



## The Deloitte On Cloud Podcast

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

**Title:** Mediating cloud complexity with abstraction and automation

**Description:** Most organizations that have moved to a multi-cloud architecture have experienced cloud complexity. In this podcast, David Linthicum tackles the complexity issue and gives advice to mediate it. According to David, abstraction and automation are essential. First, deploy an abstraction layer above and across cloud platforms for tasks like operations, security, governance, and configuration management. Then, automate as many of those processes—especially operations—as you can.

**Duration:** 00:20:55

**David Linthicum:**

Welcome to Deloitte's On Cloud podcast knowledge short exploring a specific topic related to cloud computing. This is a short tutorial talking about real world concepts in the emerging world of cloud computing. I'm your host, David Linthicum, cloud computing subject matter expert, author, speaker, and Managing Director with Deloitte Consulting. And this is, "Hitting the Complexity Wall."

So, after leveraging cloud computing for many, many years, we started to get into very complex deployments such as multi-cloud, things like that. And really, 95 percent of the enterprises out there, if you look at any number of studies, are deploying their cloud solutions using a multi-cloud deployment. In other words, they're using more than one cloud brand. It could be AWS, Microsoft, Google, it could be two of those, it could be three of those, it could be even additional clouds, managed service providers, the Oracle cloud, IBM cloud. All these are things are kind of coming together to make up the solution.

What's happening is that, as we're pushing out these very complex distributed architectures, we're hitting a complexity wall where it's difficult for us to add additional systems, additional applications, databases, operating systems, platforms, things like that, because we're hitting an operational complexity limit where we just don't have enough resources to manage the complexity of the deployments that we just made.

So, how do we know we're hitting a complexity wall? When adding and removing applications or data sets takes more time or breaks other systems we see that it's on. In other words, we may add a database, we may add an application, we may add a platform, we may add an AI system, and a business analytics system—any number of things. And in doing that, it's a much more complicated and laborious task than it really should be. That's because we have to hook up to all these various middleware layers, all these various cloud platforms out there, and do so in such a way where they're going to work and play well with the existing complexity that we're dealing with now. And if they don't, they're going to break.

In other words, we're going to delete a database element, delete a table or something like that that's no longer needed, and it's going to affect lots of things that exist in the environment because they don't have a handle on it, because of the complexity of the various dependencies that we have in building whatever cloud-based system we're talking about. But, typically, this is going to be a multi-cloud where complexity walls are going to be a core factor.

So, something else that may tell you that you're hitting a complexity wall—systems and operation costs actually go up a great deal, and even exponentially. Meaning that we're paying more for operations, as a percentage, than we did in the past when adding an application, database, platform, edge system—things like that.

So, we're always adding technology onto the platforms that we're deploying on. And if it's a multi-cloud deployment, we're going to be adding AI systems and applications, and databases, and business-analytics systems—things like that that we need to run the business. And what we're seeing when we're hitting the complexity wall is that the cost of doing those things and operating those systems is going to be far greater than it should be. And, so, suddenly, we're going back and looking at our budgets and we're only budgeted for, say, \$10 million a year in terms of operational staff, and resources, and budgets, and their tools and things like that. And in order to operate something with additional complexity, it may go up to \$15 million or \$20 million.

If you go back to a board of directors and say, "Well, listen, look, we've moved into the cloud and things are a bit more complex. We have lots of systems that are very different, it's very heterogeneous." Heterogeneity is an issue unto itself. "And, therefore, we're going to need additional money to operate the systems," they're going to look at you like you're crazy. Because, ultimately, we were told—or they were told that cloud computing is about reducing operational costs by a significant amount of money. That's why we moved into the cloud. However, this is counter to that message. In other words, we're increasing the operations budget because the complexity wall is kind of pushing us to doing that.

Another issue: Operations personnel turnover is higher than before cloud computing implementation, meaning that operations is getting tossed under the complexity bus. So, what's happened as we moved into these very complex distributed architectures—multi-cloud being an instance of that—is that we put more stress on an operations team to do more with less. And, so, in other words, we have more platforms, more databases. These systems are going to generate a great deal of heterogeneity.

