(no title)

64 MIN · YOUTUBE · HTTPS://WWW.YOUTUBE.COM/WATCH?V=VETQSCPCDGG
https://www.youtube.com/watch?v=VeTqsCpcDgg

SUMMARY

Amin Vahdat, head of Google's internal infrastructure, discusses the complexities and challenges of managing large-scale computing resources, particularly in the context of AI and machine learning. He emphasizes the importance of reliability, efficiency, and the balance of system components in delivering value to users, while also addressing the evolving landscape of energy demand and infrastructure requirements.

- *Google's internal infrastructure is among the largest on the planet, aiming for tens of gigawatts of capacity.*
- *The focus should be on delivering value per dollar spent rather than merely increasing capacity.*
- *Reliability and efficient utilization of resources are critical; underutilization can lead to wasted investments.*
- *The shift from training to serving AI models is changing how infrastructure is planned and utilized.*
- *System balance is essential; having sufficient compute, memory, and network resources is necessary to avoid bottlenecks.*
- *Energy availability is a major bottleneck for scaling infrastructure, and innovative solutions are needed to address this challenge.*
- *Google aims to be a positive asset to local communities and the grid, emphasizing sustainable practices in data center operations.*
- *The future of computing will likely involve more specialized hardware to meet diverse workload demands effectively.*

Thank you so much for joining us, Amin. Please give me a round of applause for Amin Vahdat. >> [applause] >> You guys have no idea how hard it was to get Amin to show up. Seriously. This is the one lecture that um I've been super excited about. And Sebastian, who many of you know, um who is my co-founder on Amp, wanted to be here and he's so bummed that he couldn't because he's busy working on the cluster for your guys' final projects.

Uh Sebastian worked on the Borg X Borg GQM scheduler, that designed that, too. So, we are very much uh a a Google family over at Amp, and so Amin's a bit of a rock star in in our kind of lore. Um so, to give you guys some context, Amin what is the head of basically in charge of the internal infrastructure at Google. The TPUs that make Gemini possible really would not be at anywhere close to scale they are at if it wasn't for Amin.

Okay, so pay attention to every word he says. Like think about him as the opposite of Jensen. You know, Jensen like is a rapid-fire high-throughput LLM. Um think about Amin kind of as like the distillation of like three frontier models who have been trained on like frontier like the in practice and discipline of infrastructure for the last How long have you been doing this, Amin? Coming up on 30 years, I'm sad to say. 30 years. And so, every word Amin speaks has like every token that he produces as an LLM has like universes contained in them.

Okay? And we we will probably not understand what he actually means for years. So, I'm I'm this is going to be recorded and put up on YouTube cuz I think years from now people will look back at his lecture and realize how profound his influence was on the on the industry. Um you know, to concretize that, uh how much uh compute does the internal pool at Google have today, I mean? I'll start out with the easy question that I can answer.

Um I've seen some Twitter posts that say we have among the largest computing infrastructures in the whole planet, and I think that I'm I'm willing to stand up behind that one. Okay. Would you say it's in the tens of gigawatts? Tens of gigawatts. Um I will say that we are aiming for tens of gigawatts. >> Over the next 4 years, it'll be well in the north of tens of gigawatts. >> Over some some time period, yeah.

>> Yeah. So, we crunched the numbers this morning. We think about 1 gigawatt to build out is about how much? Okay, so what 1 gigawatt is about $40 billion of infrastructure. Do the math. Okay. And as much as I hate to say it, our means infrastructure org is literally one of the most efficient on the planet. Because you know, there was a time when I was starting out Amp, and we were looking at how much single cluster utilization was across the industry, and some of our portfolio companies, you know, some of the speakers here, were running them at 70, 80% utilization, and some of the other big tech companies were similar, in fact worse. I'm sure you saw that. Um you know, the Colossus cluster is not running at peak utilization, and I think it's at 11% MFU, which is honestly MFU is kind of hard to get up. But at Google, my understanding is if the if the node allocation is less than 96%, it's considered a major outage. Yeah, so I think what what this uh really points to is when you hear numbers like uh $40 billion uh per gigawatt, and I've heard numbers like $50 billion a gigawatt from other sources, the numbers are going up.

Things are getting more expensive. I think the most important consideration isn't how many gigawatts do you have, it's how much capability and value you're delivering to your users. And this is something to really be aware of. In other words, if I've got a gigawatt here and a gigawatt there, they're not the same. How much reliability you have actually really really matters. Like I could go spend 40 50 billion dollars on a gigawatt and if I don't do the work to make sure that every one of those nodes is super reliable, so a gigawatt is let's say that's 150 200,000 TPUs, GPUs, it could be whatever you want.

One of those go goes down, maybe your whole computation stops. If you're not A making sure it doesn't fail, B when it does fail, figuring out which one it is and getting it repaired really fast, you just wasted a lot of money because your utilization and what we call your goodput is nowhere near what it it needs to be. If you have the TPUs deployed, but no one can schedule a job on them, it doesn't matter how much money you spent on So, I think that a lot of these measures are actually broken.

The measure isn't how much money you spent per gigawatt, it's actually how much value you deliver per dollar. And if I can spend half the money, deploy half the capacity and give you the same capability, awesome. Better if I can deliver twice the value from that gigawatt, I now need to build fewer gigawatts. Okay. >> Or I can only get so many gigawatts. Energy's massive problem. And um you know, we had Jensen here last week and one of the questions I asked him is how do you He said something similar which was Is this why everybody's laptop is signed by Jensen?

>> Yeah, basically. >> [laughter] >> But you should get it Well, no no Sam is going to yell at us for trying to get signatures. Okay, we got yelled at as you guys know because of physical security. >> have a GPU by the way signed by Jensen, so I'm That's a long line of it's it's a tradition, right of passage. So, how do you measure intelligence, You know, output per unit of input, right? It's ultimately what as a systems person we're trying to optimize. And if the output is this very heterogeneous output, which is coding tokens, image tokens, and so on. Like but the input is this generalizable input called compute or flops, so to speak. What How do we reconcile the fact that the the eval's are just different?

>> We're we're it's a tough close to impossible question to answer. We are working on benchmarks that measures intelligence per dollar, actually. And we publish some things externally. I can send folks references out of Google broadly. Uh that captures this question of intelligence. And then it's intelligence per dollar. But what I really want to emphasize though is that it is um how much you're actually getting out of it. And so another way to look at it is per gigawatt, how much revenue are you generating?

