SUPER DATASCIENCE
MAKING THE COMPLEX SIMPLE

SDS PODCAST

EPISODE 885: PYTHON POLARS: THE DEFINITIVE GUIDE, WITH JEROEN JANSSENS AND THIJS NIEUWDORP

Jon Krohn: 00:00:00 This is episode number 885 with Jeroen Janssens and Thijs Nieuwdorp, authors of Python Polars: The Definitive Guide. Today's episode is brought to you by Trainium2, the latest AI chip from AWS, by Adverity, the conversational analytics platform and by the Dell AI Factory with NVIDIA.

00:00:25 Welcome to the SuperDataScience Podcast, the most listened to podcast in the data science industry. Each week, we bring you fun and inspiring people and ideas, exploring the cutting edge of machine learning, AI, and related technologies that are transforming our world for the better. I'm your host, Jon Krohn. Thanks for joining me today. And now, let's make the complex simple.

00:00:59 Welcome back to the SuperDataScience Podcast. Unusually, for the second week in a row, it's got to be the first time that's ever happened two weeks in a row, we've got two guests in today's episode. Jeroen Janssens is our first guest. He is senior developer relations engineer at Posit. Previously, he was senior machine learning engineer at Xomnia, the largest Dutch data and AI consulting company. He wrote the invaluable O'Reilly book, Data Science at the Command Line and holds a PhD in machine learning from Tilburg University.

00:01:31 Our second guest today is Thijs Nieuwdorp, who leads data science at Xomnia, and he holds a degree in AI from Radboud University. My apologies. And to any of our Dutch listeners, I am surely butchering every single Dutch name that I try to say. So the reason why Jeroen and Thijs are on the show today is because they are the authors of Python Polars: The Definitive Guide, which was published by O'Reilly just a couple of weeks ago.

00:02:02 Regular listeners will know that I often hold book raffles when I have guests on the show that have written a popular book, particularly if they wrote it recently. And typically, I administer those book raffles personally. Well, this week, Jeroen and Thijs have upped the ante. Not only can you receive a free physical copy of their book, Python Polars, they are kindly taking the admin out of my hands so that they can both sign and ship your physical copy to you wherever you are in the world. That's free and signed and shipped to you.

00:02:34 So Jeroen and Thijs are giving away three free copies of these signed Python Polars books. Head to polarsguide.com/sds before the end of the day this Sunday, May 11th. That's polarsguide.com/sds to well to be in the raffle to get a free signed copy of Python pulls. We've got, of course, a link for you in the show notes so that you can do that easily.

00:03:02 Today's episode will be particularly appealing to hands-on data science, machine learning, and AI practitioners. But Jeroen and Thijs are tremendous storytellers and, frankly, very funny. So this episode can probably be enjoyed by anyone interested in data and AI.

00:03:17 In today's episode, Jeroen and Thijs detail why Pandas users are rapidly switching to Polars for dataframe operations in Python. The inside story of how O'Reilly rejected four book proposals on Polars before accepting the fifth, the moment when an innocuous GitHub poll request forced a complete rewrite of an entire book chapter and a previously secret collaboration with NVIDIA and Dell that revealed remarkable GPU acceleration benchmarks by Polars.

00:03:43 All right. You ready for this laugh-filled episode? Let's go.

00:03:54 Jeroen and Thijs, welcome to the SuperDataScience Podcast. You guys are together. Have I ever had this situation before? I can't think off the top of my head of having two guests, but that they are co-located. Where are you two co-located?

Jeroen Janssens: 00:04:12 Well, thanks, Jon. It's great to be here again. Good to see you again. And we're calling in from Rotterdam, the Netherlands.

Jon Krohn: 00:04:19 Nice. And that voice for the listeners out there, for the people not watching YouTube version, that was Jeroen. That's his voice. And for people watching the YouTube version, he's the one in the pink shirt. Oh, also his mouth was just moving, and sound was coming out of his face. But whatever's easier for you to track. And then in a charming and matching or complimentary forest green shirt we have Thijs Nieuwdorp. Thijs, what does your voice sound like?

Thijs Nieuwdorp: 00:04:46 Thanks so much for having me, Jon. This is what my voice sounds like.

Jon Krohn: 00:04:49 Oh, nice. It would be helpful if one of you didn't have a Dutch accent, but fine. We'll have to just work with us.

Jeroen Janssens: 00:04:58 We have accents? What do you mean we have accents?

Jon Krohn: 00:05:10 That's funny. So Jeroen, you've been on the show before. You were on an episode 531 back in December 2021. That was the very end of my first year hosting this show. We had a great time on that podcast. It's a great episode that people can listen to. But Thijs, am I correct in understanding this is your first podcast ever?

Thijs Nieuwdorp: 00:05:27 It is, yes. I've never podcasted before. It's just since the book is taking off, we're finally getting into that marketing, and we're kicking off with the best. So I have no clue what to do after this.

Jon Krohn: 00:05:39 I hate to let you know, it's going terribly so far. This is a bad episode. It's too bad.

Thijs Nieuwdorp: 00:05:45 Yeah. Now, well, we'll just start with my biggest fear and just break it down from there. Right?

Jon Krohn: 00:05:48 Your biggest fear. Yeah. Exactly. We'll start with your biggest fear, and then we're going to move on to your biggest beer. If you could grab one of those, I think it might smooth things along.

Thijs Nieuwdorp: 00:05:57 Nice. Nice.

Jon Krohn: 00:05:58 All right. So you guys, until recently until February 2025, you were co-workers together at Xomnia, which is the leading data and AI consulting in the Netherlands and makers of open source dataframe library, Polars. And then, Jeroen, you recently took a dev rail job at Posit. It seems like a lot of people are moving over to Posit. A lot of big names are there now, makers of our studio and lots of other great open source tools. We've had episodes on Posit before in the past. We don't need to get into that too much.

00:06:31 But what I'd like to speak about most in this episode is your new book, brand new released, actually the same day that we are recording this, which is April 1st. So now that this episode is live in May, hopefully, it'll actually be available again because right now, in a lot of locations around the world, at least if you try to buy Python Polars: The Definitive Guide by our guests today, Jeroen an Thijs, you wouldn't be able to get it on April 1st because it is sold out.

00:07:00 But O'Reilly are very good. They do on-demand printing. And so they should be able to resolve that pretty quickly. It's not like with a lot of other publishers, it could mean months potentially before a huge, I don't know, like a lot of publishing companies do orders in the 5,000. But I'm pretty sure O'Reilly, they can do printing on-demand, which is cool. So anyway, that should be resolved soon. Very popular book. Very excited to have you on the show.

00:07:26 We've had Polars on the podcast before. We've had Ritchie Vink, its creator. We've had another key contributor to the Polars project, Marco Gorelli. But the Polars library has grown a lot since then. It's about to pass Pandas in popularity if we measure that in kind of number of GitHub stars, if that's a measure of popularity. And, yeah. And now, it has this great O'Reilly book, thanks to the two of you. So what spurred you guys on to write the book? What was the experience like? Oh, and I've got to tell the listeners about this, that you're doing a book giveaway.

00:08:04 So I think we'll give them until Sunday. What do you think? Up until Sunday to?

Jeroen Janssens: 00:08:09 Sounds good to me.

Thijs Nieuwdorp: 00:08:10 Yep.

