Digital Modernization

For dummies®
A Wiley Brand

Execute on a digital modernization roadmap

Scale an organization with modernization

Innovate through data and technology

Hitachi Vantara Special Edition

Premkumar Balasubramanian
Krishnaprasath Hari
Sarat Nagabhirava
Samta Bansal

About Hitachi Vantara

Hitachi Vantara, a wholly-owned subsidiary of Hitachi, Ltd., guides its customers from what's now to what's next by solving their digital challenges. Working alongside each customer, Hitachi Vantara applies its unmatched industrial and digital capabilities to its data and applications to benefit both business and society. More than 80 percent of the Fortune 100 trust Hitachi Vantara to help them develop new revenue streams, unlock competitive advantages, lower costs, enhance customer experiences, and deliver social and environmental value. Visit hitachivantara.com.

Digital Modernization

Hitachi Vantara Special Edition

by Premkumar Balasubramanian, Krishnaprasath Hari, Sarat Nagabhirava, and Samta Bansal

for
dummies®

A Wiley Brand

Digital Modernization For Dummies, Hitachi Vantara Special Edition

Publisher's Acknowledgments

Table of Contents

Introduction

igital modernization is at the very heart of some of the most forward-thinking organizations' strategies for growth and success. By focusing on foundational strategic improvements to an organization's data, infrastructure, and application programming interfaces (APIs), later improvements can move radically faster. Operational efficiencies are gained, which opens new markets for these organizations with new products and services that are technologically modern and primed for growth.

The most successful modernization efforts provide outcome-oriented, cross-organizational improvements through the interconnection of data with business applications, and the effective use of infrastructure. But more than that, digital modernization is meant to be *disruptive*: It creates outsized value that's much greater than the sum of individual upgrades to a cloud environment or application stack.

In *Digital Modernization For Dummies*, Hitachi Vantara Special Edition, you discover how to drive better business outcomes with a roadmap that encompasses your technology properties. By using the cloud effectively, migrating applications, refactoring and cross-hosting, managing edge to core to cloud data, and more are all treated. However, the consistent focus is on the sum of these changes transforming a business to not just exist in the modern technology ecosphere but to thrive within it.

About This Book

This book is about modernizing your organization's digital footprint and strategy. It can be your roadmap to the variety of components that fall into a comprehensive digital modernization strategy.

Consider this book as a jumping off point. Each chapter, and even each section, can generate questions and notes that lead you into potentially deeper exploration of the topic at hand. You get a clear understanding of how modernizing your business can be accomplished through embracing new technology trends.

Icons Used in This Book

Throughout the book, I occasionally use special icons to call attention to important information. Here's what to expect:

Keep this information always at the forefront of your mind. It informs important decisions and helps keep you focused on your goals.

These short bits of information can save you time, money, or headaches. They are also great for giving you other resource suggestions to explore.

Ignore these at your peril! These warnings call out common mistakes and provide you pithy solutions to help avoid them.

Sometimes you just have to get into the details! These brief definitions ensure you're current on technical details relevant to your goals without forcing you to read a 500-page programming or infrastructure book.

Beyond the Book

There's only so much I can cover in 48 short pages, so if you want information beyond what's offered in these pages, check out the following resources:

» "Your Data Fabric, Your Way": www.hitachivantara.com/ en-us/solutions/modernize-digital-core/data-modernization.html

» "The New 'Art' of Data Modernization": www.hitachivantara. com/en-us/pdf/point-of-view/art-of-data-modernization.pdf

» "Modernize Applications, Your Way": www.hitachivantara. com/en-us/web/point-of-view/modernize-applications-your-way-perspective.html

Chapter **1**

Embracing New Opportunities through Modernizing Your Digital Core

D igital modernization is the process of building, operating, and optimizing the applications and data in your organization to provide greater business agility and adaptability. The business world is increasingly competitive, and markets are changing and evolving at an unprecedented pace. You'll need to update, upgrade, and modernize your applications and data not only to keep up but also to ensure a continually modern approach to the evolving market.

To be agile, adaptable, and responsive for your customers, you must ensure your business can operate where your customers are: operating from anywhere at any time. Digital modernization gives you a proven process to accomplish this transformation in your own organization.

Adopting a Cloud Perspective

The first major milestone, and a common one across modernization initiatives, is shifting from a localized infrastructure perspective to a hybrid cloud perspective. To be clear, this doesn't necessarily mean you'll first relocate all your services and applications into a cloud hosting environment. Instead, it reflects different approaches to thinking about your entire effort based on cloud principles:

>> **The cloud is a service-first paradigm.** Instead of thinking about servers, think about services. What are you exposing to your customers, to their customers, and to your own internal systems? This will shift focus from virtual hardware to value-providing components.

>> **The cloud is about usable parts, not building wheels.** You shouldn't think about servers and instances because you really shouldn't think about building up those same component stacks that everyone else has built. Instead, the cloud gives you packaged, usable options that you can employ quickly to solve problems.

>> **The cloud favors higher-level components.** The cloud doesn't give you "hired help" to manage servers. Instead, it promotes thinking about business problems and trusting a virtual ecosystem — which you don't manage secondhand, by the way — to give you facilities to solve and expose insights and actions related to those business problems.

Think about your entire organization — your infrastructure, your data, and your applications — as integrated and interconnected. Modernizing your organization means taking a holistic and interconnected look at your entire technology estate, as shown in Figure 1-1.

Most organizations have a cohesive modernization plan but then silo the work. The data team works on its program, the application development team designs applications that will consume the data through application programming interfaces (APIs), and the DevSecOps teams build and deploy the workloads (data and apps) to support these other two teams. However, these ventures are far more interconnected.
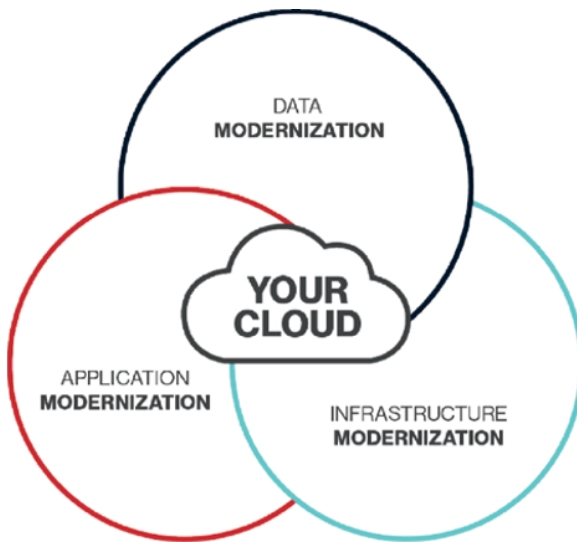
FIGURE 1-1: A holistic and interconnected approach to modernizing your organization.

