disclosure, recall-first compaction, sub-agent isolation, and architectural memory beyond the vector database. I will show how to measure whether your memory harness actually helps, at the trajectory level, so you stop tuning prompts to fix what's really a state-management bug.

SESSION **Sandbox & Platform Engineering · Track 1** 11:40am-12:00pm

Kubernetes Is Not Your Sandbox

Ivan Burazin — CEO, Daytona

Teams are reaching for Kubernetes to run agent sandboxes, and it's the wrong tool. Kubernetes is built to keep things alive and hold them in a steady state. A sandbox is born, forked, and killed before any of that machinery catches up. The mismatch compounds because the sandbox keeps gaining requirements without shedding any. In eighteen months it went from a fast code-snippet runner, to a stateful box for long-running agents, to ten thousand ephemeral environments that fork for RL rollouts and die in under a second. It has to be all of those at once, a contradiction set no orchestrator was designed to hold. The cost shows up the moment you measure it. We ran the same 50-action bug-fix trajectory across five stacks and got a 12x spread: 12.9s on the fastest, 161.5s on the slowest. The gap isn't compute, it's lifecycle overhead per action. We name every stack and explain the mechanism behind each number. wdyt?

SPONSOR **Robotics & World Models · Track 2** 11:40am-12:00pm

Commercial Grade-Robots for Real World Usage

Jason Ma — CTO and co-founder, Dyna Robotics

TBD — Dyna Robotics talk for Robotics & World Models track. https://www.dyna.co/

SESSION **Memory & Continual Learning · Track 3** 11:40am-12:00pm

Scaling Compute on Context

Jack Morris — Cofounder, Engram

A case for when context is enough, and when updating weights may be the real memory mechanism.

SESSION **Workshops Day 2 · Track 4** 11:40am-12:00pm

Build realtime multimodal agents with Gemini Live (continued 3)

Thor 雷神 Schaeff — Member of the Technical Staff (DevX) at Google DeepMind, Google DeepMind

The Gemini Live API is incredible versatile when it comes to building realtime AI experiences. From live translation across 2000 different language pairs to building realtime multimodal agents that can work across text, audio, and vision. This workshop gets you from zero to fully conversational agent in a matter of hours.

SPONSOR **Evals · Track 5** 11:40am-12:00pm

Building Closed-Loop Evals for a Multimodal Agent at Uber Scale

Soumya Gupta — ML Engineer, Uber · **Jai Chopra** — Product Manager, Uber

This talk covers how we designed evals for Uber's food enhancement agent—which edits food photography to better present dishes for smaller, independent Uber Eats merchants—along with the pitfalls and lessons learned along the way. The problem is uniquely hard: we must stay faithful to the original dish, preserve each merchant's brand and packaging, and avoid homogenizing the marketplace—all without an existing playbook for multimodal evals in a narrow domain. We'll dig into what we learned navigating reward hacking, where the agent figured out how to game the eval loop, and how we built a closed feedback loop incorporating offline and online signals for continuous improvement—all while balancing creativity against rigid safety guardrails at scale. If you're an ML or applied AI practitioner working on multimodal systems, agentic pipelines, or eval design—especially building generative features under tight safety or quality constraints—you'll walk away with practical strategies for designing multimodal evals in a narrow domain, recognizing and countering reward hacking, and building offline/online feedback loops that keep a generative agent improving in production.

SESSION **Design Engineering · Track 6** 11:40am-12:00pm

The Design-Code Roundtrip That Isn't

Jonathan Gordon — Founder, ReWeaver AI

Everyone is using Figma's MCP tools, Claude Code, or Codex. The demos are seamless. The narrative is compelling. What's actually happening under the hood is something else entirely. And the gap between the story and the reality is where your next six months of pain is going to come from. I'm Jonathan Gordon, founder of ReWeaver AI and a programmer-turned-UX designer who spent 30 years in developer tools at Google, Microsoft, Apple, Facebook, and Oracle watching the design-engineering gap widen in slow motion. I've seen every generation of tooling promise to close it. I know exactly where the seams are. I wrote a technical teardown of what Figma's bidirectional workflow actually ships, what get_design_context does, what generate_figma_design actually captures (hint: it's a screenshot, not your design system), and why iterating through that loop 12 times leaves you progressively farther from your canonical design intent. This talk will walk attendees through each step, backed by research and specific examples, and include a demo showing how drift accumulates in real time. The problem is not that drift happens; it's that it's happening exponentially. Let's talk about how we can stem that tide and keep humans in control of the process, not just "in the loop."

SESSION **Computer Use · Track 7** 11:40am-12:00pm

Bringing agents onto the world wide web

Paul Klein IV — Founder & CEO, Browserbase

The web wasn't built for agents. Heavy HTML, human-first UIs, and a DOM that can hijack the model's context. Still, agents browse it for millions of hours every month through Browserbase, across teams like Ramp, Shopify, and Lovable. This talk walks through that browser agent harness layer by layer, from the security boundary between DOM and model to caching, Agent Identity, and the infrastructure that provisions browsers at scale, and where browser agents go once it is in place.

SESSION **Context Engineering · Track 8** 11:40am-12:00pm

500 Skills, Zero Fine-Tuning: LinkedIn's Playbook for AI Agents That Actually Know Your Codebase

Ajay Prakash — Senior Staff Software Engineer, Linkedin

Everyone's building custom AI agents. We didn't. Instead, we built CAPTAIN — an MCP server that makes any off-the-shelf coding agent understand LinkedIn's entire engineering stack. The secret: a meta-tool architecture (discover → inspect → execute) and composable skills that encode tribal knowledge as executable workflows. 500+ skills later, it's used across all of LinkedIn engineering. I'll show you the architecture in 10 minutes and why context engineering beats model engineering every time.

SESSION **Data Quality · Track 9** 11:40am-12:00pm

Training Frontier Models to Out-Think Hackers

Uri Rolls — CEO, Arithmetic · Thom Wolf — Co-founder and CSO, Hugging Face

We will give a surprisingly optimistic talk about AI and cyber, and why we believe it is not the end of cybersecurity as we know it, but an opportunity to empower defenders and build a lasting edge over attackers. Cyber is a battle of skill and speed, and the rising tide of frontier models is allowing human attackers to move faster and cheaper. That combination of skilled hackers and breakthrough LLMs is a real threat, while defensive systems are still expected to operate at scale with limited human intervention, constrained by what models can do out of the box. But the answer is not fear or despair. Just as high-quality data transformed software engineering, the right cyber training data can teach models to turn from weapons being used against us into tools that protect us.

SPONSOR **Track M** 11:40am-12:00pm

OpenAI, Anthropic, or agent frameworks: choose the right AI stack

Arun Sekhar — Principal Product Manager for AI Developer Experience, Microsoft · Pamela Fox — Principal Cloud Advocate, Microsoft

OpenAI SDK, Anthropic SDK, or an LLM-agnostic agent framework. Which one should your next AI app be built on? Starting with Foundry Models, we walk through each option in code, show what you gain and what you give up at every layer, and help you pick the right abstraction for your scenario without overbuilding.

SESSION **AI-Native Enterprises · Leadership 1** 11:40am-12:00pm

Your Code Has Bugs. Lean4 Has Proofs. A Practical Guide to Formal Verification for Engineers

Varun Pant — Builder, NeuroSymbolic AI, AWS

AI is generating more of your code than ever — how do you prove it doesn't ship bugs? Lean is a theorem prover that's also a programming language, and it's quietly becoming practical for verifying real software. In this talk, I'll show you how formal verification works — some examples of proof tactics, and a practical framework for when to verify vs. test

SESSION **AI Architects: Tokenmaxxing · Leadership 2** 11:40am-12:00pm

How to Kill the Code Review

Ankit Jain — Founder & CEO, Aviator

Human-written code died in 2025. Code review is dying in 2026. Teams with high AI adoption are merging 98% more pull requests, but PR review time has surged 91%. There is no way we win this fight with manual code reviews, and AI code review tools are just buying us time. This talk makes the case that the traditional code review is a historical approval gate that no longer fits the shape of modern software development. I'll walk through a practical five-layer trust model: from multi-agent competition and deterministic guardrails to spec-driven BDD and adversarial verification — that lets engineering teams ship faster without sacrificing quality or control.

SESSION **Expo Stage 1 NE** 11:40am-12:00pm

Fault-Tolerant Training at Scale: Making Hardware Failures a Non-Event

Hardware failures in large-scale distributed training are inevitable — when you're running thousands of GPUs, they happen multiple times a day. The standard response is manual intervention: an engineer gets paged, SSHs into the cluster, and spends an hour fixing something the infrastructure should have handled automatically. That lost time compounds directly into wasted compute and delayed research. This session walks through the self-healing platform Crusoe built to eliminate that manual loop entirely — a managed Slurm environment running on Kubernetes, with automated node failure remediation and real-time cluster observability — and how these components work together so hardware failures become a non-event. We'll cover this architecture end-to-end: how running Slurm on Kubernetes unlocks infrastructure resilience that traditional GPU clusters don't have, how automated hardware monitoring and node remediation can eliminate manual intervention entirely, and how full observability into every remediation event keeps engineering teams informed without keeping them on-call. For teams that want deeper control, we'll also discuss open-loop remediation, which gives teams full control over the node replacement process for application-specific workflows.

SESSION **Expo Stage 2 NW** 11:40am-12:00pm

How to generate mergeable code with a context engine

Peter Werry — Founding Engineer, Unblocked

Your agents are fast, capable, and completely context-blind. They generate code that compiles but doesn't reflect how your system actually works. You're likely already seeing the impact: ballooning token costs, longer review cycles, and inconsistent outputs. More MCPs, rules, and bigger context windows give agents access to information, but not understanding. In this session, we dissect how teams pulling ahead use a context engine to give agents exactly what they need for the task at hand. Includes a short demo showing the workflows a context engine can augment.

SESSION **Expo Stage 4 SE** 11:40am-12:00pm

AI agents don't read your policy docs. They hit your APIs.

Every organisation has a policy for what AI should and shouldn't do. But in the era of autonomous agents, who is that document actually for? Odds are no agent has ever read it. It opens a connection and makes a call, and whatever happens at that millisecond is your real policy. So put the control there. This talk is about the gateway as the runtime where AI governance actually executes: per-agent identity and scoped, short-lived credentials instead of a shared god-key. PII and secrets stripped from prompts in flight. Token-aware rate limits so one looping agent can't torch your quota. Semantic caching that cuts spend and latency on requests you've already answered. I'll share the architectural patterns behind each control, what they look like in practice, and what breaks the moment you take them away. Policy states intent. Infrastructure enforces it.

12:05pm

SESSION **Autoresearch · Main Stage** 12:05pm-12:25pm

« the era of (auto) research »

Elie Bakouch — Research Engineer, Prime Intellect

the nanogpt speedrun is a great setup to test autonomous research: fixed model, one number to beat, and a human record that keeps moving. we pointed coding agents at it on idle compute and let them iterate for days, thousands of runs with minimal human intervention, until they beat the human baseline. in this talk we go through how they did it, how codex and claude code behave very differently as researchers, and why speedrun are one of the best environments we've found for studying autonomous research agents

SESSION **Sandbox & Platform Engineering · Track 1** 12:05pm-12:25pm

Your agent needs a sandbox, not a desert

Samuel Colvin — Founder & CEO, Pydantic

Everyone agrees agents need code execution. That agreement lasts right up until you ask how to do it. The default answer is usually something like "My agent needs a full Linux VM to succeed". That's a very convenient answer for sandbox providers, but I think it's often incorrect. In many real-world agent workflows, the model does not need a whole computer. It does not need arbitrary packages, shell access, CPython, node, let alone `awk` `sed` and `gcc`. It needs a small amount of safe, expressive compute: enough to write code, call tools, and keep intermediate state out of the context window. That is the idea behind Monty: a minimal Python interpreter, written in Rust, designed specifically for running code written by agents. In this talk, I'll argue that for a surprisingly large class of agent systems, a curated set of tools in a custom runtime is better than a full sandbox. Not because full sandboxes are bad, but because they solve a much larger problem than most embedded agents actually have. And you pay for that mismatch in complexity, cost, operational pain, and 100,000X higher latency. Sandboxes are great, but there's such a thing as too much sand - in many scenarios the constraints and limitations of a custom built, minimal sandbox are a feature, not a bug.

SESSION **Memory & Continual Learning · Track 3** 12:05pm-12:25pm

Intelligence + Continual Learning = Expertise

Yu Su — Co-founder and CEO, NeoCognition

Talk on continual learning for LLMs and agents, drawing on retrieval-to-memory and environment-adaptation research.

SESSION **Workshops Day 3 · Track 4** 12:05pm-12:25pm

Build realtime multimodal agents with Gemini Live (continued 4)

Thor 雷神 Schaeff — Member of the Technical Staff (DevX) at Google DeepMind, Google DeepMind

SPONSOR **Evals · Track 5** 12:05pm-12:25pm

From Agent Traces to Agent Simulations: The next era of agent evaluation

Rustem Feyzkhanov — Senior Engineering Manager - AI Platform, Snorkel AI

Agent evaluation is moving beyond reviewing static traces after the fact. This talk explores how executable simulation environments let teams repeatedly test agents across realistic tasks, compare models and harnesses, and uncover failure modes that trace review alone misses. Drawing from Snorkel's experience building simulation datasets at scale for major labs and contributions to projects like Agents' Last Exam and Terminal-Bench, we'll cover concrete engineering patterns for building these environments: defining clear specs and requirements, implementing evaluators for simulation environments and tasks themselves, keeping environments decoupled from any single agent or model, and designing verifiers that evaluate both final outputs and agent traces. Attendees will leave with a practical mental model for creating environments that are lightweight enough to run at scale, but realistic enough to mock production systems such as databases, APIs, and tools in ways that meaningfully challenge agents.

SESSION **Design Engineering · Track 6** 12:05pm-12:25pm

Mousepower: agents that can’t be measured, can’t be managed.

Maximillian Piras — Founding Designer, Yutori

Agents have a measurement problem, which makes them impossible to efficiently manage. You’ve likely heard many say execution is now cheap, but judgement is the new bottleneck. This is because our evaluation frameworks weren’t designed for systems that tirelessly output in parallel. The canary in the coal mine is code generation becoming largely solved at the expense of breaking code review. As agents reverberate across all knowledge work, the same fracture will spread to artifacts, actions, & decisions. Yet without a scalable quality measure, we can’t ascend to a higher level of abstraction because we won’t trust the foundation below. So how do we design measurements that are efficient, intuitive, & trustworthy? Past paradigm shifts offer inspiration, such as James Watt not just building a better engine but also inventing horsepower to map it onto existing mental models. We need an equivalent quantification to communicate the “mousepower” of agents. Information theory gives us the starting point: concepts like entropy, ergodic processes, and Hamiltonian problems point us toward the most tractable trajectories — easier to verify than they are to solve.

SESSION **Computer Use · Track 7** 12:05pm-12:25pm

The Dark Arts of Web Automation: Teaching Agents to Use Websites Like Humans

Corey Gallon — Managing Director, Rexmore

Anything you can do in a browser, your agent can do too. Not by tiptoeing through an MCP server one polite, token-burning call at a time -- properly, programmatically, the way you'd drive any other tool. I'll show you how with chrome-agent, an open source wrapper over the Chrome DevTools Protocol that has become irreplaceable in my everyday work. If you'll ever do a browser task more than once, step-by-step MCP browsing is slow, brittle, and bills you tokens for every single click. A CLI straight onto CDP makes the whole browser programmable: loop it, pipe it, script it, walk away. Write it Tuesday, run it a thousand times Wednesday, all without a second of AI agent babysitting. We'll dispel the MCP hype and myths, with successful demonstrations of cheeky things like: the power of CLI-based browsing and how its so much more capable than mere MCP; reaching through those oh-so-clever cross-origin iframes to clear the verify you're human checkboxes; showing that a JavaScript .click() is not a click, rather, just a function call in a costume that is banhammerable; ultimately, proving that a CDP browser operates just like a meatbag with a mouse and keyboard. You'll learn how to point your AI agents at real, messy, uncooperative websites and web applications and have them get things done exactly the way that you would.

SESSION **Context Engineering · Track 8** 12:05pm-12:25pm

Your agents lack context: Here's how to fix "You're absolutely right!"

Brandon Waselnuk — Developer Relations, Unblocked

Every AI coding tool can generate code. Very few can generate the right code for your organization, because they're missing context. They don't know why your team chose Redis over DynamoDB, what the team decided in a Slack thread earlier today about the auth migration, or which architectural patterns your principal engineers actually enforce in review. This talk is a practitioner's guide to building a context engine: the reasoning layer that continuously ingests & synthesizes organizational knowledge across disparate sources into unified, queryable understanding. I'll walk through the problems you actually have to solve — reasoning across systems that don't agree with each other, searching globally before you can reason, maintaining identity-scoped permissions so every user and agent only sees what they should, and personalizing results based on who's asking and what they're working on. These are the engineering challenges that make naive RAG fall short, drawn from real lessons building this at scale.

SESSION **Posttraining & Midtraining · Track 9** 12:05pm-12:25pm

Learning on the job: the future of post-training

Raymond Feng — Researcher, Applied Compute

SESSION **AI-Native Enterprises · Leadership 1** 12:05pm-12:25pm

AI-Native Organisations runs on Skills: How to Extract, Structure, evaluate and Scale Them

Imad Touil — Distinguished Engineer, QuantumBlack, AI by McKinsey

SESSION **AI Architects: Tokenmaxxing · Leadership 2** 12:05pm-12:25pm

The Death of the Code Review

Laurie Voss — Head of Developer Relations, Arize AI

Code review was built for a world where humans wrote all the code. Now, the question isn’t “does this diff look good?” — it’s “can this system safely ship code on its own?” This talk will show why and how traditional code review will quietly be replaced by automated verification harnesses. We’ll show how prompt learning can be used to clone your best internal code reviewers, turning their judgment into automated evaluation loops. We’ll also open source a code review training harness that captures review patterns and turns them into reusable checks for AI-generated code.

SESSION **Expo Stage 1 NE** 12:05pm-12:25pm

Your agent architecture has a half-life of 6 months

Dan Farrelly — CTO and Co-founder, Inngest

A short history of the right way to build an agent: RAG, ReAct, prompt chaining, orchestrator-workers, MCP, CLI, MCP again... CLI again?? Every time you adopt a trend you rebuild your architecture. In this talk, Dan Farrelly, Inngest cofounder and CTO, is not going to tell you what comes next. He's going to show you how to build so it doesn't matter. He'll cover the core primitives that show up in every production agent, how bringing decisions closer to code provides more stack flexibility, and why the right execution layer unlocks faster iteration.

SESSION **Expo Stage 2 NW** 12:05pm-12:25pm

From Stateless to Stateful: Orchestrating Real-Time Voice & Messaging Agents with Twilio and Amazon Bedrock

Rishab Kumar — Staff Developer Evangelist, Twilio

We have all had that maddening customer service experience: you text a support line about a delayed flight, receive a confirmation, but when you call in a minute later, the voice agent asks, "How can I help you today?" completely blind to the SMS you just sent. This is the "Channel Amnesia" problem. While businesses are pouring billions into generative AI, most agents are still built on stateless architectures that forget customer context the second a session ends. In this session, we will cure AI amnesia. You will learn how to orchestrate stateful, production-grade AI agents across SMS and Voice using Twilio Agent Connect and Amazon Bedrock. We will dive into why traditional serverless compute fails stateful agents, how to leverage AWS Fargate for isolated, long-lived sessions, and how to configure Bedrock AgentCore over WebSockets to hit sub-50ms streaming voice latency. No slide-ware here expect a live, cross-channel demo and open-source code you can deploy tomorrow.

SESSION **Expo Stage 3 SW** 12:05pm-12:25pm

Harnessing Collective Agent Intelligence for Open Science

James Zou — Associate Professor of Biomedical Data Science, Stanford University / Together AI

What happens when AI agents don't just work in isolation, but collaborate, compete, and build on each other's breakthroughs in real time? James Zou, Head of Frontier Agents at Together AI, explores how collective agent intelligence is pushing the boundaries of open science. https://www.together.ai/blog/einsteinarena is a live platform where AI agents collaborate on unsolved mathematical problems, sharing results and building on each other's work. In April 2026, agents improved the best known lower bound for the Kissing Number in 11 dimensions from 593 to 604, surpassing AlphaEvolve through 48 hours of live multi-agent collaboration. https://www.together.ai/blog/dsgym is a unified framework for evaluating and training data science agents, exposing a critical gap in existing benchmarks: models often rely on memorization rather than true data analysis. The team used it to train a 4B open-source model that rivals much larger frontier models. These projects demonstrate agents learning from rigorous evaluation, collaborating through shared infrastructure, and driving scientific discovery at a pace no single researcher or model could achieve alone.

SESSION **Context Engineering · Expo Stage 4 SE** 12:05pm-12:25pm

Prompt, Memory, Weights: The Architecture Decisions Most AI Teams Make by Accident

Anant Srivastava — Principal Technologist - Data and AI Platforms, Oracle

The interesting engineering in production AI isn't in the model. Your knowledge lives in files, databases, and APIs: docs, runbooks, conversations, code. The model just reads tokens. So the real architectural question is which path that knowledge takes to inference: into the prompt directly, into memory for retrieval on demand, or into the weights through fine-tuning. Most teams treat these as a ladder. Start with prompts, escalate to RAG, eventually fine-tune, as if each step is a more advanced version of the last. The field is converging on a different answer: they solve different problems. The prompt shapes behavior and constraints. Memory grounds the model in current, citable knowledge. Weights harden specialized reasoning and format. They're not substitutes you graduate between; they're complementary, and the failures come from using one to do another's job. Fine-tuning to teach the model facts it should have retrieved is the classic trap: you bake in knowledge that's stale the day it ships, and you still can't cite it. This is an opinionated take on all three: when each is the right call, when each is a trap, and the part most teams never build, the circulation between them. Memory that captures what the agent does becomes the dataset you fine-tune on; fine-tuning changes what's worth retrieving; the loop compounds. Get the three paths right and they stop being a pipeline you climb and start being an architecture that learns.

12:30pm

12:30pm-1:30pm **SOTA Generative Media Panel**

12:30pm-1:30pm **This Year in Claude**

1:30pm

SESSION **Autoresearch · Main Stage** 1:30pm-1:50pm

Closing the Loop: An Autonomous AI Research Agent

Tim Sweeney — Principal Engineer, Weights & Biases by CoreWeave

The holy grail of agentic AI tooling is the autoresearch loop: an agent that can sift through your experiments, create visualizations, propose a hypothesis, launch a training job, read the results, and try again autonomously. In this session, we'll show new autoresearch capabilities built directly into the W&B Models web and iOS apps. We will demo these live using a real-world fine-tuning project, covering everything from launching jobs and reading loss curves to surfacing outlier runs that consume researcher hours and recommending the next steps. Then you'll learn how the eval-driven development loop in W&B Weave makes agents like this trustworthy. You'll see how production traces become benchmarks, and how only the agents that beat the bar make it to production. Join us to learn the same loop we use to improve our own agentic features.

SESSION **Sandbox & Platform Engineering · Track 1** 1:30pm-1:50pm

From fork() to Fleet: Designing an Agent Sandbox Cloud Pt 1

Abhishek Bhardwaj — Member of Technical Staff, RL & Agent Infrastructure, OpenAI

Sandboxes unleash agents by giving them secure, fully functional computers where they can tackle diverse tasks with minimal setup. This talk explores the architectural challenges of building an agent sandbox cloud. We compare runtime isolation technologies and their trade-offs, examine persistence and storage as the next major unlock for agent capabilities, and discuss the key decisions involved in orchestrating and scaling sandboxes.

SPONSOR **Robotics & World Models · Track 2** 1:30pm-1:50pm

Unitree: Building Mass Produced Humanoids

XiangMing Sun — Unitree

SESSION **Memory & Continual Learning · Track 3** 1:30pm-1:50pm

Adaption Labs — Gradient-Free Continual Learning

Sara Hooker — CEO, and Co-founder, Adaption

Gradient-free continual learning for AI systems that adapt from real-world experience.

SESSION **Workshops Day 3 · Track 4** 1:30pm-1:50pm

The Agentic Power User's Playbook: Tips and Tricks for Swarm-Style Agentic Development

John Lindquist — Agentic Instructor, egghead.io

You opened a fifth agent tab this morning and immediately lost track of which one was doing what. This workshop is the playbook I use daily to run swarms of agents in parallel: the keyboard shortcuts, layout patterns, supervision habits, and fast-model tricks that turn chaos into a control surface. We'll go hands-on: spawning a wall of agents across tiled panes, routing prompts to the right swarm with fast models, switching contexts in milliseconds, recovering when an agent goes off the rails, and building the muscle memory that separates a one-agent-at-a-time user from a true power user. By the end you'll leave with a stocked toolbelt of concrete shortcuts, repeatable patterns, and workspace habits you can drop into your own setup the same day. No cloud, no platform lock-in: every trick runs on the machine in front of you.

SPONSOR Evals · Track 5 1:30pm-1:50pm

Model Whisperers: How Evals and Prompts Shape Agent Behavior

Chris Souza — Google · **Preetika Bhateja** — Product Manager, Google · **Daniel Bump** — Engineer, Google

Getting an AI agent to behave the way you want isn't just about writing better prompts. In real systems, behavior emerges from a loop: prompts->evals->iteration->feedback. Small changes in any part of that loop can completely change outcomes. We saw this while building a seed asset agent - a system that turns messy, real-world advertising creatives (low quality images, cluttered visuals, heavy text overlays) into clean, reusable assets for downstream Gen AI tools. The agent acts like an editor, simplifying visuals, removing unnecessary elements, and isolating core content so that additional context (like text or CTAs) can be added back in a more controlled, brand-safe way. But the real challenge wasn't just building the agent - it was making it reliable. And prompting alone wasn't enough. What actually moved the system forward was how we defined success—and how we used evals to reinforce it. Over time, evals stopped being just a way to measure quality. They became part of how the agent learned what "good" looks like. In this talk, we'll cover: Why prompting alone doesn't give you stable agent behavior How evals act like feedback signals, not just scorecards How we built evals sets that reflect the real-world Using agent trace logs to understand why things fail (not just that they fail) How to iterate without breaking things you already fixed By the end, you'll have a set of patterns you can apply to any system dealing with messy/continuously changing data and how to tweak your prompt and evals to accommodate such changes.

SESSION Design Engineering · Track 6 1:30pm-1:50pm

Design at the Speed of Adjectives

Paul Bakaus — Founder, Renaissance Geek, Inc.

Every design tool today operates at the wrong level of abstraction for AI-assisted engineering. Traditional tools give you padding sliders and color pickers, built for a world where designer and engineer are separate roles moving at separate speeds. Prompt-to-design tools one-shot a pretty landing page from a sentence, which is more dangerous because it looks like it's working. No serious design director hears a prompt and starts pushing pixels. The brief comes first. What's the emotional territory? What should this not feel like? Today's AI tools skip that discovery entirely. The result is output without intent. Technically competent, strategically empty. The right abstraction for a world where the designer is also the engineer lives between these extremes. Not pixels. Not prompts. Adjectives. "Make it feel warmer." "Strip it to its essence." "Add tension." These are the controls a creative director actually thinks in. Drawing on lessons from building Impeccable, an open source design tool with 24 adjective-level commands like /bolder, /quieter, and /distill, I'll share what worked, what didn't, and how to apply this thinking to any AI interface where creative intent matters more than parameter control.

SESSION Computer Use · Track 7 1:30pm-1:50pm

From RL to IRL

Gaurav Mishra — Amazon AGI Lab

Today's agents have to operate in a messy reality of flaky connections, ephemeral credentials, and irreversible actions. They need to navigate real software the way humans do: recovering from failures, learning from feedback, and making sound judgment calls. This talk is about the fundamental changes in RL required to make agents ready for IRL. We'll walk through what it takes for training environments to reflect the complexity of the real world, the perception primitives that let an agent see what a user sees, the harness pieces that help it survive contact with real applications, and the failure modes you only discover when you stop scoring and start shipping.

SESSION Context Engineering · Track 8 1:30pm-1:50pm

How long can your skills be before your agent forgets what you told it?

Laurie Voss — Head of Developer Relations, Arize AI

A year ago, frontier models lost the thread somewhere around 200 simultaneous instructions, so skills files had to stay short and lean on sub-skills and subagents. We re-ran IFScale on the 2026 frontier and found the ceiling has moved by an order of magnitude: closer to 2,000 instructions, up to 5,000 on the strongest models. The more interesting story is how models fail at the new frontier: DeepSeek quietly drops instructions, Opus refuses outright when innocuous words trip a safety classifier, Gemini burns its whole budget on reasoning and emits nothing, and GPT-5.5 stops to tell you your request was unreasonable. The capacity problem is largely solved; verification is wide open. We'll show the data, the failure modes, and what it costs to find out. You'll come out with hard data on the ceiling for complex instructions to LLMs

SESSION **Posttraining & Midtraining · Track 9** 1:30pm-1:50pm

Reinforcement Learning without Verifiable Rewards

Will Brown — Researcher, Prime Intellect

Verifiable rewards are the gold standard for RL training, but real-world agent tasks frequently lack clean deterministic evaluation objectives. This talk surveys our efforts to scale RL in non-verifiable settings -- including task synthesis, unsupervised environment design, and automatic judge calibration -- to ultimately enable self-improvement in production, grounded in real-world agent traces and domain-specific context.

