

# The Enterprise Guide to Continuous Application Modernization

Transform your AWS applications  
for long-term success in the cloud

# Table of Contents

<b>Introduction</b>	<b>03</b>
<b>Why Modernize?</b>	<b>04</b>
<b>Implementing a Continuous Modernization Strategy</b>	<b>08</b>
<b>Step One: Set Goals for Modernization</b>	<b>10</b>
<b>Step Two: Understand Your Applications</b>	<b>12</b>
<b>Step Three: Choose the Optimal Modernization Approach for Each Application</b>	<b>13</b>
<b>Step Four: Observe and Optimize</b>	<b>16</b>
<b>Step Five: Repeat</b>	<b>18</b>

# Introduction

Many organizations today are struggling to get the full value of migrating their enterprise systems to the cloud. The reason, according to consulting firm McKinsey & Company, is that companies tend to “fall into the trap of confusing simply moving IT systems to the cloud with the transformational strategy needed to get the full value of the cloud.”<sup>1</sup> In essence, organizations are simply outsourcing their data centers, which prevents them from maximizing the benefits of the cloud.

That’s where the concept of modernization comes in. Moving to the cloud is only one—albeit essential—component of modernizing your infrastructure, applications, and approach to software development.

Why move to the cloud in the first place?

Among the top reasons are increased business agility and speed—and the competitive advantage that offers. Another compelling driver for many companies is the increased productivity of their workforce, while a number of organizations shift to the cloud to innovate and transform their business with modern technologies, reimagining their applications within a modern framework.

That’s why simply getting to the cloud should never be thought of as the finish line for your modernization efforts. Creating a new application in Amazon Web Services (AWS) or “lifting and shifting” legacy applications to AWS doesn’t automatically result in applications that can take full advantage of the elasticity, resiliency, ease of deployment and management, and flexibility that AWS offers. Nor does it automatically modernize your culture and processes to enable high-performing software development.

Modernization means taking your application environment in the form that it is today (most likely legacy and monolithic) and transforming it into something that is more agile, elastic, and highly available. In doing so, you can transform your business into a modern enterprise. However, modernization is not easy or trivial, nor does it happen overnight.

This ebook introduces a pragmatic and continuous approach to application modernization as part of a greater modernization effort on the AWS platform. You’ll learn about the choices, trade-offs, and benefits of various approaches to continuously modernizing your applications, and how you can achieve your goals for transformative IT and business outcomes.

1. “Cloud adoption to accelerate IT modernization,” McKinsey, April 2018

# Why Modernize?

For the vast majority of companies that were not “born” digital—and therefore not primed to take advantage of all the latest cloud technologies—there’s little choice about whether to modernize or not. Remaining stuck in the past in your infrastructure, technology, applications, and approach to software development means ceding the race to more agile, responsive, and innovative competitors. Large legacy businesses today must evolve quickly to remain competitive and relevant.

Rather than approaching modernization as simply a way to survive in your industry, it’s more productive to view it as the primary way to achieve business agility in the modern era. Modernization and transformation power your company’s ability to deliver measurable and sustainable outcomes such as:

- Improving customer experience and loyalty with a differentiated customer experience
- Disrupting the industry and the competition
- Introducing innovation at a rapid pace
- Optimizing the cost, speed to market, and volume of new digital experiences
- Supporting mandated changes such as regulatory compliance in a timely way

The question is not whether to modernize at all, but what and how to modernize. Should you move your core applications to AWS? Which business areas or applications will deliver the largest impact or highest value for the business when modernized? Are there applications that should not be modernized because they have reached the end of their useful life? What are best practices for breaking apart monoliths?

**81% of organizations have at least one application or a portion of their computing infrastructure in the cloud.**

Source: [“Cloud Computing Survey 2020,” IDG, June 2020](#)

To answer these “what” and “how” questions about your legacy applications, you need two fundamental things: an understanding of your modernization options and deep insight into your applications. Read on for a strategy that incorporates both of these must-haves.