First, you want to tie together milestones in each team. You don't want your data team releasing new models and algorithms if the infrastructure isn't in place to support those models. Your applications should also be releasing functionality based on both at the same time. Second, plan initiatives *across* your organization. If the data team doesn't design APIs in concert with the application team, and the infrastructure isn't supporting it, change the plan. This is a perfect example of digital modernization: Move out of individual team planning and into fuller planning that coordinates and even integrates teams into a cohesive whole.

Prioritizing an Always-On Mentality

As you begin to place value on various modernization initiatives, you should significantly prioritize anything that moves your business closer to being always-on. This always-on mentality sets your business up to respond to your market and stakeholders' needs, when those needs occur.

Availability is essential for modernization

Many conversations around cloud begin and end with scalability and cost-efficiency. However, an essential element of the cloud is its always-on nature. The definition of "On" changes based on your business needs: Websites are always available, data is always flowing, logs are always being generated, notifications are always flowing, and so on.

Business continuity, disaster recovery, data protection, and high availability all become options within most cloud environments, private and public. Your strategy should incorporate valuing each of these elements of modernization just as highly as you value reductions in cost or increases in scalability.

Migrating to cloud isn't a complete modernization strategy

Migrating to the cloud is part of modernization; however, *just* moving to the cloud isn't a complete solution. Your job goes beyond simple migration and involves evaluating your data and applications and then evolving them to function as your business needs them to.

Consider an application that uses data that's sensitive with respect to a certain geography and has strict compliance and regulatory needs. A good modernization strategy may determine that the application and its data are best hosted on private infrastructure to increase data control and security. Your strategy — whether you use cloud and in *how* you choose to use cloud — must be developed with the assumption that data is being accessed globally, 24 hours a day, 7 days a week. The data is "always on," and includes the governance, compliance, and regulatory requirements that apply to that data.

In this scenario, it's overly simplistic to say that moving the data and the application to a cloud — private or otherwise — is a complete modernization strategy. You also need to decide how and where to host the application, whether to refactor or rearchitect it to use cloud services that may enable better data compliance, and how to make safe updates to both the data and its accompanying application.

ALIGNING DIGITAL MODERNIZATION STRATEGY WITH BUSINESS STRATEGY

Unfortunately, terms like *digital modernization* are often used broadly and as buzzwords. This usage can dilute their meanings rather than sharpen them.

One easy way to dispense with a lot of fluff is to constantly bring your digital modernization strategy back to your overall business strategy. If they don't align, make changes. When thinking about your strategy at a high level, consider the following questions:

- **Are you always thinking digital?** It's not an accident that the word "digital" is a part of the term digital modernization strategy. If you're not thinking about digital delivery, presentation, and interactions, you're not driving the right type of modernization or creating new business value.

- **Are you unlocking new opportunities with data?** Are you exposing data that wasn't previously available? Do you have the availability of new data that has been difficult to expose because of infrastructure limitations? If your digital modernization strategy isn't revealing new data opportunities, you likely have more work to do.

- **Are you reimagining your business?** You should be thinking about your entire business in new ways. Are you unlocking new revenue streams? Does a set of lowered costs due to modernization open up new avenues of exploration or execution?

Ask yourself these questions throughout your modernization process and ensure you can continually answer "yes" to each.

Chapter **2**

Adopting a Digital Mindset

When building your own organization's digital modernization strategy, you need to focus and widen your lens. While moving to the cloud, prioritizing always-on initiatives, and better integrating your data and applications are critical, they aren't *all* that's required. Look for new opportunities that only exist through the lens of a digital mindset.

You need to think about your various assets individually rather than collectively. How do your own applications talk to each other, and to the data they consume and produce? Is your application architecture conducive to inter- and intra-communication?

The more you can look at your entire digital estate as a collection of individually connected, value-producing pieces, the more potential you'll see in these individual pieces to produce value. Your strategy then becomes about positioning those pieces in a way that they interact and also provide value inside and outside your own organization.

Providing New Services to Internal and External Consumers

Digital modernization is more than simply applying new principles and approaches to existing data, applications, and cloud strategies. It also creates new opportunities that require evolution throughout your organization.

REMEMBER A good digital modernization strategy should redefine and broaden your definition of technology in your organization. This goes far beyond just taking advantage of the cloud; it also includes a strong move toward service-driven architectures that expose applications and data in new ways to existing and new consumers. It's assessing new opportunities alongside existing ones and prioritizing them all based on overall business value that moves you firmly into the realm of modernization as a strategy.

Everything-as-a-service business opportunities

If you've been involved with cloud initiatives much at all, you've likely heard these common acronyms:

- » **SaaS, or software as a service:** Software running on a cloud provider that isn't downloaded or run locally is typically considered SaaS. Ecommerce platforms like Shopify (`www.shopify.com`) and sticky.io (`www.sticky.io`) are good examples.

- » **PaaS, or platform as a service:** A PaaS offers you a cloud-hosted ready-to-use platform that you can build applications on top of. Heroku (`www.heroku.com`) is a great example that makes application building easy without a lot of plumbing work on the platform itself.

- » **IaaS, or infrastructure as a service:** When you have virtualized servers, network routers, and hardware, you've got infrastructure as a service. Amazon Web Services (AWS, `aws.amazon.com`) and Microsoft Azure (`azure.microsoft.com/en-us`) are common examples.

Digital modernization adds to these with the idea of *XaaS*: X as a Service. The X in XaaS stands for "everything" and is a stand-in for nearly any part of your digital estate:

» **Data as a service:** Data itself is a massively important commodity and often needs only to be exposed instead of filtered through applications.

» **Storage as a service:** More focused than a complete cloud, storage as a service allows up- and down-scaling of storage on demand.

» **Data protection (or data security) as a service:** Like any -aaS model, data protection and security are needed at certain times and less so at others; sometimes to a greater degree and sometimes to a lesser.

The goal isn't to come up with as many words that fit into the X in XaaS as possible. Rather, the key is the word "everything" that the X stands for. If you view everything you host, refactor, and rebuild as a service, you can make different choices that better align with your modernization strategy and business values.

