Ryan Abernathey (Guest):

And really, this is about bringing AI scientists, people who are doing, innovating on the machine learning methods together with domain scientists and figuring out where's that point where we can plug AI into the climate model and really enhance its predictive capabilities.

Peter Wang (Host):

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. Make sure to subscribe. For more resources, please visit anaconda.com. I'm your host, Peter Wang.

Speaker 3:

This episode is brought to you by Anaconda Notebooks. With nothing to install and nothing to configure, Anaconda Notebooks is a lightweight, ready-to-code, and fully-loaded data science environment entirely in your browser. Spin up new projects with the click of a button, with all the packages and files you need in one place—with fast and persistent cloud storage, no matter what, wherever you go, your code goes. And students? Listen up. You also get on-demand access to Anaconda's data science experts. No matter your experience level, learn through hands-on experimentation and you'll be predicting the future with machine learning models in no time! So what are you waiting for? Start coding with Anaconda entirely in the cloud on anaconda.cloud.

Peter Wang (Host):

All right, welcome. Welcome, and I'm really, really excited to invite Ryan here on *Numerically Speaking*. Really excited to have this conversation. I think we have a lot of really fun things to talk about. Ryan, you've done some really, really exciting work. We could talk about so many different things. But before we do that, would you like to go ahead and give a bit of an introduction about your background and what you work on, what brought you into this whole PyData and cloud and all these different kinds of things?

Ryan Abernathey (Guest):

Really happy to be here. Thanks for the invitation, Peter. In terms of my background, I realize I'm kind of a lifelong hacker and computer nerd. I've kind of embraced this identity. So my father worked for IBM. We had PCs since as long as I can remember, back in the '80s when I was a little kid. Started coding BASIC and got a little bit away from that as I got older. But then rediscovered my sort of passion for computing and technology in college. I got really into the whole sort of streaming media and video and audio thing. This was back in 2002, 2003. So I was really into college radio and set up the streaming system for my college radio station. After that, I worked for a while at this radio TV program called Democracy Now doing sort of video audio production stuff in this sort of indie media world. It feels like a long time ago now.

(03:06):

And then I took a big pivot and I decided to go back to school to do a Ph.D. in climate science at MIT. And then I just became a scientist, and did research studying the oceans, their role in climate. I've been working as a professor at Columbia since 2013 and continuing that research trajectory and working with some great students and postdocs. But during this past nine years the sort of hacker side of me has slowly been reemerging and, in fact, sort of taking back over my life and career. And now I'm at the point where I'm thinking about data and coding and cloud and all of this stuff almost all day. So it's been a fun journey and I feel like it's actually just getting started.

Peter Wang (Host):

I actually didn't know that part about the streaming media thing. Did you pay your annual whatever, 600 bucks to ASCAP or something to be able to stream radio back then? Wasn't there something like that? Because a buddy of mine ran an internet streaming station back then as well. And I just remember this outrageous, well, not outrageous, but there was all this stuff he had to go through like he was a real radio station.

Ryan Abernathey (Guest):

It was kind of the Wild West for college radio. We were paying already licensing fees. That's how all radio stations were doing it. But then we just started doing the streaming without, we didn't ask permission, really. Probably after I left, they had to sort that out and actually do it right. We were mostly, what is all this tech? We were setting up servers under the studio console and learning all about compression and protocols. And it's really interesting how that's actually helped my science a lot, because I felt like I had this really solid foundation in what is digital data, how do we move it around? What is throughput? What is bandwidth? Concepts I kind of took for granted that have been really helpful in computational science. I got my start thinking about audio and video, actually.

Peter Wang (Host):

That's interesting that you started there and you did the computer stuff because you had to, it was a means to an end. Then you went to go and study climate science and then you found yourself doing more computer stuff as a means to an end. And then getting sucked in more and more into that. So Anaconda and some of the open-source folks here at Anaconda have been working with you and with the Pangeo project for a number of years now. And there are a lot of really, really fun things that we've built and we've worked on a lot of the open-source tools that we've built were many of them had features and things that came out of needs that you all had in the climate science arena. It's just been really, really, I think productive collaboration between the scientific needs that y'all have, and then of course also to be clear, you were trying to do a research collaboration in the cloud.

It was cloud-native science almost, right? How do we do that? And of course, there's all sorts of sharp rocks to trip on and fall over in doing all that stuff. Not all of which are technical, some are organizational, and we'll talk about that later. But then definitely helped drive and inspire some of the work that we've done on fsspec and Intake and Dask and all these kinds of things. So I really appreciate your leaning into that collaboration, and people who are using those tools should know they're benefiting from input of your group and from you personally, of course. But as you look out right now at the state of science, just climate science, tell me about what's top of mind for you in that area as you're thinking about that.

Ryan Abernathey (Guest):

Absolutely. Well, let me start with something that's really, I think, exciting and inspirational that's happening right now in climate science. And I'm involved in some of this at the periphery, but the only way we have really solid information about what's going to happen to our future climate is through something we call climate models. And these are some of the sort of main workhorse applications of traditional high-performance computing supercomputers. That's one of…going back to von Neumann. This is why they built—

Peter Wang (Host):

Why we built supercomputers.