**66%**

**of software development leaders are running software in the cloud compared with 46% of laggards.**



**96%**

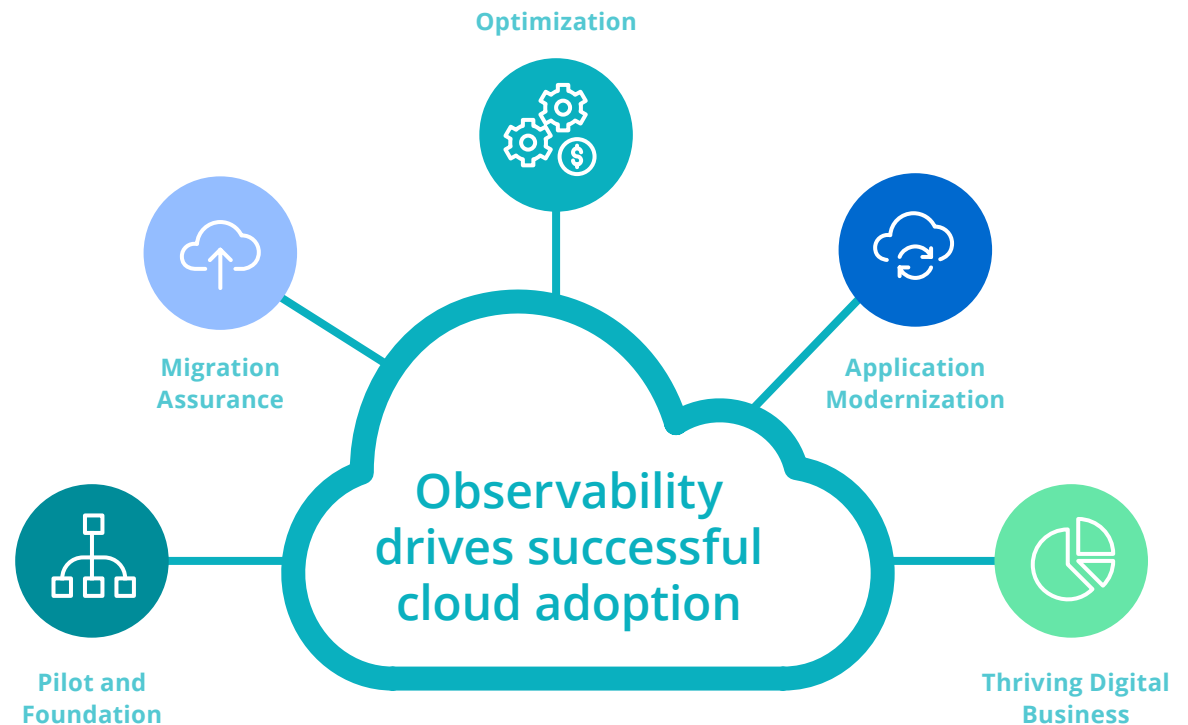
**of leaders have adopted a cloud native approach to software development compared with 24% of laggards.**

Source: [“Deeper Than Digital: Why and How More Perfect Software Drives Business Success”](#), New Relic, June 2020

# Navigating the Stages of Cloud Adoption

Before diving into the details of application modernization, it's helpful to understand how companies are adopting the cloud today. Although some companies are born in the cloud, most companies, especially large enterprises, move to the cloud in stages:

- **Pilot and Foundation phase:** In the Pilot and Foundation phase, companies build new applications in the cloud, try out new services and learn new skills, define standards and procedures, and establish observability best practices. Laying the groundwork early on sets up the organization for successful cloud adoption.
- **Migration Assurance phase:** The migration phase includes three main focus areas: before, during, and after the migration effort. Profiling and baselining your existing infrastructure and application usage, as well as architecture and complexity, help companies form a guide to migration. Using instrumentation best practices and identifying issues and roadblocks accelerate migration. Validating results against pre-migration baselines and key performance indicators proves a successful migration.