Maybe revenue's not the right measure. How many daily active users do you have for your service? So it's not how many gigawatts do you have. It's >> daily active users per Okay, got Right. Like if I'm doing Gemini app, and I have a gigawatt behind it, no one cares that I have a gigawatt behind it or two or four or half. It's how many daily active users do I have who are happy? I have many and then how's that growing?

Okay. Right? And if and now the question is how do I deliver So this is where the efficiency part comes in. Yeah. I want to make sure that every TPU is up. But by the way, if I have a bunch of TPUs and I don't have the compute and the storage and the networking to go along with it, then it doesn't matter how many TPUs I have, especially in the age of agents. You actually it's a orchestration of the whole.

Right? Because if I'm having all my expensive TPUs sitting around idle waiting for an agent to finish running its simulation through a CPU that have to go get some data from the storage that might be in a whole 'nother region, that's a problem. Okay. So it's the orchestration as a whole. I think there's too much fixation on how many gigawatts of capacity we have. By the way, I I spend a lot of time making sure that we have a lot a lot of megawatts, a lot of gigawatts of capacity, so I get it. But, there isn't enough on how much value are you getting out of it? Are you extracting the most utility out of every machine that you build and deploy? And so, if the if you've closed the loop to say, I think what I'm hearing you say is the the eval is the business metric that matters. In the case of Google, it's daily active users or whatever for the Gemini app. But, the challenge as an infrastructure person is which you have a extraordinary history and background doing, is you're always trying to general design general primitives, right? That are not overspecified for for a particular output. Yeah. And if intelligence is a humanity-scale measure, then how do you reconcile the difference between designing an infrastructure primitive that's general for all of humanity, but that might not align with the specific measure of intelligence that matters to Google? Does that question make sense?

>> It makes sense. I think it's a uh phil- great philosophical question. The good news is in practice, what we do care about are the uh business outcomes because we have to believe, and it turns out to be accurate, that people are going to vote with their feet and use the services that are giving them value. In other words, if we have

Gemini DAOs, and they're growing at a certain rate, uh for whatever reason, if it's competing against ChatGPT or Claude or uh Grok or whatever else, if people are using it, they're voting with their feet, they must be getting the intelligence and the utility that they need. If they're using uh coding in one scheme versus another, if we're delivering the value. Now, a lot of this does come down to how many flops do you have? How much HBM bandwidth do you have? How much ICI or NVLink or whatever else bandwidth do you have? All these low-level measures matter, but in the end, what it rolls up to is happy users, paying enterprise customers, uh developers who are getting their work done. That's what we're trying to maximize.

And so, if we have capacity that is sitting around idle, that's a bug. Right. Okay. Got it. The value that's delivered is a great metric. And so, what we have to now make sure is when we have these gigawatts of capacity, the infrastructure layer is fascinating because there are thousands, millions of things that can go wrong. You You know this very well. And each of them, unfortunately, matter. And so, it's about systematically going after it. And And so, in other words, there is no major breakthrough when we say, "Hey, um if in going from 99% availability to 99.9% availability, super hard."

I You want to think 99% reliability, that's pretty good. If you think about it though, that means that 3.65 days of the year you're down. That's not good. Right. In fact, might be unacceptable. Now though, I want to go come back to power for a second because power often times is your uh biggest constraint. You talked about 11% um MFU. If you look across all the fleets, I won't tell you what the numbers are, but if you look at the amount of power provisioned at the edge of a data center region, and how much power is actually used by the compute, it's probably a lot lower than you want it to be.

Reason number one, over-provisioning for reliability. So, in other words, to really get to what the power uh service wants, which is five nines of availability, which means 30 seconds of downtime a year, you basically have to have 2N. One plus one redundant feeds. One goes away, the other basically switches over immediately. That means that half your power capacity is not being used at any given point in time. That's what it takes to deliver five nines of reliability.

Now though, what if you go to your customers and say, "Hey, would you rather have 99. Let's say 9% reliability, and double the capacity, or 99.999% reliability and half the capacity. Historically, the answer would have been give me the five nines. I can't take the outage. Today though, if you go to the frontier labs and say, "Would you rather have twice the capacity, but then 3.65 days of the year or 0.365 days of the year you don't get any of it?" They'll say, "Oh yeah, sign me up."

Give me more capacity. I'll take the downtime. Is that a new phenomenon or is that a recent phenomenon? It's a recent phenomenon, right? Because again, now if you're if you're delivering historically, if you're delivering an enterprise grade service, it's five nines. Can't be down. But training a frontier model, it's about throughput. You'll take the downtime for a day or two days or three days a year if the other 362 days of the year I'm not speaking for everyone. I'm just But by by and large, your customers are telling you your internal customers are saying Internal and some external. We will take access over reliability. Yes.

This is a fascinating new development, but now even getting to that 99.9% thousand things can go wrong because the thing I want to emphasize is if we're serving a frontier model, that's hundreds, perhaps thousands of TPUs or GPUs, it doesn't matter. If we're training, it's tens of thousands, perhaps more of the same accelerators, but the computation is synchronous. What this means is that basically all of the TPUs, all of the GPUs are talking to each other synchronously. They're distributing data, all reduce, all gather, whatever else it is. One of the nodes goes down, everything goes down.

So, literally is in again, how how do we build internet scale web services to this day? Today to dates, if you're building web search, it's designed basically to have any rack go away at any point in time and no one notices. We barely notice. We do notice. We we'll go get it fixed, but there is no uh no outage. Why? Because we have a backup for all the data on that rack and at least one other place in that same cluster. And we have spare compute capacity and it's fungible.

So, if you think about TPU or GPU training inference, every node is special. Every node has a special specific expert whatever layer in the overall model that it's serving. If it goes away, propagation stops. Serving stops. So, how you manage these things that actually deliver the value at scale completely changes. And so, everything that we developed over the past 20, 25 years, that said, loose coupling, don't worry about individual failures, all that's gone out the window, too.

Do you believe flops should flow like megawatts? Well, they're they're closely related, but as you said, um what what I really believe is uh system balance is what matters most. And so, if you are over fixated on flops and you don't have enough HBM bandwidth or if you don't have enough SRAM or if you don't have enough network bandwidth, then it doesn't matter how much flops you have. Like, we we we can build infinite flops and not connect it and connect it via via thin pipes to one another or put very little HBM bandwidth or very little HBM capacity. That's easy. Scaling flops is easy.

