Windows and Surface Event
Satya Nadella – CEO, Microsoft
Jensen Huang – CEO, Nvidia
Pavan Davuluri, Jacob Andreou, Kayla Cinnamon
Wednesday, October 7, 2026

SATYA NADELLA: All right. Good morning. Good morning. It's wonderful to be in San Francisco with all of you for a Windows event. In fact, I always think about this, my life was shaped by one such event back in '91. It was at PDC, obviously not that far from here. That's the first time I came to a Windows event and heard about Windows 32, and the rest is history, as they say.

And in fact, I got up this morning, and I was thinking about what these launch events mean to all of us. They obviously mean a lot in the moment. They're about a device, they're about an operating system update, about an application, whatever, and what it all could do in the moment. But they're are also trajectories, and that's what I think is more interesting.

It's not often talked about at the moment, but those attending those events have it in their heads. For me, that was what was happening in '91. I could clearly see essentially what was going to be the x86 architecture and Win32 and not for the PC, but the server. That's kind of what got me excited is back in that time, I said, "Man, this is going to be the thing that's going to change computing, not just on the client, but on the server."

I think today's event has the makings of something like that. Today, when we think about the frontier ecosystem, what do you think about? You think about the big model release, the big model capability job. I think today is the day we will start talking and thinking about the frontier ecosystem taking maybe slightly different contours, and two years from now, five years from now, 10 years from now, that classic adage that distributed computing will always remain distributed will be more true.

So for us, when we think about the frontier ecosystem, there are three core pillars. The first is trust. There is no way to build any ecosystem without trust. So how do you ensure that you really have trust going forward at every layer of the tech stack? Obviously, it's about intelligence. And then how do you design it? In fact, this is probably the other thing that's overlooked on.

In fact, if you look at AI, success came about not by just the AI capability. In fact, if you look back at ChatGPT moment, it was the magic of Chat as an interface mixed with the RLHF back in the day and the AI. That coming together of design, the form factor, and the intelligence is what made things happen for us as users of it. That's why thinking about the flow of human ambition as the design challenge for AI to meet, that shows up in the form factors.

These are the three core things that we've been working on, and today the idea is how do you bring these three things to Windows? We are really taking Agent 365, Microsoft IQ, and Copilot. These are the three core anchors to what we've been doing. With Agent 365, this is about ITOps,

SecOps, FinOps, all the things that one needs to really be able to trust your agent. How do we not just have this in the cloud, but fundamentally build it into the core of the Windows architecture? You'll hear about that.

Microsoft IQ is essentially the system around the model. In fact, one of the things in the last, I would say three to four years, we have realized that just having a model doesn't do much for anything. You really do need to orchestrate it. You need to have memory outside of the model. You need to obviously have this harness layer that is able to take multiple models plus context plus memory and the action space, and really then prosecute the trajectory of the model's output. That is what Microsoft IQ does.

And then lastly is the form factors. It's everything from Chat to Cowork to Code, and now of course, agents and Autopilots all coming together and enabling not just our apps, but apps for anyone, anyone's age. That's what the Windows platform is all about going forward. I wanted to really Pavan him come up on stage and take us through essentially the birth of really the next generation of Windows agentic platform. So with that, Pavan, why don't I throw it over to you?

PAVAN DAVULURI: Thank you, Satya. Good morning, everyone. I'm excited to be here. The team has been building towards this moment. I'm glad we finally get to share it with you. As you just heard, trust, intelligence, and the full power of the ecosystem coming together on a device that people use every day, their opportunity. Over a billion Windows 11 users. We have a real opportunity to bring the frontier ecosystem to the lives of people everywhere, combining the power of the client and the cloud with Hybrid Intelligence.

Trust is where it all starts. It's something we think about as a team every day. Today's operating systems are built for users in an app-centric world. Agents change that paradigm. They can reason, use tools, solve complex problems, often independently working for hours at a time. We need a new set of primitives in the operating system to unlock the potential of what agents can do for people.

Agents need flexible containment, a distinct identity, and permissions that are separate from the user. We are building these primitives directly into Windows so agents can be secured, observed, and governed, making Windows the most secure platform for agents. Let's start with containment.

Existing sandbox solutions are insufficient for securing agentic workloads where containment needs can change with every tool call and every prompt. We built Microsoft Execution Containers or MXC to meet that need. With MXC, agents are contained in a runtime composable sandbox. Organizations can set policies for which files and networks and agents can access, and Windows enforces those policies at runtime. It gives IT confidence that agents can only access what is approved.

Today, I'm excited to announce that Microsoft Execution Containers is moving from preview to GA on Windows 11. MXC is a cross-platform library that works on Windows, Linux, and MacOS. Only on Windows, MXC is deeply integrated into the OS so organizations get more containment choices and more control of their data. The spectrum of containment Windows

offers ranges from process to session to WSLC to VMs to Windows 365 for Agents. Developers can start integrating their agents with MXC today.

Next, let's talk about identity. Whether an agent is working under your direction or acting on its own, every action it takes should be attributed to the agent, not to you. Only Windows enforces this attribution strongly at the operating system level. That attribution is what makes observability possible.

Next, manageability. To build in containment and observability, Windows brings a new level of enterprise manageability for agents. With Agent 365, IT sets policies once, and Windows enforces those policies on every device. Windows turns agents into enterprise manageable assets.

So let me bring it together. Local identity patterns have emerged rapidly, but they are fundamentally insecure because they have broad system access. With MXC, Microsoft Security, and Agent 365, we're unlocking the full potential of agents on Windows. This is the secure platform we're building for the world.

GitHub Copilot is one of our first partners taking advantage of this. Divya is here to show you what that looks like. Come on over, Divya.

DIVYA: Thank you, Pavan. As we outlined, we're evolving Windows to be the most secure platform for agents. A critical aspect of that is being able to separate agent activity from the user activity, and that's exactly what Windows is enabling with MXC and Microsoft security stack. This separation will allow security teams to evaluate agent behavior independently from the user behavior. Now, let's see this in action.