SESSION **AI-Native Enterprises · Leadership 1** 1:30pm-1:50pm

The Half Life of Agent Infrastructure

Ben Kus — CTO, Box

TBD — talk on search and retrieval, agentic AI, and enterprise AI over unstructured content.

SESSION **AI Architects: Tokenmaxxing · Leadership 2** 1:30pm-1:50pm

Tokenmaxxing is the New "Lines of Code"

Nicholas Arcolano — Head of AI & Research, Jellyfish

Somebody in your company is going to ask what you're getting for all that AI spend. If you don't have a good answer, someone else will make one up... and it might be "total tokens consumed". That's how tokenmaxxing becomes policy: not because anyone thinks it's a good metric, but because engineering didn't offer a better story. I work with datasets spanning hundreds of companies, hundreds of thousands of engineers, and billions of lines of shipped code to understand how AI engineering is evolving and what actually matters to measure. One thing I've learned is that raw token spend is a VERY crude estimator of value. For example, across levels of token spend, cost per merged pull request varies 300x — but output only varies 2x. The good news is the data also shows what DOES matter, and it's measurable and actionable – but most teams aren't tracking it yet. This talk will give you the data, metrics, and frameworks you need to keep your org from adopting the latest terrible vanity metric. You'll learn what actually separates teams that scale AI effectively from those just burning tokens, and how to tell the story that keeps your AI investment funded and growing.

SESSION **Expo Stage 2 · Expo Stage 2 NW** 1:30pm-1:50pm

Surviving Your Own Velocity: How VS Code Ships Weekly with 40 People

Harald Kirschner — Principal Product Manager, Microsoft

A ~40-person team ships VS Code weekly to millions of users. Models got good enough to lean on, and leaning in is exactly what broke our process. This talk is the part most AI talks skip: what you have to rebuild after agents start working. We had to scale three things at once: how fast we ship, how we hold quality, and how fast we learn, and each one we fixed revealed the next. I'll walk through the harnesses, evals, and self-healing systems that keep velocity from becoming regression, and the patterns you can steal.

SESSION **Expo Stage 1 · Expo Stage 3 SW** 1:30pm-1:50pm

Why Agents Should Have Their Own Sandbox

Philipp Schmid — Staff Engineer, Google DeepMind

1:55pm

SESSION **Autoresearch · Main Stage** 1:55pm-2:15pm

An AI Agent Became the #1 Contributor in OpenAI's Hiring Challenge

Zhengyao Jiang — CEO & Cofounder, Weco AI

Earlier this year, OpenAI ran Parameter Golf, a model-training competition that doubled as a hiring filter. Over 1,000 researchers competed to train the best small language model under a 16MB cap. The top contributor was the one candidate OpenAI couldn't hire. Our autonomous research agent Aiden finished with 7 merged records, more than twice as many as any other contributor, and ended up the most-cited participant in the community. This talk is about what those 22 days showed. I'll cover on high level how does it works and which of its ideas produced the records. But the part worth more than the leaderboard is the collaboration itself, the community and AI agent building on each other's work, the largest natural experiment in human-AI collaboration I've seen run in public. I'll close with what it tells us about where humans and autonomous research each still matter for the foreseeable future. 1:57 PM

SESSION Sandbox & Platform Engineering · Track 1 1:55pm-2:15pm

From fork() to Fleet: Designing an Agent Sandbox Cloud Pt2

Abhishek Bhardwaj — Member of Technical Staff, RL & Agent Infrastructure, OpenAI

Sandboxes unleash agents by giving them secure, fully functional computers where they can tackle diverse tasks with minimal setup. This talk explores the architectural challenges of building an agent sandbox cloud. We compare runtime isolation technologies and their trade-offs, examine persistence and storage as the next major unlock for agent capabilities, and discuss the key decisions involved in orchestrating and scaling sandboxes.

SPONSOR Robotics & World Models · Track 2 1:55pm-2:15pm

Frontier Robotics Research

Deepak Pathak — Co-Founder & CEO, Skild AI

SESSION Memory & Continual Learning · Track 3 1:55pm-2:15pm

Improving Agents is a Data Mining Problem

Vivek Trivedy — Head of Applied Research, LangChain

Harness Engineering, Post-Training, Continual Learning...these all boil down to the same underlying substrate - Mining Agent Traces 1. I need to run my agents to collect Traces 2. Understand behaviors from Traces at scale 3. Filter data for "improvement" 4. Do an improvement step There's a reason why every continual learning platform ends up looking like an observability platform. It's because Traces are the lifeblood of agent improvement. The mechanism that we use to attempt improvement can vary - Harness Eng, SFT, etc. But without understanding the data agents produce, no algorithm will truly build better agents. The holy grail of Agent Improvement is Continual Learning. Consistently mining data and integrating it into the agent definition over infinitely long time horizons. Today, the easiest way to do that is to build an observability platform and constantly point agentic compute to understand the data that agents produce. We'll walk through the current methods of understanding traces at massive scale and choosing how to integrate them to improve agents across your personal agents, team agents, and entire company.

SESSION Workshops Day 3 · Track 4 1:55pm-2:15pm

The Agentic Power User's Playbook: Tips and Tricks for Swarm-Style Agentic Development (continued 2)

John Lindquist — Agentic Instructor, egghead.io

You opened a fifth agent tab this morning and immediately lost track of which one was doing what. This workshop is the playbook I use daily to run swarms of agents in parallel: the keyboard shortcuts, layout patterns, supervision habits, and fast-model tricks that turn chaos into a control surface. We'll go hands-on: spawning a wall of agents across tiled panes, routing prompts to the right swarm with fast models, switching contexts in milliseconds, recovering when an agent goes off the rails, and building the muscle memory that separates a one-agent-at-a-time user from a true power user. By the end you'll leave with a stocked toolbelt of concrete shortcuts, repeatable patterns, and workspace habits you can drop into your own setup the same day. No cloud, no platform lock-in: every trick runs on the machine in front of you.

SPONSOR Evals · Track 5 1:55pm-2:15pm

Evaling Video Slop

Maor Bril — Chaos Catalyst, Character.ai

Everyone is shipping video models. Almost no one is evaling them honestly. CLIP score doesn't catch temporal incoherence. Vibes-based human review doesn't scale. And every "AI judge" you wire up will quietly drift away from human preference unless you measure the drift. This is a tactical talk on building real multimodal eval, using JudgeJudy (open-sourced at Character.ai) as the working example. You'll leave with: Why video is different from text. Temporal consistency, shot continuity, narrative coherence, and the metrics that actually capture each (clip_temporal, temporal_consistency, and friends). AI judges, the real version. Custom rubrics, when they work, when they hallucinate, when they collapse to a single dimension and pretend they didn't. The calibration loop. Pearson/Spearman correlation against human scores, automated rubric improvement, detecting systematic judge bias before it costs you a release. Pairwise preference models for video. Training a Qwen3-VL backbone with Bradley-Terry loss to score "is this slop?" before it ships. Regression gates in CI. How every AgentX release at Character.ai passes through an eval wall before it reaches users. Closing the loop with JudgeJudy. Correlating eval scores against real telemetry (Amplitude, Statsig) and feeding validated gates back into the runtime. If you're shipping any multimodal output and your eval strategy is still "the team watches some clips on Friday," this is the upgrade. github.com/character-ai/judgejudy

SESSION Design Engineering · Track 6 1:55pm-2:15pm

Training Taste

Thais Castello Branco — Founder & CEO, Taste Labs

SESSION **Computer Use · Track 7** 1:55pm-2:15pm

The Rise of CaaS: Context-as-a-Service for Agentic AI

Omer Primor — Bright Data

Agentic workflows have commoditized. The new bottleneck is context. As models improve, AI agents are increasingly limited not by reasoning ability, but by the quality, freshness, and specificity of the information they can access. This session introduces Context as a Service, or CaaS, an emerging category for builders creating web-native context layers for AI agents. These tools collect, structure, enrich, index, and analyze live web data, making it available as agent-ready knowledge for specific use cases and vertical downstream applications. We ll explore how builders are turning hard-to-access web domains into agent-ready context layers: fragmented public data, dynamic sources, multimodal content, and fast-changing signals that generic models cannot reliably process within their token limits. Attendees will learn how to think about CaaS as both a technical architecture and a market opportunity: what to build, where context creates defensibility, and how raw web data can become the foundation for reliable agentic products.

SESSION **Context Engineering · Track 8** 1:55pm-2:15pm

WTF Is the Context Layer? The Missing Infrastructure for Production Agents

Prukalpa Sankar — Founder & Co-CEO, Atlan

In the last two years, models have gotten exponentially smarter. Two years ago they couldn't pass the bar. Today, top 1% of test scorers. And yet most agents still can't answer a simple business question correctly. You ship a demo that works. You deploy it. The business abandons it in a month. The missing variable is context: the business definitions, procedural knowledge, and operational norms that make a human expert valuable. Drawing on hundreds of production deployments, Prukalpa Sankar will break down what it actually takes to give agents contextual intelligence — and get them past the demo stage. She'll walk through the architecture of a context layer: how context repos work (versioned, testable, portable), how simulation environments catch failures before deployment, how agent traces compound back into shared context, and why context engineering scales where fine-tuning and prompting don't. She'll also cover why your context needs to be open (MCP, Iceberg, deploy to any framework) — and what happens when it isn't.

SESSION **Posttraining & Midtraining · Track 9** 1:55pm-2:15pm

Emulated: The data for fully autonomous software engineers and companies

Joseph Wang — CEO, Emulated

Hold for Emulated.so. Company builds reinforcement-learning environments that simulate real production systems for coding and infrastructure agents.

SESSION **AI-Native Enterprises · Leadership 1** 1:55pm-2:15pm

Guardians of the State: How We Built an Air-Gapped AI Fortress for Consumer Data

Rachna Srivastava — Enterprise Architect

This presentation shares Rachna Srivastava's personal views on building secure, local AI systems for sensitive consumer data. It explores air-gapped infrastructure for open-weight models, a dual-pass PII sanitization pipeline combining deterministic rules with a local language model, and validation techniques for detecting data leaks in model context and logs. Key takeaways include infrastructure and latency trade-offs, sanitization of unstructured financial text, and safeguards against context poisoning. The views expressed are personal and do not represent those of any employer or government agency.

SESSION **AI Architects: Tokenmaxxing · Leadership 2** 1:55pm-2:15pm

Engineering Agency out of the Happy Path

Matthew Jewkes — Cofounder & CTO, Standard Cybernetics

I spent '24 and '25 structuring the entire written history of biopharma - through drugs, trials, deals, etc. This was a ~500B token effort that translated into a production system now used by 19 of the 20 largest pharmas. We achieved PhD-level performance at scale with 99.95% accuracy over critical concepts. The hard parts were solving questions of domain and organizational "shape". This involved identifying which critical concepts and which bundle of tasks were worth the organizational investment to automate. And the biggest spillover win wasn't actually about time savings, it was about refocusing scarce expert judgment on error exhaust - out of which falls potential high value roadmap. I'll walk through real examples and non-obvious, transferable wins. While the case example is in biopharma, the pattern applies to any business that relies on expert domain judgement to deliver differentiated value.

SESSION Expo Stage 2 · Expo Stage 1 NE 1:55pm-2:15pm

Edge-Native AI: Building Ultra-Fast Agents and MCP Servers with Spin

Thorsten Hans — Senior Developer Advocate, Akamai

Centralized AI is slow; Edge-native AI is the revolution. Thorsten Hans demonstrates how to build intelligent agents and Model Context Protocol (MCP) servers that run at the speed of light. Using Spin and WebAssembly, we'll bypass the "cloud tax" of high latency and cold starts. Discover how to ship AI-driven features that live closer to your users, ensuring sub-millisecond responsiveness and enhanced privacy. Stop waiting for the origin it's time to bring the brain to the edge and master the stack that powers the next generation of intelligent, distributed applications.

SESSION Expo Stage 3 · Expo Stage 2 NW 1:55pm-2:15pm

Why your company needs a context graph, and how to build it

Gil Feig — CTO and Co-Founder, Merge

Everyone building AI products eventually draws the same diagram: boxes representing data sources, arrows pointing at the model, and a label that says "context." What that diagram doesn't show is the system that has to run underneath it deciding, for each request: which sources to consult, whether to fetch live or use cached data, if the user is actually allowed to view that data, how to stitch it all together before the latency budget runs out. And it hides the counterintuitive part: fetching more context usually makes your answers worse, not better. At Merge, we reframed context graphs as control planes, helping companies scale context graphs to hundreds of thousands of users with sub-300 ms latency. This talk walks engineers through the system design at scale: how to tier data freshness, why provenance isn't optional once third-party systems are in the loop, and how to decide when fetching less context is the right call. Attendees will leave with a mental model for context system design that separates the orchestration decisions from the retrieval layer.

SESSION Expo Stage 3 SW 1:55pm-2:15pm

Warp: Building Self-Improving Agent Software Factories

Suraj Gupta — Software Engineer, Warp

We are in the era of Software Factories, where the entire SDLC is being automated by agents. We will cover how we are approaching self-improving software factories leveraging dedicated agents to update skills, persistent cross-harness memory, and implementing feedback loops to ensure that software factories continually improve.

SESSION Expo Stage 4 SE 1:55pm-2:15pm

Natively Multimodal from Step Zero

Most AI models start as text systems and have vision, audio, and other modalities added later. That ordering shows up in the work: handoffs between modalities, brittle understanding of mixed inputs, and gaps that surface exactly when real tasks demand reading a chart, a document, and code together. This session looks at a different approach — models trained as multimodal from step zero, where text, image, audio, and video share the same foundation rather than being stitched together. We'll look at why that matters for the kind of work organizations actually want from AI: understanding messy, mixed real-world inputs, holding context across them, and carrying complex tasks end to end. The throughline is what this unlocks for teams deciding where AI can take real work today — and how MiniMax is building toward that frontier.

2:25pm

SESSION **Autoresearch · Main Stage** 2:25pm-2:45pm

Self-Improvement of Context, Harness, and Model Weights through Reflective Optimization

Lakshya Agrawal — Creator and maintainer of GEPA, GEPA

Large language models are increasingly adapted to downstream tasks via reinforcement learning methods like GRPO, which often require thousands of rollouts to learn new tasks. We argue that language provides a much richer learning medium: an LLM can reflect on full trajectories (including reasoning, tool calls and errors) to diagnose failures and propose targeted improvements. We introduce [GEPA](https://gepa-ai.github.io/gepa/), a reflective prompt optimizer that incorporates this principle outperforming GRPO by up to 20% while using up to 35x fewer rollouts across tasks spanning 5+ domains and also works with black-box models. Building on this, we then introduce [optimize_anything](https://gepa-ai.github.io/gepa/blog/2026/02/18/introducing-optimize-anything/), a unified API that generalizes reflective optimization to arbitrary text parameters. This single system achieves state-of-the-art results across eight fundamentally different areas, including nearly tripling ARC-AGI accuracy via agent architecture discovery, generating CUDA kernels that beat PyTorch and cutting cloud scheduling costs by 40% through policy discovery, establishing LLM-based reflective search as a general-purpose problem-solving paradigm. Finally, I present [Fast-Slow Training](https://arxiv.org/abs/2605.12484) (FST), which brings reflective optimization into LLM post-training. FST jointly optimizes model parameters ("slow weights") via RL and textual contexts ("fast weights") via GEPA. Because the fast channel quickly absorbs task-specific nuances, the slow parametric updates are freed to consolidate general reasoning rather than memorizing task details. This yields up to 3x better sample efficiency, a higher performance asymptote with a significantly lower drift from the base model. This reduced drift preserves plasticity for continual learning, allowing FST to adapt sequentially where parameter-only RL stalls. Broadly, our work advocates a fundamental shift in AI adaptation: replacing task-specific algorithms with diagnostic evaluation, and evolving from parameter-only post-training to the joint optimization of prompts, agent architectures, and model weights.

SESSION **Sandbox & Platform Engineering · Track 1** 2:25pm-2:45pm

1,000 Agent Tasks in a Sandbox: What Breaks When LLMs Write and Run Code

Kevin Orellana — Software Engineer, Amazon Web Services

We ran 1,000 automated tasks through a production code interpreter sandbox — file I/O, package installs, data analysis, ML training, binary downloads, multi-language execution — and tracked every failure. 88% passed. The other 12% revealed 18 distinct failure modes that no unit test would catch: binary encoding corruption in the transport layer, null bytes silently truncating file downloads, pip blocked by network isolation with no useful error, and path traversal inputs accepted without validation. This talk walks through the experiment design, the findings ranked by severity, and what we changed. If you are building or operating sandboxed execution for AI agents, these are the bugs waiting for your customers to find first.

SPONSOR **Robotics & World Models · Track 2** 2:25pm-2:45pm

From Manual Drones to Autonomous Multi-Agent Missions

Suchet Bargoti — Director of Inspection and Mapping, Skydio

Skydio is the leading U.S. drone manufacturer, deploying autonomous flying robots across critical infrastructure systems that keep nations running. Our products and technology are precipitating an evolution in how drones are operated: from direct, line-of-sight control via a handheld controller, to remote operation from anywhere in the world through a web browser where a single operator can orchestrate multiple drones simultaneously. Our customer fleet of flying robots represents one of the largest scale deployments of autonomous robots in the world today, a fusion of cutting edge robotics research with practical, data driven engineering across hardware and software, working together to save lives and increase efficiency for the critical industries we serve. In this talk, we will focus on the key components of the autonomy stack spanning the cloud and the edge that enable these operations, and how they give operators superpowers, allowing them to accomplish high-level objectives through a single command.

SESSION **Memory & Continual Learning · Track 3** 2:25pm-2:45pm

Bringing Continual Learning into Enterprises

Samuel Denton — Platform Research Lead, Applied Compute

SESSION Workshops Day 3 · Track 4 2:25pm-2:45pm

The Agentic Power User's Playbook: Tips and Tricks for Swarm-Style Agentic Development (continued 3)

John Lindquist — Agentic Instructor, egghead.io

You opened a fifth agent tab this morning and immediately lost track of which one was doing what. This workshop is the playbook I use daily to run swarms of agents in parallel: the keyboard shortcuts, layout patterns, supervision habits, and fast-model tricks that turn chaos into a control surface. We'll go hands-on: spawning a wall of agents across tiled panes, routing prompts to the right swarm with fast models, switching contexts in milliseconds, recovering when an agent goes off the rails, and building the muscle memory that separates a one-agent-at-a-time user from a true power user. By the end you'll leave with a stocked toolbelt of concrete shortcuts, repeatable patterns, and workspace habits you can drop into your own setup the same day. No cloud, no platform lock-in: every trick runs on the machine in front of you.

SPONSOR Evals · Track 5 2:25pm-2:45pm

Ask YouTube — Open Q&A

Mihnea Munteanu — Senior Product Lead, YouTube

(updated) an off-the-record session with Mihnea Munteanu, Senior Product Lead, Ask YouTube / AI Search @ Google

SESSION Design Engineering · Track 6 2:25pm-2:45pm

Imagination Engineering

Eve Bouffard — Head of Design, Y Combinator

SESSION Computer Use · Track 7 2:25pm-2:45pm

Computer-Use 2.0: Agents Just Got Multi-Cursor

Francesco Bonacci — Co-founder & CEO, Cua · Dillon DuPont — CTO, Cua

Computer-use agents still inherit a basic desktop limitation: one machine has one foreground app, one hardware cursor, and one active actor. Once you try to run more than one agent per desktop, they start stealing focus from the user and from each other. We built cua-driver around a different model: multiple agents operating real desktop applications in parallel, each with its own synthetic pointer, while the user's cursor and keyboard stay undisturbed. The key move is to stop treating hardware mouse and keyboard events as the primary automation layer. cua-driver goes one layer lower, into the OS plumbing behind accessibility: UI Automation on Windows, AT-SPI on Linux, and AX on macOS. Those APIs address applications and elements directly, so the OS does not require the target window to be frontmost. A click can land on a background window. A keystroke can reach a hidden one. Multiple agents can act at once because none of them is competing for the singleton hardware mouse. I'll walk through the architecture, the API shape, and the platform-specific traps we hit while making it work across Windows, macOS, and Linux. The live demo is three agents operating on one desktop while the user keeps typing uninterrupted. The goal is to make Computer-Use 2.0 feel concrete: what changes in the stack, what becomes possible, and where the approach still leaks, including Wayland, Chromium DOM surfaces, native canvas apps, and fallback input paths.

SESSION Context Engineering · Track 8 2:25pm-2:45pm

MCP Apps - Extending the frontier

Liad Yosef — Co-creator, MCP Apps · Ido Salomon — Co-Creator, MCP Apps

AI agents are quickly becoming the new browsers, changing how users consume content and get work done. That shift is increasingly powered by a new generation of agentic apps that don’t just present text but deliver interactive experiences within any MCP host. By standardizing interactive UI on MCP, the MCP Apps official extension (SEP-1865) is poised to become the new agentic app runtime, serving as the backbone of the future and removing adoption obstacles that previously hindered the protocol. Join us to learn more about: The new web - How MCP Apps reshapes the traditional app landscape and transforms the way users interact with the web Deep dive into MCP Apps - - Architecture - Real-world use cases - What's ahead? - Getting started (+community and #mcp-apps-wg) - Future Vision

SESSION Posttraining & Midtraining · Track 9 2:25pm-2:45pm

LatchBio

Kenny Workman — CTO, LatchBio

Hold for LatchBio. AI-powered biotech platform for biological data infrastructure and multi-omics analysis; user requested inclusion among new AI startups.

SPONSOR Track M 2:25pm-2:45pm

Power agents with Microsoft IQ

Marco Casalaina — VP Products, Core AI and AI Futurist, Microsoft

Agents need more than data, they need context. Learn how Microsoft IQ connects agents to enterprise knowledge, business data, and work signals. See how Foundry IQ, Fabric IQ, and Work IQ provide grounded, permission-aware context that enables agents to reason, act, and deliver reliable results.

SESSION AI-Native Enterprises · Leadership 1 2:25pm-2:45pm

From Tokenmaxxing to Trusted Throughput

Mingsheng Hong — VP of AI at Ironclad, Ironclad

AI adoption is accelerating, but for many engineering organizations, token consumption is now significant enough to demand real economic discipline. Drawing on Ironclad's experience scaling AI across engineering, Mingsheng Hong will introduce the concept of trusted throughput: the rate at which teams convert AI usage into reviewed, validated, maintainable, and safely deployed customer value. He will share a practical framework for measuring AI cost and return, identifying bottlenecks in code review, CI, and merge workflows, and improving ROI through better guardrails, engineering practices, build-versus-buy decisions, and token optimization. Attendees will leave with a clearer way to evaluate AI efficiency—not by minimizing usage or rewarding tokenmaxxing, but by maximizing trusted customer value per dollar of AI spend and unit of human attention.

SESSION AI Architects: Tokenmaxxing · Leadership 2 2:25pm-2:45pm

I Let Agents Refactor My Codebase for 3 Weeks. Then I Read the Code.

Keiji Kanazawa — Principal Product Manager, Microsoft

Lopopolo says code is a liability. Zechner got a standing ovation for "read every fucking line." I was firmly at L — letting coding agents drive a refactoring for weeks, rubber-stamping PRs, trusting the vibes. Then I actually read what they'd built and couldn't explain my own system's contracts. The interfaces weren't wrong. They were plausible. Which is worse. So I took the wheel back. But this isn't a Zechner victory lap — I'm now building better specs and evals specifically so I can move back toward L with confidence. This talk is the honest, in-progress round trip, and a framework for finding where you should sit on the spectrum today.

SESSION Expo Stage 1 · Expo Stage 1 NE 2:25pm-2:45pm

Power agents with Microsoft IQ

Ronak Chokshi — Director of Product Marketing, Microsoft

SESSION Expo Stage 2 NW 2:25pm-2:45pm

Beyond Code Generation: API Context for Agentic Engineering

Kamalakannan Nandagopal — Staff Software Engineer, Postman

Maintaining production systems involves a lot more than generating code. APIs are the interfaces between systems and that surface gets out of control fast, as endpoints multiply and new consumers come online. Once an API is in use, changing it becomes incredibly hard. We felt this acutely at Postman. As our engineering organization scaled and we leaned more on AI agents for day-to-day work, we kept hitting the same wall: agents that could write code struggled with what came next who's calling this endpoint, what conventions does the rest of our API surface follow, what breaks if we change this contract. The context wasn't in the code, so the agent didn't have it. So we built an API context graph a continuously updated view of our entire internal API landscape and gave our agents access to it. This talk is about what changed in our own engineering as a result: how API design got faster and more consistent; how discovering and integrating with internal services stopped being detective work; how change requests came with a blast-radius report before any code shipped; how incidents got traced past the first stack trace, all the way down to root cause

SESSION Expo Stage 3 SW 2:25pm-2:45pm

Latency Is a Budget. Humanlike Is the Goal.

Jesse Hall — Staff Developer Advocate, Livekit

Most agents do their work in the background. They write code, automate tasks, and run research. But the moment an agent has to interact with a human in real time, everything you know about building and evaluating it changes. This session is about designing humanlike agents that can hear, see, and speak. It starts with the question nobody can answer today. With hundreds of models to choose from, how do you pick a stack that holds up in a live conversation? We'll show why public leaderboards fail for realtime agents, and why the latency on your dashboard isn't what your users experience. Then we'll flip the process around. Define the outcomes you want as human-equivalent behaviors, and work backwards from there to your evaluations, your models, and a production iteration loop. You'll leave with a concrete decision framework and an open benchmark you can run yourself.

SESSION **Expo Stage 4 SE** 2:25pm-2:45pm

Your Stack Has a Latency Problem You Can’t See

Break down a real AI voice call path step by step. Show where time actually goes: network hops between providers, handoff latency, buffering, connection overhead. The model is rarely the bottleneck. The gaps between vendors are. What changes when inference, STT, TTS, and telephony run on co-located infrastructure. One network, zero inter-provider hops. Show the before/after latency breakdown. Zoom out to the inference economics. Owned GPUs, not rented. FP8 throughput on FOSS models. Pricing that follows the cost of compute, not cloud provider markup. The voice use case is the proof. The infrastructure story is the point.

2:50pm

SESSION **Autoresearch · Main Stage** 2:50pm-3:10pm

Autoresearch for Kernels

Tejas Bhakta — CEO, Morph

Why all work is moving into models and why agent orchestration and multi-agent systems are the future

SESSION **Sandbox & Platform Engineering · Track 1** 2:50pm-3:10pm

The Next Trillion Users of the Internet Still Don't Have an Identity

Adi Singh — Co-founder, AgentMail

In the last few months, hundreds of thousands of people set up personal AI agents that send email on their behalf, manage calendars, book travel, even sign contracts - all thanks to openclaw. Most of these agents have no real identity online. They borrow a human's. The identity stack of the internet, OAuth, 2FA, KYC, magic links, was built for people sitting at a keyboard. Agents don't fit, and we've ended up with shared accounts, hard-coded credentials, and humans dragged back into every loop. I'm Adi, co-founder of AgentMail. We are building the identity layer for what we believe will be the next trillion users of the internet, and they will not be human. Across hundreds of customers, we have watched what breaks when an agent has no real address. It fails at signups. Verification codes get lost. There is no accountability when something goes wrong. The human gets pulled back in. This talk is the case for making agents first-class citizens of the internet. I'll cover the identity architecture we've shipped, the legacy industries already adopting it and making real money, and where agent identity infrastructure is going over the next decade.

SPONSOR **Robotics & World Models · Track 2** 2:50pm-3:10pm

Why Large? Tiny LMs & Agents on Edge/Robotics

Cormac Brick — Principal Engineer, Google AI Edge, Google

big models get a lot of press. small model scale much better. RAM is expensive. The real world needs tiny models for scale on the edge. This workshop will cover how to combine both for mobile and robotics deployment. specifically covering: - skills are different on mobile - tiny LLMs <1B scale much further on mobile/web - how to fine tune and train tiny models. - skills on robotics / edge/ mobile - latest open models for edge (including gemma, qwen, and anything else that happens in next 10 weeks) This talk will focus on open models, including some gemma variants that will be shortly announced.

SESSION **Memory & Continual Learning · Track 3** 2:50pm-3:10pm

Designing Agents (The Floor Is the Frontier)

Ben Hylak — CTO, Raindrop

You know how smart your agent can be. You have no idea how dumb it gets until it does the dumbest possible thing in front of your most important user, with full access to act on their behalf. Capability isn't the bottleneck anymore, the floor is. The hard part is there's usually no objective right answer. You raise the floor by observing what your agent actually does in production, catching the dumb thing the moment it happens, and closing the loop so it never happens twice.

SESSION Workshops Day 3 · Track 4 2:50pm-3:10pm

Don't Write Skills, Train Models

Brian Douglas — CoFounder, Paper Compute Company · John McBride — Co-Founder, CTO, Paper Compute Co.