Providing services to your own organization first

It's easy to become overwhelmed by the prospect of adopting XaaS. That X-*everything* is pretty broad. However, begin with a much nearer-term opportunity: Provide services to yourself. For example, architect your billing engine to become a service (here, the X would be billing) to your own components, such as a finance application or a recurring billing front-end for your users. You could move from your applications directly connecting to your data stores to a model in which a data access layer over your data stores provide data to *any* consumer — beginning with your own applications.

By using your own applications as your first consumers, you'll better understand what those applications require. The work to create services from data and applications prepares you to expose those services to consumers outside of your organization.

Taking advantage of established patterns

Even when building services for yourself, it can be difficult to know where to start. Just as important, it's not always clear *how* to start. You can reduce this difficulty by remembering that while your business may be unique, the process of digital modernization is not.

Spend time engaging great partners and be clear about important criteria for what makes a partner great:

» **Knowledge:** Good partners monitor what's occurring across the industry and are at the leading edge of how to effectively modernize.

» **Experience:** Good partners go beyond knowledge and have done similar things before and can leverage that experience to speed up your own efforts.

» **Proven results:** Good partners have knowledge and experience that result in better and more consistent outcomes for their clients. Make sure of this before signing on.

The end result is less time and money spent on common tasks that have been done before. You'll be able to take those saved resources and focus on what is unique to your business.

REMEMBER

A cheaper proposal from a less-experienced partner may not in fact cost you less. Knowledge, experience, and results can save you thousands of hours and result in massive digital gains, all of which are ultimately more cost-effective in the medium- and long-term.

Modernizing People, Processes, and Technologies

It will be impossible to execute on a solid digital modernization strategy without assessing your entire organization. Specifically, you'll have to likely upgrade in some fashion your people, processes, and technologies:

>> **Upgrade your people.** The transition to a digital organization often demands new talent and skillsets. You have to assess whether the best path forward is to change staff out or simply train those you have. Regardless, don't make the mistake of assuming everyone will simply "figure out what they need to do."

>> **Review and refine your process.** Process can be a dirty word in growing technology organizations, but that can be simply the result of *bad* processes (or, just as often, poorly implemented processes). Successful implementations consider how teams interact with new cloud infrastructures, software that's exposed through application programming interfaces (APIs) in far more ways, and delivering new opportunities for interaction through those APIs.

TIP

This is certainly an area where spending time with Agile processes can pay off. While never a magic bullet, focusing on delivery of value (and not just code), clearly defining use cases, and extensive testing all within time boxed work (sprints in Agile parlance) all help your strategic implementation.

>> **Evaluate goals and adapt your technology.** It's far too simple to just "upgrade technology." Instead, your technology needs to be assessed against the goals of your modernization strategy. Also consider tools here: You may need different analytics tools, different services in your cloud estate, and even different quality assurance (QA) frameworks.

REMEMBER

Each of these focus areas takes a significant amount of time, both in planning and execution. As you're building out your plan, these need to be important milestones that sit alongside things like migrating to the cloud or testing your API. Don't leave these out.

Defining the modern in modernization

Modernization has another less discussed but important aspect when it comes to people: the people that use the results of the modernization. These may be current customers, partners, employees, suppliers, and others that interact with your business. These are all your digital consumers.

Digital modernization strategies will affect these consumers through more current and efficient user interfaces, personalized experiences, and a focus on operational excellence. These have internal effects, but ultimately are focused on the experience of that end-user.

Just as you should allocate time in your plan to internal process and people, you should block out time to study and understand your customers. What are their needs, and how will your modernization approach respond to, anticipate, and fulfill those needs?

Focusing on foundational needs over specific technologies

There's one final consideration as you upgrade and assess: Avoid spending disproportionate time on any one technology, even if it's extremely attractive. Artificial intelligence (AI) and machine learning (ML), for example, are great technologies that are absolutely trending. Cloud architecture and data warehouses were the same over the last decade.

You may have a need for AI and ML. However, ensure you allocate the majority of your time to strong fundamentals: a solid cloud architecture, effective process around delivering value through technology tools and development, and relationships with your digital consumers.

It will do you little good to embrace any one specific technology — even if it is popular and powerful — if you don't have solid data models, clear and useful APIs, documentation and planning around your architecture, and user interfaces that delight and satisfy your customers' needs. For all these reasons, keep the fundamentals first.

Turning Individual Initiatives into Action

All initiatives — embracing XaaS, consuming your own services, upgrading your skillsets, and prioritizing — will add value on their own. However, to truly embrace the opportunities available,

you need to execute *all* these initiatives and do that in a way that's firmly focused on delivering measurable value.

Creating a value realization roadmap

One essential aspect of any good strategy is that it's written down. As simple as that sounds, you'd be surprised at how many decisions and plans are made in meetings but never captured via documentation. When you prioritize value, the same is true: Write down your desired outcomes.

In addition to documenting the desired outcomes of your various initiatives and then prioritizing each, as part of a good value realization roadmap, capture these additional things:

>> **Where are your gaps?** Beyond modernization of your applications, are there holes in your overall cloud estate? Do these holes hinder a move toward everything-as-a-service? Identify these gaps and build in time and a plan to fill each of them.

>> **How mature are your processes and principles?** Your roadmap should call exactly what needs you have with regard specific training, hiring, and mentoring, and those should be aligned with your targeted service offerings and cloud evolution.

>> **What new application and architecture paradigms do you need to learn?** In addition to individual application modernization, there are best practices you may need to learn: microservices, containers, and serverless are common examples. Your roadmap should call out these in the same way it calls out people and process improvement.

You can't execute effectively without clear and careful evaluation. Your roadmap is your opportunity to do just that: Build a blueprint for what you want to do to achieve bold business outcomes. And because you've already begun to think of prioritization in terms of value, your roadmap will reflect that prioritization, emphasizing value production over what's technically interesting or cool.

Evolving your roadmap to ensure continual progress

REMEMBER

Your roadmap will change. In fact, your roadmap *should* and *must* change. As you execute, you will learn, and that will help you refine and sometimes reverse prior decisions. This change isn't a reflection of poor planning, but of *good* planning.

TIP

In fact, set up regular roadmap reviews, every 30, 60, or at most, 90 days. Evaluate progress and integrate learnings back into the plan. As you do more, you will know more, and as you know more, your planning will evolve and improve.

Chapter **3**

Migrating and Modernizing

A t this stage, you should have new models for thinking, new people and processes underway or in place, and a clear understanding of which portions of your organization and applications are ripe for value maximization. Now it's time to begin the actual work of migrating and modernizing.

Adopting the cloud and migrating applications into the cloud aren't as binary as they may seem (check out Chapter 1 for more information). You must consider multiple cloud models; not all organizations will adopt the same model across all applications and systems.

