



For Cloud Professionals, part of the On Cloud Podcast

David Linthicum, Managing Director, Chief Cloud Strategy Officer, Deloitte Consulting LLP

Title: The evolution of the cloud architect's role

Description: Before cloud, system architects lived by rigid system requirements—receive them, build to spec, and move on. That was their world. However, with the ascendance of cloud, that world has changed. Architects are now expected to be just as smart and agile as the applications they build. In this episode, David Linthicum and guest, Deloitte's Vasa Vishveshwara, discuss the ever-changing, ever-expanding role of the system architect in the world built by cloud. According to Vasa, architects have to be just as well-versed in the business as they are in cloud technology. And, to stay relevant, they have to hone their communication skills and continuously expand their knowledge base—both from a business and technical perspective.

Duration: 00:22:36

Operator:

This podcast is produced by Deloitte. The views and opinions expressed by podcast speakers and guests are solely their own and do not reflect the opinions of Deloitte. This podcast provides general information only and is not intended to constitute advice or services of any kind. For additional information about Deloitte, go to [Deloitte.com/about](https://www.deloitte.com/about). Welcome to On Cloud, the podcast for cloud professionals, where we break down the state of cloud computing today and how you can unleash the power of cloud for your enterprise. Now here is your host David Linthicum.

David Linthicum:

Welcome back to the On Cloud Podcast, your one place to find out how to make the most of cloud computing and make cloud computing work for your enterprise. This is an objective discussion with industry thought leaders who provide their own unique perspective around the pragmatic use of cloud-based technology. Today on the show we have Vasa. He's one of my colleagues here at Deloitte and he's a managing director with Deloitte Consulting LLP. He's chief architect focused on digital and cloud, and he has more than 18 years of IT experience and has been on this show before and he got some great reaction. How are you doing this afternoon, Vasa?

Vasa Vishveshwara:

Pretty good. I'm glad to be here.

Dave Linthicum:

So, what's happened since the last time you were on the podcast? What kind of adventures have you gone into? What technology have you focused on? Without mentioning any names, what kind of systems have you built and designed?

Vasa Vishveshwara:

Yeah well, a lot happened. The technology continues to change. The way we are architecting things six months ago and now are almost – you know, changed big time. And also, I'm leading a next-generation architectural program within Deloitte Consulting, and this program is primarily focused on building more architects who are a lot more focused on the technology and implementation and to grow the talent within consulting.

Dave Linthicum:

Great. So, how would you define the next generation of architects?

Vasa Vishveshwara:

The next generation of architect is the one who actually understands what is happening in the industry from the technology perspective, at the same time look at the business problems, which are pretty old. There are not any new kinds of problems. But they use technology to solve in a new, innovative way. So, the architects are primarily looking at the innovative tool set that is in the market today and solving the old business problems with the new tool set. So, that is where the architects are a lot more focused on.

Dave Linthicum:

So, I've been involved in software architecture my entire career, getting into the structure, analysis, and design stuff. I even taught at the college level and then getting into the object-oriented stuff, getting into component-based development, getting into server-based development. Now it's container-based development and the ability to deal with serverless architectures and some emerging technologies after that, and probably ten generations of programming languages, all the way from C now to Python and beyond. So, what's the current state of software architecture, and what should people be focused on right now in building applications specifically in the cloud?

Vasa Vishveshwara:

Yep. So, the role of an architect is slowly changing, especially in the agile and cloud world. In the olden days the architect used to be the only guy who understands all the requirements together and then comes up with a solution. But in a new world, we are not giving them all the requirements. We are going with the agile approach where we are defining minimum viable product. So, with that concept, what is happening is that the role of an architect is becoming defining minimum viable architecture. What this means is that the architecture needs to have enough flexibility today so that we can quickly build it and put it out, and it might change in the next release so they have to keep that in mind.

The second thing that's happening is around the cloud space. All the cloud providers are releasing new types of tool sets to solve the problems, the same old problems. And architects need to understand these tool sets, these native services, and then incorporate those into their minimum viable architectures.