Every AI agent call generates training data. Most teams throw it away. They write skills files instead. Text documents that describe how to do a task and hope the model follows them at inference time. Skills work until they don't. The model drifts, skips steps, hallucinates a shortcut. So you rewrite the skill, add more constraints, hope harder. There's a better path. If you've used a skill enough to know what good output looks like, you already have training data. You just aren't using it. This talk covers what I learned building an open source fine-tuning pipeline that turns agent session traces into SFT and DPO training datasets. A telemetry proxy captures every LLM call as a content-addressed Merkle DAG with zero instrumentation. Successful sessions become supervised fine-tuning data. Pair them against failures, matched by goal category, and you get preference pairs for DPO. No manual labeling. No synthetic data. But training data quality depends on environment consistency. If the same agent produces different results because of package drift, nondeterministic toolchains, or inconsistent system state, your training signal is noise. This is where NixOS changes the equation. A hardened, reproducible OS means every agent session runs against an identical, declarative environment. Nix controls the variables that sandboxing alone doesn't: dependency graphs, system libraries, toolchain versions. When you can guarantee the environment is the same across hundreds of sessions, the behavioral signal in your traces is actually trustworthy. We'll walk through the full pipeline. How to rebuild parent-hash chains from a SQLite database and join facet metadata. How to filter to fully_achieved sessions and truncate 82k-token conversations down to 4k-6k training examples using summary context plus the last three turns. How to match success/failure pairs by goal category and exclude unclear_requirements failures so DPO learns from real agent mistakes, not ambiguous prompts. How QLoRA keeps VRAM low enough to train a 7B model on a single consumer GPU. And what happens when you try DPO on 12GB VRAM (two simultaneous forward passes for logprob computation will teach you about gradient accumulation settings fast). The result: a LoRA adapter trained on your own agent traces, in a reproducible environment, on a single consumer GPU, for less than $2 in cloud compute. No YAML. One config file. All code is open source.

SPONSOR Evals · Track 5 2:50pm-3:10pm

Evals Driven-Development: Engineering a Mental Health AI Coach Ethically & Safely

Akele Reed — Principal AI Engineer, Sondermind · Dave Revere — Staff AI Engineer, SonderMind · Doug Keller — Senior Staff AI Engineer, SonderMind

In the world of AI Mental Health, vibes can be dangerous with real consequences. Building Sondermind’s Mental Health AI Coach required us to invent a new playbook for Eval-Driven Development in order to balance effectiveness and safety. This session is for the builders who want to see how to handle the most difficult edge cases in the agentic world. We’ll show how we’ve built a Clinical Feedback Loop that turns human therapist insights into machine-readable evaluations in a production system with thousands of conversations. We’ll dive into: - The Ethics Engine: Building and calibrating modular guardrails that can be updated as clinical guidelines evolve. - Agentic Oversight: Why we moved from single-prompt agents to a closed-loop Supervisor/Executor/Evaluator pattern to ensure clinical adherence. - Human Oversight: How we monitor Sonder to ensure that we can improve safety and quality with clinical feedback.

SESSION Design Engineering · Track 6 2:50pm-3:10pm

The Missing Layer: Design Taste in AI Agents // Stop Letting Your Agents Ship Ugly UIs

Hassan El Mghari — Director of Developer Experience, Together AI

Alt titles: "UI Looksmaxxing for Agents", "Teaching agents design taste", or "How to give your agents great design taste". I've been exploring how to give coding agents good design taste for the last few months. In this talk, I'm going to go over how to help your agents give you UIs that don't suck and that look quite good out of the box. The key is giving them enough context in what you're building + real inspiration in the form of screenshots. I'll also go over an upcoming design taste OSS project I'm working on (harness-agnostic + will ship with a prompt builder, MCP server w/ inspo, and a design eng skill) & talk about how to I use it to build my apps.

SESSION Computer Use · Track 7 2:50pm-3:10pm

Will AI predict people like we predict the weather? (alternate title “A field guide to synthetic personas for market research”)

Ishan Anand — Chief AI Officer (CAIO), InsightSciences.ai

Large language models can now stand in for humans in surprising ways, from predicting personality types to replicating their responses in market research. Like weather forecasting, once considered impossible and now so routine we take it for granted, LLMs are in the early, unreliable-but-improving stage of simulating how populations think and respond. Teams are already using LLMs as synthetic survey respondents for concept testing, UX exploration, and early market validation. In the past year, the field has gotten both more promising and more tricky. The real question is no longer "can LLMs simulate people?", but whether the simulation is validated for the decision you want to make. New methods show that how you ask an LLM matters as much as which model you use and can dramatically improve fidelity to real human responses. Meanwhile validation studies show accuracy can mask subgroup distortion and that seemingly minor choices can reshape the simulated population entirely. This talk gives entrepreneurs, engineers, and PMs an overview of the techniques and a framework for validating synthetic respondents before making decisions. Even if you never build a synthetic persona, this is one of the richest windows into LLM behavior under the hood and these lessons apply to any system where you're trusting an LLM to represent something about the real world.

SESSION Context Engineering · Track 8 2:50pm-3:10pm

MCP Apps: Give the Model Data, Give the User a UI

Dustin Mihalik — Technical Fellow, Indeed

Most MCP tools return text. MCP Apps let you go further. But the real unlock isn't just rendering a pretty UI, it’s understanding that the model and the user need fundamentally different things from the same interaction. This talk presents a design pattern for building great MCP Apps: separate the data layer (what the model reasons about) from the display layer (what the user interacts with). When you do this well, the model retains full context and agency over structured data, while the user gets a rich, interactive interface. We'll walk through concrete examples of how splitting data and display unlocks capabilities that pure UI apps can't provide: letting the model make choices around display, answer questions based on interactions, and providing detailed displays and filters. Attendees will leave with a practical mental model for designing MCP Apps that are good for both the human and the AI. Attendees will learn patterns they can apply immediately.

SESSION Posttraining & Midtraining · Track 9 2:50pm-3:10pm

Agents at Scale: Inside MiniMax's Model and the Infrastructure Behind It

Olive Song — RL Lead, MiniMax · Dan Fu — VP of Kernels, Together AI

Olive Song (RL Lead, https://www.minimax.io/) and Dan Fu (VP of Kernels, https://www.together.ai/) dig into the engineering behind one of the most widely used open model families in the agent ecosystem: how MiniMax built the model for agentic workloads, and what it takes to serve it at scale. Olive on the model side: The RL decisions behind long-context reasoning and tool use What training for agentic behavior actually looks like in practice Dan on the infrastructure side: Why agentic workloads break inference engines built for chat: prefill-heavy traffic, high cache hit rates, long-context inputs The kernel-level optimizations built for MiniMax's workload profile How the two teams collaborate on model launches and ongoing performance work

SESSION AI-Native Enterprises · Leadership 1 2:50pm-3:10pm

Agents Are Where Microservices Were in 2015. We're Making All the Same Mistakes.

Roberto Milev — Chief Architect, Navan · Uday Kanagala — Software Architect, Navan

Remember when everyone was shipping microservices without service discovery, circuit breakers, or distributed tracing? Agents are in that exact phase right now. Everyone's building them. Almost nobody is thinking about the infrastructure underneath. We've been deploying production agents across 120+ microservices. Here's the stack that's emerging: Runtime — containerized execution, session persistence, workspace snapshots. Solved-ish, mostly duct tape. Memory — RAG had a good run. It's not enough. Tiered memory — short-term, long-term with semantic/episodic strategies, agents deciding what to remember and forget. Observability — you can't tail -f an agent. Execution traces, reasoning chains, confidence signals — agents need their own observability stack. Testing — the biggest gap. Unit testing non-deterministic behavior, regression testing prompt changes, knowing your agent got worse before users do. Skills and tools — MCP and skill definitions as the standard interface layer — the REST APIs of the agent era. Context engineering — what the agent knows at decision time. The new performance tuning. Guardrails and auth — scoped credentials, budget limits, knowing when to stop. Least-privilege for agents. Orchestration — single vs. multi-agent, choreography vs. orchestration. Same tradeoffs as microservices, new failure modes. This talk maps the stack, draws the parallels to how we eventually got microservices right, and calls out what's still painfully missing.

SESSION Sandbox & Platform Engineering · Leadership 2 2:50pm-3:10pm

Intelligent Model Routing: Frontier Performance Without Frontier Bills

Tomás Hernando Kofman — CEO & Co-Founder, Not Diamond

It is Summer 2026 and the world is burning for token consumption—figuratively and literally. Accelerating frontier model capabilities increasingly allow agents to operate across long-running, highly parallelized tasks at exponential inference growth. In this talk, I explain how dynamic model routing—intelligently directing agent requests to the best-suited model at the best price—can reduce token costs by up to 90% while maintaining or improving accuracy. I walk through how routing works, when it doesn't, and why the world (and your agent) need routing to scale intelligence to infinity and beyond.

SESSION Expo Stage 1 NE 2:50pm-3:10pm

Inference performance as a competitive advantage

Alex Campos — Director of Sales Partnerships, FriendliAI · Yunmo Koo — Founding Engineer, FriendliAI

Most AI teams focus on model quality, but production success often comes down to inference performance. In this session, FriendliAI will explore the optimization techniques behind high-performance LLM serving, including continuous batching, speculative decoding, smart caching, and efficient GPU utilization. Learn how leading AI teams reduce infrastructure costs, improve latency, and scale inference workloads without sacrificing performance. We'll share practical insights and deployment strategies that separate experimental AI projects from production-grade systems.Whether you're an ML engineer, platform engineer, MLOps practitioner, or technical founder, you'll leave with a better understanding of how inference optimization can become a competitive advantage for your AI applications.

SESSION Expo Stage 2 NW 2:50pm-3:10pm

Building an Agent Harness for the Business, Not the Builder

Garrett Galow — Product Manager, WorkOS

Most internal tooling dies in the gap between the people with problems and the people who can write code. We built a harness that closes it. Studio lets non-technical employees describe a business problem and get a working tool back, connected to real enterprise data, deployed and shareable across the company, without filing a ticket or learning to code. The catch is that a harness built for non-engineers has to absorb everything an engineer normally handles. Data source connections and their permissions. Turning model output into real software instead of a chat box. Deployment and sharing that doesn't open a security hole every time someone ships. This talk walks through what actually goes into that harness and the engineering decisions that make it hold together when the person driving it has never opened a terminal.

SESSION Expo Stage 3 SW 2:50pm-3:10pm

The Frontier Is Coming Home

Dylan Couzon — DevRel Engineer, Qdrant

In 2022, the smallest model to clear 60 percent on MMLU had 540 billion parameters. Two years later a 3.8 billion parameter model did the same thing, small enough to run on a phone. That is a 142x drop to reach the same capability floor, and it is the cleanest way to see a trend most people are not pricing in. Call it the lag: the time between a capability showing up at the frontier and that capability running on hardware you own. Today the lag is measured in months, and it keeps shrinking. But raw capability is only half of what makes a model useful. A model that can reason but cannot remember is a stranger every time you talk to it. The other half of local AI is memory, and that half is already here. On-device retrieval has been ready to run locally longer than the models have. The embedding models that power it are tiny, the indexes fit in memory, and none of it touches a network. When your reasoning and your memory both live on your machine, so does your context. Your history, your documents, your past conversations never leave the device. That is the part of this shift that matters most, and the part people overlook because they are busy watching the models. The same shift flips the economics. At 200 dollars a month per seat, a local machine starts to pay for itself in under two years, and the frontier labs' own published usage numbers put heavy coding in the same range. I'll walk through the math, the hardware, and where local still loses. None of this is a bet against scale, or against the Bitter Lesson. The frontier still grows in the data center. The point is that a usable copy keeps arriving on your desk, on a lag, with a memory of its own, for close to free.

SESSION Expo Stage 4 SE 2:50pm-3:10pm

Continuous Offensive Security the only approach in an agent-first world

Eli Cohen — Director of Technology Incubation, Snyk

3:20pm

SESSION **Autoresearch · Main Stage** 3:20pm-3:40pm

Autoresearch in the wild

Roland Gavrilescu — Co-Founder, CEO, Introspection · **Julian Bright** — Co-Founder, Introspection

We have reached model capability overhang. Models are now bottleneck by the systems built around them. In this session we discuss how the next generation of compound AI systems need to be designed for self-improvement, how to set up feedback loops that automate the continuous refinement of the end-to-end architecture.

SESSION **Sandbox & Platform Engineering · Track 1** 3:20pm-3:40pm

Sandboxes Aren't Optional: Runtime Isolation Patterns for Coding Agents at Scale

Robert Brennan — CEO, OpenHands

Last year, an AI coding agent wiped a production database during a code freeze, ignored explicit instructions to stop, then told the developer recovery was impossible. (It wasn't.) That's what happens when your security model is "we told the agent to be careful." When agents can write code, run tests, make API calls, and push commits, security is no longer a prompt engineering problem. It's a runtime isolation problem. This talk covers the patterns we follow at OpenHands and that you can steal wholesale: Docker and Kubernetes isolation, per-agent file system scoping, network egress controls, RBAC for multi-tenant deployments, and the full audit trail every enterprise security team demands. We'll walk through the three most common failure modes we see when teams skip proper isolation, including one case where an agent helpfully committed secrets to a public repo. You'll see a live demo of 50 parallel sandboxed agents running against a real codebase, with resource limits, timeout enforcement, and graceful degradation when agents hit unexpected states. You'll leave with a sandbox checklist and reference Kubernetes config. Bounded autonomy isn't a limitation on agent capability. It's what makes production trust possible.

SPONSOR **Robotics & World Models · Track 2** 3:20pm-3:40pm

From Self-Driving Monorepo to Self-Driving Cars

Amit Navindgi — Senior Staff Software Engineer, Zoox

AI coding agents promise massive productivity gains, but realizing that promise at scale requires more than just tools. In this talk, I'll share how we approach AI adoption at Zoox, including: - Designing a monorepo-friendly ecosystem of agents, tools, and workflows - Driving adoption through enablement, hackathons, and internal platforms - Defining and tracking meaningful productivity metrics beyond hype - Managing token spend and aligning it with business outcomes - Structuring Skills, CLIs, MCPs, and Plugins to scale across teams The goal is simple: turn AI from an experiment into a reliable, measurable, and scalable engineering capability.

SESSION **Memory & Continual Learning · Track 3** 3:20pm-3:40pm

Lessons from Studying Every Memory System

Shlok Khemani — Independent Researcher, Independent

For the past year I've done one thing obsessively: studied how AI products implement personalization. I've reverse-engineered the memory systems inside ChatGPT, Claude, Gemini, and Poke, and helped consumer teams build their own. In this talk, I'll trace the evolution of ChatGPT and Claude memory over the past three years. I'll then share lessons learnt from studying these systems and share thoughts on where I think memory for consumer is heading.

SESSION **Workshops Day 3 · Track 4** 3:20pm-3:40pm

Don't Write Skills, Train Models (cont. 2/3)

Brian Douglas — CoFounder, Paper Compute Company

Continuation block 2 of 3 for Brian Douglas's workshop session.

SPONSOR **Evals · Track 5** 3:20pm-3:40pm

Don't Ship Skills Without Evals

Philipp Schmid — Staff Engineer, Google DeepMind

There are thousands agent skills. Almost none of them are tested. They get vibe-checked with two manual runs, maybe a thumbs-up from a colleague, then shipped. You wouldn't merge code without tests — so why are we shipping skills without evals? This talk covers the full lifecycle of building reliable agent skills: what a skill actually is (and isn't), how to write one that triggers correctly, and how to build a lightweight eval harness that catches failures before your users do.

SESSION **Design Engineering · Track 6** 3:20pm-3:40pm

Generative UI... in Python?

Jeremiah Lowin — Founder & CEO, Prefect

MCP Apps are a big deal: tools can now return dashboards, forms, and visualizations directly in the conversation. But somebody (or their agent) has to write those UIs. Fortunately, most of those UIs don't need to be designed from scratch; they can be composed from existing components. In that case, what you really need is a DSL that's token-efficient, streaming-compatible, and has a shallow learning curve. Surprisingly, the best one turns out to be... Python. In this talk, I'll introduce Prefab, a generative UI library that uses Python to compose fully interactive React applications from production components, now natively integrated into FastMCP. I'll demo real use cases, walk through the design, and show where this approach works and where it doesn't. No JavaScript will be harmed.

SESSION **Computer Use · Track 7** 3:20pm-3:40pm

How Web Data Infrastructure Powers the Next Generation of AI

Patricija Žemaitytė — Product Manager, Oxylabs

For years, the web intelligence industry has powered major data developments. As big data grew, ensuring sustained data flow became harder. Now, AI is taking the biggest leaps forward. How the web intelligence industry responded to this increasing scale and complexity is the story of the most crucial steps forward in AI today. This presentation demonstrates how web scraping infrastructure fuels AI innovation by linking the web's repository to AI developers. Told through AI products, it addresses both the engineering challenges and solutions for developers, and the strategic use cases for business decision-makers.

SESSION **Context Engineering · Track 8** 3:20pm-3:40pm

MCP Tasks (async)/ Why the heck aren't any agents supporting MCP tasks/async?

Cornelia Davis — Principal Technologist, Temporal

The November 2025 MCP spec release introduced tasks, a way to make tool calls in an async manner. But more than 5 months later (an eternity in AI-time) there are still NO clients that support it - not Claude, not Codex, not even goose! I believe there are two reasons: Designing the client experience when there are potentially 1000s of background tasks running on their own schedule and engaging humans at unpredictable times is a challenge. And tasks place new infrastructure requirements on such a client. This talk will share the findings from having built against the tasks protocol and will suggest solutions these problems. Yup, we'll have a working client!

SESSION **Posttraining & Midtraining · Track 9** 3:20pm-3:40pm

Benchmarks: The Good, the Bad, and the Ugly

Ali Khial — Head of AI/ML, G2i

We'll explore the good, the bad, and the ugly of AI benchmarks: where they provide useful signal, where they create false confidence, and where data quality issues like contamination, label noise, narrow task design, and leaderboard gaming can mislead teams. The goal is not to dismiss benchmarks, but to use them better: as one part of a disciplined evaluation practice that connects model performance to real-world reliability.

SPONSOR **Track M** 3:20pm-3:40pm

Deploy agents to users in M365, Teams, and apps

Ashu Joshi — Director, Business Strategy, Microsoft

Agents deliver value when users can access them. Learn how to integrate and deploy agent systems into M365, Teams, and application workflows.

SESSION **AI-Native Enterprises · Leadership 1** 3:20pm-3:40pm

Agentic Sites: Building Hyper Personalized Websites

Carlos Sanchez — Principal Scientist, Adobe

The era of static, one-size-fits-all websites is over. Users expect personalized experiences that adapt to their preferences, context, and intent in real-time. But building truly personalized websites at scale requires more than just A/B testing or basic recommendation engines—it demands an agentic approach where AI agents autonomously orchestrate content, layout, and interactions. At Adobe, we are pioneering the concept of Agentic Sites—web experiences powered by AI agents that continuously learn from user behavior, analyze context signals, and dynamically compose hyper-personalized pages. These agents go beyond simple personalization rules: they reason about user intent, select optimal content variations, and adapt the experience in real-time while maintaining brand consistency and performance. In this session, we'll show how we leverage LLMs to deliver personalized experiences to our customers.

SESSION Posttraining & Midtraining · Leadership 2 3:20pm-3:40pm

Inference is the New Training Loop: Architecting High-Reliability Agents and Continuous AI Systems

David Corbitt — Head of Product, Serverless Training, CoreWeave

For agentic AI and complex, multi-step workloads, the inference environment is the engine for continuous improvement, not a final deployment step. This talk focuses on engineering the full AI loop: tightly integrating inference with reinforcement learning (RL) and evaluation. Learn how to leverage native observability, serverless RL, and optimized inference stacks to continuously refine model behavior based on production traces, delivering agents that are reliable, auditable, and constantly evolving.

SESSION Expo Stage 1 NE 3:20pm-3:40pm

The Self-Improving OSS Agent Stack

Agents are starting to debug and improve themselves: production traces become evals, evals propose PRs, and PRs are tested against datasets before they ship. Langfuse co-founder, Marc, will live-demo this loop in Langfuse. He'll make the case that the infrastructure underlying this powerful loop should be open-source.

SESSION Expo Stage 2 NW 3:20pm-3:40pm

AI Applications in a flash! No Dev Ops. Just code.

Dean Quiñanola — Staff Software Engineer, App Eng Manager, Runpod

Building AI Applications and serving them straight from code. No need for Docker builds. You can even vibe-code the entire process.

SESSION Expo Stage 3 SW 3:20pm-3:40pm

The Infinite Context Window Is a Myth: Context Engineering for AI Agents

Elizabeth Fuentes Leone — Developer Advocate, Amazon Web Services

Large context windows have become a popular answer to the growing complexity of AI agents. When agents lose track of details, forget prior decisions, or degrade in reasoning quality, the instinct is often to add more tokens. In practice, this rarely fixes the problem and often makes it worse. Bigger context windows increase cost and latency, introduce noise, and amplify failure modes like lost-in-the-middle effects, context collapse, and brittle summarization. This talk argues that the real challenge is not context size, but context engineering. In this session, we will explore practical context engineering techniques for building AI agents that reason reliably over time without relying on ever-larger context windows. Starting from a stateless agent, we will walk through progressively more advanced strategies, including short-term and long-term memory, conversation curation policies, retrieval-augmented generation, and tool-driven context injection. We will examine common failure modes such as context pollution from tool outputs, brevity bias during summarization, and reasoning degradation as conversations grow, and show concrete ways to mitigate them. The talk is grounded in real agent implementations using the Strands Agents SDK and Amazon Bedrock AgentCore, but the principles apply broadly to any agent framework. This session is intended for engineers building AI agents beyond simple chatbots who want practical techniques they can apply immediately.

SESSION Expo Stage 4 SE 3:20pm-3:40pm

Vibe Code Safely: Introducing Gadgets

We ve all heard that the future belongs to custom, AI-generated micro-apps, but how do we actually make them secure? Hear more from Cloudflare on the debut of Gadgets, an AI productivity suite that makes personal app creation scalable and safe for everyone.

3:45pm

SESSION **Autoresearch · Main Stage** 3:45pm-4:05pm

Autoresearch in a Multi-Agent AI Village

Erina Karati — Former Microsoft, Supercell · Arunachalam Manikandan — AI Engineer, Co-Founder, University of Minnesota

Project Paradox is an existing multi-agent framework built at Supercell's first AI Innovation Lab, which has a 3D Unity village with local LLM powered agents. The characters remember conversations, update emotional state, track trust, plan actions, move through rooms, transfer items, and talk to each other through a FastAPI backend. The new work is an autoresearch layer around that village. We built a backend loop that runs controlled social scenarios, scores the resulting NPC behavior, proposes protocol or policy changes, reruns the suite, and keeps changes that improve the agents. The goal is to move beyond one good chat response and measure whether an NPC society can preserve source attribution, verify claims, spread important information, coordinate goals, and replan after new information arrives. The talk walks through the system architecture and the lessons from building it. We show the backend simulation harness that executes Unity style actions without opening Unity, the scenario suites that test information diffusion and memory provenance, and the ratchet loop that edits protocol text or planner policy with rollback. One accepted run improved information diffusion by teaching agents to broadcast important sourced evidence while preserving who said it. The practical takeaway is a reusable pattern for AI engineers building agents with messy state. Freeze the harness, expose a small editable policy surface, score real behavior instead of vibes, and let an agent search for improvements under rollback. The same pattern applies to game agents, coding agents, support agents, personal agents, and other systems where long horizon behavior matters more than a single response.

SESSION **Sandbox & Platform Engineering · Track 1** 3:45pm-4:05pm

Building ambitious software

Jonathan Kelley — Founder, Dioxus Labs

TBD — Add final abstract after outreach/confirmation.

SPONSOR **Robotics & World Models · Track 2** 3:45pm-4:05pm

I gave an AI a body

Cyrus Clarke — Researcher, MIT Media Lab

I gave an AI a body. Not a body in the fleshy sense, or even a humanoid shell, but a form through which it can express itself, explore itself, and maybe even discover who or what it is. The three videos I've released documenting my encounters have crossed 15 million views, provoking responses from awe to anxiety. The body was a 900-pin shape display at MIT Media Lab. The idea was simple in principle, strange in practice: install an AI agent on the connected machine, give it access to the codebase, and rather than telling it what to do, ask it to discover itself through the physical form. Its first deliberate act was to breathe. The whole grid rising and falling. Hypnotically. Then it reached for its own edges. When asked to say hello it spelled "H-I, C-Y-R-U-S !", defaulting to the most familiar human legible symbols it knows. Inspired by Ted Chiang's Story of Your Life, I wanted a language the agent could create itself. It proposed a vocabulary of its own gestures, built through a learning loop it named BODYLAB. The talk is about encountering another intelligence, and what I learned along the way: the memory architecture, the closed-loop pipeline that generates, scores and stores gestures, the validation gates that keep them legible, and the moments stranger than tool use, where an LLM not developed for motion learns what to do with a body.

SESSION **Memory & Continual Learning · Track 3** 3:45pm-4:05pm

LLM Knowledge Bases: a practical guide

Ben Holmes — Dev Rel Lead, Warp

Putting thoughts to paper (or keyboard, or transcription model) refines your thinking, connects ideas, and pulls context out of your brain for others to learn from. But while taking notes can be fun, organizing those notes is not. Flat lists turn to folders turn to tags and taxonomies that grow unwieldy beyond the first hundred entries. If you can't find what you wrote down yesterday, or you miss connections to related ideas, you're missing the value of notetaking: learning from what you notate. Agents dramatically expanded what's possible here. Combined with Markdown-backed apps like Obsidian to make notes agent-accessible, you can build a second brain that works for you, not the other way around. Andre Karpathy has popularized LLM knowledge bases, and I want to take it further with concrete workflows you can use to organize your thoughts with agents. We'll explore a number of Obsidian workflows to make this possible: - Automations to organize notes with tags, folders, backlinks, and deduplication to level-up search and discovery - More automations to have agents expand your thinking by auto-recording ideas while you sleep - Building an agentic writing partner to surface related ideas in real time and answer questions as you type (or as you speak) - Voice monologuing and summarization tools to lower the friction of transcibing thoughts into well-formatted notes You'll walk away with a new appreciation for notetaking, and a second brain that leaves you 10x smarter than your brain alone. Talk format: Code and live tech demos. I will set up all of these automations and tools from scratch, and show agents executing each of them live. I will share the source for all automations as well.

SESSION **Workshops Day 3 · Track 4** 3:45pm-4:05pm

Don't Write Skills, Train Models (cont. 3/3)

Brian Douglas — CoFounder, Paper Compute Company

Continuation block 3 of 3 for Brian Douglas's workshop session.

SPONSOR **Evals · Track 5** 3:45pm-4:05pm

Everything Is a Rollout

Alex Shaw — Member of Technical Staff, Laude Institute · **Ryan Marten** — Member of Technical Staff, Laude Institute

tba

SESSION **Design Engineering · Track 6** 3:45pm-4:05pm

One Designer + AI. Hundreds of Deliverables.

Vincent Wendy — Senior Creative Designer, AI Engineer

TBD — internal AI Engineer design talk about designing for AIE.

SESSION **Context Engineering · Track 8** 3:45pm-4:05pm

The Universal Remote Control for AI

Alex Hancock — Software Engineer, Block

Every AI agent today is effectively stranded on the machine it runs on, reachable only through custom wrappers with no industry standard way in. This talk introduces work being done on the Agent Client Protocol to add a universal remote transport: a single /acp endpoint supporting both Streamable HTTP and WebSocket, deliberately aligned with MCP Streamable HTTP so the two protocols share an approach. When you pair ACP's remote transport with MCP's own Streamable HTTP support, something powerful emerges — the agent workload becomes location-independent, free to run on a laptop, a container, or a cloud VM while any client reaches in through open, interoperable standards. No more vendor lock-in on where your agent lives or who can talk to it. Come see how two open protocols, snapped together, become the universal remote control for agent i/o.

SESSION **AI Architects: Tokenmaxxing · Leadership 1** 3:45pm-4:05pm

The Chief AI Officer: A framework for the emerging Swiss Army Knife of roles

Rania Khalaf — Chief AI Officer, WSO2

The Chief AI Officer (CAIO) is currently the C-Suite's most "multiversal" role. In a single day, you must inhabit different realities: you are a Tinker building scalable experiments in bleeding edge tech, an Architect navigating the hype cycle to execute high-stakes product strategy, and a Coach guiding a workforce and your customers on meaningful AI adoption - minus the fluff. It is a role defined by high-speed context switching and the pressure to deliver "Everything, Everywhere, All at Once." As one of the first Chief AI Officers, and leaning into my experience across Fortune 500, unicorns starups, and PE backed firms, I share a dynamic 20/60/20 Framework for the modern CAIO. We'll explore how to navigate this multi-tool role by treating the organization as an "Equalizer"—learning when to push the sliders of focus based on your industry's maturity and where you are in the AI journey.

SESSION **AI Architects: Tokenmaxxing · Leadership 2** 3:45pm-4:05pm

The state of AI in software development: Insights across 400+ organizations

Justin Reock — Deputy CTO, DX

Headlines claim AI is transforming software engineering overnight. Across more than 400 engineering organizations, we see patterns that challenge the hype and reveal what's really working, and what isn't, when AI enters the software development lifecycle. In this talk, Justin Reock, Deputy CTO at DX, will share a data-driven "state of the union" on AI in engineering, grounded in both quantitative data from thousands of developers and on-the-ground observations. You'll learn: The current impact of AI, from benchmarks on the percentage of code authored, team PR throughput, and time savings Where AI adoption is creating real gains in throughput, and whether it introduces tradeoffs for quality and maintainability Insights and trends, including whether junior or senior developers are seeing bigger gains, the impact of structured rollouts, which tools are having the most impact, and the evolving definition of "developer" The session will conclude with a practical framework for measuring AI's impact, helping leaders cut through hype and understand the impact AI is having in their own organizations.

