Podcast: The Open Source Way

Episode 51: OpenSearch

Transcript

Karsten Hohage: Welcome to the Open Source Way. This is our podcast series, SAP's podcast series, about the difference that open source can be. And in each episode, we'll talk with experts about open source and why they do it the open source way. I'm your host, Karsten Hohage, and in this episode, we will talk to or I will talk to Pallavi Priyadashini, Jürgen Walter and Karsten Schnitter about open search. Hi, Pallavi, Jürgen, Karsten. Glad that you could all make it at one and the same time.

Pallavi Priyadarshini: Thank you. Thank you.

Karsten Schnitter: Thanks for having us.

Karsten Hohage: All right, let's take a look at who these people are. Pallavi is a lead engineer for search security releases and dashboards at the OpenSearch Project and at Amazon Managed OpenSearch Service. She's also a member of the Technical Steering Committee at the OpenSearch Project. Karsten is a software architect working in the observability area of the SAP BTP or Business Technology Platform. He's responsible for SAP Cloud Logging and other logging services for SAP BTP. And he is also a member of the Technical Steering Committee at the OpenSearch Project. Jürgen is an observability expert and software engineer in SAP BTP observability, mainly driving cloud logging. He's a code contributor and member of the newly formed technical advisory group “Observability” at the OpenSearch Project. You're all in some form members of the OpenSearch project, and this is also our topic. Has this, for all of you so far, been only virtual, or have you all met in real life? Maybe, Jürgen, you want to start?

Juergen Walter: Karsten and I'm sharing the same office, so that's easy. And with Pallavi, maybe we could have met in San Francisco, but there were so many people, and it wasn't clear at that point in time that we would be in a podcast together.

Pallavi Priyadarshini: I have met a few of the folks, and we regularly run into each other at conferences now. So it's great to see SAP's representatives come out and support OpenSearch in several conferences around the globe.

Karsten Hohage: All right. So you probably met, but you didn't know who you were. This happens with thousands of people every day. And next time you will know each other, that's an additional benefit of this podcast. But let's maybe

look at what we even want to talk about, OpenSearch, Pallavi, in short, what is that?

Pallavi Priyadarshini: So thank you again for having me and as a representative of OpenSearch. So for those of you who aren't familiar with OpenSearch, we are a community-driven, open-source, search, analytics, and vector database platform with integrated tools for observability, security, visualization, and AI-powered applications. And all of these are available under the 2.0 Apache license. And a little bit of history. OpenSearch started as a fork of Elastic Search in early 2021 after Elastic changed to a more restrictive license. Four years later, OpenSearch project has created its own path and it has its own identity and a trajectory of innovation driven by the community. We are a community of over 3,500 members, and OpenSearch was transitioned to the Linux Foundation in September of 2024, which accelerated the growth of the community and the technical innovation.

Karsten Hohage: All right. That sounds, from all I know about search engines and analytics and so on, like a pretty huge project there with the functionality you listed. Jürgen, how and what of that is currently in use at SAP?

Juergen Walter: So it is a big project with a lot of focus topics. But the major use case within SAP is observability at the moment. And Karsten Schnitter and I are working on a service called Cloud Logging, which is providing observability for SAP BTP, the SAP Business Technology Platform. And it supports logs, metrics and traces. To better understand what this service really provides, it's, I think, good to understand the environment we operate in, which is mainly the business technology platform, which has different environments, such as Cloud Foundry and Kyma. By the way, both also open source. And I think also Kyma was once in one of your podcast episodes. And there are environments where applications can be deployed. So where then, for example, extensions for setups of customers and SAP internal are deployed. And these applications need to be observed. And these environments come with an opinionated way of providing observability data. And we have built on top of OpenSearch some tailored integrations using parsing and analytical dashboards so they get an opinionated, all-inclusive package for observability.

