Evercore ISI Autonomous Driving, AI & Mobility Forum

Company Participants

- Ian Buck, Unknown

Other Participants

- Christopher James Muse, Senior MD, Head of Global Semiconductor Research & Senior Equity Research Analyst, Evercore ISI, Research Division
- Christopher Patrick McNally, MD, Evercore ISI, Research Division
- Unidentified Participant, Analyst, Unknown

Presentation

Christopher Patrick McNally {BIO 17580762 <GO>}

We will get started. So first, for those of you who don't know me, my name is Chris McNally. I cover the global automotive suppliers out of London. And really, on behalf of the entire global auto team, C.J. and the semi team. And really all at Evercore, we just want to say thank you and welcome to what we think is going to be a very special event.

So very quickly, this is the evolution of the last couple of years of this sort of small autonomous bus tours that we've been doing, C.J. and I, in Silicon Valley. Arndt helped organize a bunch in Southern Germany and then countless trips to Israel. So I'll keep it really brief.

We have an amazing lineup over the next two days of almost 20 companies that are really leaders in the autonomous AI and soon to be shared mobility space. And I should say, leaders and really soon to be leaders, because we have a broad spectrum of both new and startups.

So with that, I hope you have a great conference. And I will kick it off to C.J.

Christopher James Muse {BIO 18608702 <GO>}

Great. Thanks, Chris. And for those who don't know me, C.J. Muse, covering the semiconductor space as well as semi equipment. From our perspective, we have both private and public companies, servicing, primarily AI and autonomous. We've got NVIDIA, Intel, Xilinx on the public side and then a handful of emerging ASIC companies, which, I think, will be quite interesting.

To kick start, we have NVIDIA. Very honored to have Ian Buck, Vice President of Accelerated Computing, to kick things off. I think he'll present for roughly 10, 15 minutes. And then we'll do a little fireside chat. After that, we'll open it up to Q&A to the audience.

And I guess, with that , I'll -- we'll bring Ian up to the stage.

Ian Buck {BIO 18454865 <GO>}

Thank you. Well thank you for giving me this opportunity to talk. My name is Ian Buck. I'm the General Manager and Vice President of Accelerated Computing at NVIDIA. My background actually is -- as an engineer, I started at NVIDIA. I've been there for 15 years. Was hired into start a programming project called CUDA and built up the CUDA engineering. And now about five years ago, Jensen invited me to actually run the data center business. So my life -- my day job is working with both HPC and AI, whether it'd be supercomputing labs like Oak Ridge National Labs or others. And of course, all the AI players, the Googles, the Amazons, the Facebooks, the Alibabas, et cetera.

What I've done -- we have a few slides just to serve as a context for NVIDIA and maybe kick off for more discussion and then Q&A for taking questions. As usual, this is the safe harbor conversation and (inaudible).

All right. Can you (inaudible) to advance the slides because this thing is not working. Can you bring it back up, please, F5. (inaudible) it closed the presentation. It's in the middle of the desktop. Maybe the semiconductor guys could invest in Logitech to improve their products. I would appreciate it. I hate this thing. Here we go. Next slide, please.

NVIDIA as an AI computing company. Really, what we do is acceleration. We are an accelerator, which means we are a companion to that CPU. The first market that we accelerated, of course, what we are known for, are obviously computer games and graphics. We started that way with the founding mission of the company. But really it was the first workload that we accelerated.

Think about computer graphics. Computer graphics is video game where you typically have for every pixel on the screen of this video game of Batman, you might have a 1,000, 10,000, even 100,000 line program that's determining the color of that 1 pixel on Batman's face. It's taking into consideration all different light sources in this room, all of the surface properties of the skin, any environmental effects, lighting effects and lens effects that a graphic artist, game developer basically writes. They write that program in C -- and compile it to run in our GPU. Our GPUs then turn around and run that program on every pixel of the screen. That's about 1 million pixels, 10,000 line program and you have to do that in about a 60th of a second, throw all the results away. And do it over again because the camera moved.

