Please note the following timestamps are approximate.

David Liu (Guest) (00:00):

The world of compute is changing quite quickly. There's a lot of different technologies, completely different architectures, heterogeneous compute going around everywhere. And I think if you're a data scientist or an AI practitioner of any type, it's quite important to at least be up to date on what works and what doesn't work from a very high-level perspective.

Peter Wang (Host) (00:20):

You're listening to Numerically Speaking: The Anaconda Podcast. On this podcast, we'll dive into a variety of topics around data, quantitative computing, and business and entrepreneurship. We'll speak to creators of cutting-edge open-source tools and look at their impact on research in every domain. We're excited to bring you insights about data, science, and the people that make it all happen. Whether you want to learn about AI or grow your data science career, or just better understand the numbers and the computers that shape our world, Numerically Speaking is the podcast for you.

Peter Wang (Host) (00:49):

Make sure to subscribe. For more resources, please visit anaconda.com. I'm your host. Peter Wang. This episode is brought to you by Anaconda, the world's most popular data science platform. We are committed to increasing data literacy and to providing data science technology for a better world.

Peter Wang (Host) (01:06):

Anaconda is the best way to get started with, deploy, and secure Python and data science software on-prem or in the cloud. Visit anaconda.com for more information.

Peter Wang (Host) (01:16):

All right. Welcome, David. So David Liu is joining us today. I'm very excited for our conversation. David is a staff AI engineer at Intel, and he works on strategy and vision for data science, AI, machine learning products. So welcome, David. Thank you so much for joining us.

David Liu (Guest) (01:35):

No problem. Thanks for having me.

Peter Wang (Host) (01:37):

Great. So before we get into it, and I think we're going to get into a lot of fun stuff today, I'm really looking forward to our conversation. Before we get into all of it, why don't you tell me a little bit about your journey at Intel and the kinds of things you work on day to day. Things that get you excited and maybe we can build off some of those things and see where the conversation takes us.

David Liu (Guest) (01:53):

Yeah. My history at Intel is interesting because I came in shortly after getting out of graduate school and doing my own consultancy.

David Liu (Guest) (02:01):

Actually, I went to the hardware group. I did pre-silicon hardware validation and then I think for about five years. Then we did the software group, working with you guys as a partner. So I was in the software group dealing with the Intel distribution of Python, our runtime libraries, and our open-source work.

Please note the following timestamps are approximate.

David Liu (Guest) (02:17):

And then about two years ago, I moved into the sales group for AI, where I work on kind of productization, a little bit of some of the sales strategy, and some of our coordination with our engineering partners within the company, and then also working externally. So it's kind of been a strange journey within Intel and it's led to some pretty interesting adventures that I continue to do today in my role.

Peter Wang (Host) (02:43):

Great. So as you say, you've kind of been through an interesting journey there within Intel itself and also even working on a lot of different aspects of the software and the hardware stacks. I guess not hardware stack, but on the software and the hardware side. And I think especially more recently in your role, you interface a lot with customers on the solutioning side and looking at what people are really trying to get done.

Peter Wang (Host) (03:06):

So right now, the current state of the industry, of practice, as well as what is available from hardware from the various hardware manufacturers, what is your view on that? What is your general sense? Do you think people are doing kind of a [inaudible] job? Do you think we're falling far short of the mark? Do you feel like there's a lot of exciting hardware coming out or do you think hardware's sort of seen as kind of static right now? Just give it all. Tell us your perspective.

David Liu (Guest) (03:29):

Especially after being both on the inside and the outside, what I can say is we're not really doing a good job of using all the types of hardware. There was a start, especially in the AI space. I mean, let's just say the competition did a really good job of talking about GPUs and GPUs do really good compute for certain types of AI. I mean, incredible work, but then there's like the other 80% of the AI space.

David Liu (Guest) (03:52):

Those are: classical machine learning, graph, general data science work, and really a lot of that world has been untapped. I mean, I made it kind of a journey of myself and goal of myself to go and explore a lot of those aspects of hardware. And luckily at Intel, I kind of have a large portfolio of products to go play with.

David Liu (Guest) (04:11):

And diving into, say the last two years during lockdown, I worked on my project for the data science workstation that I discovered. Actually what we thought data science performance comes from is completely, probably just misguided or incorrect.

David Liu (Guest) (04:25):

It’s memory capacity can solve a lot more problems than we thought we could. Mid core count resolves oversubscription. You don't actually want high core count for a lot of things. It's like a lot of things have just been dispelled and then I'm now exploring FPGAs and what they can do. Even some of the aspects of... Where does A6... Where does their price performance value really hit home? How does it work on edge devices and IoT? How do you do enterprise-secure AI with enclaves, federated learning?

David Liu (Guest) (04:56):

We're not doing a good job identifying the nature of the AI and understanding how it works in the hardware. I think both a lesson of the industry not really being the same group. The AI kind of workers aren't the same group as the hardware engineers. And so we've never really gone into that discovery. And we've kind of let, let's just say buzzwords take it over for better or for worse. And I think that area is just very unexplored.

Peter Wang (Host) (05:22):

Well, it's interesting, right? Because there's so many factors that come into this, and we'll talk about all these things later in the conversation: the FPGAs, the data science workstation.