Jon Krohn: 00:08:10 Sweet. So there's a URL. You can say what the URL is and what the free book giveaway is. We do free book giveaways on the show, lots physical books, but there's something special about your book giveaway that we've never done before. So I'll let you guys fill the audience in.

Jeroen Janssens: 00:08:26 Yeah. For your listeners, we wanted to give away hard copies that are signed by the both of us. So in order to be eligible for a copy, you go to polarsguide.com/sds, and you fill in your name and email address. And then you enter the raffle. And then by Sunday, we'll let you know. Even if you don't win, you'll still get the first chapter for free.

Jon Krohn: 00:08:50 Awesome. That's such a cool giveaway. I'll encourage more guests to do that. And so both of you'll sign it. I guess if you're co-located, it makes it easier.

Jeroen Janssens: 00:09:00 Did I say three copies? I'm not sure if I said the number.

Jon Krohn: 00:09:02 I don't think you did. But, yeah, three copies,

Jeroen Janssens: 00:09:04 We'll give away three copies.

Jon Krohn: 00:09:05 Nice. And people can be anywhere in the world?

Jeroen Janssens: 00:09:08 Anywhere. We will take care of the shipping.

Jon Krohn: 00:09:10 Yeah. Sweet. Yeah. Super generous. Thank you very much for doing that. So yeah, polarsguide.com/sds. You have until Sunday to submit yourself into the raffle and get assigned copy of Python Polars: The Definitive Guide from both of its authors, our guest today, Jeroen and Thijs. Super, super cool. All right. So yeah. So now with that out of the way, tell us what caused you to write this book and what the process was like?

Thijs Nieuwdorp: 00:09:37 Yeah. It started with you, right?

Jeroen Janssens: 00:09:39 Exactly. So let's start with the origin story here. I joined Xomnia in January 2022. Is that right? Yeah? No. Oh, man.

Jon Krohn: 00:09:54 It sounds like he's asking somebody, but that's Jeroen asking himself. It sounds like he's having a conversation-

Jeroen Janssens: 00:09:59 I'm asking myself.

Jon Krohn: 00:10:00 ... maybe with Thijs and me.

Jeroen Janssens: 00:10:02 It doesn't really matter when. It doesn't really matter. So when I started, I was just getting to know everybody working in the office, and there was this one guy really focused working behind his laptop. Everybody was going to lunch, but he would just stay working.

00:10:19 Turned out that was Ritchie Vink, the creator of Polars, I learned later. And I didn't know anything about Polars, but I didn't have an assignment yet. I had some time to work to explore a dataset. And I decided, "Let's try out Polars." And I was immediately hooked, of course. And I immediately figured, "Okay. This is so cool. This deserves a book. This is going to be a big thing." But I already knew having written Data Science Command Line before-

Thijs Nieuwdorp: 00:10:49 Twice.

Jeroen Janssens: 00:10:50 ... twice, yeah, that I never wanted to write a book by myself anymore. So I needed-

Jon Krohn: 00:10:55 Is that the tragic story of how you wrote it once and then accidentally burned it and-

Thijs Nieuwdorp: 00:10:59 The dog ate his homework.

Jeroen Janssens: 00:11:03 I wish. No. So I needed another victim, someone to share the pain with. And Thijs, so very shortly after that, I got assigned at a client with a large code base. And Thijs was also working in that same team. So we were not only colleagues. We're also working for the same client in the same team.

00:11:30 And so I felt like, "Hey, Thijs seems to be good at this. He likes to write. Why don't I ask him?" And to which his answer was-

Thijs Nieuwdorp: 00:11:40 Obviously yes.

Jeroen Janssens: 00:11:41 Yeah. And so I had a meeting with O'Reilly anyway about whether I could do anything else for them, maybe a video course or something related to Data Science Command Line. But that's when I asked him like, "Hey, have you heard of this thing called Polars?" He say, "Yeah. Yeah. We've had four proposals so far, but we've all rejected them." And I was like, "Oh, wow. Four proposals already."

00:12:06 And so that's when I knew that we had to write a serious proposal. So we wrote one over 15 pages. We brought in all the stats that we could. And, of course, by then, O'Reilly hesitantly said yes. But after a few months they realized like, "Oh, wait a minute. This is actually going to be a big thing. We want to have this book now."

Thijs Nieuwdorp: 00:12:31 Starts feeling the pressure, started asking, "Okay. About that deadline, is everything going all right?"

Jeroen Janssens: 00:12:37 Yeah.

Jon Krohn: 00:12:37 That's really scary because writing a book, it is torture. When I wrote Deep Learning Illustrated, it was the worst experience of my life. The only thing that came close was writing a PhD dissertation. But with a PhD dissertation, there's not that much pressure because two people are going to read it. You're going to have your PhD exam, the board. They're the only people that are going to read it.

00:12:58 And then also there's this amusing thing of a number of girlfriends that I've had since doing the PhD in this early stage of dating when they're really excited about having met me. That goes away quickly. But there's this very brief window-

Jeroen Janssens: 00:13:15 [inaudible].

Jon Krohn: 00:13:15 No, but they see it on the shelf, and they're like, "I'm going to read that." And I'm like, "You're not. It's really not readable." It's designed for a really specific niche of individual in the world that you have to spend many years for this to make any sense.

00:13:39 But writing a book like Deep Learning Illustrated, the idea was hopefully more than two people would read it. And I don't know the whole time, I don't know if you feel differently about this, Jeroen, having now written several books before. So maybe you feel like, "You know what? I can write a bestseller. I know the process, I know what to do."

00:13:55 But at least for me, I've only released that one book so far. And for me, the whole time I was writing it, I was filled with this deep concern that it would come out, and everyone would realize that I was a fraud, that I had no idea what I was talking about. I don't know if you've had anything like that.

Jeroen Janssens: 00:14:13 Yeah, I recognize that, especially with the first edition of Data Science Command Line, which I wrote right after I finished my PhD thesis. I was in this groove, but I really felt like an imposter during that entire time, especially since everybody and their dogs seems to have an opinion about Linux and Unix and which tools to use. And so a lot of opinionated people there, which made it all the worse. But by the second edition, I realized like, "Hey, you said bestseller. Well, I'm not sure that our book is going to be a bestseller. Pretty sure, it's not."

Jon Krohn: 00:14:55 The Polars book? It might not be a New York Times bestseller, but I bet in some Amazon categories, it will be, surely.

Jeroen Janssens: 00:15:04 Maybe. Yeah. Yeah. It's funny how Amazon assigns these categories. Number one-

Jon Krohn: 00:15:09 Yeah, you could be number one graph database-

Jeroen Janssens: 00:15:11 ... in database design.

Jon Krohn: 00:15:11 Exactly. Graph database execution for children.

Thijs Nieuwdorp: 00:15:17 Exactly. Oh, wow.

Jeroen Janssens: 00:15:19 But what I have learned is that I and Thijs, I knew that Thijs, we can definitely write a book. You don't have to know everything. That's what a lot of people think, is that you have to be an expert in the topic. That's not true. Maybe, you think that you're an expert. But as you start writing, you'll realize that you have a lot of gaps in your knowledge. And that's when you start learning more and more about the topic.

00:15:47 By then, when we started writing Python Polars, I was pretty confident that as long as we would stay one step ahead, and what definitely helped is that we were able to implement the things that we learned at our client. And we can talk more about this later, how we actually put this into production, but you'll figure things out along the way.

Jon Krohn: 00:16:09 You put the book into production?