That is the nature of what we do or have been doing for over 2 decades at this point in computer graphics. That same -- basically, over the evolution of still graphics, it's become a massively parallel computing problem, running those very long programs over massive sets of data, designing and then at incredible speeds and throughput, simulating life, if you will. The majority -- a vast majority of our processor is dedicated to specialized cores, we call them Tensor Cores, for accelerating that kind -- those kinds of computation.

And because it's a massively parallel problem, I've got 1,000,000,000 pixels on the screen, I can put down lots and lots of cores. I have over -- in the latest Volta GPUs, over 5,000 cores. And they're very different than CPUs. CPU cores tend to more optimized for running a single program really fast, one thread of execution to run your Outlook or web browser or the operating systems. In GPU lab, there's an accelerated computing lab. We typically want to run thousands or tens of thousands, really hundreds of thousands of threads running in parallel and we just throw as many cores as we can at the problem.

Our first market was computer graphics in gaming, which has evolved to be more and more programmable for stimulating light. We expanded to actually doing accelerated -- accelerating simulation in high-performance computing. The world's fastest supercomputer in the U.S., world's fastest supercomputer in Europe. And now being built, the world's fastest supercomputer in Japan are all accelerated with NVIDIA GPUs. We're doing that same kind of simulation. An example would be the air-conditioning in this room. We want to simulate the air flow in this room. We'll divide this room up into tiny little group belts and we'll simulate each group belt in parallel. And we use the group belt during -- over time doing various kinds of equations to simulate what the air flow in this room to identify where the vents should be, what the quality is. Same thing with the air flow -- air over a ring or combustion engine. Similar math kind of problem. And of course, today, AI was the -- we found -- we actually didn't find AI, AI found us and it's in the (third) of this slide that I'll be talking next. Next slide, please.

Let's go back, please. The first market that really discovered NVIDIA for general purpose of specialized computing other than graphics was actually the high-performance computing community. These are people building supercomputers that have to look out and plan 5, 10 years out. And they were the first to notice this trend of the end of Moore's Law. Basically, Moore's Law is a transistor law, giving you more and more transistors as process technology improves. They have more transistors to put toward my products.

It also turned into a marketing law. More transistors meant more performance. And for a decade, that's how the IT industry worked. We've got more transistors. Companies like Intel were fantastic, turning those transistors into faster technology, adding things like predictive execution, effective execution, branch predictors. It had enormous cache to keep the memory closer to the processors. And as a result, the IT industry could rely on getting about 2x performance in roughly over two years.

That, of course, started to tail off. Around 2010, even with more and more transistors, we started running out of ideas. We passed out. And now CPUs today are starting to add more and more cores to try to deliver more performance. So basically, they're trying to go more parallel.

In GPU end, we're very different. We started from actually parallel. We started from this different kind of use case. We never tried to run your Outlook -- or run the Outlook system, we only solve problems that are massively parallel. And as a result, we have an architecture that's designed for throughput, not necessarily latency.

And as a result, we are able to keep continuing taking these transistors that are available to us and directly applying to more cores. We also tend to be more holistic in the way we optimize. We don't release the instructions that drive processors. You can't download it. You can't write the assembly, unlike the (inaudible).

We actually only expose a high-level software interface, which we call CUDA. And everything is built on top of that and optimized below. That gives us a much wider palette for which to innovate and improve performance. We actually do change our instruction set every major architecture generation. We actually optimize or compile the whole system software, the programming language. And then, of course, we can go higher in the softwares like the linear outlook, mass libraries. And then all the way up into the AI frameworks like TensorFlow and PyTorch and others. That's basically giving us that 10x and 10 years kind of speedups that you often see talked about. Next slide.