Peter Wang (Host) (05:31):

But the thing that strikes me about this is where I got my start in doing Python, kind of in business computing, was on Wall Street. Hedge funds. Well, they were not literally located on Wall Street, but it was in the finance sector. And so we had two very different kinds of clients. We had the high frequency and the hedge funds who, high frequency cared... Well, both of them really cared a lot about performance and they understood performance and they knew it was a matter of life or death to do accurate modeling at scale and to be able to operationalize that.

Peter Wang (Host) (06:00):

And then when we got involved in working with banks, they care about performance too, but a different way. Because the particular projects we worked on were these large-batch scale economic simulation and sort of finance simulation computation engines. Really just doing supercomputers for finance.

Peter Wang (Host) (06:15):

And both those groups really cared a lot about performance. And of course, Python had a rap back then and maybe it still does that, "Oh, it's a slow scripting language." And then it's like you show up with NumPy, this wrapping, a bunch of MKL. It turns out it's pretty fast actually.

Peter Wang (Host) (06:29):

So we had a good time with all that stuff. But then as the use of the Python numerical stack kind of made its way out into people doing essentially CSV-file data science, a lot of people doing that stuff had no idea where performance came from, how things performed. They kind of had a MacBook and it was sort of like, "Well this is the thing I'm going to use until I learn how to use the cloud. In which case, then I'm just scratching my head about how many core hours? How much memory?"

Peter Wang (Host) (06:53):

But for the most part, people are stuck on laptops and it's a hard time to requisition new hardware from IT. Even if you convince them that it's worth it, you then have to wait six, nine months to get it, right?

Peter Wang (Host) (07:05):

So the ironic thing is the data science world. I feel like they don't have... Everyone gripes about how long certain things take, like data munging and all this stuff. People do try to learn a little bit better, but they don't really eat up performance the way... Or live performance the way that the finance people I worked with 10 years ago did.

Please note the following timestamps are approximate.

Peter Wang (Host) (07:22):

On the flip side of it, AI researchers are out here talking about how much inference they can do and FLOPS per watt and all the stuff I'm used to hearing from supercomputing people. So is there a way to bring some of these things together? Is there a space to have a conversation to say, "Look, here is the minimum thing you need to know in order to understand why your stuff is slow. How you can get 10X performance."

Peter Wang (Host) (07:41):

Because 10X? People will learn. They'll stop and learn some stuff to get a 10X improvement in quality of life. But how do you feel like we're going to get there? Do you think we'll ever get there?

David Liu (Guest) (07:51):

I think we will get there. A lot about it is the two worlds have different types of motivations and different requirements. And so let's say for example, that the high-end researchers are one end of the spectrum and what do they need in order to communicate with IT or data scientists? What needs to happen? Because you have kind of the areas of... You're going to reach the use of AI from a different methodology. Although different walks of life too.

David Liu (Guest) (08:19):

You're either going to be somebody from actuarial science, statistics, mathematics background. You're going to know nothing about hardware, software, and that's one, that's extremely common today is most of them are not developers. So that's a huge, huge element that I don't think we [inaudible] should solve. That should just be... It's a persona.

David Liu (Guest) (08:36):

The next persona you're going to have: kind of the high-end researcher. Other ones, you can have ML engineers. You can have data engineers. You can have people who are just pure BI. There's a lot of personas that utilize what we consider both data science and AI. So the question is how do you actually connect those together?

David Liu (Guest) (08:53):

It's going to be about educating the community as to where the hardware and the software interact for the paths that they use for their daily work. How does pandas get performance? Where does NumPy performance come from?

David Liu (Guest) (09:05):

If you're doing... What type of model? Is it a memory bound or compute bound? Now, how do you communicate that to your IT? Or how do we educate IT? If your data scientists are doing this type of work. Say it's all HFTs or genomics. These are the type of characteristics that are going to happen. And this is how you should outfit your organization. So at work right now, we're working to get a lot of collateral created to actually educate IT. And those are not necessarily just a sales thing, but from educational perspective, as we are researching and discovering this with our partners, it's kind of our job to go and educate, on both ends, what the requirements should be to help them speak the same language.

David Liu (Guest) (09:48):

So now the data scientist can communicate to IT, "Oh, I'm doing this work. It's getting this type of performance. Here it is in the doc."

Peter Wang (Host) (09:54):

"I need this class of workstation for this kind of work."

David Liu (Guest) (09:57):

Yeah. "I need this," right? It's like deliver what type of workload...

Peter Wang (Host) (10:01):

Yeah. The interesting thing is, as you're talking about this, something occurred to me, which is data science and data analytics is not the first area of business computing or professional computing that has hardware needs. There are many others. Computer graphics. 3D digital artists are able to communicate about what their hardware needs are. Digital photographer and video processing people. People who do engineering simulation, running Pro/E, CATIA, whatever kinds of stuff to do engineering. Then people running Ansys to do physics simulation.

Peter Wang (Host) (10:31):

There's a lot of people out there who are not software engineers. A 3D digital artist is not a software engineer, but they're able to have an adult conversation with IT about, "I need this box to do my work." And IT's like, "Yep, here's a box."

Peter Wang (Host) (10:45):

And so part of what makes that possible is because the software packages they use are actually proprietary. And those companies sit there and they characterize and say, "Yeah okay, you want to edit 4K video, then you're going to need this kind of box. You're going to use Premiere to do this video. You want to use whatever Pro/ENGINEER to do this other thing." The ISV guide the buying.