Here's a website I've built. I'm using GitHub Copilot to update the version Python my website uses, following the instructions on this external web page. What I don't know is that this page contains some malicious instructions designed to redirect the agent behavior. Just like the user, Microsoft Defender detects and blocks the attempted malicious action and also blocks the agent is risky. That's pretty cool.

Now, detection tells us something went wrong. Distinguishing the agent from the user lets organizations respond precisely and contain that agent specific risk. And in fact, Windows makes this possible with voice and post-agent-level attribution that separates the agent activity from mine. That means access restrictions apply to the agent, not to me. Without this attribution, think of it for a moment. Blocking this malicious agent can also block the user, slowing down productivity. Here I'm asking the agent if it can access this document.

Watch what happens. The agent no longer has access to the SharePoint, but my access as a user of this system will remain intact so I can keep working. On Windows, organizations will be able to understand which agents are running, associate each activity with specific agent, investigate risky behavior, and also apply policies for a single agent or group of agents without relying on user-level data or device-level context.

To summarize, with Windows, Microsoft Security, and Agent 365 you control what agents access, know which agent acted, and govern activity across your organization. Back to you, Pavan.

PAVAN DAVULURI: Thank you, Divya. This is where our partners have been asking for, and many are already building on the platform, including Anthropic, OpenAI, Nvidia OpenShell, and OpenClaw. Now, security agents is the foundation; unlocking Hybrid Intelligence is the opportunity. Customers want to consume more cloud intelligence than economics can really allow. Hybrid Intelligence exists because people's appetite for intelligence is effectively unlimited.

As models continue to grow fast in size and capability, it becomes a real cost. When I talk to customers and partners for scaling AI, they often ask, "How do I get the most out of my token budgets without giving up frontier capability?" With Hybrid intelligence, there's no need to choose. We're moving toward a future where Hybrid Intelligence is always available, finding locally where it makes sense, reaching to the cloud where it needs to, and seamlessly bringing the best of both.

To bring Hybrid Intelligence to life, it takes intelligent routing, models, runtime, and silicon all working together to create a meaningful experience. We're working with the GitHub team to bring our vision for Hybrid Intelligence to life on Windows.

Earlier this year. GitHub launched HydraFusion, which routes each task to the right model in the cloud. Today we're extending intelligence to Windows, letting HydraFusion tap into models running locally. Let's talk about these models.

At Build, we introduced MAI-Code-1.1-Flash, a 130 billion parameter coding model built for real-world developer workloads: now we're bringing that same intelligence to your device. This is where full-stack engineering comes in. We've taken that model to three bits of precision, reducing its size by nearly 80% while preserving coding quality. That is an incredible feat of engineering.

But really, shrinking the model to the device is only half the story. MAI Code can work at an impressive 256K context window right on your device. Our goal is simple: Let AI run locally, stay responsive, but leaving the room for everything else on your PC.

And there are many more powerful models coming to Windows. Nvidia is working on a new Nemotron model. With over 70 billion parameters, it manages to quantize to two bit and uses just over 20 gigs of memory, making it ideal to run on RTX Spark.

And why stop there? With advanced quantization methods, it's possible to bring more of the frontier to your PC. At 1.6 bits, DeepSeek-V4.1-Flash a 284 billion parameter model is running locally on device, fitting in just (Inaudible.) fitting in just 60 gigs of memory, this model outperforms GPT-5 encoding and reasoning benchmarks.

A year ago, such capabilities were exclusive to Frontier cloud models. It's quite amazing. Sitting in just 60 gigs of memory, this model outperforms GPT-5 encoding and reasoning benchmarks. A year ago, such capabilities were exclusive to frontier cloud models. Now, you can have these capabilities running on your PC. And this is just the beginning.

To power this growing ecosystem of highly capable models, you need a performant runtime, and this is where Windows ML comes in. Windows ML is a scalable runtime to deploy models across GPUs, NPUs, and CPUs, and we're continuously evolving it. We're excited to announce we're bringing Llama CPP to Windows ML.

(Applause.)

This gives developers more choice, open source models on day one, the ability to experiment quickly with emerging models as they land.

With intelligent routing, highly capable models, and a performant runtime, a clear recipe emerges for what hybrid intelligence looks like. That's the repeatable pattern we'll apply to light up experiences across coding, security, creation, and work.

Kayla is here from the GitHub team to show you this in action. Come on over, Kayla.

(Applause.)

KAYLA CINNAMON: Thank you, Pavan. I'm excited to show you how I'm going to use the best of cloud and local models to do my coding workflows in GitHub Copilot on Windows.

To start, let's just paste in a simple prompt, and I have the Windows terminal codebase cloned onto my machine. What I'm asking is, I want our agent to detect unused code, imports, packages, and obsolete feature flags, and then give me an actionable summary. This is just basic code cleanup that I'd like our agent to do.

And then here at the bottom, you might notice in the model picker, I have auto-efficiency enabled. this is going to select my model depending on the complexity of the task. And now, with our new version of Auto, it will automatically pick a local model depending on if that is a more simple task. Here, we can see on the left-hand side, our GPU has ticked up, and that's because we're leveraging our local model here.

And I did run this same prompt before on a different code base, and we could see that it leveraged MAI-Code-1.1-Flash Local. Now, since my code is cloned to my desktop and my model is local, we could just put our whole machine into airplane mode, and this process will still continue to churn because everything is right here on the device.

This is just a simple coding task right to the local model, but what if we asked something a little bit more complex? In this session, I asked our model to, or I asked our agent to look at the issues in the Windows Terminal repo that have the needs repro labels. These are issues that our team hasn't been able to reproduce. And then I wanted to do some investigation and then hand off to

some local subagents to do further building and testing, and see if we can get more information here.