Like we're not using all of on providing all of the features, but it's mostly like an OpenSearch as a service flavoured for observability. And when I just said major use case so that you get some impressions, we're currently running more than 11,000, maybe 12,000. And I don't check it every day. Clusters, we had huge external growth. So many, many companies you might have heard are adopting

it. And we already reached like petabyte scale years ago. So that's millions of gigabytes inserted every month.

Karsten Hohage: Yep, I would have assumed so. And I would have asked you about sizes because in logging, and I guess especially cloud logging, it is just a few seconds and you have your gigabytes, right?

Karsten Schnitter: We see customers ingesting megabytes per second.

Karsten Hohage: You have, sorry?

Karsten Schnitter: Oh, hundreds of megabytes per second is what some customers ingest.

Karsten Hohage: Okay, then I was one magnitude exaggerated, but yeah, on that scale. Karsten, as you chipped in there, I believe, and Pallavi has already hinted at that, there is quite a history to the project, as she already said, coming from Elasticsearch. Do you have anything to add to that from SAP perspective?

Karsten Schnitter: Yeah, so when I joined the logging services eight and a half years ago, we were running services based on the Elkstack, so Elasticsearch, Logstash, Kibana, and that is what we used to build cloud logging in the first place. We were using Open Distro from AWS when it came out because it made our lives easier, but then Elastic basically changed the license. So at that time, because we were selling the service to customers, that was a huge shock because it prohibited us from doing that with the newer versions. So it was quite clear, okay, when the fork for OpenSearch was announced that we adopted and now also contribute to it because of the setup in a way where it cannot be taken away anymore. So that's really important. But this is basically the history how we came to OpenSearch and we have been there from the start and I'm really happy how the project is coming along.

Karsten Hohage: All right. Seems like another one of the big projects with early SAP involvement that the world should hear more about. So from there, from that history with Elastic and then going into OpenSearch, Pallavi, how is now the setup of communities, foundations, licensing model, etc. for OpenSearch?

Pallavi Priyadarshini: So OpenSearch project now is governed by the OpenSearch Software Foundation under the Linux Foundation. And it is, there are 16 companies that are part of the OpenSearch Software Foundation, including SAP, which is a premier member of the foundation. And to govern the

foundation, there is a governing board and there is also a technical steering committee, which is comprised of 15 technology leaders from 10 different companies. So what we really want to enable is open collaboration. Ever since OpenSearch project moved to the Linux Foundation in September 2024, we have really seen the growth of that community and also the collaboration that comes along. And that was the whole point of AWS moving OpenSearch project to the Linux Foundation in that AWS did not want it to be a single vendor project. So now what has happened is the community has grown. We had, like I mentioned earlier, we have more than 3,500 contributors from 400 different organizations. So these are not just contributors, but they also use OpenSearch. For the different use cases in search, analytics, observability, or as a vector database. And we really promote the model of open collaboration to give a few examples. Our most recent major version release was OpenSearch 3.0 that happened in May 2025.

And in that release, there are significant innovations coming from non-AWS contributors such as Uber. So Uber has really transformed the OpenSearch architecture into a cloud-native architecture.

ByteDance contributed significant performance improvements to the core architecture in terms of the segment replication. It also contributed conversational search. SAP was instrumental in contributing back FIPS compliance to the software and has been contributing to the security plugin as well. And there are others like DTEX, which has made significant contributions to the dashboards, which is powering the observability. NVIDIA contributed back GPU acceleration for vector index bills. So we are seeing innovations coming in from many different companies, which is the true power of OpenSearch. And because of the technical steering committee, the roadmap is also community driven. The Technical Steering Committee gives the guidance on the processes that run the roadmap. However, the roadmap is really controlled by the maintainers and the community contributors.

Karsten Hohage: Jürgen, you are the one who's also on the Steering Committee? Do I remember that correctly? No, Karsten is on the Steering Committee. Okay, sorry, Karsten. Do you want to add anything to the job of the steering committee or has Pallavi covered everything also from our perspective, from the SAP perspective?

