BofA Securities Global A.I. Conference

Company Participants

- Ian Buck, General Manager and Vice President of Accelerated Computing

Other Participants

- Vivek Arya, Analyst, Bank of America

Presentation

Operator

Ladies and gentlemen, the program is about to begin. Reminder that you can submit questions at any time via the Ask Questions tab on the webcast page.

At this time, it is my pleasure to turn the program over to your host, Vivek Arya.

Questions And Answers

Q - Vivek Arya {BIO 6781604 <GO>}

Thank you so much, and good day, everyone. Glad you could join us in this afternoon keynote session. Really delighted and honored to have Ian Buck, General Manager and Vice President of NVIDIA's Accelerated Computing business, also importantly, the inventor of CUDA, which is the key operating system underlying every NVIDIA accelerator. So really glad to have some time with him so he can share his perspectives.

So, Ian, I'll turn it over to you. I think you have one opening remark, but what I would really love to do is get your perspective on how have requirements for AI hardware changed throughout your tenure at NVIDIA. And especially, we always talk about hardware, but sometimes we forget that very key part of that is the software ecosystem. So if you could give us a perspective of how NVIDIA's software capability has really helped to cement your dominance on the hardware side in AI?

A - Ian Buck {BIO 18454865 <GO>}

Yeah. Thank you, and pleasure being with you here this morning. And of course, as a reminder, this presentation contains forward-looking statements, and investors are always advised to read our reports filed with the SEC for information related to risks and uncertainties facing our business.

Yes, we've been working on accelerated computing for quite some time. In fact, the dates all way back to 2006 when we first introduced CUDA. Initially, the goal was to address how to program this new kind of architecture, this new kind of processor that had reached a level of programmability beyond just playing video games and making beautiful pictures, but become a computing platform, a place where we can accelerate, not every workload and to this day, we always want to make sure we have the best CPUs matched with our GPUs in the right configurations, in the right ratios, but for portions of the competition that can be accelerated.

There are typically either highly data parallel or massively parallel or just compute-intensive. We work to -- with the community to figure out how to accelerate those workloads to run them on architecture that's designed for high compute and throughput needs.

It's started of course with high-performance computing, a community that obviously is looking for using computers and in some cases -- cases supercomputers, and just simulate nature to simulate physics and simulate a problem that can't necessarily be easily identified either in a wet lab or under a microscope or having a scale on -- the size and scale of the earth or the cosmos where we just need a computer can be a digital instrument -- an instrument of science.

We've -- for the -- all through 2006 up until that first AI moment in 2012, we've made our platform available as a software platform. We made CUDA available with every one of our GPUs including the gaming and graphics GPUs, the ones everyone had in their workstations, in their laptops and their PCs. And that was of course, before a lot of the cloud traction.

By building a software platform that engage developers rather than strictly a hardware platform which defines an I-S-A, an ISA. We met the developers where they were. So it made it very easy for researchers, Ph.D. students, engineers and companies to take care of NVIDIA GPU, download CUDA for free, all the libraries and the software that have been developed over time, and figure out how to apply it to their problem, to port their code whether BC, Fortran, today, Python, Java, others, and move that compute rich portion over.

That decision upfront to make it a software platform in combination with a hardware platform was really important for a couple of reasons. First, it met the developers where they were and we didn't have to wait for others to build a software exists from the ramp. Frankly, it would have been difficult to do so and taken a long time given the boot strapping problem. Second, it expanded the innovation space. We can innovate at the hardware layer, at the compiler layer, the system software layer, the library layer, and of course, everyone else had their opportunity also contribute to it. So the performance delivered over time is the compounding of all of those innovations at the hardware side and system driver and developer software, and of course, all the libraries on top. And if you track that progress over time, it's quite dramatic. That's the benefit of accelerated computing. It allows for compounding value to be delivered.