Ryan Abernathey (Guest):

Computers to predict the weather and really to project forward the climate. And these are, for the most part, really heavy-duty scientific computing applications, mostly written in Fortran, maintained by these teams at National Labs, places like DOE Labs and GFDL at, Geophysical Fluid Dynamics Lab and National Center Atmospheric Research, beasts of code with a long legacy, very sophisticated at what they do, very in a way, isolated from the rest of computational science, particularly like modern data science. And what we're trying to do now is we're trying to fuse those models and enhance them using stuff from AI. Basically, these are big PDE solvers, and they don't really incorporate data directly. Yet, there is all this data out there about the earth that we can potentially leverage to make the models better.

And so the scientific challenge is how do we actually plug those things together? How do we actually make a 40-year-old legacy Fortran code have a neural network from PyTorch running inside it? And scientifically, what is the right framework to integrate data into that type of application? And this is ultimately not just a technical problem, but really an interdisciplinary science problem. So we have this great new collaboration, couple projects I'm involved in. One called Multiscale Machine Learning for Earth System Modeling, and another called Learning the Earth with Artificial Intelligence and Physics. That's a new center here at Columbia. And really, this is about bringing AI scientists, people who are doing, innovating on machine learning methods together with domain scientists, and figuring out where's that point where we can plug AI into the climate model and really enhance its predictive capabilities.

Peter Wang (Host):

So I don't want to rattle on this part too much right now, but it does seem to me like yes, absolutely, and we talk about AI as if it's this new thing, but when you zoom out and you look at the cybernetics, and I sort of talk about this with Paco Nathan in a previous episode, that the origin of computing was in prediction. The origin of computing was in simulation. We didn't do it to go and do mass telephony. We didn't do it to go and do banking system reconciliation. We didn't do it to make computer games and World of Warcraft. We did it to predict. The whole field started in prediction and trying to make guns hit airplanes or make guns lob shells onto target from the rolling sea. And then that became more and more about simulating things, simulating things.

And then we got these big computers and there's this whole very big iron, serious, let's say, a branch of the family tree of computing that actually stayed in that world a bit. And then it ended up with things like APL and Fortran. And then from Fortran we get the influences on things like MATLAB and NumPy, of course. But the rest of the personal computing world went down a whole different path. And now we're

faced with, at a civilizational level, needing to really, really, really do no-kidding simulation if we want to understand what's going to happen to billions of people. And in doing all of that, we sort of rediscover AI. The AI winter's over, it's back again. But now what we found is not only have people lost, well there's a lack of faith in institutions, there's a lack of trust in science, now we're going to put AI into it to then make even more policy-facing kinds of predictions. And this is going to sound like, well, you're just bringing Skynet in to tell me what to do versus no, this is what we built. It was all about prediction in the first place.

But anyway, not that you're ever going to make that point to the layperson. But I do think it's very interesting. In this area in particular, we have to get quite serious, this is not some DALL-E art thing. This is something very serious about understanding, what is the AI or what are the machine learning models training on? What is in the models, what is not in the models? Are they doing a cubic fit? Or are they doing something better to project what goes forward? I'd love to talk about that in a little later bit. But before we even get to the AI part, let's talk about what is your view right now on how data-intensive science is being done? What is the state of that really in academia? And if you can talk about other disciplines, that's great. If you want to keep it scoped to climate science, that's great, too.

Ryan Abernathey (Guest):

I think this is something that transcends disciplines. And so data-intensive science is the term that I try to use to describe the problems that we are kind of obsessed with trying to solve in our research group and in our open-source work. So in a way, we have this ability to generate data right now in science much faster than we can understand it. So whether that's simulated data, those supercomputers are great at generating data. You can run simulations all day long, dump petabytes of data onto a disc. And there's interesting science in that data, whether it's turbulence or molecular dynamics or drug discovery or whatever. And likewise, with sensors, with satellites, drones, new observing platforms. In the ocean, we use autonomous robots that have sensors. They're just floating around, diving up and down and sending data back to satellites. So we have this deluge of data and so much of the challenge I see is around how do we interpret and make sense of that data?

In many cases, it involves data fusion, combining data from the robots with satellites or combining data from the models with in situ observations. And particularly, there's huge challenges around the volume of data that we have to confront now. And this has just changed in my relatively short scientific career, it's really just exploded and ballooned. And it is very hard to share and collaborate around large scientific datasets once you're measuring that in the petabytes.

We have our HPC centers that have a mandate to provide access to scientists, but they're really fortresses. I mean, they're designed to keep outsiders away from the data. Heavy security, strong restrictions on what you can run there. And what we're really trying to do in science is bring more people into the conversation, not just more collaborators from other Ivy League institutions. We really want to be, in climate science, we want to be involving people from the developing world in that science because they're the ones who are at the most risk from climate change. We need to be more inclusive and the way we're doing infrastructure is not serving that goal. That's why I, and many others, are excited about the cloud. Despite all of the caveats that come with cloud computing, it is a really very much a democratic space where we can all collaborate together, and it has the right infrastructure to truly scale to the size and complexity of the data that we're working with.

Peter Wang (Host):

Yeah, that's really interesting. Many of our podcast listeners are coming from the private sector, from industry, and the general public is probably used to this idea of scientific data being relatively open and public. And I mean, obviously, pictures come off of a space telescope, it's published on all the news sites the next day. So this idea, and if you go to, of course, an insurance company if you work in IT, in an insurance company or a bank or something like that, or if you work for the military in some military data analysis kind thing, it makes sense to you that the data infrastructure and the compute infrastructure is a fortress there. There's personal data, there's national security data. But for science, for supercomputers that the public tax dollars go to pay for to facilitate science, the idea that those are really locked-down environments is somewhat surprising, isn't it? Or, I mean, should we be surprised? Talk to me about that.