Building a coordinated supercomputer that scales out to 10,000, 100,000-ish TPUs that has the right balance point, super hard. And this balance point is the the key key insight. So, um I'll share with you all um I I I used to be professor. I I love this room, by the way. Uh seeing seeing this room, I I I took undergraduate classes in a room like this about a another school up the road at Berkeley. >> [snorts] >> We're we're we're we're we're equal opportunity systems people, right, guys?

Yes, it turns out Berkeley does pretty good work in systems, as well. >> Yes. Uh your work is great. But uh one of the things that I loved learning most about and that has really stayed with me, I'll share it with you all in case you don't know it, is uh Amdahl's law. Who here knows about Amdahl's law? Oh, no. No, Amdahl's law? >> Amdahl's law. Okay, good. Sorry, I failed as a professor. Please go ahead. Okay, so the Amdahl's law of system balance, basically this is late '60s, so before I was born, um he came up with this law that said for every million instructions per second that you built into your parallel system, your your distributed computation, you would need um megabyte per second of IO.

So, in other words, if you're going to provision a million instructions per second, think of it as flops today, you

better have that IO to back it up because compute without data is useless and you have to be able to feed it. And now, shockingly, over just a it was 1967 he came up with this, so almost 60 years, this has held. Now, he was building small scale in the late '60s. Now, we're talking about 10,000, 100,000, sometimes spread across even a wide area network.

You have to provision a network because almost all your data is across the network today. So, your IO is network IO. You have to provision for every some number of flops, some amount of HBM bandwidth, some amount of network bandwidth, or you're going to starve, you're going to uh basically waste your money. If you don't build to this ratio, you'll have huge [snorts] amount of flops that aren't doing anything. To some extent, this is what's happening today with the very low AMFU utilization that we have.

Why? Because with the with the move to mixture of experts, sparse computation, actually the hardware today, all of it, actually, isn't built at the right system balance point to manage the fact that actually you now need a lot more memory bandwidth relative to the computation ratios. Mhm. So, when you think about evaluating your system's utilization, super key. That I mean the reliability part, I really want to get this across, but then system balance is also super key.

If you don't have the right system balance, you're wasting your money. So when you say 40, 50 billion dollars per gigawatt, yes, but if you had to spend 55 billion dollars and make sure that that gigawatt was balanced or reliable, you'd do it. So I think the key here is because otherwise you're not going to get the value out of it. If you say, "Hey, I put one with my gigawatt, I got all these gigaflops, teraflops, petaflops, exaflops, yottaflops, whatever it is."

Awesome, but what do you actually get out? And what you get out depends on system balance, and it depends on reliability. But now, going back to the agents, system balance isn't just for your TPUs and GPUs, it's the balance to the CPUs that are sitting next door, the storage that's sitting next door or in the next rack, the network that connects it all together. Like the did not not the high-speed NVLink or ICI network, but the data center network that connects it all together.

It breaks my brain a little bit to try to figure out how do you decouple the individual bottlenecks in the memory storage bandwidth uh supply chains and align that in a predictable fashion to accomplish system balance. How does one even approach that problem? So this is for those of you who took architecture, undergraduate or graduate architecture, you've got your seven-stage pipeline, right, with the instruction fetch and decode and access, right? And how do you actually That's how we got super scalar performance.

>> Right. Seven stages, super complicated within the core. Now we've got like 127 stages. >> Right. Within a CPU is possible to get that microarchitecture more or less balanced, but even there hit getting the right balance point is a super tough. That's why you get pipeline bubbles. That's why you you say, "Okay, how many cycles per instruction do I really have and how do I drive that uh down actually?" Okay. So, now extend this out across 100,000 nodes, it is an impossibility. 100% MFU is not possible.

So, that that should be like there's I mean you could with a toy uh just like shot it out and say, "Go." But, in general for a real computation, you're not going to get perfect balance because there's like let's say there's just little micro variation in one cache hit rate of one TPU GPU versus another. That will cause a pipeline bubble. All right, so now your MFU because now you're waiting for the data to come from another node, your MFU just went to So, you have this compounding Yep. and it'll multiply.

And let's talk for a second cuz what you described is the computational one. Like I'm talking about now you add on network. No, no, procurement. Oh. >> [laughter] >> Like how do you like I mean literally the the world can't produce enough memory. I uh Yes. I'll ask you if this is true or not. There's there's reports that one of the frontier labs cornered the market on memory recently through a buying a bunch of call options and then the rest of the industry revolted. Is that true?

I don't know if it's true or not. I I read the same uh or I I I can't um keep up with the X. So, I have the same feet whatever it This is from a group chat this morning, but I got >> morning. Okay, this this actually came out um three or four months ago. Oh, then this the group chat is behind. It's a This one in this particular case the group chat is behind. You know, these things uh yes, at the supply chain is a massive massive issue. I'm not responsible for the supply chain and and procurement. Uh the problem is that things just continue to go up and up and up every month and the lead time is years. So, in other words, basically if you want to say, "I want a gigawatt of capacity." If I want a net new gigawatt of capacity my lead time is somewhere around two or three years.

It doesn't matter if I've got my 40 or 50 billion dollars. Just for buying everything and building it, it's a very physical process. So, gigawatt into end, I got to go get that uh capacity of power somewhere. We have a final project here, which is the one-person frontier lab, and they have increasingly less time, but look, the the project is a microcosm of life, and what you just heard is A mean saying there's a bottleneck he can't throw more money at to clear. Sure.

So, if you could prompt them to solve it from a technological perspective, what could they do to help unblock that that bottleneck? And we're going after it on on multiple fronts, because pulling that in, in other words, if I had the ability so many times so many times, actually, if I had the ability to go spend more money and get more capacity tomorrow, it'd be an easy decision. But if you're saying, "Hey, you now have to commit to how much capacity you want in 2 years' time."

Commit. Like, you can't No going back. Today, you have to say exactly how much capacity you need in 2 years' time. Okay. Basically, there's going to be one of two outcomes. There's a third that's infinitesimally small probability. Outcome number one is you predict too little, and then you're going to be really upset that you're leaving opportunity on the floor. Outcome two is you over predicted, and now you wasted a bunch of money. There's some other possibility which says you predicted perfectly, which never happens. So, if you could pull that in, and now you said, "Okay, how much capacity do you need tomorrow?"

you're probably going to nail it. Or if you over predict, you over predict by, you know, .05% or something. >> Mhm. How do you pull that lead time in? And actually, this is a technical problem. This is a truly a technical

problem, where from procurement to manufacturing, like, right now, if I wanted to have a gigawatt, I'd have to go build a new building. A big building, probably multiple buildings, actually. I have What does that mean? I have to go now get some land.

