

CLOUD NATIVE / DEVOPS / TECHNOLOGY / SPONSORED

# App Modernization: Why 'Lift and Shift' Isn't Good Enough

8 Dec 2021 6:00am, by [Jennifer Riggins](#)



*VMware Tanzu sponsored this article for The New Stack.*

Cloud transformation. Agile transformation. [DevOps](#) transformation. These names are problematic. They imply that there's an endpoint. And so do the deadlines from senior leaders and decision-makers that are chasing these so-called transformations.

But these mechanisms for change are more means than ends. Enterprise app modernization must be about applying an ongoing cultural and technical strategy. It's a constant learning process that involves bringing everyone to the table, prioritizing business and then technical needs.

When done right, app modernization — and the cloud migration which typically drives it — empowers teams to release faster, cut costs and scale more easily, while limiting cognitive overload and burnout. When done wrong, it's just money down the drain. So

## There Is No One-Size-Fits-All

As enterprises plan for app modernization, it's important to take a pause because every change you make almost immediately becomes [technical debt](#).

Moving workloads to the cloud is one of the biggest challenges for the tech industry, [Edward Hieatt](#), senior vice president of services and support for [VMware's Tanzu](#), told The New Stack. Many companies have been trying to do it for a long time, but most are still short of gaining value from it.

For instance, [less than half of participants in a 2020 survey](#) by Harvard Business Review said their cloud strategy has been effective in terms of IT management, customer experience, and innovation.


That failure rate is because many IT organizations just lift and shift — simply re-host to the cloud — to meet artificial deadlines.

“Then we may end up in a place where we may build the same as before,” Hieatt said. This migration cannot be rushed.

The chief information officer may feel pressure to achieve some sort of “cloud end state.” But for most of VMware's customers, according to Hieatt, this transition is at least a five-year process — and most aren't going all-in on all-cloud anyway.

“My strong belief is that it's got to be a collaboration or we're going to end up with the usual kind of crappy outcome. And the developers will end up with something they call cloud but doesn't behave like cloud anyway.”

— **Edward Hieatt, VMware Tanzu**

Most larger enterprises embrace the [multi-cloud approach](#), deciding to keep a mix of data centers and private clouds, along with various public cloud vendors. This decision  be based on convenience, the complexities of legacy mainframes, security,

“Step back,” Hieatt said. “Consider the problem in terms of multiple vectors, of what’s important based on each workload, instead of just chasing to the cloud.”

Companies, he added, need to ask themselves, “What are we actually trying to accomplish?”

Even that answer will vary by application. Most enterprises have a hodgepodge of newly written cloud native apps in Java and Spring, fairly modern dotNET and Java apps written a while ago, apps from the early aughts written in more traditional languages, and even some apps run on mid-century mainframes. Plus custom and off-the-shelf apps.

“If you’re an IT team who’s not all aware of what those workloads are, it’s easy to just push them into one environment,” Hieatt said.

#### Sponsor Note

The last few decades delivered great advances in



VMware Tanzu

infrastructure; now, it’s time to build apps that make the most

of those new tools. VMware Tanzu helps you build new apps, modernize existing ones, and evolve your development process around cloud native technologies, patterns, and architectures.

The first step is to discover and make an inventory of all the workloads, he said; a tool like Tanzu can help.

“An IT team is very divorced from the lines of business, from the developers, from the product teams,” Hieatt said. “In the past, their job was just a standardized set of VMs and private cloud and they really don’t know what is running.”

For VMware customers, he added, don’t even have the source code for the applications that are running.

a room of on zoom.

“We are mired in our industry in thinking that all problems are technical decision making — which solution will IT buy? Which vendor will they choose?” Hieatt said. “There are a lot of arguments and budgets and research.”

These are all good things to consider, he said, but they shouldn't be the starting point.

“What I think is constantly lacking is actually bringing together the business stakeholders who will have to maintain and operate and the IT teams that are making the choices,” Hieatt said.

Together, the business and IT stakeholders should look at the next year or two and make a plan, assigning each application or piece of application into one of seven different buckets:

### 1. Retain.

There will be data that can't be moved, or mainframe modernization that can be postponed.

### 2. Re-host.

Move the workload from a virtual machine (VM) to a public cloud. In these cases, lift-and-shift is actually the right fit, especially if on-premise has become too costly.

### 3. Refactor.

Change the application's code without changing its behavior, so there's no change in user experience, but speed or efficiency should improve.

### 4. Re-architect.

Modify the source code, which often means radically changing architecture to microservices. This usually allows it to use the cloud and improve scalability.

### 5. Rebuild.

ly rebuild in a cloud native way.



## 7. Retire.

Deprecate the application.

Business considerations should come first in making these decisions. Prioritization isn't usually about cutting IT costs, as much as looking to increase a line of business or revenue — which usually means being able to make changes quickly, and to scale quickly, securely and rapidly deployable.

But architectural considerations like feasibility still matter. This isn't always about big changes, but considering what is needed to make most workloads “cloud-friendly enough” to re-host or re-platform.