This prompt actually ended up leveraging GPT-6 Luna. This is a cloud session here, and then we've got three local sub-sessions happening on the left hand side, leveraging MAI-Code-1.1-Flash Local. This is a really great example of hybrid intelligence.

If I hop into one of these local sessions, we can also see all of the tokens that I got for free. And on this one in particular, we've got 1.6 million being sent to the model, and then 10,500 coming back from the model. And all of this was free, and this was just one of the sub-sessions that I ran.

I have a lot of sessions going. I like to take a look at what I've got running on my machine and what I've run before. We have a canvas that we created that is a really great visualizer of my sessions. A canvas is a custom UI that works inside the Copilot app, where both you and your agent can work back and forth and update the UI accordingly.

Here in our visualizer, all of my purple sessions are local sessions that I've run with our little leaf, and all of my blue sessions with the cloud are cloud sessions that I've run. And you might notice I've got the same session happening a few times. I'm doing issue triage a lot, and I'm doing it locally. And that's because issue triage is one of the tasks that takes a while, and it's something that we have to do pretty frequently.

I've actually set this up as an automation, so automations can run on a cadence or on a trigger. And I set my issue triage one to leverage the local model explicitly. I know for a fact that it's going to use just the model on my machine, and it's not going to spend any credits without me knowing. I can have this running overnight, and I'm still getting all of those tokens for free even while I'm asleep.

These are just a few scenarios that I've showed today. Imagine how much more you can save doing these workflows always on, even when you step away.

Thank you so much. I'm going to hand it back to Pavan.

(Applause.)

PAVAN DAVULURI: Thank you, Kayla. That was awesome.

Hybrid Intelligence Windows, powered by HydraFusion, will be coming to the GitHub Copilot app, CLI, and VS Code starting October 15th.

You've seen how Windows is the most secure platform for hybrid intelligence. Now, to make AI a part of everyday life, we need to bring it to places people use the most, into the flow of human ambition. We're moving toward a future where everyone can simply express what they want, and their Windows PC, with their permission, can take action on their behalf.

For hundreds of millions of people each day, Search on the taskbar is where intent begins, so it's only natural our story starts there. At its core, the new Windows Search experience is quicker, more efficient, so you can find your app's settings and files faster.

Starting this fall on Windows 11 PCs, you can take thousands of quick actions directly from Search on the taskbar. You can switch to dark mode. You can turn on Do Not Disturb to help you stay focused. You can turn up your mic volume and then use the inline slider to get to the exact level you want, all of this as fast as you can type.

And actions can do more than just settings. You can take a screenshot, directly arrange your windows all from the taskbar. And my personal favorite, you can send a text message to a friend or a colleague, just letting them know you're running a little late without having to pull your phone or your pocket, a quick, intuitive way to move from intent to action.

(Applause.)

Of course, right from the taskbar, you'll be able to connect to the new Copilot app when your request calls for it. Ask a quick question, and Copilot will bring the answer right there to you in line in Search. If you want to have a richer conversation, you can jump directly from Search to the full Copilot app.

To tell you more about the Copilot experience on Windows, please welcome Jacob.

(Applause.)

JACOB ANDREOU: Thanks, Pavan. Recently, we introduced the new Copilot with three core experiences: Home, Code, and Autopilot. Today, I'm really excited to talk to you about how Windows Hybrid Intelligence supercharges everything that we've just brought to market.

First, let's turn it on. Copilot now has access to three new things.

First, local context. Local context means that Copilot can now ground over all of the files on your PC the exact same way that Work IQ helps businesses ground over all of their business context in the cloud.

Second, local actions. Local actions means that Copilot can actually take action directly on my machine. It can move files. It can change a setting. It has the same level of the ability to control and change things that I do as a user, but always with permission.

Third, local models. Copilot will still use the cloud for the hardest tasks, but for times when cost or privacy matter more, it can delegate down to local models that run directly on your computer.

Let's take a look about how this transforms the rest of Copilot. Let's first talk about Home.

Home is the new starting point for Copilot. We started by bringing together Chat and Cowork into a single composer. We also introduced Office in Copilot, which brings full, rich document

editing and collaboration from Office directly into the Copilot app. And today, Home gets even better with hybrid intelligence. Home now knows what's on my PC. And so, the exact same local files that I've been working on locally myself can now be turned into collaboration-ready artifacts directly inside of Copilot.

Next, let's talk about Code.

Code is the easiest way for anyone to build and share an app, no developer experience required. Hybrid intelligence makes code better in three key ways.

First, you can now build a native Windows app from a single prompt. This means unlimited apps and widgets, all native on Windows, directly from the Copilot app. Building software for Windows has never been this easy.

Next, we introduced MXC support. MXC support means that your code, even locally, is running in the sandbox that we heard about earlier. This means for the first time, not just developers but anyone can write and run code on their local machine with confidence.

And lastly, local models. Copilot is now able to hand work down to a local model as it's coding. This means that projects move faster and cost less, just like we saw in GitHub Copilot.

These three things transform your PC into your personal software factory. We really can't wait to see what you build.

Lastly, I want to talk about Autopilot.

When we introduced Autopilot, we were really excited to talk about the first ever enterprise-grade, long-running autonomous agent. Let's see how hybrid intelligence makes Autopilot even better. We'll flip over to a quick demo.

Over the last couple weeks, maybe a little bit late, I was working with Autopilot to pull together a bunch of stuff to file my taxes. And so, I'll go ahead and just kick off this simple prompt. And the first thing that Autopilot will do is it'll go into my inbox and it'll find the email from my accountant. And it'll look for the specific list of files that we have to pull together.

Now, because of hybrid intelligence, Autopilot can now ground against my entire PC. And the documents I need to file for my taxes are all over the place. They're in downloads. They're in documents. Half are organized, half aren't. That doesn't matter. Autopilot's able to use hybrid intelligence to ground over absolutely everything with local context.