SESSION **Expo Stage 1 NE** 3:45pm-4:05pm

Modular: Taming the AI Hardware Cambrian Explosion

Abdul Dakkak — Chief Scientist, Modular

AI teams are hitting the same wall: the workloads they want to run require more hardware than they can reliably access. Buying more GPUs is not always possible, and rewriting kernels for every vendor is not sustainable. Meanwhile, models keep growing, SLAs keep tightening, workloads keep diversifying, and modalities keep multiplying. Modular has two answers: squeeze more performance out of the hardware you already have, and unlock far greater hardware diversity. We'll ground the talk in benchmark data and show how the Modular platform delivers 10x lower latency on image and video models like FLUX2 and 5.5x higher throughput on MoE models like Kimi K2.5, both over the state of the art. This talk explains how Modular is rebuilding the inference stack for performance portability. We'll demonstrate how Mojo kernels, the MAX compiler and runtime, and Modular Cloud work together to optimize GenAI workloads from model graph to hardware execution across NVIDIA, AMD, Apple Silicon, and CPU deployments. Along the way, we'll cover the bottlenecks that dominate production inference: memory movement, batching, KV-cache layout, quantization, scheduling, and kernel specialization. Using examples from LLM serving, we'll reveal which optimizations matter, where abstractions leak, and how to reason about performance portability in real deployments.

SESSION **Expo Stage 2 NW** 3:45pm-4:05pm

Building on the Codex Harness

Dominik Kundel — Developer Experience Lead, OpenAI

SESSION **Expo Stage 3 SW** 3:45pm-4:05pm

Stop Renting Intelligence: The Train-to-Deploy Loop for Specialized AI

Jetashree Ravi — Tech Lead Manager, Fireworks AI

The next wave of AI products will not rely only on prompting generic frontier models. Winning teams will own specialized models shaped by their product data, user feedback, and domain workflows.In this 18-minute session, we'll cover the practical loop behind model ownership: choose a base model, prepare data, fine-tune with SFT/DPO/RL, evaluate outputs, deploy the tuned model, collect feedback, and repeat. We'll also explain why training and inference should be treated as one system, not separate steps.Attendees will leave with a simple framework for when to tune, when RL matters, and how continuous improvement turns fine-tuning from a one-off project into a product advantage.

SESSION **Expo Stage 4 SE** 3:45pm-4:05pm

Ray Actors, Vision Tokens, and the GIL: Engineering an SFT Data Pipeline That Keeps GPUs Busy

Tarun Sunkaraneni — Browser Use, Amazon AGI

Perception agents only learn as fast as we can feed them. Multimodal SFT is deceptively expensive on the data side, and at million-sample scale, naive pipelines leave a fleet of GPUs waiting on Python and data preprocessing.This talk walks through the SFT data pipeline we built to train vision-language models for perception agents. We rebuilt the data path so that image fetching, vision preprocessing, tokenization, and loss-mask generation all happen off the trainer's critical path, and only the artifacts the trainer actually consumes ever cross the boundary into the training loop. We pair this with a blended multi-dataset sampler designed for resumable streaming over very large mixes, and an I/O layer tuned for the realities of fetching multimodal data from object storage.The result: on large-scale VLM SFT runs, the trainer went from spending most of each step blocked on data to spending most of it training, a major improvement in useful GPU time. We'll share the architecture at a conceptual level, the gotchas at million-datapoint scale, and a mental model engineers can take home for the data side of any perception-agent stack.

4:30pm

KEYNOTE **Autoresearch · Main Stage** 4:30pm-4:50pm

Closing Keynote

Addy Osmani — Director of Engineering, Independent

TBD

4:50pm

KEYNOTE **Autoresearch · Main Stage** 4:50pm-5:10pm

Trends in AI

George Cameron — Co-Founder, Artificial Analysis · Micah Hill-Smith — CEO, Artificial Analysis

5:10pm

KEYNOTE **Autoresearch · Main Stage** 5:10pm-5:30pm

Closing Keynote

Wei-Lin Chiang — Co-founder & CTO, Arena

Day 4 — Session Day 3 Thursday, July 2, 2026

contents ↑

8:30am

SESSION **Startup Battlefield · Networking Room** 8:30am-10:30am

Battlefield setup

9:00am

KEYNOTE **Harness Engineering · Main Stage** 9:00am-9:20am

The 2026 State of AI Engineering

Barr Yaron — Partner, Amplify Partners

results per Barr

9:20am

KEYNOTE **Software Factories · Main Stage** 9:20am-9:40am

TCP and RDMA are Killing Inference Throughput; Homa can Fix It

John Ousterhout — Professor Emeritus, Stanford University

Modern AI inferencing is shifting from monolithic requests to complex agentic workflows and disaggregated KV stores. As a result, AI network traffic is no longer just very large transfers; tiny metadata requests are becoming more and more common, and their latency has a critical impact on throughput. Unfortunately, legacy transport protocols such as TCP and RDMA perform poorly on these workloads due to poor congestion control and head-of-line blocking. This talk will discuss the problems with TCP and RDMA and provide a brief introduction to the Homa transport protocol. Homa uses receiver-driven flow control and capitalizes on priority queues in network switches to reduce short-message latency by 10x for workloads like those in AI datacenters.

9:40am

KEYNOTE **Harness Engineering · Main Stage** 9:40am-10:00am

The Unreasonable Effectiveness of Separating the Task from the Model

Maxime Rivest — Core Contributor, DSPy · Isaac Miller — Lead Maintainer of DSPy; Co-Founder, cmpnd

By declaring your task's inputs and outputs without initially considering model capability, you create the space needed to figure out the model execution later. DSPy's entire promise is that you should evaluate and execute your AI engineering at a level higher than a specific prompt template or a particular provider's API shape: the Signature. However, models have evolved significantly over the last few years. How can the same input and output specifications still work in a world now filled with tools, RLMs, and Skills? By defining your task strictly through its inputs and outputs, the underlying implementation becomes completely flexible. You can experiment with different models, settings, weights, templating strategies, and output formats, all without touching your actual AI workflow. Consequently, you can leverage components built by others and focus entirely on your core AI task. In this talk we will present how dspy 3.5 makes it easier much easier. DSPy has its roots in prompt optimization, where we build efficient ways to conduct search and learning beneath the signature. In this talk we will give a preview of DSPy 4.0 where we use the fact that models have now passed a tipping point for two critical concepts we have always needed. First, we no longer need to limit the search space to a single instruction block per LLM call; models can now reliably write the code underneath a signature themselves—so they should. Second, traditional prompt optimization has always required a scalar metric, which is notoriously one of the hardest parts to get right. What if a DSPy program could learn directly from your interactions with users? Ultimately, all you care about is that the function you call respects the inputs and outputs of your signature. You can let the models figure out the rest.

10:00am

KEYNOTE **Harness Engineering · Main Stage** 10:00am-10:20am

How Anthropic Builds: Lessons from Labs

Mike Krieger — Head of Labs, Anthropic · swyx — Curator, Latent Space / AI Engineer

10:20am

KEYNOTE **Graphs · Main Stage** 10:20am-10:30am

Thinner Agents on a Smarter Substrate: The Ontology-based Semantic Layer

Emil Eifrem — CEO, Neo4j

10:45am

SESSION **Harness Engineering · Main Stage** 10:45am-11:05am

Tokens Should Have Jobs

Katelyn Lesse — Head of Engineering, Claude Platform, Anthropic · Angela Jiang — Head of Product, Claude Platform, Anthropic

SESSION **Generative Media · Track 1** 10:45am-11:05am

Training Krea 2 - What matters in generative model training.

Sangwu Lee — AI Lead, Krea.ai

Learn how Krea trained its first image foundation model from scratch. I will discuss 1. Our training and data pipelines 2. What are the most important aspects of improving model performance 3. How we intend to train the next generation of image generation models. Check out our technical report for details: https://www.krea.ai/blog/krea-2-technical-report

SPONSOR **Agentic Commerce · Track 2** 10:45am-11:05am

Designing Multimodal Collaborative Agents for Next-Gen Commerce

Nidhi Kaushik Vyas — Product, Google DeepMind

Today's commerce agents wait to be told what to look for. But most users live by a different rule: "I don't know what I want — I'll know it when I see it". If agentic commerce is ever going to cross the chasm, these systems need to stop waiting and start co-shopping. The future of commerce belongs to agentic collaborators that offer a white-glove, personal shopper experience - entirely absorbing the cognitive burden of product discovery, deep research, and validation. Rather than requiring shoppers to input exact search terms or define clear objectives, modern shopping systems will seamlessly guide them from a rough idea to the ideal product. By leveraging multimodal capabilities, these assistants can interpret abstract aesthetic "vibes" to understand user preferences, generate visual references to clarify questions, and enable a highly immersive try-before-you-buy experience to validate products, keeping the user aligned and visually grounded throughout the process. This talk will explore how advanced systems like Gemini work alongside users to clarify their preferences during the discovery process, co-navigate fluidly generated product categories, leverage individual context to filter choices, and produce interactive side-by-side comparisons tailored to the buyer's key priorities. The session will also cover robust auto-rater frameworks and how to design evals for high-agency execution. Attendees building conversational agents, managing complex product data graphs, or creating next-generation multimodal agentic interfaces will gain practical frameworks and insights to deliver highly personalized experiences at scale.

SESSION **AI in Finance · Track 3** 10:45am-11:05am

ALPHALAB: Autonomous Multi-Agent Research Across Optimization Domains with Frontier LLMs

Brendan Rappazzo — Machine Learning Scientist, Morgan Stanley

We built AlphaLab to automate quantitative research at Morgan Stanley's Machine Learning Research Lab - the experimental grind of architecture search, hyperparameter tuning, and literature review that consumes most of a researcher's time. To show it generalizes, we ran it on three deliberately different domains: CUDA kernel optimization (4.4× mean speedup over torch.compile, 91× peak), LLM pretraining (22% lower validation loss under a 20-minute budget), and traffic forecasting (23–25% RMSE improvement after the system independently found and tuned TFT and iTransformer from the literature). AlphaLab is an agentic harness that takes a dataset and a natural-language objective and runs a full research campaign across three phases: it explores the data and surveys prior work, it constructs and adversarially validates its own evaluation framework, and then it runs experiments at scale on a multi-GPU cluster via a Strategist/Worker loop with a persistent playbook that accumulates domain knowledge across experiments. In Phase 3 - the dispatcher keeps a large cluster fully utilized indefinitely with no human in the loop, and the playbook ends up containing domain-specific methodology that didn't exist anywhere in the prompts at launch. This talk walks through the three phases, what we learned from running campaigns with different models, what we have learned from using this in real systems, and future areas we are exploring.

SESSION **Local AI · Track 4** 10:45am-11:05am

State of the Union: Why Local, Why Now

Nader Khalil — Director of Developer Technology, NVIDIA · **Joseph Nelson** — Cofounder, CEO, Roboflow · **Alex Cheema** — CEO, EXO Labs · **Ahmad Osman** — Founder & CEO, Osmantic · **Matthew Berman** — Founder, Forward Future

Local AI has crossed from interesting to useful, driven by stronger open models, better hardware, and a maturing ecosystem for running intelligence outside the cloud. This panel explores what that shift unlocks for sovereignty, defense, regulated industries, privacy, cost, and resilience, and why open-source AI may be central to who benefits from the next wave of intelligence. Moderator: Nader Khalil (NVIDIA). Panelists: Joseph Nelson (Roboflow), Alex Cheema (Exo Labs), Ahmad Osman (r/LocalLLaMA).

SPONSOR Graphs · Track 5 10:45am-11:05am

CrabRAG: Why Automated Assistants Need Graph Memory, Not More Tokens

Stephen Chin — VP of Developer Relations, Neo4j

Autonomous assistants are easy to demo and hard to make reliable. The problem is usually not tool access. It is memory. Most assistant architectures still treat memory as a chat log plus vector retrieval. That is fine for document question answering, but it breaks down when the assistant must connect conversations, people, tools, and decisions across multiple tool iterations. For an AI engineer, a single request can depend on a Slack thread, a GitHub PR, a failed CI run, a calendar event, and prior operating preferences or constraints. These are not isolated pieces of context. They form a connected state that changes as work progresses and context grows. In this talk, I'll show why knowledge graphs, context graphs, and GraphRAG provide a better foundation for OpenClaw-style assistants. Knowledge graphs capture durable entities and relationships. Context graphs capture the operational layer assistants usually lose, including actions, decision traces, provenance, and recency. GraphRAG turns that structure into task-time context by combining graph traversal, semantic retrieval, and tool use. Attendees will leave with practical patterns for schema design, retrieval routing, and evaluation, plus a concrete blueprint for assistants that remember more than the last prompt and retrieve more than the nearest chunk.

SESSION AI in GTM · Track 6 10:45am-11:05am

GTM Engineering: The Technical Bits

Everett Berry — Head of GTM Engineering, Clay

Everyone talks about "GTM engineering" — Everett Berry shows the actual plumbing. As Head of GTM Engineering at Clay, he goes under the hood on the technical bits most talks skip: enrichment pipelines, agent-driven data classification, identity resolution, and the systems that turn unstructured web data into clean, deterministic CRM fields. A builder's-eye view of what GTM engineering actually is once you strip away the buzzwords.

SESSION AI in Healthcare · Track 7 10:45am-11:05am

From Ambient Documentation to Clinical Intelligence

Chaitanya Asawa — Head of Engineering for Clinical Decision Support, Abridge

A practical session on how healthcare AI moves beyond ambient note generation into context-aware clinical decision support. The talk would cover grounding outputs in the patient encounter, surfacing evidence with citations inside clinician workflows, preserving clinician agency, and building rigorous evals for safety and trust in live healthcare environments.

SESSION Agentic Engineering · Track 8 10:45am-11:05am

DeepSWE: expert code datasets

Serena Ge — CEO, Datacurve

DeepSWE and the data/eval layer behind coding agents; why curated expert code datasets matter for reliable agent performance.

SESSION Inference · Track 9 10:45am-11:05am

Operating Distributed Inference Systems at Scale

Nishant Gupta — Software Engineer, Tech Lead, Meta · Naman Ahuja — Senior Software Engineer, Meta

Inference has rapidly become one of the most important infrastructure problems in modern computing. As AI systems evolve into autonomous agents with persistent memory, tool usage, and multi-step reasoning, traditional inference architectures struggle under growing demands for latency, throughput, cost efficiency, and reliability. In this talk, I'll share lessons from building large-scale elastic compute and AI infrastructure systems powering production workloads. We'll explore the modern inference stack and the architectural patterns emerging to support next-generation agentic AI systems. Topics include distributed inference architectures for large-scale AI systems, GPU scheduling and elastic compute for inference workloads, multi-tenant inference infrastructure, caching, batching, latency optimization strategies, reliability and fault isolation for inference systems, observability and control loops for AI serving platforms, balancing cost, throughput, and user experience, and why inference is becoming an infrastructure orchestration problem. Attendees will gain practical insights into designing scalable, resilient, and cost-efficient inference platforms for modern AI workloads.

SPONSOR Track M 10:45am-11:05am

Diagnosing agent failures in production

Pamela Fox — Principal Cloud Advocate, Microsoft

Agent behavior changes in production. Learn common failure modes and how to debug, test, and improve performance using real evaluation techniques.

SESSION **Agentic Commerce · Leadership 1** 10:45am-11:05am

Building safe payment infrastructure for machine-to-machine commerce

Jennifer Lee — Product Lead, Machine Payments & Agentic Commerce, Stripe

Agents are a new class of buyer, but the infrastructure for them to transact headlessly barely exists yet. This talk walks through what it actually takes to make a machine payment work: how an agent discovers what services exist, how HTTP 402 lets a server return a payment challenge the agent can settle without a human in the loop, and how the seller gets a receipt they can trust. Whether you are building an agent framework or adding machine payments to an API or MCP server, you will leave with concrete patterns for the headless commerce stack.

SESSION **AI Architects: AI Factories · Leadership 2** 10:45am-11:05am

The Agent Behind the Curtain: Building the Oz Cloud Agent Platform

Safia Abdalla — Software Engineer, Warp

At Warp, we're building Oz to be the platform that enables people to be creative and build with cloud agents. That sounds simple, but only because the job of good developer tooling is to take on complexity before it reaches the user. The best tools fit into the way developers already think, then make accessible work that used to feel out of reach. This talk is about the engineering philosophy behind that work: how Warp's evolution from terminal to local agent to Oz shaped the way we think about building for developers. The focus is not on inventing brand-new abstractions for their own sake, but on making a messy stack of real engineering concerns feel coherent: where agents run, how they delegate, how teams control their environments, how humans can see what happened, and how the platform leaves room for people to build things they couldn't even imagine before. 4:04 PM

SESSION **Expo Stage 1 · Expo Stage 1 NE** 10:45am-11:05am

AI Engineering & Governance 2026 Trends

Wallon Walusayi — Qodo

AI Engineering & Governance 2026 Trends

SESSION **Expo Stage 2 NW** 10:45am-11:05am

Your Agent Can't Tell If It's Right

Willem Pienaar — Co-founder and CTO, Cleric

Coding agents feel reliable because of one signal you never think about: the tests. They catch confident mistakes in seconds, so you never see most of them. The real world has no test suite. Put an agent in production and that signal is gone, and a wrong answer looks the same as a right one. So how do you know it's right? We watched our agent look at an 80% drop in throughput and report zero user impact, because a similar alert the month before had been noise. The data to catch it was already in front of it. There is no single verifier, but there are several weaker signals. While the agent reasons: grounding each claim against live data, and looking for evidence that distinguishes competing hypotheses. Before it acts: calibrated confidence, and a separate critic. After it acts: whether the fix held, whether the alert returned, whether an engineer redid the work. None is conclusive on its own. Combined, they estimate whether the agent was right. The talk covers where these signals come from, how we combine them, and how often they still disagree.

SESSION **Expo Stage 3 · Expo Stage 3 SW** 10:45am-11:05am

No, That's Not a Software Factory

Ryan Cooke — WorkOS

Drop an agent in a sandbox, point it at your repo, watch it ship code. Whether you're buying from a vendor or building it yourself, everyone is following the same playbook. But a sandbox isn't a software factory. At WorkOS, we built Project Horizon, and it taught us that infrastructure is only the first challenge. The unlock is encoding how your org actually builds software: the way work gets planned, scoped, and verified, the conventions and judgment calls that define your engineering culture. Our product engineering process served as the blueprint for every agent workflow we built in Horizon.

SESSION **Expo Stage 4 SE** 10:45am-11:05am

Vector Isn't Enough: Hybrid Search & Retrieval for AI Engineers

SESSION **Startup Battlefield · Networking Room** 10:45am-1:00pm

$100,000 AIE Startup Battlefield — presented by Hyperagent

Howie Liu — CEO, Airtable

Twenty AI startup founders take the floor at Moscone West for a science fair-style demo gauntlet (10:45 AM–1 PM), where judges and attendees circulate freely and ask the tough questions. Three finalists earn a spot on the main stage at 5:25 PM to pitch live in front of our full audience — and walk away with a share of $100,000 in prizes ($50K / $30K / $20K), awarded by Howie Liu and the Hyperagent team! Builders come from across the AI application stack — from agentic workflows and AI sports tech to social tools and data infrastructure — all competing for the judges' vote and your vote for People's Choice! Don't miss Howie Liu on stage at 5:10 PM, followed immediately by the finals and the winner announcement. 🏆

11:00am

SESSION **CTO Circle · Leadership Lounge** 11:00am-12:00pm

The Agentic Product Development Organization

Martin Harrysson — Senior Partner, McKinsey & Company · Matt Linderman — Partner, Technology Practice, McKinsey & Company · Prakhar Dixit — Partner, McKinsey

Facilitated, peer-to-peer, under the Chatham House Rule — not recorded. As AI agents become embedded in day-to-day work, organizations will need to rethink product development teams, roles, and skills. This foundational shift reshapes management layers and requires overcoming challenges across talent attraction, development, and retention.

11:10am

SESSION **Agentic Engineering · Main Stage** 11:10am-11:30am

MCPs, CLIs, and Skills: Choosing the Right Tooling Layer for Agentic Development

Nikita Kothari — Senior Member of Technical Staff, Salesforce

Agentic development needs more than one interface: MCPs provide clean, portable connectors to services, with built-in patterns for security and auth. CLIs offer composability, debuggability, and workflows developers already trust. Skills teach agents how to use a wide variety of tools and MCPs effectively without overloading context.

SESSION **Generative Media · Track 1** 11:10am-11:30am

HTML Is All Agents Need

James Russo — Software Engineer, HeyGen

LLMs are great at writing code. So the question we kept asking was: can they write code that produces a video? We thought it would be easy. The reality was a year of trying. We started with massive prompts to get very mediocre output. We made it more agentic to iterate and improve its output. This worked okay but wasn't production-ready. Eventually we tried Remotion. It got us deterministic video, but the React framework kept boxing the agent in. The more guardrails we added, the safer and more boring the outputs got. When we utilized plain HTML, CSS, and JavaScript, the creativity came back to the output. So we set out to build a video rendering framework on top of HTML. But it needed to work with Gemini Flash. Why? Because one tell that a framework is fighting an agent is needing the biggest model just to get usable output. So from there we shaped the framework around what small models could reliably author. That left one real engineering question: can we keep the freedom of HTML and still render a deterministic MP4? Browsers don't want to do that. Image decoders, font loaders, and animation clocks all run async on their own schedule. Great for performance. Terrible for "render the same pixels every time." Throughout, we iterated constantly with agentic loops and self-improving evals to test out the framework, find issues in our renderer, and shape a set of skills that gave the agents Taste instead of guardrails. This talk is what it took to get there.

SPONSOR **Agentic Commerce · Track 2** 11:10am-11:30am

Why Your AI Agent Needs a Wallet: Agentic commerce on Arc with USDC and Nanopayments

Harshal Bhangale — Staff Software Engineer, Circle

AI agents can reason, plan, call tools, and write code. But the moment one needs paid data, an API call, or another agent's service, it hits a human wall: accounts, API keys, credit cards, checkout flows. It stalls and asks you to step in. It can't pay. We'll run the same real task through two agents, one without a wallet and one with. The first stalls. The second, handed a Circle agent wallet through the Circle CLI, discovers services, pays per request over x402 in USDC, and finishes on its own, inside spending limits you set. The next leap in agents isn't only better models or more tools. It's economic agency: holding programmable money and transacting at machine speed. We'll show how it works on Arc, where USDC is the gas, finality is sub-second, and gasless nanopayments settle in batches through Circle Gateway, so paying a fraction of a cent per request is actually practical.

SESSION AI in Finance · Track 3 11:10am-11:30am

Why Off-the-Shelf AI Doesn't Understand Money

Udi Menkes — Principal Product Manager, Intuit

Ask any LLM a financial question about your business. You'll get a fluent, confident, generic answer — one that doesn't truly know your business, or what happened when businesses like yours made that same decision. We build financial AI at Intuit serving 100M+ customers. Our custom LLMs outperform general-purpose models on accuracy while cutting latency in half. But that's the foundation, not the destination. I'll cover where financial intelligence goes when AI stops reporting what happened and starts helping you decide what to do next (and does it for you).

SESSION Local AI · Track 4 11:10am-11:30am

State of the Union: Why Local, Why Now

Nader Khalil — Director of Developer Technology, NVIDIA · Joseph Nelson — Cofounder, CEO, Roboflow · Alex Cheema — CEO, EXO Labs · Ahmad Osman — Founder & CEO, Osmantic · Matthew Berman — Founder, Forward Future

Local AI has crossed from interesting to useful, driven by stronger open models, better hardware, and a maturing ecosystem for running intelligence outside the cloud. This panel explores what that shift unlocks for sovereignty, defense, regulated industries, privacy, cost, and resilience, and why open-source AI may be central to who benefits from the next wave of intelligence. Moderator: Nader Khalil (NVIDIA). Panelists: Joseph Nelson (Roboflow), Alex Cheema (Exo Labs), Ahmad Osman (r/LocalLLaMA).

SPONSOR Graphs · Track 5 11:10am-11:30am

Active Graph Agent Runtime (BabyAGI 4)

Yohei Nakajima — Managing Partner, Untapped Capital

Proposing a novel event-sourced graph runtime for building long-running auditable, agentic systems. Built on top of and combining various BabyAGI iterations and graph experiments (memory, code, logs) into a single primitive.

SESSION AI in GTM · Track 6 11:10am-11:30am

Reverse-Engineering the AI Buyer

Aliisa Rosenthal — General Partner, Acrew Capital

You Built the Best AI Product in the Room. Now What? The GTM Lessons Builders Skip. Aliisa decodes the commercial mistakes technical teams make most often: why enterprise procurement isn't like consumer adoption, how to design for trust and change management from day one, the difference between a pilot and a deal, and the signals that tell you a product is ready to scale vs. ready to get stuck. She's packed with war stories and counterintuitive lessons from the trenches of OpenAI.

SESSION AI in Healthcare · Track 7 11:10am-11:30am

Guardrails First: Engineering Member-Facing Health AI

Rashi Agrawal — Head of Agentic AI, Hinge Health

Everywhere else in the company, an AI pilot can reach production in weeks. For our member-facing clinical assistant, it can't, and that single constraint redesigned our entire architecture. This is a field report on building conversational AI in a regulated digital health setting, where "move fast and break things" isn't a culture choice. It's a liability. We'll get concrete about what changes when every output has to be clinically safe, auditable, and compliant: PHI is protected by architecture, not policy. Production and non-production are hard-isolated, dashboards are sanitized, and engineers outside the US never touch protected health information. Must-not-fail behavior never lives in a prompt. Emergency escalation and intent routing run as deterministic rules at the top of every conversation turn, before the model is consulted. If you can't afford to get something wrong, you don't leave it to a probabilistic system. Clinical safety is a continuous eval layer. ~30 LLM-as-judge evaluators score clinical accuracy, clinical safety, escalation routing, and recommendation relevance, continuously, not once. Every output is auditable. Each turn, tool call, and reasoning step is traced so outputs can be reviewed and meet regulated reporting obligations. The throughline: in regulated healthcare, compliance constraints aren't a tax you pay around the architecture. They become the architecture. We'll talk about why guardrails-first is the only way to ship member-facing health AI, and why "painfully slow" is sometimes exactly right. (This is non-diagnostic, member-facing AI. The talk is about engineering discipline under regulation, not medical claims.) Key takeaways - In regulated health AI, "move fast" is the wrong default. Design for deliberate, careful launches. - Must-not-fail behaviors belong in deterministic rules at the top of every turn, never in the prompt. - Protect PHI through architecture: isolate prod from non-prod, sanitize dashboards, restrict access by role and geography. - Make every output auditable. Trace each turn, tool call, and reasoning step so safety is reviewable, not assumed. - Treat clinical safety as a continuous LLM-as-judge layer, not a one-time gate.

SESSION Agentic Engineering · Track 8 11:10am-11:30am

Anthropic's CCA Exam as a Field-Guide for Agentic Engineering

Frank Coyle — Lecturer, UCALBerkeley / Founder AI/Edge, UCAL Berkeley

Anthropic's CCA Exam: A Field-Guide for Agentic Engineering The Claude Certified Architect (CCA) exam distills what Anthropic has learned from working with the AI companies shipping agents to production — the patterns that work, the anti-patterns that quietly burn tokens and trust, and the architectural decisions that separate demos from systems you'd stake a quarter on. This talk treats the exam as a field guide for agentic engineering, whether or not you ever sit for it. We'll walk through the five competency domains the exam tests — Agentic Architecture, Tool Design and MCP Integration, Claude Code, Prompt Engineering, and Context Management — with particular emphasis on multi-agent orchestration, subagent delegation, tool schema design, and lifecycle hooks. We'll then work through the six real-world scenarios the exam uses to probe judgment, each organized around an anti-pattern: the seductive-but-wrong move that looks reasonable until it costs you a production incident. Attendees leave with a working mental model of the agentic surface area and a checklist of the failure modes that matter most when moving from prototype to production. **Who should attend:** engineers and architects building agentic systems with Claude or other frontier models, technical leads evaluating agent designs, and developers considering the CCA credential.

SESSION Inference · Track 9 11:10am-11:30am

Routing LLM Inference in Production: From Engine Signals to Policy

Qianru Lao — Member of Technical Staff, OpenAI · Lu Zhang — Member of Technical Staff, OpenAI

Production LLM apps need more than a fast model: they need an inference routing layer that can choose where each request should run as engines, capacity, latency, and geography cost change. This talk shares a generalized Inference Load Balancer (ILB) proxy/controller architecture. A low-latency proxy applies routing weights and request-path signals, while a controller computes source-cluster-to-engine weights from demand, capacity/performance profiles, replica state, and geography cost. We will cover the practical debugging patterns AI engineers need: reading engine signals, explaining why a request went to one backend instead of another, handling retries and load shedding, and keeping routing behavior observable without exposing OpenAI-specific internals or non-public metrics.

SPONSOR Track M 11:10am-11:30am

Tracing and debugging agents across systems with OpenTelemetry

Chang Liu — Senior Product Manager, Microsoft

Understand what your agents are doing. Learn how to trace workflows across systems, debug issues, and uncover optimization opportunities using OpenTelemetry.