Ryan Abernathey (Guest):

So, I think those are expensive resources, but part of the thing that cloud allows you to do, is sort of separate the compute part from the data. Traditionally, the way we share data in science is through the download model. So the data provider will make a FTP server, and you can download the data and that works fine in the small data regime. What those HPC centers offer to scientists is a place where their data and their compute are already together, next to each other. And that's what we want. But it's a sort of monolithic service in the sense that it's all bundled together, and all paid for by the same organization and it's really oriented around the compute. Those computing centers, they exist to maximize their CPU utilization. That's the metric for any HPC center. We were at 99% utilization. We've spun the CPU's day and night. Mission accomplished.

So it's a compute-centric vision. But for data-intensive science, it's really the data that's at the center of the workflow. And that's what cloud is more like. In cloud, really, the compute exists to serve the data and cloud has this key differentiation, which is that you have this sort of multi-tenant architecture. We can have one copy of the data and many people can come and compute on it under their own account, under their own bill and it doesn't have to be downloaded. You take it for granted if you're used to cloud computing. But so much of science in so many fields is still about, okay, go download 20 terabytes of this data, that's step one for the project. Put it on hard drives and then we can start the project. I just think that's incredibly inefficient.

Peter Wang (Host):

The irony is, so let's see if I figure out how to say this concisely. When Travis and I were really trying to think about the next steps, so 10 years ago, when we were thinking about what is the next step. What's different about the Pythonic model of computation, as we came from the SciPy ecosystem, and how is that different than all the stuff that we're seeing in business computing, when we'd go and do our consulting like on Wall Street or in some, whatever, big Fortune 100 company and they've got some backend compute job or some scientific modeling they need to do? And what we found was that when we thought about the problem…him and I are both, I have a physics background, Travis has an electrical engineering background. We both came from the SciPy world and thinking about how can I efficiently take this code and apply it to as much data as possible.

There's very much a SIMD approach, single instruction, multiple dispatch. And what we found, when we started, and I was more in the technical lead architect role in a lot of these conversations…go to Wall Street as you do kind of traditional backend computing, let's say for business applications. There was not

an appreciation for this kind of architecture. There was much more…people would marshal data items, do row-at-a-time stuff. It was completely not a vectorized mindset at all, right?

So we were thinking, you know what, really to do the next generation of problems that are coming, and this is 2009 and '10 that we're having these conversations. We're talking about, for the next generation of compute problems coming, for the next scale of data problems that people have, we cannot do this row-at-a-time stuff. The time you don't do row-at-a-time stuff is when you ship a query off to the database and you'll let that be some database vendor's problem.

But the average Java dev that we dealt with was shamelessly going and marshaling floats and ints, moving them around everywhere, and it's like, you're spending a million cycles to move one int from A to B. And so, we found this like, oh, the best way to talk about this is to use the terms from supercomputing and scientific computing HPC. Move code to data.

And so you'll find if you go back and look at some of our presentations, our initial Blaze vision and what some of the thinking that led to things like Dask and Intake and Numba, it was all about how do you move high-level code, the right code, to massive amounts of data. Because that's a lot easier than shipping data around and shuffling around and marshaling ints everywhere.

So then, the whole thing turns around and inverts, right? Now it turns out that the supercomputing facilities, which is where we would've drawn these lessons from, the architecture of the facility and the economics of that facility, their particular metrics and KPIs, if you're a supercomputer manager/owner, you're absolutely right. It's utilization. You cannot say, I have my computers, here's whatever, a hundred-million-dollar supercomputer, and it was 90% idle last year. That does not fly.

So they're solving for this utilization thing. And the way they architected for multi-tenancy was not really around data centrality, data gravity, across multiple different tenants and stakeholders. Whereas cloud kind of is, it's closer to it than not. So I think this is really interesting, the tension between these things. But ultimately, what you're talking about is that the dynamic in the problem space is there's data gravity. Absolutely there's data gravity. These data sets are so large, we cannot be moving them around between different research groups. And that gravity simply pulls into the cloud now. This pulls up into the cloud.

Ryan Abernathey (Guest):

And this is really strong in climate science, exactly as you say, because on one hand, we're really blessed in the environmental sciences in general. We have great open data values. You mentioned this earlier. People do not generally hoard data or just sit on data. It's all about sharing data. We have very strong policies from the top down, from our scientific agencies. Our data must be made open. And we have this really strong culture around FAIR data: findable, accessible, interoperable, and reusable and this has pervaded the field.

And yet, on the other hand, we don't have an actual platform to enact that. And so there's this big sort of debate going on in science right now. What is our architecture? What is our field-wide architecture we will use to empower people? And there's a bunch of experimentation going on. You've obviously got the

traditional HPC camp, that they're really good at what they do and they want to become the data facilities for the field.

And then you've got people like me who have been experimenting with cloud computing. And I think to some, I'm perceived as this big cloud advocate. They think I'm on Amazon's payroll trying to get everyone to move to the cloud. It's really just about, what does this technology enable? Where I think things are headed, that I'm pretty excited about, is more cloud-like infrastructure but not necessarily owned all by AWS.

So to give an example of this, one project that I'm super excited about is something called Open Storage Network. It is a collaboration funded by the NSF and Schmidt Futures. It's run out of San Diego Supercomputing Center and the National Center for Supercomputing Applications. So it's very much from the HPC world, but what they're doing is they're building object storage. They're placing it on the internet with really high bandwidth configurations. It's S3 compatible, you can hit it from the cloud, you can hit it from the HPC centers. Everybody can compute on it. You don't have to download the data. And it's really enabling this sort of data fabric for really much more flexible workflows. And I'm really excited about that sort of thing and I want to see that grow. It then becomes a bit of a social and political problem of, okay, who should own that? Whose job is it to provide that infrastructure? Is it the university? Is it the agency? Is it each scientist pony up 20 bucks a month? No one has figured out actually the business model.