Peter Wang (Host) (11:05):

But when it comes to the open-source world, I guess maybe Anaconda could do something like this, but we can't speak sort of ex cathedra for NumPy and SciPy. And the developers of NumPy and SciPy, they're working really hard just to keep the projects going. They don't have a lot of time to sit there and characterize and make hardware recommendations. So I feel like there is something missing here when these open-source tools are the big things that people are using…

Peter Wang (Host) (11:29):

How do we, as an industry have a bit of a vendor-agnostic place where we can just talk quite honestly about, "Okay, if you're doing this kind of work, if you're doing this kind of analysis, files of this size will require machines of this kind of spec."

Peter Wang (Host) (11:41):

Because I think people will say, "Well, okay, you're doing AI stuff. You need a GPU." Well maybe, but sometimes yes. Oftentimes yes, but not always.

Please note the following timestamps are approximate.

Peter Wang (Host) (11:49):

And over here, if you're a data scientist, you're like "I have my MacBook and that's what I've got. And if it doesn't fit, I'm going to go and try to learn Kubernetes, I guess."

Peter Wang (Host) (11:56):

So somewhere between this, we need to have better sensemaking about what performance really means for modern data analytics, because there's a lot you can do on a regular machine. There's a lot you can do. Most people are not taking full advantage of them.

David Liu (Guest) (12:07):

The laptop solves, I think at least 50% of the work that I do. And then the minute it goes past 64 gigs, I start pulling up the bigger local machines and past that I start pulling up a server rack, a 4U server rack. And you can also use cloud as well, but there's a lot of hiccups and I have some great little bits on that too.

David Liu (Guest) (12:27):

I think from a fundamental perspective, when I look at the AI space and I look at the other classical spaces that do video and audio, because I do a lot of that work myself as well, I would characterize it as... Both the data science and AI space are very early. And really the industry has never had a situation in which it's had to respond to "How do you use open source that's not tied to an ISV to then go and educate the community or your end customer about it?"

David Liu (Guest) (12:56):

It's never been a consideration. I mean, let's be honest. Most of the hardware companies and OEMs really didn't embrace open source until very recently. So we are dealing with a large gap that I think, for me, I feel it as my personal responsibility to go work with my company. And I would urge others that work at other hardware companies or OEMs to go and do the same. Because from that standpoint, we want to make sure that this large community of domain scientists are actually getting the performance that they expect. And they have a clear understanding of why or why not they are getting that performance.

Peter Wang (Host) (13:36):

It's kind of a thing of like... If you have several different car companies, everyone wants to sell more of their own kind of car. But at the end of the day, it behooves everyone to invest in driver education. Because if people get in a car and they don't know how to shift out of first gear you can't sell them a fancy sports car; if they drive around and they're constantly running into the light poles in the parking lot, they're not going to have a good time. And they're not going to see the motivation to procure whatever next-gen thing you may be building.

Peter Wang (Host) (14:01):

So I think back to the ‘80s, early ‘90s maybe, and things like *PC Magazine* were out there. They were educating, well, there'd be reviews. Consumer personal computing magazines would have reviews of 2D rectangle fill speed for different kinds of chips from different vendors and different PCs, running different versions of different kinds of software.

Peter Wang (Host) (14:27):

And it's like your average person doing productivity software, doing spreadsheets and stuff, they don't know what blitz speed means for rectangles, how many rectangles per second can be drawn. But at least these magazines are trying to educate users about the differences in, what does it mean to you, to your daily productivity if you care about these different machines? You buy this Gateway machine versus that Dell versus Micron machine. You get different kinds of productivity speed if you want to do 3D graphics or 2D graphics, business graphics kind of stuff.

Peter Wang (Host) (14:54):

So I think we're maybe back in that mode where all of the hardware companies, because there's such a gap between the users to the hardware companies, there's the missing commercial ISV space with data science. It's all open source. That gap has to be bridged by someone. And we try to do a little bit of that, but we can't do all of it, but I think that's absolutely right. That's an area where the hardware vendors are going to have to invest a little bit if they want people to actually care about whatever next-gen hardware they're producing.

David Liu (Guest) (15:19):

I think it also takes a bit of, I guess the way I would characterize this as... It takes a very specific set of skills and motivations for a person who's in a hardware company to actually go and explore that at a hardware company.

David Liu (Guest) (15:32):

I mean, it took an understanding of both hardware and software backgrounds for me to go and do the research that I did for data science performance. And I didn't even do it down to the specific algorithm. And it took a lot of effort to convince management to go and give me this amount of servers to go and do the scale-up tests to understand exactly what memory pattern like K-means has versus PCA.

David Liu (Guest) (15:54):

It is a very strange type of motivation that has to be driven within each manufacturer to go and discover it, publish it, and share it. It's almost like a completely different business problem that has to be solved of now basically being a research organization within your company and bringing that publishing out. That's the way I see it. It's kind of a strange positioning and that might be one of the reasons it's been a challenge.

Peter Wang (Host) (16:23):

And also it's an emerging area. So there's definitely... Even if someone were interested, there's a lot to learn and things are changing all the time. But I think maybe one of the takeaways also here is to think about talking to the IT manager, or the CIO, right? And they of course want to support whatever their business's AI development initiatives. And I don't necessarily know that they're getting any really good information about what that really means.

Peter Wang (Host) (16:46):

And something I've heard you say in the past is that AI isn't a singular compute type. There's GPUs, there's CPUs, there's even FPGAs, as you mentioned. It depends on what your data scientist is really doing. And so I think in a lot of enterprise IT, there is a desire to manage large pools of relatively homogeneous compute or compute resources, infrastructure, and offer that up to many different lines of business or user groups.