It also allows NVIDIA to innovate in an extreme click. We are not constrained by being by the interface at these lower levels like instruction sets. We're only constrained by the problems we think we can address up here. And if it requires us to change our architecture, to change our instruction set, to build an entirely totally different kind of GPU or build a GPU that can talk to other GPUs so we're going to be linked and scale across GPUs and system or GPSs in a rack or across the entire data center because we define the interface up here and how we engage, we can do all that and do at an extremely rapid click which allows our engineers to produce new GPU architectures roughly over two years now and some cases sooner. It allows us to think differently about how CPUs and GPUs can be connected, and also allows us to expand to the entire data center being our canvas for innovation for making change for influencing.

So that first decision I think to basically think about it from a different engagement point up here has allowed us to really innovate, move quickly, and we invite everyone else to participate in that ecosystem. And we've been doing it I guess now for along approaching about 20 years in NVIDIA. So there you go.

Q - Vivek Arya {BIO 6781604 <GO>}

All right. What part of that software stack, Ian, is substitutable? So for example, in the early days, it made a lot of sense, right, to couple the two, but now you have so many other people who are also involved in the ecosystem, whether it's the hyper-scalers or whether it's the R&D -- software R&D teams of many of your hardware competitors. So what part of your software ecosystem is substitutable? Can I take an application written four NVIDIA and find a way to port it over somebody else's hardware as an example using a combination of these third-party tools and other open-source software?

A - Ian Buck {BIO 18454865 <GO>}

Yeah. A great question and I get asked this a lot. Certainly, it is possible to think one workload or one AI model or one specific algorithm and get it working on anyone's hardware and platform. What makes it hard is to make it a platform for continuous optimization and evolution, and be a platform that can run all the workloads that we run inside the data center.

So today, if you look at our software stack, we have, of course, multiple hardware platforms ranging from PCIe cards that are being run at 70 watts, fit in any server and L4, to larger 300 watt PCIe cards up to HGX-based boards, which have multiple GPUs talking over every link, and we even shared how we can scale to entire rack scale or even row scale GPUs effectively.

So the -- then on top of that, you have of course the system software, all the compilers and libraries that then get integrated of the opening ecosystem, and these include our -- the hyperscalers, software like PyTorch, software like Pax ML And the wonderful part about AI is it's so open that we can all innovate together in that ecosystem.

So it's certainly possible to spike different implementations of different models into those stacks. What makes it hard is those platforms that I've mentioned that isn't in the community, you need to run on all of the different workloads to operate today across the entire data center. You don't build a data center around one model. You're going to run a data center to run -- to do large language models, do all of generative AI, as well as other data science or other use cases we need to do. You also want to accelerate end to end, and you often also see someone inspect a particular layer, a particular model. But to deploy an AI service, we have to do all of the ingestion, data prep, run the query, run model, as well as produce the output and in some cases, perform multiple other stages of AI like I wanted to have it talk back to me and so just or apply in the text, and that's also now being done in AI.

The other part of that is it has to be a platform for innovation because large language models and generative AI is not standing still. A few years ago, I still be talking to you about resonance or I'm talking to you about a coalition around networks. I'll be talking to you about some of the units and other recommender things. These things are still important but with so many people innovating inside of OEMs and from generative AI, what models are being innovated as equipped that's way faster than we're actually producing new architectures.

So in order to be a platform for that, you have to firstly investing in the data center scale, which of course is a huge capital investment and takes a lot of time. You need to be a platform where you can trust the innovations that are happening in generative AI, you're going to be able to run it really well. And again that comes to the end-to-end performance and optimizations that we're trying to make.

Certainly, you can pay for a model to get the -- to run all the models and the innovation platforms as a much more challenging to ask, and one that requires a connection and a benchmarking and all of those customers that are giving you that input in order to continue make your platform improving over time. And we find optimizations from everywhere. One of the benefits and front parts of working at NVIDIA is we get to work with all the different AI companies. So we get to optimize those layers of the stack that matter.

There isn't just one part of the stack that needs to be -- that is to be simply replaced in order to port. You really have to get to the end-to-end workload. And again, it is possible, but it's challenging to be sustainable.

Q - Vivek Arya {BIO 6781604 <GO>}

Now, let's talk about generative AI. Obviously, it has caught everyone by surprise in a good way, right, and demand seems to be exploding. So we'll talk first about training and then generative AI inference.

