

DATABASE RENAISSANCE MAN

DataStax's softly spoken co-founder **Jonathan Ellis** has successfully made the transition from the cubicle to the corner office. **Mayank Sharma** grills him on databases.

The Cassandra project doesn't have a Benevolent Dictator For Life, but if it did that title would surely go to Jonathan Ellis. The co-founder of DataStax, Jonathan has been involved with Cassandra since the time it was open-sourced by Facebook. Once the project graduated from the incubator at the Apache Software Foundation (ASF), he served as its first Project Chair for the next six years. It was difficult to tear Jonathan from his fans at DataStax's Accelerate conference in Washington, but Mayank Sharma lured him by disguising himself as one as well.

Linux Format: You've always been a database guy... How did you find your way to Apache Cassandra?

Jonathan Ellis: It's true that I've always been interested in database technology, but I initially thought that the database space was just making incremental improvements on well-understood solutions until I joined a cloud backup company called Mozy in 2005. I built an object store there that scaled to petabytes of data and gigabits per second of throughput, and one of its features was single-instance storage. That is, no matter how many users uploaded the same video or the same binary, we'd only store one copy in the backup storage.

This in turn meant that we needed a way to track which users had copies of which files – scaling to millions of users and billions of files. That's when I realised that we needed new database architectures to deal with the challenges of web and mobile applications. Existing databases were optimised for applications that dealt with a single company's worth of users, but now we needed to scale to an entire country. It was a very different problem that required different trade-offs.

Just a few months after Facebook open-sourced the Cassandra project, Rackspace hired me to work on the challenge of scalable databases. I got to dig deep into Cassandra and the alternatives that were starting to grow in this space, and I was really attracted to its marriage of a rich, tabular data model with a fully distributed, masterless approach to scalability and fault tolerance. As a result, I started working on the code and on building the community, and when Cassandra graduated from the ASF incubator, I became the first project chair. Getting involved with Cassandra is one of the best decisions I have ever made.

LXF: I heard you once broke Facebook... is that true?



JE: Actually, I'm probably one of the only people who doesn't work for Facebook that has broken Facebook. It happened after I was made committer on the Apache project; I think the feature [which was changed] was adding support for deleting rows, which didn't exist when Facebook open-sourced it – that's how raw it was.

So I committed that on, I think, a Friday afternoon, and over the weekend I'm getting frantic instant messages from Avinash [Lakshman, co-author of Cassandra] at Facebook: "What happened? We deployed the latest version and everything broke." That's how we learned how you don't deploy from trunk, and also that CI/CD is an important part of your development chain.

LXF: When did you realise you could spin a profitable company around the database itself?

JE: A venture capitalist named John Vrionis tracked me down in the spring of 2009, just a couple of months after Cassandra joined the ASF. He was looking for early-stage projects in the big data space, and we had a good talk about NoSQL databases in general and Cassandra in particular. In retrospect, I wish I had been less timid, but I told him that I thought it was too early to start a company around Cassandra. What gave me the extra push was when an early Cassandra user went with a different, worse technology because the alternative had commercial support available. So

ADOPTING NEW TECHNOLOGY IS HARD

"Facebook likes to say 'Move fast and break things'. But if you go down to the data lair, there are some crusty greybeards there saying 'No, if it ain't broke, don't fix it.'"

LXF: And you probably stopped being so cavalier about pushing changes.

JE: Well, almost literally they were the only people using Cassandra at the time, and I didn't realise that they were running from trunk. It was probably about another six months or so before we even had non-Facebook people using it.

LXF: Twitter?

JE: Twitter would have been, and the timeline's fuzzy, but Twitter was one of the early ones. Digg also. Actually Digg was before Twitter, and then Twitter hired a bunch of the Digg engineers and that's how Twitter started using it. So they were early users.

about a year after first talking to John, I started DataStax, and John and Lightspeed Venture Partners decided to lead our series A investment round.

LXF: I talked to Robin Schumacher about the kind of symbiotic relationship between DataStax and the open source Cassandra community and the positives of such a relationship. But this relationship has also created some issues right? With Twitter, for instance, back in 2010?

JE: There are going to be times when the needs of the community and a business are the same, and there will also be times where what the community sees as its

goals are not aligned with what the business wants to achieve. One of those times was when I stepped back from leading the Apache Cassandra Committee (PMC) in 2016 to give members of the community more scope to lead on what they wanted to create.

