The Open Source Way
Episode 01 - Open Source at SAP

Karsten Hohage: Welcome to The Open Source Way. This is SAP's podcast series in which we'll talk about the difference that open source can make. In each episode we'll talk to a different expert and we'll talk to them about why they do it the open source way. I'm your host Karsten Hohage and in this episode I will be talking to Peter Giese from the SAP Open Source Program Office. We will listen to why open source is important, especially in a global enterprise, why it's a triple-win situation and what an Open Source Program Office actually does. Peter Giese was working as the Head of Central Architecture team before he focused on open source mainly. Before SAP he has worked as a researcher for the Fraunhofer Institute and now he has been leading the Open Source Program Office at SAP since 2018. The company itself, he joined in 1996 already. That's 24 years at SAP. Welcome, Peter.

Peter Giese: Welcome. Good morning. Wow, 24 years sounds like an awful lot of time. But at SAP it's always changing and always exciting to work on new projects.

Karsten Hohage: Yeah, and it's especially in this interesting time right now. I am personally very happy again that I'm working for SAP, really.

Peter Giese: Yeah, absolutely, I think we are at SAP really in a fortunate situation. Our business is more or less completely digital, so we can continue to run our business and to develop our software to serve our customers without more or less any disruption or interruption. And this is, of course, not the case for a lot of other industries. So we are really in the fortunate position. And working from home is also something that we are used to, video conferencing that works fine. But of course, we also have our challenges, just like everybody else when it comes to home schooling of children or we have some new employees who joined the company and our team during the lockdown period. And for them, of course, it's much harder to establish the network and get into the company just on a remote working basis.

Karsten Hohage: So you are still working remote from your home office, Peter, is that right?

Peter Giese: Yes, I am.

Karsten Hohage: We have gotten some publicity or some press at least for the Corona-Warn-App that we were a part of. Is that something where SAP...I don't know, kind of got out of the closet as an open source player, really?

Peter Giese: Definitely, I think, of course, we are engaged in open source already for a long time, actually, since 1998 when we started to port our R3 system to Linux, which helped a lot in establishing Linux in the enterprise to get it enterprise-ready and accepted in the enterprise. But a lot of people, the general public, wasn't really aware of SAP's open source engagement and especially not in the consumer space. And with this Corona-Warn-App, which is now installed on or downloaded more than 18 million times in Germany of course, somehow, we appeared on the radar screen of a lot of new people who weren't really in touch or interested in SAP before. And that was a great opportunity. And of course, we did not do this alone. We did it together with German Telekom, with Robert Koch Institute, with Fraunhofer Institute and others and it was really great for everybody working on this project. And the project, by the way, is still going on to be able to help maybe a little bit in mitigating this crisis and fighting the virus.

Karsten Hohage: So what you're saying is now SAP has become visible, but even though we're a proprietary software vendor, usually, and for that the world knows us, SAP has had a part in open source for long already. Is that right?

Peter Giese: Yeah, absolutely, as I mentioned in 1998, it all started. In 2004, for instance, we have been the founding member of the Eclipse Foundation. We donated some of our projects to Eclipse and we used Eclipse IDE also as our main IDE at that time. And if you think about it 2004, that was the time when Facebook was created, right? So, we are really a long time in open source business.

Karsten Hohage: Well, it's like when you just said 2004 is the time when Facebook was created. I mean, I'm also old enough that 2004 doesn't seem to be that long ago. But when I think about, okay, that's 16 years. So, we've been a player for, what, 22 years in open source at least you just mentioned 1998, right?

Peter Giese: Yeah and 1998 by the way was also the year when the term 'open source' was officially coined. The concept of free software was existing before, but so to say

from the year onwards when the term was coined, we were also at SAP continuously engaged in consuming open source, but also heavily in contributing.

Karsten Hohage: Was it only that the term was coined, or the principal was defined in 1998? I mean, I remember there was free software on shareware before and everything.

Peter Giese: Exactly. So free software - the idea was already there before. If you think about the history of Unix that was already working in large parts to, at least on the university side, to open source principles, but yeah, the term and the way open source licenses are treated. That started more at that time.