Maybe I've got some land buffered up. But if I don't, I'm in trouble, because I now have to go do permitting. That'll take months. >> Right. Indeterminate. Right. Who knows, etc. So, may but by saying, "Okay, well, you know what? The land is kind of cheap. So, let me have a bunch of land on the side." Okay, now is the land prepared for a building to go down? Actually, you probably have to grade it. Okay, let's let's go ahead and spend the money to grade it ahead of time, too.

Now, I'm ready. But now, I put down the pad. Do I go procure the power? That starts getting expensive. I do I go to the utility? The utility now, everybody's going to the utility saying, "I want a gigawatt. I want 5 gigawatts. I want 10 gigawatts." They'll say, "Sure. I'll get you that. But you have to agree to pay me for all of that for the next 20 years." You want a gigawatt? Sign this contract that says you will pay me for a gigawatt 24/7 for 20 years. Why? Because there's no capacity to back it on the grid anymore.

It used to be if I went to the utility and said, "I want a gigawatt." They'd say, "Sure. I've got a gigawatt." Well, I wouldn't go for a gigawatt. I'd say, "Give me 10 megawatts." And they'd say, "Sure. 10 megawatts, no problem. I've got that. You don't even have to You don't need to sign a contract. It's so much slack capacity. I'll get you 10 megawatts." >> [clears throat] >> No No longer true. But is My understanding is the reason grid-connected capacity is so acutely under supplied is because hyperscalers are saying, "Well, we only want sites that are expandable." Mhm. And so, everything under 100 megawatts is just stranded. Yes. But that's a bunch of stranded unutilized capacity in America. What if you could If you were the chief energy officer of America and you're trying to, you know, drive up utilization of those stranded assets, what what would you do? Um so, I think the 100 megawatts if if you look at it, it'll add up to something, but it's not going to add up to the majority of the of the demand. I think that just from a scale and operations perspective, if we really want to go after this, we actually would We should unstrand some of those 100 megawatt sites, for sure. I think that as serving takes off, that will happen naturally. I see. So, in other words, we are up until recently in a place where most demand was for training, and training does need large contiguous chunks of infrastructure. As we move to more and more of the demand going to a serving, that's going to shift naturally. Because the serving is more fungible.

>> It's more fungible, it's smaller. I don't need a gigawatt to do training. I don't need 500 megawatts to do training. I can serve some number of tokens uh per minute coming from a smallish deployment. So, I think we're going to understand that somewhat naturally, but I don't think that's going to fulfill the needs because there is going to be benefits uh to scale. Uh and we are going to have to figure out how we get larger amounts of power concentrated delivered to some number of locations.

>> Yep. Makes sense. I could go on for hours. I mean, uh but we should get switch to questions. The question is, if you were Stanford student again, what technical problem would you obsess over? You know, I I will say that uh I get a uh this I think it's a really good question, but the answer I'll give is to go the all of them really, really matter. Honestly. In other words, there is no one bottleneck. And predicting the future is really hard. So, let me

give you an example. When I was a graduate student, uh what uh everyone said is absolutely, positively don't work in artificial intelligence. Like it's this the worst thing to work work in.

And that that was true again after 10 years, and then true after another 10 years, and now look what's what's happened. Trying to predict the future really, really hard. I I would say pick the problem domain that you are most intrinsically excited about. Because that that passion for it is that's that's what's going to carry you forward. And then um in this model, I would say everything from algorithms to hardware engineering to chip design to operating systems to model architecture to it all matters.

Which which is really good. So, probably pretty good chance that uh what you pick is going to be really really important. And so if you And if you pick something solely because your prediction is that it's going to be the most important one, but you don't like it, I think that that outcome will be the bad outcome. Because also pretty good chance that you'll have mispredicted. I [clears throat] have a quick I have a quick question based on the You know, many of you submitted your your project ideas and there were 500, so it's taking me a while, but I'm steadily reading all of them cuz I don't want to have Claude hallucinate. How many people here feel like you picked a project idea because you were truly intrinsically motivated by it?

Good number. Okay, that's actually very helpful. Wasn't clear to me based on the readings cuz there's surprising similarity between many of the problems you guys are interested in. And I wish we were seeing more diversity in in those problems, but that's that's for another time. Next question. Question is what's your favorite story from your time at Google? There were a lot of uh favorite stories and um yeah, thank thanks for reminding me of the great time I had at Duke as a as a professor.

You know, the the stories that are I mean, we've had of course many joyous moments, many funny moments, um but I think that for me, um the moments that are best are the ones where you learn the most. And so the one that actually comes to mind just top top of mind is when the original TPU V2 design was happening. And we were going to go build this supercomputer at the time, 256 nodes, it's gotten much bigger, over 9,000 nodes now.

And we were debating what um network to use. This was around 2015, what network technology to use. And uh you know, my primary area of research understanding at the time was networking. And the conventional wisdom from 45 years or whatever at the time of networking was any whatever you were going to do in networking, you were going to use Ethernet. And some really smart folks said, "No, this domain, we want a distributed shared memory system, read-write semantics, point-to-point, not switched.

And Ethernet is the wrong solution." You know, I I was like, "What what the heck? I mean, look, I I have 40 years of history behind me and always been right. Me and me and a thousand other people have always been right." But then when I When we dug into it, back and forth, and it was one of these super spirited debates. Not a not an angry debate, to be clear, right? But it was a, you know, smart people, whatever you want to say, really going at it. And really convinced um uh that they were right.

So, it turned out I was wrong. It turned out that actually you don't want to use Ethernet for a TPU supercomputer. And that has stood the test of time for the past decade. Um I got it wrong. I I learned something. And And so, the best thing about Google, actually, I would say is uh how often I get to learn something. In that story, who was the person who was the first principles thinker that came to that conclusion first and then evangelized that standard? Hard to say, but probably Norm Jouppi.

Stanford PhD. So, yeah. Yeah, Norm is >> Maybe maybe he learned something. Um next question. Question, what What was it like during the ChatGPT code red? You know, I think this was it was a great time. And I think it remains I think that Google has changed as a company. I was This is um when I really first started seeing Sundar in action up close. And I now report to him. I didn't at the time, but one of the things that he did in that moment was he did a fairly big reorg. Uh the biggest part of it was bringing Brain and DeepMind together. Probably many of you have heard that. Uh it was a fantastic move. He also brought different infrastructure teams together.