So on the training side, it seems like every day, somebody is launching yet another large language model, and NVIDIA dominates the market for training a lot of those models. Do you see a point at which we get to some kind of cliff or maturation or demand for training? And do you think as people start to then look at optimizing the size of these models that, that actually somehow puts pressure on the demand for training hardware? How sustainable is the demand for AI training when we are already producing so many large language models?

A - Ian Buck {BIO 18454865 <GO>}

Yeah. Large language models are different. We made them so and we'll make -- why are they so large is one question you could ask. And large language models unlike computer vision models in the past or simple -- the more simpler recommender models. Large language models are effective because they're directly acting with humans typically, and in order to directly interact with humans, they need to understand human knowledge.

So one of the reasons why GPT is so large is it's trained on -- it's trying to represent and interact with those -- with the corpus of human understanding compare so that -- they take the -- they download the Internet a few and they teach it what humans know so we can have a -- started a baseline foundational model that captures human understanding and knowledge, which obviously, is much larger than perhaps what you would need for a computer vision model, which is still very important, but you can be trained on a set of images and eventually, it can be known that those sets of images what they are.

So they tend to be very large models and they also tend to be a great foundation loss for specialization. So you can specialize for different workloads and you can specialize it when starting from that foundation model toward perhaps your data. So you're starting from something that understands humans or understands how to interact with humans or machines, best to use and then take it to your proprietary data and then be able to interact with or to ask questions at that data, and of course, leverage the general capability.

So when you ask about the capacity and how this is going to grow over time. This is that it is how you interact with computers, with the cloud, with your data, and that's hugely immensely valuable. It's immensely valuable for improving how customers want to interact with companies, how people who are helping customers want to understand and have an assistant sitting right with them to be able to ask questions and get the prompted information from a knowledge base and other things to have a better experience.

It allows -- large language allows -- enable recommenders, people who want to give content -- provide the right content is your -- on your newsfeed or in your e-commerce to have to get the right words and the right context being shared with you. So it literally touches every part of e-commerce of company interactions with customers and as sort of the answer to understanding the decades of big data we've been living in.

Does this tail off? I think it becomes a continuous space for innovation just across the board. And there is no -- there's not going to be one model to rule them all. It will be a -- there'll be a large diversity of different models based upon the innovation that are going to continue down the space and also specialization across all of these fields. And by the way, we're seeing in healthcare and science and drug discovery, large language models doesn't have to just be the language of humans. It could be the language of biology or physics or material science as well.

The -- so what is the growth vector and what does it look like? It becomes at how many -- it's the rate of which -- how many innovators are adding and defining and betting new optimization techniques, new kinds of models. And they start from some of these heroic amazing models coming from people like OpenAI, with the GPT models and what we're seeing there. But much of this research is being published or the models are being published that influence and create alternatives or derivatives.

So that is the scale we should be thinking about for generative AI and large language models. And the scope isn't necessarily the size of the model per se. They're going to remain large. And since that they have to remain large in order to be -- or have a baseline level of foundational intelligence. It is really -- the scale will grow as more and more industries and more and more companies and the rest of the enterprise beyond -- adopts this technique for how they interact with customers' data and apply it to their businesses.

And certainly, the hyperscalers, we're the first to jump on it. They obviously have the talent and capital and the ability to basically invent much of this technology side by side with NVIDIA to experience that. It was a fascinating experience. They continue to do so and continue to push the limits and figure out how to apply it. And we can see them starting to scale AI across their businesses and now it's branching out to the rest of the enterprise -- the rest of the industry. And you're seeing a whole tier of both more cloud offerings. We're seeing specialty regional GPU, data centers being popping up everywhere to serve the market. That has operated differently, a little more agile, perhaps a bit smaller but can be more focused, and then a large litany of middleware and solutions and software companies that are trying to help enterprises and other companies deploy this technology across the board.

So there's definitely a broadening of the large language model ecosystem, the adaptation of generative AI and language models to business is really the scaling factor that we experience and that will continue for sure.