Karsten Schnitter: So Pallavi really pointed out what the general idea of the technical steering committee is, so what is usually done there. For SAP, it's important that we are there to also shape where the project is going and discover and discuss our vision or requirements that we have. So we have heard that we

are contributing a FIPS compliance for the core. Of course, this is not everything that needs to be FIPS compliant, But this is something that was discussed with the maintainers and with other contributors. So a shout out to Scandinavian Airlines Services, who have contributed much FIPS compliance to the security plugin, actually. And we ran tight collaboration with them. But here, for example, it really pays out to be part of the technical steering committee, because we could coordinate this with the major release of OpenSearch 3.0, which contains upgrades to the Java version, which otherwise would have conflicted with the change and still keeps us busy because this change is not done yet.

It's almost done, but not quite. And this is where the community really comes together and showcases that for such a major change, it is important to collaborate with everybody who's there and have really good coordination is required. So we also have the huge thanks to the folks from AWS who are reviewing all the changes, actually. So this is Andre Redko and I forgot the name.

Pallavi Priyadarshini: Andrew Ross and Craig.

Karsten Schnitter: Andrew Ross, Craig. Yeah. Yeah. Sorry for that. But yeah. So it's really nice to see how everything comes together and how the collaboration works out.

Karsten Hohage: All right. And as SAP has been a member from the start, was it mainly the FIPS compliance or the other contribution we should mention? And there's one other thing I'd be interested in after that.

Karsten Schnitter: So actually, we've started with a different focus. So the FIPS is the latest thing that we do to, this is really high grade enterprise feature, like having FIPS compliance. This is like in US governmental standard that you have to fulfill for that they're non-military appliances. So if you want to have governmental contracts, but showcasing that you can do that also enables digital sovereignty, because basically it's a set of algorithms for security that you need. And if you can configure them, it's basically also for EU use cases. This is really important. But we actually started with observability where Jung and I are at home. And this is when we saw the emergence of open telemetry. SAP ran an internal architecture board where it was decided, okay, that this is the future action that we would be going to. So this is where I was also a member of. And then I said, okay, if we now want to enable open telemetry for all SAP teams, we better have a service that can take in the data. And this was our first or one of the first major contributions that we did through the project. So support for

traces was always there, but metrics and logs were lagging. And that was the first contribution. I think that was really a big contribution that we did to the project. So the data proper ingestion tool has learned by our contributions to speak open telemetry for metrics and logs. And that's also how I became a maintainer of that project, actually.

Karsten Hohage: All right. Okay. That was, at the time, that was not even your task yet. And you did this as one of the, I don't know, free roaming contributors? Or because you said that's how you got to the project?

Karsten Schnitter: Yeah, we just contributed. We contacted one of the maintainers, David, which we have a really good contacts to. I talked to him a lot in Amsterdam on the last OpenSearchCon, and this is how the contribution to that particular project started. I think this is one of the major contributions that we did, and there's more to come for sure.

Karsten Hohage: Yeah, and I guess, as I said in the beginning, we said what search, analytics, and that with the interfaces and the engines and everything sounds pretty huge. So where will it go from here? What's next? What trends are there?

Karsten Schnitter: We established with the TSC a technical advisory group for the OpenSearch project that is concerned with observability. That was also discussed and put in place with our AWS colleagues, and everybody is invited to join this group. And for example, we invited Juergen into it.

Juergen Walter: He needed an expert.

Karsten Schnitter: So now we can discuss in this group where the observability of OpenSearch should go. With all the efforts that we did earlier, like the efforts done by AWS and the contributions by us, we are in the place where you can make OpenTelemetry data into OpenSearch, either using DataPrepper or in part the OpenTelemetry Collector. And you will find predefined content in the OpenSearch catalog to visualize the data. And since this year, all the data formats are aligned so that you can really set it up easily. So that's something that we had internally in SAP for quite some time. And now with all the contributions, there's something that is there and we plan on going forward with that. So building on top of that here, right?

