MCP Dev Summit Bengaluru 2026 - June 9 Livestream

124 MIN · YOUTUBE · HTTPS://WWW.YOUTUBE.COM/WATCH?V=HH16IAVV9B0
https://www.youtube.com/watch?v=Hh16iAVv9B0

SUMMARY

The transcript features a mix of musical elements and a series of keynote speeches from the MCP DevSummit in Bangalore, focusing on the evolution of the MCP protocol and its implications for the AI ecosystem. Key speakers discuss the growth of the Agentic AI Foundation, the importance of community engagement, and the need for interoperability in AI systems.

- *The MCP protocol has seen significant growth, with SDK downloads nearing 100 million and widespread adoption across various companies.*
- *A new release candidate for MCP aims to simplify operations by transitioning to a stateless protocol, removing the need for session management.*
- *Key features being deprecated include logging and sampling, with a focus on integrating with existing observability platforms.*
- *The Agentic AI Foundation emphasizes community-led development and encourages participation in working groups to drive innovation.*
- *The importance of machine experience (MX) over user experience (UX) is highlighted, with a shift in focus from human-centric design to agent-centric design.*
- *Trust and reputation in AI systems are critical as the ecosystem expands, necessitating better governance and identity management.*
- *The Gates Foundation and other organizations are applying AI innovations to real-world problems in healthcare and agriculture, emphasizing the need for tailored solutions.*
- *The event concludes with a call to action for attendees to engage in building and contributing to the evolving AI landscape.*

[music] My heart my heart for you.

Uh-huh. [music] [music] >> [music] >> My heart.

[music] >> [music] [music] >> For you ah [music] for you.

>> [music] [music] >> My heart my heart be I for you.

>> [music] [music] >> I want to feel Fire.

[music] Ah. for you. [music] Fire.

>> [music] [music] >> Hearts be heart. My heart dips. >> [music] >> Fire.

My heart. [music] [music] >> [music] >> My heart is your heart.

[music] for you. Fire. [music] >> [music] [music] >> Heartiph My heart.

[music] [music] [music] >> [music] >> It's my heart.

for you. Uh-huh. [music] Are you alive? [music] [music] My heart beh my heart in my you [music] [music] [music] know >> [music] >> I can't [music] be happy.

Ah ah for a for you. [music] [music] >> [music] >> Ohip be Ohip My heart be my heart.

[music] >> [music] >> I can't. [music] for [music] your love.

>> [music] >> Ohip dip dip dip dip dip dip dip dip dip dip. [music] I for you. [music] >> [music] [music] >> I want your heart.

[music] >> [music] >> Ah, a fire. My heart [music] for you.

[music] >> [music] [music] [music] >> heart.

My heart be [music] my heart. [music] >> [music] [music] >> My heart.

[music] Ah, [music] for you.

>> [music] >> for you. [music] Fire.

>> [music] [music] >> My heart beep beep. My heart beep. [music] Fire you fire.

[music] I feel [music] for you.

[music] >> [music] >> Fire. [music] [music] Heartip heart my heart >> [music] [music] [music] [music] >> I can't [music] ah Uh-huh.

for you. [music] Are you [music] [music] [music] My heart [music] my heart you >> [music] >> Are you [music]

crazy?

>> [music] [music] >> Ah, [music] for you. for you. [music] >> [music] [music] >> My heart be my heart.

fire. [music] Fire. [music] [music] I feel [music] deep.

>> [music] >> Fire. [music] Heartip [music] my heart [music] fire Oh, [music] hey.

[music] Fire. [music] Hey. Hey.

Ah [music] ah you ah for [music] your >> [music] >> I think I My heart [music] my heart [music] for you.

[music] >> [music] [music] >> for your heart. [music] Ah, for real.

>> [music] >> for [music] [music] your love.

Heartip [music] dip. My heart. [music] [music] [music] I feel [music] ah for you.

[music] >> [music] >> Fire.

My heart be my heart. >> [music] [music] [music] >> Fire.

Ah. [music] Ah fire [music] for you. [music] [music] Oh yeah.

Uh-huh. My heart. [music] Are you [music] fire?

Ah, [music] [music] hey. Ah. My heart. [music] [music] [music] [music] >> [music] >> Heartip dip.

Oh, I fire. [music] [music] [music] I feel [music] for you.

[music] Fire. [music] [music] My heart [music] be my heart.

[music] Oh, hey. [music] [music] [music] You're [music] a Ah, a fire.

[music] [music] >> [music] >> Fire.

[music] My heart [music] fire you. [music] >> [music] [music] [music] >> by your Ah, ah for you.

for [music] [music] my love.

My heart my heart be Fire. [music] [music] Hey. Hey.

I feel [music] ah for you. [music] >> [music] >> Fire.

[music] My heart my heart for you.

>> [music] [music] [music] [music] >> It's Are you [music] [music] happy?

[music] Fire. [music] Are you alive? [music] My heart beeps.

My heart dips. [music] Fire. [music] Fire. >> [music] [music] >> I think you're >> [music] >> Ah, for you.

>> [music] >> Fire. [music] Hey. Hey.

Heartip dip dip heart dip. I know I Fire. [music] [music] >> [music] [music] >> for you.

[music] Hey [music] >> [music] >> Hey.

>> [music] >> My heart >> [music] >> be making Please welcome Angie Jones, vice president of developer experience at the Agentic AI Foundation. [music] >> [music] >> Good morning. Hello.

So good to see you all. Welcome, welcome, welcome to MCP DevSummit Bangalore. We are so thrilled to have you all here. You all look amazing out there. What an amazing crowd. So we have MCP contributors and builders from all over the ecosystem. And today and tomorrow are going to be full days of learning, sharing ideas with one another. Don't be shy. Meet a friend, meet a stranger, share what you're working on. All of us are trying to figure this stuff out together, okay?