Q - Vivek Arya {BIO 6781604 <GO>}

Now kind of a similar question but now applied to the generative AI inference side. What is NVIDIA's strategy for generative AI inference because the perception is that on the training side, the Company dominates, but most of the products are very expensive? So when it comes to really scaling generative AI inference which is really I think the way your customers will monetize that right at the end of the day, how are you going to help them monetize that? What's your product pipeline look like to help them with gen AI inference? And does the competitive landscape change as you move from training to inference?

A - Ian Buck {BIO 18454865 <GO>}

So this -- thank you for that question. And I think people often get a little bit confused, perhaps. Certainly, your starting point for some of these models for deploying them begins with their training and clusters. And so they all stand up an infrastructure and previously, A100's HGX systems. These systems are designed for eight GPUs and getting connected via the maximum possible performance, and of course, have InfiniBand at scale across entire data center. Today, it's being deployed right now with Hopper (Technical Difficulty)

What you train on is the natural platform for what to do inference on. Since training and inference are highly related, the model -- in order to train the model, you have to first infer and then calculate the error and then apply the error back to the model to make it smarter. The first step of training is inference with every event repeatedly.

So it is natural that customers are deploying their inference models with their training clusters with the HGX. It's not the only place where we probably see -- the only place we see inference. We see inference happening across the spectrum from all the way down to the L4 GPU, which is -- I should have brought one, it's a 72-watt GPU. It's half by half length. It's about a candy bar size. As small as my phone and fits in any server. Any server that has a PCIe slot can now become an accelerated server. And we -- in fact, we have seen the clouds adopted and the OEM and the rest of the system instruction probably because it's great for inferencing.

It's -- it has the video encode and decode capabilities. So we're seeing it used for smart city applications and image processing. It can also run small LOMs for recommenders or small tasks, and we also see it for generative AI for generation from running stable to fusion-like models, and it provides at a price point that's very comparable to CPUs. So, in fact, it makes it a better -- a much better TCO than -- the CPU run the same model. Maybe I'll finally return on that.

If you need to go up a click, you have the L40, which is a full-sized PCIe card and it runs -- which is often used for larger inferencing and fine-tuning tasks. So you can take an existing foundational model and then to fine-tune it to do that for last mile specialization for your data workload is a much lighter task than the larger training cluster and to could be done on an L40 or an L40S PCIe-based server, again available in -- across -- with every OEM system.

So this provides different price points and different capabilities and even you all the way click up to an NVLink-connected system. And for NVLink connected systems, we often see people running on a single node, and there, you just need to hit, you have model of certain size that just needs to execute in certain now latency say to be interactive half a second of latency response for Q&A, for example. So by connecting them with NVLink, we can basically build eight GPUs in terms of one GPU, and actually just run the model that much faster prior that real-time latency.

So our inference platform consists of many, many choices to optimize for TCO for workload and for deliver performance. In case of inference, usually it's about data center throughput at a certain latency, and that's important. The other part of it -- the roadmap is software. So I want to go back to that because it's easy to look at a benchmark result and see a bar chart and assume the speed of the hardware, but it's often under-reported numbers when I say investments NVIDIA makes in the software stack for inference. It's actually even -- you can apply even more optimizations that you can do within -- than just in training because in inference you are kind of at the last mile. So you can do further optimizations of the model beyond what is perhaps capable in training to optimize further.

For Hopper, for example, we're using -- we've just released actually last week, a new piece of software called TensorRT-LLM. TensorRT is our optimizing compiler for inference and LLM version. The optimizations we made in that software just in the last month doubled Hopper's performance on inference and that's came through a whole bunch of optimizations in both optimizing for the Tensor core that's in H100 using 8-bit floating point and improving the scheduling and execution software of managing the GPUs resources to increase its effective throughput and competition efficiency.

It's really hard to ask. You're trying to basically optimize by using reduced precision, by using -- by serving all different sides requests from quick Q&A to summarization tasks to write me along email or generate a full PowerPoint, a data center that's going to be running Hopper, or a data center running inference -- generative AI, it's going to be asked to do all those things.