REMEMBER Uniformity is a great goal in a cloud and digital modernization strategy, but it shouldn't override value prioritization. Don't strive for a single cloud model for all your applications at the cost of driving maximum value creation.

Choosing the Right Approach to Digital Modernization

Key to digital modernization is an "always–on" mentality. This mentality also affects the approaches you take up for modernizing your applications. Each approach has different requirements and yields different degrees of benefit.

Understanding the approaches

Understanding the approaches available to modernization helps you decide where to start and potentially how to progress through different models over time. The approaches include the following:

» **Rehost (lift and shift):** This cloud migration strategy quickly and cost-effectively moves current applications into the cloud. The downside is that it doesn't leverage elasticity or other native advantages of the cloud, so you may not gain the operational savings you expect. Rehosting requires proper configuration to ensure that data center applications continue to communicate after they're placed in the cloud. Without proper configuration, updates may be lost, applications may not be optimized completely, and cloud migration and modernization issues may arise both in the short and long term.

» **Replatform (lift, tinker, and shift or containerize):** Replatforming is about moving applications to the cloud without major changes to take advantage of the cloud environment. It involves some up-versioning, such as adopting a managed database or leveraging a dynamic autoscaling functionality. The containerization aspect effectively captures an application and its environment in a "container," which makes it easy to move the application to another environment. Although this strategy takes longer than "lift and shift," replatforming provides a degree of cost-effectiveness, functionality, and time-savings, without the significant resource requirements of refactoring.

» **Refactor (cloud-friendly):** Leverage the cloud environment in this approach by modifying existing code and moving applications to better suit new infrastructure. Proper DevOps skills are critical to take advantage of this process. Although

refactoring is more complex and possibly more time-consuming than the other approaches, the positive value far outweighs the negative. Businesses typically see a higher return on investment (ROI) after they've completed the process of refactoring.

» **Rearchitect (cloud-native):** This approach builds from scratch. This cloud-native strategy may also incorporate continuous delivery and microservices. The result is resilient, agile apps that are portable across cloud environments. They exploit the continuous innovation model of the cloud, which provides improvements in functionality, operations, security, resilience, and responsiveness. Rearchitecting allows you to speed go-to-market efforts that build business success.

No "best approach" exists, although your business context will give you insights into what may work best for your immediate needs and how to potentially transition into more cloud-native models over time. Figure 3-1 shows how each approach is a different investment in time and a different realized value.
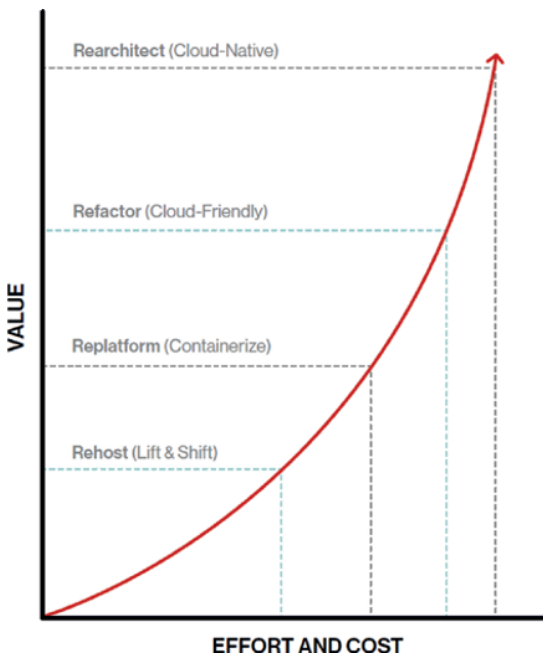
FIGURE 3-1: The cloud modernization continuum.

The effort and cost of modernization has a direct effect on the value realized in your modernization. For more information, visit www.hitachivantara.com/en-us/web/point-of-view/modernize-applications-your-way-perspective.html.

TIP View your modernization as a process rather than a single task. You may choose to embrace lift and shift for a time and later move to more cloud-native approaches. It's acceptable and even normal to progress through approaches over time.

Choosing the right approach

When evaluating the approaches to modernization and deciding which is best for your business, consider a four-step approach:

1. Envision.

Think about the specific outcomes you want to achieve. You also should keep your people and processes in mind to understand which approaches fit your current organization and goals.

2. Evaluate.

Identify the gaps between your organization and what you want to achieve. Build a roadmap to fill these gaps. Assess which models best fit this roadmap and your timelines.

3. Execute.

When you begin modernization, continue to measure and assess, and don't be afraid to make changes as you go. Your approach can change to meet the outcomes you determined in Step 1.

4. Evolve.

Continue to change, evaluating new approaches and the expected value from those approaches. As you modernize, this evolution should become faster and more cost-effective as well.

Migrating at Scale

Migrating and modernizing are two connected actions, but they aren't the same. Your ability to clearly distinguish between the two helps you choose the right approach for your current needs

and drive the most business value for your organization. As good definitions to distinguish between the two, you can use the following:

» **Migrating is moving your infrastructure and applications into the cloud.** When you take advantage of cloud instances — virtualized servers, shared databases, networking and routers — you're migrating. This is at the heart of a rehost or lift-and-shift approach.

» **Modernizing is updating your infrastructure and applications to take greater advantage of cloud best practices.** In the cloud, you have a huge array of services and decisions that allow you take great advantage of cloud services that reduce your maintenance and overhead. If you use a database service instead of hosting on your own virtual hardware or build an API without using a server instance, you're modernizing your infrastructure and applications. Replatforming, refactoring, and rearchitecting all involve modernization, in increasing degrees.

REMEMBER

These definitions are focused on migration and modernizing in a cloud context, but the general principles apply to any technical effort.

Migrating is part of all modernization approaches

Migration will be a part of nearly every modernization approach, regardless of whether you rehost, replatform, refactor, or rearchitect. Even in a complete rearchitecture, you will almost always have at least some data and applications that need to be migrated into the cloud as is. This means that regardless of approach, you need a flexible and reusable strategy for migration. The degree of migration may change — in a rehost, you're migrating all your infrastructure and applications, and in a rearchitecture, you may be only migrating smaller data stores and older applications — but the migration itself needs to be a careful and repeatable process that's well-tested and stable.

Automating migrations to make them repeatable

Given that you're going to likely have multiple migrations of pieces of your infrastructure to the cloud, and that you could have

10, 30, even 50 iterations of *each* of those migrations, you need to start thinking early about automating everything you can in your migration strategy. Automation simply means looking at each and every individual step and trying to remove the "hands on the keyboard." Write a script, use a tool, author a configuration file — do whatever you can to build a repeatable set of steps.