That's why everyone came today. So, we have an incredible lineup of speakers for you all today. So, get ready to be inspired. Speaking of, I want to take a moment to thank my wonderful program committee who reviewed hundreds of applications to speak at this conference and they had a very very tough job of selecting the program. So, uh I think that they did an amazing job.

Did you all like the schedule? Yes. All right. Give them a round of applause. Thank you to the program committee. Also, very special thank you to our diamond sponsor AWS who helped to make this event possible. Please give it up for AWS as well. [applause] So, this evening after the sessions, don't run home. I invite you all to stick around, okay? Uh we're going to have an attendee reception. So, there'll be food, there'll be drinks, there'll be wonderful conversation with your peers.

So, don't miss it. All right. So, shall we get into some MCP? >> All right. When I say MCP, you say, "You know me." All right. Y'all down with MCP. >> You know me. >> That's right. Okay. So about this time last year, I spoke at the very first MCP Dev Summit. It was about 150 of us in San Francisco. Uh at that time, the protocol was only about six months old.

Uh and it was starting, you know, to get a lot of attention, but people were just kind of tinkering, right? Just starting really to learn about what it was. And at that point, I shared uh my my speech with the audience about how we were using MCP and we rolled it out across our 12,000 person company and everyone in the company was using MCP. Back then, this blew people's minds, right? Since then though, MCP's growth has exploded.

The registry has grown eight times. SDK downloads are approaching 100 million. GitHub momentum has skyrocketed and most importantly adoption is moving beyond just individual builders because now the CTO's are considering MCP default standard. So, as MCP scaled across thousands of companies of all shapes and sizes, some of those little teenage pimples began to show, right? And this is a good thing because this allowed us to see and identify and address issues that were rooted in reality that, you know, it wasn't hypothetical, it wasn't theory anymore. This is actually what people were struggling with. And there's a new release candidate. How many of you have heard this? If you haven't, I'm breaking the news. Oh, only like three people.

Okay. So, there's a new release candidate that has come out and this is for a July 28th release. Okay. So, this was just announced and this release provides several ways that the protocol addresses some of the warts that we've seen in production. So the upcoming release makes MCP easier to operate. So historically MCP used protocol sessions. Okay. This release removes that requirement and it makes the request self-contained.

Now this is becoming a stateless protocol, right? which means it no longer needs to remember like who is doing what between the calls and this makes MCP simpler to use and to scale but you still need some state right so when applications need to maintain state across multiple interactions they can do this through explicit handles that are passed between tool calls MCP clients, they're connecting to multiple servers. How many of you are using multiple MCP servers at the same time in your workflow? Exactly. Right.

Each server has its own ooth provider, its own login flow, its own refresh rules and so on. So this release is tightening that. It's going to tighten around uh the rules around identity and and trust so that clients can safely manage all of those connections without making assumptions about who issued the token or where an authorization response came from. Part of growing up is also realizing that you don't have to try to do everything. Yes. Yes. So instead of built-in logging mechanisms, the protocol is now going to lean into observability uh platforms that teams are already using, right? Why try to reinvent the wheel? So developers can use standards like open uh telemetry to understand what's happening across their MCP workflows. This way MCP can focus on uh the client and serve a contract and then let the the established observability systems handle logging and diagnostics and things like that. So logging is being deprecated sampling. How many of you are familiar with the sampling feature? Okay. So this is the feature that let MCP say to the client, hey that that LLM that you're using, I need to borrow that right quick. Ask it a question and let me know what it says. Right? And so you were borrowing this from the server. Um this is being deprecated in the new release.

This one is sad. I know I love sampling. Um but also I get it right. the MCP community is increasingly drawing a line between the protocol itself and then the systems around it. Right? So from that perspective then yes it's reasonable to argue that hm maybe model invocation doesn't belong in the protocol. Okay. So servers that do need that functionality are going to need to integrate with LLM providers themselves. So if your server was using sampling, you'll need to adjust that. Um roots, how many are familiar with the roots feature in MCP?

So this is one of the OG features of the protocol. This is what gives clients uh it tells the the server the client tells the server this is your workspace, right? This is where you can work. So this release candidate deprecates roots as dedicated protocol feature and moves it more towards explicit uh approaches where the workspace scope is coupled with the request itself. Right? And this is one of those benefits of going to a stateless protocol. So instead of establishing the context once uh and expecting the server to remember it, you send this off with the request. Okay.

Now while these features are being deprecated there are alternative mode uh modes to achieve this same functionality. So one of the hardest problems in any young protocol is ensuring that it can evolve right. So this release candidate makes that possible. MCP has opened the door for a flood of innovation and many of these ideas simply need time and space to mature and to grow. But there's kind of a dilemma here, right? Either every young good idea gets prematurely stuffed into the protocol or maybe folks might take their ball and go play elsewhere, you know. So if they did that, that would fragment the ecosystem and we don't want that. So what this release does is give extensions a more structured path forward. So they can evolve independently. They can go and gain adoption with clients and servers.

Um and they can do this outside of this core specification. So another important addition is the formal life cycle for protocol features. So features are now classified as active, deprecated or removed. And this gives implementers a predictable way forward as the protocol continues to mature. So for example, logging, sampling, roots, they're all marked as deprecated in this release. If that deprecation remains during the final release, then that means you'll have up to an year before they may be removed.

So it gives you that guarantee of okay if we deprecate something you have at least a year to figure something else out. Okay. So the release candidate also ties conformance more directly to the standards process. So new protocol features now need uh matching scenarios in the conformance test suite right before they can reach final status. Anyone who knows me knows that I love testing. And what I love about testing is that it creates this shared definition of what correct behavior looks like. So this reduces the the misinterpretations of a feature. So everybody's not just doing or copying whatever like the biggest implement is doing, right? We all are kind of on the same page here.