Peter Wang (Host):

Well, there's so many different laws here. You think about Alan Kay's statement, that people who are serious about software need to make their own hardware. And I would append that with a corollary to say, in this case, since you're serious about the software, you get serious, you're making your own hardware choices. You're saying, I'm not actually wed to this particular thing.

And there's also this kind of thing where, in technology, we can look at things in tiers, here's the strata of the software. This is the software layer, or this is the platform layer, this is the operating system layer. But at the end of the day, those are all just models we layer on top of a pile of silicon, energized to go and basically run a finite-state machine really fast. And so when you look at it from that perspective, isn't the only reason that we would have these separate supercomputing facilities be that we have some different kind of exotic hardware?

For a while, it was like they have exotic hardware. Okay, sure. And then it was like, well, they've got commodity chips. The same chips that you find on the server on cloud, but they have exotic interconnects. So sparse matrix problems or other kinds of things, your NUMA architectures and RDMA kind of things, let you do problems in a way that you cannot do in a generic kind of thing over here.

But then, good friends of mine are the ones from Cycle Computing, who were doing supercomputing applications in the cloud 10 years ago, they pioneered that kind of thing. And they got bought by Microsoft. And they were showing the feasibility, even 10 years ago, of doing these big supercomputing jobs in the cloud.

And so, as we move forward, I think there is, if we zoom out past the budgets, past the organizational political dynamics and all the things, and just ask the question, what is in the best service of science? What is in the best service of reproducibility, agility, compute efficacy and all these kinds of things? And it's an open question as to, why would someone run their own infrastructure? And I'm not saying, I'm not asking that in a rhetorical sense. I'm asking that in a real sense. What are some reasons why one would run their own infrastructure? Actually be racking servers. What's different about the servers you're racking versus the data centers that Amazon or Azure or Google or whoever is racking?

Ryan Abernathey (Guest):

Is that a rhetorical question or do you want to go into that?

Peter Wang (Host):

Well, no, I am actually interested in what you think are reasons that would get you off the cloud, to prove to us you're not on the AWS payroll.

Ryan Abernathey (Guest):

Well, I can tell you about some of the things that a lot of the scientists I talk to worry about when we say let's move our infrastructure to the cloud. A huge one that comes up is egress. People are very worried that, we've put our data in the cloud, we won't be able to get it back. I think a lot of that is a little bit unfounded, but there's definitely the idea that the business model is to build a moat around this data that was ultimately collected through public funding or generated at great effort through the scientists, it doesn't gel. Whether or not it's a valid technical argument, it's a very strong, sort of, psychological barrier for people putting a lot of data into the cloud. So whatever we can do to get rid of egress fees. And there's stuff happening around this, Internet2 and all the cloud providers, have sort of egress waiver agreements, treaties to not charge egress.

So that's one. And then there's a lot of fear about lock-in to the cloud. The idea that we're going to entrust this really precious scientific enterprise to this company whose values might not align with ours. And I think the solution there is to just, I want cloud to be as much of a commodity as possible. I really don't want a lot of specialized services that are very bespoke to each cloud provider. I want compute, storage, database, a few basic things, and I want them to run really well and be cheap. And I want there to be competition in a marketplace. Same way that we do for electricity. It's a utility. So that's how I want cloud to be. I don't think that's how AWS wants cloud to be. They want to differentiate, and add value and stuff, but there's always that tension there. Those are some of the issues that I see.

Peter Wang (Host):

Yeah, so around the egress thing. It occurs to me that you could much more cheaply than keeping a data center up, and maintaining it and all that stuff, you could require the hyperscalers, the cloud hyperscalers, to do something like, like AWS has their Snowball thing. And there's also, they actually have Snowmobile, which is literally a 40-foot shipping container full of hard drives. I think it's a hundred petabytes in there. So you could require that the NSF funds are, there's some kind of escrow thing or something, where you have the right, a few research institutions who were the recipients of the grant, they have the right to require almost like a data escrow, an open data escrow thing. They just need an 18-wheeler full of hard drives to be delivered at the end of the contract or at some period of time to get data updated.

I think there's ways around this. It's not like we don't have the technology. But I do understand those concerns and it's a good concern. You don't want all of science to be captive, or to be contingent on a few centralized providers. So that all makes good sense. I guess for me, I always try to boil things down to a moral point, which is the number of brains that can think about the science, and the scientific computing code, and the infrastructure, and the all, whatever, and they commit their lives to doing the hard work of, and sometimes very thankless work of, research science. Those few precious souls, we need them to be doing non-commodity things. We need them to be doing cutting-edge things, not sitting there doing stuff that is absolutely commoditized by the hyperscalers. Just like where you don't have them soldering chips or trying to design their own fricking, like, floating point units. That just doesn't make any sense anymore. I think this level of infrastructure, I think that the commoditization point has moved beyond this at this point.