Jeroen Janssens: 00:16:11 Not the book in production. It's one of the first companies that has actually Polars code running in production. So it has that now for over a year before the 1.0 release of Polars. So, yeah. So quite confident. And I guess the biggest takeaway here is that you don't have to know everything when you start writing a book. You'll figure things out along the way.

Thijs Nieuwdorp: 00:16:41 Yeah. It turns out that the imposter syndrome, it's a natural part of the writing process.

Jon Krohn: 00:16:45 Curious about Trainium2, the latest AI chip purpose-built by AWS for large-scale training and inference? Each Trainium2 instance packs a punch with 20.8 petaflops of compute power, but here's where things get really exciting: the new Trainium2 UltraServers combine 64 chips to deliver a massive 83 petaflops in a single node.

These Trainium2 instances deliver 30-40% better price performance relative to GPU alternatives. Major players in AI like Anthropic and Databricks, along with innovative startups like Poolside, have teamed up with AWS to power their next-gen AI projects on Trainium2. Want to see what Trainium2 can do for your AI workloads? Check out the links in the show notes. All right, now back to our show.

00:17:35 Nice. And Thijs, what is it like working with a tyrant like Jeroen?

Thijs Nieuwdorp: 00:17:41 You want me to leave the room, blink twice?

Jon Krohn: 00:17:46 Yeah. Exactly.

Thijs Nieuwdorp: 00:17:49 In my opinion, it went very naturally. I think quite early on, we already noticed that we have a relatively complimentary writing style that I just start putting words on paper and start restructuring and moving it around and refine it more.

00:18:05 Something that stems from the time I was still writing a thesis where I couldn't get anything on paper because I was so judging everything you put down like, "Nah, that's not quite it." And you get stuck in that. So I learned to just get stuff out on paper, and it may not be proper and the right format and the right semantics, not exactly the nuance you want to catch.

00:18:25 But ultimately, it gets you to where you want to be. It's just like the first 80% needs to come first, and that's not perfect yet. And Jeroen, one of his qualities is that he can use his perfectionism in such a way that he's very good at the refining phase. So when I put some meat in the chapters already, he comes and moves stuff around like, "Have you thought about this? Or shouldn't you word it like this?" And that really does the eyes. So in that sense, not necessarily a dire end. It's just a very effective perfectionist.

Jeroen Janssens: 00:18:57 Thank you.

Jon Krohn: 00:18:58 It's a fine line.

Jeroen Janssens: 00:19:00 Yeah. There is a fine line, and I am very well aware that there is such a thing as preparing too much as overthinking things. And it really helps when there is already something on the page. So for example, that could be text written by Thijs or by myself. What I sometimes do as a trick, whenever I feel I'm stuck because this is a book that involves a lot of code, I'll first write all the code cells, all the code chunks so that I can then fill in the gaps with text along the way. That's one of a couple of tricks that I could apply here.

Jon Krohn: 00:19:37 Yeah. Very nice. How long do you think it'll be before book writing is? We will just be completely... My next question is kind of a joke, but it's just such an annoying question, and I'm even regretting that it's going to come out of my mouth. I'm going to do it because now I've started down this road, but I was going to have that classic thing. It's like when I'm out for drinks with friends that are, say, not data scientists or AI engineers, software developers, but maybe they listen to the podcast.

00:20:11 And so for example, I have a friend who's like, "Oh, you're really lucky that you got your book, Deep Learning Illustrated out before the ChatGPT era so that people know you really wrote it." And then so I was going to have this question, which is kind of trite, and you don't have to spend much time answering this. We'll get into some Polars topics next. But do you think there will be a time in the foreseeable future where O'Reilly just asks a machine for a proposal, it creates a 15-page proposal, and then it says, "This is it." And then it goes and writes.

Jeroen Janssens: 00:20:48 If you want to keep on regurgitating existing knowledge, then, yeah, then using a stochastic parrot is great. But if you want to produce actual new knowledge, then I believe that humans are very much indispensable here.

Jon Krohn: 00:21:03 Nice. Good answer. That was nice. Really a much, much better answer than I was anticipating, especially given the quality of the guests. But anyway.

Jeroen Janssens: 00:21:12 April Fools.

Jon Krohn: 00:21:15 Yeah. We are recording on April Fool's Day, and I guess it's still the morning in Hawaii or something at the time that we're recording. So nice. Let's talk about Polars. I do think your book is going to be a bestseller because there is a Polars moment right now, as I talked about at the outset of this episode with the popularity, at least in terms of GitHub stars, probably going to surpass the number of stars that Pandas has this year.

00:21:42 Ritchie Vink and Marco Gorelli's episodes of this podcast last year were very popular in terms of both listens as well as social media reactions. And by the way, if people are interested, because we've been careful with this episode with the topics that we've curated, we won't be overlapping with Polars topics from Ritchie or Marco's episodes.

00:21:59 So if you want even more Polars after this episode, you can check out 827 with Ritchie or 815 with Marco or both. They're outstanding episodes. Both are highly technical people just like Jeroen and Thijs are. These are all complimentary episodes covering different aspects of the library. But, yeah, very popular episodes, very popular social media reaction. I would not be surprised at all if your book did. If it's sold like hotcakes, everyone loves hotcakes.

00:22:30 So let's talk about the grammar. So like R's tidyverse, which they make at Posit where you now work, Jeroen, with Polars, there's also a grammar and a naming convention that is encouraged to preserve semantic clarity, which means that not only can you understand your code better when you come back to it later, but other people that you're working with can understand it more easily as well.

00:22:56 In the book, you two compare expressions in Polars to recipes. So specifically, I'm going to read a little snippet of your book here. If you think of an expression as a recipe, then the operations would be the steps, and the functions and methods would be the cooks. So how does this metaphor shape your philosophy about best practices with data transformation design in order to deliver clean readable pipelines, especially in large collaborative projects?

Jeroen Janssens: 00:23:25 Big chunk. Short answer is no more brackets. When you read Pandas code, there are many brackets in there. And in a lot of cases, it's very difficult to reason about what the code is actually doing. And so with Polars, you take a different approach, not only with those expressions, which are indeed the building blocks, those small recipes, but also the part where you use the expression namely in the entire query.

00:23:58 So it's almost like you're writing a paragraph to come back to book writing. You're writing a paragraph of things that you want to do, so logical element in your entire pipeline. And that's much easier to reason about.

Thijs Nieuwdorp: 00:24:22 Yeah. And I think one of the things I've always liked most about Polars is the very declarative approach of what you're writing down. So in Pandas, it can be the case that you're very focused on specific operations in parts of your dataframe, which can make it hard to follow what exactly is going on under the hood. But with Polars, you declare what you want as end result. It would just leave the specific processing and optimization to the engine. And it makes it way easier to read.

Jeroen Janssens: 00:24:49 Maybe, this is a good moment also to clarify that we are very appreciative of Pandas, right? We're not here to bash Pandas at all. That's what you sometimes see online, is these comparisons that are not done in a very-

Thijs Nieuwdorp: 00:25:07 Elegant manner,

Jeroen Janssens: 00:25:09 Not in an elegant manner. That's not us. Without Pandas, there wouldn't be Polars. So we are very much appreciative of Wes McKinney and his team have done.

Thijs Nieuwdorp: 00:25:23 Absolutely. The standing on the shoulders of giants, right?