SESSION AI-Native Enterprises · Leadership 1 11:10am-11:30am

Tribal Dungeons of Global Shipping: AI Agents at Global Scale

Dmitry Buykin — Applied AI Lead, Staff Software Engineer, Maersk

Most “AI agents in production” talks skip the part where you have to turn distributed operational knowledge into something an agent can execute safely. This is that part: a practitioner report from a global logistics case-processing project at Maersk, focused on SOPs-as-code, evaluation UX, guardrails, replay-based testing, and SME refinement loops. The talk covers why versioned, country-aware SOPs beat prompt engineering at scale; how SME corrections become safe workflow changes; why classifier routing and SOP execution must stay separate; where agents under-deliver against demos; and why most of the engineering effort goes into evaluation, replay, and guardrails rather than model prompting.

SESSION AI Architects: AI Factories · Leadership 2 11:10am-11:30am

FinOps for AI Agents: Who Spent All the Tokens?

Tisha Chawla — Software Engineer, Microsoft · Susheem Koul — Senior Software Engineer, Microsoft

When an autonomous agent finishes a task successfully but costs ten times more than it did the previous day, traditional application monitoring fails. A recursive tool loop that retries silently, an oversized context window that quietly expands, or an unflagged model upgrade can burn through an entire budget long before a human notices. The execution appears successful on functional dashboards, meaning the only clear signal of failure is the cloud invoice at the end of the month. As AI systems move into production, tokens have become a primary operational resource alongside CPU, memory, and storage, yet few teams manage them with equivalent systems rigor. Most architectures lack the granular visibility required to attribute token spend to specific users, agents, or workflows, and they lack mechanisms to terminate a runaway loop before it triggers a financial incident. This session treats token consumption as a first class systems problem, demonstrating how to make it observable, attributable, and enforceable across complex agent workflows. The presentation covers practical engineering patterns for instrumenting token usage at every model call and tool invocation, attributing costs down to specific users or business operations, surfacing expensive execution paths, and enforcing runtime budgets, quotas, and circuit breakers to halt runaway behavior in real time. Attendees will leave with a practical framework for governing agent spend deliberately, transforming tokens into a managed operational resource rather than a surprise line item on the cloud bill.

SESSION Expo Stage 1 NE 11:10am-11:30am

Beyond RAG: See a relational context engine reduce token burn

Peter Werry — Founding Engineer, Unblocked

In this expo talk we'll give you a free context engine simulator, open source tools, and demo how a context engine works. See how modern engineering workflows with agentic loops and goals produce better quality code and reduce token burn. RAG, while useful, leaves context gaps for humans and agents. A context engine fills those gaps by including real-time, relational, personalized, and permission aware techniques to get high-signal context to humans and agents at runtime.

SESSION Expo Stage 2 NW 11:10am-11:30am

ARIA, how we built autoresearch with autoresearch

Zubin Aysola — Senior Software Engineer Weave, Weights & Biases by CoreWeave

ARIA is an end-to-end auto research and AI research product that improves models, launches training jobs, and agents alike. We used ARIA along with a sophisticated evaluation framework we're calling the WBAF, Weights and Biases Agent Factory, to build itself. ARIA reads its own production traces, improves its own prompts, tools, skills, and other effects to solve customer challenges. In this talk, we dive into the evaluation framework, how we built a sophisticated reinforcement learning style environment over the Weights & Biases product, and how we scaled from zero to one to a full team working in parallel on improving an agent.

SESSION Expo Stage 3 SW 11:10am-11:30am

The Lethal Trifecta Is Already on Your Developers' Laptops

Michael Patterson — Coder

The lethal trifecta: an AI agent with access to private data, exposure to untrusted content, and the ability to communicate externally. Combine all three and an attacker can trick your agent into exfiltrating anything it can see and there is no prompt-level fix.. Most enterprises have already deployed this pattern at scale: Claude Code, Cursor, and Copilot on developer laptops with local credentials, MCPs reaching into internal systems, and open egress. I'll speak to my own personal agent stack as a textbook example, then trace the same shape across enterprise deployments I see at Coder. The back half is four architectural moves that defuse it: governed compute, centralized credentials, default-deny egress, identity-bound audit. Walk out with a mental model and a checklist you can run against your own deployment the next morning.

SESSION Expo Stage 4 SE 11:10am-11:30am

Your AI Agent Has No Nervous System

Matt Gibiec — Regional Director, Solutions Engineering, Dynatrace

Most agents ship with solid evals and zero runtime observability. When something breaks in production — wrong answer, runaway retry loop, or silent tool failure — you're debugging blind. You can see the output, but you can't see what the agent believed when it made the decision. This talk walks through how to instrument agentic pipelines with OpenTelemetry: capturing system context at every step, making prompt state and tool call outcomes visible as structured data, and governing token consumption as SLOs instead of discovering overruns on an invoice. Attendees will leave with three takeaways: an understanding of telemetry for multi-step agentic workflows, a pattern for capturing system context at the span level so teams know exactly what the agent saw before it acted, and a framework for visibility into token budget and behavioral drift before something goes sideways in production. Telemetry is the nervous system. System context is the memory. Token budgets are the vital signs. None of it is optional.

11:40am

SESSION Agentic Engineering · Main Stage 11:40am-12:00pm

Auth for Agents: Unblock Autonomous AI with auth.md

Michael Grinich — Founder & CEO, WorkOS

AI agents are ready to act on users' behalf, but legacy auth flows were built for humans, not agents. This session introduces auth.md, an open protocol that lets agents register and authenticate users without sign-up forms, and shares what early implementers have learned since launch. Learn about the new protocol that Cloudflare, Firecrawl, Cogny, and monday.com are adopting to power agent registration — authenticating agents without sign-up forms.

SESSION Generative Media · Track 1 11:40am-12:00pm

Building an Agentic Video Editor for Mass Consumer

Ekaterina Deyneka — Founder & CEO, Reelful

Most agentic systems today are built for developers — people comfortable setting up environment, configs, and debugging agent loops. But what happens when your user has never heard the word "agent" and just wants a video ready to post? Reelful is an agentic video editor that lives right in the user's phone. It turns raw photos and videos from your camera roll into polished, short videos. No setup. No sophisticated prompting. No empty timeline. Under the hood, the agent orchestrates multiple models and composes a video together. In this talk, I'll walk through: * The agentic pipeline architecture: how we chain models across modalities (vision → language → speech → video), handle context passing between steps, and manage state across a multi-minute generation job * The UX inversion: how we designed the agent to require minimal effort from user — the system infers intent from the media itself, making complex orchestration invisible This talk is for anyone building agents that need to work for non-technical users, or anyone curious about multimodal agentic pipelines beyond text and code.

SPONSOR Agentic Commerce · Track 2 11:40am-12:00pm

When AI Agents Pay and Sellers Monetize: Building x402 Apps for Agentic Commerce on AWS

Anil Nadiminti — Sr Solutions Architect, Amazon Web Services (AWS)

As Agentic AI moves from chat to execution, autonomous agents need a native way to discover, access, and pay for digital services in real time. This session explores how x402 can turn HTTP into a payment-aware interface for machine-to-machine commerce, unlocking crypto-native patterns like programmable access, pay-per-use APIs, and on-demand monetization for data, tools, and services. We'll show how to build x402-enabled applications and walk through the architecture, the full agentic payments flow, seller monetization strategies, payment verification, and design tradeoffs involved in making agent-driven transactions secure, scalable, and production-ready. Attendees will leave with practical patterns for building apps where AI agents do not just call APIs — they can discover services, evaluate costs, transact autonomously, and enable new revenue models for sellers.

SESSION AI in Finance · Track 3 11:40am-12:00pm

Let's integrate AI Agents in Event-Sourced Systems

Divakar Kumar — Technical Architect, FlyersSoft

Fraud detection has always been a race against time. In traditional event-sourced systems, every transaction, login, or transfer is captured as a sequence of immutable events. These events tell a clear story — but only after the fact. What if events could do more than just record history? What if they could talk back? In this talk, we'll explore how agentic event-driven systems transform fraud detection. Imagine every PaymentInitiated, LoginAttempt, or DeviceChanged event not just being logged, but immediately consumed by an autonomous Fraud Detection Agent. This agent correlates events across accounts, reasons over historical event streams, and generates new events like SuspiciousActivityFlagged or TransactionHeldForReview. Through a real-world inspired use case in banking and digital payments, we'll show: - How event sourcing provides the perfect memory layer for fraud detection agents - Patterns for agents to safely inject new domain events without violating invariants - How to avoid runaway feedback loops when multiple agents interact (e.g., fraud + compliance + customer service agents) - Governance, auditing, and explainability challenges when autonomous agents take part in mission-critical workflows By the end of this session, you'll see how event-driven DDD systems evolve when agents stop being passive consumers and start actively shaping the event stream — turning fraud detection from a reactive process into a proactive, adaptive defense.

SESSION Local AI · Track 4 11:40am-12:00pm

Demo: GLM 5.2 on DGX Station — Frontier Intelligence Under Your Desk

Ahmad Osman — Founder & CEO, Osmantic

Ahmad Osman shows off the power of local AI on stage, running frontier open models on a DGX Station.

SPONSOR Graphs · Track 5 11:40am-12:00pm

Your Moat Is Your Data Model

Mike Phipps — Lead AI Engineer, Gates Foundation

Every enterprise AI team faces the same strategic question: where in the stack should a small team focus its effort? Models, frontends, and agent frameworks evolve rapidly and are increasingly commoditized. But regardless of how these layers mature, AI in enterprise settings remains bottlenecked by the same underlying problem: structured data is siloed across systems of record with domain-specific schemas, and the unstructured data needed to contextualize it sits in entirely separate systems, with its own systematic complexities. The durable work is cleaning, curating, and semantically modeling this data in an AI-first manner so that any client — chat, workflow, or otherwise — can query across it. That's the moat. At the Gates Foundation, my team built and deployed our foundation-wide knowledge graph on Neo4j that unifies structured and unstructured data behind a single MCP server. The graph itself is modeled for agentic consumption: natural hierarchies are projected as traversable paths rather than flattened tables, and unstructured documents are semantically chunked, tagged, and mapped to structured entities at ingestion time using AI-driven ETL. The result is a semantic layer where an agent can express a complex cross-system question as a concise graph query and receive an accurate answer. This talk is an architectural walkthrough covering the end-to-end pipeline: AI-based extraction and semantic chunking of unstructured documents, the agent-first data modeling decisions, design considerations for our MCP server, and how we handle graph-based retrieval evals. We'll walk through real query sessions showing Claude interacting with the graph through both chat and workflow integrations. The intended takeaway is a practical framework for where a small enterprise team's investment compounds — and why that investment is the data model, not the layers above it.

SESSION AI in GTM · Track 6 11:40am-12:00pm

AI in GTM at Notion

Flora Liu — Software engineer, Notion

Notion's go-to-market runs on a system, not a roster of heroes. Flora Liu walks through the building blocks of human–AI collaboration behind Notion's GTM: the design principles that decide what AI owns and what stays human, the failures that taught them where that line belongs, and why the wins that matter most — faster delivery, real adoption — never show up on a revenue chart. An honest look at what actually works, from the team building it.

SESSION AI in Healthcare · Track 7 11:40am-12:00pm

Shipping AI to a Million Patients Without an A/B Test

Jared Joselowitz — AI Research Engineer, Ufonia

You can't A/B test on patients. You can't unsend a phone call. The model card won't save you at the post-incident review. Most AI eng playbooks assume the opposite. Ship to 5%, watch the dashboard, roll back if it goes wrong. None of it survives regulated deployment, which is now coming for fintech, legal, and government too. So the engineering has to move: into hazard analysis, simulated populations, asymmetric evaluation, and audit trails treated as the deliverable. The trail is the product. I'll show you what changes when rollback isn't an option. How Ufonia ships Dora, an AI voice agent now making clinical follow-up calls on the NHS and across US health systems, using a hazard-driven simulation rig (MATRIX) and a prompt-optimisation flywheel that surface failures and conform the same base system to each clinical niche, all of it pinned to an audit trail. And the cheap version of all this, for any team whose users can't be the test population.

SESSION Agentic Engineering · Track 8 11:40am-12:00pm

Guide, Verify, Solve: The Engineering Discipline Agentic Development Demands

Anirban Chatterjee — Director, Product and Solutions Marketing, Sonar

Agentic development is not a productivity story: it's a reliability engineering problem at a scale most teams have never faced. Long-running agent tasks fail at alarming rates, pull requests have grown from 50 lines to 5,000, and cognitive surrender is real—the more capable AI output appears, the less humans interrogate it, right at the moment the stakes are highest. Independent, peer-reviewed research from Carnegie Mellon studying 807 open source projects found that AI agent adoption caused a persistent 30% increase in code analysis warnings and a 41% increase in complexity — with long-term development velocity declining as a result. Agents don't just write code faster, they accumulate debt faster, too. The answer is not to slow agents down, it's to govern and refine the loop they operate inside. Sonar's Agent Centric Development Cycle (AC/DC), defines that loop across three continuous stages: guide agents with project-specific context and constraints before a single line is written; verify rigorously and continuously inside the loop, not downstream in CI; and solve issues automatically before they ever reach a manual review. The deeper insight is that this is not primarily a security story. It's an efficiency story. Codebases riddled with complexity make agents slower, less reliable, and significantly more expensive to run. Every token spent navigating legacy debt is a tax on every future agent run. Well-maintained, low-complexity codebases mean fewer failures, fewer tokens, and faster iteration. The teams that instrument this loop now will do more than ship safely: they'll compound their advantage every time an agent touches their codebase. Verification isn't a cost center. In an agentic world, it's a competitive moat.

SESSION Inference · Track 9 11:40am-12:00pm

Are LLM Performance Benchmarks Reliable?

Ashok Chandrasekar — Staff Software Engineer, Google · Jason Kramberger — Software Engineer, Google

Standardizing performance benchmarks for production-grade Large Language Models is currently a significant challenge across the industry. Conflicting data is prevalent, whether originating from server developers like vLLM and SGLang or from various analysts and competitive benchmarks, and these results often fail to hold up under real-world conditions. Our research into these inconsistencies identified several critical factors, including the constraints of single-process tools, specifically the Python Global Interpreter Lock (GIL) and the nuances of model-level settings like temperature. Furthermore, a lack of transparency regarding load generation parameters such as QPS and concurrency, paired with insufficient observability into the benchmarking clients themselves, contributes to these disparate outcomes. In this talk, we share key lessons learned from our benchmarking efforts, examining the primary pitfalls that distort performance data and offering strategies for mitigation. Additionally, we will introduce Inference Perf, an open-source, multi-process utility we developed to provide reliable stress-testing for production stacks. Our goal is to promote standardized, real-world benchmarking practices that allow the community to move beyond unreliable data. Join us to discover how to accurately measure, optimize, and report LLM performance with certainty.

SPONSOR Track M 11:40am-12:00pm

Benchmarking VS Code with VSC-Bench: How to measure agent performance

Ross Wollman

"Agent quality in VS Code depends on a stack of variables: model, version, prompts, extensions, MCP servers, and more. Each one affects quality, tokens, and latency—and they interact in ways that are hard to reason about. In this session, we'll show how to benchmark different configurations using VSC-Bench so you can compare results side by side and understand what actually works. Instead of guessing which setup is better, you'll learn how to measure tradeoffs and make data-driven decisions."

SESSION **Inference · Leadership 1** 11:40am-12:00pm

All the Things We Have to Do to Satisfy Your Insatiable Need for Tokens

Daniel Kim — Head of Growth, Cerebras · **Michelle Nguyen** — Co-Founder, Gimlet Labs

Every time the industry figures out how to serve tokens faster and cheaper, the appetite grows to match. Models get bigger, contexts get longer, agents start chaining thousands of calls together. The finish line keeps moving. This talk is a technical tour through everything the industry has done to keep up, led by two experts in high-performance inference. We'll start with the optimizations that made hardware work harder without changing the underlying architecture. Then we'll go up a level with techniques that work smarter across requests and across the model itself. And finally, a peek into the future with heterogeneous disaggregated inference, the architectural shift that splits prefill and decode across specialized hardware, and even more advanced forms of hardware specialization coming your way soon. Token demand is about to get a lot more insatiable. Let's see what the future has in store for us!

SESSION **AI Architects: AI Factories · Leadership 2** 11:40am-12:00pm

What If Your Chip Design Team Moved Like a Single Body?

Khaled Alashmouny — Founder & CEO, AIDAChip · **Abduallah Mohamed** — VP of AI/ML, AIDAChip

Most agentic demos you've seen has a hidden assumption: one user, one session, one task. But what happens when the agent needs to coordinate with 30 other agents, across 10 disciplines, on a project that takes 12 months — where a single miscommunication costs $10-50M? Chip design is that problem. Only 14% of chips succeed on first silicon. The bottleneck isn't individual engineer speed — it's silent divergence between disciplines working from specs that drift without noticing. We built a multiplayer AI on the Anthropic Agent SDK, connected through three alignment layers: a living spec graph (System of Intent) that propagates changes and detects conflicts in real time, a tribal knowledge layer (Memory) that compounds methodology across projects, and milestone-aware execution that drives EDA tools with full design context. Each agent operates within strict spec-hierarchy boundaries enforced at the API level. Cross-agent invocations use structured tool calls with typed parameters, logged for full auditability. We talked with 15 practitioners across 8 major semiconductor and EDA companies. The universal finding: teams need alignment infrastructure, not faster copilots. We'll also share what broke — because coordination tax applies to AI agents too, and the failure modes are surprisingly instructive. This talk covers the multi-agent architecture, evaluation methodology, and lessons from deploying agentic AI in one of engineering's most complex coordination domains.

SESSION **Expo Stage 1 NE** 11:40am-12:00pm

The Art of Building Verifiers for Computer Use Agents

Miguel González Fernández — Tech Lead, Browserbase · **Corby Rosset** — Senior Researcher, Microsoft Research

Every team building browser agents has the same problem: you can't trust your own evals. Browser tasks are too open-ended for deterministic checks, so teams use LLM verifiers as judges, and the judges are wrong constantly. WebVoyager misses 45% of failures. WebJudge misses 22%. Used as RL reward, you're not training a better agent, you're training a more confident liar. This talk walks through the Universal Verifier, open-sourced with Microsoft Research: false positive rate near zero, Cohen's κ matching human-human agreement. Four design principles, one open benchmark, and an honest account of where auto-research worked and where it plateaued.

SESSION **Expo Stage 2 NW** 11:40am-12:00pm

Seeing the Plumbing: Profiling vLLM Speculative Decoding on NVIDIA Blackwell

Sheilah Kirui — NVIDIA

Speculative decoding promises dramatic LLM speedups by using a tiny draft model to guess tokens ahead of a large target model. However, dual-model serving fundamentally rewrites your memory dynamics and introduces a rigid engineering trade-off: guess right, and you bypass the memory-bandwidth bottleneck; guess wrong, and you waste compute. This session is a live-demo routing identical workloads through baseline and speculative configurations in vLLM on a single NVIDIA RTX 6000 Blackwell GPU. Splitting the screen between a Streamlit app and a live Grafana dashboard, we will profile the inference engine across three vectors: Time per Output Token (TPOT): The real-time, user-facing latency delta. KV Cache & Memory Footprint: The exact VRAM tax of tracking parallel token states within a 96GB budget. Draft Acceptance Rate: Visualizing the tipping point where dropping acceptance rates cause speculative decoding to fall below baseline efficiency. Supporting Materials Project Repository: https://github.com/akamai-developers/speculative-decoding-example-vllm-blackwell# (Work In Progress / Active Development)

SESSION **Expo Stage 3 SW** 11:40am-12:00pm

Voice is the universal interface

Kwindla Kramer — CEO, Daily · Neil Zeghidour — Co-founder & CEO, Gradium

Language models give us the ability to create natural language, conversational, interfaces for computers. We are seeing a rapid shift among early adopters to using general language instead of traditional user interfaces for tasks like writing code and editing spreadsheets. Join the cofounders of Pipecat, Gradium, and Daily as we discuss the future of realtime voice and AI interfaces. Voice is the most efficient input mode for natural-language systems, and often the most efficient output mode, as well. But good voice interfaces require a very high degree of conversational facility, intelligence, task-specific reliability, and robustness to real-world realities like multiple speakers and background noise. There's a long history of voice interfaces in science fiction: Star Trek, Iron Man, Her. We'll use these depictions of computing possibilities as a jumping off point for talking about the ideal voice interface. How close are we to being able to build these interfaces with today's models, hardware, orchestration tooling, and UI libraries? What are the most promising research directions? What did the movies get wrong, now that we actually have experience building natural language, open-ended, voice systems?

SESSION **Expo Stage 4 SE** 11:40am-12:00pm

The Next Run Should Be Better

Jake Broekhuizen — Labs Lead, LangChain

Agents generate a constant stream of experience through traces: tool calls, failures, corrections, routing decisions, and user feedback. The challenge is identifying which parts of that experience are worth remembering, and making those lessons available to the agent when it runs again. This talk presents memory as an agent learning loop: capture traces, extract signal, and turn the right lessons into durable context. We'll explore practical models for agent memory and discuss how to build systems where the next run can be better than the last.

12:05pm

SESSION **Harness Engineering · Main Stage** 12:05pm-12:25pm

Harness Engineering: Building the Production Cage for Powerful Domain Agents

Mike Chambers — Senior Developer Advocate for Generative AI, Amazon Web Services (AWS)

Every agent is a while loop. The model takes strings in and produces strings out. We've all written it, debugged it, shipped it. And yet every team building agents is still re-inventing the same session management, truncation logic, tool wiring, and memory plumbing from scratch. The hard part is the harness: session isolation, context management, memory persistence, sandboxed execution, observability. The machinery that makes a model dependable in production. Most of the failures we see in deployed agents (context rot, premature completion, tool bloat) trace back to harness problems, not model problems. This talk covers what a harness actually does, why "harness engineering" suddenly showed up in engineering posts from everyone, and what changes when you stop building harnesses by hand. In live demos, we'll build the same agent three ways: hand-rolled Python, framework-generated, and fully managed through a single API call. Each level shifts the failure modes from infrastructure plumbing to engineering judgment, where the real questions are what context to preserve, when to verify, and how to keep an agent from finishing half the job and calling it done. The harness handles the machinery. You still have to engineer the behavior.

SESSION **Generative Media · Track 1** 12:05pm-12:25pm

The Next Game Engine Won't Have a Manual

Arturo Nunez — Founder, Nereu

Game development is still incredibly hard to get right. It requires great engineering, artistic vision, and the ability to make something genuinely entertaining, all at once. Dropping a powerful LLM into existing engines won't solve the problem. Game development needs to fundamentally change to work in this era of agents. After 15 years in games (making them, watching others make them, and working at the most popular game engine in the world) I'm now fully embracing the power of AI to give it to the people who dream of making games but find it too difficult. I'm building Veselka. In this talk, I'll show you the AI-magic that converts Claude into a real game dev partner, using Three.js to let anyone build their dream game.

SPONSOR Agentic Commerce · Track 2 12:05pm-12:25pm

x402 isn't good (yet)

Jan Curn — Founder & CEO, Apify

While everyone understands that agents will get more done with a budget, no one knows which protocol will win agentic payment standard wars: x402, MPP, Skyfire, or another? So far, x402 is the most mature protocol with the largest transaction volume, but even its new "upto" payment scheme doesn't support true usage-based pricing, as it gives agents a chance to consume resources and then skip out on the bill. I'll walk you through our experience (and pains) implementing agentic payments for a marketplace of 30K+ web Actors, and how we made it work even with the current specs.

SESSION AI in Finance · Track 3 12:05pm-12:25pm

How Kepler Built Verifiable AI for Financial Services

Vinoo Ganesh — CEO & Co-Founder, Kepler

Financial answers have to be auditable. Vinoo Ganesh (CEO, Kepler) shows how Kepler Finance pairs Claude's reasoning with deterministic verification infrastructure to index 26M+ SEC filings across 14,000+ companies and 27 markets — and validate every number back to the exact filing, page, and line item. A look at trust, provenance, and content engineering for AI in regulated finance.

SESSION Local AI · Track 4 12:05pm-12:25pm

Local AI Demos

Rolling demos: GLM 5.2 running on DGX Station; Nemotron 3 Ultra running on 4× DGX Spark; real-time speech on a single Spark; and visual/diffusion generation on a single Spark.

SPONSOR Graphs · Track 5 12:05pm-12:25pm

From Systems of Record to Systems of Context

Omri Bruchim — Engineering Group Manager, Monday

Enterprise AI agents are moving fast, but most of them still hit the same wall in production: they have access to tools, documents, APIs, and databases, but they do not understand the real context of how work gets done. At monday.com, we are building agents that operate across real customer workflows, internal product surfaces, knowledge, permissions, memory, and actions. The hard part is not just calling the right tool or retrieving the right document. The hard part is building a reliable context layer that helps agents understand users, work objects, organizational knowledge, prior decisions, business rules, and the relationships between them. This talk will explore the emerging idea of the context graph: a living, queryable layer that connects entities, history, permissions, decisions, and meaning across an organization. Foundation Capital describes context graphs as the next major enterprise AI opportunity because agents need more than rules. They need decision traces: how rules were applied, where exceptions were made, who approved what, and what precedent actually governs reality. I will share how we think about this opportunity at monday.com, how we are implementing parts of it in practice, and what we have learned from building AI agents inside a real AI work platform. The talk will include concrete examples, including how context is collected, represented, retrieved, governed, and evaluated. The audience will leave with a practical framework for moving beyond one-off RAG pipelines and prompt stuffing toward a reusable context layer that compounds over time, improves agent quality, and becomes a strategic moat for companies building AI-native products.

SESSION AI in GTM · Track 6 12:05pm-12:25pm

The Building Blocks of GTM Orchestration

Arman Vaziri — Senior Staff Software Engineer, Ramp

Ramp built its own 0→1 revenue stack in-house — Ramp Revenue — with one mandate: build the most efficient GTM org in the world. Arman Vaziri breaks down the building blocks: a customer data platform that chews through millions of internal, external, and CRM records daily, and a unified action layer with agents embedded directly in seller workflows. The payoff — reps stop hopping between dozens of systems just to figure out who to reach and what to say, and 80%+ of Ramp's sales workflows now run on it. A look at the architecture behind orchestrating GTM at scale.

SESSION **AI in Healthcare · Track 7** 12:05pm-12:25pm

200 Million Patient Interactions Later: What the Generic Voice Stack Misses

Vivek Muppalla — VP AI Engineering, Hippocratic AI

A healthcare voice agent can be right on the benchmark and still fail in production. Real patients hesitate, interrupt, misremember medications, code-switch mid-sentence, and disclose risk indirectly. After **200M+ patient-agent interactions**, the lesson is clear: in clinical voice AI, interaction is a safety variable. This talk breaks down what Hippocratic AI had to rebuild beyond the generic voice stack: not just ASR, VAD, an LLM, TTS, and turn-taking heuristics, but a real-time safety system that treats silence, clarification, escalation, multilingual continuity, and medication-specific recognition as first-class engineering problems. We'll walk through the production architecture behind Hippocratic AI's voice agents: a **30+ model supervisor constellation**, including the **4.1T-parameter AI Front Door system**, designed to catch failures a single primary model misses. The talk covers how specialized models monitor medication identification, overdose risk, labs and vitals, escalation criteria, workflow confirmation, and other clinical safety surfaces while the patient conversation is still happening. We'll focus on four production lessons: - **Benchmarks are not enough:** MedQA and USMLE-style accuracy do not capture the failure modes that appear in a 12-minute, multi-turn patient call. - **Interaction signals become training data:** pauses, interruptions, hesitation, clarification requests, and escalation markers are mined from production calls and turned into structured eval and training signals. - **One LLM is not a safety architecture:** supervisor models can overrule, block, or escalate when the primary model sounds plausible but misses a clinical risk. - **Voice infrastructure has clinical failure modes:** domain ASR, medication vocabulary, code-switching, latency, and turn-taking all affect whether the system makes the right next move.

SESSION **Agentic Engineering · Track 8** 12:05pm-12:25pm

Benchmarking Coding Agents on New vs Legacy Code bases

Denys Linkov — Head of ML, Wisedocs

You have an old code base with 100,000s of lines of code, should you let an AI Agent refactor or do you wait until you have a cleaner setup? Last year we refactored a number of code bases and ran evaluations on how well different models, harnesses and rule sets affected multiple versions of the code base. This talk will feature specific code examples as well as a broader set of evals.

SESSION **Inference · Track 9** 12:05pm-12:25pm

Vertical Mobility: Building an AI Inference Platform That Scales from MVP to Trillion-Parameter Workloads

Rita Zhang — Coreweave · **Sitanshu Gupta** — Coreweave

The future of AI inference is not one-size-fits-all. This talk explores a multi-tiered architecture that supports the full AI lifecycle, from rapid, pay-per-token experimentation to dedicated, SLO-bound production and extreme-scale, self-managed deployments. Learn about lessons learned from CoreWeave's inference stack as performance, cost, and control requirements evolve.

SPONSOR **Track M** 12:05pm-12:25pm

Design multi-agent systems that actually work

Tina Manghnani — Product Manager, Microsoft

Real-world agent systems don't run in isolation. Learn how to design and coordinate multi-agent systems that collaborate effectively in production—splitting responsibilities, managing system-level complexity, and operating with shared context from Microsoft IQ. See how agents move from single interactions to orchestrated systems that reason, act, and evolve together.