The third one I would say is the cost. Before they used to have this full architecture defined, they used to have capacity planning, and they used to know how much it's going to cost. But with the cloud coming in and defining minimum viable architecture, the options are way too many to solve the same set of problems. So, it's important for an architect to understand what are the costs of these native services and which may be a good fit for minimum viable architecture. So, the role has twisted compared to old versus new.

Dave Linthicum:

So, when I was CTO and like I said, when I was building applications as a consultant, which I was before I was a CTO, and I had to move into designing and building products, it was a completely different kind of architectural process to make this thing happen. You have to deal with resiliency. You have to deal with the ability to operate in a multi-platform environment. All these things kind of come into play. So, what are the different types of architectures out there, and why is product architecture a bit different?

Vasa Vishveshwara:

So, in the typical scenario when we build a product or a project you need to think about multiple dimensions of it. Some people mention about solution architecture or application architecture. Some will talk about integration architecture. In some areas we only purely talk about data architecture. So, for a given product or project, you need to understand all these dimensions from an application, integration, data, deployment, infrastructure architecture, security architecture to have that full view of it. Now some level of architects only define the application-level architecture. They focus on their small application and how that application is going to perform and how the systems are going to interact with each other from the architecture standpoint.

But when we look at the product and the product rollout, as we are releasing product feature set in an iterative manner, an architect needs to think about all the domains, okay in multiple dimensions I talked about, starting from application to integration to data, along with how it's going to be rolled out from the deployment standpoint. So, in the product concept you want to be more efficient and quickly roll out your feature set, at the same time have less investment and more cost efficient. In the project scenario you may not see all those non-functional requirements that are applicable to the product side (Inaudible). So, it's very important that as an architect you understand both the functional side and the non-functional side requirements and then architect according to the needs of the business and the business case.

Dave Linthicum:

So, when we consider innovation and innovation around architecture, you kind of get some strange looks, the ability to in essence innovate around defining architecture in different ways. And so, there was EAI and service-oriented architecture and cloud-based architecture moving forward. And so, what role does an architect play in developing things that are innovative? And what recommendations would you provide the audience so they can move to more of an innovative state?

Vasa Vishveshwara:

Right. So, the one big thing I think the architects play in the innovation is that they should be the ones who have the right skills to understand what is happening in the marketplace. Architects are the ones who know, who understand the new innovations that are happening in the marketplace, and how do

you use those innovations to solve the problems in a quick and innovative way? If you look at all the cloud players, the technology continues to change very rapidly. And if you are an enterprise and if you want to take advantage of those innovations, then you need to let your architects wear their innovative hat and understand these new tools and technologies, and then see how they can incorporate it.

As you mentioned about cloud or EAI, these are all new, cutting edge in some areas. In those scenarios you need to let architects allocate some time so that they spend enough time understanding those innovations. Otherwise your enterprise never moves on pace with the innovation that's happening in the marketplace.

The second one I would articulate is around operations side of it. Innovations are happening across the full spectrum of the software development lifecycle. One of the big areas is the metrics or the measurements of your product or an application. There are quite a bit of innovations that have happened in the last few years on how do you easily measure metrics on your product and your application and quickly make decisions based on those data sets so that the next architecture can be more improved? So, it's important that the architects take advantage of those innovations and incorporate the feedback into the next set of architecture.

So, overall I feel that architects are the right practitioners who understand what's happening in the marketplace, who understand the business problems, and they can bridge the gap to solve the problem in more innovative and more efficient way in this new world.

Dave Linthicum:

So, moving forward we're changing how we're building software. So, it used to be the waterfall methodology, and I even taught that to college students, where we go to design to development to testing to deployment to operations. And everything was dependent on the previous step being completed, and obviously that had limitations to it. People just had a problem in terms of their ability to get things done in a fast amount of time and business was complaining about that. It took an average of six months to a year to build any kind of significant systems, and perhaps half that time to change these systems. So, we moved to agile and the ability to kind of get to an iterative development process and even automated by DevOps moving forward. And so, architects have to exist in those environments, and so how do they change and how do things keep on changing these days? And also predict the future. Where are we going from here?

Vasa Vishveshwara:

So, one of the key things that happened is with agile, people thought that they don't need architects anymore. And surprisingly at this stage we need more architects trained rather than before. As in the waterfall phase, you have all the requirements in one place, architects spend time in designing the architecture, and then afterwards they tend to go away or spend on our next project. But in the agile world we never come up with the full requirements. So, we came up with the minimum viable product, and you probably have a six-week or ten-week release, depending on how you're implementing agile.