Now the purpose of a release candidate is to give the community a early heads up, right? this is what's coming. This is your opportunity to test it and your opportunity to provide feedback. So this means between now in July, you all have some homework to do. Yes. So if you're building MCP servers, MCP clients, MCP SDKs, then this is the time to validate that release candidate against your real workflows. So can your sorry can your requests move across server instances without losing important context? If not you need to look for hidden

session dependencies. You should also review any uses of roots of sampling of uh logging.

So these are likely again to be removed within a year. So you might want to start migrating to alternatives now. also validate your client authorization flow against the updated guidance. So clients are managing many MCP servers, many authorization servers, credential sets, right? Make sure that everything behaves across all of those connections. And then the release candidate introduces full JSON schema now. So you can be more expressive about your inputs and your outputs including conditional shapes and and things like structured content. But listen, don't get cute here, okay? You want to tighten your server tool schemas. You want to be deliberate about schema validation and your external references so that you don't degrade performance or inintentionally introduce some security risk. Okay. So when people look back at at this release, I don't think that they'll remember the nittyritty details of the protocol becoming stateless or you know specifics about uh tool schemas or you know for example what they're going to remember is this is the release where MCP began phasing out of some of that that youthful chaos and started growing up.

It became easier to operate. it became more focused and it became easier to evolve. And that's really important with open standards because they succeed when lots of different people can innovate, can experiment, can even disagree while remaining compatible. And that's ultimately what we're aiming to accomplish at the Aentic AI Foundation. not just building protocols, but creating the conditions for healthy ecosystems to grow around them. MCP has had an incredible year, and this release makes me optimistic about the next 10.

Speaking of the work required to build healthy ecosystems around open standards, I'd like to introduce someone who is uh helping to lead that effort. So our next keynote speaker is David Neely. David is the director of developer experience at AWS and today he is here to talk about how Aif is driving community-led development of Agentic AI standards and encouraging broader open source participation. Please give it up for David >> [music] [music] >> I'm really excited to be here. Uh, thank you all for coming to join us. I will say it's a little intimidating following up Angie. I actually hoped that I would have a little bit of an advantage today because I've been uh in India for a day or two longer than she has. uh but she she comes with so much energy. Uh I want to talk to you a little bit about uh what the Agentic AI Foundation is doing in terms of interoperability because a lot of that is what we're trying to build and there's there's often a lot of confusion about what Aif is trying to do and so I just want to spend a little bit of time talking about that. But before we start there, I want to call out just how fast things are moving. Angie told you about how MCP has uh almost 100 million downloads.

That's crazy. MCP hasn't even been around for that long. You can see uh you can see just the the pace that things are moving at because it seems like every single day there's something new in the industry that's changing the direction that we're all headed in. And I've been around open source for a long time. I've been contributing for over 20 years to open source projects. uh I I was at the Linux Foundation as an employee when uh in the very early days of the cloudnative computing foundation.

So I saw Kubernetes come to the foundation and the the massive swell that happened there. I also served as the president of the Apache Software Foundation and watched the big data revolution come in and transform how

we thought about big data with things like Hadoop and Spark. And yet I'm still just amazed at just how fast things are growing. So, you know, if I look at AIF in the first three months, it achieved more than CNCF did in terms of members showing up to do work in an entire year of CNCF. And so, like the pace has easily quadrupled. We talked about the uh Angie talked about the 86,000 stars and nearly a 100 million downloads for MCP. Like a project getting to that stage that quickly is is mind-blowing.

And that tells you that there is really so much interest in this. And there's even changes here. When the Agentic AI Foundations was started, there were three projects. We had goose, we had MCP, and we had agents.MMD. D. And just a couple of days ago, we announced the latest project, agent gateway. And agent gateway gives you a control plane to apply consistent security, observability, and routing uh and and a governance layer across your AI agent and application traffic. And so I'm super excited to see this. It's not the last. You're going to hear uh you're going to hear in coming days about even more projects because the technical steering committee has a constant influx of people wanting to contribute projects. They're evaluating them to see if they're a fit, if they're mature enough, and if they're broadly uh adopted enough to be included in to the AIF. And so I'm excited because this shows that there the ecosystem at Aif is growing.

So I do want to talk a little bit about what AIF isn't. We're not building models. Don't expect uh openweight models to exist coming from AIFF. That's not what we're trying to build. We're focused on protocols. We're focused on tooling. We're focused on standards. And the reason that we're doing this is that we think that a a environment that allows for interoperability is going to be one that moves so much faster.

When you think about what MCP brought to the landscape, this ability to allow a model to call a tool and interact with uh various sources of data via those tools, that was a gamecher. Not really because the idea of calling a tool was something new. Like the industry had a number of protocols that would allow you to do things like that. But it was the first place where the industry had come together and said, "Hey, this makes sense. Let's all do this." And now we've seen an explosion of MCP servers. We see people saying, you know, if if this application or this software doesn't have an MCP interface, I don't want it.

And as a result, that's allowing the use of agents against a number of different tools to accelerate. And we think that acceleration is something that AIF can help drive. And so you're not going to see uh things like PyTorch. PieTorch is a great tool. Uh the PyTorch Foundation is something that I admire. That's not what we're building. We're focused on how do we get all of these agents to work together faster and in more standardized ways so that everyone can can go and innovate above the layer of communication and interoperability.

So just to give you an idea I talked a little earlier about momentum. Uh I really wish they had made some more progress on this. I really wanted this to be 200 member organizations. It's 199 today. I know we're in talks with a number of others. So, this number is going to go up. Uh but also 199 is a big deal. Uh that is a gigantic foundation and this foundation did not exist uh eight months ago.