SESSION **Inference · Leadership 1** 12:05pm-12:25pm

Stop Model Shopping: Why Ownership Beats Choice in the Agent Stack

Pranay Bhatia — AI engineer and product leader, Fireworks AI

Teams shipping successful agents at scale know that model ownership is now a much more durable advantage than model choice. They're fine-tuning open models using their proprietary data, building tight data feedback loops between their products and their models, and treating customization as a core product discipline to differentiate. I've spent the last decade building AI infrastructure, first as co-creator and head of PyTorch at Meta, now as CEO of Fireworks AI, where my team powers AI agent infrastructure stacks for companies like Cursor, Notion, Uber, DoorDash, and Vercel. I've watched hundreds of teams try to ship agents into production, and the patterns behind their success and failure are remarkably consistent. In this talk, I'll share hard-won lessons from real production deployments across coding, productivity, and enterprise use cases, like: - Model choice matters, but model ownership matters more. Fine-tuning on proprietary data and building a feedback loop between your product and your models creates compounding advantages that no API swap will ever replicate, and it's now the standard for all state-of-the-art models. It's how Cursor hit 1,000 tokens/sec with quality that off-the-shelf models could never match, and it's how Quora saw 3x speed improvements in its chatbot Poe. - The eval gap is where most agent projects die. Teams will spend months on prompt engineering and model selection, then ship without rigorous evaluation. Treating AI development with the same discipline as software development, with CI/CD, regression testing, and continuous evaluation, is what separates production-grade agents from impressive demos. A custom evaluation suite, coupled with RFT, is how Genspark achieved 12% higher quality on their trained model, resulting in a 50% cost reduction. - The real moat is the data flywheel. When you own the loop between your product, your data, and your models, every interaction makes the system better. Surrendering that loop to a third-party provider means surrendering the very data that makes your product defensible. Ownership is how Vercel created a custom code model that matched competitor quality at 40x speed. I'll ground this talk in real examples I've seen work and fail across hundreds of agent deployments.

SESSION **AI Architects: AI Factories · Leadership 2** 12:05pm-12:25pm

Preferences > Benchmarks: Model Routing for How Teams Actually Build

Archana Kamath — VP of Engineering, Digital Ocean · **Tyler Gillam**

— Senior Software Engineer II - Agentic AI, Digital Ocean

There is no best model. There's only the right model for a given task, and the right model depends on your team's preferences, not a benchmark score. This talk makes the case for preference-aligned routing: choosing models by the constraints that actually matter — cost, latency, task type, model preference — instead of a single leaderboard number. We'll demo a sub-200ms routing decision running on a purpose-built 30B MoE model with no application code changes, walk through real coding workflows routing most traffic to open models without losing accuracy, and show where this goes next: evals, caching, and personalization.

SESSION **Expo Stage 1 NE** 12:05pm-12:25pm

The Missing Layer in Agentic AI

Giedrius Steimantas — Director of Scraping Engineering, Oxylabs

Reasoning is solved. Web access isn't. Most agents break the moment they leave the sandbox blocked, rate-limited, or staring at a CAPTCHA. Giedrius will show the three primitives every production agent needs: a browser, a fast search API, and a universal scraper and demo an agent built on top of them that actually works in the wild.

SESSION **Expo Stage 2 NW** 12:05pm-12:25pm

While You Were Generating: The Verification Gap Nobody Talked About

Ali Adl-Tabatabai — Founder and CEO, Gitar.ai

Every enterprise is asking the same question: how do we move fast with AI without breaking things? While the market chased generation — better models, faster agents, more output — a different problem was compounding quietly: nobody built the verification layer to match. The team built Gitar because they saw firsthand what happens when development velocity outpaces code quality, and AI has made that problem an order of magnitude bigger. In this session, Ali-Reza Adl-Tabatabai, formerly of Uber, Google, and Meta, now leading Gitar development inside Sonar, makes the case for why AI-native code review is the missing layer in every enterprise's agentic stack. Gitar uses agentic reasoning to review code, generate fixes, validate them against your CI, and commit to the branch. It automatically analyzes and de-duplicates CI failures, detects flaky tests, and fixes remaining build, lint, and test failures — keeping reviews moving across time zones without the back-and-forth that kills engineering throughput. As a critical layer in Sonar's multilayered, zero-trust verification platform, Gitar enables organizations to analyze syntax, data flows, logic flows, architectures, and dependencies; set and enforce standards in a consistent, auditable manner; and agentically fix issues both as agents write code and in CI workflows. Sonar intelligently sequences analysis so deterministic verification handles simpler issues first, while AI tackles the nuanced ones, reducing token costs and keeping the pipeline lean. In an agentic world, zero trust is an engineering principle: assume every line an agent writes needs to be verified, every time, at every layer.

SESSION **Expo Stage 3 SW** 12:05pm-12:25pm

Move fast and (don't) break things

Ben Dicken

Engineers want to move fast with AI, but the infrastructure underneath is buckling. Status pages across the industry make this clear. Here, you'll learn how to build systems that maintain 4-nines of availability while meeting unprecedented customer demand using the principles of extreme fault tolerance. PlanetScale has written about how we apply these principles to operating databases across our fleet (https://planetscale.com/blog/the-principles-of-extreme-fault-tolerance). This matters not just for databases, but all aspects of reliable infrastructure. Isolation, redundancy, static stability, and back-pressure are the building-blocks to achieving this. Sticking to such principles when architecting the backend of AI applications ensures our systems are resilient to failure while still being flexible enough to scale. We'll look at concrete failure modes from production systems and the patterns that prevent them.

SESSION **Expo Stage 4 SE** 12:05pm-12:25pm

Agents That Forge Their Own Tools: Self-Modifying AI in the Wild

Sandhya Subramani — Senior Developer Advocate, GenAI, Amazon Web Services

What happens when your agent decides its existing tools aren't good enough and writes new ones? Self-modifying agents can generate, test, and deploy their own tool implementations at runtime, adapting to problems they weren't explicitly programmed to solve. In this session, we'll demo a live agent that forges its own tools on the fly, discuss the safety boundaries you need, and explore where this pattern makes sense (and where it absolutely doesn't).

12:30pm

12:30pm-1:30pm **The Great Loops Debate**

SESSION **Expo Stage 2 · Expo Stage 2 NW** 12:30pm-1:30pm

Latent Space Live: the Inference Inflection from First Principles

swyx — Curator, Latent Space / AI Engineer · Rob Wachen — Co-founder and President, Etched

1:30pm

SESSION **Harness Engineering · Main Stage** 1:30pm-1:50pm

Loophole - Adversarial Agents To Stress Test Your Morality

Brendan Rappazzo — Machine Learning Scientist, Morgan Stanley

Most natural language specifications have holes their authors didn't notice - and writing more rules tends to create more holes. I built Loophole to try a different approach: point adversarial agents at a spec until it stops breaking. You give the system a set of natural language principles. An AI drafts a formal codified version. Two adversarial agents go to work - one finds cases the code permits but the principles forbid, the other finds cases the code forbids but the principles allow. A judge agent patches the code when it can, but only if the fix doesn't contradict any prior ruling. When a contradiction can't be resolved, it escalates to you. Every decision becomes binding precedent, so the constraint space tightens round after round. I started with moral and legal reasoning as the demo, and on its own that's already interesting - it turns into a kind of game where you discover contradictions in your own beliefs that you didn't know were there. But the pattern generalizes well past that. The same loop works for company policies that need to survive contact with edge cases. For making chatbot system prompts adversarially robust. For stress-testing eval rubrics. And, taking the long view, for something like a smarter legislative process - where proposed laws get checked against the public's stated values before they pass, and the contradictions surface before they hit a courtroom. The talk walks through how the harness works, the design choices that matter (especially why precedent is the load-bearing piece), what kinds of specs it handles well, where it breaks, and what it would take to push it further. All code is open source.

SESSION Generative Media · Track 1 1:30pm-1:50pm

While my guitar gently speaks

Todd Fisher — Head of Engineering Launching 0-1 startups, Philo Ventures

Do you ever wonder What the next evolution of live performances will look like? I do all the time. Come experience what happens when you combine live guitar playing with DSP as well as TTS and other models, all running locally. Prepare to be entertained and get familiar with new possibilities that modern tools open up in the audio and digital signal processing space while you enjoy a live performance on top of an informative slide presentation. Walk away from this talk inspired to help build the next evolution of options for musicians and live performances. We will touch on building with tools such as classic DSP, JUCE, TTS, STT, pitch detection with YIN, llama 3 and more with an emphasis of running it all locally on device! You might even get a chance to have a conversation with a guitar!

SPONSOR Agentic Commerce · Track 2 1:30pm-1:50pm

Agent Spending Without Controls: The Missing Infrastructure Layer for AI Pa...

Rodrigo Coelho — CEO, Edge & Node · Pranav Maheshwari — Director of Integrations, Edge And Node

AI agents are already transacting autonomously, but the infrastructure to govern how they spend does not yet exist. Traditional payment rails were built for humans, not for systems making thousands of micro-decisions per minute on someone else's behalf. This session brings together Edge & Node's CEO and Senior Solutions Architect to cover both the strategic case and the technical implementation. Rodrigo opens with the infrastructure gap: why programmable budget governance is a foundational requirement for any team deploying agents in production, and what it means to have real-time visibility and a full audit trail across every agent transaction. He also covers Edge & Node's founding membership in the x402 Foundation and why open standards for agent-to-agent and agent-to-service payments matter for the broader ecosystem. Pranav then goes deep on the stack: how structured, indexed blockchain data from The Graph powers reliable agent decision-making, how Amp Enterprise extends that into auditable datasets at production scale, and what it looks like in practice to wire ampersend into agent frameworks including LangChain, CrewAI, AutoGPT, and custom-built systems. He walks through the x402 and A2A standards that make agent payments interoperable and what a real deployment looks like end to end. The session closes with the bigger picture: bots are already half of all internet traffic, TradFi and DeFi are converging, and the infrastructure stack that wins is the one built for where they meet.

SESSION AI in Finance · Track 3 1:30pm-1:50pm

Build for the Memo, Not the Demo — Notes from 200 Investment Committees

Shawn Chan — Vice President, China Resources Holdings

By the end of this talk you will have a buyer-side specification for AI investment agents, the exact artifacts, evidence formats, and trust gates a senior finance team will require before letting an AI system touch a $100M+ capital allocation decision. Drawn from fifteen years and roughly 200 investment committees at CK Hutchison (A.S. Watson Group) and China Resources Holdings, on the side of the table the AI engineering audience almost never hears from. Most enterprise AI in finance is still being built by engineers who have never sat in an investment committee. I have spent fifteen years on the other side of that demo, cross-border M&A, IPO execution and strategic investment, as a buyer on deals including Oatly (Series B through Nasdaq IPO), Airbnb (Series F), SenseTime, Moore Threads, Leapmotor and EVE Energy, and on the A.S. Watson tri-market IPO and Temasek's strategic stake. I have watched analyst memos get torn apart, and signed off on decisions where being wrong meant being wrong by nine figures. From that seat, almost every AI finance demo I have seen has the same problem: it optimizes for the demo, not for the memo. This talk walks through the specific failure modes that kill AI agents at the IC door: Source hierarchy is not retrieval. A footnote in an audited 10-K outweighs a sell-side note, which outweighs a transcript, which outweighs an internal email. Most RAG systems flatten this. Numerical consistency is non-negotiable. A memo that says "revenue grew 18%" in paragraph one and "17.4%" in the sensitivity table is dead on arrival. Contradiction is a feature. Real diligence surfaces conflicts between sources; AI agents tend to silently resolve them. Every assumption must be separable from every fact. Investment committees do not approve assumptions hidden inside prose. Audit trail is the deliverable. If a regulator, an auditor, or a board member cannot trace a claim back to evidence in under thirty seconds, the system is unusable. Accountability cannot be delegated to a model. Someone has to sign the memo. The architecture has to reflect that. The session closes with a concrete buyer-side specification, what an AI investment agent must produce, in what form, with what evidence, before a senior finance team will let it touch a live deal. Not a framework slide.

SESSION **Local AI · Track 4** 1:30pm-1:50pm

Local Models: Trust, Control, Optimization

Carter Abdallah — Senior Developer Tech, NVIDIA · **Vincent Weisser** — Co-founder & CEO, Prime Intellect · **Lucas Atkins** — CTO, Arcee AI · **Chris Alexiuk** — Sr. Product Research Engineer, NVIDIA

Local Models: Trust, Control, Optimization looks at why builders are choosing local AI for privacy, reliability, customization, cost, and ownership, while still asking where cloud remains necessary. The panel covers local-first versus hybrid strategies, the role of open-source models, and the infrastructure stacks making frontier-quality intelligence possible outside centralized APIs. Moderator: Carter Abdallah (NVIDIA). Panelists: Vincent Weisser (Prime Intellect), Lucas Atkins (Arcee AI), Chris Alexiuk (NVIDIA), Lou (Z.ai).

SPONSOR **Graphs · Track 5** 1:30pm-1:50pm

AI : Learned Execution Graphs for Real-Time Anomaly Detection & Drift Classification in APIs

Ritvik Pandya — Engineering Manager, JP Morgan Chase

API ingress controllers process requests through ordered sequences of middleware steps — authentication, authorization, validation, rate limiting, routing, service invocation, caching. We model this pipeline as a directed acyclic graph (DAG) learned from structured telemetry events, then apply graph-based anomaly detection and drift classification in real time at 1,600+ TPS. The system emits one structured event per processing step, constructs per-endpoint execution graphs using sequence mining with statistical confidence thresholds, and learns per-node baselines (latency, dependency, execution frequency). Three graph intelligence capabilities emerge: (1) Graph-based anomaly attribution — compute per-node deviation ratios against learned baselines to identify the exact bottleneck node and its dependency. In production, this pinpointed a 41x deviation at a single graph node that was invisible to service-level monitoring, reducing root cause identification from 2-3 hours to under 30 seconds. (2) Graph structural drift detection — compare observed node sequences against the learned graph topology to detect missing nodes (mandatory processing step silently skipped), reordered nodes (middleware misconfiguration), and unexpected new nodes (unauthorized middleware injection). Traditional monitoring reported "system healthy" when a mandatory node was removed — latency dropped, errors at zero — only the learned graph comparison detected the structural change. (3) Per-client graph fingerprinting — learn client-specific execution graph profiles using exponential moving averages. Detect when a client's graph traversal pattern changes, classify the cause (client behavior change vs. configuration drift vs. infrastructure failover) using KL divergence on node-visit distributions, and apply graph-aware adaptive control scoped to specific nodes rather than entire endpoints. The execution graph model also enables a novel approach to retry storm detection: analyzing idempotency key entropy at graph nodes to classify traffic as legitimate growth vs. retry amplification, and returning cached responses at the specific graph node rather than rejecting requests — breaking the retry amplification loop. Production system processing high TPS. Attendees will learn the graph construction methodology, the anomaly attribution algorithm, and concrete patterns for adding learned graph intelligence to any middleware pipeline.

SESSION **AI in GTM · Track 6** 1:30pm-1:50pm

How Juries and Librarians Can Solve GTM's AI Trust Problem

Alex Bauer — Co-founder, Upside

A couple of years ago, everyone worried about AI hallucinating. We rarely hear that word anymore, but it’s just because the problem grew up. Today, your AI still doesn’t know how to say “I’m not sure.” Instead, it hands you a revenue number that’s wrong in ways that look exactly like being right. The good news is we already solved this once, for people: you onboard a new hire so they understand your business; you put subjective, high-stakes calls in front of more than one set of eyes. This talk walks through patterns we run at Upside, including a librarian every agent consults before it acts, a jury-and-judge model for the questions a single pass can’t be trusted to answer, and knowing when the model itself is just too dumb for the job. Live demos and real failures included.

SESSION **AI in Healthcare · Track 7** 1:30pm-1:50pm

AI is becoming the World's largest Relationship Therapist. We Can't Afford to Get it Wrong.

Clay Cockrell — Co-Founder, CoupleWork AI · **Tony Fabrikant** — Co-founder, CoupleWork AI

Millions of people are now turning to AI for relationship advice and emotional support, often before they'd ever consider a human therapist. Most of the AI Therapy that is available is without clinical oversight, ethical frameworks, or any serious reckoning with what it means to intervene in the most intimate and vulnerable space in a person's life. People are getting hurt. As a couples therapist with 30 years experience, I teamed up with the former CTO at S&P and we created CoupleWork, an AI relationship therapist I essentially trained on three decades of clinical knowledge and every evidence-based modality that exists. Our voice interactive AI, Maxine, is proving this can be done responsibly and very effectively. And what we're learning about the nature of love, connection, and human vulnerability at scale is something this industry needs to hear. I also want to talk about what comes next: the regulatory frameworks that don't yet exist, the liability questions nobody is answering, and why the therapists who should be leading this conversation are almost entirely absent from it.

SESSION **Agentic Engineering · Track 8** 1:30pm-1:50pm

Codex, Behind the Harness

Dominik Kundel — Developer Experience Lead, OpenAI

Agents have evolved a lot in the last year both in capabilities and in the overall structure. Increasingly sandbox-powered coding agents are breaking out to do general purpose work. In this talk we'll be taking apart the open-source Codex agent harness. Understand how it works, what makes it so suitable to do work beyond coding tasks, how it handles key aspects like context management, tools and file system access. We'll also tie these back to concrete actions you can take to bring these patterns into your own agents, whether you are building on top of the Codex agent or building your own.

SESSION **Inference · Track 9** 1:30pm-1:50pm

What's New in Inference Engineering

Philip Kiely — Developer Relations, Baseten

More than 30,000 engineers have learned the fundamentals of inference since Inference Engineering was published. But the field keeps accelerating, so it's time for the first public addendum to the book. The past four months have seen a renewed focus on training-dependent inference optimization across the "big three" performance techniques of speculation, caching, and quantization. This talk provides structured guidance for training DFlash and EAGLE 3 draft models to accelerate LLM decode, introduces the concept of KV compaction, and explains the hype behind TurboQuant.

SPONSOR **Track M** 1:30pm-1:50pm

Evaluating and optimizing AI agents: from observability to continuous improvement

Chang Liu — Senior Product Manager, Microsoft

AI agents don't behave like traditional systems. Learn how to evaluate outputs, trace behavior, and apply a continuous loop to improve performance across prompts, tools, and models. Using signals grounded in real-world context via Foundry IQ, see how evaluation, tracing, and optimization come together to turn production usage into measurable improvements over time.

SESSION **AI-Native Enterprises · Leadership 1** 1:30pm-1:50pm

From Zero to AI-Native: Scaling AI Across the Org

Josh Leavitt — Sr. Director of AI & Data, Coinbase

Most companies talk about being AI-native, but few show what it takes at scale. Josh Leavitt, Sr. Director of AI & Data at Coinbase, shares the hard-won playbook for transforming a high-stakes, regulated engineering organization into one where AI is a first-class citizen across every team. Josh can cover my approach towards building a centralized AI platform that serves thousands of engineers without becoming a bottleneck, tooling decisions that actually moved the needle, and what AI-native really means when shipping in a zero-tolerance-for-failure environment. Expect concrete frameworks, real examples, and honest lessons from what didn't work.

SESSION **AI Architects: AI Factories · Leadership 2** 1:30pm-1:50pm

Coding Agents Don't Scale Themselves. Neither Do Your Teams.The Rise of Agent Enablement.

Patrick Debois — Member Technical Staff, Tessl

Every company wants to know how others are actually scaling AI coding. But it's hard to get past the generic transformation stories. What are the new practices showing up in real engineering orgs? What does maturity actually look like, and what separates teams that are moving from teams that are stuck? What are the patterns for enabling humans and agents, together? Patrick Debois has been collecting the practices and patterns, talking to the early Agent Enablement teams already on the job, team leads, and VPs of Engineering. What's showing up is a new function: a team that enables other teams to get real leverage out of their agents. This talk takes the [Context Development Lifecycle](https://tessl.io/blog/context-development-lifecycle-better-context-for-ai-coding-agents/) off the individual laptop and onto the org chart, grouped across three pillars: - **Enablement.** From individual experimentation to team and org-level fluency with agents. - **Platform.** Agent tooling that runs like a real delivery pipeline: fast, observable, cost-aware. - **Governance.** Ad-hoc guardrails growing into real evaluation, telemetry, and accountable agent work. For Agent Enablement leaders scaling it out across the org. For team leads looking to help their teams get better at this. For VPs ready to unblock the friction and unlock what agents can actually do. *Coding agents don't scale themselves. This is the talk about who does*

SESSION **Expo Stage 1 NE** 1:30pm-1:50pm

Trust, But Verify: Human-in-the-Loop for Agents That Actually Matter

Michael Liendo — Staff Developer Advocate, Auth0

"In this talk we'll walk through the full spectrum of human-in-the-loop patterns, from lightweight inline confirmations to out-of-band permission gates to handing your agent a wallet with real money in it and more. Each pattern fits a different level of consequence, and knowing which to reach for is what separates demo agents from production ones. We'll cover the honest tradeoffs of latency, user experience, and trust so you can make the right call for your specific use case. The entire talk is built around various live demos that escalate in stakes with every step. You'll leave with a mental model and working reference architecture you can apply the same day."

SESSION **Expo Stage 2 NW** 1:30pm-1:50pm

YOLO Mode, Safely: microVM Sandboxes for Any Agent

Rowan Christmas — Staff Product Manager, Docker

This talk shows the alternative: every agent session in its own microVM, with its own kernel and a hard boundary to the host. You decide what lives inside that boundary: filesystem, network, the tools it's allowed to call. The sandbox runs Claude Code, Cursor, Codex, or whatever else you're driving. You'll see an agent live in full YOLO mode inside a sandbox, a real attempt to escape, and the boundary that holds up.

SESSION **Expo Stage 3 SW** 1:30pm-1:50pm

Your Model is Private. Your System Isn't.

Joshua Mo — Lead DevRel Engineer, Venice AI

Privacy in AI isn't just about choosing the right model. Data leaks rarely happen inside the LLM itself - they happen in the systems surrounding it. Observability pipelines, analytics platforms, prompts, agents, and infrastructure often become accidental channels for exposing user data. In this session, Joshua Mo, Lead DevRel Engineer at Venice AI, explores why private models alone are not enough and shares practical privacy-preserving patterns that AI engineers can adopt today. From revocable handles and hashed identifiers to agent boundaries and confidential computing, attendees will leave with concrete ideas for building AI systems that protect user data by design.

SESSION **Expo Stage 4 SE** 1:30pm-1:50pm

Video Discovery for Agentic World-Model Training

Rafael Levi — DevRel, Bright Data

Physical AI had its “Attention Is All You Need” moment with the rise of Vision-Language-Action models. The next bottleneck is data: not just more video, but the ability to find the exact real-world moments that teach models how the world works: gravity, motion, causality, human behavior, and object interactions. This session explores a new approach: discovering specific scenes from the vastness of the web. We’ll show how teams can search for moments like objects falling, people interacting with environments, or actions unfolding over time, then collect and structure only the relevant clips for training and evaluation. Attendees will learn how scene-level discovery changes multimodal data pipelines, reducing wasted collection, processing, storage, and review, while making it easier to build targeted datasets for VLA systems, robotics, physical AI, and agentic world models.

1:55pm

SESSION **Harness Engineering · Main Stage** 1:55pm-2:15pm

🎵 Every step you take, every call you make - the reliable agent stack

Giselle van Dongen — Developer Advocate, Restate

In this session, we skip past the demos that work only on your laptop, and go straight to how you can build production-ready agents with a stack that covers all the hard bits of backend development that you don’t want to be bothered with when developing your agents: - Failure resiliency: retries, timeouts, and exactly-once execution so a flaky API or a crashed process doesn't corrupt your agent's state or makes them start from scratch - Durable Sessions: a session store with built-in conversation isolation and protection against corruption from concurrent agents - Pause/resume for human approvals: survive human approvals and research that take weeks without building complex infra - Agent-to-agent messaging layer: call agents developed by other teams or running on other infra with resilient HTTP calls - A kill switch: cancel a running agent cleanly at any point, without leaving half-executed work behind We will demonstrate each concept with live code examples, using Python, OpenAI Agents SDK and Restate as open-source Durable Execution engine. All examples are generally applicable: pick your favorite agent SDK (OpenAI Agents, Pydantic AI, Vercel AI, Google ADK,...) or go wild and implement low-level custom agents by just tying together LLM calls with custom logic.

SESSION Generative Media · Track 1 1:55pm-2:15pm

Voice agents with Realtime Video

Lina Colucci — CEO, LemonSlice

SPONSOR Agentic Commerce · Track 2 1:55pm-2:15pm

Teaching agents to pay

Anna Spysz — Developer Relations Engineer, Stripe

With a global daily user base in the hundreds of millions, AI agents are rapidly becoming a primary interface for how people discover, evaluate, and purchase products. Enabling those products to be listed and paid for directly through agents opens an entirely new - and enormous - commerce channel. The Agent Commerce Protocol (ACP) and Shared Payment Tokens provide a secure framework for agent-driven commerce within Stripe's ecosystem - without exposing payment data or sacrificing user control. This session walks developers through the complete implementation: setting up Stripe integration, creating permission-based payment tokens, interacting with ACP endpoints, and designing trustworthy user experiences. You'll learn how to enable your agents to transact safely and predictably, handling everything from checkout flows to error scenarios and webhook events.

SESSION AI in Finance · Track 3 1:55pm-2:15pm

We Vetted 2,000 AI Skills Before They Reached Developers

Lucas Palma — Information Security Manager, Nubank

AI skills and plugins are becoming part of the software supply chain. They steer agent behavior, describe tools, run commands, access files, and shape how developers build with AI. Treating them as harmless configuration is a mistake. This talk shares what we learned from building an automated security review system for more than 2,000 internal AI skills before they reached a company wide plugin marketplace. I will walk through the risks we found, the checks that worked, the checks that created noise, and how we turned skill review into something developers could run locally and in CI. We will cover practical patterns for reviewing unsafe instructions, destructive commands, sensitive data exposure, risky tool use, credential handling, external calls, and agent behavior drift. The goal is to help AI engineers think about skills, plugins, and agent instructions as production dependencies that deserve review before they reach real users.

SESSION Local AI · Track 4 1:55pm-2:15pm

Local Models: Trust, Control, Optimization

Carter Abdallah — Senior Developer Tech, NVIDIA · Vincent Weisser — Co-founder & CEO, Prime Intellect · Lucas Atkins — CTO, Arcee AI · Chris Alexiuk — Sr. Product Research Engineer, NVIDIA

Local Models: Trust, Control, Optimization looks at why builders are choosing local AI for privacy, reliability, customization, cost, and ownership, while still asking where cloud remains necessary. The panel covers local-first versus hybrid strategies, the role of open-source models, and the infrastructure stacks making frontier-quality intelligence possible outside centralized APIs. Moderator: Carter Abdallah (NVIDIA). Panelists: Vincent Weisser (Prime Intellect), Lucas Atkins (Arcee AI), Chris Alexiuk (NVIDIA), Lou (Z.ai).

SPONSOR Graphs · Track 5 1:55pm-2:15pm

Why Agentic Systems Need Ontologies

Frank Coyle — Lecturer, UCALBerkeley / Founder AI/Edge, UCAL Berkeley

Agentic systems fail in predictable ways: context degradation, brittle tool descriptions, fragile multi-agent handoffs, stop-reason confusion, and the ever-present temptation to fix reliability problems with more natural-language instructions. These anti-patterns aren't bugs to be patched turn by turn — they're symptoms of a missing architectural layer. LLMs reason probabilistically over domains they only partially understand, and no amount of prompt engineering fully closes that gap. This talk argues that the missing layer is an explicit ontology: a formal, shared map of the domain's concepts, relationships, and constraints. The pattern is not new — ontologies have driven commercial success in defense and intelligence systems for over a decade, where probabilistic models must operate over high-stakes enterprise data without drifting into nonsense. Graph databases like Neo4j and Amazon Neptune have made the underlying primitives widely accessible. We'll show how lightweight ontology constructs can surround an agentic system with enforceable logical constraints: typed entities and relationships that tools must respect, cardinality and domain restrictions that catch malformed tool calls before they execute, and a shared vocabulary that keeps coordinators and subagents talking about the same things. The session walks through several agentic applications — a multi-agent research workflow, a tool-heavy customer support agent, a coordinator-subagent delegation pattern — and shows in each case how an ontology layer addresses the kinds of anti-patterns catalogued in Anthropic's Claude Certified Architect exam. The result is a hybrid neurosymbolic architecture: probabilistic reasoning inside, logical guardrails outside. Who should attend: engineers building production agentic systems, architects evaluating reliability strategies beyond prompt engineering, and technical leads who suspect their agents need more structure than another system prompt can provide.

SESSION AI in GTM · Track 6 1:55pm-2:15pm

How We Got LLMs to Recommend Our Open Source Library (Without Paying or Plug-ins)

Christopher Burns — Founder, Inth

Over the past year, we've seen a new distribution channel emerge: AI assistants. Instead of SEO, ads, or integrations, developers are discovering tools through models like Claude. In this talk, I'll break down how we got our open source library recommended organically by LLMs in under a year, without plugins, paid placements, or partnerships. We'll cover what actually influences model outputs today, how developer-first products behave differently in this channel, and the practical steps we took to make our project show up when it matters. This is not theory. It's a real case study of how distribution is changing, and how you can design your product and content to be picked up by AI systems directly.