The whole app modernization process, Hieatt said, usually kicks off with about two weeks of collaboration over stickies on whiteboards. “In this world of trying to figure out how to do this faster and better, it starts people-focused. Let the work inform the plan.”

Once you figure out the people side, then you can bring in the automation, tooling and cloud management to pick up speed.

## How Modernizing Can Help Retain Engineers

App modernization is about creating a set of best practices and competency building. It's about continuous learning — which is very attractive for highly recruitable tech workers.

[Kerry Schaffer](#) is senior director of information technology at OneMagnify; her job includes overseeing data center operations. In 2020, OneMagnify had a very tight customer deadline to deliver a feature for taking reservations for the pre-launch of an iconic vehicle.

With microservices hosted by the Tanzu application, Schaffer's team just had to make a few continuous integration/continuous delivery (CI/CD) deployments. The team ered on time and the customer got double the reservations it anticipated.

that in a modern way, we've been able to reuse all that architecture.”

Before its app modernization journey, she described OneMagnify as “developing mostly monoliths.” When the company began to modernize its applications, it prioritized the ones that had the most amount of changes.

“This year, with the talent shortage, if we hadn't modernized our applications and our application stack in 2018 and 2019, there's no way we could've delivered the customer outcomes that we've delivered this year.”  
— **Kerry Schaffer, OneMagnify**

As the teams made those changes, they would take those chunks out of each application and turn them into microservices, tacking that architectural change onto customer requests.

The OneMagnify developers saw a 33% efficiency gain. It took them, on average, six hours less per deployment with zero tickets, versus when the monolithic separation of duties meant tickets at each environment stage.

“In the past, you had Linux and Windows environments — and for tickets, you had to wait for people in Shanghai to wake up. Now they code overnight [for U.S. time] and move to QA and then you ask the client to test that first thing in QA,” Schaffer said.

“It wows them and brings them an extra layer of satisfaction because they are getting their changes at a much more accelerated pace. It has revolutionized the way that we do work for our customers.”

But the most significant impact of application stack modernization for Schaffer is how it cuts down on the amount of after-hours work. At a time of [the Great Resignation](#), this better work-life balance has helped OneMagnify retain staff.

Being cloud native helps with recruitment because everybody wants to be working that's new and modern. It all contributes to OneMagnify consistently featuring on

Cerner, which supplies healthcare information systems, has a tech stack dating in part back 25 years, and is at various stages of modernization. About seven years ago, according to [Greg Meyer](#), one of the company's distinguished engineers, Cerner ran into a 36-hour downtime, which revealed that it could no longer maintain its infrastructure culturally or architecturally.

Five years ago, Cerner began its move to the public cloud.

This cloud transformation to what was then Pivotal Cloud Foundry — which was later bought by VMware and rebranded Tanzu — kicked off as a grassroots effort from Cerner's engineering teams, as opposed to the common top-down mandates.

The engineers began their C-level advocacy campaign by creating value stream maps, based on real measurements, because, as Meyer said, "You've got to prove value for customers and to business as a whole."

Here are the three app modernization value streams the engineers highlighted for their business colleagues:

## 1. Database Loss/Restore.

While the engineers were maintaining the legacy architecture during the migration to Tanzu, they lost the legacy database — it turned out the developer database wasn't doing backups anymore.

By switching to Tanzu self-service, they were able to go from a week down with all the data lost, with four developers trying to fix it, to just four hours of the database down, no data lost, and only one teammate working on it.

## 2. Improved Patching.

Prior to modernization, it would take an average one month and two to three people to perform patching. With Tanzu, they would use a standard common data environment and one person would click a button to trigger an automated release.

## 3. Reduction of All Node/Cell Loss.

“But Tanzu figured it out in one minute, sent alerts out at 5:30 in the morning,” Meyer said. “Some 50 minutes later, got everyone on the phone, Tanzu had already rectified it automatically in under 20 minutes.” Now the engineers are able to achieve high availability of four or five nines.

These are just three of at least eight business value streams discovered by the Cerner engineers’ proof of concept. This granted them the ability to move forward — but, a year into their journey, they wanted to get more teams involved. That comes down to showing outcomes.

“This whole transformation is all about the outcomes,” Meyer said. “If you can’t get good outcomes from a business unit or customers, why are you doing this in the first place?”

Once the engineers were able to move past the early proofs of concept, he said, they were able to get past architectural changes to better embrace cloud native culture, becoming more agile with CI/CD processes.

Previously, Meyer said, the developers and the DevOps teams were two separate entities: “When we moved over to cloud native culture, we really turned into embracing a fully cross-functional team of engineers and DevOps working together, a process flow from one environment to another, to another.”

Now, the same people work together throughout the software lifecycle, from development to production.

The first stage of Cerner’s app modernization finished going live about three years ago. But they are “definitely not stopping,” Meyer said, and are currently looking at ways to improve CI/CD processes and observability. After all, app modernization is a journey, not a destination.



Tanzu is a sponsor of The New Stack.