However, we are keen to stress how committed we are to the community. At DataStax, we're increasing our support for Cassandra, continuing to invest in improved drivers and contributing support to get Cassandra 4.0 production-ready through testing and bug fixes. Alongside the code and drivers we contribute, we want to make it easier for more developers to get started with Cassandra, so that means maintaining our documentation and hosting more events to expose people to what Cassandra can achieve and what's new in the latest version.

We believe this approach lets the community set the goals for where they want Cassandra to go in the future, while we help the community achieve those goals for more people. I think of this not as being the hand on the steering wheel but the fuel in the tank: we are not directing or leading the community where we think things should go, but helping the community get to its desired destination.

LXF: In that respect, one of the themes of the conference is DataStax re-engaging with the Cassandra community. But

what does that actually mean in quantifiable terms?

JE: Well, I would say the most quantifiable piece is that we have a distribution of Apache Cassandra now, and so we've always contributed bug fixes from DataStax Enterprise back to Apache Cassandra. But it's been on kind of a "We'll do that when it's convenient" kind of basis, but our incentives are a lot more aligned, now that we actually have a product that supported open source, to contribute those back in a more immediate fashion. So internally we have been setting up the processes to make that happen and make sure we're not lagging behind in either direction.

LXF: In your keynote you mentioned that eradicating complexities when rolling out a Cassandra cluster was one of the main motivations behind the announcements at the conference. But there's more to operating a Cassandra cluster. What's the next problem that you want to solve?

JE: We could talk about Kubernetes, Kafka integration, or DataStax's new Graph release, but the theme that ties these together is making DataStax and Cassandra easier to operate and easier to develop against. When I talk to our customers, almost none of them complain that we're not powerful enough or fast enough or anything along those lines. Where we sometimes struggle is making

that power available, understandable and consumable. The next frontier is really about making things easier, and all of those things fall into that category.

LXF: There was a time when people said, and you've acknowledged this, that if it's not SQL it's not a database. How has that perception changed?

JE: Are you quoting one of my talks at me?

LXF: I have absolutely no idea. I have it in my notes here. It could have been you or somebody else at DataStax.

JE: I gave a talk a few months ago titled "Five lessons in distributed databases". My first point was "If it's not SQL, it's not a database". So when we had the NoSQL 10 years ago, it was a good name in the sense that it got attention, right? Like, if we called it "scalable databases", that's too... it's not attention-grabbing. "NoSQL" is attention-grabbing. But the problem with it is that it brought in people who just hated SQL versus people who needed to solve problems a relational database couldn't solve. I was firmly in that second camp. I don't hate SQL, I just don't like the 30 years of architecture baggage that you have to buy into to use SQL, circa 2010.

Today we've been able to add SQL and SQL-like query languages on top of databases that made different trade-offs, and do scale, and SQL does a fine job of trying to solve the problem it's trying to

» THE MIDDLE MANAGEMENT

During our conversations we were joined by Mani Srinivasan, Sr. Product Director at DataStax, in charge of the DataStax Insights performance management tool that was unveiled at the conference.

LXF: You know what NoSQL and Cassandra are good at. What's the one thing it isn't good at?

Mani Srinivasan: Jonathan alluded to it this morning: ease of use and configuration.

LXF: But that problem you've solved. What's something that you haven't?

MS: Well, what we'd like to do is make the lives of the operators a lot simpler than what it is today. So we throw a bunch of metrics, we throw a bunch of things and let people interpret whatever they would want to do. What we'd like to do is to solve that problem altogether by removing the operational layer completely and solving the operational

complexities of managing Cassandra. Because even managing Cassandra is pretty difficult, as a matter of fact. You can throw a bunch of dashboards, you can throw a bunch of metrics, but after that... Especially for customers that come from a relational background, it is very difficult to transition to this kind of a world. What DataStax is trying to do is to solve this problem. I would say that's one thing.

JE: If you wanted to look at it from a different axis, I'd say that Cassandra is not a data warehouse... It's not going to suck all your data in from all parts of the company and run your analysis against that. We are focused on being the operational data layer. And that does include some analytical features but that's intended to enable the operational parts, rather than being a pure-play analytics offering.

LXF: In that regard, do you tell people it's a database or a database

management system?