Juergen Walter: Yeah. So I also want to add a little bit and maybe explain it for the less experienced listeners. So with OpenTelemetry, we really have the standardization of observability signal formats. And this really goes deep. Like

now there is standardization on how trace spans for databases look like for HTTP calls. And now we can really build cool stuff on top, which we previously had to build for each of the environments. So we have had or have, still have, dashboards for Cloud Foundry, dashboards for Kyma. And now like the whole world is innovating on the same standardized formats. And that's really a game changer.

Karsten Hohage: Which it usually is in any space where the world can agree on certain standards. I once heard one of our colleagues give the example of the cross-Atlantic shipping that multiplied once the world had agreed to use containers instead of then use that as a negative. It's probably a classic example in the world of standardization. Yeah, always beneficial, I guess Pallavi, from your perspective, what's the roadmap?

Pallavi Priyadarshini: Both on search and observability, we are seeing some exciting innovations come in. The way OpenSearch is positioned today, it's really becoming the AI foundation to power all these different use cases. So on search side, what we are seeing is it is evolving or it has already evolved from a keyword-based search. It went to semantic search, hybrid search, multimodal, where you can do images, unstructured data, audio, all kinds of data together, right? And the most recent trend is around conversational search, and it's also becoming the foundation of this agentic AI use cases, bringing in a new way of interacting with LLMs, etc. And OpenSearch is in a unique position with its vector database and its AI search capabilities, which is pretty important for the agentic AI use cases. And it is only because of open source that we are able to move so fast. Like on the search side, we are seeing innovations come in from our partners, such as I had mentioned earlier, ByteDance is contributing to the conversational search.

Similarly, IBM DataStax, it contributed a new vector engine, JVector, to open search. Now, on observability as well, we are seeing that OpenSearch is becoming the central place to drive your analytics slash logs, metrics, traces, experiences. And again, we are seeing contributions coming in. Like we already talked a lot about SAP is critical partner for the OpenSearch project. It is a founding member of the OpenSearch Software Foundation since day one. And other companies also look up to what SAP is doing. In fact, the case study that SAP published in terms of its use with OpenSearch, it was very popular with the other set of partners. We had a conference last week in San Jose, OpenSearchCon, where Hari from SAP, he presented the keynote, which was very well received. And we also saw other partners which are coming together to make the observability side of things more robust. In fact, we had a discussion last week with

Contributors from Apple who are interested in the pipe processing language upgrades to OpenSearch. And we just had core maintainer announced from Apple as well because they have started contributing to the code base. So the bottom line is OpenSearch is becoming the AI foundational platform to power a bunch of these use cases on search, observability, and analytics. And what we are seeing is with the right foundation or the platform we have different companies are coming in with their own use cases and contributing code back okay

Karsten Hohage: All right just a second yeah Karsten.

Karsten Schnitter: Let me comment on that so it is really great to see all the ai and vector search use cases of open search and now we come to the holy grail for observability and this is to somehow enable this kind of features for observability so the problem is observability data each and every set that each log message is usually relatively small. And if you want to do machine learning on it and create a vector, you multiply the data. Like if you have a 1K log message that's already pretty long, but the only thing that is bigger is stack traces. But if you have the usual log message that's relatively small and now you create a vector that is 4K, and that is not a good idea. So in the end, it comes down to how to combine your data into larger blocks and then apply the machine learning on top of that. And that is something that is really interesting, something that can be done with OpenSearch. This is the PPL support, for example, that Paobaki mentioned and others. And this will become very, very interesting.

I want to give a quick shout out to our SAP colleagues because internally I've already seen a POC that is a REC-based approach that is looking inside your observability data, mainly the logs, and then giving recommendations how to reconfigure your application or what to fix because you are experiencing crashes. That's at the moment on POC level, but I hope it becomes a product soon. And that already showcases how much you can do the AI features of OpenSearch and how much you could integrate it, given that there's also an MCP server in OpenSearch 3, for example. So there's a huge potential, but we have to figure out how to apply it for our use cases.