Karsten Hohage: OK. OK. And we as a proprietary vendor, as I said before, we, of course, have special considerations when we deal with a concept like open source. Are these considerations the main reason why we need an Open Source Program Office or why does one need an Open Source Program Office?

Peter Giese: Ok, somehow you now raised two questions. Why is SAP, that is known more for its proprietary software products, an open source player - that would be the one question. And the other is, why do we need an Open Source Program Office, right?

Karsten Hohage: It's actually more about the differences of processes between proprietary and open source and how that leads to needing an Open Source Program Office.

Peter Giese: Yeah, maybe I start with this proprietary versus open source question. So, SAP is, of course, selling business software but we are not doing this in a vacuum. We are running on run-time stacks. And for instance, now that we are a cloud company, we use a Cloud native stack, which has normally Linux as an operating system below. Then we have something like when it is container management on top and we even have our own Kubernetes cluster management solutions, our open source project Gardener, then on top you have things like path platforms, Cloud Foundry or Knative, especially now. And we also have our own contribution to that with our Kyma project. And then on top you have application programming frameworks, UI frameworks, where we also have, for instance, Luigi and Fundamentals as open source projects, which we published. So

even though our business solutions are proprietary, they are based on a lot of open source and we are also contributing to open source on lower layers, because it's much better to collaborate with the community and to share the development load for work class solutions on the space than doing everything on our own. And that leads me to the question why SAP or other companies need an Open Source Program Office. Somehow the usage of open source is growing exponentially, for instance at SAP the number of open source components that we are using in our software has more or less doubled from the year 2018 to 2019. And such a dramatic growth also somehow requires that you reduce manual steps, for instance, when it comes to license or security scanning. And that means somehow that you have to automate more of the open source management tasks that you are doing, if you consume a lot of open source components. And we also want to contribute bug fixes to these components or features that we are requiring, we want to contribute to the open source community. That also means that our open source contribution, so-called outbound processes, have to be streamlined and should be simple and not too bureaucratic. And in the end, as you said, we are also developing proprietary software, but still also for those developments we can benefit from open source principles, just applying them in-house. So that different teams at SAP can contribute also to our proprietary projects in-house, in an open source fashion. This is what is called InnerSource. And for instance, one of the three major tasks of the Open Source Program Office at SAP is to support the InnerSource movement and to educate teams about it, to help them to grow and nurture InnerSource at SAP.

Karsten Hohage: So, as a short summary, basically the program office first records needs, defines processes and then supports in the execution of the processes. It is not so much dealing with the technology itself, right?

Peter Giese: In the end, the Open Source Program Office is about managing open source in the enterprise. That comes in three flavors. That comes in our contributions to open source, that's what we call Outbound Open Source. That regards consumption of open source, that's what we call Inbound Open Source. And that's about applying open source methodologies also for our proprietary in-house development, that's what we call InnerSource. And in the past, when companies started to work on these three areas of open source, it was a lot about risk management. And that's why, if we are honest, it was a little bit about policing. If people wanted to use an open source component, they

had to fill out a lengthy document and then they needed approvals from different instances. Or if they wanted to contribute to open source, they needed approvals as well. A certain level of approvals is still required today, but the change that the OSPO did was we wanted to turn the whole thing from policing our developers to supporting our developers. Now it's much more about empowering them, giving them guidelines of what they have to consider if they want to consume open source or contribute to open source, but less about lengthy approval processes. So, we have very much streamlined all of the processes. Instead of weeks or months, it only takes a few days or weeks to get, for instance, approval to publish a project as open source. And instead we see us, in OSPO, more as a supporting entity where we try to coach them, to give consulting, to help them in reaching their goals by using or contributing to Open Source.

Karsten Hohage: Now, that was a lot about SAP developers contributing to open source projects in the Open Source Program Office. It's also your job the other way around, right? Defining the processes for the usage of open source or is that somewhat different?

Peter Giese: Yeah, that's what I called Inbound Open Source.

Karsten Hohage: OK, OK.

Peter Giese: Go ahead.

Karsten Hohage: Now, only because you talked about the processes for the outbound pretty much now and I seem to remember there is often equally much legal and process considerations when it comes to using open source, due to heterogeneous license models and so on, right?