Peter Wang (Host) (17:10):

But with the landscape of what AI looks like, if you could make some users 10X, 50, 100X, more productive by getting the right... Somewhat more specialized or customized compute for them, it really behooves you to do that. And so, do you feel like there's a push there to be made? Or do you think that that's going to be a really big lift to get IT to think differently about that?

David Liu (Guest) (17:31):

So we are actually actively trying to work on that. So we're actually working with multiple market reporting agencies that are common that CIOs get their information from. And the biggest takeaway that we want them to have is when they're posed with the question of, "We're running AI," the very first question that should come out of their mouth is, "What type of AI? What are you running?" Because that then starts to delineate what style of compute you're going to need.

David Liu (Guest) (17:59):

And I think that the vast majority of people in the industry today use homogeneous systems of one type or another. And I think the thing that I'm finding is, and that many specialized customers are already finding is that it's heterogeneous. It's like you need this type of hardware for this style of compute to be quick. This helps with one stage of the pipeline when you're working with a multi-stage model or an ensemble, and having the information available to the CIO is extremely useful.

David Liu (Guest) (18:31):

We've identified that it's a persona who has to be really well educated from the hardware side to have a clear understanding of what needs to be delivered because ultimately their operational expense, they need to make sure the ROI is there. And that is a problem where we're actively trying to work on in terms of education.

Peter Wang (Host) (18:49):

I could imagine if you offered a very small course... Some kind of a certificate or something that you someone could say, "Look, I went to the Intel ML Performance Bootcamp and coming out the other side of that as an IT guy or as MLOps guy or whatever, I have a lot more information as to how to even think about the problem."

Peter Wang (Host) (19:10):

Because the issue here isn't that people are thinking about the problem wrong, it's that people aren't even thinking about the problem. And just this idea of asking... Because when you ask a software development group, "What hardware do you need to build this app that you're building?" Some internal business app.

Peter Wang (Host) (19:22):

They'll say, "Okay, we need a database of this size. We expect this many transactions per second or per day. We need hardware at this level. We're going to run this version of the JVM. We're going to run this

.NET stack." You know, you sort of spec out the hardware you need and you turn on the software and it kind of goes.

Peter Wang (Host) (19:35):

And so I think IT is used to servicing that kind of thing. Actually more and more of that becomes a self service. If you think about a platform as a service or infrastructure as a service in the cloud, you just self-service and that stuff.

Peter Wang (Host) (19:44):

But with these kinds of things where you have very specialized hardware configurations that are not some of the basic systems you might spec out for those kinds of apps, IT just has to do more work. And I think you're right. You guys, if you want to move more hardware, you Intel and other hardware manufacturers have to make the case for why that hardware makes a difference. What is the ROI? Data scientists are more efficient, you can reach more accuracy with the same amount of cost investment, et cetera.

Peter Wang (Host) (20:09):

But all that being said, how many people are thinking about FPGAs for ML/AI? Tell me about that. Educate me.

David Liu (Guest) (20:18):

Oh, it's incredibly rare, incredibly rare. I mean, I started playing around with our company's product line recently. If you go look on a few distributors' websites, you can buy some of the newer-generation FPGAs that are quite expensive, but we're releasing various lines that are targeted at different price points coming up in the next few years.

David Liu (Guest) (20:41):

And so for me, it was an opportunity to envision if I were a scientist or I was a developer of say a core maintainer, say like scikit-learn or NumPy, how would I want to play with this? At what level would I want to play with this at? And what would I want the user experience to be? And I think that there's not that many people who think about that because you're thinking about the people who are enabling essentially the framework designers, who then those people then enable the end customer.

David Liu (Guest) (21:11):

So there's a few pieces in that chain that have to be knocked out first to do the initial R&D, do the pathfinding and let's not even talk about the fact that the software stack is... You're trying to mesh a complete C++ world with a Python world. That is a completely new space.

David Liu (Guest) (21:28):

The thing that comes up to my mind is Numba. I'm like [inaudible] spits out LLVM IR. Okay, fine. I'll just throw it at Numba, right? In my head, that's what I have going on as I play with this hardware. I'm like, "Ooh. Where should this go?"

David Liu (Guest) (21:40):

I think the hardest part is just a change in workflow. Because if you think about it, you have to program down to the FPGA. Let's say, for example, you start up a Jupyter notebook. You would write out your code, you put a decorator on, and then…that decorator would program to the hardware and then you'd run your data through the bitstream of that.

David Liu (Guest) (21:57):

But then when you close it, the FPGA is in an unknown state because it still has whatever's on there. You should blast it. So the use case and how this would be used, that's like one where you're doing explorative. But then if it's fixed function, then it's used like a pure accelerator card, and it's an extension of whatever it is. So I think the use cases are probably the crux of the problem.

David Liu (Guest) (22:19):

It's like I have to identify a few personas, like the scientist who wants to explore different types of accelerators, but in the Python and NumPy space. And then I have the types of users who say, "I want an alternative to GPU where GPU falls short because the FPGA can do a lot of parallel signal processing in places where GPU cannot compete," right? I mean, that's why FPGAs are classically used for signal processing.