Ryan Abernathey (Guest):

Totally. I really dig what you're saying. And this is also what I like…to get back to the PyData ecosystem. I think this is, just in terms of now just focusing on software, I think PyData has started to really penetrate into the mainstream science. It's probably now the dominant stack in my field. But MATLAB is probably a close second. But still, I think what we see is a lot of people are using the stack at a very basic level. They're using NumPy and Matplotlib, which at the level of functionality is not really that much different than what you're getting in basically any scientific computing language. MATLAB, IDL, R, you got your arrays of data, and you can plot it. And then the rest is up to you.

I'm really excited about our stack is the layers, the different functionality that can be built that doesn't then have to be rewritten and repeated and recreated by grad students over and over. I see this as the story of the Pangeo project. So just for a little history on that, I'm a core developer on Xarray. And back in I think 2016, we were having this discussion on the Xarray mailing list. Hey, Xarray is really great for climate science. Wouldn't it be great if we could coordinate our efforts and try to raise awareness about it and also add features, add functionality, and sort of grow this community so that everyone else that we know in this field can have the same superpowers that we feel like we have when we're using Xarray. And so that's what led to the Pangeo project, which differs from something like Astropy, where they were writing actual software under the name Astropy. Pangeo was all about integrating that software and to some degree marketing it to both our funding agencies and our colleagues as a solution to some domain-specific problems.

A typical specific problem that it would address is, okay, NASA is distributing data about ocean temperature. And there's one file per day. And there's like 20,000 files, going back to the beginning of the record. We want to query that as one object. So when we bring to bear and we want to do something like calculate statistics, what are the trends? Are there more marine heat waves shown in this data? Are habitats for fish and other marine life going to change? The combination of say, Xarray, Dask, HoloViews, this whole higher-level stack. And then you can bring in Zarr if you want to really optimize your storage. Just makes those calculations turn into one-liners. Things that before we were writing hundreds of lines of code looping through the files, writing the aggregations, figuring out bespoke parallelization strategies. They become like one-liners. And that's what we've been really trying to sell. And I think that's where there's still a lot of work to do to educate users and different communities about how powerful higher layers of the PyData stack can be.

Peter Wang (Host):

So that's really interesting, because when you think about scientists, right? Scientists are, they're busy, they have lots of things. They want to work on their research, they want to work on their science. But when it comes to the software, some scientists take it seriously, most see it as a means to an end, which it kind of is. And yet, you could say the same about math. But you'd be insane to think of a scientist actually arguing that, "Oh, well I'm okay with arithmetic. I don't really need algebra, or maybe I know algebra, but I don't need trigonometry and certainly not calculus. I'll have someone else do that. Or when I really need it, I'll go read the books or something." And that seems insane. But what you're saying is, I think, somewhat in line with this. That there is a computational skill literacy something, almost like any other skill they would use, but not quite, maybe it's not quite like the other skills.

Like making good slides. They have to learn the tech, they have to go and write their papers in it. And so, this idea of putting a little bit more effort in, to help your own brain being augmented by the computer, the brain-computer interface really is what the programming language is. Optimize that interface, go from four-bit to eight-bit, maybe to 16-bit interface on that bad boy. Stop beating your head into for loops and NumPy.load NPZ files, and do something a little better. And I think some of it is marketing. Your point about Pangeo being somewhat of a marketing effort, somewhat of education, moving the whole community forward, it's very sort of forward-looking of you to realize that, yes, open, free developer tools actually need to have marketing. Who knew? But it's a thing.

Ryan Abernathey (Guest):

And one thing we did crack with that effort, that I really want people to replicate, is we figured out how to get the NSF to pay for sustainable, truly valuable open-source software development. And we did that by bundling the software development together with scientific use cases and then partnering with Anaconda at the start of that project where we were outsourcing a lot of the development work, not all of it, but there was a real partnership. If you want to think of it as, from a traditional product point of view, we as the scientists were sort of acting as, to some degree, the users and the product owners who had certain requirements. And we were iterating quite quickly with people. Matt Rocklin was very involved early on and Dask and Jim Bednar and the HoloViews tools. This iteration was pretty productive and instead of creating new software, we augmented what was already there.

And that's a pattern that in general, NSF funding hadn't been able to unlock. And I think we found a way to do it and I want people to just repeat it over and over. So if anyone's listening, you're a scientist and you want to do this, write a proposal to your agency that focuses on science outcomes, that has a big chunk that says, okay, here's software dev we need to do to get those outcomes and then partner with one of these great open-source shops, whether Anaconda or Quansight, QuantStack, makepath. I mean, there's a lot of great companies out there that can do really high-quality dev work fast, much faster than you can hire a postdoc to do it. And just iterate. And you do that for three years and you can really move the ecosystem forward. And I think it's a good model.

Peter Wang (Host):

Well, I definitely appreciate the kind words. And I think that the collaboration part of it is actually one of these things where it's one plus one equals three. I appreciate we're talking about science and I just gave you invalid arithmetic. But it really is that partnership of the scientists actually in that collaboration. The scientists don't have to choose to be these hybrids, of like, am I a software developer now trying to do

maintainable, good software architecture, or am I a scientist actually thinking about the science of what I'm doing? In these kind of partnerships, the folks, again, whether it's Anaconda, whether it's one of these other companies, we have experience, we have an eye towards thinking about what is sustainable, good architecture. Or we may even push back and say, this is not actually an appropriate thing to try to build as its own standalone project.