In that scenario, an architect needs to spend enough time to define the architecture for that program VP. So, we start calling this MVA, minimum viable architectures. Now that architecture needs to be flexible enough to say that this release, this the particular architecture, is good enough. But once the requirement changes or new features that are added in the next release maybe after ten weeks, then again you need architecture time so that you can tweak your architecture or you may be rethinking your components that are needed for a new sort of feature set. So, it is completely changed that you definitely need an architect and his time more than before.

The second point I would make is that the minimum viable architecture also demands the flexibility and the components. Both concepts are there before, but in the new way when you architect the solution, in the release one you might leverage one cloud tool set. And as your product becomes more successful, that particularly component in the architecture may not be feasible or may not be cost efficient. So, in the next release you probably literally need to change a specific component and completely replace into a new component, a new set of a cloud-native service. So, the architect needs to think of those ahead and then incorporate that into their architecture. So, the life of an architecture once you define it, it may not be more than six months. Before it used to be years. Once a system was implemented, people used to use it for many, many years. But these days with the innovation that's happening so fast, the life of an architecture diagram is shortened big time.

Then you asked about where we are going with this. You know, in the next few years the way I see it is that the architects who are hands-on, who are learning, understanding new technologies and tools and who are incorporating these tool sets into their next batch of architectures, whether it can be an application architecture or whether it can be a deployment architecture, they should be leveraging these innovative tools, incorporate those components into their architecture. Then they will continue to thrive. If an architect wants to stick to what they're doing and does not innovate themselves and does not understand the changes that are happening in the industry, then their role is going to be diminished.

So, I think that definitely the more time is required from the architects in the future. And what I mean the architect's time is that the practitioners who are innovative, understand the technologies, at the same time incorporate those into their agile architectures.

Dave Linthicum:

So, moving forward, a follow-up question about architecture, we're in essence trying to optimize what we're looking to do. So, for good architects, we're trying to get to a state where the resources that are involved and the applications, how they run, CPUs, databases are efficiency-optimized and cost-optimized. And the deal is, is that architects have different ways of thinking, different ways of processing architects, and they always come up with different answers. And the great thing about technology is pretty much everything works. I can make any database work with any application, all these sorts of things, but there really is one efficient way to do it, one optimal way to do it. And I don't think we're thinking about that, are we? Are we thinking in terms of optimizing the architectures or thinking in terms of assembling things using our belief system that we developed through our entire career? In other words, is there a bias toward technology when you're dealing with architects? And if there is, how do we get that bias out of our thinking?

Vasa Vishveshwara:

Very good question. Most of the architects grew up in enterprises as a programmer, became the tech lead, then became an architect. So, if you think about it from their mindset it's always about, "How do I solve the problem with given technology options?" But they don't have the bigger picture of where is the business going? What does the business really want? And what are the business drivers for this particular solution? I think if our architects can understand

the bigger picture, then drive the architecture where they're coming of it from the business side of it, then I think it's going to be a game-changer. It's a simple change. It's a mindset change on how do they do it.

I talk to a lot of enterprise architect teams and leaders of enterprise architecture, and always their focus has been that, "Oh, this new innovation came up; can I try to use it into my business?" Rather than looking the other way around, saying that, "Oh, here's a business problem that my business is solving; now what innovations are there in the marketplace which can help me solve this business problem most innovative way?" So, there is a subtle difference on how do you think, how do you frame the case, and people who understand the business lens are a lot more successful in convincing others and making it implementable. So, I think that is a shift that needs to happen.

The other articulation I give is defining the non-functional requirements. This helps the architects lay on the ground and say that, "I'm creating this architecture – great, but how am I going to review this architecture and say that this architecture meets all the requirements?" Now when I say requirements I'm not just talking about the functional side of the requirements but I'm talking about the non-functional requirements. Especially when you define the minimum viable architectures, you probably define an architecture – keep in mind that this will be used by a small set of users.