With my permission, it starts to organize. It moves everything into a new folder, pulls things together, and actually starts to rename files. Now, all of this organization renaming is actually happening with local models. My tax details are not going to the cloud. It creates some subfolders here to give a little bit more structure, and then it finishes by zipping everything that it found into a single zip file that I can easily send to my accountant.

Now, Autopilot, of course, always goes one step further, and it also actually goes ahead and composes an email for me to send directly back to my accountant with the zip file that I just created attached. But of course, it pauses here and it makes sure that I'm good to go. I review everything.

This looks great. We can go ahead and send the email, and just like that, all of these files are now sent off and in my accountant's hands, and we're good to go.

(Applause.)

That's Windows Hybrid Intelligence. Copilot now truly understands your PC. It's able to take action on your computer. It's able to keep sensitive work on your device, and at every single step, you're in full control. Really, it's Copilot built natively for Windows. Copilot features that are powered by hybrid intelligence are coming to Copilot over the next couple months.

And with that, I'll hand things back to Pavan.

(Applause.)

PAVAN DAVULURI: Thank you, Jacob. We're excited to bring hybrid intelligence to Copilot.

Windows is the largest open end-user computing platform. Along with great Microsoft experiences, our broader ecosystem is also taking advantage of Windows platform capabilities. We're seeing popular agents like Hermes, OpenClaw, and Perplexity building local AI experiences on Windows.

Here's an example from Perplexity.

Let's say you're an aspiring photographer trying to get off the ground, and you have a folder of photos from a recent photo shoot that you want to turn into a professional gallery for your client. Normally, that means you'd be paying for a service, resizing images, building a page, bringing it all together. That takes multiple steps and multiple tools.

With Perplexity's portable computer on Windows, you can just describe what you want, resize the photos, and build a local gallery. Perplexity takes it from there, finding the photos, checking the tools on my PC, and building the gallery. You define the outcome, and Perplexity figures out the next steps. The model is running locally on the PC. Processing stays on the device without using Perplexity computer credits.

And here it is, 12 photos in a browsable gallery, complete with titles and image details.

We're also excited to announce Muse is coming to Windows as a native app with MXC integration.

(Applause.)

To achieve our vision of bringing unmetered intelligence to every desk in every home, you need a broad range of PCs and silicon, from a mini desktop PC to a commercial laptop to a supercomputer.

Let's start with the mini desktop PCs or (claw?) boxes, as people call them, which millions of people use today. It's a natural home for a device and an agent that needs an always-on device, an agent that keeps working even when you step away.

Today, we're excited to announce a new get-started experience in Windows. I'm excited for this because it's something customers are asking for, a simpler, more secure way to get started running agents. This experience makes setting up popular agents fast just a few clicks right out of the box, no terminal, no complicated configuration.

And for those of you using OpenClaw, we're introducing a native Windows gateway with a simpler setup. And because all the popular agents are adopting MXC, you can now get these agents running securely on Windows.

This is just the beginning. We'll Continue to learn from our customers. Our ambition is to bring secure agentic experiences to the billion people who use Windows every day.

Next, Copilot+ PCs, your PC for everyday AI. We introduced these thin, light, powerful PCs two years ago with power-efficient NPUs that run everyday tasks locally. Today, over 40% of laptops being built for business are Copilot+ PC. Tens of millions of PCs a year. And we're seeing over 2 trillion inferences per month running locally, improving customer experiences across search, imaging, and video calls. The Hybrid Intelligence experiences you just saw from Jacob will take advantage of local AI on Copilot+ PCs.

And then, there's the builder tier. To create a new class of devices for builders, we needed to understand what matters to them and push the boundaries of hardware engineering. I want to introduce you to Surface Laptop Ultra, the most powerful surface we've ever built. Let's take a look.

(Video segment.)

PAVAN DAVULURI: This is Surface Laptop Ultra, we built it for developers and creators, pushing the limits of performance, for who want portability with exceptional craft. I've been using Surface Laptop Ultra for the past few months, and it's an absolute game changer. Developers push their machines hard.

Here's an intensive developer workflow with GitHub Copilot, but a demanding task shouldn't mean putting everything else on hold. While that work continues, a film is being mastered in DaVinci resolve. This kind of mastering takes serious performance, the CPU, the GPU, and memory all working together while the developer workload is still running.

And *Gears of War* which released yesterday, has been running on this machine the entire time, and it runs awesome. It runs simply awesome at 60 frames a second. Work, create, and play so you don't have to choose just one.

Now, it wouldn't be Windows and Nvidia without talking a little bit more about gaming. At launch, hundreds of AAA titles that you love and the indies that you play will be great on RTX Spark, and we're super excited to announce that in 2027, *Call of Duty* will be coming to RTX Spark. Getting here took Surface, Windows, and the Nvidia team building a new platform for the ground up.

When three teams come together, you get something none of us could have built alone, and for that I am deeply grateful. We built Surface Laptop Ultra around Nvidia RTX Spark. It took end-to-end engineering, design, and full-stack thinking to unlock what local AI can do. With up to 128 gigs of unified memory and up to a petaflop of AI compute, you can run models on this laptop that simply don't fit on a traditional machine.

Surface Laptop Ultra is engineered beautifully inside and out. It has more than two and a half times the thermal capacity of our existing laptops, so it stays faster even under sustained load. And you don't have to worry about sacrificing performance when you're unplugged on battery. It can maintain over 99% of its power while on battery power. And that's great because it's portable. Under 18mm thin and less than four and a half pounds.

I also love our new PixelSense Ultra Touch display. It has the brightest peak HDR of any laptop. It looks incredible. Our Surface Laptop Ultra has the most ports on a Surface Laptop. Our customers have told us they love magnetic charging, and they love the versatility of USB-C, so we invented Magnetic Connect, the world's first built-in magnetic USB-C charging. Just bring the connector close, and it snaps right into place. Easy magnetic charging without giving up the USB-C port.