You probably should have this as either just a part of what you're doing for this particular research project. And so we can give that kind of feedback as well on things like this. And I think that's the kind of thing that I absolutely agree with you would be…we love to be part of those kinds of things and we do some of that kind of stuff in our services work for big enterprise companies. We're helping them to build and tweak internal dashboarding kinds of things, or whatever kinds of stuff, doing predictive and data scientific sort of work. So it's definitely an area that we want to help in. And science, the climate science stuff is, I guess all science is impactful. There's medical research, there's all these kinds of amazing things happening right now. But climate science is the one that I think so many people understand the importance of now. It's just hard to say, after the last several years, I can't imagine anyone has got their head in the sand about the fact that it's real.

Things are changing, and we have to be able to predict what's going to happen. Otherwise, we'll be caught very, very unaware when hundreds of millions, and we're talking about, these are sovereignty-ending kinds of catastrophes that could happen, and impact hundreds of millions of lives. And so in this regard, I mean, is there anything…in the climate science arena, I'm glad to hear there's so much open collaboration in the ethos and all that. But are there aspects that you think are set to really change our understanding in a dramatic way? Are there things that are going through an inflection point, or are there big things on the horizon, that if we could do the right kinds of scientific computing, infrastructure, modeling, all that stuff, that we could dramatically advance and improve how we're approaching climate science?

Ryan Abernathey (Guest):

So I think one trend that we have to definitely pay attention to in climate science is the emergence of a real private sector. That's something that's super interesting. Because when I got into oceanography starting in 2006, I mean, I thought of it like, think of astronomy. Astronomy is a beautiful, rich, fascinating subject. It doesn't have a lot of direct bearing on our economy. But it's a rich, great scientific problem. And that was really where I was coming from in oceanography. I just love to look at the data, try and understand what's going on, turbulence and all that stuff. But in that period, the realities of climate change have really emerged from the noise. Now in climate we have the internal variability, which you can think of as the noise, and then there's a forced signal, the response to our greenhouse gas emissions. And that forced signal is getting more and more clear and it's only going to accelerate going forward.

And so we're seeing society respond, and we're seeing the economy respond and we are seeing business respond. And unfortunately, our government has been pretty much inactive on this. The big climate bill, the inflation reduction bill, whatever they had to call it to get it passed, that's going to make a huge difference. But even before that bill passed, business has been mobilizing, because they know the impacts are going to be felt on their bottom line. And so we have this whole climate tech sector with hundreds of billions of dollars being invested. And this is a really interesting time for climate science and climate data because all those companies need data. They're working with the same data sets. And so

we have this scientific infrastructure that's been oriented towards just academic research that now also has to serve those business needs. And that's creating a lot of opportunity. It's creating a lot of tension. It's creating a lot of interesting times for the field.

Peter Wang (Host):

That is an interesting thing because I feel like when companies are themselves using this data to make predictions about logistics or insurance and reinsurance and things like that, you can see a lot of that stuff where people are, there's still sharing of this data commons. But then at some point, I wonder if there's something a little bit darker that could creep in, where data withholding and things like that are part of the model advantage. Or data quality leading to model quality becomes a part of the competitive advantage. Not just de-risking but actual competitive advantage for some of these enterprises and whatnot. It's a little bit of a moral hazard because if you know a particular area is more likely to go underwater and you don't tell them, you're just going to use it to make a little bit more money on your premiums or how much you model certain mortgage portfolios. This seems like there's a moral hazard there, isn't there?

Ryan Abernathey (Guest):

Walk me through that a little bit more.

Peter Wang (Host):

Well, I'm just thinking—

Ryan Abernathey (Guest):

The moral hazard?

Peter Wang (Host):

The moral hazard is, if you can do accurate modeling of something where you know it's going to have a human impact, it seems like there's some kind of obligation for you to tell people about that. To say, you know what? Don't use the NOAA model, don't use the NASA model. We have a much better one which can predict, let's say, the storm track, and we can give people 12 more hours of warning. Don't you have a moral obligation to then make that model open?

Ryan Abernathey (Guest):

And you're referring to the case when a company would develop that model independently—

Peter Wang (Host):

Yes, I'm talking about the privatization of the—

Ryan Abernathey (Guest):

Of the research enterprise.

Peter Wang (Host):

Yes, yes, exactly.

Ryan Abernathey (Guest):

That's really interesting. Okay, I totally get what you're saying and absolutely it's true. And then that idea of IP and proprietary knowledge comes into direct conflict with the open data ethos of the academic research enterprise.

Peter Wang (Host):

Exactly.

Ryan Abernathey (Guest):

The truth of the matter is, right now, and this could change, but the private sector is not really capable of making climate projections for the simple reason that…remember those HPC centers we were talking about trying to model 20 minutes ago? They don't have those, right? And that's actually where all of the climate projections originate from. And they can talk all they want about using AI, and this and that. But even the same with weather prediction, no one really is running a real weather model outside of government labs, the way the National Weather Service does or the European Centre. So, I mean, we could get there, but we're not there today. Most of what these companies are doing is taking that data and enriching it somehow, or fusing it with other data.

In terms of the moral hazard, yeah, I mean, I think it's real. I think it was problematic if companies are using some specialized knowledge about the climate to say, help Goldman improve their portfolio, but not help children in Bangladesh avoid catastrophic heat waves or something. But the fact is, I believe interests are aligned in almost all cases. So in terms of adaptation to climate change, we've seen what happens to people when supply chains are disrupted. Empty shelves, no diapers. That's a case where everyday people's interests and business interests are highly aligned to keep those supply chains functioning. That's a, supply chains are a big place where climate change is going to be a very direct impact.