Jon Krohn: 00:25:27 So Wes the creator of the Pandas library, which for people who aren't already practicing data scientists, you may not be aware, the Pandas has been, for some time now, for a decade at least, the de facto standard for working with dataframes which are a kind of data that's like how you could imagine data R in a spreadsheet, so in an Excel kind of tool where you can have columns that represent different kinds of data. So you're not restricted to having just a matrix of numbers of float values, for example.

00:25:58 With a dataframe, you could have similar to this idea of column names, and then you could have one column that's string information, like company names. Another one that's number information how much revenue those companies had in a year. And so for a decade or more, Wes McKinney's Pandas library-

Thijs Nieuwdorp: 00:26:17 15 years. Yeah.

Jon Krohn: 00:26:19 15 years. Wes McKinney's Pandas library had been the standard for working with dataframes in Python. And they're hugely important because you're constantly... As a data scientist or data analyst, you're constantly working with different types of data like that. And so working with them in Pandas has been key, but Polars has taken off again, like hotcakes. It's burst onto the scene recently. And Ritchie has led development of this, Ritchie Vink.

00:26:49 So I realize that it's not nice to bash Pandas, but why are so many people switching over to Polars today? What's the nuanced argument that even maybe Wes McKinney himself would forward?

Thijs Nieuwdorp: 00:27:05 So I think when we were talking with Ritchie over the many times we talked with him over the course of writing the book, one of the main experiences that shaped how he wants Polars to work was some frustrations that he had when running his pipeline. And only 20 minutes in you run into some trouble, and it crashes. And that's not something you could have seen upfront.

00:27:30 So this is one of the experiences that helped him shape what Polars ultimately became. And there's also a lot of good things that he saw in how Pandas works that he wanted to take and put in Polars. But generally, Pandas became a big inspiration, both good and bad for Polars and also other libraries. I think like Spark especially, you can see that the syntax of the Spark is a lot like how Polars turned out. And there's other elements, for example, from the Rust language that Ritchie took to implement in Polars because it just made it work so nicely.

Jon Krohn: 00:28:10 Yeah. He talks a lot about Rust in his episode. Cool. All right. So that gives us a bit of a foundation around your book and around Polars and why people are using it so much for dataframes operations today and more and more and more. So earlier in the episode, you mentioned about a real-world implementation of Polars and maybe as you said, maybe the first ever production instance of Polars. And so am I right in understanding that's Alliander? I'm probably butchering the pronunciation of that.

Thijs Nieuwdorp: 00:28:37 Yeah. Alliander, it's a power grid provider in the Netherlands. Also, they provide the infrastructure for both electricity and gas in a third to half of the Netherlands, I believe.

Jon Krohn: 00:28:48 Yeah. So the largest utility company in the Netherlands therefore. I can't even say Netherlands. That's how bad I am at Dutch pronunciation. Netherlands, that's actually easier, isn't it that way, isn't it?

Jeroen Janssens: 00:29:02 For us, it is.

Thijs Nieuwdorp: 00:29:03 Oh, that's what you're talking about. I was wondering what is this country?

Jon Krohn: 00:29:07 Where are these Netherlands?

Jeroen Janssens: 00:29:08 That ain't no country I haven't heard of.

Jon Krohn: 00:29:10 Yeah. So tell us about that project and what it was like. And actually, it'd be interesting to know was there overlap in working on the book and working on that project and did working on a Polars book help with a real-world implementation? Anyway, that's an interesting side question.

Thijs Nieuwdorp: 00:29:27 Yeah.

Jeroen Janssens: 00:29:28 Yeah. So the origin story here is that Thijs and I, we were both very excited about Polars. We were writing a book about it. And then all of a sudden, it became clear that at Alliander we needed to speed up the pipeline, we need to lower cost, we needed to process much more data. And in the current state, that just wasn't possible.

00:29:51 It was a combination of not only Python and Pandas, but also R Code. So it was very inefficient. To give you an idea, we were running this on a single AWS instance that had over 700 gigs of RAM, 700 gigs of RAM. And so yeah, we can provide you a link with more backstory to this with some actual numbers, but we were very excited, and we were like, "Hey, let's try this out. Let's do this."

00:30:20 At first, the team was very hesitant where there are two people or three actually, we had another colleague, three people promoting Polars that is being developed at Xomnia. So they were very skeptic, understandably.

00:30:36 So what we did in order to convince them is to just take on a very small piece of code, some low-hanging and benchmark it and re-implement the Pandas code into Polars and then just show the numbers. And by then, they were immediately convinced, "All right, this is indeed way faster, uses way less memory. Let's try this out. Let's take on this huge code base piece by piece, by translating not one-to-one, because you can't do that. You really have to reason about the inputs and the outputs and then do it in an idiomatic way."

00:31:15 You just translate Pandas to Polars. And I think it took us, well, what, six months, a year? I don't even remember. But eventually, I left that client at that time. But there was a moment like, "Okay. We can now get rid of R and Pandas as a dependency of this project." And it's been running smooth ever since.

Thijs Nieuwdorp: 00:31:40 Yeah, definitely. Yeah. I think ultimately, the size of jobs at the beginning was about 500 gigabytes for just that task of doing one calculation, and we shrunk it down both being a consequence of implementing Polars, but also, as we were going rehashing some of the code structure that we were using in the project, we hashed it all the way down from 500 to 40 gigabytes, which makes it a lot more doable to-

Jon Krohn: 00:32:08 Wow, 10X.

Thijs Nieuwdorp: 00:32:08 ... my calculations.

Jeroen Janssens: 00:32:11 And so the second part of your question was, okay, how did this influence each other, the book writing and putting it into production? And yeah, it was a perfect match because when you actually need to put it into production, when you have a real problem to solve, that's also when you start to notice the limits or maybe inconsistencies or missing functionality.

00:32:41 For example, there was this random sampling with weights. That's something that you can do in Pandas. You just give it another column that indicates the weights for the sampling. That's something maybe even up until this point, something that Polars doesn't have. Luckily, that was for an ad hoc analysis that we had to do. But at that point, it becomes clear what Polars can and cannot do.

00:33:11 Also, when you write, you start to look at things from a little bit of a higher level. So sometimes, we noticed inconsistencies in naming or missing methods like, "Hey, why is there no inline operator for the XOR operation?" That's something that nobody ever thinks about. But when you need to put in a table in your book and you need to fill in all the pieces, that's when you start noticing these kind of things. So we were able to also submit some issues, maybe even a few pull requests to Polars itself along the way.

Jon Krohn: 00:33:51 This episode is sponsored by Adverity, an integrated data platform for connecting, managing, and using your data at scale. Imagine being able to ask your data a question, just like you would a colleague, and getting an answer instantly. No more digging through dashboards, waiting on reports, or dealing with complex BI tools. Just the insights you need - right when you need them. With Adverity's AI-powered Data Conversations, marketers will finally talk to their data in plain English. Get instant answers, make smarter decisions, collaborate more easily—and cut reporting time in half. What questions will you ask? To learn more, check out the show notes or visit www.adverity.com.

00:34:35 Very cool. So you're actually influencing the library itself as you're writing the Polars book, as you're working on consulting projects, bringing Polars into the real world, getting huge benefits in terms of memory footprint, that 10X figure that you gave there, 500 gigs of memory down to 40, that's massive. Definitely makes it a lot easier to be working with the data. And no pressure if you don't, but it was nice to kind of get that 10X for memory, that 10X improvement. Do you happen to know what it was for compute time? Was it about 10X as well kind of thing?

