Why Modernizing the Data Layer Requires More than New Tools

APRIL 1, 2021 | DERRICK HARRIS

Podcast   Modernization Best Practices   Data   Tanzu Labs

While architectures and platforms like Kubernetes get a lot of attention in discussions about application modernization, we ignore the data layer at our own risk. How applications and users access data is a concern that gets more important by the day. It's a trend we've seen playing out for a while, as technological concerns around latency and scalability have ceded ground to business-level concerns around compliance, security, and data privacy.

In this episode of **Cloud & Culture**, Kevin Muerle—director of data transformation services for **VMware Tanzu Labs**—shares his insights from decades helping enterprises get a handle on their data. Muerle explains the considerations that organizations need to address in order to match their data systems with their business needs, as well as with their future goals. It's a wide-ranging discussion that touches on AI, cloud, data gravity, GDPR, Hadoop, and much more.

Highlights from the interview are included below, but if you're looking to re-architect your data systems to bring your applications and analytics up to speed, you'll definitely want to listen to the whole thing.

Cloud & Culture

▶

00:00:00 / 00:32:27   ↺15   *1.0x*   15↻   ⚙

Listen on
Apple Podcasts

You can also listen on **Apple**, **Google**, or **Spotify**, or by searching in your favorite podcast app.

Making legacy systems work for you

"Where it's appropriate to fully modernize a legacy system—where we break apart this monolith into more just distinct components or change the actual data-access mechanism from what might be very SQL-oriented access into something that is more event-driven—in time those systems can go away.

"But, quite often, these systems are very tightly ingrained in the corporate infrastructure, so there's audit and compliance activities built around those systems. And, so, they're going to be there for a while; they're often the system of record and they're not going to move in many cases. So we will find ways to interact with those systems to make it more efficient for these modern applications that we're helping customers build.

"So then the third pillar [is] really helping modernize these customers' data platforms. And so that could be ... phasing out these data monoliths. It could be enabling the platform that they're using for analytics or data science. So we help operationalize the idea of data science, machine learning, and AI by transitioning these old legacy platforms into something that's more agile, that's a more modern

data platform ... And then, of course, modernizing these old, expensive platforms, too—so, taking advantage of NoSQL platforms, taking advantage of messaging, et cetera."

The use case matters a lot—especially in the cloud

"A lot of [customers' obstacles are] access-based. Can they just get at the data they need in a timely fashion? Is the data being brought into one system or multiple systems? And then, do they have to do additional steps to make the data available for use? Whether that's reporting or business intelligence or data science or application access, getting the data to a place or in a state that makes it available can be time-consuming and difficult.

"... As they're looking at the combination of multi-cloud, or on-prem and cloud deployments, then access has a different sort of connotation because now you're looking at, 'Where's my system of record? Where does my data live and where is my data going to be used? And how efficiently can I move the data or access the data across these virtual and physical boundaries?'

"And so that's another aspect of data transformation and modernization—making the access to that information efficient. 'If the system of record is on-prem because of audit and compliance reasons, but yet I want to have applications based in the cloud, do I have to go from cloud to on-prem for every single piece of data access, or should I create a cache or an operational data store more local to where my applications are going to be running? Or, do I just need an event-based system where I'm moving events in a stream to get to those applications or to allow those applications to get at the data they need?'"

Future-proof without chasing fads

"[Hadoop] was a good concept, but the reality was that it was really hard to do. Hadoop's really hard to manage. Hadoop's a great place to land information, it's not necessarily a great place to access information from. And so customers and

companies learned over time what you could do with Hadoop, what was efficient with Hadoop, and then what wasn't so efficient. And sometimes they learn very painfully because they jumped into the pool without really looking first. Or they conceptually understood it, but practically there were issues with how fast they could access small bits of information, for example.... So, customers have looked at Hadoop and some backed away from using Hadoop for everything to solve all of their data problems.

"NoSQL was another great example of, 'Hey, let's put everything into NoSQL because I'm going to get really fast access,' but then they realized the ad hoc access to a NoSQL database is maybe not as robust as they were used to with their Postgres system or their MySQL system or something that has a more fully formed and mature SQL interface to connect to Tableau or Cognos or your reporting application of choice.

"And so there's the old 'right tool for the right job' concept. Let's make sure we understand your requirements and then we can talk about the technology to best fit what you're trying to accomplish for today, as well as try to help you be a little bit future-proof moving forward."

Learn more about data transformation

Tanzu Data Services

Understanding the Differences Between RabbitMQ and Kafka

Running Persistent Data in a Multi-Cloud Architecture

Solving Data Sprawl with Modern Postgres

Analytic Workloads from BI to AI with VMware Tanzu Greenplum

About the Author

Derrick Harris is a product marketing manager at VMware.