Uh under my leadership. that was the the lower headline, but I think also turned out to be a good move, not because of me, but because it allowed us to move with sort of more speed and more unification. I I I would say that seeing how the people came together was was really fantastic. The culture at Google is different than it was 3 and 1/2 years ago. I would say it's been a reinvention. I think that we're actually through that now. You know, if you'd ask me a year ago, I'd say that we were through it, probably not. I think we're now at this point through it. Sundar deserves a lot of credit. Demis Hassabis and Jeff Dean deserve a lot of credit for for it as well, but really I I'm I I speak of November 2022 often actually internally and frankly fondly.

I can repeat the question if you if you like. >> do. Yeah. So, I think one of the premises networking is a bottleneck at all all layers. We at Google have been leveraging uh optical circuit switches to remove that bottleneck. And so, are is are you worried am I worried that we're going to limit ourselves given the fact that we can't reconfigure these optical circuit switches at per packet granularity? Is that assumption Sorry, I interrupted you. Go ahead. Yeah, go ahead. Uh good question. So, we we don't restrict ourselves to optical circuit switching.

Optical circuit switching plays a role in our networking, but I mean the the lecture which you're referring to, the presentation I made in terms of all layers, for instance you would not use optical circuit switching for the on-chip network. No way. Not not applicable. And you would not use optical circuit switching for portions to large portions of the WAN. But even within the data center, where we do use it extensively, it's not the sole technology. It's an augment. In other words, we have a lot of electrical packet switches, a lot of electrical packet switches. And if you look at the TPU, within a rack, it is a point-to-point network, but every connection today between TPUs within a rack is copper.

Like there's a direct cut because that is the right technology. Between racks, we have optical circuit switches. But the optical circuit switches essentially creates today a three-dimensional torus. Mhm. Why do we do this? The reason is reliability. So, if you think about it, if I lose a TPU, I now have again lost my entire um lattice. If information is flowing through this torus by pairwise connectivity, I lose that one TPU, everything is gone away.

What I can now do with my optical circuit switches I can remove that rack wholly. I can plug in another rack. And those within a rack today we have 64 TPUs. Those 64 TPUs can take in the exact position of the 64 TPUs that I took out. But what does the optical circuit switch do? And this would require some pictures and um uh some slides probably. Basically, what it then says is imagine that I have the ability to take fiber, unplug it, replug it to another rack without any humans.

That's what the optical circuit switch does essentially. So, what is a optical circuit? It's a um chip about this big, square. It has 136 mirrors on it. More could be more, could be less. Each mirror can be rotated in three dimensions. Essentially, what we do is we take every rack and all the fiber that's coming out of that rack will be connected to the optical circuit switch. The fiber, now it's light shining out through the fiber, comes into the optical circuit switch shining down on those mirrors.

So, light comes in, hits a mirror, gets reflected in a particular direction depending on how I rotate the mirror under Mem's control. These tiny mirrors just and tiny motors. It will get reflected precisely to go out and out the port. But I can program what output port it goes out. So, in other words, essentially what it gives me is a programmable topology. So that if I decide that a rack needs to be virtually removed, virtually removed.

This is all under software control. And then another rack gets plugged in in the exact same position that that other rack got removed, I now can maintain my topology. The torus becomes whole again. And I can do this in let's say seconds. So, essentially what the real differentiator has been for TPUs is the ability to have much higher levels of availability. I can now recover from failures instantaneously. Right? As long as I have a few spare racks, quote-unquote, lying lying around.

And the spare racks, by the way, could be doing smaller computations. They don't have to be doing the gigantic computation. That's place one. Place two that it becomes useful is let's say I told you about the compute problem and the storage problem. Right? We're doing agents. I now, one more level above that, have a different optical circuit switching layer where I can say, "Point the mirrors to that cluster over there where the storage that I need is located."

I now can short circuit many layers of a general purpose electrical packet switch that I would have to have normally provisioned and built to go to that distant cluster, and basically create a direct connect. So, really think of an op- So, do I I still have lots of electrical packet switches. But I now have many fewer than I would have needed where I can program which cluster I can talk to. This is You're right, it's not per packet. But if I know that I'm going to run this 5-hour job, and this 5-hour job needs the storage over there, point the mirrors over there. Mhm. The next 5-hour job needs the storage over there.

Okay, as part of Borg, scheduling the job, it would say, "Point the mirrors over there for the next 5 hours." I see. That that saves me from provisioning layer upon layer upon layer of network and miles and miles of fiber, essentially allowing me to not have infinite bandwidth wherever I want it. It's not fully fungible because you're right, if at a second granularity I said, "Oh, wait a second, I want to go over there." It's not that I can't, I still have electrical packet switches over there.

Just not with the full bandwidth. The full bandwidth is pointed over there for the next 5 hours. Or however long I decide I need to move back over here. It's a kind of a deep question, but so optical circuit switches, they have their role. They're not a magic bullet that solves all problems. We use a lot of electrical packet switches. Why is the torus the topology settled on versus others? Originally for uh ML training, the number one collective was uh all reduce rather than all tall.

And for an all reduce, actually, you the torus is the perfect um topology because you essentially are disseminating parameters to everyone with potentially a little bit of computation, a little tiny bit of computation on each distribution. So, the best and fastest way to do dissemination of data for this particular style is with an all reduce. Now, if you are doing an all to all, turns out the switch topologies have have their benefits as well. For um that regime, what is the optimal topology?

Optimal if if you truly need to do all reduce, I'm sorry, all to all uh with arbitrary communication, the switch topology, the standard factory clo- clo- topology would be the best, but it winds up that model designers can work around the topology in very clever ways. And they do. Yep. Uh next question. The question I'm not going to take your assumption your assumption was all chips are becoming obsolete, that is not true. However, your question was how does Google think about hardware depreciation, correct?

Okay, let's take that. Yeah, so all chips are um not becoming obsolete. There's so much demand that our um older generation chips continue to see very heavy use at Google, and this is true at uh whether it's older generation TPUs or GPUs, it's true across the industry. H100s are massive demand despite the fact that the Reuben has been announced, etc. Fantastic chips as well, H100s and H200s, and V200s, and GB200s, etc. as well. So, we depreciate our hardware, our compute hardware over 6 years at Google. I think that is more or less standard across the industries. I think some people, a few people might do five, but 6 years I believe is standard. We are seeing use at least for that period of time, and typically longer for for our hardware. So, it works works out well.

How do we plan? This is the problem that we were talking about earlier. It's it's very very hard to plan for the future because we're having to make these predictions fairly far in advance for one saving grace is when we're provisioning watts and data center space. That's fungible. In other words, it could be generation X, it could be generation X + 1, it could be generation X + 2, it could be generation X - 1. So, we first need to have an envelope for watts.