Peter Giese: Absolutely, most of the companies even start on the consumption side, so they are first consuming open source and normally of the form that they are using some open source components to sell their own solutions. And whenever you use open source software, then you have to make sure that you are compliant to the according open source licenses. So that's always a topic about license compliance. You have to make sure that the open source components you are using are not containing known vulnerabilities. That's why you're doing security scans. That's the topic of security. And

also, of course, you have to be able to operate those open source components as part of your solution. And those three areas are also typical tasks of OSPO, to automate as much as possible of that and to also provide guidelines for how to do that.

Karsten Hohage: There was just a question forming in my mind, that is because we talked about processes and defining processes and also about technical aspects. The question that formed in my mind was, as the head of the Open Source Program Office, do you have more interactions with management and the legal department or more interactions with actual developers?

Peter Giese: Yes, that's actually a very good question, and I would say it depends a little bit on where you are within the lifecycle of your Open Source Program Office. When you establish a new Open Source Program Office, then typically you are dealing with the basic tasks of managing open source. That means managing the consumption. For that you need processes and tools for license scanning, for security scanning and things like that, and also for contributing to open source, you need guidelines or at least some kind of approval form for new outbound projects. And that's where, of course, you are more interacting, probably with management, with legal teams, with tools teams. And then once you have done your basic homework of this phase one of an OSPO so to say, then normally you start much more in contributing to open source, starting your own open source projects. And once a company is doing that, then of course the whole company, with these outbound open source projects, is more engaged with the communities and with development and also the OSPO then normally is more engaged with the developers driving these open source projects and contributions and also communities.

Karsten Hohage: And of course, the obvious follow-up question is, which phase are we in at SAP?

Peter Giese: As always in life, there are iterations. So, the whole open source management is more or less a task of continuous improvement. So, also on the tooling and processors side, we try to continuously simplify things to improve them, to improve the degree of automation. But I think we are also very much in phase two. So actually, in 2019, according to the Open Source Contributor Index, SAP was the 9th largest

commercial contributor to open source in GitHub worldwide, which I think is a significant sign of our contribution level.

Karsten Hohage: That sounds a lot higher in the ranks than most people would probably put us by the sheer name SAP, right?

Peter Giese: Yeah, absolutely. The TODO group, which is a group inside of the Linux foundation where Open Source Program Offices are collaborating, also we from SAP are collaborating, they did a survey in, I think, 2019 where they asked like 2.700 companies how they perceive different open source players, as a good open source instance or not. And there was also a question about how SAP is perceived and if I remember correctly, more than 50 or around about 50 percent of the respondents said they were not aware of the fact that SAP is engaged in open source at all, which is a strong contrast to our being..

Karsten Hohage: Rank 9 of contributors.

Peter Giese: Exactly.

Karsten Hohage: Can you just name some examples of who's on places 1 to 8?

Peter Giese: I don't have the complete ranking order in my head, but I think on the top you have definitely Google and Microsoft, RedHat probably, Facebook, I think is...

Karsten Hohage: Like, basically, that's as expected. The unexpected thing comes on rank 9, when all of a sudden you have: wow, SAP.

Peter Giese: You could say so, yeah.

Karsten Hohage: Yeah, and is that basically what we just talked about, would that already define why one needs or why one should have an Open Source Program Office? Basically, you need definition of processes and then support of processes? Or what else would you add to the role of and the importance of an Open Source Program Office?

Peter Giese: Indeed, on the one side, with our simplified contribution processes, I think we have supported our developers that they have an easier time to contribute to open source and that's why we see us climbing the ranks. I think right now, for the last month, there is also a monthly ranking, we have even been on place number 8, so to say. But I don't want to talk about these places. It's not about the ranking, but it also shows, and that is also part of our OSPO mission, that we have to do a better job in communicating about what we do in open source to our customers, to the open source communities so that they know about it. And of course, we want to invite them to also collaborate with us on our open source project.

Karsten Hohage: So that's an additional role? You also take care of rolling out SAP's open source activities and make them known?