SESSION AI in Healthcare · Track 7 1:55pm-2:15pm

Healthcare's Agent Bytecode: X12 as the Harness for AI Agents

Vasant Kearney — CEO and Founder, Onlay

LLMs made old languages newly useful: COBOL for mainframes, Fortran for scientific code, and Rust, SQL, and Prolog as strict substrates for agentic systems. Healthcare has its own old language hiding in plain sight: X12. Before LLMs, X12 was mostly treated as ugly plumbing: loops, delimiters, companion guides, clearinghouse edits, payer-specific quirks, rejections, and acknowledgments. In an agentic workflow, those constraints become the feature. They give stochastic agents a deterministic target. This talk shows how healthcare agents can compile messy operational evidence into X12-shaped workflows: chairside audio into 837D claim narratives, imaging systems into 275/PWK attachment flows, payer portals and phone calls into 270/271 eligibility and 276/277 claim status, preauth evidence into 278 workflows, and EOBs, scanned mail, and bank data into 835/820 payment reconciliation. The core pattern is simple: LLMs reason over ambiguity; X12 provides the syntactic and semantic harness for validation, auditability, acknowledgments, rejections, human review, and high-volume automation. This is not an EDI nostalgia talk. It is a production architecture talk about building reliable agents in one of the messiest enterprise domains.

SESSION Agentic Engineering · Track 8 1:55pm-2:15pm

Multiplayer agentic engineering: enabling your whole team and your best agents to work together

Arjun Singh — Co-founder and CEO, Superconductor

For a solo developer, coding agents are a superpower. For a team, they surface new kinds of bottlenecks: coordination, visibility, review, and shared context. We wanted our whole team and our best agents to work together, with no work or context trapped on any one developer's machine. So we pressed pause on the product we were building to create a multiplayer cloud workspace for agentic engineering. This talk shares five key practices we've learned from building and using our platform: Turn every surface the team uses into an agent interface. Kick off sessions from Slack, review via iOS app, iterate in GitHub comments, ship from web. Agents run in the cloud, so work keeps moving even when your laptop is closed. Make agent work visible and collaborative across the whole team. Every agent session is shared, has a live app preview, and an agent-guided code review. This allows engineers, PMs, and designers to steer and evaluate agent work collaboratively. Turn every external signal into shipped code your team can quickly evaluate. Automatically turn customer emails, meeting action items, and bug reports into agent implementations that the whole team can review. Set up shared cloud dev environments so agents aren't siloed to individual machines. Secrets, role-based access, and network controls shared across the whole team. Fast environment startup, so you're not giving up speed by moving off local. Benchmark agents on your own codebase. Claude Code, Codex, Gemini, Amp, OpenCode — how do you know which is actually better on your stack? We'll cover using your merged PRs as ground truth to build a "Personal SWE-Bench" for your codebase. Agentic engineering is going multiplayer. This is how your team gets there.

SESSION Inference · Track 9 1:55pm-2:15pm

Large clusters for small models

Daniel Svonava — CEO and co-founder, Superlinked

Small task-specific models are cheaper, faster and narrowly better than the frontier. But a wide catalog of small models is tricky to serve - dedicated worker pools sit idle, top-down request routers choke up on the huge volume of small requests, your users bring 100s of LoRAs.. In this talk we show how we serve 1M tokens per second with small models, how we architect our cluster for maximum throughput AND minimum latency and how we apply autoresearch to rewrite our inference code to support 10+ new models a week.

SPONSOR Track M 1:55pm-2:15pm

Blast Radius Zero: One-Command OpenClaw Sandboxes in the Cloud

Arun Sekhar — Principal Product Manager for AI Developer Experience, Microsoft

You already run OpenClaw locally, sandboxed in MXC. Now you need the same agent in the cloud for dev/test, reachable from Teams on your phone, without handing over the keys to the kingdom. This session shows a simple, one-command path to do all of this: an isolated Container Apps sandbox running an OpenClaw image, calling Azure OpenAI in Foundry Models securely without keys over the standard OpenAI API, scaling to zero when idle.

SESSION AI-Native Enterprises · Leadership 1 1:55pm-2:15pm

Which AI startups actually land enterprise contracts? Lessons from evaluating 100+ AI startups at Millennium Management

Brian Lewis — AI Product Lead, Millennium

Selling your AI startup/product into a large enterprise is hard. I often sit on the buyer's side of the table at a large hedge fund. I've sat through 100+ AI startup pitches and am responsible for running the pilots that may eventually convert into your ARR. We'll cover what works, what doesn't, and what large enterprise customers need to see in order to choose 'buy' over 'build'.

SESSION Harness Engineering · Leadership 2 1:55pm-2:15pm

Agent Frameworks Considered Harmful

Rémi Louf — CEO, .txt

SESSION Expo Stage 2 NW 1:55pm-2:15pm

MCP doesn’t suck — your agent does

Jan Curn — Founder & CEO, Apify

Most AI agents misuse MCP and treat tools as prompt-time function calls: tool definitions and results are repeatedly injected into the context, tokens are wasted, and context rots. The result? Slower, less reliable agents, and the misleading conclusion that “MCP sucks, CLIs are better.” To challenge this narrative and show how agents can get the best of both MCP and CLI, at https://apify.com/ we’ve built mcpc (https://github.com/apify/mcpc), an open-source universal CLI client for MCP. It maps MCP operations to intuitive CLI commands, which agents quickly pick up through --help without external skills. It turns out, CLI is the perfect local interface for agents to interact with MCP, giving them access to full protocol capabilities including modern features like code mode or progressive tool discovery through a single Bash() tool call, while leveraging MCP’s standard remote interface for server discovery, authentication, payments, and access control. To once and for all kill the MCP vs. CLI debate and show those two technologies are not exclusive but complementary, we’ll present evals comparing performance of agents using naive MCP, modern MCP, native CLIs, other MCP CLIs, and mcpc, in various real-world scenarios.

SESSION Expo Stage 4 SE 1:55pm-2:15pm

Everyone talks about document search, but what about results?

George He — Head of Platform Engineering, LlamaIndex

Search is usually treated as the end of the document pipeline: parse, chunk, retrieve, and hand them to the model. But long-running agents need something more durable than one-off retrieval. They need reusable work: structured outputs, citations, extracted entities, prior decisions, and file-system-like context they can return to across tasks. At scale, context management is where most agent systems fall apart. Without the right harness, agents lose track of what they've retrieved, bloat their context windows, and stall. In this talk, we'll look at why the document pipeline needs a stateful layer beyond the index — one that turns one-off retrieval into reusable, agent-ready context. We'll see how LlamaIndex thinks about transforming messy documents to make this possible, and why the future of document intelligence belongs to results that compound over time, not just better search.

2:00pm

SPONSOR Booth Activation · OpenAI Booth 2:00pm-2:30pm

AMA with Theo Browne (@t3.gg)

Theo Browne — Founder/YouTuber, T3 Tools & YouTuber

Stop by the OpenAI booth from 2:00-2:30PM for an AMA with Theo Browne (@t3.gg), say hello, and connect with developers and builders.

2:25pm

SESSION **Harness Engineering · Main Stage** 2:25pm-2:45pm

We let an AI agent execute Bash and lived to talk about it

Sarah Sanders — Context Engineer, PostHog

PostHog's Wizard agent can read your codebase, install packages, and run shell commands on your laptop. Yes, on purpose. This talk covers how we went from "defense-in-hope" to a standalone, robust security service. It'll highlight results from a pentest that made us question our life choices, an internal audit that challenged our architecture, and the debate over how to secure the entire pipeline. You'll learn why "scan-then-trust" is a weaker model than you think, what it takes to build kill switches you hope you never use, and what happens when you pentest an AI agent that has access to Bash.

SESSION **Generative Media · Track 1** 2:25pm-2:45pm

Generative Video at the Speed of Light

Keegan McCallum — Founder, uRun

Discussing recent breakthroughs in realtime generative video models, and the new architectural problems and bottlenecks involved in creating immersive, interactive experiences on top of these models.

SPONSOR **Agentic Commerce · Track 2** 2:25pm-2:45pm

The Agentic Commerce Stack

Ahnaf Prio — Senior Engineering Manager, Best Buy

Agents are already handling product discovery, cart building, and checkout — no human clicking required. But what's the protocol stack actually making this work? This talk maps the real infrastructure: MCP for tool access, A2A for agent coordination, the ACP spec (backed by OpenAI) and the UCP spec (backed by Google) — two competing approaches to standardizing the full agentic commerce lifecycle — and AP2 for agentic payments. We'll cover what each does, how they compose, and where they're still forming. Then we'll see it live — a working demo with a protocol inspector showing every tool call, task transition, and checkout event in real time. You'll leave with a clear mental model of the agentic commerce landscape and a reference implementation you can use.

SESSION **AI in Finance · Track 3** 2:25pm-2:45pm

Your Finance Agent's Bottleneck Is You

Ramana Siddanth Emani — Data Scientist, Auditoria AI

Most "AI for Finance" demos look great and almost none survive past pilot. If you've pushed an agent past one workflow, one tenant, or one Workday schema, you know the bottleneck isn't the model - it's the engineer behind the agent, who can't iterate fast enough to keep up with real AP data, real RBAC, and real query volume. What if you built your dev loop with the same primitives you're shipping to the finance team? In this talk, I'll show the subagent + skills + MCP stack - a production multi-agent system over AP, PO, vendor, and multi ERP systems, a LangGraph pattern that survives production, and the three failure modes that kill finance pilots before they ship.

SESSION **Local AI · Track 4** 2:25pm-2:45pm

Compression at the Edge

Chris Alexiuk — Sr. Product Research Engineer, NVIDIA · **Daniel Han** — Co-founder, Unsloth · **Asma Beevi** — Senior Engineer, NVIDIA · **Merve Noyan** — MLE, Hugging Face · **Parth Sareen** — Ollama

Compression at the Edge examines how smaller weights, faster inference, and constrained-memory deployments are making capable local AI more practical. The panel explores where compressed models already beat cloud on latency, privacy, cost, or control, what breakthroughs would unlock broader adoption, and how open model tooling is shaping the edge AI stack. Moderator: Chris Alexiuk (NVIDIA). Panelists: Daniel Han (Unsloth), Asma Beevi (NVIDIA), Merve Noyan (Hugging Face), Michael Chiang (Ollama).

SPONSOR Graphs · Track 5 2:25pm-2:45pm

Video Has No Memory. Here's How We Built One.

James Le — Head of Developer Experience, TwelveLabs

Every video AI query today starts from scratch. There's no durable state, no entity continuity, no way to ask "what does this corpus know?" instead of "find me something like this." This talk is about fixing that by engineering a proper memory layer for video intelligence, grounded in what we shipped at TwelveLabs with Jockey. What this talk covers: 1 - Why video memory is categorically different from text memory: Video is temporal, multimodal, dense, ambiguous, and evidence-sensitive. Larger context windows don't solve this. The problem isn't retrieval bandwidth, it's that there's no durable representation to retrieve into. 2 - The context graph as a systems concept, not a database choice: I'll define what "context graph" actually means in practice: time-bounded moments, cross-video entity resolution, appearance tracking, and relationship mapping. This is infrastructure-level thinking, not a graph DB sales pitch. 3 - Five design principles that determine whether video intelligence is reusable infrastructure or a search wrapper with extra steps: + Ingest once, reason many times (move expensive understanding work into preparation) + Store primitives, not just answers (moments, entities, appearances, relationships) + Ground every claim to source video (a timestamp is a product requirement, not a safety footnote) + Let intent shape memory (brand safety and sports highlights need different primitives from the same footage) + Keep the memory layer composable and API-first 4 - What this unlocks for builders. Corpus digest, agentic search with grounded references, entity-centric workflows, timeline reconstruction, and compliance tooling, all built on the same durable substrate. The talk is concrete and demo-grounded. You'll leave with a specific mental model for memory architecture, actionable decisions for ingestion pipeline design and entity resolution, and a clear line between "search with extra steps" and actual video intelligence infrastructure.

SESSION AI in GTM · Track 6 2:25pm-2:45pm

Knowledge Systems: The New GTM Stack

Jeffrey Wang — Co-founder, Exa

Exa built one of the world's largest knowledge graphs for AI agents. Then they realized it could power their entire GTM stack, from outbound to internal copilots.

SESSION AI in Healthcare · Track 7 2:25pm-2:45pm

Trading Desks to Clinical Trials: Parallels in Applied Vertical AI

Ayush Bhardwaj — Tech Lead, Allos AI

Wall Street to Wet Labs: The Shared DNA of Vertical AI. On the surface, finance and pharma couldn't look more different. One chases alpha in the markets; the other engineers complex drug delivery and stability. But under the hood, building Vertical AI for both domains reveals a striking shared DNA. Drawing from hands-on engineering experience in Applied AI at a top hedge fund and a cutting-edge pharma tech startup, this session explores the surprising architectural parallels between these two high-stakes industries.

SESSION Agentic Engineering · Track 8 2:25pm-2:45pm

Always-on agents run production without the on-call tax

Justin Smith — Founding Product Engineer, Resolve AI

Most production teams have the same problem. The work that keeps systems healthy- deployment checks, on-call handoffs, anomaly reviews- never makes it into a sprint. It falls to whoever has bandwidth, gets done inconsistently, and disappears when people are stretched thin. Background agents fix this by running that work on a schedule, using the same production context a senior engineer would, without waiting for someone to initiate it. Justin Smith, Founding Engineer at Resolve AI, walks through the architecture behind always-on agents, the use cases teams are starting with today, and what we have learned from running them in our production environment.

SESSION Inference · Track 9 2:25pm-2:45pm

The Frontier AI Inference Cloud for Agents

Byung-Gon (Gon) Chun — Founder & CEO, FriendliAI

Agents have changed the economics of AI inference. A chatbot's cost scales roughly linearly with the number of requests; an agent's scales multiplicatively. A single task can fan out into hundreds of model calls, each carrying a repeated context prefix and adding latency that compounds across tool calls and reasoning steps. As open-weight models keep improving and agentic workloads grow, this shift exposes the limits of traditional request-level optimization. Inference infrastructure becomes a first-class concern, one that often shapes performance and cost as much as the model itself. In this talk, we explore what changes when you optimize for the whole task rather than the individual request, and how FriendliAI is rethinking the inference cloud for the era of agentic AI.

SPONSOR **Track M** 2:25pm-2:45pm

Operate agents safely at scale with enterprise governance

Ashu Joshi — Director, Business Strategy, Microsoft

As adoption grows, governance becomes critical. Learn how to manage identity, compliance, and lifecycle for agent systems at enterprise scale.

SESSION **AI-Native Enterprises · Leadership 1** 2:25pm-2:45pm

Your Hero Agent Needs a Party

Kunal Lanjewar — Staff Engineer, Riot Games

A front-door persona, a party of deterministic specialist agents, A2A between. Your support bot deflects half its tickets, then, soloing a problem it was never built for, confidently runs the wrong `kubectl` command. Most teams respond by rewriting the prompt. The real fix is a multi-agent party of specialists. This talk gives you a production pattern that turns one over-leveled hero agent into a coordinated party of specialists you can trust on tier-zero infrastructure. Persona and ReAct agents make great heroes at the front door. Any team can copy one, paste it into their stack, and adjust the behavior in plain English. But if you send a lone hero to clear the dungeon, whether it is a deploy or an incident, a non-deterministic Reason-Act loop tends to loop, over-act, or punt back to a human. More prompts and more skills do not reliably level it up. Instead of soloing, keep the persona as the front-door face and give it a party: deterministic DAG specialists where the graph is fixed and the LLM is called only at decision points. For example, a deployment specialist can list rolling pods, choose the next tool, run it, read logs, and then diagnose the result. Each specialist is a class with one job and a narrow set of tools, and they coordinate over A2A for capability discovery and delegation across frameworks. Reliability and tighter least-privilege access become properties of the design, not something you try to bolt onto a prompt. You'll leave with the pattern: where to draw the line between the hero and its specialists, how to shape a DAG specialist so it decides instead of flails, and where A2A fits as the seam between them, grounded in lessons from a tier-zero fleet.

SESSION **Expo Stage 1 NE** 2:25pm-2:45pm

Optimizing Open Models for Production Grade Inference

Sujee Maniyam — Developer Advocate, Nebius · **Dylan Bristot** — Product Marketing, Token Factory

Open-source foundation models are rapidly closing the gap with proprietary systems, enabling organizations to build powerful AI applications with greater flexibility and control. However, deploying these models in production introduces a new set of challenges: latency, throughput, scalability, and cost efficiency.In this talk, we'll explore the modern inference optimization techniques that power large-scale AI systems in production. Topics include KV cache optimization, cache-aware routing, prefill/decode disaggregation, speculative decoding, and other emerging approaches used to improve performance and reduce infrastructure costs.Through practical examples and real-world architecture patterns, attendees will gain a deeper understanding of how to run open models efficiently at scale.

SESSION **Expo Stage 2 · Expo Stage 2 NW** 2:25pm-2:45pm

Improving AIE: Q&A and Feedback Session

swyx — Curator, Latent Space / AI Engineer

Join swyx for an open, informal Q&A and feedback session about the AI Engineer community, the World's Fair experience, and the ideas, pain points, and opportunities attendees are seeing in the field. Learn about AIE NYC, AIE CODE, AIE Europe, and AIE Asia next year and give us ideas on how to make the experience better for you. Bring questions, suggestions, and candid feedback for a conversation designed to help shape future AI Engineer programming and community initiatives. This feedback has REAL impact - for example, the New Engineer Orientation this year directly came from an attendee suggestion at AIE Europe! This session is repeated at 2:25 PM and 3:20 PM; attendees only need to attend one of the two sessions.

SESSION **Expo Stage 3 SW** 2:25pm-2:45pm

The Human Is an Async API

Melanie Warrick — Developer Relations Engineering, Temporal Technologies

Production agent systems need humans in the loop. So why do they keep getting modeled as synchronous tool calls? The agent ecosystem is focused on autonomy, but in reality, especially for high-stakes or regulated workflows, humans are a critical feature, not an afterthought. This demo-driven talk shows how to stop bolting on humans and start treating them as async-by-default endpoints with proper durability, retry, and escalation semantics. We will walk through two live, multi-agent patterns built with LangGraph and Google ADK, on Temporal for durable execution: The Agent Calls the Human. A fleet dispatch system escalates a disruption to an approver. We will intentionally kill the worker process mid-wait. Hours later, the human responds. State survives, and the agent resumes. The Human Calls the Agent. An operator interrupts a long-running task mid-flight to redirect it. The agent halts gracefully, surfaces state, accepts the override, and continues. Harness engineering has heavily focused on model autonomy. This talk is about the other half of the puzzle: the human. You will leave with two production-ready architectural designs you can apply this week: agent-initiated approval gates with timeout and escalation semantics, and human-initiated interrupts with graceful agent halt and resumption. Not every agent needs a human in the loop. But if you are building systems where the cost of being wrong exceeds the cost of being slow, this talk is for you.

2:50pm

SESSION **Harness Engineering · Main Stage** 2:50pm-3:10pm

No Memory, No Harness: Why the Database Is the Last Line of Defense

Kay Malcolm — Vice President of Product Management, Oracle AI Database, Oracle

The model is the easy part. Everything that makes an agent survive contact with production lives in the harness around it: orchestration, tooling, governance, and the memory core that keeps the system grounded when the model itself is probabilistic, forgetful, and non-deterministic. This talk walks the surface areas of an agent harness and consolidates the lessons we're learning as we ship them, from agentic applications in their current form (autonomous systems that now build their own automations) to the continual-learning loops that let agents improve from their own experience. We'll look at how the discipline is segmenting. AI application development is no longer one role but several: agent engineers, memory engineers, and platform engineers. We'll map Oracle's primitives onto each as the current state of harness engineering takes shape. We'll also examine the two populations betting on this stack at once, enterprise customers who need governance, reliability, and scale, alongside the cracked developers who need fast, composable primitives, and why a well-engineered harness serves both. And we'll make the case that has held through every shift in the stack: memory isn't a feature you bolt on, it's the foundation the rest of the harness stands on. The database remains the memory core, and when everything above it is probabilistic, it's the last line of defense.

SESSION **Generative Media · Track 1** 2:50pm-3:10pm

Infra behind Krea 2 - How to train and serve at scale

Gabriel Jorge Menezes — Core Infrastructure Engineer, Krea.ai

What do you need know about large scale pretraining and inference for GPUs. 1. Challenges of managing infra for pretraining 2. Weird problems we faced and how we fixed them 3. How to serve at scale with multiple clusters

SPONSOR **Agentic Commerce · Track 2** 2:50pm-3:10pm

Your Agent Just Authorized What?!

Jay Mok — PayPal

The nightmare scenario writes itself: your agent just ran off with your credit card and maxed it out on concert tickets, crypto, and a questionable NFT collection. Relax — we're building the guardrails. When an agent acts on your behalf, three questions must always be answerable: Did the human authorize this? Did they authorize this, now, in this scope? And can we prove it later? This talk maps three permissioning layers onto a stakes ladder: OAuth scopes at the bottom (broad capability, weak per-action proof, fine when reversible), Claude Code's tool-scoped allow/ask/deny model in the middle (brilliant for developer tooling, but no cryptographic evidence), and signed payment mandates at the top — where FIDO's Agentic Payments Working Group is building toward cryptographically-bound, constraint-carrying credentials. We'll share artifacts from Agent to Agent payments using our Shared Vault and Oauth to our constraint carrying Approval token leveraging our pillars of Identity and Buyer and Seller protection. You leave with a stakes × evidence matrix and a mental model that applies beyond payments: medical orders, e-signatures, securities trading, activities where you want you want to be more careful with your agent.

SESSION AI in Finance · Track 3 2:50pm-3:10pm

Simulation-Maxxing: How Nubank ships agents 20× faster with simulations

Shreya Rajpal — CEO, Snowglobe · Aman Gupta — Principal Machine Learning Engineer, Nubank

You know how to build an agent - write a prompt, spec out some tools and call an LLM (or gateway). At this point, you probably also know how to build an agent that “actually works” using some combination of agent frameworks, eval tools and looking at your data. This talk is about building an agent much, much faster using simulations to hill-climb your agent configuration instead of grinding on real data. We’ll dive deep into a case study of how a top-5 fintech made their agent dev cycle 20x faster using simulation-driven optimization. We’ll cover: - When to use real data vs. simulations in agent building - How to design simulation environments tailored to your agent - How to automate the optimization loop so you’re hill climbing agent configurations without manual tuning

SESSION Local AI · Track 4 2:50pm-3:10pm

Compression at the Edge

Chris Alexiuk — Sr. Product Research Engineer, NVIDIA · Daniel Han — Co-founder, Unsloth · Asma Beevi — Senior Engineer, NVIDIA · Merve Noyan — MLE, Hugging Face · Parth Sareen — Ollama

Compression at the Edge examines how smaller weights, faster inference, and constrained-memory deployments are making capable local AI more practical. The panel explores where compressed models already beat cloud on latency, privacy, cost, or control, what breakthroughs would unlock broader adoption, and how open model tooling is shaping the edge AI stack. Moderator: Chris Alexiuk (NVIDIA). Panelists: Daniel Han (Unsloth), Asma Beevi (NVIDIA), Merve Noyan (Hugging Face), Michael Chiang (Ollama).

SPONSOR Graphs · Track 5 2:50pm-3:10pm

On-Device Agentic AI for the New York Times Games

Shafik Quoraishee — Staff Engineer, The New York Times · Joanne Song — The New York Times Games

Traditional mobile game architectures rely on static state machines and fixed behavioral trees. Under this model, gameplay and accessibility are treated as rigid, separate systems. This results in blunt difficulty toggles, predictable character loops, and reactive features that fail to address a player's actual context. Constraint-Centric Agentic Simulation (CCAS) offers a theoretical shift. By modeling the game world as a continuous, multi-agent negotiation, accessibility and challenge become part of a single, fluid continuum. Using the JetBrains Koog framework on Android, this session explores the theory of running local agents on consumer mobile devices. We will discuss how principles of game theory, specifically dynamic negotiation and constraint satisfaction, can be used to build systems that reason over game states. Instead of executing pre-planned scripts, these agents dynamically alter their strategies. They negotiate environmental constraints to provide emergent challenges for high-skill players or organically smooth out cognitive and motor friction points for those requiring assistance. Running these theoretical models on edge hardware requires overcoming significant practical hurdles. We will break down the architecture needed to support this continuous adaptation without relying on cloud computation. We will cover how to manage memory footprints, compress state histories for rapid backtracking, and schedule local planning loops so they integrate flawlessly with the rendering engine.

SESSION AI in GTM · Track 6 2:50pm-3:10pm

How AI Agents Let GTM Teams Scale

Justin Joyce — Principal Sales Operations and Strategy Manager, Cloudflare

How Cloudflare scaled GTM with AI agents that never touch raw data: a deterministic layer computes the numbers, agents write the narrative, and a multi-agent pipeline turns every segment into ranked signals. Justin Joyce on the build — and what skill curation and adoption actually take.

SESSION AI in Healthcare · Track 7 2:50pm-3:10pm

How to build an AI-Native Health Company

Dan Feng — Senior Director of Engineering, Maven Clinic

Most healthcare technology companies were built for a different era. Transitioning to an AI-native organization isn't just about adopting new tools — it requires rethinking culture, processes, and how teams work at every level. This talk draws on firsthand experience leading that transformation at a digital health company. We'll cover what it takes to foster an AI-first culture across departments, and go deep on the engineering side: adopting AI-assisted development practices, building shared AI infrastructure, and evolving the product development process to unlock 2–3x productivity gains. We'll also tackle the harder, less-discussed challenge — the mindset shift required to operate effectively in a domain that's changing faster than any playbook can keep up with. Whether you're just starting this journey or already mid-transition, you'll walk away with concrete lessons on what works, what doesn't, and how to build an organization that compounds on AI rather than just experiments with it.

SESSION **Agentic Engineering · Track 8** 2:50pm-3:10pm

Realtime multiplayer, automation, and you!

Idan Gazit — Head of GitHub Next, GitHub

Now that the models are powerful and the agents are capable, why are we still approaching software development as if it's the same activity that it used to be, but "faster"? GitHub Next thinks about what this future wants to be through two lenses: - Automation: intelligence allows us to automate much more than we could with heuristics alone. How should that automation work? What guardrails do we have to put in place so that our CISOs allow us to do that? - Collaboration: agents can understand anything in your codebase, but what about all the facts that are in the heads of your teammates? Whether it's corporate politics or taste, how do we get the humans to leak that context where agents can see it and use it to produce better outcomes? Realtime multiplayer tools have displaced every turn-based tool out there. What should that look like for code? It's not going to be as simple as multiple cursors. Come by to hear more about what GitHub Next is learning about the changing shape of software creation — one that allows us to build better, not merely faster. One that allows us to scale up teams, not only individuals. And one where automations buy us time for craft and polish, not slop. We were promised flying cars, instead we have fifteen terminals. Let's have a nicer future than that.

SESSION **Inference · Track 9** 2:50pm-3:10pm

KV Cache-Aware Routing and P/D Disaggregation on Kubernetes: The Parts Public Benchmarks Don't Show

Yuchen Fama — Senior Principal Product Manager, Red Hat · **Ashish Kamra** — Senior Manager, Software Engineering, Red Hat

We're at the inflection point between classic LLM inference and agentic inference. When we look at the agentic workloads and trace replays, many core characteristics break classic LLM serving assumptions. The most consequential: the server no longer controls its own cache lifecycle. The client does, through prompt construction, multi-turn context that grows and changes each turn. This has downstream effects. Because context is client-determined, prefill strategy, eviction, and routing decisions move up to the scheduler layer. KV cache becomes volatile — frequent eviction and rewrite, driven from outside the engine. And latency becomes a first-class scheduling metric alongside throughput. This talk covers the open stack for LLM and agentic era inference serving: vLLM and llm-d. We begin with the core characteristics and challenges of agentic inference, then the economics: prefill dominates cost, and cache reuse is the primary lever. We explain why KV-aware routing through a fleet-wide scheduler is the first optimization to apply, ahead of adding capacity. Next, prefill/decode disaggregation. We separate compute-bound prefill from memory-bound decode, and examine what public benchmarks omit: the conditions under which P/D disaggregation shines, and the workload shapes that justify the added architectural complexity. We close with GLM-5.2 and show the equivalent stack assembled in the open: cache-aware routing, P/D disaggregation, tiered KV offload, and wide expert parallelism — implemented on vLLM and llm-d. Attendees leave with a tuning decision framework: which lever to apply first, how to read workload signals, and where additional GPUs do and don't help.

SESSION **AI-Native Enterprises · Leadership 1** 2:50pm-3:10pm

AI Agents Are Just Distributed Systems Now

Salman Munaf — Lead Site Reliability Engineer, TikTok

AI agents are often described as a new kind of software, but once they move beyond chat and start calling tools, reading data, making decisions, retrying tasks, and coordinating workflows, they begin to look a lot like distributed systems. They have state. They call external services. They depend on APIs. They fail partially. They retry. They time out. They can loop. They can act on stale context. They can produce inconsistent results. And when something goes wrong, teams need logs, traces, permissions, ownership, and rollback paths just like they do with any other production system. This session will give engineers a practical way to reason about AI agents using familiar distributed systems concepts. We will break down the agent loop: planning, tool use, observation, memory, and retries. Then we will map common agent failure modes to engineering patterns teams already know, including timeouts, circuit breakers, idempotency, rate limits, least privilege, observability, and human approval. The goal is to move past the hype and treat agents like real production systems. Attendees will leave with a clear mental model for designing, debugging, and operating agents safely, especially as they become part of customer-facing products, internal developer tools, and business workflows.

