PUBLIC

Podcast Inside SAP S/4HANA - Transcription

Inside SAP S/4HANA Episode 100:

Clean Core: Achieving Modern, Flexible and Cloud-Compliant ERP Implementation feat. Accenture

Yannick: Welcome to our podcast Inside SAP S/4HANA. There is no customer success without product success and project success. I am Yannick Peterschmitt, your host for this podcast episode and we will talk about Clean Core, a concept and an implementation approach to achieve a modern, flexible and cloud compliant ERP implementation with SAP S/4HANA. Today, as guests, we have Stefan Batzdorf our Chief Technology Officer for SAP S/4HANA within SAP and Dirk Appelhoff, EMEA SAP Business Group lead from our partner Accenture. Guys, warm welcome to this episode.

Stefan: Welcome Yannick, hi.

Dirk: Welcome. Thank you, Yannick. And hello, Stefan. Thanks for having me here. I'm pretty excited. It's a great topic to discuss.

Yannick: And so am I, to be honest, before we talk about the topic itself, I'd like to introduce yourself, guys. Maybe you'd like to tell something to our audience that people don't know yet about yourself, so it gets remembered.

Stefan: Yeah. Stefan Batzdorf my name. And 22 years with SAP and in the S/4HANA space since 2013, since its inception. Started as developer SAP former program lead of SAP S/4HANA also in development. And now I'm back in development as a chief technology officer for cloud ERP. Private points. I love bicycles, everything. So I love mountain bikes. Uh, gravel bikes, race bikes, especially doing it, conquering the mountains. But I also loved restoring bikes. I restored for old vintage bikes and, for some people that have meetings or customers that have meetings with me in my home office, one of this bike is also hanging behind me on the wall.

Yannick: Thanks, Stefan. And lucky me, I can see the bike.

Dirk: That looks pretty awesome. So let me share a little bit. So my name is Dirk Appelhoff. I'm in the consulting business related to SAP for decades I would say so I actually started with R2. Believe it or not, I'm really passionate about that topic. But in private there's two things which stuck out. I would say one is music and the other one is sailing. So there's two things I want to share. On the sailing side, when I was young, I actually participated in the round the world race, actually in one leg. So I sailed from Japan to Hong Kong, which was really an adventure for me at that point in time. Pretty exciting. And then on the music side, I actually used the Corona time to get back to music myself. So I played the violin when I was young, but I always wanted to start playing the piano, so I actually took lessons and

educated myself on piano since the corona time and advancing more slowly than when I was young. But I'm really enjoying it and is a good, relaxing moment after a long, long day of work.

Yannick: Very cool. Thanks to both of you for sharing those insights. Now back to the topic of the day. I'm sure that part of our audience is familiar or partly familiar with the concept of clean core, but of course not everyone. I think this concept of clean core as a target to achieve a modern, flexible, cloud compliant ERP. Stefan, what is clean core? Concretely, what is included and eventually what is it not?

Stefan: So for clean core, to describe clean core, the best case scenario describes having an up to date system on the latest release, with a cloud compliant extensions and integrations, as well as an optimal master data quality and process design. This sounds super, but why is it so important? Because the technology is changing so fast. And now even with AI, generative AI, supply chain resilience, etc., the faster the software is changing, the more often customers need to adopt these software. And in order to be flexible, to stay on the latest release and to consume the upcoming innovations that any software vendor is delivering, you need to put yourself in a place to upgrade your systems more frequently, and this can only be, or more easily be achieved if you reduce your system complexity. If you reduce your customizations and stay close to a standard. If you build your product extensions in a way that decouple them from the software code. So providing creating extensions using only public APIs, cleaning up your master data and your datas and your records, doing archiving to reduce the system's size and using cloud applications, for example, because at least at SAP, SAP has a cloud strategy and the future innovations will mostly only be built in the cloud. For example, on the business technology platform. It's not an on premise only world anymore. And the more you foster to have a clean core using cloud native integrations, for example, the easier it is to consume side by side innovations like, for example, generative AI use cases to automate your processes.