So, 199 I'll just give you the NASCAR slides really quickly. We have the eight platinum members uh and then we have three pages worth of gold members. So I want to thank these folks for showing up contributing both

time and effort into the foundation. We also have a number of silver members. Uh and again these are folks who are showing up to contribute and to participate in building out this these standards and this interoperability and the foundation does not exist without people coming up and contributing and people signing up as members. So I want to thank them for uh recognizing the opportunity but also uh effectively endorsing and and joining with the foundation in driving this interoperability.

And then finally we have associate members as well uh the the research labs and the universities who are participating with us because they also see value in expanding the the pace of this. But that's not really interesting right that that's almost uh telling how many people believe in the future but not actually what the future is. So, I want to talk just a moment about project velocity, and you heard a great update on MCP from Angie, so I won't uh I won't harp on that. Uh, Goose just released version 1.37 on June 3rd, uh, which came with a a number of amazing new features, including a new great uh, uh, user text user interface, the working groups. So we have a number of working groups that are focused on things that are not single project related. So things like security and identity. Uh those folks 500 people are showing up every single month and they're working on driving the overall state of things like identity and security or commerce uh and driving that to closer to standards. And so we're seeing a lot of interest there. I'd actually encourage folks to to go look at the working groups because there's so many. Uh you heard about the the release candidate for MCP. There's more projects coming in the wings as I mentioned earlier, but I want to give a couple of things that I would like for you to do.

So, first of all, do go look at the working groups. If there's a place you're interested in, please join that working group. uh we we move faster by people participating and open source basically open source projects live and die based upon their contributors and so please do get involved with that uh if there's a project if you care about MCP go look at the pull requests if you care about goose or agent gateway please go invest time there contribute to the projects Or maybe you have a cool agentic uh project that you want to contribute.

You can contribute a project and bring it into the governance at a aif and we would welcome folks to make submissions there. You can check out aif.io. There's a process for submitting a project in the uh projects dropdown. And then finally, something that Angie actually built. So, uh, Angie just recently announced that there is an AIF ambassador program and I know she's had a ton of interest. Applications I think are still open for um, maybe another week or so. Um, this is a great place where you can make considerable impact by helping people understand how an Agentic AI future is going to impact them. And so, uh, if you follow this QR code, there's a link to be able to submit your application. I'd encourage you all to do that. And then finally, I I'm I'm really excited to, uh, get to spend time in India. And some of my team members here were saying, hey, we should we should go spend time with actual developers and build things. And so, they are running. They're they've uh reserved a cafe. We've got room for some people and basically we want to come build some things together. So, let's come build some tools around Agentic AI, build some applications around that.

I'll buy coffee. Uh the QR code will get you there. And they've termed this build with David. That feels a little funny announcing. So, build that. Come build with me. Thank you so much. I'm so excited that you're all here. I hope we have a chance to connect. I hope you get a chance to learn a lot from the presenters who are here and

on behalf of the Agentic AI Foundation, thank you so much for coming.

[applause] >> Thank you so much, David. Listen, the AIF is where the Agentic stack is being built and we invite all of you to contribute. David gave you multiple opportunities to do that. I challenge you to take them. And the reason is because this is something that's brand new for all of us, right? We should not sit in the background and let it happen to us the way that specific companies want it done, right? You have a voice. You can contribute what Aentic AI should be.

There's working group. Davis went through all of it. There's so many ways that you can get involved. jump in anything that interests you, any specific niche. You don't have to solve it all. Find a specific area and jump in. Okay. All right. So, our next keynote speaker is Riddic Roy. And Riddick is a platform advocate at the Cluster Labs. And today he's going to discuss five missing layers in the MCP ecosystems and strategies for scaling the protocol. Please give it up. Welcome Rick to the stage.

So hi everyone. MCP started couple of years back and now we are having a conference for that not just in India across the world. So that is a great feat. But today MCB is quickly become as one of the core protocols for the agentic stack. And today we have thousands of servers running everywhere. But the talk today focuses on what will we take to reach a million servers.

When we talk about vastness and scaling, we can't forget our close friends in CNCF. We have a huge landscape and CNCF isn't vast because of tools. It is vast because of the standards and how things come together in form of a huge landscape. Similarly, I want to point out on two numbers today. One is 13 weeks and the other is 13 months. MCP become loadbearing in 13 weeks only.

But when we see CNCF, it took 13 months to reach there. We can debate a bit about the numbers that is not important over here. But the thing we want to focus today is the ratio. When we see things expanding and moving so fast, sometimes the problem is the plumbing underneath kind of gets missing. So from building one server to governing others there is a stage when we talk about 2024 we were mostly concerned about can we build MCB servers what were the toolings required and how do we implement all those things together when we point out to 2024 we started building thousands of them and we started deploying them across our companies across our personal use cases but right now We don't need to prove MCP is a standard anymore. It works and it works very well and we are expanding that to a millions of servers across different of the organizations.

But between now and the future where we scale our service from thousand to million, there is a gap and the gap is between the governance and maintaining all those things together. To focus on the gap, we need to understand where we fit in. There are consumers and then there is the standard. Consumers generally used to be us humans but right now that has moved to become more AI agentic. And then there is the MCB protocol that connects between agents and tools.

And then there is the infrastructure the Kubernetes cloud database APIs all those things which connects

everything together. But in between we have couple of more things. There is discovery the federation between different MCP servers. There is identity. Who and what is in the picture? There is obsibility. How do you manage traces and what your identic AI systems are doing? There is reputation and there is metering around cost. So if we focus on the core parts, Kubernetes didn't containers. It won because of the entire ecosystem from service discovery to work identity.

There is a gap which we are here to build and discovery federation and workload identity and all those areas are key part of it. And now let's start with discovery. Imagine your agent needs to go and get a MCP server. You have some in your organization and you have someone to use. So the problem is there might be a lot of MCP servers. your official registry, a vendor catalog, your private enterprise registry. But the problem is there is no share resolution in between. So you will get a lot of option and your agent might be confused what to use and where to use and that is a problem overall because you want to be using the trusted one and not get leaked in the flow.