In other words, lots of different databases, lots of different operating systems, lots of different application development platforms, operations platforms are looking to deploy and operate. And they're typically not given additional money to hire the skills they need to get the tooling they need to deal with these additional platforms. So, they may have gone from, say, 1,000 cloud services under management three or four years ago, to 2,000 to 2,500 cloud services under management now, and they're doing so with the same amount of money, the same resources. And, so, they view themselves as being set up to fail.

They may find that higher turnover is going to be in these organizations and the operations teams. In fact, you can kind of tell when businesses are hitting the complexity wall, because they are getting to this point where turnover is much higher to the fact that it's very obvious to people outside the organization. They're posting lots of operations jobs around CloudOps systems and database systems that they're operating, things like that. Because the complexity has reached such a state that the operations team is stressed under the additional complexity, and they're basically stating that if we don't get the resources we need to do our jobs, we're going to find other places to work. So, you may see higher turnover as an indication of the fact that you hit this complexity wall.

Finally, you may see that leadership questions of value in the business case is way off versus what was sold to them in past years. And, so, I mentioned this earlier—so, we may go to the board of directors and say, "We've reached a certain point of complexity in building out our multi-cloud deployments where the cost is going to go way up." And if they don't see the value that's coming back to the business directly—in other words, you can't map it directly—that's going to be an issue.

So, they're going to consider the fact that cloud computing was sold to them 10 or 12 years ago when cloud started to show up as a viable alternative to the way in which we do IT, as something that's going to be more cost-effective, it's going to be operationally less expensive, it's going to be more agile, it's going to basically take the business to the next level. And if they see us coming back, and they see the operations teams coming back, and talking about the fact that we've hit this complexity wall because of the number of systems that we've built is starting to stress our operations in terms of heterogeneity, complexity, or ability to be agile, ability to onboard new systems, do things like acquire businesses.

So, therefore, we need more resources to make it happen. That's contrary to what they heard in the past. And by the way, that's contrary to what the press said in many instances. So, you have to keep an eye on that.

So if you basically said yes to one, two, three, four, or all of these issues that I just mentioned that are part of your cloud experience, you may have a complexity problem, one that needs to be corrected as soon as you can, and, also, you need to put mechanisms in place to ensure that, moving forward, you're reducing complexity, not increasing it as your enterprise continues to leverage cloud computing and do so in effective ways.

In other words, this is about being very pragmatic with our architecture, understanding that if we move into these very complicated multi-cloud deployments, that complexity, and our ability to operate systems that are very complex, is going to be one of the walls that we're going to have to overcome. And it's one thing to try to avoid it, because I think if you avoid it, you're going to find yourself not leveraging cloud as the true force multiplier for what it is and what it can be for the business.

In other words, providing people with best-of-breed technology that allows them to do their job better. So, if I'm a developer, I'm able to build something that's much more optimized and much more innovative. If I'm able to look at all cloud services as areas that I can pull from into my solution, versus just saying we're going to go with one particular cloud provider and we're going to have to build systems in their walled gardens.

So, what we're saying here is that we understand why multi-cloud is the path going forward in the future. In other words, best-of-breed, most agile, the ability to adapt to change in a much faster and accelerated way for the business. The downside is going to be that complexity is something we're going to have to manage.

The idea here is not necessarily to accept the fact that we're going to have to throw money at the problem to get around the complexity wall, but this is about doing architecture differently. This is about thinking differently in terms of how you deal with complexity, using things such as abstraction and automation. We'll talk about that a little later.

So, cloud complexity is not a new diagnosis. Certain people look at IT architecture as something that exists to serve a single application, or a small systems domain. These days, many organizations fail to select a cloud technology to serve enterprise IT as a whole. They deal with a series of tactical application use cases that all have one-off cloud architectures, and they do this 20, 30 times, and then you have a real problem. Complexity sneaks up on you.