So now we've transitioned with era of AI and it really feels like many of these -- the same technology transitions that we've experienced in the last 30, 40-plus years. But first, of course, the PC revolution with the advent of Wintel, the mobile revolution with the advent of Android platform, the cloud revolution with the likes of Amazon and containers. And today with AI, it's yet another new kind of computing technology that's now going to be pervasive.

I think it's important to talk about AI as a computing technology. It's not necessarily a market in and of itself. We see AI showing up in every one of the markets we touch. It's not just in cloud services like Okay Google. But we're seeing it affecting and influencing our high-performance simulation communities we're doing. People are using AI for better weather simulation, for better -- for processing all the data in their data sets. Anywhere you have enormous amounts of data that you can make predictions to improve a service, the process, the scientific discovery that's where AI is being deployed. So its capability is really as a data-driven computing technology that's effectively replacing traditional software, traditional software where I write and describe perhaps what a stop sign looks like. I'll describe what a stop sign looks like and see it has a hexagon, it's red, it's got letters S-T-O-P. I'll write that up in code. And with AI, I'll just basically give it lots of data, lots of pictures of stop signs and it can learn how to see -- what that -- what -- how to recognize stop sign from an image, basically write that software for you. That same story plays out in every one of our industries as long as you have the data to drive it. Next slide.

The history is -- well, I could go on for a long time about the history of AI to explain it to you -- to all. I think, AI found GPUs very early on with advent of CUDA. We've decided to make it available on all of our GPUs, not just our data center ones but our -- even down to our consumer-level GPUs. But anyone can go to Fry's or Best Buy who is a gamer and also download our CUDA tool kit for writing programs and see if they could benefit for their computing.

Back in 2012, a guy named Alex Krizhevsky was a graduate student in Canada. He was working on a thing called neural networks, deep neural networks. This is still in the AI winter. He was -- he happened to have GeForce GTX graphics card because he was also a gamer in his dorm room on his PC and he realized that the math that he was doing for these convolutional neural networks that he was studying with Geoffrey Hinton was the same kind of math we were talking about in HPC at the time. But he imported his AI code to the GPU, called it cuda-convnet. And that was the beginning of AlexNet. Alex Krizhevsky, it's named after himself. And he went -- submitted that network to a computer vision contest called ImageNet, which was the predominant competition for computer vision researchers. You are given a million images and you have to predict what's inside of each image. Computer vision researchers were competing. They were getting about 70% accurate. Alex came in as a nobody in the computer vision industry and went in -- won at 83%. And they were stuck at 72%, 73%. So within overnight, he became the #1 researcher in computer vision without ever studying or participating in the computer vision contest.

That was the catalyst moment. Geoff -- Alex, now with Geoff Henson, now works, of course, with Google. You have Alex Krizhevsky that we're seeing now with Facebook and the whole industry going from there. Basically, AI got because of the GPUs that Alex had trained on, he could take his training that took literally an entire semester to train these neural networks and brought it into about a month. And because that -- by making -- that's a terrible way to get a Ph. D.. But now that he could train in only a month, today, you can train AlexNet in a matter of, I think, 15 minutes or something like that. We can accelerate this whole industry to become practical.

You're seeing AI now hit every one of NVIDIA's markets and certainly anywhere where computing or AI can be used as a tool to improve prediction, improve speech for speech recognition, we're seeing for speech synthesis. One of the reasons why Okay Google or Alexa sounds so good is because they're using AI to generate the speech, to figure out the inflections and the right -- and how to say certain tenses and words.

You're seeing it for ad recommender systems. So given a web page, what the -- how likely am I to click on this link, which, of course, is difficult for advertising. We're seeing it for video inflection for space (cities). We typically can run about 40 HD streams on a single GPU, running a neural network in every frame to look for suspicious behaviors, to catalog vehicles, a lot of content filtering, detecting hate speech. These are things that are modern-use of AI today.

Then, there's all sorts of creative super resolution, which we now have magical enhanced button, where you can take a fuzzy image and make it crisp because we have neural networks that have been trained on images and can detect from incomplete data to make high resolution and high quality data. Truly amazing stuff. Next slide, please.