Thijs Nieuwdorp: 00:35:09 I think, ultimately, the compute time, because along the way, one of the things why we had to optimize the code was because the requirements for the amount of samples that we were running for a certain simulation were supposed to hit 50 samples. It was like what the stakeholders asked us to strive for. And the 500 gigabyte instances was already 25 samples. So we couldn't push it higher because it just stacked higher and higher. And that's at the end, ultimately, we were able to do those 50 samples in the same timeframe that it took to do the 25 samples at the beginning.

Jon Krohn: 00:35:43 Cool. Very nice. I want to move on to a slightly different topic from your book, but it's related to this idea that you just mentioned around improving code bases, improving the Polars library and the flexibility, the full breadth of capabilities that it has.

00:36:00 In your book, you introduce a way to style tables with a package called great underscore tables, Great Tables. But, yeah, if you're typing it out, the package is great underscore tables. And in a talk, Jeroen, you actually mention the tables are underappreciated in visualization. So could you elaborate on why this Great Tables package was created, what it does, maybe what its advantages to existing approaches out there?

Jeroen Janssens: 00:36:30 In hindsight, so now that this package exists, Great Tables, it's strange that there wasn't already a package because tables are everywhere, especially when people are working with Excel. A lot of people really like to add styling to this in order to make it presentable to stakeholders, add some color, use currencies, what have you, maybe some mini-graphs in there.

00:37:04 So now that it's there, it's so obvious that there should be a package for this. So Rich, I'm actually not sure how to pronounce his last name. The creators of Great Tables Rich Iannone, I'm butchering that, but I do know the co-creator, but they're both my colleagues. I should know, but I just call him Rich and Michael Chow, great folks. You should have them on the show as well.

00:37:34 They created the Great Tables package. And so just only a few days ago, I saw a post by someone about Polars advocating or actually recommending that, okay, it's useful to add in the dollar sign when you're presenting currency, but what he was doing, he was actually changing the underlying data like, wait a minute, that's not the way to do it.

00:38:01 You want to change how it's represented, this layer on top of it. That's what you need to do, and that's what Great

Tables can provide. So you're not changing those floats or integers to strings in order to format it. That's not the way to do it. No. So there should be another layer, and Python has a myriad of data visualization packages. But when it comes to producing tables, well, I only know of one, and that's Great Tables. So with Polars, you can indeed style dataframes using the Great Tables package created by Rich Iannone and Michael Chow. So you use the DF style accessor, and that will then use the Great Tables package under the hood.

Jon Krohn: 00:38:47 There you go. I'll try to explain back to you the example you just gave me there with the dollar signs, and you can tell me if I'm getting this right, that basically you're saying if you have this huge... It doesn't matter if you have a very small table, if you think about a spreadsheet with a hundred rows, it doesn't really matter if you write some kind of find replace that goes and adds in dollar signs at the beginning of every number in a column.

00:39:12 But if you have a gigantic piece of data, then trying to edit each of the individual items in that gigantic column would be very computationally and memory expensive. And so with Great Tables, you have this abstraction above where you don't need to individually change that information in all the rows. It's like an attribute of the column that changes.

Jeroen Janssens: 00:39:38 Yeah. I wasn't really hinting at the performance issues right there. It doesn't feel right to me that you're changing the actual data. You want to keep the data, the data because you never know what you want to do after that. Maybe, you want to have a subsequent calculation going SA.

Thijs Nieuwdorp: 00:39:54 You have to strip the dollar sign again?

Jeroen Janssens: 00:39:56 Yeah. Also, when you want to have round numbers or, yeah... So there's so many, many instances where you just want to change the representation and keep the underlying data intact.

Jon Krohn: 00:40:11 Nice. Okay. Yeah. Great explanation. Cool. Listeners should definitely check that out, the Python Polars Definitive Guide, or should say it properly, Python Polars: The Definitive Guide.

Jeroen Janssens: 00:40:27 The Definitive Guide.

Jon Krohn: 00:40:30 And yeah, I don't know, Riley, now you can get it anywhere in the world. And if you're listening to this the week that this episode has come out, you can also head to polarsguide.com/sds to go into the raffle to get a signed copy by both Jeroen and Thijs which is awesome.

00:40:45 I'd like to move now on a little bit to topics beyond your book. So we have been talking in this episode obviously about Polars a lot, which is a popular Python package. But another Python package that is really taking off recently is UV. And so it's a Python package and project manager that climbed from zero GitHub stars to... I actually don't have the number in front of me right now, but a very large number. Lots of people are talking about UV, and it has exceeded poetry as another longtime favorite for package management. So Thijs, in a blog post, you mentioned ditching poetry for UV. You talk about increased speed, reliability and ease of use as the reasons for that. Do you want to tell us more about UV, poetry and whether people should be calling it UV?

Thijs Nieuwdorp: 00:41:45 [inaudible] Yeah. This is also one of the things that we decided to do for the book. We started out with poetry and did all the version management of Python versions with pyenv and other tools around it. But ultimately, when we were prepping the repo that can be used by the readers of the book that contains all the notebooks that come with the chapters, so you can follow the chapters along and execute the code yourself, play around with it, obviously, you need to set up an environment easily that can work on many different systems that all your readers might have.

00:42:19 So in the beginning, we were thinking maybe to go for Docker because that generally is the easiest way to make something run on different kinds of configs. But as we were writing the book, UV became bigger. And at one point, I just started experimenting a little bit with UV to see how easy it's to set up. And it boiled down to installing UV and then running UV sync, and it sets up everything. It sets up the right Python version. It just finds the right dependencies for your system. Everything just clicks. So that's ultimately what we went for is the final solution for that repo to allow people to just install UV and just make it work.

00:42:57 And one of the reasons I started playing around with UV was mostly because it goes with the trend of the Rust-based tooling, which shows that very much like the performance of tooling is a feature in itself. It's one of the things that Polars showcases, and it clicked very well. UV has the same kind of thing going for it. Also, the Rust-based tooling which leaks faster. It's going to be more than 10 times faster. That combined with the single command setup has made it a very quick, an easy win.

Jeroen Janssens: 00:43:31 Maybe, you can say a few things about the regression that you found in Polars.

Thijs Nieuwdorp: 00:43:35 Yeah. At some point, UV is so fast that you can, on a fly, set up an environment like an ephemeral environment that's just set up for just that command and then torn down again.

00:43:46 And with that, I was playing around with that to benchmark the different versions of Polars to see what the speed is on different queries, different kinds of setups, and iterating over the versions and just bumping it every time to see what happened. At one point, I found, I think, in version 1.2 point something that there was suddenly a regression that the query started taking 10% longer to run the full benchmark. And it didn't really go down again.

00:44:10 And drilling down, we were able to pinpoint the two specific queries of the benchmark that we were running just spiked up on a certain version. And because UV just sets it up so quickly at one point with a script for git-bisect, which allows you to pinpoint the exact commit version in the Polars repo where it started occurring, allowed us to find which specific commit caused this regression.

00:44:35 And funny enough, when I communicated it to the Polars guys in that week, they hit the same code. And for some reason, they couldn't quite figure out quickly what exactly caused it. But they hit the same code and refactored it, and it resolved itself. So ultimately always good. But it was interesting to finally have a package manager that was able to be used so quickly that you can start using it for complete new use cases that you couldn't have thought of before.

Jon Krohn: 00:45:01 Nicely said. Yeah. Very cool. I haven't been using UV myself yet, but it sounds like I should be.