Peter Giese: Exactly. You could call this open source communication and the podcast series that the two of us are just starting with this very first episode, which I'm very excited about, that is, so to say, also a part of our endeavor to try to better communicate about what we do in open source.

Karsten Hohage: Absolutely! Did you just - between the lines - say to stop asking me stupid questions? Of course, we're rolling it out because we're making this podcast here, Peter?

Peter Giese: No, no, I didn't. I wanted to explain it to the audience. Maybe they wonder why the heck SAP is now starting a podcast series.

Karsten Hohage: Just kidding. But maybe we can get to even more detail. Now, we added to the process definition and the process support, we added the roll out. How is the Open Source Program Office internally organized? What kind of people can I find there that will do what for me? Or will ask what of me?

Peter Giese: Yeah, indeed, that's an interesting question. We are organized as a virtual team consisting of mainly 10 other SAP teams, which are spread across different board areas. And I think that's also important. You need the close cooperation with colleagues from legal, with colleagues from licensing, with colleagues from global IP, with colleagues from tooling, security. So we need these diverse teams and the diverse

backgrounds of everybody. That's why we have chosen this form of a virtual team and I'm leading the OSPO core team, we call it. That's just the team of 7 colleagues. And we try to orchestrate the other 10 teams and the working mode that we have chosen is close to what most of our development teams at SAP are doing. So we are working in scrum mode with four-week sprints. That means we have a backlog where we collect all the topics that we want to work on and that we want to improve in the company. So we have several hundred items on our backlog. And of course, at any point in time, you can only work on a certain number of these points. That's why we have planning meetings every four weeks where we look at the highest prioritized backlog items and decide which one of them we want to tackle within the next four weeks in our cross-scrum teams. And then after the four weeks, we do a review meeting to see what we have achieved and to start the next planning round for the next four-week cycle. So that is very close to what our development teams are doing as well. And then we have organized ourselves into the 6 workstreams. One is on processes, one is on tools, one is on strategy, one is on policy, and one is on communication and one is on training. And in that way, we also have like 6 cross-functional scrum teams working on these according points.

Karsten Hohage: And these are all virtual, put together from different teams, always only with a representative from the program office in the project, or do I understand correctly?

Peter Giese: We tried to, of course, put more or less 100 percent of our time into the different workstreams because that's our main mission. Some of the other teams are not only working on open source topics, for instance, in legal, of course, they are also working on a lot of other topics. So that's why they cannot put all their forces just only to the open source topic. So every team contributes as much as they can. And that also varies over the year, depending on the workload they might have for other tasks.

Karsten Hohage: I mean, sometimes that's difficult with virtual teams were not everyone is in the same line. Does that largely work nicely, or do you have an example of where that works the very best and where it could work better, possibly?

Peter Giese: Yeah, in general, it works pretty well. But of course, sometimes different teams have other obligations, or they might have higher priorities. And then it's also

sometimes a challenge, of course. I think it always works very nice when we need cross collaboration with different expertise to reach a goal. For instance, right now we are working on a new version of SAP Open Source policy, which is binding company-wide for every employee, every developer. And to work on such a policy, which also then has to be approved by the SAP executive board, by the different social partners and the countries and the different development departments, it really helps to have cross-functional teams, experts from licensing, from IP, from legal, from Open Source, from development.

Karsten Hohage: But that policy now is something that's an example of "everyone has a vital interest in that", and so it works pretty well, even in the in the virtual set up, right?

Peter Giese: It's not finished yet, but so far, we're confident.

Karsten Hohage: That's kind of like a 360-degree policy? Or is that for a contribution or for use of inbound, outbound?

Peter Giese: It covers consumption and contribution, yeah.

Karsten Hohage: Ok, so 360-degree open source policy, all new. Ok, that could take longer than three days sometimes.

Peter Giese: And of course, all major companies have such a policy. We also have one, but it has to be updated from time to time and for instance, if you want to get an open chain certification, then such a policy is one of the requirements and of course, it's also the task of an OSPO to work on these things. Open chain certification, basically, documents that you are in compliance with correct consumption of open source when it comes to licenses. And that's, for instance, also the task of an OSPO to do the safe certification for open chain so that if we supply software to somebody else, they can rely on the fact that we are 100 percent compliant to all the licenses that are included in our bill of material for open source components.