Getting that around efficiently and be able to manage all that workload and keep GPUs 100% utilized is actually pretty hard, mathematical, statistical, AI system software and even hardware-level optimization. So we will continue to do that, and just in the last month, we've doubled our performance on Hopper for inference and we'll continue to do so, and you'll see in -- as we continue.

Q - Vivek Arya {BIO 6781604 <GO>}

And do you think that the industry has the right cost structure for generative AI inference at scale? I see that more as a user, when I go to take your pick of search engines, right, whether it's Bard or ChatGPT or what have you, even when we put in inquiries today, it takes several seconds to get an answer, right? It's a very different experience than we are used to in traditional search engines. So do you think the industry is there?

Today, it seems like everyone is training a lot of things and trying a lot of things, but do you think the industry actually has the cost structure to take generative AI and scale the inference side because I imagine that's what it will take to really grow this industry in a very sustainable way over the next several years?

A - Ian Buck {BIO 18454865 <GO>}

Yeah. It's a great question. Today, most of the live inferencing you're experiencing, of course, is on previous-generation GPUs. That's just naturally when it was originally developed and optimized and deployed on. And many of our large customers actually just now are bringing on their Hopper variants. So you'll see that 8X, you -- I mean, from -- so in terms of our performance, where we were -- so in terms of our performance where we were, with honest, where we were (Technical Difficulty) from the hardware side and activate those capabilities and another bump again from the software side.

So I expect the interaction that you guys are experiencing to get better and get more intelligent. I think there is a fixed latency that we all want to experience and then it becomes a question of size and capability of the model that can fit in that latency window. So the process of continuous improvement.

You asked about search. Can every search I type in be take advantage or be fully optimized if it takes this long? There are aspects of generative AI and language models that are being used today that you may not know. When type in the search, they're not using those words literally to index in. They actually are applying language models to generate more optimized query string to search on based on your history and other things.

So we are seeing aspects of that. We also see things like transformers and large language model technology is being applied in last-mile recommender systems. So as they get down to the last 100 documents or piece of information they wanted to understand and adjust to produce a result. Can I run a smaller and more constrained transformer base model in order to provide the last mile recommender from the tens or hundreds or a thousand whatever I can afford last mile for the recommender?

So you are seeing some of that technology being deployed today and to put on GPUs today. The next click up, of course, we'll be having more richer experience with search. I expect to see more of that with Hopper. It may take a few more clicks. With every generation of our GPUs and with every invention of new software optimization techniques and every invention by the community, whether it'd be the -- with the next Llama to (technical difficulty) GPT, and we bring down the cost of inference.

Hopper product A1 are brought down by 8X. TCO is also on the order of 5X. And the -- you compound that with continuous software program and compile with new model and algorithms techniques, there's an order of magnitude more capability that's going to be available to everyone. And the best part, it's on the GPUs they've already purchased and that's already there. In fact, this performance we're delivering with every one of these new piece of software where the performance that's capable with this more optimized GPUs, more -- I'm sorry, more optimized AI algorithms or models. Three, continue on this improvement in that TCO and that performance and that experience.

So it's a fascinating time. It's super busy. We're seeing new innovations coming in all the time, and it's definitely keeping NVIDIA and the community busy optimizing -- continuously optimizing the platform.

Q - Vivek Arya {BIO 6781604 <GO>}

Got it. I wanted to get your perspective. We're now on the competitive landscape. When we look at the the demand profile for NVIDIA's accelerated products, right, tens of billions, right, and expected to increase next year. Doesn't that give a lot more incentive to your hyperscale customers to create more custom ASIC solutions? One customer has already with the TPU product they have had a custom solution for a long time. There are -- the others are -- there's a lot of headlines about them wanting to have it done.

So first of all, what is the right positioning of your product versus their internal solution? Do they use one for -- one kind of workload and one for the other? Or does it become a greater competitive threat for NVIDIA going forward?