- **Optimization phase:** This phase is about understanding and realizing the benefits that the cloud offers, governance of over- and under-provisioning resources, and using observability to make data-driven, right-sizing decisions based on historical usage and real-time demand.
- **Application Modernization phase:** With a full appreciation of how the cloud can benefit their business and a commitment to move beyond basic cloud-based architecture deployments, this phase is where companies transform their applications and organizations to capture those benefits. Modernizing applications through replatforming and refactoring transforms them to reduce technical debt, improves each application's value and performance to the business, and supports the shift to DevOps in an organization.
- **Thriving Digital Business phase:** In this phase, companies can deliver a digital customer experience that is perpetually available, continuously flexible, and high quality. To deliver such an experience, outperform market competitors, make internal operations efficient, and work with agility and speed, companies need to empower their software and operations teams with complete visibility and business context.

## Modernization and DevOps: Better Together

Application modernization enables DevOps success and vice versa. As legacy applications and their environments are modernized, DevOps teams can spend more time on developing and delivering new features and less time overcoming friction in the software lifecycle of existing systems.

Find out more about doing DevOps right:

[DevOps Done Right: Best Practices to Knock Down Barriers to Success](#)

[DevOps Without Measurement Is a Fail: How to Measure and Track the 5 Critical Drivers of DevOps Success](#)


# Implementing a Continuous Modernization Strategy

To deliver better quality software faster, and with greater frequency that enables the business to stay agile and respond rapidly to changes in the market, IT organizations must transform their culture, tools, and processes. The DevOps movement was born from this need, and includes core tenets such as collaboration, transparency, automation, and continuous delivery of software.

However, for the majority of businesses, legacy applications are still an essential part of the company's value chain, providing core capabilities and data for the business. The architecture, infrastructure, and technology used by legacy applications do not easily lend themselves to modern software development, limiting the ability of your IT teams to deliver software with the speed and stability the business requires.

The answer is to incrementally—but continually—modernize your legacy applications. By gradually transforming them, you can reduce technical debt, improve the value and performance of each application to the business, and support the shift to DevOps in your IT organization.

**Here's a pragmatic strategy for continuous application modernization:**



**“Organizations have to change, now... the path forward is clear: Accelerate digital business or risk the survival of the organization.”**

Source: [“Why Now Is the Time to Accelerate Digital,”](#)  
Smarter with Gartner, September 2020



Before we explore each step in more detail, there's a critical success factor you need to keep in mind: **your data**. Not just any data, but contextual information about your infrastructure and applications and their impact on your customers and your business. Without this data, you're essentially going into the modernization process blind, with limited ability to minimize risk; avoid missteps; and accurately assess, predict, and confirm improvements.



# Step One: Set Goals for Modernization

How will you measure the success of your modernization efforts? Simply moving a certain percentage of applications to AWS is not a good measure of success in and of itself. Instead, it's better to link modernization efforts to measurable business and IT goals, such as greater agility and faster time-to-market with new capabilities. This will allow you to show incremental progress and success as you continuously modernize your applications.

One way to get started is to identify the highest priority business-critical applications and explore how they would benefit from better performance, scalability, and reliability by leveraging the economies of scale that AWS delivers. For instance, depending on your industry, target market, and other variables, you can identify how modernizing critical applications could impact areas such as profitability, revenue, churn, market share, differentiated customer experience, and more.

Once you define how your current applications impact business success, then you can link modernization goals and priorities back to business outcomes. For example, modernizing a billing application so that it can integrate with a new e-commerce system could drive greater customer satisfaction and reduce customer service costs, as fewer customers need to contact the business about problems with their bills.