Karsten Hohage: Sounds important, especially for, as I said and as we know, a usually proprietary vendor of software where there is probably a larger damage when we do not comply to such things, right?

Peter Giese: That would be, in general, a bad thing to not comply, and especially in industries like in the car industry, where there are suppliers and basically every intelligent part nowadays also contains software and they deliver the whole car to their customers. For them, of course, it's important that not only the software that BMW has written themselves, but also all the software parts that they got from their suppliers are fully compliant.

Karsten Hohage: Are you saying that even in my car, there are open source components?

Peter Giese: Oh, absolutely. In open source, we have the so-called legal notices. So, if somebody uses an open source component, they have to give credit to the author, so to say, and to document that fact. And in some of the books that come with your BMW or your Daimler, you will find such a list. And it's lengthy, really lengthy.

Karsten Hohage: Ok, that's because Linux is used as an operating system or all kinds of different things?

Peter Giese: All kinds of different things, even on your iPhone somewhere, if you go to general and then this legal stuff, you will find all the legal notices and if you print it out, I think it's a PDF document of over 150 pages or something like that.
.
Karsten Hohage: Ok, so that would then be the things that the Open Source Program Office at Apple had to take care of.

Peter Giese: Absolutely.

Karsten Hohage: Yeah. Is that one of the things, like one of the very dry day-to-day tasks? Putting together the list of used open source components? Is the Open Source Program Office involved in that or is that the job of the development department?

Peter Giese: Yeah, at SAP for every software that we build and deliver to our customers, of course, we have a bill of material describing what parts the software consists of and that contains our own written parts, that might contain commercial

software that we have licensed from third parties, and that also contains the open source components and of course, the job of the OSPO is to make sure that the open source components are correctly listed in our bill of material, that the licenses are conformant to what we at SAP are allowing as licenses. So, we are not allowing to use arbitrary components for some arbitrary types of licenses and software licenses, or open source software licenses also come with obligations sometimes. So if you use an open source component, you might have the obligations, for instance, to provide the source code on request to the people you are delivering a software to and in these kinds of occasions, of course, SAP has to make sure that we adhere to all of them. These are typical tasks of an OSPO to ensure and to create the according link to ideally automated or at least to create guidelines is mandatory.

Karsten Hohage: The open source specific tooling integrates then with the inherited tools that development has anyway to maintain their components that they use, right?

Peter Giese: Absolutely, in an SAP case, we use a proprietary tool called PPMS for our software bill of material and we have to integrate our software scanning tools, for instance, with that to automatically include the open source components in our bill of material.

Karsten Hohage: Actually, I knew that, of course. I've worked with that in my product management times with PPMS and I think even with some open source components like 15 years back or something. With some of the HANA predecessors. What else do I want to ask you, Peter? We actually talked quite a bit about SAP's perception as an open source player out there already. Is there anything? Are we are focusing a little stronger on being part of open source events than we did in the past? I think there was a little difficulty when I think like 5, 10 years back. Are we putting some more emphasis on that?

Peter Giese: As part of our communication plan, we from the SAP OSPO try to be, to participate in open source events more than we did in the past.

Karsten Hohage: Mm hmm.

Peter Giese: If we do so, then we normally participate in the OSPO related tracks of these events and talk about best practices or tooling, but we also encourage our development teams and our open source projects, like the colleagues from our Kyma project or our Gardener project, Open JDK project or Luigi or Fundamentals, whatever, you name it, that ideally, they also present at open source conferences.

Karsten Hohage: What's the most perceived of these projects these days anyway?

Peter Giese: You mean in the sense of most popular?

Karsten Hohage: Most popular, most non-SAP contributions, I don't know. There are probably different indicators by which one could meter that or judge that.

Peter Giese: It always depends a little bit on how you look at it. So, we have a known implementation of Open JDK, which we call SapMachine, and in certain releases of Open JDK, we have been the 2nd largest contributor behind Oracle. So, in such communities, this is probably known. On the Kubernetes side we have the Gardener project for managing Kubernetes clusters, large scale. That is probably known quite well in that community. Yeah, but maybe that's a little bit an explanation of why SAP is not so well known yet for open source in comparison to some other companies. Normally we are not developing applications for the consumer space, right. And that's why we don't have open source projects at SAP, which are very popular with normal end users or consumers so to say. And that is maybe one explanation why we are not so well known.