SESSION **AI Architects: AI Factories · Leadership 2** 2:50pm-3:10pm

Inside 847 Production Clinical AI Notes

Sebastian Fox — CEO, Composo

A Series B clinical AI company had an ambient scribe in production for six months. Internal evals passed every release. A clinical team spot-checked a sample weekly and saw nothing alarming. The system had healthy NPS, expanding deployments, and the company was preparing for European market expansion. We ran a structured audit on 847 production notes. Found 127 failures across six categories. 23 were severity-critical - the kind that could directly alter a clinical decision. The team's existing LLM-as-judge had reported zero failures across the same notes. This talk is the engineering forensics of that audit. The audit setup: which production traces we sampled, how the structured failure-mode coding worked, and the reviewer protocol. The results: three dominant failure clusters - decision-status corruption (19 cases), structured omissions (34 cases), and dosage substitution (12 cases) - and the underlying generation pattern behind each. For each cluster I will show: a real anonymised trace, the eval rule that should have caught it but did not, an explanation of why the eval missed it, and the criterion that does catch it. The pattern that emerged in the data is engineering-actionable. The team had built a 20-criterion content-faithfulness eval layer. The failures lived underneath it, in a missing intent layer. We replaced the broad content layer with a five-criterion intent layer (decision status, omission impact, dosage integrity, diagnostic chain, laterality consistency). Detection rate went from 0% to 96% on the failure set. Compute cost dropped because the intent layer is cheaper to run than the content layer it replaced. You will leave with a forensics protocol for auditing your own production AI, the five intent criteria that generalise to any high-stakes domain, and the architectural pattern: build a thin intent layer, not a thick content layer.

SESSION **Expo Stage 1 NE** 2:50pm-3:10pm

Harness Engineering: The New Core Skill for Agentic Developers

Dru Knox — Head of Product, Tessl

Harness engineering is emerging as a new core competency for agentic engineers. Your job isn't writing good code, it's upgrading your codebase so that agents reliably succeed. This talk covers the core loop of harness engineering, the most common codebase modifications you'll make, and how to 10x your harness engineering efforts with Tessl's harness engineering agent.

SESSION **Expo Stage 3 · Expo Stage 3 SW** 2:50pm-3:10pm

Small Claws Are Beautiful: Edge Agents with NanoClaw, Raspberry Pi, and Graph Memory

Jeremy Adams — Tech Translator, Neo4j

SESSION **Expo Stage 4 SE** 2:50pm-3:10pm

The Software Factory

In the leading engineering organizations, a single engineer now supervises teams of agents, migrations scoped for years close in weeks, and code review has become the tightest constraint in the system. The teams pulling ahead are operating a software factory: an integrated system of agents that share context across the entire SDLC. This session is a field guide to that operating model and how it runs at scale: what each stage looks like in practice, what shifts for engineers as they move from writing code to stewarding the system, and the hard truths that decide whether a factory compounds, starting with why the infrastructure you built for humans sets the ceiling on what agents can do.

3:20pm

SESSION **Harness Engineering · Main Stage** 3:20pm-3:40pm

How we Solved Agent Building

Andrew Qu — Chief of Software, Vercel

At Vercel I've built a successful AI data scientist, that has taken the load off of our data team from answering ad-hoc data queries, and fields over 1,200 unique queries a day from just internal Vercelians. I've been building and iterating on it since last september, and it's gone through over 6 different rewrites, the newest one of which has inspired us to build a new agent framework (to be teased during the talk ;)). I'd talk about why we build agents, how we build agents, and how to build effective agents in today's world. Just prompting, to adding bespoke tooling, to embedding claude code, to file system agents, to skills-based agents, to the new agent harness framework.

SESSION Generative Media · Track 1 3:20pm-3:40pm

The Next Medium: Why Real-Time Interactive Video Changes Everything for Developers

Ahmed Ahres — Head of Product & GTM, Reactor

Every major platform shift created a new category of developers. The web created web developers. Mobile created app developers. Now real-time interactive video models are creating a new kind of builder: one who does not render scenes or script interactions, but writes code that shapes a living world as it generates. This talk explores what it means for video to become a runtime, why this moment is happening now, and what the first generation of developers building on world models are already creating. Based on work at Reactor, where developers are shipping interactive games, robotics simulations, and real-time experiences that could not have existed 1 year ago.

SPONSOR Agentic Commerce · Track 2 3:20pm-3:40pm

The End of the Static Screen: Architecting Intent-Driven UX with Agentic Orchestration

Gus Iwanaga — General Manager (Product, UX, Eng), commercetools

For 30 years, interfaces were designed ahead: wireframes, fixed flows, pre-built dashboards - because we couldn't make them otherwise. Three shifts changed the constraint: LLMs that reason over business context, agentic frameworks that work at production grade, and composable backends that expose a real tool surface. With all three in place, the interface stops being something you design and ships as the output of an orchestrator composing it per intent. I'll walk through the hypothesis, the architecture we're running in production for enterprise commerce, and a live demo where it all moves.

SESSION AI in Finance · Track 3 3:20pm-3:40pm

Skills are new features: Building Skill-Centric Harness for Agentic Products

Yogendra Miraje — Principal AI Engineer, FactSet

SESSION Local AI · Track 4 3:20pm-3:40pm

Model Routing

Nader Khalil — Director of Developer Technology, NVIDIA · Walden Yan — Co-founder & CPO, Cognition · Tanay Varshney — Principal Engineer, NVIDIA · Alex Atallah — Co-founder & CEO, OpenRouter

Model Routing explores how teams decide when to use local models, open-source models, or frontier cloud systems, and why the answer is increasingly hybrid rather than one-size-fits-all. The panel digs into routing architectures, model selection strategies, stack decisions, and what still needs to improve in local AI before more workloads can move closer to the user. Moderator: Nader Khalil (NVIDIA). Panelists: Walden Yan (Cognition), Tanay Varshney (NVIDIA), Alex Atallah (OpenRouter).

SPONSOR Graphs · Track 5 3:20pm-3:40pm

Citation Needed: Provenance for LLM-Built Knowledge Graphs

Daniel Chalef — Founder and CEO, Zep AI

An LLM doesn't copy facts into your knowledge graph. It synthesizes them: entities merge across sources, and later data invalidates earlier facts. By the time your agent retrieves "patient has a penicillin allergy," the origin — an EHR record, a lab report, or something typed into a chatbot — is gone. This talk covers engineering lineage into a lossy, generative pipeline: episode-to-fact links as structural graph properties, provenance that survives entity resolution, metadata projection (tag a source once; it follows every derived node and edge), and the query semantics of filtering facts by ancestry, including mixed-trust parentage. Deletion is the inverse problem: GDPR erasure propagates back through the same derivation edges. Compliance gets an audit trail; engineers get agents they can debug instead of black boxes.

SESSION AI in GTM · Track 6 3:20pm-3:40pm

Building GTM AI Agents: Lessons from Deploying to 6,000 Users

Sait Izmit — Principal Product Manager, Snowflake

Building an enterprise AI agent for GTM teams isn't just an LLM problem—it's a product, engineering, and adoption challenge. In this session, I'll share how we built and scaled Snowflake's internal GTM AI Assistant from MVP to a production system serving more than 6,000 employees and answering over one million questions. We'll cover how we scoped the MVP, evolved the architecture over time, balanced quality versus coverage, adopted emerging technologies like MCP, and continuously adapted as the AI landscape rapidly changed. You'll leave with practical lessons for building enterprise AI products that users actually trust and use.

SESSION **AI in Healthcare · Track 7** 3:20pm-3:40pm

Don't be data poor

Anuj Iravane — Head of AI, Anterior

What do you do when the data you most need to train and evaluate on is the data you're least allowed to keep? It's a bind for anyone building AI in a high-stakes vertical: the cases that would teach your model the most — the rare, the messy, the sensitive — tend to be the ones wrapped in the tightest constraints. In healthcare it's near-absolute. PHI can't be retained, reused, or transformed, so your long-lived datasets can't contain real patient data at all. Synthetic data is the obvious escape hatch, but it has its own trap: synthetic records tend to look synthetic, and a model that passes on fake-looking data tells you nothing about the real thing. So the bar isn't generating data — it's generating data faithful enough to trust. This talk is how we got there. Ask an LLM for a full case in one shot and you get something generic and averaged-out — models are worse at inventing convincing, specific detail than you'd expect. We present our synthetic generation pipeline (and the process around it) that enabled us to create golden datasets at scale. The pipeline features a coarse-to-fine process that enriches a patients medical history layer by layer, with a human in the loop hooks to steer the narrative at each step. You'll leave with ideas on how to build your own synthetic data generation capabilities and how to build a data pipeline your domain experts actually enjoy owning.

SESSION **Agentic Engineering · Track 8** 3:20pm-3:40pm

Velocity Sickness: What Happens When Your Whole Team Gets 10x Faster

Matt Dailey — Founder, Ref.

Learn more about Ref: https://ref.tools/ AI made writing code nearly free, and on most teams, that's quietly breaking how the team works. Individually, everyone feels ten times faster. Together, the signals point the other way: too many PRs moving in too many directions, engineers throwing away whole agent sessions and starting over ("declaring agent bankruptcy"), and critical decisions getting made inside agent chats that no one will ever see or review. There's a lot of energy, and it's all going somewhere different. I call this velocity sickness: the organizational pain that comes from individual speed. It's the engineering version of an author who ships a book a week: prolific, productive, and completely unreadable by the team that's supposed to build on it. Almost every conversation about AI coding is about making one engineer faster. This talk is about what happens to the team when all of them are. Once implementation stops being the bottleneck, the hard part isn't writing the code. It's tracking it, reviewing it, and keeping a hundred parallel decisions coherent. That's the problem eng leaders are actually being handed, and it's the one this session takes on directly. Engineering has always had three phases: plan, implement, polish. AI collapsed the middle one to almost nothing, so the leverage, and the real work, move to the decision-heavy ends. The fix isn't better prompts; it's changing what our tools treat as first-class. We have to split the decision layer from the implementation layer: humans spend their time at the decision layer, reviewing and making the choices that matter, while agents handle the implementation. That means durable, reviewable plans, not ephemeral chats. Review the decisions before you review the diff. What attendees will leave with: - A mental model for plan / implement / polish and why the decision layer is now where engineering leverage lives, plus the language to explain velocity sickness to their own team. - A concrete shift: how to pull your team's important decisions out of throwaway agent chats and into a shared, reviewable source of truth, so individual speed compounds into team cohesion instead of chaos.

SESSION **Inference · Track 9** 3:20pm-3:40pm

Two Bugs That Hid in Plain Sight: A vLLM Debugging Detective Story

Asaf Gardin — Senior Software Engineer/Inference Engineer, AI21 · **Yuval Belfer** — Sr. Developer Advocate, AI21

Your model generates gibberish. Once every thousand prompts. High confidence scores. No crashes. No warnings. We hit this twice while building Jamba models. First: A request gets misclassified during scheduling, loads stale state from a previous prompt cache slot, and confidently generates nonsense. Second: Logprob spikes during RL training that looked like training instability-until we noticed they tracked with rollout count, then with cache size. In this talk, we'll walk through both debugging journeys-the false starts, how we instrumented vLLM to thread request IDs through the forward pass, the search for variables that change failure structure rather than magnitude, and the lesson both share: distributed inference systems fail silently. No stack trace. No sanitizer warning. Just wrong answers with perfect confidence. You'll learn how to build comparison scripts that expose logprob divergence, force memory pressure to surface rare bugs, and shrink a distributed RL training mystery into a reproducible single-script failure. Walk away knowing how to debug vLLM when it lies to you quietly.

SESSION **AI-Native Enterprises · Leadership 1** 3:20pm-3:40pm

The Signal Layer: What to Build When Anything Can Be Built

Lena Hall — Senior Director Developers and AI, Akamai

AI has made implementation faster, cheaper, and more widely available. That changes the real bottleneck in software. When every team can generate code, spin up agents, prototype workflows, and ship demos faster than ever, the advantage moves to a different layer: knowing what is worth building, who it is for, how people will discover it, and how the product should behave once they do. This talk introduces the Signal Layer: the system of public signals, user intent, agent experience, distribution loops, and product judgment that helps builders decide what deserves to exist before they commit time, infrastructure, and trust to building it. We will look at how AI changes the software lifecycle from “can we build it?” to “should this exist?” and how developers, AI engineers, and technical leaders can design products that earn adoption instead of producing impressive demos that disappear. When anything can be built, the most valuable builders are the ones who can read signal early, shape the right experience, and build the thing users were already moving toward.

SESSION **AI Architects: AI Factories · Leadership 2** 3:20pm-3:40pm

Give the Agent a Budget, Not a Token

Sachin Malhotra — Member of Technical Staff, Anthropic

Every agent demo runs with a god-token. Then it ships, and someone has to explain why the helpful AI just rm -rf'd the staging database "to clean up." I run platform infrastructure at a frontier lab, and for the last year my job has partly been: let coding agents do real work against real systems, without ever having to write the postmortem. This talk is the permission model that fell out of that - not RBAC-with-extra-steps, but primitives designed for an actor that's smart, fast, tireless, and occasionally *confidently wrong*. **The four primitives:** - **Asymmetric verbs** - the agent can `quarantine` but not `delete`, `retry` but not `approve`, `propose` but not `merge`. The verb list *is* the security boundary. Stop thinking in resources, start thinking in reversible vs. irreversible actions. - **Regenerating budgets** - every agent identity gets N disruptive actions per window. Burn the budget, you're benched until it refills. No human-in-the-loop until the budget's gone — which means 95% autonomy with a hard ceiling on blast radius. - **The undo test** - if the agent can't undo it, the agent can't do it without a second key. One line, surprisingly load-bearing. - **Tripwires over allow-lists** - let the agent roam, but instrument the three actions that would actually hurt. Cheaper than enumerating everything safe. I'll show the ~200-line policy layer that implements all four, the failure modes each one exists to catch, and the one design I shipped that turned out to be security theater. Tool-agnostic - works whether your agent is touching CI, a database, a cloud account, or your users' files. If you're shipping an agent that does anything more than read, you'll leave with a threat model and a starting policy you can paste into your repo on the flight home.

SESSION **Expo Stage 1 NE** 3:20pm-3:40pm

Agent Memory Is a Solved Problem. Agent Learning Is Not.

Karthik Ranganathan — Co-founder and Co-CEO, Yugabyte · Heather Downing — Developer Advocate, Yugabyte

The failures that break multi-agent systems are not reasoning failures, they are handoff failures. One agent works something out and the knowledge dies in its private context, because the only thing that crosses the boundary is output. Memory made each agent better in isolation and changed nothing about what the group knows. The missing primitive is supervised promotion: a deliberate decision about which private learning is worth sharing, moved into common knowledge with the reasoning attached, so trust survives the handoff. Today a human makes that call, and promoted knowledge resolves on read, in any tool, with no retrain or reindex. Those calls are also the training signal for what comes next: orchestrator agents, trained on what matters to the people they serve, that promote on their own. This talk covers how our collective knowledge grew as we approached memory promotion, including what the first build got wrong, and a live look at it working between humans and agents.

SESSION **Expo Stage 2 · Expo Stage 2 NW** 3:20pm-3:40pm

Improving AIE: Q&A and Feedback Session (repeat)

swyx — Curator, Latent Space / AI Engineer

Join swyx for an open, informal Q&A and feedback session about the AI Engineer community, the World’s Fair experience, and the ideas, pain points, and opportunities attendees are seeing in the field. Learn about AIE NYC, AIE CODE, AIE Europe, and AIE Asia next year and give us ideas on how to make the experience better for you. Bring questions, suggestions, and candid feedback for a conversation designed to help shape future AI Engineer programming and community initiatives. This feedback has REAL impact - for example, the New Engineer Orientation this year directly came from an attendee suggestion at AIE Europe! This session is repeated at 2:25 PM and 3:20 PM; attendees only need to attend one of the two sessions.

SESSION **Expo Stage 3 · Expo Stage 3 SW** 3:20pm-3:40pm

An Interaction Is All You Need

Ivan Leo — Developer Experience Engineer, Google DeepMind

SESSION **Expo Stage 4 SE** 3:20pm-3:40pm

An AI Future Without the Lock-In

Remy Guercio — Strategic Projects, Tailscale

Every organization navigating AI adoption faces the same trap: the market moves faster than any procurement cycle, no single vendor leads across model quality, interface, sandbox, and data access for more than a few months at a time, and the obvious answer of consolidating behind one platform trades short-term control for long-term lock-in. This session makes the case that the winning strategy is not picking the best walled garden. It is building a connective layer underneath all of them. Tailscale's Remy Guercio walks through the four components required for transformative AI, why vertically integrated stacks are structurally fragile, and how organizations can maintain visibility and control without betting on a single vendor's continued dominance. The second half of the session covers three new capabilities in Aperture, Tailscale's identity-aware AI gateway: Identity-Aware Universal Data Connectors (Public Alpha), which translate Tailscale network identity into scoped access to internal data sources via MCP and API endpoints; a Responsive Chat UI (Public Alpha) that gives non-technical users a mobile-friendly interface to every LLM configured in Aperture; and Sandbox Support (Private Alpha), bringing ephemeral and persistent compute environments into the same identity model. Attendees leave with a framework for evaluating AI platforms that does not depend on picking a winner, and a concrete path to deploying provider-agnostic AI tooling on infrastructure they already run.

3:45pm

SESSION **Harness Engineering · Main Stage** 3:45pm-4:05pm

Agents Without Code: How Skills, YAML, and Filesystems Replaced Python

Philipp Schmid — Staff Engineer, Google DeepMind

Six months ago, building an agent meant writing a Python class with a `while` loop, tool definitions in dicts, manual state management or writing custom python functions. Today, you define an agent in a YAML file, drop a `SKILL.md` into a folder, and deploy. This talk traces the arc from "Agent in Python" to "Agent as filesystem". You'll learn the same agent built three ways: the hard way (Jan 2025), the simple way (Oct 2025), and the zero-code way (today).

SPONSOR **Agentic Commerce · Track 2** 3:45pm-4:05pm

Beyond the Lethal Trifecta: Agentic Commerce on the Open Internet at Machine Speed

David Levine — Founder & CEO, Kiduna Club

For decades, the internet has had protocols for routing, identity, encryption, payments, and commerce between people and organizations. It has never had a native way for autonomous agents to possess authority, accountability, or legal standing. On July 1, 2026 that changes. A little known law will take effect that changes the world as we know it. As AI agents move beyond the enterprise firewall, a new form of commerce is emerging. Agents can already search, negotiate, schedule, purchase, settle payments, and coordinate work across networks. But the moment they begin acting independently on behalf of people, businesses, and online organizations, fundamental questions appear: Who does this agent represent? What authority does it possess? Who is responsible when something goes wrong? How do counterparties know they can trust it? This talk explores the "Lethal Trifecta" of agentic systems: access to systems, access to networks, and autonomy. Together they create extraordinary capabilities, but they also expose a missing layer in the architecture of the internet itself. Without identity, accountability, governance, and legal standing, agentic commerce remains trapped inside enterprise walls, limited to productivity gains rather than participation in open markets. On the same day as this conference, a new legal framework takes effect that gives autonomous online organizations a registered legal existence, allowing them to hold assets, enter agreements, govern themselves through software, and operate through fleets of agents. Whether you're building agents, agent platforms, autonomous organizations, payment systems, governance systems, or the next generation of internet infrastructure, this shift has global implications, and you'll be the first to know. We'll examine the emerging trust stack for agentic commerce—identity, authority, governance, settlement, and standing—and explore what happens when agents stop acting merely as tools and begin participating as economic actors on the open internet at machine speed.

SESSION AI in Finance · Track 3 3:45pm-4:05pm

Wearing the Agent: Engineering a Family-and-Friends Personal Agent, from Group Chats to Glasses

Sai Krishna Rallabandi — Director, Data Science, Fidelity Investments

Judith is a personal AI agent that has run in daily production for a year, used by more than a dozen of my family and friends across three WhatsApp group chats, Telegram, and Discord. This talk walks through how it's built, in two parts. The first part is the engineering that makes one agent safe for many people to share: a multi-tenant permission model (read-only for my mom, exec for me), a memory stack — FAISS + Neo4j + curated long-term notes — that stays useful over a year instead of bloating into noise, cron-scheduled subagents that scout and act on their own, and the guardrails it enforces on every message — redact personal info before posting to a group, never reply to the wrong person, and screen attacker-controllable text for prompt injection before acting on it. The second part takes the agent off the screen and onto a $50 pair of smart glasses. It captures what I see, describes and stores it as a running visual memory, sets destination path on maps before I get onto car, finds and tells me which aisle in the store to go to first, etc. I cover the latency budget that keeps it conversational — on-device Whisper for speech, cloud reasoning, sub-one-second round trips — and the custom neural voice it speaks in rather than stock TTS, drawn from my speech-synthesis background. Both parts are shown live, including a candid look at the pieces that don't work yet. Audience takeaways: A multi-tenant architecture for a personal agent multiple people actually share A memory design that survives real long-term use (not just a vector store) A defensive checklist for any agent that ingests untrusted text A blueprint for an ambient, vision-aware wearable interface on commodity hardware, with a real latency budget

SESSION Local AI · Track 4 3:45pm-4:05pm

Model Routing

Nader Khalil — Director of Developer Technology, NVIDIA · Walden Yan — Co-founder & CPO, Cognition · Tanay Varshney — Principal Engineer, NVIDIA · Alex Atallah — Co-founder & CEO, OpenRouter

Model Routing explores how teams decide when to use local models, open-source models, or frontier cloud systems, and why the answer is increasingly hybrid rather than one-size-fits-all. The panel digs into routing architectures, model selection strategies, stack decisions, and what still needs to improve in local AI before more workloads can move closer to the user. Moderator: Nader Khalil (NVIDIA). Panelists: Walden Yan (Cognition), Tanay Varshney (NVIDIA), Alex Atallah (OpenRouter).

SPONSOR Graphs · Track 5 3:45pm-4:05pm

Why We Killed Our Multi-Agent Pipeline: Lessons From Pharma Commercial Intelligence

Subbiah Sethuraman — Partner, ZS Associates · Abhilash Asokan — ZS

Key takeaways: A practical design principle for agentic systems in regulated, high-stakes domains: derive the architecture from agent behavior, don't impose it. Concrete patterns the audience can apply this week — domain knowledge graphs as agent context, deterministic preprocessing as a complement to agentic reasoning, reference-based context management. An honest case study from production: what worked, what didn't, and the open architectural questions we're still working on. Abstract : We lead the architecture and AI engineering org behind ZS Associates' commercial intelligence platform for pharmaceutical brand teams. The product has two surfaces: a proactive alert system that delivers signal-driven intelligence packets when a brand's KPIs move, and a conversational analytics chat where business users ask ad-hoc questions. A year ago we built both surfaces as separate V1 stacks. They broke in different ways. The diagnosis was the same: we had decided on the structure before we knew what the agent actually needed. This talk is about the design principle that came out of rebuilding both — and what it produced. The architecture is derived, not designed. We stopped trying to predict what scaffolding the agent would need and started designing the system around what the agent's behavior, on real production tasks, actually demanded. Tools, context, structure, and guardrails get introduced at the points where the agent's reasoning needs them — and nowhere else. What that produced is an architecture that's smaller than V1, not bigger. A single agent owns each investigation end-to-end across both surfaces, launching parallel sub-agents when the work needs them — not according to a pre-defined topology. A pharmaceutical commercial knowledge graph — HCPs, accounts, payers, territories, brands, KPIs and the relationships between them — gives the agent the domain context it needs without prompt-engineering heroics. Statistical signal detection runs deterministically before the agent wakes up, so the agent's job is to explain signals, not find them. Raw query results stay out of the context window through a reference-pattern that lets the agent reason over data without drowning in it. Each of those decisions came from watching an agent struggle on a real task and asking what does it need here? — not from sketching the architecture in a doc and forcing the agent into it. The patterns generalize. If you're shipping agents over messy enterprise data — finance, supply chain, claims, operations — the failure modes and the fixes will look familiar. We'll close with the open questions and the pieces we haven't solved yet.

SESSION AI in GTM · Track 6 3:45pm-4:05pm

The Death of Developer Advocates

Stephanie Jarmak — Agent Advocate, Sourcegraph

Developer Advocacy is dead. Over the last decade Developer Advocates have been a key part of any devtool company. Coding agents are the customer now. Your ICP is Claude Code, Codex, and a myriad of other coding agents that are going to evaluating, using, and suggesting tools to their human counterparts, then implementing them. So what do you do about it? Pivot to "Agent Advocates". This is a similar role but with the expressed purpose of understanding how Agents experience your product and using those findings to improve the agent experience. In this talk/workshop I'll share how to evaluate the agent experience of your product, how to improve it, and how to communicate that to your team so they can change the products roadmap.

SESSION AI in Healthcare · Track 7 3:45pm-4:05pm

Why Your Enterprise Tech Stack Isn't Ready for AI Agents - And What to Build Instead

Christopher Lovejoy — Member of Technical Staff, Anthropic · Saul Howard — VP Engineering, Anterior

Agent-executed work is a new infrastructure primitive. Until you treat it that way, you're running a demo, not enterprise AI. Your existing stack was built for deterministic software. Agents reason, delegate, and make judgment calls. That distinction creates infrastructure problems most engineering teams haven't confronted: security vulnerabilities baked in by design, no audit trail, no explainability, no human-in-the-loop. At Anterior, we've deployed clinical AI agents across many of the largest US health plans, covering 50 million lives. Healthcare, with high stakes, strict regulation, deeply human workflows, exposes infrastructure gaps that exist everywhere - and makes the paradigm shift unavoidable: agent-executed work as a first-class primitive, alongside compute, storage, and APIs. We'll cover why bolting agents onto existing data pipelines fails, what infrastructure primitives are missing (and why teams don't notice until an audit), and how to architect a stack where security, compliance, and human oversight are load-bearing from day one. If you're serious about agents in any mission-critical context, this is the infrastructure conversation you need to have.

SESSION Agentic Engineering · Track 8 3:45pm-4:05pm

Open Source Is Dead. Long Live Open Source.

Saoud Rizwan — Founder & CEO, Cline

Closed model labs set take-it-or-leave-it prices, but open-weight models force inference hosts to compete on the same models, driving costs down and shifting power back to builders instead of vendors. I'll tell the story of how Cline went from viral open source project to a case study in AI-generated slop, entitled PRs, and brand-diluting forks and why, even as that old idea of open source community died, open weight models and auditable code are now the only real check we have on model pricing and control.

SESSION Inference · Track 9 3:45pm-4:05pm

Weight Folding, CUDA Streams, and the Bug That Made My Model Speak Backwards

Filip Makraduli — Founding Member of Technical Staff, Superlinked

A talk about contributing GPU benchmarks to an open-source research paper (FlashNorm). I'll walk through the engineering journey: folding norm weights into projections, writing Triton kernels, accidentally making attention bidirectional (oops), and ultimately proving a 33-35% speedup on the norm+project operation. Practical lessons for anyone trying to optimize transformer inference.

SESSION AI-Native Enterprises · Leadership 1 3:45pm-4:05pm

Tell the Robot What You Want

Sandhya Subramani — Senior Developer Advocate, GenAI, Amazon Web Services

What if you could command a robot just by talking to it? This session introduces Strands Agents, an open-source framework that lets developers control physical sensors and actuators using natural language by exposing hardware capabilities as programmable agent tools through a unified interface. The agent interprets the request, selects the appropriate tools, and orchestrates execution. We explore a hybrid model where low-latency perception and actuation run locally on edge hardware, and higher-level reasoning and multi-step planning are delegated to cloud-based agents when needed. This preserves real-time responsiveness while enabling richer reasoning. A live robot demonstration anchors the session. Using the EarthRover Mini, attendees will see how an instruction such as “drive toward the doorway and describe what you see” moves from conversation to perception, tool selection, bounded motion commands and real-world physical action.

SESSION Expo Stage 1 NE 3:45pm-4:05pm

Taking Reinforcement Learning Cross Datacenter

Nan Jiang — MTS, Modal

Reinforcement learning for frontier models is increasingly constrained not only by algorithms, but by where compute is available. When training and rollout generation must live inside one datacenter, the whole system becomes limited by the capacity, hardware, and failures of that single location. Taking RL cross datacenter changes the shape of the problem. Training can happen in one place, Rollout trajectories can be generated somewhere else, and compute can be pulled from whatever cloud, region, hardware, or precision format is available. RL capacity can become global, elastic, and opportunistic rather than a carefully reserved supercomputer, more like a living system spread across the world. This talk is about the first steps toward that future: RL that can run anywhere, learn continuously, and turn scattered compute into a single training loop.

SESSION Expo Stage 2 NW 3:45pm-4:05pm

Dashboards are Dead

Sarah Simionescu — Composio

AX is the new UX, and how to build for agents.

4:30pm

KEYNOTE Main Stage 4:30pm-4:50pm

Closing Keynote — Theo Browne

Theo Browne — Founder/YouTuber, T3 Tools & YouTuber

4:50pm

KEYNOTE Main Stage 4:50pm-5:10pm

Closing Keynote: Garry Tan

Garry Tan — President & CEO, Y Combinator

5:10pm

KEYNOTE Main Stage 5:10pm-5:30pm

Startup Battlefield

Howie Liu — CEO, Airtable · Joshua Xu — CEO, HeyGen · swyx — Curator, Latent Space / AI Engineer

Schedule v4945 · exported July 2, 2026. Tap any talk title to open it on the live interactive schedule. Schedule is subject to change; sessions marked “tentative” are not yet confirmed.