Fast forward to 2012 to today. NVIDIA, at its core, is a technology company. We try to make our products and our technology available on every channel. Today's modern GPUs -- so this is our Volta GPU. It kind of look -- it doesn't like your son's or daughter's gaming card anymore. This is what's going into data centers at Google and Amazon, supercomputers around the world, basically we build the fastest, largest possible things the SMC will allow and prioritized into a module that will run as fast as we can because the thirst for computing for AI is extreme.

To put that into perspective, you typically need around 10 million images to train a neural network to 90-plus accurate prediction rate, which is -- in each path so that modern neural network, like ResNet-50, is about a gigaflop each. So it's an enormous amount of computing necessary to train along with the data.

We make it available through GPUs like Volta through OEMs. We also now make our own bespoked systems to highlight AI systems to lead the market. We enable our OEM partners to build out their servers with the technology as well. We help -- we make the same technology available on every cloud. So you can have ramps of Volta or DGX at Amazon, Microsoft, Google and Alibaba.

We are innovating on the software side so we don't just stop at the hardware. We work at the numerical library level, the programming level. I have teams that are working with the TensorFlow team, the PyTorch team, Caffe2, (Palo Palo) in China and other frameworks to help with the training software and, of course, the important software as well for running these neural networks in production. Then, we'll even make it available at the consumer level, at the developer level, the next Alex guy. So he can have the fastest possible consumer-grade PC GPU to do his AI research to develop the next workflow to make that available. Next slide, please.

And let me go -- there's all sorts of markets here. We can talk about markets later. Keep going. I don't want to drag on too long. Next slide, please.

Just a couple of fun ones. Ohio State is doing neural networks to -- for health care. Health care is, I think, literally a $1 trillion market. Everyone needs health care. They're using AI to do AI-assisted radiology. A huge amount of time and energy and money is wasted on throwaway scans, on MRIs that didn't come out clean or MRIs that may have missed very early detections of cancer. So what these guys are doing is actually training neural networks. And have an enormous amount of medical imaging data available to us in the world. Everything is archived in electronic medical records. We can now take that data and train neural networks to tell a radiologist, "Hi. look here, this is a potential spot" just like (inaudible) therapy. It's not to replace doctors. I think that's not -- no one wants to rely entirely on a computer for the diagnosis. But to aid the doctors in identifying the spot really early.

Transportation. Kansas City, you won't think of Kansas City as the bastion of innovation for AI. But we are actually deploying neural networks to detect when potholes can form. They take all their traffic light data, all of the weather data, all of the -- all their street data and pump it through a neural network. And they can predict within -- I think, they're getting to 90% accurate that a pothole is going to form at that intersection. So they can actually send a truck ahead of time and target those cracks on the road before that pothole forms and they actually think they can eliminate potholes in Kansas City using AI. I wish they were doing that in San Jose where I live.

The last one is preventative maintenance. So GE is doing good work on actually taking all the data that they're getting from gas turbines and doing -- and predicting when, ahead of time, when those machines are going to start to fail so they can do preventative maintenance early rather than later where they have to take it off-line, potentially interrupt service. That'll cost a hell of a lot more. So they're roughly saving about $50 million a year per plant by doing preventative maintenance, which is AI-driven.

There is ton of these stories and I can go on. I'll spare you the details. But you can see how AI is a general purpose computing tool for taking all their data and making good prediction and applying to all sorts of opportunities. These AIs that we've developed they need data scientists and researchers to ingest the data, understand it and then build up these services. And they need a lot of compute to do it.