Karsten Hohage: Because we're always hidden. It's kind of like when I asked you, you mean there is actually open source in my car. It's not like I noticed first-hand that there is open source in my car. I'd have to look into the disclaimers or the whatever booklet you just mentioned before.

Peter Giese: Exactly. If you take the example of Kubernetes, a lot of people maybe know Kubernetes for container management, but who really has the challenge to spin up 50.000 Kubernetes clusters, right? That is only a handful of companies who require such technology, and that's what we do with our Gardener project. But, of course, then the target group for such tasks is a little bit limited, I would say.

Karsten Hohage: Yep. I think about most of the specific projects that you named, Gardener, Kyma, also the Corona App, we will probably hear in future episodes of this podcast. I hope so, at least.

Peter Giese: That's the plan.

Karsten Hohage: Open Source Program Office says no, you're not allowed to talk about this in the podcast, but I don't think you will. Why does Open Source pay off for a company like SAP?

Peter Giese: If it's done right, I would say it's not only paying off for SAP, but it could pay off for every company.

Karsten Hohage: Mm hmm.

Peter Giese: Of course, the trick is to always find the right mix. And if I would generalize, I would say most companies would probably not share with the public the parts of their solutions that are core to their solutions as open source but they more share what is not mission-critical for themselves.

Karsten Hohage: Mm hmm.

Peter Giese: So, at SAP, for instance, we are not in the business of selling cloud infrastructure solutions or selling Kubernetes as a service or so. We are in the business of creating intelligent enterprise and business processes. That's what we sell. And of course, it makes sense for us to share what we do on the lower layers of the technology stack. So we benefit on the one side from the community, from the work they do, for instance on Kubernetes. And we also give back to the community what we do, for instance, on the cluster management side or on the Kyma side with respect to integration of different systems and providing something like Function as a Service with Knative. And yeah, that way I think for every company they have to find, of course, their own way of how to combine their commercial interests with open source. There are companies like RedHat which are completely open source. Everything they do is open source. They have found a different way but there not too many companies doing that.

Often times companies are running open source in the cloud as a managed service and then they don't sell the licence to run your software locally but they basically sell a subscription to their service and then open source is a nice way of building it and getting contributions from others.

Peter Giese: And some companies have a model that they run their project as open source but they have some special features which are especially relevant for enterprises and sometimes are not open source. Then they sell these enterprise versions of their software to enterprise customers. That's a different way. But as I said, for SAP, I think it makes a lot of sense for us to work on open source, whenever it's not really core to our core business. And we also have a long tradition of co-innovation with our customers and partners. I mean, if you really think about it, it was with our R3 systems, the ABAP source code was not open source but it was always visible to our customers and partners and they had the chance to read it, understand it and modify it with extensions. That wasn't open source, but it was shared source. And that's why I would say SAP has a long tradition in working in such mode where we can go innovate with our customers.

Karsten Hohage: It always had one of the characteristics that also open source has, right?

Peter Giese: That's one of the things which make open source so successful that you can look at the code, modify it, extend it and even though we did it under a different license, I think we have a long tradition in doing so. And we have really learned that this is the right way to cooperate with your customers and your partners.

Karsten Hohage: So I got you right, you basically named all three sides that you say have a benefit. That's SAP, that's the community, and that's the customers and partners, who benefit from, always have benefited from, the shared code and now are benefiting even from other effects of open source software. So it's a triple-win, basically.

Peter Giese: Maybe, and of course, there are also the developers as a group which we can talk a little bit about and also if you talk about customers and partners. If you look at it from their point of view, and then, of course, every customer would like to avoid a vendor lock-in, they also have to hire developers to build what is specific for their

solution or to build extensions. And for them, if a solution is a large part based on open source, then they have a chance to get support for those open source parts, not only from the direct vendor, but also from other sides, from other vendors, from the community. If large parts are built on open source, then they have a better chance to attract developer talents to their own company because normally, if people come from the university or from somewhere else, they prefer to work on open source. The knowledge that they build up on open source is independent of a single company. So a developer with open source can kind of also avoid a company lock-in. Right.