But the lead time for these chips are also significant. You got to get your orders in early, and you have to plan plan for those as well. I can tell you that we have a planning is a massive effort, massive and complicated effort, and fast changing. Because let's say that I have a plan, and then a new use case comes up. There's a new invention internally at Google, a new product launch, and it needs a particular kind of capacity. Now I have to figure out how to fit that in. I have to replan. So, essentially, by the way, another very interesting domain is how do you plan under uncertainty, and how do you dynamically replan quickly based on all the new information that you have, demands that you have, customers that come in. A new cluster cloud customer comes in and wants to buy a bunch of GPUs, but it's not the GPUs that I ordered. It's a different kind of GPU.

How do I order these new ones, get them, and by the way, they want they want to build close to their cluster in Minnesota. I'm making all this up. But so, like all these constraints come in, and now we have to replan dynamically, and essentially daily based on the new information that we get. Awesome. Next question. Yeah. How do you see robotics capabilities being unblocked? Yeah, I think really exciting domain and I think that this is you know, to me if I think about the internet revolution, it really was the coupling with the mobility revolution that made it truly the impact that it was, right? Basically taking the internet into the real world, making it mobile. I I I think I'm biased, so you you all can check this, but I think that the best example that we have of really advanced robotics out there in the world working in very complex scenarios is Waymo.

And so I think that's a good example of this scaling approaches. In robotics, I think in many cases you're going to find that latency really matters, but safety is the primary consideration. And I think you're going to have very similar scaling requirements, but safety, reliability will just shoot through the roof in terms of your considerations and that's going to then argue for locality and essentially whatever you want to call it single-threaded programming. I don't mean single-threaded as in okay, there's only one one core on the CPU or whatever on the TPU, but essentially you you can't have variability. Like if if there's a safety question, you can't say oh wait, I had a context switch of 10 milliseconds and I wasn't running when the safety whatever algorithm needed to be running.

So I do think that the similar scaling laws are going to apply, but the scale that you can count on for robotics is going to be much much less. If you're counting on 20,000 TPUs in a data center 1,000 miles away for for your robotics application to work, probably depending on the robotics application, may or may not work. The question is are there do you have any thoughts on the SpaceX Anthropic partnership that was announced today where they're going to you Anthropic is going to be able to use some compute from the former XCI Colossus cluster.

Similar announcement on cursor. So, cursor is going to be leveraging a bunch of capacity on SpaceX XCI. And I think what you're seeing here is massive demand for inference compute today. And so, really if you think about it, you'd have to say that coding agents really exploded. They've been around for quite some time. So, I I do know that, but they really exploded 4 5 months ago. And nobody nobody predicted it at this level. And so, nobody essentially had enough lead time to say I need more GPUs, more TPUs to handle this explosive demand for serving.

People are now looking around and saying what capacity can I get where? And I don't know the inside story of whatever Elon and Dario discussed or whoever. But, you know, clearly good opportunity for Anthropic to leverage a bunch of available capacity that SpaceX had less useful. What got me into this field? Uh, and what um convinced me to switch from being a professor to my job at Google. Uh, I was lucky in that for whatever reason, I was I remember I was 6 years old. I was in um uh, Iran at the time actually. My family moved to the US when I was 6. So, it was right before we moved. I saw a magazine cover.

And it had a computer on the magazine cover. And somehow I I decided I was going to become a computer programmer. Never seen or touched the computer, but I decided that. Um, I think my defining characteristic is

I'm very stubborn. I never change my mind. And fortunately, I loved it. So, when I was in high school, I was I was the kid I was that kid, right? And this was a a while ago. I was in the lab programming all the time. So, boring story. I I still love it uh, to to this day.

Uh, and then I loved it so much that I really um decided I had to get a PhD. I I needed to understand the material. It wasn't about um anything other than really love for the material. Becoming a professor was natural. I came to Google because I was I'd been a professor for 12 13 years and actually never had a real job. I had jobs in research labs, but that didn't count. So, I said, you know, if I'm teaching all these people, I better know something about what it's like to be in industry. So, I came to Google on a one-year sabbatical.

I loved being a professor and actually I was quite um haughty about um people working in industry. Meaning, I couldn't understand why anyone would want to work in industry. No No offense to uh anyone here because I was so biased. I admit I was biased. Um I got to Google very very fortunate. So, Google at the time I joined, 2010, there were seven people between me and the CEO. All seven of them, uh including Eric Schmidt, the CEO at the time, had a PhD in computer science.

So, here's this guy who knew nothing about um industry. Literally nothing. Uh any other place I would have gone, I think that there would have been like organ rejection or I would have been like, "Oh, I I was so right. Industry's terrible." Uh Google was a match to me. And uh took me a while, probably 3 years, to figure out that uh I was having so much fun that I didn't I wouldn't go back to being a professor.

But I I miss it actually and I love it. A fantastic job. Uh one of the best jobs ever. Uh but the opportunity to really put ideas into practice and Google is the kind of place where yes, it is about um business impact and it's about the outcomes, but it's also about um doing the right thing for people, our users, and doing the right thing about before technology. In other words, it's like solving hard technical problems.

Really valued at the company. Good question and I think there are a lot of good firms out there, honestly. So, I I think it really uh I'm very optimistic about the space and I think there are a number of uh strong firms. Really it was um uh evaluation of their technology, evaluation of their people, how far along we were with them relative to others. Uh it really I wouldn't read too much into it about um this one is the very best or this one is the second best, etc. Cerebras is fantastic. We're big believers, obviously. Uh but I think there's going to be a a number of winners in this area.

The question, what do you what do you see as next for TPUs to beat um GPUs? Are you saying that even a goal? Is that even a goal? Not even a goal. I I mean I do get this uh question fairly frequently. I think it's a good and reasonable question, but I think that uh good news is that the market is expanding so dramatically that there is no beating or there's no competing per se. In other words, there's no winning and uh losing. I think it's about driving impact. So, I mean we we buy and sell uh huge number of GPUs. We use a lot of the huge number of GPUs. GPUs are fantastic products.

And I think they're going to and I have, by the way, all the respect in in the uh world for uh Jensen. Would uh

uh would would call him for advice uh on on a number of things uh for sure. He's he's amazing. His company is amazing. But I would say that we're going after different uh domains and uh different uh customer use cases, etc. What I'll say broadly is for TPUs, uh we just uh a couple weeks ago announced our latest eighth generation TPUs, 8i. I stands for inference and 8T. T stands for training.