David Liu (Guest) (22:43):

I used them a lot when I was doing AI that had a lot of radio signal filtering first and you'd have to clean up the signal a ton. So the GPU had trouble with that. So we just stuck it all in FPGA and that worked fantastically. But it's a much tougher, but more flexible type of compute unit versus say an ASIC or Habana-based ASICs. They're fixed function. They do a specific style of AI with a few different variations. And they're great for what they are because they don't have to be modified as heavily or don't have a weird use case, but you lose a little bit of that flexibility.

David Liu (Guest) (23:16):

So along that continuum of like CPU, and then you go all the way to fixed function ASIC, there's all sorts of compute in the middle. And you then have to go and try to determine how to use all the ones in the middle that nobody's ever played with or the ones at the very end that nobody's ever played with.

Peter Wang (Host) (23:30):

Well, and this is the thing that already in the Python world we have at the software level, we suffer a little bit of what is it, an embarrassment of riches. A lot of different tools for doing all sorts of different kinds of things. And just getting users educated about what tool they could or should be using for a particular thing is really hard.

Peter Wang (Host) (23:49):

So we see all the time is people... I think about just something like graphing, for instance, or charting, plotting, whatever you want to call it. A lot of users, they learn some seaborn, they learn some Matplotlib, and they just kind of get by with it even though for their particular statistical analysis or for their particular thing they want to visualize, there may be other tools, more advanced, that can really make a much better visualization of what they've got and tell a much better story. They kind of go with what they know.

Peter Wang (Host) (24:15):

And I feel like with cranking away at numbers, like the basic bread and butter, that these kinds of tools, these kinds of hardware things can accelerate… It's almost like you have to fight. If you want to deliver a new technology in this space, you have to fight against the incumbent inertia of, "It just works."

Peter Wang (Host) (24:32):

I can learn Numba to decorate this and give it a target and it'll automatically compile an FPGA. And that's all amazing. But if I just sit there and write a for loop and I can just go grab a coffee and come back, that just works.

Peter Wang (Host) (24:43):

So if you think about not just getting people to use it, but then the debug pipeline. If something breaks or something isn't right, how do you even understand what went wrong? All of that complexity. I feel like for this stuff to really get adopted at large scale, you have to make it much simpler and make it much easier for people to understand, "If I have X kind of problem, I need to use Y kind of stack."

Peter Wang (Host) (25:05):

And it almost has to be that prescriptive and that simple for people for them to even have the motivation to play with some of the new things. Do you agree with that? Or do you think that's underestimating people's motivation?

David Liu (Guest) (25:20):

I would say there's a large element of truth to that. I mean, personally, if something seems a little too hard, I'm like, "Eh, I'll just throw it on my big workstation and come back. It'll be done. I don't need to deal with cloud. I don't need to deal with distributed." And that element of "It just works" shouldn't be ignored because if we're not making it easier for the end practitioner, then I think we're doing a disservice to the community.

David Liu (Guest) (25:44):

I think what needs to be explored is how do you make that "It just works?" What does that experience look like? And when you do have a problem, what does that experience look like? And how does it get debugged? And so the user experience across every type of hardware and framework today... Like I said, it's the Wild West right now. We're still very early in this and creating industry standards for what a debug session should look like in AI. That hasn't even been talked about.

David Liu (Guest) (26:16):

And it's very entertaining to me, but these are the types of problems that I busy myself with as well, because we're incentivized to try to engage with our customer and understand them so we can solve their problem. And I think from one aspect, we've been looking at a lot of…just kind of stomping out the small smoke and fire of like, "Oh, there's this new technology. Go deal with it."

David Liu (Guest) (26:39):

We're not looking at the core fundamentals as if you were to give me all of the algorithms today in let's say all of classical and then some of deep learning, and some of graph, I should be able to look up some table and say, this is the type of algorithm and this is how every type of compute handles it and what's the most important thing…

David Liu (Guest) (26:59):

For K-means, it's completely different from decision trees, to linear regression. Everything will stress CPU, cache lines, amount of cores, CPU to CPU, bandwidth or it'll stress pure memory capacity. Most of data science is almost all memory capacity. And then you have memory bandwidth to the memory for things like decision trees. And then it's core to core for things like K-means.

David Liu (Guest) (27:24):

But where is that information today? There's no place to go and discover what I'm talking about. And I think that those elements have to be detailed at some point to create that easy mode situation where if I go and do it... Say I go up on scikit-learn's docs. I should be able to go, "Oh shoot, okay. I might be using the wrong type of hardware for this, and it's going to be hard to debug. Maybe I should just go use something else." That's how easy it should be, in my opinion.

Peter Wang (Host) (27:54):

As you're talking about this, I'm thinking about the... Performance is such a weird thing as a value proposition from a business standpoint. Because I had a conversation with... It must have been supercomputing 2011 timeframe. I was chatting with Mike McCool, who was... His company had just gotten acquired by Intel at the time. And he said, "What people don't understand is that if they're even getting 1% of peak performance of their processor, they're doing great."

Peter Wang (Host) (28:19):

I mean, he wasn't being facetious. He was just saying, most people are leaving 99% of the performance of their processor on the table. And at the time I found that to be this really interesting statement because I thought, well, we should do something about that. We can make the open-source tools better, take advantage of all this stuff. Performance [inaudible] all this stuff. Because people are complaining on mailing lists about things being slow or "I wish this were faster, that were faster."

