Application Modernisation with Speed

How Telecommunications and Financial Services Can Transform Fast

vmware®

Table of contents

Introduction

In forcing the world to move to a remote or contactless way of working, the COVID-19 pandemic has exposed two truths. First, even the world's largest organisations can move quickly when motivated, and second, being adaptable to change is the key to survival in the business world.

Large businesses form much of the structure and influence the way modern societies operate. However, as we've seen in the past, they must have the flexibility to change with evolving technologies to keep pace with evolving needs. Over the past decades, certain industries have moved more rapidly than others, adopting new technologies and being driven by up-and-coming start-ups that can move with agility and have innovation at their core. For complex and highly regulated industries like financial services and telecommunications, this rate of change can bring significant value for their customers. But to truly achieve these results, they need to move quickly, improving the speed to market of new technologies and services while embracing a DevOps approach to application development and delivery.

There are many leading businesses already achieving this. How? By replatforming quickly, then refactoring at their own pace to modernize existing apps alongside the development of new ones, all using cloud-based infrastructure.

In this white paper, we'll cover our best practices for how organisations in the financial services and telecommunications industries can be just as adaptable as technology start-ups. With the right processes and approaches to application modernisation, organisations of any size can move quickly and bring legacy apps into the future.

Why you need application modernisation

Applications are intrinsic to the operations of modern businesses. For those with greater complexity, sensitive customer data, and responsibility for the operations of essential services, the management of these applications can become complex. Often, organisations in industries like financial services and telecommunications operate with years of legacy infrastructure, using this to run services essential to their operations and customers' experiences.

Most developers work on existing applications: products and services that have been built, maintained, and updated over long periods of time. Normally, these apps exist as a web of tightly coupled, sparsely documented systems. This is where there can be a downfall; these systems often have strict security requirements due to the nature of highly regulated industries.

Over time, these large organisations develop layers of manual processes designed to minimize risk and ensure compliance. This is even further amplified for the likes of highly regulated industries, where legislation or standard practices can influence access to and storage of data or other information. As a result, software releases are often infrequent and full of high ceremony, requiring significant effort to get across the line. For businesses looking to compete in a digital age, this rate of development can't keep up with new, agile competitors that move without the burden of historic infrastructure.

But what can be done? The question you need to ask is, "How can I refactor my most important apps, so that I can get new features to production faster?"

VMware has helped organisations across the world tackle this challenge, and with great results. How? With a modernisation methodology that leverages cloud-native patterns and continuous delivery (CD) automation across your existing application portfolio.

This methodology takes an iterative approach that allows you to tackle app modernisation efficiently and quickly. Rather than trying to achieve everything in one enormous project that would drag on for years, it's an incremental approach that starts in a small, focused way. We pair with you to update and automate processes incrementally, while transforming your applications.

There are many reasons why organisations embark on a modernisation journey, yet they typically fall into two categories:

1. **Portfolio transformation** is a broad and strategic effort to migrate, refactor and transform an existing portfolio of applications to cloud-based technology. Organisations with this objective are typically on their way to the cloud but must address high levels of technical debt in their app portfolio.
2. **System modernisation** is about migrating core, business-critical systems that are expensive to update, technically complex and under active development. Cloud technologies and cloud-native development patterns provide an opportunity to solve such problems.

Both have unique challenges and require a considered approach to move application management and development to cloud-based infrastructures.

Who you need to be successful

Successful projects rely on three things: people, process and technology. Although it's important to get all three right, in our experience the "people" part of the equation is the most important—and often hardest—to get right.

In this day and age, we have sufficiently advanced project management techniques and technological capabilities to the point where solving almost any business challenge—from a purely technical standpoint—can be relatively straightforward. But whether or not the project ultimately succeeds in achieving its goal relies heavily on the people involved.

The first thing is to ensure that your project has executive buy-in. Modern software development relies on a continuous and iterative approach, which allows developers to launch a product and make incremental improvements based on feedback from real users. This ensures the product is as fit-for-purpose as possible, at all times.

Conversely, large organisations like banks and telecommunications providers have historically focused on projects with a defined start and end date. The relevant executives should understand—and be on board with—continuous development to ensure they can provide the right environment for project success.

Citigroup recently embarked on a long-term digital transformation to help get products to services more quickly, and compete with fast-moving fintechs in the space. Since working with Cloud Foundry, they have experienced a 57 percent increase in the speed of development. But to get there, CEO of Global Consumer Banking observed that the company had to first grow its perspective, skills and capabilities while becoming more open to continuously learning new things. If the executive team was not bought in from the beginning, efforts to replace incumbent waterfall practices with agile development would have hit a brick wall.