That's right. It's kind of awesome. But the real measure of all of this power is what happens when it's in your hands. When you can close the distance between what you can imagine and what you can create, that is amazing. That matters because the future of AI is not about having just one model. It's about having the right models, the right moment, frontier capability when you need it, smaller, faster models when you want speed, responsiveness, and the room to keep the rest of your workflow running.

Surface Laptop Ultra is built for that future. It brings new ideas to life with advanced graphics, unified memory, and local AI working together. Let me show you how one of our designers put all of that power to work, combining local and cloud AI to bring his creative vision to life.

(Start demo)

DANILO MAKIO SAITO: Hi, I'm Danilo, and I'm a senior designer at Microsoft. In my day-to-day work, I'm building products, but outside of work, I have a true passion for creating concept art. I love creating concept vehicles and sci-fi worlds, and I'm about to show you a workflow that used to take me days and now just takes me a few hours.

One of my favorite tools is Blender. Here I'm building a spaceship from scratch, modeling all the details, texturing, lighting, and rendering. I could keep going, but if I wanted to bring more of my story to life and create an entire world around the spaceship, it would take me days or even weeks. That's where local AI can be really helpful. It lets me explore ideas faster.

Here I'm using ComfyUI, an app that lets me download and run AI models locally. I started by importing some of my renderers as reference images and use a Gemma 4 an image-to-text model to generate a description of the world, and a list of assets that match the style and elements of my images. Then I import another image with close up details as reference, along with asset descriptions generated by Gemma 4.

Using FLUX.2 [dev], an image-to-image generation model, I create an image of each asset. After running several generations, I end up with a folder full of assets. Next, I import my favorite images into Hunyuan3D-2.1, an image-to-3D model to generate the 3D assets. I then brought all these assets into Blender so I can start building my scene.

Using a highly capable coding model running fully on my device, I wrote a basic prompt and generated several layout options using my assets. After choose the one I liked, I kept exploring and refining my scene, adding more elements, and tweaking the overall composition. Once I was happy with it, I render the scene, brought the images into Photoshop, and painted over areas I wanted to refine, adding elements like smoke and adjusting the overall color and mood.

Then I took the finished image back into ComfyUI and used Wan2.2, an image-to-video model to generate the final animation. My final step I used GPT-6 Astra in the cloud to take things even further. I uploaded my final files and asked the agent to build a scene in Unreal Engine so that I could actually walk and fly around in. So this is the result. This will normally take me days, but now only takes hours, all on my Surface Laptop Ultra.

(End demo.)

PAVAN DAVULURI: What you just saw is Danilo using multiple local and cloud models in a variety of different apps, bringing his creative vision to life, shaping the design, refining the details, staying in control of his entire workflow. What used to take Danilo days now takes hours. That's what we mean by Hybrid Intelligence.

The new Surface Laptop Ultra is built for this new era of creative work. Across the models we tested, Surface Laptop Ultra delivers faster time-to-first-token performance for language models and shorter image generation time than M5 Pro-powered laptops. This is where having both the large memory and the petaflop of AI compute on a single device is a distinct advantage. For builders using local AI, that means less waiting and more creating on Windows.

Since Build, many of you asked about the Surface RTX Spark Dev Box. For those of you who want to push local AI development even further, it ships with Project Zenith, a ready-to-code windows experience preconfigured out of the box so you can build from day one. I'm excited to share that both Surface Laptop Ultra and the Dev Box are available today for preorder.

In addition to Surface, we built a broad device ecosystem. Nvidia has been our close partner on this journey. RTX Spark brings powerful AI compute to the laptop; DGX Station takes it even further. Look at the ecosystem bringing this to life. Lenovo Yoga 9i 2-in-1, a lightweight convertible design with pen support. Asus ProArt P14, an incredibly thin design with a 4K OLED display. Dell XPS 16, a seamless glass haptic touch pad with integrated speakers. MSI Prestige M16, a versatile flip form factor with pen and UHD+ OLED display. HP OmniBook Ultra 16 with a distinctive tower hinge design and gesture controls on the haptic touchpad. And of course, the Surface Laptop Ultra.

All of these RTX Spark laptops are available for preorder starting today and will begin shipping next Friday. And finally, we have the supercomputer. DGX station for Windows brings data center class performance together with enterprise grade security and manageability. This unlocks the power to run frontier class models locally, from llama V4 Maverick to Kimi K2.6 to DeepSeek V4 Pro at more than a trillion parameters. Capabilities that once required renting a cluster now in a deskside supercomputer.

With RTX Spark, we're bringing last year's frontier to your laptop. With DGX Station, we're bringing last quarter's frontier to your desk side. Take a look at this. For the first time, DGX Station running Windows live. With Dell Precision Pro, frontier models and agents work locally, taking months of complex work into moments with more control over sensitive data and cost. With the HP ZGX Fury AI station, it shows what happens can work and scale simultaneously across 32 agents, a virtual token factory for development teams.

Both coming later this year, running the Windows platform organizations know and trust. This is an amazing moment. On behalf of Microsoft, I want to say thank you to all of our partners. Next, Satya, Jensen, and Sriram Krishnan are going to take the stage in a few moments to share their thoughts on the opportunity ahead. Thank you.

(Video segment.) (Applause.)

SRIRAM KRISHNAN: Wait, wait, wait! No phone calls on stage here, Jensen. (Laughter.)

First off, was it cool or what? What do you guys think?

(Applause.)

Huge congratulations to Pavan, Jacob, the teams at Microsoft and NVIDIA. I will say they were kind enough to send me a Surface Laptop Ultra for the last few days. It is amazing, but I think I want the workstation for Christmas. (Laughter.)

Now, there's a bunch of stuff about the exciting future to get into, but before we do that, I was looking both of you up last night on YouTube. And it's quite rare to see both of these gentlemen on stage together. And I was thinking about how you both had a history. These two great companies had a history going back decades. You both individually had history of working with each other, going back decades.