Next is oath. This is not just a single oath where you login or something like that. This is more deep. Now we don't have just agents. We have agents doing everything for us from writing emails to sending and monitoring our entire log systems. So if someday at 3 a.m. your agent deletes your database, you want more visibility about what happened. We have OBO and others as well as multihop, single hop and all those things in between where you get more visibility about what is happening, why it is happening, who is causing them and what's the owner. Here there's an entire loop about how bearer tokens are exchanged like for example in OBO you can send your tokens and that are exchanged for newer tokens which have may act loops. So you don't get complete access, you get minute access to what and why something is happening which is very important.

Next we talk about obsibility. One agent today can cross multiple MCP servers and this is by the nature of agents. Agents are meant to complete in reconciliation loop till the task is not finished. But today there are some work being done over here. But it's very hard to understand how you trace a agent between different servers and there is no semantic convention over there. There is a lot of work being done. But if there is no portable tooling which shares and connects all the traces together to give you a overview and a single pane of view towards everything that becomes a problem as we scale to not from a thousand servers to a million of them.

Lastly there are many MCP servers and there are many running on your systems. What we learned when we had a lot of docker images and everything together trust becomes very important and now there are MCP registries and all into the picture but there is not a way where you can score trust and reputation together even if you go and search in MCP registry GitHub you will find thousands of servers over there so how do you validate trust and reputation over there so it's just not a server name it is more than that overall the One more important thing is like we can't govern what we can't measure. Now if you talk with your MCP server and give it a prompt to generate a report, it will go generate a report by calling 25 MCB calls for LLMs and all those things. Your platform will see there is a cost but there is no attribution to that cost and what it happens what it cost depending upon business unit that becomes a problem as we scale because now tokens are being token maxing is a thing but as soon as there is phenos coming to the picture we also s the launch of the token foundation it becomes a problem because companies are limiting the tokens so that overall creates a problem on how

attribution comes into the But things are changing for the good.

There's a stateless MCP July RC coming to the picture and this is very important because there is no shared session stores. So shared session takes a lot of context and a lot of tokens but now with going to the stateless manner. The good part is that you can have multip multiple instances of your servers which can talk to the agents and then there is more saving and more helpfulness over there. But that is not just enough there is more. You have discovery layer when you can talk to the server to get the menu of the things it can do. There is work on identity that helps you to log into servers more safely. There is obsibility. So there is standardization of every request. So you can see what's happening and why it's happening. And one of the most important thing there is metering which comes into the picture. So with gateway can finally count and cache all the calls. So your MCP method and MCP name can be used in there. Overall there is a lot of work being done and if you are interested to learn more there is an awesome blog where you can just scan this which talks about MCB is growing up and the different things that is happening over there. Overall we are in a great phase where Engine AI tokens consumption and people are having more agents right now and the protocol together makes it happen very easily.

So yeah, thank you so much. And the one takeaway I'm here to give is that there is a lot more to do and we are just getting started. Thank you so much. [applause] Thank you. Thank you. Great job. Okay, so our next keynote speaker is Sam Party. Sam is founder and CTO of arcade.dev. And today he's going to look at how software design is evolving from human first UX to AI first machine experiences. Please welcome Sam to the stage.

>> [music] >> How's it going everybody? Well, so I have an interesting thing to talk about today that's not how the spec is evolving. Um, even though you've heard a lot about that and I'm very excited about that honestly, but it is actually about this topic called machine experience. Now, some of my engineers call themselves machine experience engineers now. And why is that? Well, because they're not always programming for another programmer.

This is one of those times when C became Python that the abstraction changes the programmer's job and responsibilities, the things they think about. It's not memory management anymore, right? But it is encapsulation, right? And so if you think about these kinds of shifts, right, that we've been through in the entire ecosystem, we are undergoing one of those right now and we're actually at the precipice of it. Why is this? Well, in about 18 months, we went from here's a good idea to everybody in their mother making an MCP server. Okay, so almost every single company out there at least has their attempt at an MCP server, even if it's just wrapping their APIs, like I'll talk about later, being not a great approach, almost everybody has invested a little bit in its ecosystem. And why is that?

because we are at that time where you can be a part of something just like the folks at Xerox Park were or the piece the people at Motorola or you know Nokia making the first mobile phones and things like that that is the moment that we are hap that we are in right now and it's this transition from building applications for a specific user or a specific consumer that is a human and actually shifting that into possibly an agent or even just a server

running an agent. IC program. So what is the shift?

It is from UX to MX. Now MX is something that we've largely made up. So don't take this as gold or put it in a textbook yet. Um but user experience was one thing that we designed for because the user was relatively well understood. this ecosystem changed very rapidly and some of those fallouts that we've had are some of the things I'm here to talk about today in the sense that machine experience isn't always well understood right now and there's some problems with it that we need to go over and make sure as a community that we address some of these are addressed in the spec but some of these necessarily have to be designed and solved by the application developer that is building on MCP and so that's really the The point of what I'm going to drive home today is if you're going to be taking this next iteration of the spec and you're going to be building your company around it, you need to be aware that you have a choice of a new abstraction to build against and that that option is open to you because that opens up the opportunity for more people to build on the UX like generation that came before but now in this new machine experience generation. So I'll go over a couple of these today but some of the ones that they've exposed um I'm if you're coming to Mumbai I'm doing the other two in Mumbai but today I'll talk about identity and composition.

Identity and composition really the first being I'll talk actually about composition first is to talk about tasks and intent for your agents and not necessarily APIs. And so why is that better for an agent? Well, you'll see there's a mismatch here. If I tell my agent, find something about a customer, send them an email, what am I doing? I'm giving it a task. I'm giving it an intent. What does it have to do right now to actually go and perform that action to satisfy that intent?