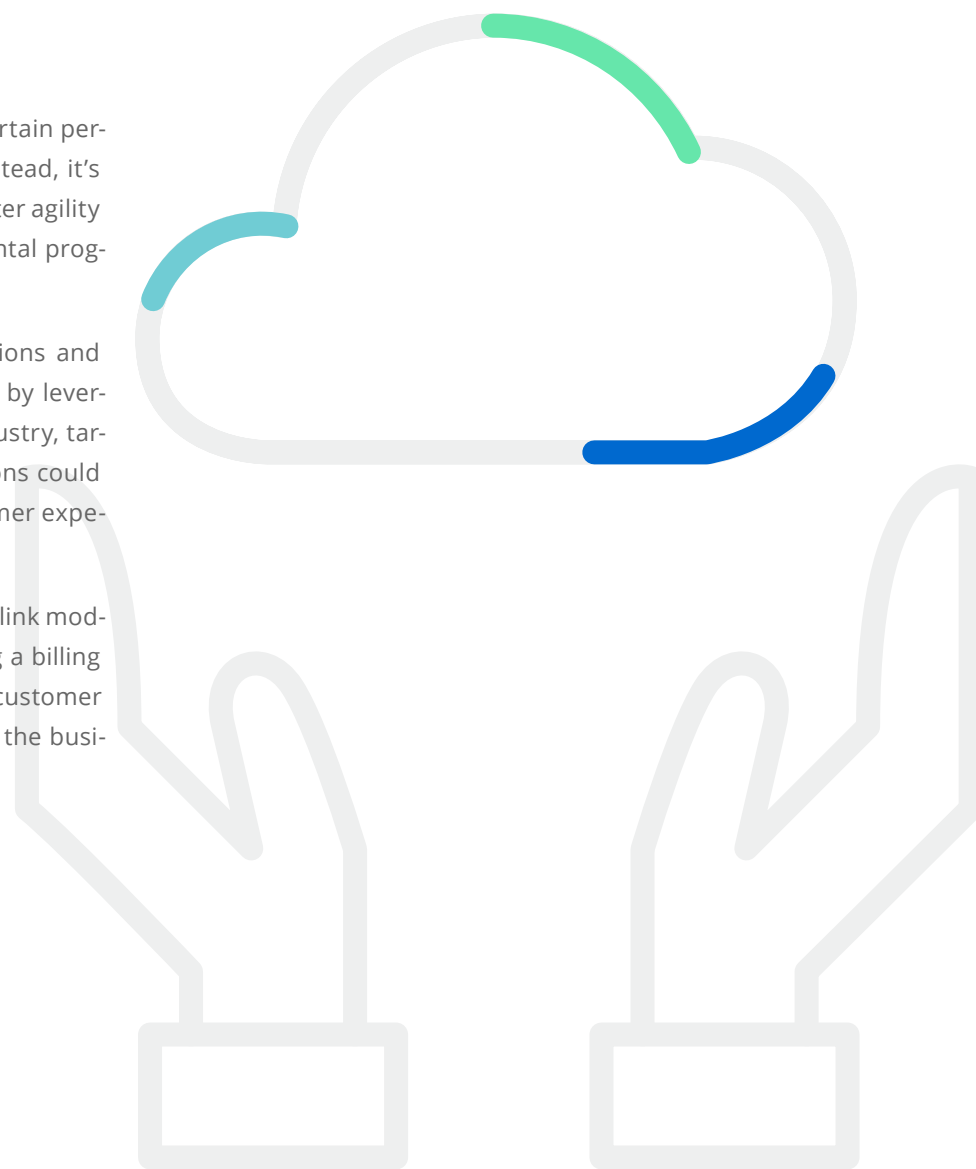


TABLE 1

SAMPLE GOALS	IT IMPACT & KEY PERFORMANCE INDICATORS	BUSINESS IMPACT & KEY PERFORMANCE INDICATORS
<p><b>Increase revenue</b></p>	<p>Respond faster to business needs and market opportunities. KPIs:</p> <ul style="list-style-type: none"> <li>• Lead time for changes</li> <li>• Frequency of code releases</li> </ul>	<p>Drive revenue in current business models and adopt new models that create additional revenue streams. KPIs:</p> <ul style="list-style-type: none"> <li>• Average revenue per user</li> <li>• Conversion rates</li> <li>• User base growth rates</li> </ul>
<p><b>Improve profitability</b></p>	<p>Optimize IT costs through better resource utilization and higher quality software (less time spent fixing problems). KPIs:</p> <ul style="list-style-type: none"> <li>• Application resource usage</li> <li>• Mean time to resolution</li> <li>• Application error rates</li> </ul>	<p>Reduce customer service costs due to software problems and optimize IT spend to improve margins and profitability. KPIs:</p> <ul style="list-style-type: none"> <li>• Customer service costs</li> <li>• Cloud spend</li> </ul>
<p><b>Deliver improved customer experiences</b></p>	<p>Improve quality and performance of applications and deliver new capabilities faster. KPIs:</p> <ul style="list-style-type: none"> <li>• Uptime</li> <li>• Application response time</li> <li>• Application error rates</li> <li>• Lead time for changes</li> </ul>	<p>Improve customer satisfaction and retention. KPIs:</p> <ul style="list-style-type: none"> <li>• Customer satisfaction survey results</li> <li>• Churn rates</li> <li>• Increased volume</li> </ul>
<p><b>Grow market share</b></p>	<p>Outpace competitors to deliver innovative new features faster. KPIs:</p> <ul style="list-style-type: none"> <li>• Lead time for changes</li> <li>• Frequency of code releases</li> <li>• Session count by geography</li> <li>• Total users by geography</li> </ul>	<p>Total users by geography Increase customer base and share of market. KPIs:</p> <ul style="list-style-type: none"> <li>• User base growth rates</li> <li>• Market share</li> </ul>