It ties to me. My first job was at Microsoft. My first paycheck, a GeForce FX to play *Half-Life 2*. (Laughter.)

Maybe to start us off, and Jensen, what is your favorite memory or fun memory of working with Satya and Microsoft?

JENSEN HUANG: Well, NVIDIA was founded because of Windows. It was 1992, late '92, Windows 3.1, and we were just starting to see the vision of what a personal computer really can be. And at the time, frankly, I'd never even see a PC before. I largely worked on workstations. We had file servers and supercomputers that we used to design chips with.

And the founders and I talked about this future where maybe a personal computer with 3D graphics would enable it to do things like games and be a workstation and design things. And so, we imagined about this world. If not for Windows, NVIDIA wouldn't have been found.

And then, of course, we started in 1993. Windows 95 really reinvented –

(Break for direction.)

JENSEN HUANG: Windows 95 made possible the GPU to be connected to a personal computer, and what revolutionized the GPU was DirectX. Direct 3D was the API that really enabled us to reveal the capabilities of GPUs into applications, but then DX8 came along. And we co-invented with Microsoft the world's first programmable GPU, which is programmable shaders, which a few steps later evolved into CUDA as we know it.

And so, if you look at the entire journey of our company, Windows was at the core of it. And if not for Windows, there would be no GeForce. If no GeForce, CUDA wouldn't come to the world. Without CUDA, Alex Krizhevsky and Ilya Sutskever, Yann LeCun, Andrew Ang, none of them would have found a computer to do deep learning. And here we are.

SATYA NADELLA: And Jensen's now going to do a little transfer of market cap back to me. (Laughter.)

JENSEN HUANG: I do that every day. I do that every day.

SATYA NADELLA: It's quite extraordinary, that journey that Jensen recounted. I mean, it's the history of computing in some sense. And one of the things that Jensen sparked for me was he has been perhaps the most consistent with the vision of what accelerated computing can be.

And going back to the trajectory, you've been the one person in our industry who has had this one idea, whose trajectory has been the most expansive, quite frankly, more so than even any single PC form factor or chip architecture, because I look back. One of my first memories was we were talking about essentially saying, in the cloud how big can be accelerated computing or high performance computing. And the first deal I remember was in Azure, how do they make HPC a bigger thing?

And by the way, if –

JENSEN HUANG: Satya was the first user of InfiniBand inside a cloud, which brought the first supercomputer to OpenAI, which, boom!

SATYA NADELLA: That's right. If we had not done that, it's fascinating. Life's full of accidents, but if we had not done that for HPC, we wouldn't have had the supercomputer to give to OpenAI to go train a GPT model, and rest would have been history. I mean, it was one of those happenstance that we had done the HPC stuff when we were building an InfiniBand-based supercomputer. And comes Sam and says, "Hey, do you have computing?" And we said, "Yeah, maybe."

And so, but the thing that I really give Jensen all the credit, quite frankly, is that consistency of vision of what this can be, not today, not tomorrow, but as a secular trend. And that's what brings us even to this moment.

JENSEN HUANG: And then it led to a conversation. It led to a conversation, I remember quite well, where Satya and I were talking about, so what does this mean to the PC? How do we make the perfect PC in the post era of all of this stuff?

And of course, the personal computer is the ultimate tool. It's my ultimate tool, and for a whole generation of people, it's our ultimate tool. What happens in the era of agents when the agent is on your computer, it is now your personal assistant, but it wants to have access to all of the amazing tools that we have?

But the amazing tools are spread out in the NVIDIA ecosystem, into gaming PCs, into workstations, and that's where most of the tools were. All the (Daso?) tools, and all the Siemens tools, and Autodesk, and Adobe. And you saw many of them just now. And we want the world's best Blender, the world's best Unreal, the world's best Omniverse, the world's best Catia, everything, all sitting on the same PC as the AI. And so, how do we go create that?

That, by the way, took four years. (Laughter.) That was a four-year-ago conversation, and just incredible.

SRIRAM KRISHNAN: Now, that's a good segue to today. And Satya, I think the last few months, we've been inundated with these stories of agents and security. We all use agents. I'm sure everybody here is using Muse or Instinct, and you are giving full disk access, and you're having it access you.

And today, when you guys showed MXC, it seemed to me like a fundamental new operating system primitive for agents, which is a break from where we have been. Maybe talk to us about why the OS is the right place for this, and what this primitive, if you will, opens up.

SATYA NADELLA: Yeah. First of all, remember, this all started with coding agents all running on the desktop. I mean, in fact, if you look at where is the agent that needs this security primitive the most, it's the local harness and it's access to the file system and to be able to take action.

And so, that's why, in fact, working with Jensen and team, we said, look, how do we wire this in into the primitives, whether it's in the cloud or in the operating system locally? And given the fact that essentially, coding went there first, where the harness is local, even if you're using a cloud model, we needed to make the desktop the most secure place for agents to execute.

And so, that's where MXC comes from. It's fantastic to see Open Shell now fully integrated and native with MXC. The controls that Pavan talked about, which is containment itself has to be done at every level. It's not about doing one thing and leaving the other. It has to be at the micro VM. It has to be at the VM. It has to be at the session. The piece that I'm most excited about is the Windows session having containment. It's so, so critical.

Then, okay, containment is wonderful, but then I need full observability. How do you get observability? You get it if you have an identity for an agent that I can track. I can then see the entire exhaust of the agent, all of it, and then I can govern it. That means I can classic set policy and have it govern.

That is now what's going to give peace of mind, whether it's to IT ops, to SEC ops. And by the way, the other thing that we should also not forget is the ability to track the token spend and where it's happening.

And so, that's the entirety of what we have tried to bring together. It's not about any one thing. It's about stitching all these things, having seen all these adoption cycles, whether it's the consumer adoption cycle, the enterprise adoption cycle. I would say the thing that blocks broad diffusion is not having that complete thought on what is required. If you do one and leave the other, there's no trust and what have you.