Peter Wang (Host) (28:42):

But then when you do that, you realize "Actually most people don't care about raw performance. What they care about is: does a particular thing…if I'm a human in the loop doing a thing and I'm building something out or exploring a model or doing whatever. There's a very bimodal distribution of tasks.

Peter Wang (Host) (29:00):

There's things that take less than five seconds. And things that take more than five seconds. And I don't want to have to think about "How hard was this thing for the computer to do?"

Peter Wang (Host) (29:10):

All I'm really experiencing is how long I have to wait for the last spinning hourglass to resolve, how long before I get a result back? The people typing the code, more often than not, have no way of modeling how long any particular thing will take. This is sort of the fundamental problem is that they're casting spells into the ether and sometimes it takes a millisecond and sometimes it could just take 10 minutes before it runs out of memory.

Peter Wang (Host) (29:37):

So I guess the point of all that was just to say, if you're selling performance, if you want to give people this data science workstation, which you should maybe describe that a little bit, but then if you're going to give people that box, how do you make them really care about the value that it provides in their workflow? The experience of it as a user in the cockpit, fingers on keyboard typing stuff, you really have this "Oh, it was fast" versus "Oh this just took forever," bimodal kind of experience of latency.

David Liu (Guest) (30:07):

Yeah, it was really weird. I think the aspect that I was trying to solve was the one where either you run out of memory or you can't actually do any of the data exploration on the system.

David Liu (Guest) (30:18):

So one of the challenges is if it doesn't fit in system memory, you have a limited set of algorithms and tools and the pandas API is cut by three quarters. So you're essentially removing a lot of the industry standard tools and capabilities that you would use to actually do the discovery. And so the way that I was looking at it, performance, to me, was being able to explore it at all. And so it almost became a binary…

David Liu (Guest) (30:44):

It's like, if you can fit it in memory, then you technically solve the problem. It doesn't matter how long what some calculation will take because you can do it versus trying to do your own merge and group by Lambda in distributed land is like, you're playing with fire. Whereas I'm like, I'll just stick it all in memory. It's no longer a problem. It's no longer a big data problem.

David Liu (Guest) (31:04):

I just use pandas and cut it up, and I might come back in three minutes and it's done. And that's preferred over sitting and programming towards all sorts of different frameworks and cloud and other types of really weird problems that are unforeseen. And that's the user experience I was solving. So in that context, performance was, "Can you get it done or not?" Not "How fast can you get it done? Can it even be done at all?"

David Liu (Guest) (31:29):

Now if we start flipping the coin to a lot of the inference deployments that we see in industry, it's really funny when I talk with end customers, because they don't necessarily know what performance metric they're trying to hit. So a lot of the times they're like, "Well we need this much throughput."

David Liu (Guest) (31:47):

And I'm like, "Well how often are you running this?"

David Liu (Guest) (31:49):

They're like, "Once a week."

David Liu (Guest) (31:51):

And I'm like, "Do you really need it that fast? Isn't that quite expensive?"

David Liu (Guest) (31:55):

Or some of them are saying, "Well I need it for this type of video."

David Liu (Guest) (31:58):

I'm like, "60 frames a second, right?"

David Liu (Guest) (31:59):

They're like, "yeah."

David Liu (Guest) (32:01):

I'm like, "Okay so you need this type of throughput. You actually need this type of hardware. Is it running inside of a facility that has enough cooling?"

David Liu (Guest) (32:07):

They're like, "No it's literally remote."

David Liu (Guest) (32:09):

I'm like, "Ooh there's some adventure in that one." And different types of accelerators have heat requirements that you can't cool down. A GPU is quite hot at the edge in certain cases. So it is playing with those elements of performance or even in instances where you're doing a... I guess you would say collaborative filtering or any recommendation engine stuff. How fast do you actually need it? That's based upon the human response time.

David Liu (Guest) (32:35):

So they've done tests on any of the marketplaces and for average human, when you click it and you eventually wait for it to load, it's literally dictated by the latency of whatever network you're on. That latency is... Your performance metric is, "I just need to get it done within this big chunk of time" versus, "Oh, I needed to have it subsecond." And so that's...

David Liu (Guest) (32:57):

Performance will change depending on what the end use case is. And I think that's where that spectrum of performance... You get high price to lower latency, to higher power cost, to it's a ton of... I guess you would say it's a lot of dimensions to the problem if you want to look at it, but that's what hopefully we're trying to help solve in educating customers, but also some of the collateral that we're putting out.

Peter Wang (Host) (33:24):

Well. So in terms of... And I agree with you that feasibility is the first optimization as they say. So if you do it at all, that's better than not being able to do it. And then if you do it faster, that's even better. But that's a “nice to have.”

Peter Wang (Host) (33:35):

But I absolutely... There's this energy as you're describing this as like, "Man, this guy really doesn't like to do distributed computing. What can I do? How many more RAM chips can I jam into this motherboard so I don't have to go and figure out how to make this into a distributed algorithm."

Peter Wang (Host) (33:50):

And then I think a lot of people can appreciate that because there's a sense of at the end of the day, being on a single machine, logged into Jupyter Notebook or sitting there running a script at a command line, it all sort of still makes sense.

Peter Wang (Host) (34:02):

You kind of know what's going on. Here's a file system. You can look... Here's how many processes there are. You can look at how much free disk, free memory. It's easy. It sort of fits in your head.