Dirk: I couldn't agree more, Stefan. And let me chime in here on Digital Core. We are using kind of a picture to make it more grabbable, but the key point you mentioned is agility and flexibility. You want to have faster time to market in terms of innovation, and you want to have a more direct investment in differentiating capabilities rather than spending, you know, on a broad level into your complex system landscape. When we talk about digital core, we always use this picture of four layers. And I think that's quite helpful. So we say the digital core is SAP comprises four layers. And the initial layer is the cloud infrastructure and the security like you mentioned it. So you need to have the scalability and the resilience in that

layer. The next layer actually is the foundational enterprise system, which in our case is the software as a service, a very cost efficient operation system you need. And then you have the composable technologies who are connecting those applications and enabling the collaboration. The data topic you mentioned comes into play, and you want to have interoperable solutions, as we say. And then finally, you have connected solutions. That's the agility and flexibility on top, where you have your innovation, where you can apply GenAI quickly, where you can take the innovations from the market faster and more quickly and more distinct in terms of where you need them. And those four layers in our terms, define a digital core.

Yannick: So I hear you guys. Speed of innovation, adoption of innovation, agility, flexibility. Now of course we're talking to customers that maybe are thinking about the next upgrade going to the to S4, right. And Dirk like you mentioned, you started with R2. My first upgrade was with R3 three OB. I remember that many customers really had not a fear, but we all know that it was always very disruptive with a very thorough project plan, a very thorough upgrade plan, etc., etc. and my take from what you've said, guys, is that, uh, today and especially for the future, with the speed of innovation and the time to market, the customer are not willing to have this disruptive event. Correct? But most of the upgrade come with manual efforts. So what else do they need to consider as a challenge when they approach a migration and upgrade planning?

Dirk: Very good question, Yannick. Definitely. It's always easy to talk about it in the way we are doing it more from a managerial perspective than doing it actually. But there's obviously solutions for some of the challenges. So let me go back to the importance and you mentioned some points, but I just want to rephrase them. So all our clients need to get into a more agile and flexible architecture to respond to the changing market circumstances. So it's all about agility and flexibility. They're all asking for, how can I do my investments more in the really differentiating areas, in the innovation parts, rather than spending a lot of money in the areas where I look for very low total cost of ownership, that's where we go for standardization. And then the third point is the overall lowering of the operating costs. So if we take that, we can easily translate that in or relate that to some of the challenges you mentioned, some when you teed up the question. Our clients have the burden of the complex legacy. They have built and invested in capabilities, and they have an excellence in those, and they also have a certain pride in those capabilities, untangling that landscape and really, quote unquote, cleaning it and going back to standard and making the decision which is differentiating in the industry, in the market and which is not differentiating.

Dirk: This is one of the biggest challenges we have in the dialogs. So that costs and needs a lot of attention. The second challenge I would point out is something Stefan already mentioned also is the existing data structure. If you look at the legacy systems, if you look at this web of applications they are sitting on, that means you have a siloed data structure. And if you want to really gain the benefits of a digital core, then you need to re-architect your data structure. So that needs attention and work. And then last but not least, there's a big component of change and resilience. As I said, all of those clients have a certain excellence in their capabilities, and they have built capabilities. They have experts around and asking those to go away from something which is proven and working and gives them their market position right now, and moving in something which is not crystal clear. This is a big, big topic for programs to be successful. So change and overcoming that resilience, I think, is really one of the most important challenges to address.

Yannick: Great. I like a lot when you said they have a certain pride in this legacy, because this is where they build the excellence and as well where the differentiator from competitors eventually. And of course, this is where you build the comfort zone. Stefan, we all know that the big strength of R3, of ECC is the ability to customize as much as you want. And I'm like to look at the topic of the ABAP custom code from the customers. What is the future of this concept?

Stefan: That was the strength in the past of SAP that we shipped the core code, the ABAP code, with every ERP on premise release, and customers were heavily using it and also partners. They cloned the code, they modified it. They built additional code, add ons, partner add ons, customer add ons on top as all the code was available. And you for example, you have to build an own quotation engine and you're missing a data object or a business functions. You just searched in the ABAP workbench or in the data for objects. You found it and all of a sudden you created also a dependency, for example to pharmaceutical industry solution or oil and gas, etc. everybody did it also SAP. And with this we created a high detangled code. It's kind of spaghetti code with lots of references. Now the world is is different, it's fast changing and we build more flexible code, more modular code, customers and also partners, they look for more flexibility in the cloud for composable structures, also composable ERP structures, as Dirk said, you want to have a core running, running your core system, your financial system, your production systems, etc. but you want to have the flexibility to implement, supply chain solutions for warehouse insights, for track and trace functionality. Net business networks, etc.. So we want to become more modular. And for this you need to also rethink the way how you build your extensions.