So, the results of complexity really is predictable. Many people, I wrote about this in my InfoWorld column five years ago, saw this coming a while away, and lots of other folks did in the industry, as well. So, the cloud promised to lower costs and increase agility, but you face complex, massive technologies built by one project or another. Things are difficult to change, and the value that the cloud was supposed to bring isn't there. Moreover, you toss the keys to those who are in charge of operating these systems, the CloudOps team, you're likely to see them fall down from the number of moving parts and heterogeneous complexity that they're going to run into when they finally operate the systems. In other words, when they're tasked with the long-term viability and maintenance of the systems and have to deal with the complexity firsthand.

So, companies today commit a huge amount of time and money in migrating applications and data to the cloud. This makes IT infrastructure such as storage and platforms, open and closed based cloud systems, more complex for administrators, developers—ultimately, end users. Keep in mind that legacy systems still have to run, and you, the IT leader, now have public and private cloud-based systems to deal with as well. Things continue to get more complex and without proper planning, the symptoms and what we'll see as the rise in operational complexity is going to get much worse.

So, we talked about complexity and kind of stated the problem in a lot of ways, because I think people are going to have to recognize the complexity wall when they hit it. And most organizations out there who are doing anything significant in the cloud, and certainly moving the multi-clouds, are going to hit the complexity wall. And as long as you understand that you're about to hit it or have it and can manage your way around it, there's things you can do to work around how you're dealing with complexity.

So, how do you deal with complexity? Well, the thing is, there's no over-the-counter remedies, here. You can't ignore the problem and expect it to work itself out. If you do, the number of systems, cloud and legacy, will eventually become cumbersome and impossible to manage. So, the first step is acknowledging IT complexity as a chronic issue with potentially enterprise-wide ramifications. Address the symptoms, but also grow your IT organization with an eye on complexity and say an ounce of prevention ultimately is worth a pound of cure.

So, I'm urging you to kind of do a couple of things. Number one is understand what the complexity wall is, understand the complexity wall exists, recognize it when you're about to hit it or have hit it, and also be willing to take actions to work your way around it.

So, let's talk about what those actions are. Well, as I mentioned, abstraction and automation are the main weapons that you have to manage complexity. In fact, at Deloitte, we have a cloud complexity management approach in problems. In fact, we have a cloud complexity management calculator—you can go out there today, just Google for that, and figure out how much complexity is costing you and how much complexity is an issue in your current state of IT, and also look in where you're going and figure out if complexity is going to get worse and how much you're paying for the complexity.

We solve the problem by stop creating redundant solutions. So, the one reason that complexity is such an issue is the fact that we have to build little silos of technology for each of the cloud providers, for each of the platforms. So, we have security for our legacy system, security for our AWS deployment, security for our Microsoft deployment, security for our Google deployment. And all these things are really native to the particular cloud providers, which, unto itself, is a pretty easy choice to make. Because obviously, a security tool that's native to a particular public cloud provider is going to work well in that particular cloud provider. And if we're providing security just for that provider, it's going to work well in that silo. In this case, the silo is the public cloud provider.

So, if we're going to mediate complexity, ultimately, this is about putting technology that runs in between the clouds. If you think about it, multi-cloud really is not just about the clouds. It's about what exists between the clouds. It's common security systems, it's common operations systems, it's common governance systems, common deployment in management platforms, common configuration management solutions. So, the idea is that we're reducing the complexity by eliminating a lot of the technology redundancy, by not necessarily picking technology for each of the platforms that we're running that are specific to those platforms, but technology that's able to run across cloud, and even across cloud and legacy systems.

So, common security layers—able to do identity access management and basically manage security through a single layer. Common governance layers. And probably most important, a common operations layer. So, instead of just figuring out operations on each of the particular cloud providers, or the private cloud you may have there as well as legacy systems, we're able to look holistically at the solution and put an operational layer, typically using abstraction and automation—I'll explain what those are next—that are able to run across the various cloud providers to remove a lot of the complexity in how we deal with all those services on those cloud providers.

So, instead of dealing with the different native storage environments on each particular cloud, we're dealing with the holistic solution of storage and managing it through one layer of abstraction. Therefore, we're not having to run into the complexity wall because the complexity is hidden from us. In other words, abstraction is able to allow us to manage very complex systems—in this case, cloud storage on multiple platforms in multiple clouds—and do so as a single holistic concept, as a single dashboard.