I feel very, very good that we have brought it together in the cloud, in the client, across IT Sec and FinOps, and now it's heterogeneous. It's not about any one piece of Microsoft technology. It's about supporting it all.

JENSEN HUANG: What Satya just said is going to be the foundation of the next generation of IT as we know it. The words he used, what he described, the architecture, deeply profound for the next generation of computers. Just as Windows and DirectX revolutionized how applications were built, MXC is going to revolutionize how agents are built and deployed. Without it, non-starter. Without it, complete non-starter. And so, this big, big deal.

SRIRAM KRISHNAN: Now, this is, I think, closely tied to the next piece for both the future, which is hybrid intelligence. Now, this feels a little different. I think everybody here uses large model in the cloud. You use it. It runs the tool, maybe using Blender, and you get back a response. We all used that for the last in a few years.

This feels different, where you have a choice of models, some running locally, some in the cloud. You're going back and forth. And one sentence from the presenter for GitHub, which stood out to me was, "You're getting these tokens for free."

And so, maybe talk to us about what this opens up because this feels, again, something new, where you are running part of the workload here locally on your laptop, part of it in the cloud, and intelligently routing between it. What does this open up?

SATYA NADELLA: Yeah. First of all, I think this is one of those moments where some of the things that you've been using all click together. In fact, I was thinking about this. In the late '80s when I was in grad school, I had a terminal, my VT100 and a VAX. And then I had a PC at home, and I was using it all. In fact, it's interesting. On my PC, I had essentially a modem with a terminal connecting to my VAX, and then I had all my local DOS and early Windows and what have you.

And then suddenly, it took all – and by the way, I had Use Net, all other stuff, except it took all the way, maybe until 95, and even Mosaic, the first browser to bring it all together as one seamless experience. It's not about local cloud, what have you. I just want to use my computing power wherever it is in a seamless flow of what I'm trying to get done.

When I was seeing even what Daniel was showing, that demo is mind blowing. I mean, think about that moment. I think we're going to come back years from now and say, oh my God, that was when we saw what he did was he went to Comfy UI, he brought down a bunch of models, built a bunch of assets, took it to Blender, used that to even enhance those assets, then, again, brought it back to Comfy – oh, went to Photoshop, used a local model perhaps to enhance it, took it back to Comfy UI, then took it all to the cloud with Astra, with Unreal.

But he just used apps on his desktop, models, local models in the cloud. It all just worked. That is what this is about. That's where today was about multiple fantastic devices, all these things, but the trajectory for me is so clear.

There will never be a moment where we will go and look and say, oh, this runs locally. This runs in the cloud. You will expect this hybrid intelligence to be everywhere and pervasive. And I think that that's what, to me, is pretty exciting.

JENSEN HUANG: Yeah, that one application, what you just said, Blender was accelerated, Comfy UI was accelerated, Azure was in the cloud. The agents were locally. I mean, that's just the surface area – surface are, that's cool – the surface area of computing was magnificent. It brought together literally the entire might of every single technology company in human history came together for that one application.

SATYA NADELLA: That's right. And by the way, going back to the way you talk about accelerated compute, I mean, his point about why he was excited, what took him multiple days takes him hours, the fact that he was excel in his ambition. And that, to me, is pretty cool.

SRIRAM KRISHNAN: Which I think this ties closely to something else because these models have gotten very good at tool use, but the tools often live locally. The files live locally.

By the way, Jacob and I, both of us are filing taxes at the last minute, apparently. And I think there is an aspect here about I want my data to be secure and safe and control. I want the tool use to happen near where the data is, where if you look at what Jacob showed as Copilot, it really opens up something new.

SATYA NADELLA: Yeah. I mean, to me, one of the things, if you look at it, in fact, when the coding agents became the thing, I was excited to see everybody was in love with, one, the console, and everybody was in love with file systems again. It just shows how powerful of an abstraction a file system is.

I think agents are going to be the largest users of file system. They need to be secure. They need to be, because they're a heterogeneous collection space. And what his tax example, Jacob's, was a great one. Basically, what it did was it organized from, as he said, downloads, from even cloud files, email, everything into a file system container, and then used models locally and in the cloud to go do stuff with it. And that is the new computer.

I use the computer. My agents use the computer. The agents need these primitives as well. And so, I think OneDrive. When was the last time I thought of OneDrive as a cloud thing or a client thing? It's there everywhere.

And so, to me, that is what I think we are beginning to see, is, hey, I want my primitives for my agents to use 24 by 7. The form factor when I'm using it on my PC should feel like I'm using it on my PC. When I'm roaming, it should roam with me.

JENSEN HUANG: What's amazing, as you know, is that these agents are going to be better users of tools than us. And the reason for that is just, we know 10-15% of the features of any application, but the agent knows every single feature.

SATYA NADELLA: That's right.

JENSEN HUANG: And so, the fact that the tools, these incredible tools, are sitting on your PC, they're accelerated in this incredible computer, Surface Ultra, now the agent has access to all of that capability. Who knows what we're going to be able to do? I mean, just look at what it did. It was the Blender, the Unreal, the Comfy UI. It was incredible.

SRIRAM KRISHNAN: Yeah. Show of hands, how many people here have tried using one of these apps locally with the model in the last couple of months or a few months?

Great. And so, I think this just goes to show where you can actually bring the tool use locally.

Now, Jensen, I want to ask you about the hardware piece of this. We very often see you talking about these big, massive Vera Rubens, Blackwells, NVL 72s, these intelligence token factories that power data centers.

Here today, we're talking about something different. We're talking about the Spark, the workstation, which I really want for Christmas, which runs in your office. Talk to me a little bit about how you see this evolution about tokens being produced right next to you, as opposed to Vera Rubin in the data center.