I think that's the last slide. Oh. And then I know you guys want to talk about cars later. I thought I'd frame that a little bit. Self-driving vehicle is an example of a vertical industry. NVIDIA doesn't invest in many vertical industries. This is one of them where we are investing. So we are actually building a self-driving car platform. We believe that from the bulk of computing, if you can leverage the cloud, that's a much more efficient way of doing, running AI. Your hockey puck in your kitchen, if you can connect to the cloud to do with AI, you get the horsepower of the cloud and keep the hockey puck a $20 device. That's -- I think the hockey puck that's always connected to the cloud. We can't rely on that for self-driving. So we have to bring the AI supercomputer into the car. We typically run around 12 different neural networks in our self-driving cars. This is one of them that actually does drive around. We can get in the car with our hands not touching but close to the wheel, leave our building, come around San Tomas, merge onto 101, drive around, just fine.

And to do that, we basically take our data center GPUs. The highest end, Level 5 system, actually has 2 of these GPUs for redundancy, 2 of our Tegra SoCs and is running all those neural networks in parallel to do task prediction, lane detection, collision avoidance. Actually multiple cameras looking at the driver or looking if our driver is distracted. Also, it's different use cases. This is an area where it's an incredibly complicated problem, lots of challenges. Getting to a production-worthy Level 5 car is very challenging. A lot of it is having simulation as well. So we're taking our same GPUs that we use in the data center, same graphic capabilities and simulating all these things. We might be able to drive 1 million miles on the road to test drive a real car. But we can do billions of miles in simulation. So a big part of our data center business actually is trying to show up in the automotive companies, buying racks of our data center GPUs and servers for doing simulation of self-driving experiences to teach the AI in simulation before it ever hits the road.

I think that's my last slide. I want to thank you and we're happy to start the conversation and take some questions.

Questions And Answers

Q - Christopher James Muse {BIO 18608702 <GO>}

So I've got a whole laundry list of questions. But I think we'll have 20 minutes of time. Just to start off at the high level, Dave Patterson from Google has talked about a renaissance for computer architecture as the end of GPU, TPU and he is positing the hypotheses that we're just about to inflect in AI. So you've built a business that's almost at a $3 billion run rate, what's your view on kind of where we are in the AI cycle?

A - Ian Buck {BIO 18454865 <GO>}

Yes. Certainly, I agree with Dave in the sense that it is accelerating. AI as a tool is becoming a new way to write software, a new way of doing computation and, obviously, the market is affecting our huge health care, cars, economies. The kind of architectures you need in order to accelerate AI are different than what we traditionally have in CPUs. In fact, our latest Volta GPU is a redesigned architectural (inaudible). And we call it Tensor Core, which is designed for -- specifically designed for doing tensor operation, which is, if you -- it's how -- it's the basic building block of frameworks like TensorFlow, why it's called tensor. And PyTorch It is challenging. The other part that's happening in AI is diversification. There's not just one neural network to rule them all. We started with convolutional neural networks for doing image processing. We now have RNNs, we have LSTMs, we have GANs. There's TQMs. There's all sorts of -- there's parsed LSTMs. Each one of these neural networks is very different in their structure, in their sort of mathematical operations and what they need to do. So -- but then that's just what's been invented in the last two years. I mean, as more and more people swarm the field there, it's only going to explode. We track over 200 different neural networks for performance tuning alone and it's expanding.

So I think, the reason why I mentioned that is because programming is really important. Having a place and platform that people can program and develop these neural networks and evolve them because they tend to usurp and replace each other in real-time is critically important. And as we -- as AI field is evolving, we're cranking at new architectures, well, almost annually now. And because of how fast this industry is moving and what people want to do. I (inaudible). We can innovate at so many different levels and which is why we go from top to bottom, at the same time, you have to be at the forefront of it because it's evolving so quickly.

Q - Christopher James Muse {BIO 18608702 <GO>}

I think one of the major areas of focus is competitive positioning. And so can you kind of, I guess, start off with CUDA and talk about the flexibility, programmability, nature. And why you think that gives you a competitive advantage versus sort of emerging ASICs that are now evolving as well CPUs and other?