Peter Wang (Host):

No, of course. This is an interesting area and I think about it, now we're going to go two levels up on the philosophy and the economics and the politics of this. You bring up Goldman, which is, of course, they are very active and they're a very, very renowned firm in the space of financial services and investment banking and prediction of financial things. And one could argue, of course, that when traders are out there, this is not really about Goldman, it's just about the way that the financial sector works. When traders are sitting there and they're shorting this or they're arbitraging that, they're making predictions and they're, the way they allocate capital affects people's livelihoods and people's lives. So there is a human impact to what they do, no doubt about it.

But that being said, the data they're trading on, ultimately it's a big casino. They're still making a bet, they're taking a position, there's some arbitrage available and various kinds of things. But at the end of the day, they're having to take a position on what the future may look like, and it's through a cloud of, there could be geopolitical risks. There's all sorts of things. Who knows what people are buying or not buying, or watching or not watching?

But when it comes to something like the weather, we now have a prediction environment and a prediction problem that is just as materially impactful as, sorry I said weather, I should say the climate. When it comes to climate, right? That's a data-centric prediction problem that is just as impactful as anything is trading signals off of Wall Street. But to me, it seems like there's a much clearer connection between the value of this as a commons, good modeling, good prediction as a commons, as opposed to it being a scarce resource for people to go and try to arbitrage each other on.

And this is the thing that we're going to have to figure out because I think it's not just climate. So many other things as they come online, as predictions become more and more an online part of how businesses are making decisions about logistics, about customers—

Ryan Abernathey (Guest):

Okay, Peter, let me interrupt you for a minute. I think I see where you're going about this potential divide that could emerge between the companies working on this and the rest of the world. And I really do think it's a problem that can be solved through effective cooperation around both the data and software commons that all of the stakeholders in this problem have. We want to incentivize companies to find solutions to climate change, but we don't want to do that by having them build a wall around common data or around tools that can benefit everyone. And everyone I talk to who's operating in this space feels that way. It's not that they're in it to get super rich. They're trying to use the market to drive innovation and find solutions. The academic climate research enterprise needs to meet them halfway by thinking about things like data infrastructure.

If we make our data systems so inaccessible and obscure, then there's going to emerge a whole sector of this industry that's just going to repackage and resell the data to people.

Peter Wang (Host):

That's true.

Ryan Abernathey (Guest):

And you can already see this happening in the crypto space. You can go read about all this climate data, you can get in on Web3. They say the NASA data is so obscure and weird, it's not usable by business. Join our coin and we'll give you clean climate data to integrate into your apps. Let's not do that. Let's not do that. Let's, as a research enterprise, make our data system so great that it's easy to use for scientists to make discoveries. It's easy to use for businesses, to leverage that data that was collected through taxpayers and let's make it easy to give back.

Right now, the data exchange is pretty one way. Generally, NASA and NOAA are collecting a lot of data and generating data and others are consuming it. But I'm really inspired by companies like Planet. They launched their own fleet of satellites. And they've been a really good citizen, contributing a lot of open data back to the community. A lot of great software and infrastructure work has come out of Planet. So if there are going to be more companies like that operating in this space, I am very optimistic about the future. But we, on the research and government side, have to meet them halfway and modernize our infrastructure.

Peter Wang (Host):

I think you're absolutely right. There's something about, and we sort of notice this a little bit in the software side on the open-source stuff too. We produce a lot of this great scientific software that's fit for purpose. It's made by scientists to solve their own problems. But if it's hard to use, no one adopts it, they will go to a cloud vendor that, or I won't say cloud vendor, but they'll go to any vendor that cheesily repackages it and says, well, here's an easier thing.

So I do think actually in this way with Anaconda, making some of these that are easier to install was a way to prevent that from happening. And I think with data, with models, the same thing has to happen. The whole community of maintainers, people who believe in the necessity of this being open and being an open commons, we have to really respect what the end user's needs are, and go an extra mile to make it easy for them to participate in the way that we want them to show up in this kind of ecosystem. Otherwise, others will step in and then direct them in whatever direction. So that's a really great comment. I appreciate you're giving me some hope in that regard.

Ryan Abernathey (Guest):

Can I tell you about our Pangeo Forge project, which is—

Peter Wang (Host):

Please, absolutely.

Ryan Abernathey (Guest):

Is trying to help solve this?

Okay, so probably most of your listeners know about conda-forge, right? So conda-forge, they really democratize the sort of production of conda packages by creating a sort of cloud-based environment where you could provide a recipe. Before conda-forge, you could build your own conda package, can compile everything up and put it on the website. But conda-forge made it even easier to contribute to the library of conda packages. That really opened things up and I think it's really cool.

Now, Pangeo Forge is trying to do something similar with analysis-ready data in the cloud. So a lot of the data engineering work in this space means taking data in some archival format from some data provider. Let's just pick NOAA. NOAA's great, I'm not trying to pick on them, but they basically have a FTP server full of a bunch of files in some obscure format like GRIB, that you can download.

And what we want is essentially we want that in a database or a data lake, where they're accessible to compute, ready for analysis. Where we don't have to do a bunch of data engineering legwork just to get to our first plot.

So that kind of work happens every day in scientific labs. That's happening every day in these companies we're talking about in the climate tech space. And it's really tedious, it's full of toil and it's ultimately repeatable. Many people are doing the same job over and over. And so the goal of Pangeo Forge is to

create a sort of data commons where we can build this analysis-ready data in the cloud from all kinds of different sources, in a crowdsource, collaborative way.