# Step Two: Understand Your Applications

The next step in the strategy is to get a clear understanding of each application and its interdependencies. To do this, take a baseline measurement that helps you figure out how each application currently performs. This gives you the foundation for making data-driven decisions as you create your initial road map for modernization. With this understanding, you can identify and prioritize applications to modernize and determine how you want to approach modernizing each application.

To achieve this, you need an observability platform that delivers a single source of truth across your applications and infrastructure and supports both pre- and post-modernization environments. Otherwise, you'll be relying on guesswork, anecdotes, and "gut feelings" to understand how your application is functioning now and whether or not modernization improves it.

Once you have your full-stack observability in place, begin capturing data about your applications that will help you understand the following characteristics for each application (see Table 2 at right).

With an objective and holistic view of the application—including its running state, processes, infrastructure, business KPIs, codebase, and issues—you'll now be able to make decisions about how best to modernize the application.

TABLE 2

<b>Application Infrastructure</b>	<ul style="list-style-type: none"><li>• How are resources being used?</li><li>• Which resources are being used by this application?</li><li>• What chronic resource issues exist (e.g., over- or underutilization)?</li></ul>
<b>Application Quality and Performance</b>	<ul style="list-style-type: none"><li>• How is the application performing?</li><li>• What is normal baseline performance?</li><li>• What errors does the application currently have?</li><li>• What are the application dependencies?</li><li>• How is the end user experience?</li><li>• Where is your application spending most of its time?</li></ul>
<b>Impact on the Business</b>	<ul style="list-style-type: none"><li>• How does performance and reliability impact revenue?</li><li>• How much time do users spend on the application?</li><li>• How does the application impact conversion rates or order value?</li><li>• Do errors/downtime impact customer service costs?</li></ul>

# Step Three: Choose the Optimal Modernization Approach for Each Application

Now it's time to use all the data you've gathered to make an informed decision about how to begin modernizing or further modernize each application. Although there are six industry-recognized approaches that you can consider, only the last three—rehost, replatform, and refactor—involve modernizing the application (see Table 3 on the following page).