And so, for the first time we're launching two chips in one year. Why am I mentioning these two? It's because we're we for the first time are specializing the TPU line. In other words, previously we had one chip for both serving and training. And that was the right decision based on everything we could see because we could have probably we always could have built two chips, but if one chip is 5% better for one and the other chip is 5% better for the other, it's actually better to have the one fungible chip.

Right now, the needs are diverging so much that we're actually seeing big uplift, major uplift in specializing for inference and training. What I see coming um moving forward is uh further increase in specialization. Why? Because general-purpose CPUs, they've for many years, a decade plus, have slowed in their rate of performance efficiency improvement year over year. And so, what that means is that now you actually have to pick the workloads that um are large. And you can't necessarily say, "Hey, just wait a year and your CPUs will get twice as fast." because that won't be good enough to keep up with the demand. We have to pick our big workloads. Inference and training are two great examples where we can now say, "Hey, we can actually do something, let's say, twice as good." because we specialize. The lesson in hardware design is the more you specialize, the better performance you can get for the subset of workloads that you can run. CPUs, of course, by the way, CPUs aren't going away. Like they're they're general-purpose, they can do anything.

Uh a TPU can't do anything, but for the domains where it runs, it's literally 100x more efficient than, let's say, CPU. So, we're we're we're in the process of finding those use cases one by one and saying, "Okay, now and maybe it won't even be a TPU." Like maybe there's going to be some other big workload that doesn't require tensors, matrix algebra. Maybe. Or there'll be some other one that needs a different system balance point. By the way, that's the key observation between AI and AT. The memory-to-compute-to-networking ratios are different.

Right, so you you actually would design the chip differently because that's what that application needs. We're going to keep looking and specializing for the different domains. The The questions are on on unblocking your own production bottlenecks from from provide vendors and suppliers like TSMC. Yeah, we're we're deeply engaged across the the supply chain. And um and so, I I I'll say it's um the simple answer is it's a domain that we're comfortable with. uh You know, my team right now is in Taiwan and South Korea and Thailand etc.

As well as as we speak. So it it is a complex issue but I I'm actually not worried about being able to secure a supply. Our fair share of supply at Google. I think the challenge is again it comes down to the efficient use of that capacity. That's going to be as key as anything. Now the total demand in in the world is going to be significant but I think from a supply chain perspective if you I mean maybe I'll just give a generic answer.

If you are a vendor for a component. Let's say it's let's say it's a capacitor. Do you want to have one customer? I'll

leave it as a as a hypothetical. And let's say that customer was going to say I'm going to buy you out for 3 years [clears throat] all your capacitors whatever you got. I'll buy it all up. I would say that's not good for the vendor actually. Even if they might make more money in one or two or three years.

Because so the flip side of it is um as component vendors they want to have some diversity. You know again for whatever it is SEC filings who's your how many how many customers make up 90% of your revenue. If that answer is one or two investors aren't so super happy because now you're beholden to exactly one or two customers. I I think this is a sort of misunderstood point and I'm going to try to connect two different questions here just to help synthesize cuz we are lucky enough to have a professor who's better [laughter] than me.

But if you've noticed many times when you guys ask questions you place on context and there's an assumption in there about the industry and then you ask the question. And many times I've noticed over the course of the quarter you guys use these words like winner loser you know, there's a sort of embedded zero-sum mindset that I've picked up in this class and I don't know why that is. But it's a it's a it's a constraint of your own making. There's no such thing as winners and losers in the real world. They're just people who get done and who don't.

People who have impact and who don't. And so, I would encourage you guys to really uh think first principles about some of these assumptions. I mean, just here we've had somebody who's who his answer just demonstrated that, right? He said I think the question had some assumption like, "Oh, you know, Nvidia is locking up all the all the production at TSMC. What are you going to do about it? You're going to lose." He's like, "Well, actually, you know, turns out vendors don't want concentration risk. If you break down from a first principles how their business works, then you can see they actually want Google to have some percentage of their production demand."

And in infrastructure and mission-critical supply chains, you need to have redundancy built in cuz earthquakes happen, geopolitics happens. And if you want to be a reliable stable partner to your customers, you plan for that. So, generally I would just let's tone it down a little bit on the whole competition stuff because it it only holds you back. You know, having I don't know if you'd agree with this, but I'm fine with the questions, by the way, but I think the advice back is um is great in that really um I I view what we're doing at Google as a participating ecosystem to lift the entire industry, but also lift all the users. It's not going to happen on the back of any one company. There's no one company that's going to come out of this as the winner for for sure. There's going to be many winners. And by the way, the other thing that is true is um the uh huge number of the winners haven't even been invented yet.

I some number of you in this room are going to start some of the winners, no doubt, over the next uh several years. Uh there's going to be use cases and opportunities that none of us, certainly not me, can predict that that you all are going to invent. There's going to be a There's going to be a lot of winners. One one caution I want to say though is we are also going through and this is not about companies a time of societal transformation. So so if I if I may just I know this isn't on the topic of this conversation, but it's the top of mind for me. I I would also encourage this group who is thinking about technology to also think about our responsibility as technologists to make sure that we are building in guardrails and safety as we deploy our inventions in terms of how we help

drive the societal transformation. I mean I think 5 years from now, 10 years from now, how we work, how we live, how we learn is going to look a lot different. And we we do want it to also be better as as a whole, maybe hopefully significantly better.

And in the ecosystem as this transition is happening, it's stressful for a lot of people, there's fog of war, people don't know, you know, information is not being disseminated out. What are What are some areas of misalignment across the ecosystem that you would encourage not just them, but other speakers in this class who are who are watching each others lectures to think about? Oh, it's a good good great question. And by the way, congratulations to to you and Mike in terms of this class and and all these students and this this thoughtful question. I'm just blown away honestly by the It's all Mike behind the scenes.

the the quality of the discourse here and the fantastic questions here. Blind spots, um I think that frankly I I thought that your feedback to the room here is fantastic. Probably across the ecosystem there is a notion of uh single winner a bit a bit too much. Yeah. And and probably also a bit focus on individuals winning and losing. So that sort of pairwise fight, I won't name any names, you all know what the names are, but person X is out against person Y.

And I don't know how much value that's adding for to anybody. From your perspective, what do you think true bottlenecks are? Yeah, it goes back to the question of what you would study if you were coming out of Stanford. There is no one bottleneck. If I What is the primary bottleneck? Honestly, it shifts daily, weekly. Yes, I mean, I hear about memory getting locked up in the supply chain or some other issue that that might be coming up. And on a particular day, the bottleneck might be the reliability of a particular cluster that for training our next foundation models. That might be the bottleneck. I would say the one that I have least understanding [clears throat] of the solution is energy.