Stefan: We have created now under S/4HANA a new extensibility structure, what we call the three tier extensibility model. So building cloud compliant extensions and modernizing your legacy custom code is the most important part of the clean core. And if you follow this so-called three tier extensibility model for private cloud and on premise helps a lot. So what is this three tier? It's not database application server UI. So tier one describes all cloud compliant code. So cloud compliant extensions and code that run in the public cloud private cloud and on premise and are super upgrade safe. Tier two describes public API wrappers around classical bodies where there is a workaround that we created now in case SAP is missing some public APIs and the tier three are all your legacy custom ABAP objects. This is currently your set coding that you have, and your goal should be under Clean Core to get rid of most of the tier three and move them to tier one. So tier one are all extensions that you create that follow the strict public cloud extensibility model, which means they only use public released APIs. They are built either inside S/4HANA using ABAP cloud, or they are built side by side on the BTP, connecting via a public OData API or a business event to the S/4HANA back end. So the goal is to review your custom code, rethink what you really need, move as much as possible into this so-called tier one. So build your extensions in the same way, like you have to do in the public cloud, because the public cloud is always clean code by default. Now during this exercise you will find obstacles and struggles. You will find APIs, public APIs that are missing. Why? Because in the old ABAP world, in the old R3 ECC world, there are more than 11,000 APIs bodies and Bapis and function models you can use. S/4HANA in the public cloud has I don't know don't nail me down, 6000 read APIs via CDs views and something around 1500 public OData APIs and business events. And this will increase over time. Now you will come to the point that somewhere in transportation management, you are missing a public API. And this is the reason this, what we call two tier SAP, is now defining nominating classical bodies where SAP says this is a body and you can use it as a public API. SAP won't change anything in the future, we will keep it stable. And then you use the ABAP cloud development tools. Codename embedded steampunk in S/4HANA, you create a wrapper around this body and have a public API, and this public API can be then used in your tier one coding. So this is a workaround, but it's also clean core that when you build these wrappers around stable bodies and then you have the tier three. This is really then that contains your custom legacy objects that customers converted from ECC to private cloud and to S/4HANA. And they do not comply with the Clean Core principles. For example, you use direct table access, direct file access, native SQL, etc. so getting moving everything into a writing completely, everything new doesn't also make sense. So the goal is first delete unused parts and delete as much as possible from your custom code that you have, renovating the extensions and moving as much as possible to tier one. And if you have 50,000 set objects, it's okay to have 20,000 left in tier three then

govern them, govern them, control them, let them run. Keep them as is. If they don't create pain during upgrade, it's okay if some points of your legacy custom code creates problems during upgrades, SPDD, comes up, etc. then you have to take an action. If not, leave it as it is. Clean core doesn't mean to rewrite all your code that you have. Clean core means control it, get it under control, and if someone in your IT department develops new extensions, they shall do it according to the tier one Public Cloud extensibility model, only using public APIs, etc. and not going to ABAP workbench, looking up in EdTech any kinds using generated code, etc. so it's a change in mindset.

Dirk: I like that very much. Stefan, I think it's important that when we discuss the topic, we are not saying that the legacy world is something less valuable or not. It's all about making conscious decisions and starting a journey. And I think you described it very nicely in terms of there's no need to move everything to new from day one to day two. We have some time, but it's important to make a conscious decision when to do it and when not to achieve your overall objective, which is this agility and flexibility you want to have and less dependencies. How do you stitch the modules together and the components together in a more agile way, to allow you to react more to the market circumstances? That's the key point, and it's all about a journey.

Yannick: Exactly a journey, maybe from what Stefan called the spaghetti code model to the cloud compliant extensibility model. And many of our customers are already on this journey, so Dirk I'd like you to please share concrete example of a recent project where this clean core concept and the journey that you've accompanied has really helped the customer to make a difference. How can a partner like Accenture concretely help there?