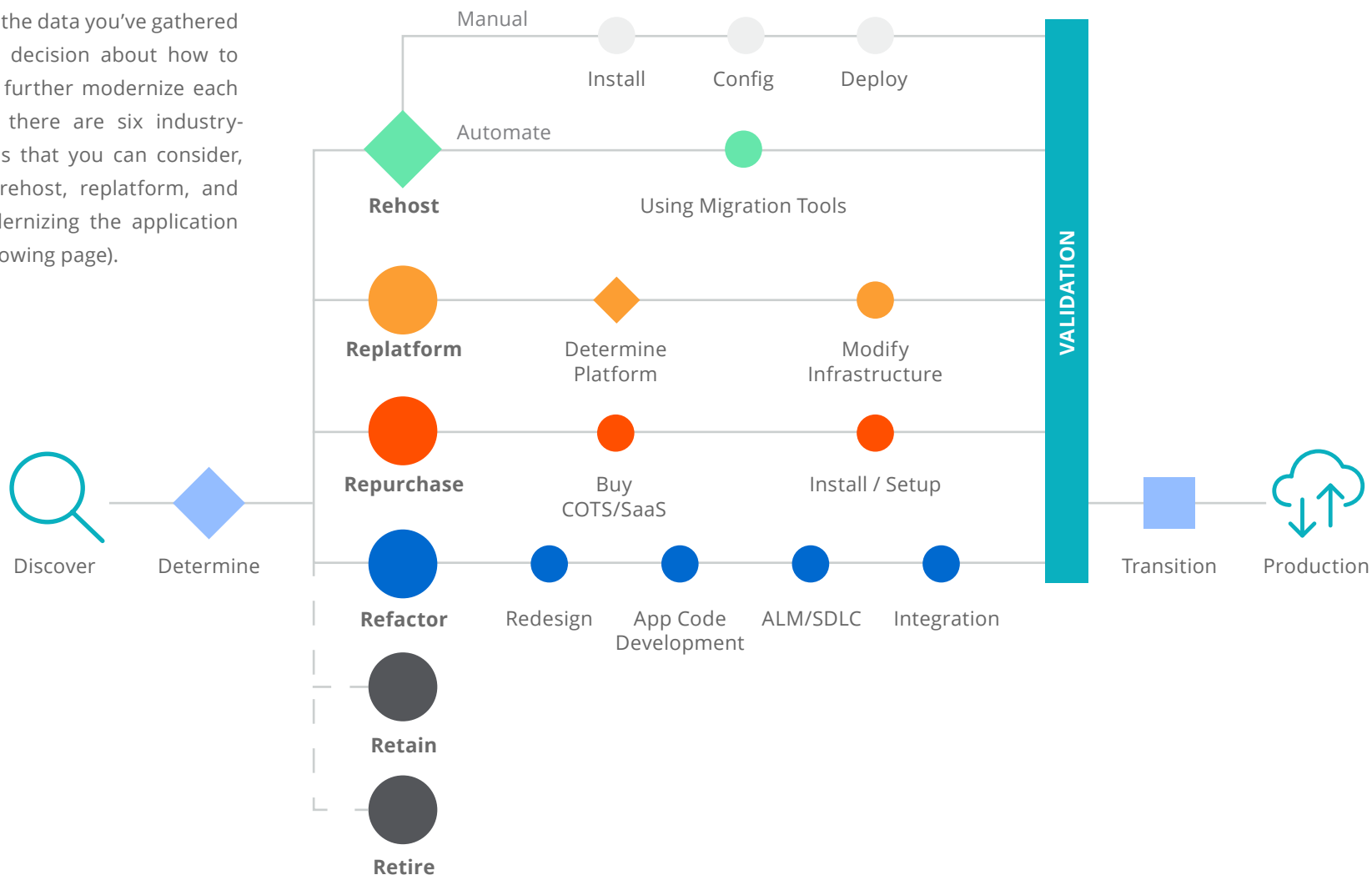


TABLE 3

APPROACH	RISK/ IMPACT	REASON TO CHOOSE THIS OPTION	ACTION REQUIRED
<b>Retire</b>	Low/Low	The application no longer provides value to the business.	Remove the application altogether.
<b>Repurchase</b>	Low/Low	The application is an undifferentiated support system such as email.	Replace the current application with a new, off-the-shelf package.
<b>Retain</b>	Low/Low	The application is not worth modernizing currently, because it's of low value to the business and low impact on IT.	Keep the application and the infrastructure it runs on as is and revisit the decision in the future.
<b>Rehost</b>	Low/Low	Moving the application to a more modern environment would create benefits such as cost savings, performance gains, or ease of operations.	Migrate to a more modern infrastructure such as cloud with no changes other than those related to the rehosting.
<b>Replatform</b>	Low/ Medium	Moving the application and modifying some of the infrastructure it uses would generate additional benefits such as time and resource savings.	Take an existing component of an application and move it to a managed service (such as a database service) with no changes to business logic.
<b>Refactor</b>	High/ High	Re-architecting the application to take advantage of modern cloud services, architectures, and technologies would fundamentally improve the quality and performance and enable the rapid delivery of innovative new features.	Make code-level changes to improve performance, scalability, manageability, reliability, and resource usage as well as developer productivity and the ability to innovate rapidly.