So we directly copy the sort of pattern from conda-forge. You have a feedstock that describes the sort of transformation pipeline. Where am I going to ingest the data from? How am I going to transform it along the way? What sort of format am I writing it to? And then, the backend of Pangeo Forge just executes those, interfacing with Git, triggered on commits to the repos, and it stages the data. And we're populating this library of open-access data that's accessible to everyone, can be used by researchers, can be used by businesses. It's not like one guy who has to manage all the pipelines. It's a community framework where we can do this collaboratively.

This is a project I'm super excited about. It's really hard. It's very ambitious and it's way beyond our capability as just our team to actually realize the potential. So whenever I get a platform, I like to sort of plug it a little bit, and encourage people to get involved. Because we could use way more hands on deck for this project. At the same time, it's up to date, it's functional and it's really cool what it can do.

Peter Wang (Host):

That sounds great. I mean, look, data munging is always thankless work. But at least you're doing it once, making sure no one has to do it again. And furthermore, it seems to me like the data transformation thing you're doing here, and the way that you handle data, scientific data formats, the fact that it's running kind of in this cloud way, but all the infrastructure is open, that's reproducible. You can clone stamp that for astronomy, you can clone stamp that for so many different kinds of things, right? I'm happy you plugged it. I'm glad you plugged it. And I encourage our listeners to go and check that out. And also check out Xarray, which you mentioned you are now, went from being a user to being a maintainer/developer of, right?

Ryan Abernathey (Guest):

Absolutely. Yeah. I love Xarray and I'll hype it up whenever I get a chance, right? So basically, what Xarray is, for people who don't know it. The way we think about the PyData stack is usually kind of you have NumPy at the base. It's like the foundation of everything. And then pandas is this abstraction that sits on top of NumPy in many ways. It uses NumPy under the hood, and it provides things like indexes and group by and a bunch of cool features. pandas is really oriented around the tabular data model. So in pandas really, you've just got your various rows and you got your columns. Whereas NumPy actually has, you can have n dimensions to any of your arrays, right? So what pandas ultimately does under the hood, is it just manages a bunch of one-dimensional NumPy arrays, Xarray aims to provide that higher-level API around multi-dimensional data, right?

So what we get in Xarray is say we've got a data cube, temperature on earth. Rather than just thinking about axis zero, axis one, and axis two, we can think about time axis, latitude axis, longitude axis. We can have an index on each of those axes. So query it, not by position integer offset within a nameless array, but by…okay, give me 2001, at this latitude and this longitude. We can query the data. And we can get things like group by, rolling reductions, all of the awesome convenience features that really help analysts write better analysis code, write faster analysis code.

And then Xarray also wraps many, many different array-like things, not just NumPy arrays. So Dask was one of our first integrations, so Xarray can hold a Dask array. And bringing those two together, you have this sort of amazing scale-out framework for array computing that…I know of no other thing that does that. There's a lot of frameworks that'll allow you to do large-scale analytics on tables. I mean, there's probably, from traditional databases to all your Sparks and Presto and Trino, and everything does tables. So pandas and Dask dataframe are just one of two dozen different solutions to that. I know of no other framework that allows you to do this scale-out analytics on arrays the way Xarray and Dask do. It's really a superpower.

Peter Wang (Host):

Yeah, it's not just the multiple dimensionality you have, but also the fact that it is labeled, that you're able to do, really use meaningful semantic indexing on this stuff. I do understand that all the cool kids call multidimensional array tensors now. So if you were to call it, like, hypertensor or something, maybe you get a little bit more of an adoption uptick and a few more stars on GitHub. But...

Ryan Abernathey (Guest):

As a computational physicist, I just cannot get behind the tensor thing. A tensor is a different thing. Tensors are about symmetries and properties under transformations, and arrays are just big multidimensional buckets of numbers. And that's what these things all are.

Peter Wang (Host):

You could call it distributed buckets of numbers, too. That would also be kind of hilarious. I see that trending on Hacker News, buckets of numbers.

Ryan Abernathey (Guest):

Our ecosystem has made a lot of advances in terms of array API. So now we've got sparse arrays, we've got GPU arrays, CuPy. We've got Pint unit-aware arrays. So in Xarray now, you can do analytics on Xarray, wrapping Dask, which is then pointing at CuPy arrays and compute on a GPU cluster. Or you can have unit-aware arrays in there.

And the next step that we're working on is integration with the machine learning tensor libraries. So PyTorch tensors, JAX arrays, so that you can hold all of those things within Xarray, compute on them using xarray's API, which people love. And then get things like automatic differentiation, all of the machine learning and other features that those array libraries bring. So we really think of Xarray as kind of this ecosystem glue that brings all these other things together, and it's kind of the top layer of the stack that so many of us use.

Peter Wang (Host):

That's fantastic. And I'm so glad to see that come around. And if we had more time, I'd love to delve a little bit more into, a kind of, these global array concepts, and maybe talk about Chapel or some of these other things. But we're somewhat out of time, but I've just tremendously enjoyed this conversation with you, Ryan, and want to thank you for bringing so much energy into it, and all the great things that you've done for scientific computing and for science. And I encourage our listeners to Xarray to check out Pangeo Forge and the Pangeo project in general. And I look forward to future conversations. Thank you so much for joining us today on the podcast.

Ryan Abernathey (Guest):

Thanks, Peter. It's been so much fun.

Peter Wang (Host):

Thank you. And as a reminder to the listeners, of course, in the show notes, you can find links to all these projects, so be sure to check out our show notes.

Thank you for listening, and we hope you found this episode valuable. If you enjoy the show, please leave us a five-star review. You can find more information and resources at anaconda.com. 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.