Karsten Hohage: Mhmm.

Peter Giese: Because he is not acquiring proprietary knowledge, but knowledge that is of general validity also in other companies. And yeah, that's one of the advantages that developers have. And of course, open source also allows developers to work with world-class experts on specific topics. If you take the topic of Kubernetes cluster management, then our developers are able to work with the specialists worldwide on the topic because we do it as open source out in the open. And that is, of course, also raising the engineering level on a lot of sites because there's a constant exchange of best practices and of world class experts and...

Karsten Hohage: And is that really pretty open? Is it really a "each one, teach one" attitude or is there still some, I don't know, "This is mine and this is yours" thing in these kind of projects like Kyma, for instance?

Peter Giese: I mean, the fact that we publish the source code, first of all means, of course, everybody has a chance to look at it and read it. That's why it's very democratic. And whenever somebody has an idea that something should be done better, he is free to create a pull request. And then we have the concept of peer review. There will be reviews by others and that way, with a pull-request and the peer reviews, normally I would say it's like Darwinism. The best solution wins. Sometimes there might be a little bit of politics or egos involved, of course, but we're all human beings. In general, I think that's the most successful model of coming to the best and most technically solid solutions.

Karsten Hohage: The last part was what I was getting at. Is it more or not so much on the organizational side, but on the individual people side? Do you feel that this is really a very cooperative environment that open source is creating? And emotionally, does that work?

Peter Giese: Yeah, absolutely. I mean, for most of the developers that have started working on open source, they never want to go back, so to say.

Karsten Hohage: OK.

Peter Giese: They do it sometimes because of egos or politics, they might leave a certain project because they might be frustrated with how things are handled in the specific project. But the general concept, normally, once people are infected with that open source virus – like I said, once they come, they never turn back.

Karsten Hohage: Sorry, it always seemed like that to me, but I don't hang around open source forums too much, so that's why I wanted to find out.

Peter Giese: Yeah, I'm not sorry to generalize here, but that's at least my personal impression. And if you look at major areas of technology like machine learning or the cloud native computing, the whole stack, IoT, Blockchain, cryptocurrency. Basically, all of these topics are, I would say, to 99% driven by open source nowadays. So this speaks for itself.

Karsten Hohage: Yeah.

Peter Giese: A single company, even a company the size of Microsoft or Google, has a hard time nowadays to come up with better proprietary solutions than the world-wide open source community.

Karsten Hohage: Ok, sounds cool. When I just said, sounds cool, let's maybe get to the key takeaways of the day that we have. I think we talked about most things that we can talk about around the Open Source Program Office or is there any field, any topic that you actively want to share, Peter?

Peter Giese: I would say if you take one thing away about the Open Source Program Office from this podcast, then I would say the SAP Open Source Program Office is here to nurture, grow and support on the open source side instead of policing. So internally and externally, whenever you have an inquiry or you need help for open source related to SAP, please approach us. And yeah, I think we definitely want to enter the next phase of having more interactions with the world-wide open source community, even more than in the past. And we invite everybody to also have a look at our open source projects and to contribute to them. That's basically what I would like people to take away from this.

Karsten Hohage: You have already said your key takeaway is to take away that the Open Source Program Office is there to help you, externally and internally, in all means that we have talked about. So, Peter, if unless there is anything else to say, I want to say thank you for being here today. It was great talking to you. Thank you, everyone, for listening in. I hope you will do so again. If you learned about this podcast through openSAP, then be aware that this is a regular podcast that you can also find on Apple Podcast, Google Podcasts, whatsoever – just what your usual way of accessing podcasts is. So it's not only accessible through the openSAP page. If you enjoyed this session, please tell everyone. Share the link. Subscribe to the podcast. Hope to listen to you again. Thank you again, Peter. And we'll hear again in two weeks from now. Thank you.

Peter Giese: Thank you. Bye bye.

Karsten Hohage: Bye bye.