A - Ian Buck {BIO 18454865 <GO>}

When we look at this with the -- with just happen at the GCP next conference that was a conference, I think it was two weeks ago. They announced their new variant of their processor on that day in their keynote, and in that same keynote, Jensen joined them on stage and talked about all the innovation that we're doing together with Google, both GCP, and not just new instances and you bring -- they announced GA of their availability of their A3 instance, but also the integrations of GPU into their Vertex AI platform. Many of the research innovations that are happening on GPUs inside of Google elsewhere.

It gives you an example of how the fact that -- well, hyperscalers absolutely have the means to invest and optimize and build something that may be tailored for obviously important workloads for their business. They continue to partner deeply with NVIDIA and our GPUs and our software teams. There's two big companies advancing what we can do together, helping -- us helping them. Us partnering together on many of the software platforms to continue to innovate and what you see -- and you see that. You see NVIDIA there as an open platform, of course, available in every cloud and as an open software ecosystem to help advance the state-of-the-art in AI and data science and accelerated computing holistically.

That lift comes from almost 20 years of investing in a software developer ecosystem. And you'll continue to see some of the hyperscalers, of course, building their own silicon, if they have a means to, to optimize for specific workloads that perhaps are taking focus on for their businesses, but they still remain in close connection with NVIDIA because they see the opportunity to not just serve a broader ecosystem, but also innovate NVIDIA platform for accelerating computing across-the-board. And that is something we've been quite comfortable with and it's been a good partnership, and we -- it was really evident on -- in that.

Q - Vivek Arya {BIO 6781604 <GO>}

Got it. Do you see that change at all as we are moving more towards generative AI? You know, just where the cost of training is so expensive and the cost of inference is also going to be quite expensive that do you think it increases their desire to bring on more ASIC solutions than they have done in the past?

A - Ian Buck {BIO 18454865 <GO>}

That's a choice for them to fit where they want to optimize and invest. One thing that is -- NVIDIA is spending and investing billions in R&D to optimize for generative AI for training and inference scale. And with every generation of our GPU, with every generation of our interconnect and InfiniBand and CX networking technology, with every innovation of NVLink, those things bring the TCO and increase performance dramatically, and also bring down the cost of training.

Now, there obviously motivated to scale up what they can possibly do in order to develop something uniquely advanced and uniquely new or different that they can capitalize on, but by working with NVIDIA, they can basically opt new leverage the billions of dollars of investment that we're doing on those core workloads of training and deploying for inference for large language models for generative AI in that work space. And it's a question for them of where they're going to decide to optimize and to take that step further to do something which may be different and doesn't necessarily take advantage of all the time and energy and investment that NVIDIA is making.

So that's the choice that they have to consider and make. We're going to continue regardless to innovate at a pace that they all mean -- that may benefit them and the entire. So we will continue to see those things happen. I'm sure it would make sense, but the -- our focus hasn't changed. It continues to swarm and innovate to increase our performance at lower costs and also increase capability for generative AI and large language model.

Q - Vivek Arya {BIO 6781604 <GO>}

Thanks. Next topic, Ian, I wanted to broach was this -- these emerging classes have kind of converged CPU, GPU products there, for example, Grace Hopper and your competitor is also announcing, right, some of their own products. So what are the pros and cons of using those kind of? I don't know whether converge CPU, GPUs the right we don't refer to them, but how do they stack up against the more discrete solution where I'm just using standard X86 CPUs with one or many GPUs? What's the pros and cons of moving to this kind of converged architecture?

A - Ian Buck {BIO 18454865 <GO>}

Yeah. It's -- we've been optimizing -- in the communities we are optimizing for accelerated computing and AI for 20 years. We've moved a huge amount of competition to the GPU at this point. So for many workloads, including many of the (Technical Difficulty) 95%, 99% of the computing is done of course on the GPUs and directly communicating with each other or through NVLink or across InfiniBand, and all of the CPU workload can be either -- is either a small or can be optimized or done in parallel in overlaps with the GPU competition.

What combining this, I appreciate you have to a high-performance CPU there to do the other tasks, usually around data perhaps scheduling, managing and coordinating the execution. And every time we increase our GPU performance, of course, we need to make sure that our CPU performance keeps up or we find -- so we don't leave out the -- the model like Amdahls Law.