WARNING

For the record, a Microsoft Word document with a set of manual steps that someone must follow is *not* automation. In addition to making your migration repeatable, you're building in scale to your solution. You're ensuring that you can do it the tens, or in some cases, hundreds of times you'll need.

Making Value-Driven Modernization Decisions

There was a time when the first step to any cloud migration or digital modernization effort was to perform a rehost — straight to replatforming and rearchitecting. Consider that one global bank recently estimated it would take nearly 35 years to rewrite all its applications at the rate of current transformation. Conversely, a simple lift-and-shift approach didn't give this bank the return on its investment that it desired and didn't move the bank far enough toward the always-on mentality it desired.

This scenario is typical and reflects the choices that you and all businesses must make. You need to prioritize the applications that are worth refactoring, replatforming, and rearchitecting, while keeping many other, less value-producing legacy applications running in a rehosting configuration. It's critical, then, to value applications individually than as part of a single "one size fits all" modernization approach.

Establishing clear value criteria

You can quickly see that there is no "one way" to decide which modernization approach to take, even when you do consider the approach for applications one-by-one. You need to make clear decisions within your business about what you value, and how to rate those values.

To help determine which applications would benefit most from more cloud-native approaches like refactoring and rearchitecting, use the following key criteria:

» **Effort:** The further you progress toward rearchitecting, the more effort you will generally need to apply to modernization.

» **Initial cost:** As with effort, the cost of modernization rises as you move toward refactoring and rearchitecting and is usually less for rehosting.

» **Ongoing cost:** The ongoing cost of maintenance is often inversely related with the initial cost. Cloud-native approaches like refactoring and rearchitecting are typically more cost-efficient in the long run, helping to offset the initial costs involved.

» **Time and urgency:** If you have to get your applications into the cloud in a month, you have far fewer options than you do if you have a year. Consider how this urgency affects each initiative, and plan for both the short-term and the long-term.

» **People and skillsets.** People play a large role in your modernization strategy. You need to make decisions about approach that align with the people you have available to execute your plans.

Each business will value these factors differently and even change the value of each in different contexts. Your planning and evaluation should consider each of these and make decisions based on the values you assign.

Cloud-native tradeoffs

It's easy to tout the benefits of ditching server maintenance, taking advantage of refactoring, and generally embracing rearchitecting when possible. However, all of these advantages require tradeoffs that you need to consider.

» **Cloud native requires new skills and likely new people.** As has been said before, you can't take a generic IT administrator and give them a DevOps responsibility without major training and expect results. The more you dig into

cloud-native models, the more you're going to need to allocate significant time and resources to bring in new skills and people to support those models.

» **Cloud native requires different cost models.** You're going to have to evaluate your cloud estate differently than your local or hosted systems. Cloud-native operating costs are typically lower but getting to cloud native may require more upfront costs.

» **Cloud native requires different architecture.** Beyond just skills, the architecture of service-based, serverless systems is very different than an n-tiered server-based architecture. Beyond raw technical skills, you'll be able to solve different problems and provide different services, which will inevitably change your business.

None of this should dissuade you from modernization. However, if you plan for these tradeoffs, you can meet them and overcome them. And if you prioritize against them, you'll know when less modernization is actually the right answer for your organization's business needs and strategy.

Iterating Over Your Modernization Decisions

REMEMBER

At the heart of all your decisions is the envision–evaluate–execute–evolve process. And this process isn't static; it's iterative. This means that you may decide to rehost several legacy applications, replatform several more, and rearchitect your highest value–producing application in your initial modernization plans.

But after that phase completes, you go back to envisioning again, and evaluating again, so you can execute and evolve once more. This gives you additional opportunities to further modernize, to move more applications into a better "always on" posture, and to drive additional value into your business.

Chapter **4**

Managing and Operating Hybrid and Multicloud Environments

I n earlier chapters, there's been little distinction between a single cloud provider, multiple cloud providers, and hybrid cloud environments. That's intentional: Much of your planning and initial lift-and-shift migrations will look similar across these different providers.

However, as you move into refactoring and rearchitecting, and even more importantly, operations, there are specific considerations for handling multiple cloud providers and hybrid cloud installations.

These differences are substantial, and, in many cases, require as much adjustment and evolution as the initial shift from traditional hosting to cloud hosting thought. The way you choose to use different providers affects your deployments, automation, staffing, and even how you test and approach finding and correcting bugs and errors in your applications.

REMEMBER While this chapter and most of this book treats cloud providers uniformly, that's at best a helpful abstraction in planning. In reality, each cloud provider has significant differences, and you should account for learning those differences in your detailed planning and architecture.

Managing Your Cloud Seamlessly

Adding multiple cloud providers will increase the overhead and complexity of managing your cloud estate. The same is true of implementing a hybrid cloud strategy. So why would savvy organizations not focus on one cloud provider to optimize these processes?

You can look at using more than one cloud provider for a number of reasons:

» **No magic bullet cloud provider:** Each cloud provider is different, and there's no "better" that fits all situations. There may be a "better" for a specific set of use cases, but those use cases are often *very* specific. Larger organizations with differentiated interests will typically find that multiple use cases demand multiple cloud providers, each with particular strengths.

» **Redundancy at the provider level:** Moving to the cloud provides redundancy almost impossible to arrive at with a local infrastructure. This redundancy is attractive as it reduces risk dramatically. Adding additional cloud providers can then further de-risk a modernization plan by providing failover at the provider level — of the cloud itself, in addition to failover within the cloud.

» **Uniformity of operations applied to application diversity:** That's a mouthful, but an important one. Different cloud providers can be managed in uniform ways at a high level, while still providing different applications widely varied services.

» **Security and data compliance advantages:** Some organizations will look to hybrid cloud in particular to manage data and compliance concerns. These organizations place outsized value on "owning" segments of their data locally while still valuing cloud services for the applications that consume and expose that data.

>> **Vendor affinities for specialized use cases:** There are cases where a certain cloud vendor just really does a particular task well. Microsoft's Azure is better than competitors at running and managing Citrix-based workspaces, and Google's cloud is exception at analytical and machine learning workloads.

TIP

As a slight over-simplification, consider multi-cloud a tradeoff, and these points to be items in the "pro" column. The additional overhead of managing multiple environments is generally only worth it if you need at least two of these major advantages. Less than that and the overhead will likely outweigh the upside.