The second thing is to ensure you have at least some representatives of your core users in the room. If you're launching, for example, an application to be used by employees, it would make sense to run some focus groups to ensure the product will have a high adoption rate to justify the project.

Where the end user is the customer, companies should aim to have one or more customer champions on the team who understand the key pain points and will ensure the project stays on track. Larger organisations with more scale might also choose to run focus groups or beta tests where customers can give direct feedback on the new application.

Finally, it's important to get the project team's structure right. Though there's no one-size-fits-all team structure that works for all organisations and cuts across all key disciplines, there are some core roles that are essential:

- The **product owner** has an acute understanding of the business need, as well as the needs of the end user (the aforementioned customer champion). They must keep the team on track and solve issues as they arise.
- The **product anchor** is a technical leader who works closely with the product owner to guide technical practices and oversee quality, and has a hands-on role in technical work.
- **Developer/s** are skilled engineers who understand the existing app, as well as the target state and work on building the solution.

Our approach

Speed to market is important when launching a new app, but it shouldn't come at the expense of product quality. Our approach to modern app development helps in that regard—this is where we introduce things like *test-driven development*, *user-centered design* and *pair programming*.

But first, our recommendation to businesses is to consider the regulatory and financial risk. Businesses in highly regulated industries—like telecommunications or financial services—should take stock of the regulatory environment and ensure that any new apps are compliant within this environment. These companies deal with petabytes of customer data, so it's important not to run afoul of far-reaching regulations like the EU's GDPR or Australia's MDBN.

Businesses should then assess where they can afford to take some financial and technical risk. This usually means starting with applications as far away from the core system as possible, so that any issues that arise with the new application can be contained and won't bring down the whole system. Teams must have the freedom to fail, which is why an incremental approach to application development is crucial.

From here, we can apply several practices to ensure product quality is high, while iterating quickly:

- Test-driven development is the best way to maintain software release speed and avoid legacy software slowing you down. In a nutshell, it involves running automated tests throughout the development process and fixing the code as you go, rather than running a single test once the product is finished.
- User-centered design prioritises contact with end users to determine value. It emphasises designing the product around how the user can, wants, or needs to use it, rather than seeking to change the user's behaviour to fit the product.
- Pair programming involves two people working on code in sync and switching back and forth to take turns on each other's work. This keeps the pair engaged and ensures that when they hit a problem, they can use the power of two brains to solve it.

When working on projects that require more speed, we employ a practice called Extreme Programming. With this, we create collaborative, cross-functional, self-organising teams that deliver software in an incremental and iterative way, at a consistent speed and quality, in the face of changing requirements.

In 2020, Fiserv needed to deliver financial relief to thousands of small businesses at the peak of the COVID-19 crisis. When the U.S. government passed the CARES Act, Fiserv was thrust into the middle of a complex workflow between the federal government and thousands of banks around the United States.

Fortunately, Fiserv had already embarked on a modernisation journey with VMware Tanzu Application Service. This meant that once Keith Fulton, SVP and CIO of account processing, built his cross-functional team of 15 developers, they were able to write more than 100,000 lines of code and ship 436 product releases in only 28 days. Beyond the tech excellence his team displayed, the tangible outcomes were phenomenal: the new workflow helped secure 18,000 loans worth more than USD $1.4 billion and keep thousands of people in a job during the height of the crisis.

How to get started

For large organisations, particularly in the financial services and telecommunications sectors, it can be tricky to identify where to start when it comes to app modernisation. Many have complex legacy systems, architectures and critical infrastructure; disrupting this without the right process in place could lead to challenges down the track. But in following the right process, organisations can embark on a modernisation journey with low risk and high reward.

Recently, a multinational investment bank was looking to leverage its scale and move into retail banking. Their business analysis showed them the best path to follow was to go to market with an online-only neobank offering, so they chose a strategic tech partner to launch a digital credit card.

As their developers started building out the credit card application, they ran into some challenges on the infrastructure side. The bank's foundational applications didn't seem capable of delivering the key functionalities they knew were necessary to deliver the experience customers would expect—namely self-service provisioning, zero downtime and built-in scale and security.

The bank's app development team worked with VMware's Tanzu team on a proof-of-concept for one of the card's core service applications, built on Tanzu Application Service. The VMware team outlined the key success criteria with the bank's team and was able to systematically deliver each component to have a pilot of the credit card ready within three months.

There are four main elements to a successful application modernisation initiative:

1. **Start small.** No matter how large your volume of apps, identify a business unit as a starting point and a few key applications to begin.
2. **Automate everything.** Continuously integrate and deploy using test-driven development processes to reduce SDLC costs and manual processes.
3. **Learn by doing.** Use the hands-on process to experiment, build skills in the team, and measure results as you go, refining to find the right approach for your organisation
4. **Iterate and adapt.** Break down your approach into small segments and continuously improve on small slices of complex systems.