Karsten Hohage: Just in case that didn't become clear, you said that's on POC level, so on proof of concept level, right? but one back, you said PPL, that's what again?

Karsten Schnitter: That's the pilot processing language that OpenSearch uses to analyze data on the fly and reformat it, transform it. It's similar to Splunk

processing language, but it's a complete kind of its own. Maybe, Pallavi, can you explain it a little bit better?

Pallavi Priyadarshini: Yeah, I think you covered it, Karsten, but basically PPL is a powerful language, pipe processing language that allows you to discover more insights from a lot of data, right? So it provides these rich syntax with which you're able to power a lot of your observability and analytics experiences.

So what's happening is with every release of OpenSearch, we are upgrading the PPL language. And then along with the PPL language itself, we are also focused on the natural language experiences. I think that's the most recent trend we are seeing where the users don't really want to learn one or the other language and plus these agentic search use cases. So what OpenSearch is doing is it's focusing on the natural language experiences where OpenSearch does the language conversion for you. And again, going back to OpenSearch as an AI foundation or a platform, it is rolling out features that allows builders to build those experiences, right? I think Karsten mentioned that the MCP server, the model context protocol server support was unveiled in OpenSearch 3.0 and in OpenSearch 3.1 and 3.2. And we have a very tight eight-week release cadence. So in spite of the thousands of contributors we have, not from a single organization,

From hundreds of different organizations, we make sure we take all the payload and release every eight weeks. So in 3.1 and 3.2, we have expanded on the agentic support along with the MCP server. There is also something called agentic search, which allows you to have natural language experiences. And then there is agentic memory, which allows the semantic search to learn from

Prior interactions for higher quality search results. So we have built a foundation and every release is seeing upgrades. And from a dashboard and visualization perspective, we are also plugging in AI-generated experiences. How can you generate more personalized dashboards from natural language based on your previous history, etc. So I think this is a trend that we will continue to see more AI-enabled observability and search workflows, and which we want to be simple to develop, right? All these plugging AI use cases is difficult. You have to bring together many things. So part of the focus at the OpenSearch project is also how do we make it simple for builders to assemble all these different steps that come together. So we are focused on improving the getting started experiences across all use cases of search and observability.

Karsten Hohage: Concerning that, when we speak about all these, let's say, you mentioned the MCP server, you mentioned the Vector Engine for the

embedding semantics and so on. The one thing I was going to ask for five minutes now, maybe I've missed it. Does it run an own large language model or is it only these parts that surround it that I have just listed some of?

Pallavi Priyadarshini: So, Karsten, there are many ways. Again, OpenSearch gives the foundational blocks, but OpenSearch has something called ML Commons, using which there are these different connectors to different models, right? So you can connect out to external models or you can choose to host your model within your OpenSearch deployment. So it gives you the flexible options to either connect to any model or to host it within the OpenSearch deployment. And what the team is focused on is building the blueprints so that whenever these models come out, we are able to ship connectors as soon as possible. So to give you an example, when DeepSeek came out, like we all woke up one day and we saw a lot of news around DeepSeek, right? So OpenSearch contributors were able to ship out a connector for DeepSeek in two days because there are these blueprints using which you can easily build connectors to any model that comes out. So we support models from OpenAI, Cohare, basically a wide variety of them.

Karsten Hohage: Okay, but the simple answer would be OpenSearch does not create and run its own large language model. You do a lot around that. Some of these things I understand,

some I don't, but it doesn't produce its own large language model.

Pallavi Priyadarshini: It doesn't produce its own large language model. But I also want to say one thing before Karstina hand over to you. Off late, we are seeing a trend to go towards more lightweight models like sparse models, right, and have sparse embeddings, which give you better quality search results with very low resource consumption as compared to the dense embeddings and large language models. So that's a trend. And we have enabled the sparse model support in OpenSearch as well. We just shipped something called automatic semantic enrichment, which automatically generates those sparse embeddings. Karsten, over to you.