Thijs Nieuwdorp: 00:45:08 I can definitely recommend it.

Jon Krohn: 00:45:09 Nice. Yeah. This next one is maybe, well, this is a bit targeted more, Jeroen though. Thijs might have lots of opinions about this as well. This is about another open source question, another open source library, well, actually a whole open source language, which is the command line, so bash at the command line. And so your whole previous episode on this show, 531, was all about data science at the command line. Obviously, you've written two editions of the book as discussed. You've written R packages that make the command line more interactive and playful. I don't know if I can pronounce them properly. There's Raylibr, R-A-Y-L-I-B-R.

Jeroen Janssens: 00:45:54 Oh, Raylibr.

Jon Krohn: 00:45:54 Raylibr.

Jeroen Janssens: 00:45:56 Well, Raylibr is a wrapper around Raylib which has nothing to do with the command line, but that's a C library to create video games.

Jon Krohn: 00:46:06 To create video games?

Jeroen Janssens: 00:46:07 To create video games. Yeah. Yeah. And actually, I've given a talk at NYR a couple of years ago where I advocate for some of these things that video game programming offers like 2D and 3D graphics and interactivity, how that can be used for doing data science. So that's Raylibr. That was a fun project that you can actually create 3D environments from R. But it has nothing to do with the command line.

Jon Krohn: 00:46:39 No.

Jeroen Janssens: 00:46:40 So let's talk about the command line.

Jon Krohn: 00:46:41 Yeah. So let's talk about the command line. I don't know if any of the other, the spec or Tmuxer, if those have anything to do with the command line.

Jeroen Janssens: 00:46:52 Oh yeah, yeah. So Tmuxer, that's a wrapper around Tmux, the terminal multiplexer, if you want to run multiple terminal sessions at these sessions. And you can interact with that programmatically from R using the Tmuxer package.

00:47:06 Now, that's actually a triad of packages that you just... Well, you mentioned two of them. So Tmuxer. There's Rexpect, which is from R expect, I'm not sure if you're familiar with the Expect.

Jon Krohn: 00:47:18 I'm not. No.

Jeroen Janssens: 00:47:20 That allows you to automate things on the command line, so log in to a server automatically and then do certain things based on certain outputs. I wrote a wrapper for that. And then there's also Nitter active. And now, I'm probably the only one who has used these packages at all, but I needed those.

00:47:40 That's, of course, why you shoot right software in the first place for yourself. But I needed those three packages in order to be able to write a book about the command line using bookdown, right, using Nitter, which is the system that I used at the time.

00:48:01 So a lot of, what's that called, yak shaving or bike shedding, lots of work, not actual writing, but lots of work. And we had some of that for the Polars book as well. But yeah. When you're an engineer, when you're a developer and you're writing a book about development, there's always some kind of developing that you need to do on the side for the book, whether that's just to get in the groove or whether it's actually helpful-

Thijs Nieuwdorp: 00:48:30 Just to make life easier.

Jeroen Janssens: 00:48:31 Yeah, yeah.

Jon Krohn: 00:48:33 This episode of SuperDataScience is brought to you by the Dell AI Factory with NVIDIA, helping you fast-track your AI adoption - from the desktop to the data center. The Dell AI Factory with NVIDIA provides a simple development launch pad that allows you to: perform local prototyping in a safe and secure environment. Next, develop and prepare to scale by rapidly building AI and data workflows with container-based microservices. Then, deploy and optimize in the enterprise with a scalable infrastructure framework. Visit www.Dell.com/superdatascience to learn more. That's Dell.com/superdatascience.

00:49:13 Thank you for that extra context and explaining what those packages do. And all of that was basically just to bolster in a few minutes your expertise on doing data science at the command line as a real expert in that. So something that I want to highlight here is that in your course, embrace the command line, which folks can check out. It's online, and there's more information at jeroenjanssens.com/embrace.

Jeroen Janssens: 00:49:38 Sorry to interrupt you.

Jon Krohn: 00:49:39 No, please do.

Jeroen Janssens: 00:49:41 This was a course that I've given a couple of times. This was a cohort-based course using Maven. And I've given it a couple of times, and I no longer do it. So unfortunately, it's not available online. We can cut that out.

Jon Krohn: 00:49:58 No, it's okay. You just leave it in. I don't mind my mistakes being on air.

Jeroen Janssens: 00:50:03 Neither do I.

Jon Krohn: 00:50:05 Nice. Okay. Yeah. But nevertheless, in that course, or at least in the course information, you say that the command line is as powerful as it is intimidating. So for our listeners out there who maybe haven't crossed that emotional barrier, maybe they do program, they use Python, maybe they use R, or whatever programming languages they use, but they haven't crossed that threshold, that emotional barrier, to start using the command line.

00:50:34 What do you recommend to students to get past that emotional barrier and see the command line shell as a great creative space for data science and software development?

Jeroen Janssens: 00:50:44 Yeah. It's unfortunate that when you first see this window, this terminal, this blinking cursor with a prompt waiting for your commands, it's such a shame that this is indeed so intimidating. Of course, when Unix or Linux was first created in the '60s and '70s, at that time, they didn't even have screens. They weren't all flashy.

00:51:14 So there is indeed a hurdle for you to take, for you to embrace the command line. And there are certain tricks that you can apply, certain changes that you can make in order to make the command line a more pleasant environment, a more forgiving environment. So things that I always like to do are, let me try to come up with a couple of them.

00:51:41 First of all, use colors that you like, use a font that you like, add in aliases so that these long commands, these long incantations that you don't have to remember them by heart. So you make the experience more ergonomic.

00:52:01 It also helps to work in an isolated environment so that you know that you won't be able to break anything. Docker can be used for this. And I think if you do these kind of things, experiment with the command line every day for a little bit. Don't try to do everything all at once. I don't. I just use it here and there as a complementary set of tools in addition to, well, all the other data science tools that you want to use. And then, yeah, you'll gradually build up more and more appreciation of the command line. You'll be able to embrace it more and more, make it your own.

Jon Krohn: 00:52:44 Very nice. I love that. Nice. And so I realize now I've strayed, I've already switched gears and taken us away from your Polars book. But I remember now that there were a couple of stories that we discussed before coming on air that I really wanted to cover before this episode ended. So we're going to have this creating experience for the audience of going back to your book, but maybe that's a nice place to end anyway.

00:53:12 And so first one I wanted to talk about, and so you guys may or may not be aware of this, but in 2025, two of the biggest sponsors of this podcast to whom we're very grateful because it allows us to keep the lights on and make this show for everyone are Dell and NVIDIA. And it sounds like for the appendix of your book, Dell and NVIDIA, you had some kind of partnership with them that allowed you to do more, explain how they're involved with your book.

Jeroen Janssens: 00:53:43 Yeah. So at a certain point, I got a LinkedIn message from NVIDIA. It was something about being an influencer. And at first, I didn't think much of it. After a week or two, I decided to reply like, "All right. I'm interested. Let's chat." And it turned out that they actually wanted to collaborate with us. They were quite eager to send us some hardware so that we could benchmark Polars on the GPU. And we're like, "Great." Only thing is we don't have anything to put that video card in.

00:54:21 So that's when they brought in their partner, Dell. And Dell was able to supply the rest of the hardware. Yeah. So that was a fantastic collaboration. And the way we did this, Thijs can say more about the software side of things, but in terms of hardware, it was all in the states. So Dell had this laboratory where they had a beefy machine, and they were able to swap out different NVIDIA video cards. So we did the RTX 6000.