## STEP 3 CONTINUED

Keep in mind that with an iterative modernization strategy, you can start with one approach, reap some initial benefits, and then continue to modernize through other approaches to obtain more benefits. For example, you could rehost an application with no code changes, then replatform it to swap out the current database for a cloud-based database service. With a different, higher priority application, you could decide to go straight to refactoring it to take advantage of additional cloud technologies.

	APPROACH	TIME	COST	AGILITY
LOW	Retain	■	■	N/A
	Retire	■	■	N/A
	Rehost	■ ■	■ ■	■ ■
	Repurchase	■ ■ ■	■ ■ ■ ■	■ ■ ■
	Replatform	■ ■ ■ ■	■ ■ ■	■ ■ ■
HIGH	Refactor	■ ■ ■ ■	■ ■ ■ ■	■ ■ ■ ■

MIGRATION COMPLEXITY

# Step Four: Observe and Optimize

It should now be clear that you need to fully understand how your applications perform before you can decide how and whether to modernize them. The baseline picture of application performance gives you data not only to inform your decisions, but also to serve as a comparison during and after your modernization effort. It also helps you identify any issues that you need to address before you begin the modernization.

After the initial modernization iteration is complete, you can demonstrate success by comparing your previous baseline against current performance, customer experience, and business outcome data. Ideally, you'll see improvements and identify areas where further modernization and optimization can help.





TABLE 4

SAMPLE KPIS TO TRACK BEFORE / DURING / AFTER MODERNIZATION	
<b>Application Infrastructure</b>	<ul style="list-style-type: none"> <li>• CPU percentage</li> <li>• Memory used percentage</li> <li>• Average load</li> </ul>
<b>Application Quality and Performance</b>	<ul style="list-style-type: none"> <li>• Availability percentage</li> <li>• Page load time</li> <li>• Throughput</li> <li>• Perceived response times of key transactions</li> <li>• Application response time</li> <li>• Database response time</li> <li>• Apdex</li> <li>• Application error rates</li> </ul>
<b>Business and IT Impact</b>	<ul style="list-style-type: none"> <li>• Cost to operate</li> <li>• Lead time for changes</li> <li>• Mean time to resolution</li> <li>• Conversion rates</li> <li>• Average revenue per user (ARPU)/Revenue</li> <li>• Churn</li> </ul>

# 94%

of More Perfect Software leaders (those companies that have made the most progress delivering software excellence) say observability is key to developing software compared to 56% of laggards.

Source: [“Deeper Than Digital: Why and How More Perfect Software Drives Business Success,”](#) New Relic, June 2020

# Step Five: Repeat

This last step is neither an actual step nor the last activity of the modernization process; rather, it's a reminder that modernization is continuous, and that even as you gradually modernize your applications, there will always be new technologies and capabilities to incorporate and support.

However, with each cycle of modernization, your organization is making significant strides in enabling and fostering digital transformation and the culture, process, and technology changes that must happen as part of it.

New Relic can support you before, during, and after each modernization iteration, providing data-driven insights to help you make smart business decisions when considering which applications to modernize and the right approach to take for each. New Relic One, the most powerful cloud-based observability platform, is designed for everything that makes modern software great, with a connected, real-time view of all your operational data in one place. Using New Relic One throughout your modernization efforts helps you reduce risk, set priorities, and demonstrate your impact at every step.

Using New Relic One together with our Cloud Observability Framework helps you reduce risk and accelerate your cloud journey in any phase—from pilot to migration to optimization and modernization to thriving digital business. To learn more about how New Relic One and New Relic Cloud Observability Framework can help you achieve your modernization goals, read our solution sheet [AWS Cloud Adoption with New Relic](#).





Visit [newrelic.com/aws](https://newrelic.com/aws) to learn about New Relic's partnership with AWS, our integrations, competencies, and case studies.