Deploying changes instead of applications

This is less a cloud-specific technique as much as a practice that has developed alongside an emphasis on cloud. The older model of traditional whole application deployment is no longer ideal. It brings with it greater risk and in many cases, extra time to run tests and deploy.

A more modern approach is to deploy the portions of an application that have changed. This will require you to have thorough testing and a test or sandbox environment that mirrors production, but allows you to deploy much smaller units of code. This can be accomplished through more loosely connected code — often using a microservices model — or by deploying various portions of code in containers using Kubernetes.

Whether through microservices or containers, the less code you have to deploy to update an application, the better. Your updates and the accompanying testing are focused on delivering specific value while still avoiding the risk of whole-application deployment.

Automating cloud management

Automation is a key part of a modernization strategy. But, much like lift-and-shift services, many common automation tasks are nearly identical across organizations. Automation, then, is an area where selecting a partner with proven solutions can yield major cost and effort reductions.

You should look to automate in a number of essential areas:

- **>> Monitoring:** Logs should regularly be ingested and filtered, and alerts raised as needed.
- **>> Deploying:** Deployment should not only be simple to execute but also eventually an automatic part of the process of committing reviewed code.
- **>> Scaling:** Performance and usage triggers should cause your applications to scale up or down as needed to both meet demand and not overuse resources.
- **>> Security:** Security is more than a static, one-time task in the cloud. New data should be secured and, in some cases, encrypted, just as new instances and services are automatically permissioned.

    In multi- and hybrid- cloud environments, automation is even more important. The overhead of multiple providers *must* be made manageable through common scripts and process, as well as consolidated dashboards.

AUTOMATION FREES TIME AND CREATES OPPORTUNITY FOR MORE SKILLED WORK

There's a common fear that if automation is used heavily, it will cost people their jobs. This is an ungrounded fear. Automation removes the need for skilled labor to monitor logs and "watch lights," leaving that work to repeated, unmanned processes.

The reclaimed time for employees previously performing these tasks should be used for actual skilled work. Cloud networks can be optimized, scalability can be built out, and databases can be tuned. In other words, your high-valued operations engineers can actually create greater value by spending their time on more skilled work.

Building an Always-On Business

You've certainly picked up by now that there is a significant shift when moving from traditional IT to a cloud-based architecture. This goes beyond skillset, though, and actually gets at the fundamental tasks involved for your operations teams. An "always-on" mentality is critical, both in a maintenance and scalability context.

Moving from routine maintenance to scalability

A significant amount of the "keep running" task involves the underlying hardware: patching, compliance, security issues, and with each change, ensuring the things that worked prior to the change work after the change.

In the cloud, minimal effort is involved in keeping systems running. The majority of patching and security issues are resolved as a matter of course and backward compatibility is rigorously tested by the cloud provider.

The recouped time, then, goes to building a resilient and reliable system that can scale up and down, ideally based on triggers and automation. Your IT staff should transform into a group of cloud engineers and site reliability engineers (SREs) who are looking to tune your system, not just keep it running.

Broadening operations

In addition to pure scaling, your operations team should become a much broader organization. Figure 4-1 shows a wide array of activities. While many operations and SRE teams won't do all these, you should look to move into as many of these areas as make sense for your organization.

This is more than just a simple adjustment in the skills for which you look. Instead, it's a shift in thinking: A site should be reliable, and while that requires Site Reliability Engineers (SREs), it also requires a mindset focused on stability and operational readiness.
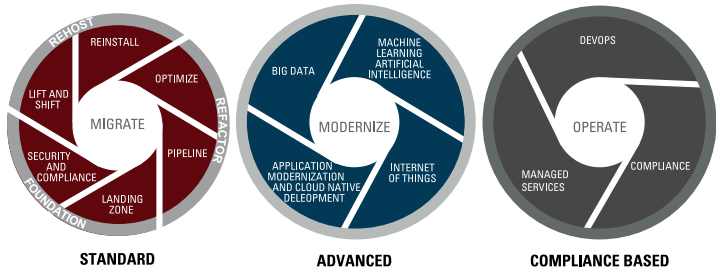
FIGURE 4-1: Broaden your organization with a wide array of activities.

Planning for Volatility

There's no organization that will succeed if it cannot handle change. The more you assume change is a constant and plan for it, the more responsive your organization will become. Responsiveness directly creates value by increasing customer satisfaction and decreasing defects and waste. If you architect your applications smartly, you can support change more readily without significant ongoing cost increases.

TIP

There's a fairly significant element of cloud architecture that actually encourages volatility. Your applications will need to run on different servers without hard-coded network addresses, for example. The sooner you embrace this sort of volatility, the better your architecture will serve you.

Reacting to customer needs

There's nothing about digital modernization that will reduce the needs of your customers. However, there is a *lot* related to digital modernization that makes addressing customer needs more immediately.

First, begin with the premise that most customers interact with your organization in two ways: through the standard operation of your applications (purchasing an item or a service, for example), and when that standard operation doesn't meet their needs. The second category could be any of a number of things:

>> **There's a defect in your application(s).** In this case, something legitimately isn't working and needs to be fixed.

>> **There's a feature gap between what you provide and what your customer wants.** In this instance, what you have works, but the customer wants some functionality or workflow that you *don't* have.

>> **There's a misunderstanding between your customer and your applications.** In this instance, you may have poor documentation, a confusing workflow, or poor user experience. Things work, and the customer *should* be able to do what they want, but are confused by how to do that task.

In all three of these instances, you should see decreased overall closure times on customer issues when you modernize. You should measure ticket creation to closure time, use something like Net Promoter Score (NPS, `www.netpromoter.com/know`), and usage metrics to determine if you're being successful in meeting customer needs with your modernized applications and platforms.

REMEMBER

The goal of a good customer interaction isn't met when a support rep hangs up a phone or closes a digital chat. The goal is to actually resolve the issue — and this is where automated deployment, only deploying changes, and making changes across multiple resilient environments all comes into play.

Prototyping without fear

If you've ever been a part of a new application or product launch — whether that's a true MVP, a beta, or a full-blown production launch — you know the risk involved with deployment. No matter how much testing, there always seems to be something that comes up when you push to that production instance for the first time.

In the cloud, there is still risk, but it's mitigated because you can simply tear down and throw away a broken deploy. You can even throw out the virtual instance itself and start from automation scripts to rebuild the whole system.

REMEMBER

Everything in the cloud is ephemeral, and you should build scripts to rebuild *anything* you use — from a bare virtual instance to a complex API gateway with multiple web application firewalls. If you need it at all, then automate creation (and recreation).