JENSEN HUANG: Well, the number of engineers that worked on this project is incredible, over 4,000 engineering years. This is a massive project. I mean, look what we've done. We've reinvented the personal computer completely. For the first time, we took all the capabilities, what used to be in a gaming PC, in a laptop, in a workstation, in a supercomputer, in the cloud, we unified the architecture and we made it run on one super chip.

This super chip, it's rectangular because it's two giant chips fused together. And so, it's a beast. It has one petaflops. Just put in perspective, the computer that started the AI revolution called DGX-1 was one petaflops. And it was a quarter of a billion dollars, and weighed 500 pounds. And now, it's in your laptop.

But what's amazing is this is also the only computer, the only computer in the world that runs DirectX and all of its applications, Open GL and every single version of it, because we need all of the tools and design tools, creative tools. We need everything to run. We need Unreal to run incredibly well. DirectX has to be amazing, and it runs every single CUDA application incredibly.

What Satya and I were talking about four years ago is we're going to have a whole new generation of software developers, and these software developers are going to be AI software developers. They're either going to be creating AI applications and AI models, or they're going to be using AI to develop their software.

And so, the computer has to be revolutionized, and it has to be really good at CUDA from the ground up. And so, this is the only computer on the planet that runs everything natively on the device, the first time ever. And so, it took thousands of us to do this, but it also took the entire industry.

Look behind us, every single laptop maker, every single desktop maker, every single workstation maker joined us.

SATYA NADELLA: Yeah.

SRIRAM KRISHNAN: Wow, yeah.

JENSEN HUANG: This is the first time, I think, a new piece of hardware, a new computer architecture, have come together, and the entire industry, top to bottom, left to right, everybody's involved, every single geography. And so, this is a complete reinvention.

We understand the seriousness of the work that we're doing here. We've now tested some over 1,200 applications, the most intense applications in the world, all fully tested, performance

tested, compatibility tested. Everything has been optimized. We're going to keep on going. The support for this system, the amount of software we're going to bring to it, middleware, algorithms, I mean, it will never end.

This is just a magnificent thing, that when you when you finally look at it, you look at this beautiful laptop, Pavan stands up here with this beautiful laptop, but the number of engineers between our two companies and all of the companies around the world that did this, incredible, all because we believe that the personal computer was our tool, tool for people. Now, it's going to be tool for people, but it's also going to be our personal assistant that uses the tool.

SRIRAM KRISHNAN: Now, one last thing for both of you. I think if you've been on X for the last few days, there has been this intense conversation about, I would say, personal agents. We've been talking about productivity and maybe enterprise agents here. Whether it is Muse, Instinct, there's a question about, hey, when this agent goes out and makes a purchase, is that me or the agent? Who is the liable one for it? And if I'm a service provider, how do I react to this human being no longer showing up on my website? And people are trying to build standards. And as an industry, I think we're trying to figure this out. So I'm curious if you kind of have to peer into the crystal ball a bit. Where do you see this headed? Is it standards? Is it the industry coalescing around some new behaviors? Where do you see this?

SATYA NADELLA: Yeah. I mean, first of all, I think it starts with a little bit of the conversation we had in the early part, which is trust. And trust starts with things like, hey, are these agents contained? Do they have identity? Is the exhaust of the agent with the identity separate from the user?

Which is one of Pavan's thing was how do we make sure the agent ID that's working on behalf is separate from the user? And you need it for every day usage, but you also need it for what I think you're referencing, which is how do I then trust? What do I delegate?

In fact, if you look at the first decision that we are making is what is my delegation authority that I'm giving to the agent on my behalf? And that is a decision that's going to be based on trust, and it'll keep increasing. The more autonomous it is, you trust it and go. Then I think on the consumer space, this will come down to -- I think of it as insurance will price this.

SRIRAM KRISHNAN: Insurance?

SATYA NADELLA: Because at some level we can all talk about, it's not like OAuth is hard to do. I think the protocols are good, we should do them in tech, but ultimately it's going to be pricing essentially the agent working on my behalf, and how does the marketplace price it?

JENSEN HUANG: Your question at the foundation of it, MXC is so foundational, and OpenShell, the work that we're doing together is about trust, it's about containment, it's about observability, and monitoring. If we can't get that right, we just can't get it right. So that's the first part. We have got to secure that part of it so that the agents that are operating on our behalf doing things on our behalf, we give it minimum rights, it has credentials, it is monitored and trusted. Only then can we allow these agents to really go off and do some task on our behalf.

SRIRAM KRISHNAN: Today is a powerful moment. I remember when OpenClaw came out from there with the terminal hacking together your own gateway to here, where it is an embedded piece of Windows with tokens running accelerated. It's quite something.

JENSEN HUANG: So much engineering went into that, though, today.

SRIRAM KRISHNAN: Amazing. And it is a great device. Let me just tell you that honestly. Well, Jensen, Satya, thank you so much. It's a great day. Thank you for having me, and I'm going to throw it back to Pavan.

JENSEN HUANG: Satya and Microsoft team, we have reinvented the computer as we know it. I just want you to (Inaudible.)

SATYA NADELLA: Thank you so much, Jensen. Thanks for the partnership. Really excited today. And I think, as you said, one thing if I can end with it. Jensen just in passing talked about the vision for this was, how do we -- I mean, Windows has always been about how do we not leave any piece of code, any piece of framework behind, and bring it all? The fact that he talked about, "Hey, we're going to bring all of Windows apps, all of the graphics apps, all of CUDA, all of OpenGL," and now with the token factory. That is what is mind blowing.

JENSEN HUANG: Windows has been with me for 34 years. Looks like it's going to be with me for 34 more.

SATYA NADELLA: There you go. Thank you so much. Thank you. Thank you.

PAVAN DAVULURI: It's a bit hard to follow, but an absolutely incredible moment for our industry. Just to wrap, we're building the OS for Hybrid Intelligence. It provides the most secure platform for agents, unlocks unmetered intelligence. We're bringing AI into the ambition of humans and their flow. And of course, we're delivering it all on PCs that bring it together. Thank you all for your time today.

END