Well, right now, it has to think like a human. It has to use APIs we've designed for human programmers to go and use. And that gives it five chances to mess up. And so this is where composition comes in. Tools are not APIs. The mismatch here actually creates a staggering number of the failures that we see in the MCP ecosystem. We at Arcade have been building everything from MCP gateways to tools to servers themselves.

And the most failures we see are when people just take an open API spec, wrap it, and call it an MCP server. That gives it the utmost chances to fail. And that is a great example of the difference between what we designed for a user, in this case a programmer user, right? That is an interface change from that user now to an agent user, the machine experience user. And it sounds a little heady. I know you got to bear with me on that, but it is true in the sense that if you look at the numbers, you can go check out this Nurup's paper that we contributed to or any of the other ones like taskbench, right? As soon as you get you're going to go wrap your open API server. As soon as you get to eight tools, you're looking at an abysmal failure rate. Okay? And these are all published papers. You can go check them out yourself. These are the problems that we see at the application level. You, the developers using the MCP protocol are in charge of this. And as important as the spec is, it's only as good as the people building on it. And so if the people that are building on it don't understand this change that is happening this phenomena this change from user experience to machine experience well then the MCP spec won't really do that much and so if we understand this problem of composition this being one more example where we can say things like instead of having to get the user ID right this is a very pseudoy example right but instead of having it go get a UYU ID which we all know wasn't in the pre-training data set and was definitely

not very well understood in terms of the semantics that actually large language models understand then it's going to do a poor job of putting in the right UU ID. This is again an example of that shift. Don't think about a you know a case switch. Think about what is the easiest thing for this machine this agent in its experience of using this interface to use and that will make the application layer significantly more accurate as proven by much of the research that has come out and just personal experience now making over probably 60,000 tools for arcade.

The next one that I'll talk about is off. And this is a big change for a lot of people. It's not just getting in the front door or getting in the front door with a static service token. It's getting in the front door as the user and then authorizing the agent to act as that user to perform some action. So if you act as a user, that is actually typically a new token. Think about getting in the front door. OC C authentication. You log in. It's like social login. You go into that website and you have a static set of capabilities, right? That has been granted to you by that site. But what if the thing you want your agent to do is send an email and you're on some website like Zenesk, but you want it to send an email with Gmail. New token. That's an authorization token that is granted to the agent to act as that user. This is called delegated authorization and it is part of the tool authorization part of the elicitation spec that we contributed to MCP. This is a very big topic and I'll actually discuss more of it in my next talk but it's very important for the application developer to understand these two paradigms right now and not make the mistake of just passing the user token as much as I love openclaw uh just passing the user tokens to these various services and using them in conjunction. If you look at a number of the hacks that have happened right now that are increasing in prevalence, they are typically happening when OOTH scopes are too privileged or service tokens are too privileged and that is giving an overpermission set to the agent. Lots of demos and PC's you see that get killed are because people are using service tokens or user tokens that are underpermissioned and then your demo isn't worth anything. So the only way that we can accomplish the intent, the task, the machine experience that our users want is to perform their off seed to the server, their authentication to get to the front door, the social login we've been slapping on our websites for the past 20 years, it feels like at least. and then the delegated agent authorization to me a new token to give that agent the power to act as that user in the finest grain possible the least amount of privilege possible that the user has to delegate to that agent and that often has to be at runtime that often has to be in the chat where the user is actually chatting with the agent and to have that agent actually recognize the cases and situations in which this happens. Yes, you heard it once again a machine experience problem.

And so it's very important that these two paradigms of composition and of identity and the next two in the Mumbai talk if you're coming please do come by that these are well understood by the application developers in this room because you are what the next generation will be defined by and I do believe this we are at a Xerox Parklike moment in this ecosystem where a lot will happen and a lot will be defined and at this precipice moment we can be someone who makes a significant amount of change if we are involved in the community. So you can be the next, you know, Bell Labs laboratory assistant that actually comes through with, if you know who I'm talking about, the most important invention of the next century because that is the type of moment that is defining of where we are right now. Um, thank you so much. My name is Sam Parti.

I'm at arcade.dev and I appreciate the time today. Please do come by the Moon by Talk. Thank you so much. >> [applause] >> am. Great job. Are y'all learning a lot? Yes. >> Okay. Wonderful. Wonderful. So, our next keynote

speakers, we're gonna get three for one with this one. So, we have Arjun Vincad Raman and uh Jagadish Babu and Nha Jagadish. So, Arjun is the senior officer of artificial intelligence at the Gates Foundation. Jagadesh is the chief operating officer at EPSEP Foundation and Neha is the lead for AI engineering at ECARE. So today they're going to come and discuss building trustworthy scalable AI systems for India's health care and public services.

First up is Arjun. Welcome to the stage. >> Hi. Uh good morning everyone. My name is Arjun Winkat Raman and I'm a senior officer for artificial intelligence at the Gates Foundation's India country office. Usually I'm the uh tech guy in social impact conversations. So it's a it's a privilege I guess being a social impact guy at a tech conference for a change. So we'll take a little bit of a step back. I think the morning's been a fairly technical uh morning of you know what's been happening on the MCP protocol and so forth. I'm going to take a step back and talk a little bit about how we see some of these innovations applying in the real world to problems that most of us care about. Uh so the Gates Foundation for the last 25 years has been taking leading edge research and innovation and applying it to some of the most intractable problems and difficult problems that humanity faces like managing maternal health in remote and rural areas, managing to eradicate diseases, helping farmers adapt to climate change and changing times. And what we've realized is that for innovation to truly make a difference, it needs to be underpinned onto solid foundational systems. Um, our AI strategy is kind of similar in in u it generally follows three broad pillars.