Thijs Nieuwdorp: 00:54:56 Ada generation.

Jeroen Janssens: 00:54:57 The Ada generation. so these were all professional video cards, not the consumer level.

Thijs Nieuwdorp: 00:55:02 Yeah. The work station variants.

Jeroen Janssens: 00:55:04 Yeah. So it was very important for us that we were able to benchmark things ourselves that we wouldn't just copy numbers from some leaflet, some promotional material. We wanted to produce these numbers ourselves if we were going to put them in our book. And that was all fine.

00:55:28 NVIDIA and Dell thought it was a great idea. And so eventually, we're able to try out five different video cards for a number of different settings and packages. And that's all reported now in the appendix of the book. But Thijs maybe, you can say something about how you actually benchmarked.

Thijs Nieuwdorp: 00:55:46 Yeah. So to start off with a little more context is that NVIDIA has a team called RAPIDS which is working on creating all kinds of general purpose computing packages that can run on the CUDA platform. And CUDA is the calculation platform that NVIDIA opens up. So you can run any kind of calculation effectively on the GPU. And the difference between normal CPU and GPU is that GPU has many relatively dumb, simple processors, but just many of them.

00:56:19 So if you are able to bend a problem, a calculation problem into something that the GPU can run, it oftentimes accelerates by a lot, by factor up to 10. So they also did this for packages like Pandas. They have cuDF is what their package is called. It's a dataframe library but runs on a GPU. And they wanted to collaborate with Polars as well.

00:56:46 But since Polars has this layered architecture where it runs through an optimizer first and only then gets sent to an engine, it would be a waste to just put the Polars API on cuDF and just translate it to normal cuDF functions because a lot of the performance enhancements from Polars comes from its optimization.

00:57:06 So instead, RAPIDS worked together with Polars and designed a GPU engine that gets input from that optimization layer. And because they recently finished the open up a beta for this new package, they got in contact with us to ask, "Hey, you guys are working on the book of Polars. Do you want to collaborate?" Well, with the terms that we could test stuff ourselves and benchmark ourselves, we definitely said yes because it turned out to be a lovely collaboration as well.

Jon Krohn: 00:57:37 That's a cool story.

Thijs Nieuwdorp: 00:57:38 Yeah. Definitely.

Jon Krohn: 00:57:40 What were the results? Is that what you're going to tell me now? Please, tell me the results.

Thijs Nieuwdorp: 00:57:43 It's a lot faster. Yeah, it is. Yeah. So we already noticed that in the beginning the promotional material was a bit careful with what kind of size of data set it would be beneficial from, and it turned out from the test that we were doing that it's quite quickly already because data needs to be transferred to the GPU, you get a small overhead.

00:58:05 So you start seeing the difference when the data set size grows, but it's already from one gigabytes and up. So it's relatively quickly because most data that you would work with in a professional setting usually tends to grow a lot. And we also noticed that even the relatively, you'd say smaller GPU cards with less processors already have a big speed-up from just using the GPU engine.

Jon Krohn: 00:58:34 Very nice. Cool project, great results, unsurprising results given everything that we know about Polars is already including the examples that you gave at Alliander earlier in this episode. But cool to have that comprehensive benchmarking there on four different NVIDIA cards. And cool that Dell supplied the server for you to be doing all that benchmarking yourself.

00:58:54 All right. And then one final story that I want to get in here, so I mentioned already how Marco Gorelli... So Marco Gorelli was our first ever Polars episode on this podcast. So that was episode 815. And then he introduced me to Ritchie, the creator of Polars, who came in not long after that, a couple months later in episode 827.

00:59:12 And now you guys, you're the final episode in the trilogy on Polars. Well, probably not the final. We'll have more. But for this, it's like the original Star Wars four, five and six. Your episode six of the original Star Wars. So I understand that there's an amusing story involving Marco somehow sabotaging your book and forcing you to rewrite an entire chapter.

Jeroen Janssens: 00:59:42 Yeah. It's amusing now. So we have to go back a little bit further. Was at a Christmas party organized by Xomnia where Ritchie was also present. And Ritchie was like, "Yeah, Polars is going to have data visualization capabilities." What? Python doesn't need another package to do data visualization. There are so many out there.

01:00:08 So at first I was like, "an, to keep expanding the library, we just want to finish this book." So I was quite upset at first. After a while, I started to realize, "Okay. Maybe, it's not so bad." If the book has a chapter about data visualization, maybe it'll sell better if it has some pretty pictures. So I started writing, I was quite happy to find out that Polars itself doesn't do any data visualization. It has the Df.plot namespace. But then every method in that namespace calls out to another package, hvplot.

01:00:49 And I wasn't familiar with hvplot yet. It's this meta package which can target Matplotlib and Plotly and another one. Okay. Thank you. And so, okay, I really had to get into hvplot, but I didn't just want to write about hvplot. I also wanted to include Great Tables. I was a big fan of that. And you could argue that presenting a table is also a form of data visualization.

01:01:23 Plotnine, I'm a big fan of plotnine, so it was going to be a huge chapter. So I had written this. And then all of a sudden, I see on GitHub this pool request by Marco Gorelli. He was like, "Okay, I'm going to change out hvplot for Altair." I'm like, "What? Now, I need to rewrite the entire chapter, or at least a big portion of it. Marco, what are you doing?"

01:01:52 Now, I know that Altair is a very good choice for this, especially for when you are working in a browser and you want to create interactive data visualizations that is something that plotnine, for example, doesn't support. So Altair definitely has its use cases. And I should have known better as well. Hvplot at the time, or the whole plotting functionality in Polars was marked unstable.

01:02:21 So I should have known better. I was just too happy to get it out there. And you know what? Marco and I, we get along really well. We collaborate now on getting Narwhals his project into plotnine so that plotnine better supports

Polars and as well as Pandas. But, yeah, that was the story that I had to rewrite nearly everything inside Chapter 16, visualizing data.

Jon Krohn: 01:02:54 Nice. Great story. And it is funny to imagine Marco sabotaging your book because he's an extremely nice, because actually the episode with Marco, I don't know if you know this or not, but I recorded it with him in person in London. And he took a train-

Jeroen Janssens: 01:03:08 [inaudible].

Jon Krohn: 01:03:09 Yeah. He took a train from Cardiff to London, which is three hours or something, to come and record the episode. So we went for dinner afterward as well. And you really get this impression of a man who is exceedingly kind and conscious.

Jeroen Janssens: 01:03:28 Yeah. I hate it. I hate it. I wish he wasn't like that. No, he's a very generous and kind person. Definitely pleasure to work with.

Thijs Nieuwdorp: 01:03:37 And I definitely love his dry sense of British humor. It's perfect.

Jeroen Janssens: 01:03:43 Yeah. Every time he speaks at a conference, he tries to incorporate an expression from that country. So when he was presenting at PyData Amsterdam, he used the expression [foreign language] which translates to too bad peanut butter. Doesn't make any sense if you're not Dutch. And in Germany, he talks about the concept of [inaudible] to open up all your windows and think in PyData in Paris, he had a data set about the pronunciation of pain au chocolat. At one point, the further south you go, he changes something else. Something like that worked in. He's a funny guy. Definitely. Yeah.