Dirk: I am picking an example where actually we work together with you guys in a large industrial client in South Germany, and it has been a quite a discussion coming back to my point about the journey, but I think we are now doing the right things after thinking through everything in joint workshops and sessions. So we have defined jointly SAP and Accenture with the client, the "north star", so to say, where we want to go. And then we looked at how we can go there. And of course we had all kinds of accelerators and tools and ideas and methodology. But the main point I want to emphasize here is that we took us a long time together to find a good governance to address the challenges. And this has started as a more, you know, technology IT driven initiative and now has evolved into a complete business initiative. And I think that's really important because while you define your clean core or digital core, you always want to look at the value you can create for the business side. So what's the business value for the investments you are doing? That's exactly why

we are emphasizing this standardization so much Stefan talked already about. So let's standardize wherever we have no differentiation in the market where we want to go for low total cost of ownership. And let's spend wisely the money in the areas where the business desperately looks for speed, time to market. Sometimes they want to try something and then after four months they go to a different solution. You can't do that with 20 year old architecture and solutions, it's impossible. In this client, I think we have a very, very robust governance now, which is led by the businesses very much in collaboration with the IT. Accenture obviously helps with a lot of, as I said, methodology and tracking and, you know, running the program as we speak. But at the same time, of course, with expertise in specific functional areas, with deep industry understanding to help with the business cases. So you need to you need to really start from the strategy down in the business model and how you want to steer and operate your business before you go into technology. So I think the benefit of us in this circumstances and in general in the market is that we have an expert pool, which really ranges from strategy consulting to management consulting to real in-depth technology consulting. Implementation skills. When I talked about the digital core and the four layers, we can bring in infrastructure and security people, which is really important if you if you want to make sure that your future architecture is secure, but it's also robust, it's completely different from what what the client has today. Today they are sitting still on data centers and old architecture. So it's a really leapfrog. And you need to make sure that you have the expertise. And that's why it's important that we partner with with you guys in the market because we can combine our capabilities. You have so much expertise in the solution itself, but also in the infrastructure and in the technology space, which is really, really valuable for the client. So I think bringing that together and having an ecosystem which is helping the client to achieve the objective is important, and we are doing that. It's still the beginning of the journey, I have to admit. I mean, we are not done, but it's a it's a big, complex thing. And I think that this is one of the lighthouse, activities we have in the market right now in our region.

Yannick: I think you said the key thing here. The governance is led by the business. You don't go for IT driven decision. IT is the enabler obviously, or your business case driven. And Stefan have the questions when it comes to extensibility and business case decisions. Where do you what I've heard many times that some colleagues even within SAP but of course outside too understand that when you want to build extension you go by default on BTP. I understood from you already there are more options. You mentioned embedded steampunk, but what would be a good decision tree you know, where do you go left, where to go right? Can you please allude on that one?

Stefan: Yeah, absolutely. Because keeping the core clean does by no means mean moving all extensions onto the BTP. The business technology platform is a key integral part of SAP S/4HANA Cloud, and we are eating our own dog food at SAP and development. We are building future innovations and solutions side by side on the SAP business technology platform. Think about sustainability management, a set of BTP apps, green ledger that we will build or Carb, the carbon accounting hub, how it is uh called is also a set of BTP apps. The digital manufacturing cloud and digital twin business networks also built on the BTP. So when you want to extend SAP S/4HANA, you have two options. First of all, to build it on, stack inside SAP S/4HANA or side by side on the SAP Business Technology platform. When to use what? So let's assume you want to extend an existing S/4HANA process. You want to extend the financial accounting process or somewhere in production planning MRP, where your extension sits close to the core S/4HANA data. Then you do it on stack. If you extend the financial accounting process, it doesn't make sense to replicate the whole universal journal A to the BTP to do logic and then copy it back. Whenever your extension depends on fast SQL server access to the SAP data. Heavy data exchange, a lot of data loads. One example you want to extend the MRP, the material requirements planning process with a BOM explosion of millions of data. You do it on stack with ABAP in S/4HANA. If your extension needs strong consistency with the SAP transaction. Yeah. At the moment a sales order or production order is stored, your customer object also needs to be stored, do it on stack in SAP S/4HANA and we have in S/4HANA our Low-code No-code so-called key user extensibility tools. But now also ABAP cloud embedded in S/4HANA, we call it um, code name embedded steampunk. You have all the ABAP world flexible coding in S/4HANA. You can do then extensions, but only against public APIs. But for all other cases, please do it side by side on the business technology platform. And especially when you build a hub scenario, when you want to connect multiple S/4HANA systems together, like you want to create a supplier portal or a customer hub or something like this. If you are a partner and you want to build a multi-tenant partner SaaS solution, you build as a partner a track and trace functionality that you want to share multi-tenant with multiple customers. Do it on the BTP. If you build an extension, like a customer portal or a dealer portal for car manufacturers, and those users should not have access to the back end system to S/4HANA do this all the BTP. It's perfect because you have separate segregation of duties. And if your extension, or your extensibility, your application needs higher frequent shipments, you want to update your BTP application or your extensions monthly, daily, hourly. And the core S/4HANA system only upgrades yearly twice a year, etc.. So these cases do it on the BTP. It shall be wisely checked when to use what and we promote this as a decision tree and guidance. It's also written in our extensibility guidelines and papers. And let me give you one customer example here, because a customer has also done this. And let me talk about