We try to take AI innovation and then apply it uh to these problem areas that we are going to be focusing on for the next 20 25 years uh as we close out the foundation's life cycle. And the first pillar there is taking AI and whatever innovations are happening in the in the space and then applying it to core problems in health, education, agriculture, figuring out what the actual use case is. Is somebody going to use a voice and language uh AI tool to understand their own health better? Is the farmer going to use it to identify pests better? Figuring out the use case, figuring out who's going to build that use case, and then figuring out how to scale it up within uh within the public ecosystem. This requires building out government and uh public capacity to select solutions, evaluate impact and then make sure that the funding that is being uh used for scaling up these solutions is being used well. And then the last part that we focus on is building out the domestic uh AI ecosystem or the innovation ecosystem here in India and in other parts of the the global south.

Um our general investments in this space look something of this sort. uh there are two broad categories we invest in enablers which are sort of underlying crosscutting artifacts or assets or infrastructure that uh that other people can build on. So India's DPI ecosystem is one of those underlying substrates that we rely on a lot for scale up of our programs. Similarly, data is going to be a huge requirement for AI and for particularly for the use cases that we care about. Data is hard to come by. So enabling that data to flow between different kinds of public systems is another core area of focus and then tools like the agentic AI frameworks MCP and taking those and making sure they're purpose fit for deployment in in these domains is is that last part of AI infrastructure and tools. Our use cases uh typically fall across five different categories. We work a fair amount on direct to consumer tools where a like a farmer or a pregnant mother or a frontline worker might be using this to solve a problem for themselves or it might be a tool that is used by a last mile service enabler like a like an asha worker or an Angani worker to actually go and uh do their job better which means tracking their beneficiaries indicators giving them the right advice and so forth. And then there are three other categories which are sort of

more traditional AI.

So predictive intelligence for things like weather modeling, uh figuring out when kids are potentially likely to drop out of school and so forth. Uh and then applying some of this to workflow systems and efficiency within government departments, making some of those processes flow easier, helping governments use their data to make better decisions. And then the last part there is uh research and development acceleration. So taking AI and then applying it to solving sort of frontier problems. So an MCP and agentic AI there is is looking really promising as a research tool. So enabling u things like new drug discovery using AI enabling uh improved weather modeling that responds sort of more to adaptive changes uh better. So those sort of things would fall under the R&D acceleration part.

And uh this here is our partner ecosystem uh here in India. You'll see it spreads across a fair range of folks. So we have the consulting and uh sort of traditional partners who do program management and implementation. We have technology and innovation partners u some of the folks in the room uh as well and we'll have a care join shortly and talk a little bit about how they're leveraging India's core infrastructure to develop healthcare solutions. We work a lot with academic and research institutions. So here in Bangalore we have triple IT Bangalore. We've worked with uh IIT Madras quite a bit and so you'll see a few of our partners listed over there. uh we work a lot with government so both line ministries as well as state governments and central ministries uh most recently the India mission has been one of the areas where we've been working closely with uh the central government and then we have a fairly vast uh philanthropic partner network as well and I'll now hand over to somebody who's been part of the DPI ecosystem in India since its very inception and can speak a little bit to how that underpins a lot of the thinking so I saw session earlier was talking about things like discoverability identity. All of these concepts have been implemented for public services in India using the DPI ecosystem starting with Aadhaar and then UPI and then more recently the national digital health mission or the Aish man bharat digital health mission and Jagish will talk a little bit about how they see that substrate that core foundational layer contributing to better innovation in the AI space. So that handing over to Jadish.

[music] Hi. So from technology to purpose uh it's very exciting that uh I'm here to talk about today. [snorts] So I will be briefly talking about an idea called open agreeet. when we can bring data together to fire the purposes of agriculture for small and marginal farmers. That's the idea today, right? What excites me the most in this last 3 four years technology development is first time technology can reach an illiterate digitally not so literate those who do not have a smartphone.

That's the most exciting part of last three four years and I'll briefly present to you an idea that we are harnessing these technologies for the purposes of small and marginal farmers in Asia especially in India and Africa. So the larger idea if you think about DPI you can think about DPI as data on rails. Simple idea, put data on rails, right?

When that data becomes institutional data, the trust of the data increases. When institutions come together on rails and that data gets connected to AI, that's when trustworthy systems can be unleashed. That's the larger idea. And this is the conception of DPI to the power EI in simple terms. When we take this design and apply it at state, national and regional context. That's the idea of open agreeet network of institutions, organizations

coming together.

At one level, it is powered by identities like agree stack in India. which has identity of farmer, the land, crop. This is what helps to personalize and another level various systems and their data coming together like warehouse information. Imagine you have crop is harvested. I need to keep it somewhere. Option one, access the warehouse app. Otherwise, access agent that actually is represented by the warehouse system connected to one larger grid. Right? Similarly, market information. Similarly, advice on what to grow, how to grow etc. All of these contextualized, personalized for the benefit of the farmer.

So with this idea in India, Bharat Vistar as an idea was launched on 17th of February. And this has been in works for a few years. And here significant state and central information systems are coming together. Previously they were all having their own websites and portals. Now they all bring their data systems together using this framework of data coming together and AI coming on the top of it. This as a DPI power brought by AI is now live in four states and in the next 6 months it'll have many more states joining this.

So I'll briefly show you another interesting development. In the last year and a half, voice technologies especially in Indian languages and in African languages have become very usable. When you connect this framework to voice with personalized information, things can get very exciting. So here is your So as an idea in Amul they have integrated all the records of the cow and its health insemination status everything into the system highly personalized secure yet very accessible on a feature phone you don't need smartphone right the idea that systems can be harvested now has become much more powerful because they are all connected head in a real time. You don't have to wire up technologies. A agent will take care of the job has been very powerful. The other interesting story is that when you actually package all of these into systems, open source systems, the time to instantiate has come down from 9 months to 3 weeks. Amul went live in 3 weeks. You may have heard about Nandan talk about it during the India AI summit when he was sitting with prime minister.