Peter Wang (Host) (34:11):

The instant you go to distributed it's like, "Oh my God. What is going wrong here? What proxy didn't talk to something else? What thing is hanging because some JavaScript thing couldn't connect to some Jupyter front end thing to some other thing?" It becomes rapidly a complete cluster. Pun unintended.

Peter Wang (Host) (34:26):

So I think you'll find a lot of people who agree with your perspective of, "If I could just do work on my local machine and just get it done, that'd be great."

Peter Wang (Host) (34:35):

But in terms of the pitch for the data science workstation, maybe the thing there really is to just say that "even if you can't handle the full data set on it and you have to subset, you can get a richer subset. And you can do many algorithms on it actually even faster than you could on the cluster because it's all local and the latency, especially for things like graphs and things like that, where access time, random access time is by far the determining factor on performance."

Peter Wang (Host) (35:01):

Things like that I think just give people a sense of "This is the machine you should use if you're doing data science, trust me. It's the machine you should use." And just leave it at that.

David Liu (Guest) (35:11):

You can use more of the data set in memory that you can actually play with. And that'll give you a clearer picture versus an even smaller subset. So the idea here is just to do the best effort and due diligence to get there. And that has primarily been how we've been trying to tell our customers the value proposition of why one of these machines might actually solve their problems.

Peter Wang (Host) (35:33):

Well, so another thing on the flip side of running out of memory is what if your memory was persistent in the first place? What if you didn't run out of memory because it was essentially just disk?

Peter Wang (Host) (35:43):

So when the 3D Xpoint technology first came out, I was very, very excited about all this stuff. And now of course it's rebranded as Optane. Tell us where that's at. Can I get one?

Peter Wang (Host) (35:54):

There's a lot of really interesting stories about these things, but it felt to me, if you could actually have one of those things plugged into a data science workstation, your data's all there and you could put all these different compute things around it, that'd be a far more efficient model.

David Liu (Guest) (36:06):

I think the scariest part is you predicted what actually would happen and that's kind of funny is the data science workstation, to get that large memory pull, it uses Optane persistent memory in non-persistent mode and is a memory pool. And that's how you're able to get three or six terabytes in some of our OEM partner machines to get that type of single memory, memory density.

David Liu (Guest) (36:33):

When we look at technology as technologists or nerds or whatever we want to call ourselves, scientifically interested nerds. We start looking at these technologies that some companies put out and they're like, "Oh, that's clearly a solution looking for a problem. But when it does find the problem that it's meant for, it's pretty amazing when it actually does work. And it was interesting to actually go and test it for the first time.

David Liu (Guest) (36:58):

And I remember having the call with the client group who makes our workstation products and said, "What if I built a system that did this, is this possible?"

David Liu (Guest) (37:05):

And they're like, "Well theoretically, if you ran the machine with this mode, yes."

David Liu (Guest) (37:10):

And I'm like, "Could you build that and send it to me?" This is the middle of a lockdown.

David Liu (Guest) (37:14):

And they're like, "We're not supposed to send it to your house."

David Liu (Guest) (37:17):

And I'm like, "The work site's closed." And then I sat there and I played with it and that was just an awe-inspiring moment to actually see something that, like you and I said, as a nerdy technologist, being able to explore with that and it actually doing what you think it does was an amazing moment.

David Liu (Guest) (37:34):

Now I'll address the availability. So interestingly enough, Optane persistent memory was kind of sold as a solution through to OEM so the average consumer cannot actually buy it. That's also the same with some of our server products. You have to go through an OEM to purchase a full system.

David Liu (Guest) (37:50):

We are working with our OEM partners to make them installable after the fact from their systems. But also the availability is tough because I did it on a system that had already been validated. So there's all sorts of hoopla on re-validating a system to do this. So it's complicated, but that's why we're working very, very hard with our OEM partners to make the system available because there's a lot of excitement and those who have gotten to play with it are just blown away.

Peter Wang (Host) (38:20):

Well, and so I guess we haven't quite explicitly said this, you just dropped some numbers, but when we talk about this data science workstation thing... And this is, I guess I'm a little remiss, because I didn't give you a chance to explain in more detail, but one of the key defining aspects about this thing is that, whereas now most peoples' workstations on a laptop, you'll have 16 gigs of memory, maybe in a server kind of thing you'll get a 64, 128 gigabyte memory, but with the data science workstation that you've been building and working on there, what is the storage size that we're talking about there?

David Liu (Guest) (38:47):

So depending on how many DIMM slots you have per-

Peter Wang (Host) (38:50):

Memory size, right?

David Liu (Guest) (38:51):

Yeah. So the memory's capacity per socket. So that's one and two socket depending on which type of system from what OEM you get and how many DIMMs per socket. So if it's eight DIMMs per socket, I think it's four terabytes. If it's 12 DIMMs per socket, then you can get six terabytes and that's for a Cascade Lake machine that most of our OEMs sell. I think if you get a Supermicro machine, you can buy their workstations. I think they have 16 DIMMs, which is ridiculous. And so I want to say it's up to six terabytes per socket. It's ridiculous.

Peter Wang (Host) (39:27):

And to be clear, you treat that like memory. That's memory that you can use.

David Liu (Guest) (39:33):

It literally appears as memory. Yeah. I remember making a video on my YouTube channel and when I showed the Windows... You know, pull up the Task Manager, you see how much available memory. People were just gawking at this massive system memory pool. And they're like, "It just shows up as memory?"