A - Ian Buck {BIO 18454865 <GO>}

Yes. So when we started CUDA, 2004 and launched it in '06, we wanted to give developers something to program. GPU was around. That term has been used. But you had to basically have a Ph. D. in computer graphics in order to use the GPU. So we invented this new programming model, which is just based on C and later Fortran to allow anyone with C and C++ background to program the GPU very easily. This functions around the GPU and when I called this function, instead of running it once, like you would do in a CPU, you just say how many times you should run in parallel. Really simple. So if anyone understood C or C++ and now, of course, Python and Java and Fortran could understand programming GPU. So -- and then the second thing we did is we made it available everywhere. Every GPU that we've shipped in 2006 supports CUDA, the most massively pervasive available parallel computing architecture processor available in the world and you can get it at any country. So engaging that developer base and allowing things like Alex Krizhevsky, the guy who invented AlexNet, just keep us aloft to find that opportunity. It was a difficult decision to make at that time. But clearly has paid off for us in the long term. And AI is evolving how they develop and program and move these frameworks forward and not just the AI frameworks. But also the inferencing softwares, the production softwares, take advantage of the technology, it's paramount. People don't want to learn new programming languages. They want to use the languages they already know and innovate on top of them.

Q - Christopher James Muse {BIO 18608702 <GO>}

So if the world migrates to 3 or 4 AI frameworks. And I'm not sure that is the case. But if we went into that world, would that obviate the need to program on CUDA? And I guess, can you share your view of how you think AI frameworks will evolve over time? Will they shrink and migrate to a handful? Or do you think they'll proliferate?

A - Ian Buck {BIO 18454865 <GO>}

They are -- they will proliferate. So at the beginning, I think, there was a bunch of AI frameworks, by and large, trying to do the same thing. And you saw that. Then, of course, companies bought them and merged them and there's some consolidation because they all overlapped in capability. But we're starting to see a little bit more diversification of frameworks starting to happen. (Colby) you may not have heard of Colby . But it's the #1 framework use for speech and it's specializing in speech processing and speech everything. There's -- in health care, there's one called CANDLE. And they're specializing in genomics and understanding cancer. Their whole purpose is how can AI be used for cancer, understanding AGCT and correlates AGCT DNA sequences with a doctor's electronic medical records or database. It's a specialized kind of processing that, I think, needs a different kind of framework than just generic TensorFlow. There will always be TensorFlow and there will always be PyTorch and those 2 companies will keep those things afloat. But we're seeing AI be for spaces and all those workflows and eventually into the software itself. So I think, while the vast majority of AI today is probably done on TensorFlow or PyTorch and legacy Caffe, you're starting to see this emergence of the rest of the AI software ecosystem developing new software. And there's just, again, how do you be productive on the platform, have access to the technology, access to GPUs, whether they be in the cloud or on their desk or in your data center and how fast you can develop and innovate suitable formats. Neural networks always seem to take about six days to train. No matter how smart and how fast we keep measuring those GPUs, someone out there is coming up with the next neural network that's going to get smarter and be bigger in the cloud more computing. So it seems to be a 6-day -- I don't know it's a week thing, it must be. But anyway, that iteration, performance matters plus productivity matters, program really matters and we're going to start seeing diversification in the frameworks to compound it all.

Q - Christopher James Muse {BIO 18608702 <GO>}

So how do you see the competitive landscape evolving within training? If you were to look out over five years, a, is this a rising tide? And b, how do you think of including FPGA ASIC along with GPUs in that world?