shortly about Hitachi High-Tech. It's one of our reference customers and Hitachi High-Tech Japan they are semiconductors, high tech company, some 3000 employees. And they are, they have currently a corporate ERP system and S/4HANA private cloud and two S/4HANA public cloud subsidiary systems. So private cloud plus public cloud. And they started with S/4HANA already in 2018. And back then they had 9000 add ons or 9000 extension components. They reduced them over the last five years. They reduced them to 470 by deleting a lot, cleaning up moving processes back to standard functionality. And out of these 470, 22 extensions are on stack in S/4HANA, and the rest, the 450 are outside on the BTP and they are using public APIs and they have really separated this and kept the core clean with only clear 22 extensions there. They are strictly governed. And what is the benefit? Back then in 2016 they were on PHP seven and they did an upgrade to PHP eight. It took them overall one and a half years, 15 months, roughly um, with unicode, enhancement packages, etc. Now they are on S/4HANA private cloud, public cloud and they are doing yearly upgrades 2020. They have done 2021 and now 2022, and the last upgrade to 2022, took them 30 days. So end to end with setting up the project, testing, upgrading, etc.. Why? It sounds super, but why? Because they have their extensions under control. Most of them are using their decoupled from S/4HANA code via public APIs. But the real major thing is they trained their upgrade muscle we call it, because they did so many upgrades between 2016 and now 2023. The first years it was painful. It took quite some time. And you get better, you get better the more often you do it. And now it's an easy way and they can fast consume the innovations. And this is mainly the mantra about clean core. To put yourself in a position where an upgrade is not a painful event anymore.

Dirk: Fantastic example. Stefan, I really like it. I think the couple of points I want to highlight. It indicates faster getting the innovation in the market and wanted. Right. I mean, one is the, you know, the burden and the complexity a few years ago compared to today. But you can grab the innovation in the market more quickly. That's one. Number two, there's obviously a lot lower costs, right? If you do it in 30 days rather than 15 months, and you don't have all your talent stuck in some standard work, which doesn't add a lot of value necessarily. Last but not least, what I really like is you describe this also as a journey, as I said before. So this is not something which I can switch on and switch off. It's a learning for the organization and it is learning also for our clients in the markets. One thing where I think we need to get better as partners is to explain what you started with explaining in terms of really making sure that the market better understands, our clients better understand, you know, the logic and the decision tree. And what do you put where and why? What I feel is that there's still a lot of uncertainty and probably lack of understanding and knowledge of this future architecture in terms of how you can best leverage the BTP, what you put in, what you shouldn't put in.

There's probably also a bit of based on the uncertainty, there's tendency not to make a decision to grab more understanding before you make a decision. So I think we need to educate and guide our clients a bit more and better on that topic, because this is key to get what you what you just described on this excellent Hitachi example to get to the North Star and get to where you want after a couple of months, years and so on.

Yannick: Thanks a lot, guys. I loved what we shared. We talked about the why this is needed. So what it is concretely, where should we do something that needs to very close to the transactional side of the house? Where can you differentiate with extensibility on the BTP? Like a lot the customer example and of course understand there from you, the partner perspective is really to educate the market, educate our stakeholders on, you know, how to bring that concretely to the streets. And question that I have now of course, is what is the roadmap of all these concepts maybe Stefan, you would like to allude here, what does SAP in store to further advance the clinical strategy?