Through a single pane of glass—you'll hear that word in the operations space. Well, you do that for compute, and you do that for AI operations, and do that for database administration and database management, and you do that for governance management, and you do that for security operations. And really get into the habit that, instead of picking the easiest solution—in other words, just picking whatever native security product runs in the particular cloud providers, and therefore, if you have five cloud providers, we're going to have five different security solutions, at least; usually, there's more, and also the ones that run on the traditional systems such as legacy.

But the ability to, in essence, look at this as an architectural optimization exercise where we're going to try to reduce as much technology as we can that we're deploying to solve the same problems, therefore eliminating a lot of redundancy using abstraction cross-cloud services, things like that, and the ability to automate these various systems. So, instead of us doing backup and recovery operations on each cloud provider using a separate product, we do so across clouds. We do so with one product that's able to back up and restore all the storage systems and all the databases that exist across the cloud environments, not individual solutions that exist natively on each particular cloud.

So, you get the idea, here. It's common to look at complexity mediation as something where we're restricting the number of services we're able to use. In other words, we could say, "We can only run on this particular cloud brand and our existing legacy systems. We can't do any more, because that's going to make things much more complex. We can't spend the money on the additional operational complexity. Therefore, we're going to limit ourselves in our ability to be innovative, because we're not able to leverage the services we need to leverage to kind of take our business to the next level, because we're limiting the services that the developers, the innovators in the company, are able to leverage."

So, we go into a multi-cloud. And we understand the reason why we're doing that, because there's a valid business reason—the ability to leverage best of breed technology to reach out and deal with these various cloud providers and provide whatever best-of-breed services are going to be needed from the cloud providers that offer them up. But still, mediate the complexity by not overcomplicating common services such as security, and operations, and governance and the ability to kind of look at holistic solutions they're able to run across the various technologies.

Now, if you read the press recently, this has names, and people are calling that the super cloud or some people are calling it the metacloud. The idea is that, ultimately, it's about building this layer of technology that sits above the cloud providers, not within the cloud providers, as the ability to mediate complexity. And that's really kind of our only chance of solving this problem.

So, I urge you, number one, to ask yourself, "Do I have a complexity problem?" And if you do, what can you do about it? And looking at technologies out there and looking at different architectural efficiencies, reading my stuff and stuff that other people are writing about this and solving this issue, and then putting a plan together in how you're going to solve the complexity issue before you hit it, and before you see it, and before the wall hits you right in the nose. And believe me, it's easier to solve it at that point before you run into it.

Anyway, if you enjoyed this podcast, make sure to like us, rate us, and subscribe. You can also check out our past episodes, including those hosted by my good friend, Mike Kavis. Find out more at [DeloitteCloudPodcast.com](http://DeloitteCloudPodcast.com), all one word. If you'd like to contact me directly, e-mail me at [dlinthicum@deloitte.com](mailto:dlinthicum@deloitte.com), that's L-I-N-T-H-I-C-U-M.

So, until next time, best of luck with your cloud journey, and you guys stay safe.

**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](http://Deloitte.com/about).

This publication contains general information only and Deloitte is not, by means of this publication, rendering accounting, business, financial, investment, legal, tax, or other professional advice or services. This publication is not a substitute for such professional advice or services, nor should it be used as a basis for any decision or action that may affect your business. Before making any decision or taking any action that may affect your business, you should consult a qualified professional advisor.

Deloitte shall not be responsible for any loss sustained by any person who relies on this publication.

Visit the On Cloud library  
[www.deloitte.com/us/cloud-podcast](http://www.deloitte.com/us/cloud-podcast)

As used in this podcast, "Deloitte" means Deloitte Consulting LLP, a subsidiary of Deloitte LLP. Please see [www.deloitte.com/us/about](http://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](http://www.deloitte.com/about) to learn more about our global network of member firms. Copyright © 2022 Deloitte Development LLC. All rights reserved.