Reducing cost by reducing waste

One often overlooked but important benefit of cloud architecture is the recycling and reuse of virtual hardware. In traditional IT, if you need to build a prototype, deploy a new version, or duplicate a database, you need additional hardware. Then you push your changes and updates, and (hopefully) retire the old hardware, reclaiming resources and cost.

Of course, the reality is that older resources are often forgotten. They are either left running and ignored (perhaps without a single network route to the old applications), shut down but left to collect dust, or in some cases, resold at a fraction of the original cost. In all of these situations, there is cost associated with the *waste* of the older resources.

In the cloud, it's easy to either manually mark resources as unused and available to be reclaimed, or to set policies that automatically reclaim old and unused resources. Further, all major cloud providers are constantly upgrading older hardware (virtual and the underlying provider-owned systems), without passing those costs on in most cases.

So this becomes yet another valuable avenue of change. If your applications can run seamlessly on various hardware and software stacks, they can easily be moved to newer resources. If you're deploying changes often, you'll naturally build out a rhythm of retiring unused application stacks as well. The result is a savings of cost but an entire application estate more resilient to change.

Chapter **5**
Innovating to Grow New Business Streams

I t's actually quite easy to get deep into the execution of a digital modernization strategy and lose sight of why you started the process in the first place. There's value in moving to the cloud and refactoring and rearchitecting your applications, and there's cost savings in automation and moving away from traditional IT. But these aren't the core drivers of a good digital modernization approach.

The goal is to grow your business — and ideally, to add new streams of revenue. You're not looking to just improve your technology stack or architecture but to make available new ways to use your data and engage your customers. But, like every aspect of digital modernization, you need to be intentional about making this happen.

TIP

The digital modernization roadmap and value realization roadmap (I cover these in Chapter 1) — are great ways to maintain this intention. Schedule specific meetings and milestones related to creating and executing on new business streams in those roadmaps to ensure that it happens.

Enabling a Product-as-a-Service Model

Platform as a Service is typically abbreviated PaaS, but that's the same abbreviation now cropping up for Product as a Service. You'll have to pay attention to context when you see PaaS to ensure you know which is being discussed. As a rule of thumb, though, PaaS still typically means Platform as a Service.

Product as a Service follows from another popular model, Service as a Service (SaaS). (Yes, the "as a service" moniker may be getting over used, but the gist is still important.) In both of these models, value is provided on top of, or in addition to, another product.

For example, consider selling a digital banking account to a customer. That's an initial product sale that has ongoing revenue attached through recurring fees. In the Service as a Service model, you may provide an additional set of fraud protections and a package for intentional wire transfers for an additional fee. This additional set of services are "pluggable" and can be integrated right into the original product, the banking account.

Product as a service goes even further: It offers entire product lines. Here, you may offer another digital product, like a stock trading dashboard for casual traders. But it integrates into your original product deeply, and the sum of the two products is actually greater because of those integrations.

This is the idea of Product as a Service: Because you're in the cloud with interconnected data and applications, your products can be provided as services — add-on integrations that are easy to purchase and create new or increased revenue streams.

Monetizing Data and Technology

After you have a clear intention to examine new business streams, you have to take practical steps. The first is to look at your data in particular and see how it can be used in new ways. Then, you can take those new approaches and turn them into viable products or services (as-a-service, of course!).

Treating data as a commodity

Your data is critical to your organization. By having it in the cloud (or stored onsite but available through a hybrid cloud), that data can be used by all your applications and integrated with your other data. Good modernization architectures allow any data to be accessible by any relevant application.

In that sense, then, your data doesn't just drive applications. It becomes the raw material on which applications, and more importantly, *all* applications can build. Purchasing patterns from one application can inform another, and usage logs from your most popular applications can inform scaling patterns for less used applications.

Further, look at cases where one application could use data from another application . . . and do it! Unbound by being on the right network or having easy routes to a different database, all your data should be used as needed, in many more than just a singular application.

Turning data into customer insights

In addition to simply exposing new data to new applications, you can add data analysis capabilities and services to your architecture. In this case, you're actually doing data analysis that may not have been available due to cost or architecture in an off-cloud architecture.

TIP

Machine learning (ML) and artificial intelligence (AI) come into play here. However, don't be put off if you don't (yet) have ML and AI experience on your team. Even basic data analysis — filtering, sorting, and calculation — can yield benefits from a large data estate.

Look for the following insights:

» **Trends exposed by combined data:** Look across your data supersets (combined data sets) to see if trends emerge because you have larger sets to examine.

» **Purchasing patterns:** This example is a specific trend but an important one. Can you learn more about how your customers engage with your products at purchase time? Do they tend to buy item A and add items D and E? Maybe you could group those items to increase total sale value.

>> **Purchasing funnels:** Do customers tend to make more purchases from one landing page than another? Do they buy from offsite links or spend time browsing product pages? Is your Contact Us form or your chatbot widget more effective?

Don't get too worried about whether something is important enough to be an "insight," either. Just look for things that seem actionable and try them out.

REMEMBER

Chapter 4 covers automation and the ability to deploy changes and applications quickly. This is where that pays off. If you can experiment with data findings at low cost, you'll almost certainly find some of those findings yield real value.

Reacting faster to your customer

Chapter 4 discusses the planning for volatility that's an essential part of and benefit from digital modernization. In this context, a key part of that increased reaction time is the data you have available from individual customers as well as your larger customer base.

A great way to begin applying what you know about your customers is through recommendation data. Designing cross-sell and upsell opportunities based on a customer's purchasing pattern takes the data you have and puts it to work. This also feels "fast" to a customer — decisions they make affect future decisions, even when the "future" is only a few minutes later.

You can expand opportunities by responding to customers based on patterns observed from similar customer cohorts. You may glean insights that show midwestern customers prefer certain styles or price thresholds, or that customers with accounts over a year need to see less marketed items because they're more familiar with the core catalog. Applying these to an individual customer creates more personalized experiences that generally return more sales.

Designing digital experiences

All this work creates what are now often called *digital experiences.* This is the idea that a customer is engaged with and interacting with your application or website, rather than just browsing a digital catalog.

Every time your site or application offers a different option, or actively messages your customer (not just through SMS but even in the application itself), or makes a suggestion at the point of sale, you're moving from a static site to a digital experience. You want your customer to feel like they're interacting with your site, not just searching and clicking around.