In other words, if I can roughly make up answers [clears throat] that I have some confidence in for for most topics, but for us to scale energy to the level that we need to across the planet, um there are ways to do it. There a lot of them are brute brute force and expensive and expensive not just in dollars. So, uh the biggest innovation bottleneck, I would say in terms of really getting what we need, energy abundance, which also means affordability, is Yeah, that's it's probably energy.

And in the energy space, which solutions do you think are being under explored or which vectors should be could be more systematically explored? >> I think that here in the in the US, we could look a lot more at wind, [clears throat] solar, batteries. We are We are at Google, for sure. But this is a manufacturing and scaling process that has some physics involved with it. And And physics meaning just some time.

So, this is an area where we're probably under invested as again as a as a community. There was 2 days ago there was a company that just announced some money they'd raised to build data centers um as a network of distributed floating uh bods. Mhm. Is Is that a promising vector? How would you analyze that solution? Yeah, and of course we and others are looking at the data centers in space. Um I think that there are a number of really worth in in space energy uh 5x more efficient. And if you get into a a sun-synchronous orbit, 24/7, no you

know no no or very little battery needed. I would say there are a number of promising directions like this. Um they're all fairly far out and all carry some risk. So, for me it would be a portfolio. Uh the proven technique elsewhere in the world is solar, wind, battery.

And pretty pretty affordable, pretty fast to manufacture, pretty fast to stamp out uh significant capacity deployments in short amounts of time. Well, when you say far out, you're talking roughly a decade or so, right, of 5 to 10 years, we can argue. 5 to 10 years. >> That's pretty short. It's pretty short, but we have a lot to do over the next uh A with some risk. And we have a lot to do over the next 5 or 10 years. Good question. At what point does the hardware stop being a a bottleneck?

Uh no point in the future that I can see does the hardware stop being a bottleneck. So, in other words, I would say that um right now massive model innovations, uh but also massive bottlenecks. So, we are in a place right now where um this is uh Rich Sutton. Uh won the Turing award a couple years ago. He he wrote this article. I encourage you all It's short. It's an essay uh called the bitter lesson. And it says 70 years of AI experience as throw more computer at the problem and you're going to get better results. And we're we're living that. Um I don't see again Well, I'll go with the 5 or 10-year view. I I don't see computes not being a bottleneck for the next 5 or 10 years. I I'd wait if Personally, I'd go longer. I'd go much longer, probably.

But um here's the way I'd look at it. If If we came up with a um massive algorithmic break breakthrough. And if you think of transformers, before transformers there was a previously dominant algorithm for learning LSTMs, long short-term memory. Transformers roughly 5x more efficient. Like same results, five times less compute. Amazing. If we had another transformers-like thing, transformers prime, 5x more efficient, I'm I'm pretty sure that we'd still be constrained on compute.

Like all that capacity would would get used usefully. Maybe not overnight, but quickly. The question is how are you thinking about infrastructure, equity of access, of and and the the impact on the environment? Yeah, I I love this question, appreciate this question. You know, our goal at Google, my goal at Google is that our data centers should be a uplift for the local community [cough] and an uplift for the grid. And so whether it's noise, water, and power, across the board the goal is that that these should all be viewed as positives. Of course, jobs and access to technology, but we should be in in my opinion must be coming with uplift to the community. Now, there are concerns, by the way. I don't I don't want to understate them, etc., across the country, across the world, but we really are working proactively. For ex- Let me give an example here.

Um PUE, power usage efficiency. Historically at Google, up until the last few years, we had two designs that we considered for how we build our data centers. One that was more power efficient by 10%. 10% is a lot. That says, you know, if you have a gigawatt, you're 10% more efficient, that's 100 megawatts that you now get to use that you otherwise wouldn't be able to use. Two designs, one that used more water and one that used essentially no water.

The one that uses no water, 10% less power efficient. As a whole, okay, maybe that makes sense. Maybe that makes sense from our bottom line to say, "Well, go use more more water, but you get 10% power efficiency."

But in a particular community, that could make zero sense. Right, that would be a net negative, a huge net negative for the community. So, what we've done, uh what I've done, is we've said, "You know what? Actually, unless there is abundant water in a particular community, where the community would say, 'Actually, we'd rather you use less power, we're going to go in with the less power efficient design, but the one that uses almost no water."

That needs to apply across the board. In other words, it this needs to be a asset. Another example here is we've recently um developed uh technologies to have a gigawatt of demand response. What this means across the country. What this means is I told you about how the grid over provisions. They over provision for the homes, the communities, for that one week of the year where weather's the coldest or the hottest, where they have to have the most power available for people's homes.

What we want to be able to do is we want to say, "Okay, we'll take power the one week of the year, the two days of the year that you need it. You tell us, and we'll give you back 100 megawatts. We'll power things down on our data centers." This goes back to the 99% reliability bit. Right, so we'll work with the utility where actually now they can provision less while guaranteeing that the houses, the homes in the community that need them, have the power that they need without having to have 2x the provisioning for the bad 2 days or the bad week of the year. We're happy to take that down time. We're happy to be again an asset to the grid, an asset to the community. We have to do more, by the way. I'm not at all suggesting that this is done, but we very much are taking this. I'm very much taking this super seriously. And what should this I would wrap on this, but what should other Well, that's Let's say that's like a what's your what you're doing there.

What should other cloud What what should other infrastructure folks who are scaling capacity in the ecosystem be doing more of that you think we're not doing enough What what I'm casting in the positive what I'm proud of is that when we say and this goes back to the even the first question so it's coming back our first discussion point that you raised. We're not trying to figure out how to build capacity at any cost. It's not hey we need a gigawatt we got to go spend 40 billion or 40 whatever the number is.

It's optimal scaling is the goal. >> It's optimal scaling and that is efficient delivery of that capacity for our users or customers but it's also how do we make sure that actually we're a great asset a community asset and welcome like that gigawatt is not just an abstract gigawatt in somebody's spreadsheet. It's a massive deployment in the state of Utah and it needs to be an asset for them and that check mark needs to be there. So I would encourage all hyperscalers all builders of capacity to be thinking of it end to end not just go get me a gigawatt but use it efficiently deliver it effectively have it be an asset for the community.

Thank you. We might need some of your professorial insights on how we do that. You know we're anyway Thank you so much. I mean