Ways to manage that, one, to first use the best possible CPUs, which we encourage and do use ourselves. You can also adjust the ratio of CPUs to GPUs. Today, if you look at our DGX system, it's two CPUs for eight GPUs, but we can do one to -- we can go one to four. We can do one to eight. We can do of course do two to one, or in Grace Hopper, we went all the way to one to one -- one next to each other. That's one angle.

The other part, though is about conversion. What happens when you combine CPUs and GPUs and do something different than a traditional x86 architecture or CPU sitting over here and your PCIe to GPU over or there? First, by bringing the two converged together, you can dramatically improve the bandwidth, the communication between those two processors. Today, 100s of gigabytes a second versus the 60 gigabytes or maybe 100 gigabytes from PCIe connection. You got also to be a much more coherent so you can bring the two memory systems together. The memory on the GPU, which today we ship an 80 gigabyte HPM GPU, we're going to and we announced going into up to 144 gigabytes per GPU. But you can then connect it to Grace, and because the connection is so fast, the 600 gigabytes of memory around the CPU basically becomes a combined fast memory platform, allow you to run even larger models. Basically, effectively making 600 gigabyte GPU.

This activates certain different -- both lodged around larger models with a single platform, a single GPU, CPU complex, and it opens up new avenues for new kinds of workload acceleration with -- especially working on large datas, applications like vector databases, applications like rationale networks which you see a lot in the finance and fraud and e-commerce, also used for recommenders. These are very large datasets that often want to either be run on -- they can run -- you can run today across many GPUs, but could be run more perhaps optimally or a different TCO point by having a much larger CPU like Grace Hopper, 600 gigabytes combined in one because they've been tied together.

The third thing about convergence is that it allows us -- it's another vector for innovation. We can add things to a CPU that could -- can optimize for the workloads we already know about or other opportunities we see in the future to innovate in CPU ecosystem -- in the CPU space in addition, the GPU addition and networking and data center scale. And that -- you see that in the work we're doing with our DGX GH200 by connecting even more GPUs together. Having the excellent CPU-GPU ratio, having the NVLink, and having the large memory really gives a vision of the future of infrastructure for generative AI, one where you have basically 256 GPUs connected all the NVLink, and fully backed by 256 Grace CPUs. And because it's only can effectively access one extra flop GPU, which is an amazing generative AI platform for both training and extremely large language model inference, where we might need multiple GPUs connected optimally for the (Technical Difficulty)

So it's really those -- I mean, some of those three things, larger -- provide larger memory as a starting point for building block. It's a great scale for our platform for inference as a result. Grace Hopper fits any server. It's a complete complex for CPU, GPU and memory. It allows us to play with the ratios and explore different ratios for different kinds of workloads for CPU, GPU and it's an innovation space. So many innovations that we've made in Grace, while we're using an ARM-based core, the SoC architecture of Grace and how those cores can talk to each other is quite powerful and showing up in many of our benchmarks and it provides a great companion to those compute rich workloads that NVIDIA has been focused on for the last two decades.

Q - Vivek Arya {BIO 6781604 <GO>}

Got it. I know we only have a few minutes left, but I wanted to get your take on the last two questions, Ian. One of which is, what's the role of the networking stack in the optimized generative AI clusters? So how much of an advantage NVIDIA have because you're able to leverage InfiniBand? But then that InfiniBand changes over to Ethernet, then does it mean conversely, you will lose some of that advantage also because hyperscaler wants to move more to Ethernet? So first, what is the role of that networking as part of the cluster? And does anything change when it moves from InfiniBand to Ethernet?

A - Ian Buck {BIO 18454865 <GO>}

Yeah. A great question. So there is basically three interconnects at this point that our choice and how the design and deploy AI. There's NVLink, which previously was inside how GPUs can talk directly inside of a system, now going to some more of the rack and room scale. You have InfiniBand, which is certainly developed for HPC from supercomputing industry for the lowest possible latency and data center scale, and it really was designed for that. And then, of course, Ethernet industry established designed and of course for high manageability and capability, and comes with a rich ecosystem of all the features that, not just enterprises, but the clouds need in order to do manage a software-defined infrastructure.