Karsten Schnitter: Yeah, I think like you are already alluding to like where the actual challenge for us lies. So we are using observability. That means we have, as I said, small documents in OpenSearch, but at a really high rate. And so basically we cannot go out and for each log message that you send us, try an LLM to give us a vector for that. That is like really expensive. But like having these kind of smaller models that can be directly deployed would be an approach that we could take at the moment cloud logging has disabled all of

machine learning or almost all of it because we have to determine what parameters are required first and

Juergen Walter: What kind of

Karsten Schnitter: Support we need to give to our customers if we ever enable or if when we enable it not say if we ever like when we enable it so at some point in time we will do that but we need to really be careful on what does this mean actually and how like which of the blueprints or what other blueprints SAP needs to add so that it makes really sense to apply this to observability. But it's a great community there and that is really supportive. You can ask a lot of questions, get really detailed feedback. I really appreciate all the models and all the work that is there, but also all the support that we receive.

Karsten Hohage: All right, great. Thanks for that clarification and addition. Jürgen, anything to add from your side concerning roadmap?

Juergen Walter: Maybe just about that observability use case, maybe to better understand where this could go. On the one hand, we have the observability data, but we also have then potentially other contexts like knowledge bases, code bases, which you then can all connect in an agentic workflow. And then the real beauty comes, there's like an outage and it's automatically detected that this is caused by a deployment of a change and that then the system could create a pull request to revert that change. And so this is where it's going. And first tests really show really promising results on this.

Karsten Hohage: Sorry, I don't know if I understand that correctly. is the charming thing in that is that you have all the information from the logs and traces and that's why you can probably infer where the error is or.

Karsten Schnitter: We actually have more in the cp so this is not like the common open platform but there is always abstraction on top so if you build an application on btp you're usually using some other kind of automation on top like deploy with confidence, build code. So there are other frameworks that actually help you generate the applications in the first place or manage them. So there's external context even. So there are some application frameworks like the CAP framework that are used, and then there's this kind of deployment automation on top. And so we have not done that yet, but there's a huge potential for integration with these kind of frameworks to get more context. So maybe you get an exception and you know the line of the code. You know where this application lives. You know where the code lives. You know when it was deployed. You can reach out to those systems or these systems can reach

out to us getting the exception and then provide the developer with much more context. So like look your second to last version that you deployed yesterday in production that fails in production only it was fine on your staging cluster but now it fails and probably the buck is around here so go and do that and in the meantime here is how to roll back the change so these are all automations that can be built on top of that this is vision to some extent but the poc that i mentioned goes for example into that direction okay

Karsten Hohage: But right here we are in the vision space right okay just So don't anyone out there take this as a promise or anything.

We are, I say it again, this was the vision space. Before we maybe dive any deeper into vision space, let's maybe rather come to our traditional last two questions of the podcast, which are, if people want to get started with using OpenSearch or contributing to OpenSearch, even where's the first places they should go?

Pallavi Priyadarshini: Overall, we are seeing that multiple organizations are adopting OpenSearch and these are not just big organizations. I do want to take this opportunity to say that it's also the smaller organizations OpenSearch really values any contribution. And we're seeing contributions not just from the big names that we talked about, but also a lot of individual contributors and smaller organizations like universities across the globe. So what's happening is in the community, we have built the mechanisms to easily onboard contributors, right? There is a community Slack, there is a community forum, there is an events page, we have a documentation,

Then obviously the community members, different teams like search team, storage, observability, indexing, they all set up their community triage roadmap meetings. And it's all decentralized, right? Although the overall processes are governed by the technical steering committee, but the maintainers and contributors operate independently to discuss their respective, we have something called RFC, request for comment, where you can pitch a new idea and the maintainers come together in the community meetings, discuss that. And if they are on board that this is a good thing to have it gets folded into a release you know into an upcoming release so there are multiple ways to get in involved obviously there's the open search project website that is like the entry point for everything i mentioned the slack the forum and we also have easy first issues that we tag for new contributors to pick up and you can reach out to any of the maintainers and they're super excited to onboard any new contribution and

Karsten Hohage: And then we, of course, have the famous case study by SAP, right? I assume that's public as well. Yes. And from the conferences like you just went to, there's probably also summaries and videos and talks and everything, right?