Stefan: Yeah, definitely. We are constantly helping our customers really to to move here. And so especially on the BTP side, uh, with the SAP build code tools, it's way easier now to build extensions on the BTP. In future, especially this year, we will also more put more generative AI into the code generation will be much more easier and comfortable to build on the BTP cup applications cloud application programing model. The same we will do also in the ABAP cloud side when you build according to RUP, restful upper programing model. As said above, ABAP cloud is the new cloud like ABAP language version. SAP is using it also to build S/4HANA public cloud versions here, and we ask any kind of partners customers to move from classical ABAP to ABAP cloud. Um, rollout has been done on TechEd last year and will be also done this year. In addition, on the rise with SAP, we are currently creating a clean core dashboard. It's a dashboard Fiori applications that analyzes your customer system, shows, non-compliant code, shows outdated data structures in compliant master data. Gives you hints and also connections and links to tools to analyze it. You can define KPIs, for example, usage of public APIs, usage of non public APIs, and etc.. In addition, on the rise with SAP, we are also discussing now to promote customers differentiated pricing migration incentives for those customers that strictly follow the clean core. Because when you are on the private cloud, on the rise with SAP and have achieved a clean core, we drastically also can reduce your TCO. It's also easier for the partners and also for SAP to perform the upgrades. And we are discussing partner certifications for clean core compliant add ons. We want to really also promote partners that change their classical ABAP add ons to clean core compliant, decoupled from the core, either moved on the BTP or using ABAP cloud. And here also, this is on the focus for 2024.

Yannick: Great. So we are approaching the end. What I just understood is that tools are here to support. There's a problem TCO impacts. And before we close this session, I'd like for the two of you to to help give a concrete takeaway to our audience, like a step by step that the company should follow to achieve the clean core. What are your top recommendations? Maybe you want to start, Stefan.

Stefan: A call out to all customers that are facing a lot of that code and are sitting on an on an ECC and want to move to S/4HANA. Don't fear the code. Get it in structured. The way to clean core is, first of all, analyze your custom code. Delete as much as possible that you have. If you have subqueries or set transactions that are only used once a year or haven't been used for the last three years. Deleted. For the rest of the custom code, structure, it, check where you can move to standard, functionality. If you have an old track and trace functionality, use the standard one. If you have an old travel management solution, move to conquer, etc.. Um, then you are the path for clean core and for the rest of your custom code, gradually refactor it and it's code hygiene. The refactoring needs to be done. It's it's getting control over your old code, and then you are on the perfect way and move to the cloud. And whenever you want to consider moving to public cloud, public cloud is clean corporate default. If you currently do not achieve clean core in your on premise system, you're most probably not ready for the public cloud. So do it, get it done, and start it. We are happy here to help.

Dirk: I would have that said the same. I would say to my clients: Initiate the journey. There's multiple steps to do moving your SAP ECC into S4 rise cloud, whatever solution, public or private. This is only one step in a longer journey, as we have discussed. So, I think start now is is the key point. Make conscious decision is the other point. I think we, we elaborated on that, having a decision tree, I think it's worthwhile to spend time to think through more from the business perspective in terms of what your strategy is or what business models you you most likely need, what management cycles you have. Steering model, operating model before you move in technology as I said so the four layers. Yeah. So when you define the North star and you look in the technology, you need to have a robust alignment with your strategy and your with your business. And then you can look into how will future infrastructure and security have to look like in the cloud, how the SAP should look like, how the composable technologies and the connected solutions tie into that. And then you can define a roadmap and that could be flexible in terms of which step you take. If you have a plan, which step you take at one, what point in time, you don't have to do everything at the same time. But I think there is a bit of an, urgency to start the journey because it's as we

described, it's quite a certain complexity you have to manage, you need to think through stuff. You need to make a conscious decisions. There's a couple of positives in the market. Obviously, there's a huge acceleration. If you think about Gen AI, in terms of what it gives you in terms of capabilities, but also in terms of delivering programs like that. So there will be a higher level of automation over the coming months as we speak. There's a lot of investments in the market getting into, um, accelerating delivery as much as providing new capabilities on the process side. So we will be able to achieve targets with lower cost and in shorter time than we have done that in the past. If we take all, all our good things together already today, we have an acceleration, visible acceleration in how we deliver those things. And I expect over the next 12 to 24 months an exponential increase of capabilities, which will help us to deliver those programs to lower costs and more tailored to the value case we have created jointly. So start the journey. It will be exciting. It will be challenging, as we discussed. But, you have to start. Otherwise you will not have the or the benefits we described in finally having a digital core in the market.

Yannick: Great closing words. Thanks a lot. In a nutshell, there's a sense of urgency. So start the journey. Make conscious decision. Don't fear the code, but refactor the code and then you are on the right path. I'm sure our audience has learned a lot from your respective insight and experience, so thanks a lot again, Dirk and Stefan for joining us on this episode. As always, feel free to drop us a note at the insides4@sap.com and let us know which topics are of your interest. Tune in next time and always be Inside SAP S/4HANA. Bye, bye.

www.sap.com