What you will see is, of course, NVLink will continue to be very closely tied to the innovations we'll be making inside of our own GPUs, and there we -- it's come as fast as we can go because we know bringing those GPUs, as GPUs get faster that and we want to connect things as quickly as possible in order -- and continue to allow them to operate as one. And to get the lowest possible latency for inference, some of these giant models, you need to be doing techniques around model parallelism, which have extremely high inner-communication requirements so that basically can split the model this way instead of just that way to decrease latency.

As InfiniBand also continues to grow, its design point, of course, is the lowest possible shortage latency, and as a result -- as well as provide excellent bandwidth that it does. And we see that it does provide a significant performance improvement over leveraging perhaps Rocky, a converged Ethernet stack, which still has a lot of the -- which can deliver comparable performance. In fact, we support many clusters and many deployments in our cloud at scale with Ethernet, with Rocky, and it works great.

For the best possible performance, InfiniBand gets that extra click up and that basically comes from its HPC heritage of having the lowest latency with high bandwidth. And we can do other optimizations as well as in network competition. We're going to do some math inside the switch and inside the network fabric with InfiniBand.

I fully expect the Ethernet and we are working with the community to actually improve Ethernet's performance as well, and -- which is great because it comes with all that manageability and that's software-defined. And the two exist -- there three will exist in the ecosystem for a while and continue to get the best as three layers of performance and scale, and of course, requirements between reliability or manageability, or security or enterprise deployment versus maximum possible performance over time. The roadmap will continue for instance will go up. I expected to be staggered, but they'll continue to learn and absorb from each other those technologies.

Q - Vivek Arya {BIO 6781604 <GO>}

Got it. And finally, I would love to get your perspective on, where are we in terms of rolling out generative AI because when we look at applications, they seem to be in their infancy, right? There are not that many applications, but when we look at just the rate of the growth of your data center business, that seems to be a very big proportion of what is the total spending pie, right? So what gives you pause and thinking about, we are already such a big part of the spending pie, how sustainable is this growth rate for NVIDIA over the next several years?

A - Ian Buck {BIO 18454865 <GO>}

So it's a fascinating question to think about. Today, if you think about where we are in the growth we're experiencing right now, it's people taking their existing data centers and make optimizing them to incorporate more and more GPUs, more and more OM and generative AI workloads. And that may be coming from the hyperscalers themselves, enterprises wanting to get on board using the clouds, for example, or now seeing these -- the GPU regional and specialty providers also standing up infrastructure. But we're largely going into the data centers that already exist because you can't just build data centers overnight. It takes two years plus save to build out the infrastructure.

What I see is the world looking to cheap hit it to how they're building data centers in the future and we're seeing really exciting growth in. They all realize they need to build out more capacity, and of course, be able to build not just the data centers they have before, which were generic in nature, perhaps more CPU focus because that's the majority of the servers going in, now building everyone from hyperscale to regional to on-prem to basically building out GPU data centers at scale.

So if I look at the growth of data center build out, you can kind of see the opportunity for OMs continuing to grow beyond what's being able to in some cases, quite literally crammed into these centers they already have and then to establish where we are today versus the size of the opportunity and the size of the market just from a data center footprint growth capacity.

We've gone from being a corner of the data center to the what data centers are now being designed for which is really exciting, and it gives me the confidence in the continued growth of our business to see how much companies are investing, the world is investing and building out that infrastructure for all the different demand and all the different needs.

Q - Vivek Arya {BIO 6781604 <GO>}

Excellent. So on that exciting note, Ian, thank you so much for taking the time to be with us, sharing your perspective. Really appreciate that. And thanks to everyone who joined this webcast. I got another 45 questions on the chat so I'll see if I can work with Simona to help answer some of those questions. But really, thank you so much, Ian, for taking your time. It's immensely useful to get your perspective. Thank you so much.

A - Ian Buck {BIO 18454865 <GO>}

Always a pleasure, and thank you very much.

Q - Vivek Arya {BIO 6781604 <GO>}

Bye there. Take care. Thank you.

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.