Jon Krohn: 01:04:26 That's really funny. Yeah. So people want some more of that humor. He was extremely technical. In this episode, we haven't gone and that wasn't really the point of the episode in that one as well as Ritchie's episode, so 815 with Marco Corelli or 827 with Ritchie Vink. In different ways, you get into the nitty-gritty of why Polars is so fast under the hood.

01:04:52 And so if people want to check those out, we'll have links to those in the show notes. All right. And so that brings us to the end of this episode basically. It's been awesome having both of you on the show. Jeroen, welcome back.

Jeroen Janssens: 01:05:05 It's been a pleasure.

Jon Krohn: 01:05:07 Yeah. Hopefully, we'll have you back again soon. Thijs, I hope your first podcast experience wasn't too painful to end things. Great. Thijs, let us know if you have a book recommendation for us to wrap things up here.

Thijs Nieuwdorp: 01:05:21 I do. Yeah. Normally, I'm more into the fantasy side of things to escape, but one of the things I love is when someone is able to explain something complex in a way that has a proper story. And one of the books are recently read was Immune by Philipp Dettmer, the main writer of Kurzgesagt, which is also a YouTube channel which has many explanation videos on all kinds of intense topics.

01:05:51 But this book, he dives into the immune system, which is exceedingly complex, yet he still is able to explain it very, very well in a very both informative and entertaining way. So that's definitely one of the books I recently finished that I love reading.

Jon Krohn: 01:06:06 I love that recommendation. It has been advertised to me at the end of a number of Kurzgesagt videos. Kurzgesagt is one of the few YouTube channels that I subscribe to, and it is excellent.

Thijs Nieuwdorp: 01:06:18 Yeah. That's perfect.

Jon Krohn: 01:06:19 If I could go back in time and somehow be responsible for one YouTube channel, it would probably be Kurzgesagt. I think it's amazing.

Thijs Nieuwdorp: 01:06:26 Good choice.

Jon Krohn: 01:06:27 And for our listeners out there who don't speak German, Kurzgesagt means shortly said. And so it is, just like you described, his Immune book being well-spoken, easy-to-understand on a complex topic. That's what all the videos on the channel aim to do, and I highly recommend Kurzgesagt. So many fascinating topics, scientific ones, but also philosophical ones. It really gets into the big questions of life in the universe. It's interesting. Nice. And Jeroen, do you have a book recommendation for us?

Jeroen Janssens: 01:06:59 Sure, I do! I am currently enjoying UNIX: A History and a Memoir by Brian Kernighan. It's a short book. I haven't finished it just yet. But as we have talked about, UNIX can be very dry topic, but it is so interesting to learn about the history and the people behind it and the politics and the things that went on in developing such a profound piece of software.

Jon Krohn: 01:07:29 Truly revolutionary. It doesn't understate. If you're watching the video version, you can see what the book looks like. But, yeah, truly revolutionary UNIX is. It's staggering to thank the shoulders of the giants that we now get to stand on inventing our relative trivialities in computing, which, although interestingly, even though they're relative trivialities compared to say UNIX because of what's to come in terms of this age of intelligence that we're emerging into with intelligent machines, maybe in decades, they'll be looking back and thinking about Great

Tables. And the big difference is the history of Great Tables book problem.

01:08:14 Guys, it's been so great having you on the show. If folks want more of your humor and brilliant insights, how can they follow you after the program?

Jeroen Janssens: 01:08:23 Well, the easiest way is probably go to the same website that we already mentioned, polarsguide.com And from there on, you can find links to our LinkedIn pages and other places where we are. That's probably easiest.

Jon Krohn: 01:08:37 Cool. Yeah. And so basically you would say LinkedIn is the main social medium for you both.

Jeroen Janssens: 01:08:42 Yeah. Yeah. I'm trying out Blue Sky. I've been pretty active or I've been enjoying Twitter for a long time, but that has changed. LinkedIn, we seem to get a lot of response on LinkedIn. Whenever we post something about the book, there's a good vibe. There's a lot of other stuff going on LinkedIn. But yeah, it works. It works.

Jon Krohn: 01:09:11 Yeah. LinkedIn is working these days, which we can't say for every social media platform out there. That's definitely where you can find me the most active as well. Awesome. Thanks so much. This was another great episode. Really had an awesome time with you guys. Appreciate you co-locating yourselves in the well-appointed perfectionist Thijs and Jeroen studio over there. And, yeah, I look forward to welcoming you guys on the show again sometime.

Jeroen Janssens: 01:09:44 Thanks, Jon.

Thijs Nieuwdorp: 01:09:45 Thanks so much for having us.

Jon Krohn: 01:09:51 What a tremendous episode with Jeroen and Thijs in it. They covered Python Polars and how it's a high-performance dataframe library that uses a declarative approach allowing users to state what they want as an end result while the engine handles optimization.

01:10:05 We talked about how the book Python Polars: The Definitive guide by Jeroen and Thijs provides comprehensive coverage of Polars and includes benchmarks showing Polars can reduce memory usage and compute time by up to 10X compared to Pandas, the standard, well, or at least until recently, until Polars came along the standard dataframe operations library.

01:10:25 Polars uses a grammar where expressions function like recipes, operations or steps and functions or methods are the cooks creating readable code without excessive brackets. We talked about how when implemented at Alliander, a Dutch power group provider, Polars reduced memory requirements from 500 gigabytes to 40 gigabytes, so a 10X reduction and doubled processing capacity.

01:10:50 We also talked about other things than Polars. So we talked, for example, about how the UV package manager is emerging as a faster Rust-based alternative to poetry, allowing for quick environment setup and tear down for benchmarking or whatever. It doesn't need to be for benchmarking. It can be for whatever reason you would need Python packages.

01:11:10 And finally, we talked about Great Tables and how it provides styling capabilities for data tables, allowing presentation ready formatting without modifying the underlying data. As always, you can get all those show notes including the transcript for this episode, the video recording, any materials mentioned on the show, the URLs for Jeroen and Thijs's social media profiles, as well as my own at superdatascience.com/885.

01:11:34 And if you'd like to engage with me in person as opposed to just through social media, I'd love to meet you in real life next week at the Open Data Science Conference, ODSC East, running from May 13th to 15th in Boston.

01:11:50 I'll be hosting the keynote sessions and along with my longtime friend and colleague, the extraordinary Ed Donner. I'll be delivering a four-hour hands-on training in Python to demonstrate how you can design, train, and deploy cutting edge multi-agent AI systems for real life applications. That is going to be fun.

01:12:10 Thanks, of course, to everyone on the Super Data Science podcast team, our podcast manager, Sonja Brajovic, media editor, Mario Pombo, Nathan Daly, and Natalie Ziajskii on partnerships, our researcher, Serg Masís, our writer, Dr. Zara Karschay, and our founder, Kirill Eremenko.

01:12:27 Thanks to all of them for producing another laugh-filled episode for us today for enabling that super team to create this free podcast for you. We are, of course, deeply grateful to our sponsors, you listener. You can support this show by checking out our sponsor's links, which are in the show notes. And if you yourself are interested in sponsoring an episode, you can get the details on how to do that at jonkrohn.com/podcast.

01:12:51 Otherwise, share, review, subscribe, edit videos into shorts to your heart's content. But most importantly, just keep on tuning in. I'm so grateful to have you listening and hope I can continue to make episodes you love for years and years to come. Until next time, keep on rocking out there, and I'm looking forward to enjoying another round of the SuperDataScience Podcast with you very soon.