A - Ian Buck {BIO 18454865 <GO>}

Yes. I think there's bifurcation happening. We have a training world and an inferencing world. Training being building, designing and developing neural networks and inferencing which is the production and deployment of AI. In the training space, we want the highest performing systems, resources available because our data scientists if you hire one today, you're probably spending $0.5 million to $1 million in salary recruiting that person. Your job is to keep that person productive and their neural network is getting bigger. And at scale, it's getting bigger. So you want the highest performance stuff. Then on inferencing, you typically -- you want to start -- we believe and we're seeing AI introduced in all of the hyperscale businesses for processing every data element that come into the cloud, wants to be run through AI to be inferred. There's a lot of -- it's difficult to understand. There's a lot of benchmarking going and other such things. GPUs offer the fastest AI training, at scale, in the node or on a single chip. They also offer the lowest latency and highest throughput for inferencing as well. On ResNet-50, we can infer at 1.1 milliseconds for a full ResNet-50 model batch one. And we can also do upwards of 6,000 to 7,000 inferences per second throughput at that rate on a large (dash) scenario. On the training side, it's becoming about how we can gang the GPUs together to build super GPUs. We're seeing technology called NVSwitch so we can gang multiples up to 16 GPUs together to work as one. We've had -- we've invested in high-speed interconnects called NV Link to link those GPUs together and high-speed switches called NVSwitch to gang this together so that really can operate as one to do a single TensorFlow training, operating upwards of 15,000 images per second in a training group, which is obviously the target. Now where is the rest of the world and where is competition going to fall? I think, I guess, there's always room for specialization. One area we're not investing is at the IoT Edge, the doorbells, the thermostats, the shoe laces, whatever. In that scenario, I kind of believe that you're either going rely on the cloud for truly intelligent AI services or you can specialize for very specific use cases. We don't need -- I mean, to build an AI chip that can recognize the word Alexa. You can take -- and to get its cost down to that whole hockey puck is $20. Now it's worth the R&D and effort in time and specialization to reduce the cost of that AI chip, that Alexa AI chip down to next to nothing, especially if you're going to sell millions and millions of them. So I think that's an area where I think people can do all sorts of things. We've actually -- and it's also not particularly engineering hard to do so. We've open-sourced some of our technology to help that market. By the way, the sooner they get more AI, the sooner we get AI in the cloud, more training so we want that to succeed. So we've open-sourced some of our technology. One is called the DLA, which is for the neural networks that just need matrix multipliers. We've open-sourced that hardware and made that available to the IT community to go play in AI and make their own products.

Q - Christopher James Muse {BIO 18608702 <GO>}

So I know you can't share too much about your future technology road map. But I guess, one of the core focuses of the company is a single architecture. As you take silicon and you come up with new offerings roughly every year, at what point do you start maybe morphing into an AI-specific piece of silicon and/or perhaps more custom for specific workloads and customers?

A - Ian Buck {BIO 18454865 <GO>}

I mean, I think, AI is starting to affect all of the markets. You're already seeing shortened graphics. People are doing cool stuff with AI-driven rendering or ray tracing. We have actually demo-ed this where you can actually fill in the holes of array tracing engine with -- by using AI to predict what the missing pixel's color should be based on trained on real-life images. So I think -- our -- we'll continue to build one GPU that is best at computer graphics, best at simulation and best at AI, because many of those things are -- in the long term and in the road map those things are all getting intertwined. Graphics is becoming more and more about simulating light. AI is becoming a general computing tool that could be used in computer graphics and rendering, not just in looking cool. So -- and the other benefit is that we can take the entire horsepower of -- and all the money generated by all those markets and funnel that entire investment back into that one architecture again. That's great in all those things. And of course, the benefits of being leaders in the industry we can help guide the industry to make those transitions happen faster and keeping them aligned to obviously what we know we can build and take the next step.

Q - Christopher James Muse {BIO 18608702 <GO>}

Any questions in the audience?