David Liu (Guest) (39:50):

And I'm like, "Think about all the applications that you can enable with this." I no longer have to program directly to this API for the persistent memory or the App Direct. I can open whatever I want. I actually did a DaVinci Resolve video for one of my videos and I loaded all of the footage into memory. All of it. So I could just scrub through it.

Peter Wang (Host) (40:09):

That's just amazing.

David Liu (Guest) (40:10):

I was just scrubbing through it.

Peter Wang (Host) (40:12):

See that's such a game-changer and all you got to do is... I guess you flip a bit or something and then it's persistent? Or how does that work? How does the persistency work on that?

David Liu (Guest) (40:21):

The persistency is based upon the BIOS setup. So the BIOS setup just tells you to run that Optane persistent memory in one of three modes. I think it's memory mode, App Direct, and persistent mode, which they'll all have different aspects.

David Liu (Guest) (40:36):

So App Direct is good if you need a front-side cache for databases that are shared on a server and then persistence is like, it'll be persistent after the fact. So certain types of startup of databases... If your database goes down [inaudible] a power outage, but you having backed it up is really, really good. Then a lot of this stuff will still be in memory, so there's different types of things that you can program in to do it. But I think the persistence one is harder because I think it requires a use case in which something loses power pretty often or it needs to shut down often.

David Liu (Guest) (41:10):

Whereas memory mode, you can use it just kind of as is, but the flexibility, I think... It's down to people like me and others who do research at work, to go discover what applications and what use cases you can use these things for. And that's why it's exciting to be at the company.

David Liu (Guest) (41:27):

It's like they put up a new piece of technology every few months and I'm like, "I'm going to go play with it and try this. Let me go ask that manager for a sample and see if I can do something fun with it."

Peter Wang (Host) (41:38):

Well, that's definitely one of the fun things about working at a hardware place. We're coming up on time but one of the reasons I was excited to talk to you is I want to make sure that in the data science world and in ML and everything right now, so much of the conversation is around the software, the open-source software, MLOps frameworks, data processing frameworks, this and that and the other.

Peter Wang (Host) (41:56):

And I feel like there's not anywhere near a sufficient level of discussion about the hardware for daily quality of life stuff. Not exotic high-end GPUs for doing whatever deep learning thing, but just people... So many millions of people, their daily quality life would be better if they paid a little more attention to the actual underlying hardware that runs their data science workflows and their machine learning.

Peter Wang (Host) (42:18):

And to all the listeners, I would encourage them to learn a little bit more about how some of that hardware works.

Peter Wang (Host) (42:26):

And David, you mentioned you have a YouTube channel, right? There's videos you've made (and we'll link those in the show notes) to give people some resources about looking at the workstation you put together. But also if you could suggest some helpful places, the starting points so people can understand, "How do I think about memory or how much cache is on what version of the CPU? What does that really mean for me, if I'm running certain kinds of algorithms?"

Peter Wang (Host) (42:47):

The study that you did looking at different algorithms and what their performance looks like, that is absolutely fascinating. And so I think more people should be aware of that work. Is there, do you have any kind of parting thoughts here as we're kind of coming up on time? Any kind of last remarks that you want the listeners to think about relative to hardware and where we're at today in the state of AI practice?

David Liu (Guest) (43:07):

Yeah, I think the way that I would say the parting thoughts is the world of compute is changing quite quickly. There's a lot of different technologies, completely different architectures, heterogeneous compute going around everywhere. And I think if you're a data scientist or an AI practitioner of any type, it's quite important to at least be up to date on what works and what doesn't work from a very high-level perspective, because you know, are going to be potentially, I won't say quizzed on it, but at least asked by IT about it or asked by your CIO, your business or your management.

David Liu (Guest) (43:41):

And so being a little bit up to date on what type of algorithms and what class of algorithms do well on what type of hardware from a basic standpoint is really important.

David Liu (Guest) (43:51):

And I think the other aspect is as a company, Intel is looking to provide a lot of that information and education over time. And we're also trying to work with other partners like Anaconda to make a lot of this information available as well. So hopefully keep an eye on both companies as we try to make sure that education gets out into that space.

Peter Wang (Host) (44:11):

And there are no stupid questions around this kind of stuff because the hardware stuff is really changing quite rapidly. So as people watch your YouTube videos or whatever, I would encourage them to leave comments and whatnot and questions if anything doesn't make sense to them.

Peter Wang (Host) (44:23):

But yeah, well, David, this was absolutely a blast. I really appreciated you answering some of my questions and sharing your thoughts. And I look forward to working more with you and with Intel or the broader Intel team on trying to improve the quality and the level of practice of data science on modern hardware and get people really understanding what's possible instead of working on paradigms that are 20 years old, working on paradigms that are more modern. So this has been great.

Peter Wang (Host) (44:50):

Thank you so much for joining us today. And I really appreciate you sharing your thoughts.

David Liu (Guest) (44:53):

No problem. Let me know when you want me to come back on again.

Peter Wang (Host) (44:56):

Okay, great. Thank you very much. Take care. Thank you for listening. And we hope you found this episode valuable. If you enjoyed the show, please leave us a five-star review. You can find more information and resources at anaconda.com.

Peter Wang (Host) (45:10):

This episode is brought to you by Anaconda, the world's most popular data science platform. We are committed to increasing data literacy and to providing data science technology for a better world. Anaconda is the best way to get started with, deploy, and secure Python and data science software on-prem or in the cloud. Visit anaconda.com for more information.