Pallavi Priyadarshini: A hundred percent. So in terms of community expansion, there are different, like we are not, OpenSearch is not just doing conferences, which are big, but there are also smaller meetups that's happening across the globe. And that's why we see that in terms of contributions, it's coming from different parts of the world. It's not just North America, Europe, we're seeing from Asia, we are seeing from South America. So we have a heat map of contributors and we see them popping across the globe. So in a way, it's really a momentum, a global momentum that has started. It's all because it is open source and all under Apache 3.0 license governed, 2.0 license governed by the Linux Foundation. And we're also seeing organizations, big and small, realize that the innovations, especially in the AI space, be it AI observability or AI search, it's happening so fast that no single organization can really build everything on their own. Folks are realizing that we are better off moving to a shared innovation model where the project is giving the foundational elements. And a lot of organizations might be thinking of similar use cases, right? What Karsten is mentioning, you want to find the insight from your logs, metrics, traces. Nobody wants to do those manual workflows.

We all have incidents where we run production installations, right? So rather than having a large scale event, everyone getting paged, everyone joining a call, these are things that can be automated. And we're already seeing some deployments in pockets and some in proof of concept. But there's so much that is possible behind this automation and AI enablement of observability and search use cases that multiple organizations are throwing their weight behind a community driven open search.

Karsten Hohage: All right. That was a great final pitch for why one should get interested or even join the project. The links and places where people can get started will include in the accompanying material with this episode anyway.

So concerning places, people should go to find out more. Now, in very brief, the final question of any of these episodes here is, what would be your key takeaways that you want listeners to remember? We usually do three, so you're three people, one each. Who wants to start? Maybe Jürgen, you have been silent for a while.

Juergen Walter: Pallavi was so excited to share everything, and otherwise I would have been getting in the same mood. So the key takeaway, SAP is running a substantial amount of OpenSearch clusters for observability. We're really big, we're petabyte scale, and we're standing on the shoulder of giants.

Karsten Hohage: All right, Karsten.

Karsten Schnitter: Yeah, so adding to that, so SAP is a really active member in the OpenSearch community, sharing our insights, contributing code and other content to the project, answering questions, pointing out documentation issues, and we just invite everybody to join us.

Karsten Hohage: Okay, great. And then key takeaway, Pallavi.

Pallavi Priyadarshini: Yeah, we are all really excited about open search and in general about open source and different organizations wanting to standardize on a common open source platform for observability, search and analytics and build on top of that solid foundation. Rather than building everything on your own, why don't we benefit from what the community has built and then build the functionality you don't see and contribute it back? That's the open source way and that's also the open search way. And I truly hope that you all will join us as a user, as a contributor in any capacity.

Karsten Hohage: Okay, and I hope Disney won't sue me if I say this is the way. Thank you, Pallavi. Thank you, Jürgen. Thank you, Karsten, for being our guests today.

Juergen Walter: Thanks for having us.

Pallavi Priyadarshini: Thank you. Thank you for having us.

Karsten Hohage: And thank you all out there for listening to The Open Source Way. If you enjoyed this episode, please share it. Don't miss the next one. We're on a quite irregular schedule by now, but it's often Wednesdays that we publish. You'll find us, and you're well-subscribed, of course, and you'll find us on SAP.com podcasts and also in all of the players that you usually use be that apple spotify open source podcast clients whatever thanks again and bye-bye.