JE: I've always told people that it's a database. I think we are heading closer to the direction of the database management system with the stuff that we announced today. But the distinction, where you cross the line from one to the other, is a bit fuzzy.



solve. Early on people were willing to use Cassandra in 2011 without SQL because there wasn't any other way to get that functionality. But now that you can get that, the bar's raising. People are demanding SQL, they are demanding ease of use. Whereas before it was enough just to solve the hard problem, now you have to solve that and it needs to be easy.

LXF: You've been kind of firefighting since forever. People used to say that you were in the right place at the right time, that you had clients from the get-go. But you've also had detractors. I remember in early discussions on Slashdot people were quick to point out that Twitter doesn't use Cassandra to save tweets. And then Facebook tried its best to say it doesn't use Cassandra.

JE: There's kind of a nerd-politics, right? Like when you have an architect at a company that wants to do things one way, there's like political capital there. It's not "What's the right technical decision", there are egos on the line. But I think the bigger factor with Facebook or Twitter is that both of these companies have some seriously extensive tools that they've built and has to manage tens of thousands of MySQL servers.

So it's a solid engineering principle. Facebook likes to say "Move fast and break things". But if you go down to the data lair, there are some crusty greybeards there saying "No, if it ain't broke, don't fix it. You stay the f*** away from this." The engineer in me respects that; I am not looking to go to someone and say, take what you have and throw it away and build on Cassandra because it's new and it's shinier or it's cooler. If I'm not solving a problem for you then I'm fine with that: use whatever you want, use what works for you.

I want to go to people for whom MySQL replication is a real pain point and solve their problems, because I can do that easier than a MySQL consulting shop. If you look at some of Facebook's subsidiaries, Instagram is a big Cassandra user. I believe it's their main datastore at Instagram. Netflix is another example where they went from Oracle on premises, and they said "We're going to move to the cloud and we're going to adopt a better database technology as we are doing that", so they went with Cassandra. The database tent is big enough for lots of families to live in.

LXF: You highlighted the low latency numbers of the database. What about Scylla? They've gone to town saying they can beat your low latency numbers.



JE: Scylla has some public benchmarks and we've run those against DataStax Enterprise, and we do beat them with their own benchmarks – but we are not looking to get into a mud-slinging benchmarking battle there.

LXF: Some Sylla marketing people stood at the shoe shine and were giving out their fluffy soft toys after asking if any of the people passing were here for the database conference. I have never seen that – it's a big enough tent, like you said. But they've been doing that since 2015, right?

JE: Yeah, they actually launched their product at our Cassandra Summit in 2015. They submitted a talk "How we accelerated Cassandra", and we said "Oh, that sounds interesting, come and speak". And they said, "Oh, we actually rewrote Cassandra".

LXF: You've been doing conferences based around Cassandra for almost a decade now, since the Cassandra Summit in 2010. Who is your audience this year?

JE: To quote former Microsoft CEO Steve Ballmer, for us it's about "developers, developers, developers." The community that exists around Cassandra is great, and we want to bring that awareness to a much bigger audience.

We are excited about supporting ApacheCon this year, about the new features we're contributing to the Cassandra drivers, and about what will happen when Cassandra 4.0 launches. We launched our inaugural DataStax Accelerate conference this year to bring the community together and evangelise about what Cassandra can do, and we'll be expanding our range of developer events so that more people can share their experiences and learn more.

You mentioned our Database as a Service in passing earlier. We're very excited at how dramatically easy this makes it to run Cassandra. Cassandra is unquestionably a great database, but it is fair to say that you needed experience in order to make the most out of it. In response, we've automated the hard parts as part of our Constellation cloud platform that's launching this year, and we're confident a lot of people will find this resource valuable.

The growth of the cloud over the past few years has been massive, but people don't want to be stuck with one single provider. They want to take advantage of their investments in their own infrastructure, as well as the strengths of the cloud, and get the best of both worlds. Cassandra is unique in being able to run across multiple cloud services, or across internal IT and external services, as one seamless database.

As more people think about the cloud and how to make it work for them, Cassandra will play a big role in running those services at scale.

LXF: Do you think it'll be every year?

JE: I think so, yeah. I think the community needs the event, and DataStax needs the event, so there's a good alignment there to make that happen on a yearly basis.

LXF: You've resolved the issues you had with the community?

JE: Ah, I mean the issues were never really between the PMC (Project Management Committee) and DataStax, it was more the Apache Board of Directors. The short version is that we are on good terms with the PMC, and we'll leave it at that.

LXF: You're like the Linus Torvalds of databases.

JE: I'll take that as a compliment... **LXF**