That is the power when you actually package all of this. You can actually bring institutions together for firing up a purpose. That's the idea of open agrenate. We have actually packaged all of this as open source and now it's being used. The other exciting aspect of uh the technology today is that irrespective of the data you could connect systems and harness them. You don't have to harmonize data. So that means that the existing technologies institutions that are powering data can come together very easily and that's been great excitement there. So here an example of a farmer question an advisory engine that can conservatively without significant reasoning infer from existing data and provide safe responsible advice to farmers. This is the best of the technologies coming to smaller marginal farmers on a feature phone. Right?

So Arjun showed about quite a few partners and here again open agreeet in India and Africa powered by host of partners bringing and playing different roles. What excites me is when state center data gets connected, ecosystem data, market data, communities data gets connected, that can increase the amount of convenience and empowerment and agency for farmers exponentially. That's what excites me. Right? So with that, I'll invite Nha. Thank you very much.

And if you are interested in open agreeet, you can join the community online and contribute to the cost of small and marginal farmers. Thank you. [applause] So let me start with something very simple here. If you break your hands, god forbid, would you go to an orthopedic or a general physician? The answer is pretty obvious, right? But there's a point to make in that that health care is very specialized and domain knowledge runs very very deep.

Let me add some complications to it. Let's take language for example. In Karnataka, someone might say or someone up north might say does a generic model understand all these contexts? Probably not. And add the complexity of Indian drugs to that. A drug like Jorac if you were to suggest that to a particular model which is a generic model it might just go ahead and spell correct you. So the thesis for the five minutes of my slides is that one sizefits all does not work for healthcare and that's where agents come in right the agent universe has the building blocks of skills tools via MCP context workflows and all of these put together solve for the problem. Let's take an experiment that we ran. So what we did is we ran a drug identification test with a generic model and the model was accurate 83% of the times. We added the AA drug MCP to it and the accuracy went up to 99.7%.

This jump from 83% to 99.7% is the difference between a prescription and an incident. Having said that, there's a subtle catch to it. There is another MCP of ours, the Indian protocol MCP which again we ran an experiment with. On one side there was a certain model which actually improved by 15% and another model uh fared slightly lower. Why would that be? That's because it was calling the tool but it was not using the response of the tool to answer. Instead it depended on his memory. So the takeaway here and it's a very important takeaway for this room is not about just having tools.

It's about using them correctly and how to use them correctly. And that's where evaluation creeps in which is the foundation of any agentic system. At AKA this agentic system we apply it across three fronts. One is the doctor, the patients and the enterprises. Let's talk about doctors. So why voice right? So imagine you're in a consultation room with 50 patients expecting a doctor to write notes. Absolutely not. But he's already having a conversation with the patient. So why not let your agent list listen to that?

That's where our model uh AKA's purpose-built healthcare model for paratlet comes in. The second part is the headless by design. Every doctor requires structured output which is catered to his needs. Instead of saying that let me build a UI and custom fit a doctor's needs into that, we say that you can ask for what you want and let the interface be adaptive to a doctor's needs. The last point is context and memory. A patient context is the most important thing when it comes to building systems for doctors. Uh imagine a patient who's coming in twice a month and complaining of chest pain versus if you don't have the context of that two times, it's a completely different scenario. So built on these guard rails, our AKAScribe is running on uh big hospital chains like Midanta, NIC by the government and a lot of other clinical workflows where real world doctors are using this.

The second use case is the enterprises. So it's similar guard rails, right? It's the same agentic guardrails, but it runs horizontal. Ask an enterprise what they want. It could range from being doctor booking to healthcare service booking to insurance claim acknowledgements and it could be anything that's the need. So when you're

building horizontally that's where skills come in using the right skill and to load the right skill at the right time would be the need of the art.

This is where basically MCPS require a lot of discipline not loading all the tools together because choose your favorite popular MCP and it might eat up to 20% of context of your um context window before the model even gives the right answer. So choosing the right tools and when to load the tools is what would solve for this use case. On this platform we have Apollo one of the biggest hospital networks in India running their agentic assistant using these.

Last is the patients. We have deployed a self-care agent in South Africa where patients interact with it and we kind of provide advice education as well as in certain cases we prescribe OTC's. So in case of this one of the biggest problems or one of the hardest problems that we did solve was the fact that when do we ask the models not to prescribe. So this is basically how we solve for healthcare. Now look at the numbers. Our platform call numbers are more than 70 million calls in a month. This talks about something. a doctor in Karnataka, a patient in South Africa and an enterprise in uh Delhi all use the same agentic guard rails which is based on which their solutions are built and that's what this is about. It's not about a generic solution or a generic model. It's about a purpose-built or a customuilt solution. We open source everything that we build upon. So thank you very much. Please do come and build with us.

[applause] So amazing. Amazing. What a fascinating [snorts] talk. All right, everyone. Thank you all so much. Uh what a great way to kick off the event. Let's give a round of applause to all of our keynote speakers this morning. We're now going to take a break so you can enjoy some coffee and some uh communication out in the foyer and then we're going to start the track sessions at 11:00. Enjoy the rest of the day and I'll see you at the afterparty.

>> [music] [music] [music] >> Fire. Come on.

Heartiph. Heartiph. [music] [music] [music] >> [music] [music] >> in my heart.

Heat. Heat. [music] Fire. You [music] [music] My heart my heart dip.

>> [music] >> Fire. My heart. [music] [music] >> [music] [music] >> I think you fire your heart.

[music] Heat. Heat. >> [music] >> for you.

[music] Fire. [music] >> [music] >> Heartip my heart.

fire. My heart [music] [music] [music] for your heart.

[music] >> [music]