But suddenly, your assumption fails and say that you've planned for 100 users per second for your architecture and suddenly it goes to 10,000 users, right? So, if you don't define those set of non-functional requirements, ground the functional teams and the business teams, then it always will become blame on the architecture; the architecture cannot scale. But originally because you haven't defined that non-functional requirement that can happen, but if you defined a set of non-functional requirements clearly and made assumptions to those non-functional requirements, then it becomes much more easier to articulate that, yes, this architecture is defined from this business context and these technology assumptions. Then you'll be on the successful path.

Dave Linthicum:

So, I'm an experienced architect and I'm looking for career advice from you in terms of what Vasa's going to say and how I'm going to manage my career and how I'm going to become a better architect moving forward, and also people who aren't architects now but want to be architects. I notice that architects are typically developers who in essence get more holistic design skills and move into the role, which I think is a good career path for developers moving forward. I don't think you can build architects directly out of IT. You need to have experience with many levels and multi dimensions in terms of IT. So, what advice would you give the experienced architect and what advice would you give developers who want to be architects?

Vasa Vishveshwara:

Yep, definitely. So, for the experienced architects I will articulate two key things. The first one is about stay relevant and be hands-on. One of the things I'm observing is that once somebody has started doing architectures for ten years, they lose touch, and we start calling them PowerPoint architects. So, they just can create PowerPoints but they cannot articulate how that can be implementable. So, it's very, very important that if you want to continue to be an architect you need to be relevant in the industry. Experiment with the different tool sets so that you are on hands-on and educated in your stuff.

The second one I would articulate is the ability to convince others. If you are a strong architect, if you come up with an architecture which is a solid architecture but you cannot communicate, if you cannot convince others on why your architecture is solid, is the right one, then that architecture does not add a lot of value. So, it's very important that all the experienced architects build their communication skills, build their convincing skills. I don't want to say sales skills, but it's kind of selling your architecture, but in different terms. So, the ability to make others understand your architecture is a key part of it. And you have to tailor your message to your audience. If you go and talk deep technology to a marketing person he may not understand it and he may not appreciate the value of your architecture. So, that's why it's very important. So, that is two aspects for the experienced architects, stay relevant, be hands-on, along with develop and refine your communication skills constantly.

Dave Linthicum:

So, last question. How do you stay relevant, Vasa? How do you keep up?

Vasa Vishveshwara:

Yep. Again, this has to be your passion. Read every day. You know, there are so many resources which are available in the marketplace which will help you stay relevant. You can subscribe to so many articles, so many newsletters from cloud providers or industry providers and continue to read it. And that way you are relevant. And then play with these tools and technologies. Otherwise you will always have a basic knowledge but you don't have that next set of knowledge to explain and convince others why this tool or technology is a good fit.

Dave Linthicum:

That's great advice. Let's leave it at that. So, if you enjoyed this podcast make sure to like and subscribe on iTunes or wherever you get your podcasts. Also don't forget to rate us as well. Also check out our past episodes including the On Cloud Podcast hosted by Mike Kavis, my good friend, and his show Architecting the Cloud and his book by the same name. So, if you'd like to learn more about Deloitte's cloud capabilities, check out DeloitteCloudPodcast.com. and if you'd like to contact me directly you can reach me at DLinthicum@Deloitte.com. So, until next time, best of luck in your cloud projects. Thank you, Vasa, and we'll talk again real soon. You guys take good care.

Operator:

Thank you for listening to On Cloud for Cloud Professionals with David Linthicum. Connect with David on Twitter and LinkedIn and visit the Deloitte On Cloud blog at www.deloitte.com/us/deloitte-on-cloud-blog. Be sure to rate and review the show on your favorite podcast app.

Visit the On Cloud library
www.deloitte.com/us/cloud-podcast

The views, thoughts, and opinions expressed by speakers or guests on this podcast belong solely to them and do not necessarily reflect those of the hosts, the moderators, or Deloitte.

As used in this podcast, "Deloitte" means Deloitte Consulting LLP, a subsidiary of Deloitte LLP. Please see www.deloitte.com/us/about for a detailed description of our legal structure. Certain services may not be available to attest clients under the rules and regulations of public accounting.

Please see www.deloitte.com/about to learn more about our global network of member firms. Copyright © 2018 Deloitte Development LLC. All rights reserved.