Q - Unidentified Participant

(inaudible) Can you guys comment about (inaudible)

A - Ian Buck {BIO 18454865 <GO>}

Well data flow is interesting. When I was first getting my Ph. D., the first version of parallel programming that we're thinking about was the data flow architectures because you can keep stuff on check. I think it hasn't -- one of the challenges of data flow is the programming model, being constrained to, let's say, I do this and then -- the pipeline does this, the pipeline does this, the pipeline does this, restricts the program model to be fairly constrained. And there's always a question of why can't you fuse most of the layers together. Well we end up -- the other problem, I think, is that neural networks are getting bigger and faster. There are roughly -- I think, it's 100,000x in five years in terms of the networking size that go all the way out to the translation models. Some of those translation models have to understand the entire English language used on the web and the entire Mandarin used on the web and try to translate those too. These models are massively huge. So trying to express them in a data flow architecture is actually really hard. So we haven't seen that -- or haven't seen the program model that makes it -- that pivot easy. Instead, people just want to express their computation with a level of (inaudible) like architecture and let the memory system solve the problem for them in terms of providing large amounts of L2, L3 caches or having enough HPM memory to not worry about it. Today, training neural networks is compute-bound with HPM. We've got headroom. And overall, it's going to take the best possible memory and interconnect technologies to help us scale up.

Q - Unidentified Participant

(inaudible)

A - Ian Buck {BIO 18454865 <GO>}

Yes, great question. So your question is people are programming Python, right. But they're just expressing their neural network in Python and letting the framework just do its job and doesn't that mean we don't need compilers anymore? It's actually the exact opposite that's what's happening. So what happens is you have a data scientist, right. They're not a computer scientist. They don't know what a cache is. They would fail a job like in software in NVIDIA. But they understand the neural network, they understand the workflow, they understand -- they're designing that neural network. They don't really bother with the performance stuff. So what's happening is everyone is designing their neural network compilers inside the frameworks. And so they basically -- they express the problem as they understand it. They see the layers, they see the sizes, they see the shapes. Here's my batching, here's my rulers, here's my batch form, here's my activation functions. Hard part is then you take and then you compile that neural network to fit optimally on a machine. You actually do the blocking, the register blocking to pivot the calculation, looks like this, looks like that. It's back to that vector versus SIMD kind of transformation that are happening. But now they are happening at domain level like way up here. So we have a huge software team that basically works on compilers, take what neural network is described and recompile it and we've turned the output around to optimally execute on that architecture. And because the neural networks are getting more and more diverse, you need to actually think of it as a compiler problem, not just a pattern matching one. So it's actually not just a headache in silicon but also a headache for our compiler engineers to build -- we have something called TensorRT and what makes TensorRT is actually (outliers) for inferencing. We will recompile it down to the GPU. Not only does it do those corner turns and reblock the computation to run more efficiently to keep in memory, it will also reduce precision. So instead of expressing the neural network in 32-bit floating point, it'll automatically compile it down to 16-bit floating point or even 8-bit and measure our operations, which can operate 4x faster and be 4x smaller. But -- so the compiler guys are thrilled. They now have this other thing they can play with. Those are beginning inputs at super high-level, which gives them all sorts of room to play. And they get to do this thing called precision reduction. So they often sit with the data scientist -- they then call the data scientist to try to figure out if this is actually going to work. So it's a headache for compilers to -- they're getting harder to hire.

Q - Christopher James Muse {BIO 18608702 <GO>}

Unfortunately, I think, we've run out of time but, Ian, thank you so much.

A - Ian Buck {BIO 18454865 <GO>}

Thank you.

Q - Christopher James Muse {BIO 18608702 <GO>}

We appreciate it.

This transcript may not be 100 percent accurate and may contain misspellings and other inaccuracies. This transcript is provided "as is", without express or implied warranties of any kind. Bloomberg retains all rights to this transcript and provides it solely for your personal, non-commercial use. Bloomberg, its suppliers and third-party agents shall have no liability for errors in this transcript or for lost profits, losses, or direct, indirect, incidental, consequential, special or punitive damages in connection with the furnishing, performance or use of such transcript. Neither the information nor any opinion expressed in this transcript constitutes a solicitation of the purchase or sale of securities or commodities. Any opinion expressed in the transcript does not necessarily reflect the views of Bloomberg LP. *© COPYRIGHT 2024, BLOOMBERG LP. All rights reserved. Any reproduction, redistribution or retransmission is expressly prohibited.*