The advantage of our approach means organisations can replatform core applications to cloud-native infrastructures, then refactor at their own pace. Nearly any custom-developed application created in the last decade can be replatformed to adhere to the minimum 12 possible factors for cloud while preserving functionality. From there, applications can be refactored in order of priority while reaping the benefits of cloud.

Often, large organisations have applications or services that run as monoliths. They've served your company well, but are complex and near impossible to scale. The goal of modernisation is to make testable changes to applications and make them run natively on the cloud, aiming for 15-factor compliance. What this means is breaking monoliths down into microservices.

But how do you tackle these? After prioritising the apps in your chosen portfolio using methods like SNAP analysis, there's still likely to be a group that need immediate attention. It's best to begin with applications that will have the most impact—and fast. This often means those that impact the customer experience and have a flow-on effect for revenue.

For financial services and telecommunications organisations alike, it can be best to start with customer-facing applications. These have the greatest immediate impact and can often be less complex than existing internal infrastructure. Given that 5G requires microservices and containers to operate, applications designed to enable activity on 5G networks should be prioritised when coming close to launch in market. Many banks have

also used their mobile banking applications as the testbed for modernisation and agile processes to great success, gaining the trust of customers and then adopting these practices more broadly across the business.

World-leading telco British Telecommunications (BT) is focused on delivering outstanding customer experiences through super-fast broadband, TV and mobile. To achieve this, it's constantly improving its applications and finding ways to deliver new functionality and customers, particularly as it looks to roll out 5G in the future. To achieve a more agile application base, BT partnered with VMware to deploy a microservices architecture for key application components and services. Using VMware's suite of solutions—including VMware Tanzu Application Service and VMware vSphere7—BT can better service its 30 million customers. It's able to operate a single stack that unifies infrastructure, allowing the business to manage virtual machines and containers with a single set of people, processes and systems. This enables BT's developers to add value to the business and improve the deployment of applications for its channels.

Tracking success

Like with any good project, you need to use objectives that are specific, measurable, achievable, relevant and timely. And though agile projects must have a degree of fluidity, there are some metrics project leaders can stay on top of to ensure its successful delivery.

When looking at establishing your project's objectives and key results (OKRs), it's important to avoid choosing an OKR just because another company did it. To accurately measure the success of your project, you must create meaningful key results. Within an agile framework, these OKRs could look like:

Velocity	• Feature delivery frequency • Feature KP/usage metrics • End user feedback loops
Development time	• Time from ideation to end user consumption • Time from ideation to development • Time from development to end user consumption
Test coverage	• End-to-end testing success rate • Regression testing success rate
Bugs found	• Mean time to repair • Severity and components of bugs found

Project leaders should ensure they're clear on the business's mission and vision and have time-boxed objectives that align to them. Ensure OKRs are meaningful and modify them if the project requires you to do so. And finally, challenge yourself—if you score 100 percent on your OKR, they were probably too easy to begin with. Remember: failure is an important component of agile software development, provided your team learns and adjusts. Without pushing in this way, it will be challenging to deliver truly innovative ways to make a difference to your end users' lives.

Next steps

Organisations in the financial services or telecommunications industries typically have a wealth of data at their disposal that can help them create amazing experiences for their customers and deliver a competitive advantage in their respective markets. But to use it in a constructive and sustainable way, they must take a modern approach to software development that supports a culture of continuous learning and innovation.

Take a look through the following to learn more about how we partner with organisations like yours:

- *Application Modernization at VMware Tanzu* [Webpage]
- *5 Steps to Financial Services App Modernization* [Webpage]
- *App Modernization with .NET Core: A Journey from Mainframe to Microservices* [Webinar]
- *How Communication Helps T-Mobile Keep its Applications Up* [Webpage]
- *Monolithic Transformation: Using DevOps, Agile, and Cloud Platforms to Execute a Digital Transformation Strategy* [eBook]
- *Swisscom: Bringing New Digital Services to Market Faster* [Webpage]
- *Microservices eBook: Migrating to Cloud-Native Application Architectures* [eBook]
- *The Top Ten App Transformations* [Blog]
- *Citi: Accelerating Transformation at a 200-Year-Old Institution* [Webpage]
- *How a Large Fintech is Breaking Up Monolithic Applications into Microservices* [Webinar]
- *Application Modernization Recipes* [Technical Deep Dive]
- *Application Modernization Should Be Business-Centric, Continuous and Multiplatform* [Gartner Report]

If you have insights about application modernisation you'd like to share, please contact us at *tanzu@vmware.com*.

VMware, Inc. 3401 Hillview Avenue Palo Alto CA 94304 USA Tel 877-486-9273 Fax 650-427-5001 vmware.com