TIP

Ask your digital modernization partner if they have user experience (UX) services that they can offer. Experienced UX designers can help you quickly convert traditional applications into more engaging ones and will spend less time getting your own digital experiences up and running.

Discovering New Business Models

Ultimately, everything you've been doing has opened up new ways to do business. You should now have several key facilities that you likely didn't before:

» **Automation to support rapid growth:** New products can be added and made available within days, and eventually hours and even minutes, when data analysis exposes new opportunities.

» **Digital delivery pipelines:** The work you've done to automate your applications and deployment of code can also often be applied to delivering digital products to your customers.

» **Always-on services:** Your services — either used by customers for purchasing or as products themselves — are available 24/7 at a fraction of a cost of traditional IT.

» **Hosted products and services:** If you're only delivering physical products, you now have opportunities to make digital offerings and add-ons available. Your infrastructure can support it and you should have data to understand what your customers want.

TIP

This is another area where your partnership with an experienced digital modernization service provider is critical. They can often provide meaningful insight into what similar businesses may have accomplished through their own modernization efforts.

At this stage, with a stable application estate and iterative cloud deployments, you should be able to take a metaphorical step back and be creative. What data has been unavailable, but now can be accessed by customers or your applications? What insights might you make that inform new applications or new customer segments?

This sort of thinking will ultimately multiply the value of your investment in modernization. By looking at new ways to engage with existing customers and new customers that are ready for engagement, your entire business can be transformed by modernization.

Chapter **6**

Ten Digital Modernization Tips for Success

You may face many challenges on your digital modernization journey, but this chapter helps you with tips for success in your strategy.

Get and Keep the Right People

There's no single factor that will make or break your modernization more than the right (or, in some cases, the wrong) people. But getting great people with modern technology skillsets isn't easy.

First, you need to upscale your hiring process. You want to expose quality candidates to senior leadership, likely give them exercises to complete and discuss, and insist on detailed explanations of difficult problems solved in the past. You also almost certainly have to spend more than you expect to secure great architects and leaders.

Second, you need to have a clear means of expressing your organization's problems — and cast them in light of a challenge. Good people want to join and stay with organizations that are solving hard problems. In fact, a well-expressed hard problem is *just as important* as the compensation packages you offer.

Understand Your Digital Portfolio

In Chapter 1, I tell you that part of a good strategy is prioritization. Don't look for a one-size-fits-all approach to your applications and data. However, to segment your modernization approach by priority, you're going to need to assess and catalog every bit of your digital estate. Small, untracked support scripts or one developer's personal sandbox or an ephemeral test environment used by your deployment pipeline all need to be identified.

Also consider this catalog a living document. After you begin to modernize, keep it current. At any given point in time, you should have a current view of all your digital properties, the state that they're in, and links to any future modernization plans.

Secure Your Data

There are huge 800-page books written on security, so securing your systems — and especially your data — is a major concern. Build into your plan time to examine security options for your data, both in structured form (databases) and unstructured form (file and object storage).

In particular, devote time to securing your data *at rest,* meaning when it's not being used, and when it's *in transit,* meaning when the data flows to and from application programming interfaces (APIs), front ends, and across networks. You need a strategy for both to ensure data security.

Train Data Models

Like security, artificial intelligence (AI) and machine learning (ML) are vast subjects unto themselves. Still, a couple of key points are important. First, as you modernize and move into the

cloud, you're going to have access to cloud-based AI/ML services and massive processing power. Look to take advantage of these facilities early and often.

When you have your data in the cloud — even while still migrating applications — look to form datasets that you can train ML models on. These models can then be applied to your broader data estate and form actionable insights that may inform the later stages of your modernization plan.

Another subtle, but important, benefit of training data models early in your modernization: You'll create a data-driven culture. The more you focus on data and insights from your data, the more your decision making is grounded in empirical evidence. Yes, data can be deceiving (check out the next section, "Avoid Data Fallacies"), but your ML models are generally better decision makers than conjecture or unproven theses.

Avoid Data Fallacies

Although good data modeling is important, be careful to avoid common fallacies. Here are a few principles to follow to avoid data errors:

» **Start with a hypothesis; don't apply one at the end.** This is as much scientific method as data principle. Don't look at patterns that emerge and then form theses that fit those patterns. Instead, build your hypotheses, create tests to validate it, and see what the data says. If you're wrong, then start from the beginning again. Use what you've learned, but don't simply look at the data outcomes and form new theories.

» **Distinguish correlation from causality.** Just because two things happen together often doesn't mean that one causes the other (or vice versa). Use different datasets and also examine what additional factors may be causing what you're seeing. This is also an area where you shouldn't abandon common sense; you should recognize that a decrease in horse-drawn carriages is likely not the cause of a decrease in communicable diseases over the same time span (as a poor but obvious example of a non-causal relationship).

>> **Don't overfit outliers.** You'll often find patterns that seem to explain 85 percent or even 90 percent of your data, leaving just a few outliers — data points that don't fit a pattern. Don't stretch your pattern to make everything fit. Instead, examine outliers. They could be genuine exceptional cases that do *not* disprove the pattern you've found. In other cases, they might help you find an entirely different pattern. The only way to not get value from these outliers is to completely ignore them.

Set Realistic Cost Expectations

Modernization has significant costs associated with the process. Those costs should be measured against the return on investment that a more modern, capable, scalable system can provide. And of course, every business wants to minimize costs and maximize returns.

That said, there are three key points of awareness you need to realistically estimate:

>> **Phased migration is more expensive but much less risky.** You often begin with a lift-and-shift approach and then later refactor and re-architect. This means more expense than a one-time massive modernization, and you should plan accordingly. The realization of the extra cost is *far less risk*, though, and almost always worth that extra cost. Check out Chapter 1 and 3 for more info.

>> **Cloud cost savings are realized over longer time periods than often expected.** You're not going to get massive savings back from migration until you've gone beyond lift-and-shift, and even then, it may take 18 to 36 months to see major decreases in costs. Be aware that you'll often be told this horizon is much shorter by vendors, so be wary of overly aggressive estimates here.

>> **An experienced migration partner will pay back their expense multiple times over.** You should budget significant expenses for partnering. These will often be the largest part of your budget, and also the first you'll be tempted to cut back on. *Resist this urge!* You can literally save millions of

> dollars with a good partner helping you avoid mistakes. There are thousands of stories of failed migration and modernization efforts that *then* pull in a partner and spend even more to undo a poorly planned strategy.

Monitor Everything

One major benefit of the cloud is that regardless of provider, monitoring is woven into every aspect of the infrastructure. In fact, in AWS, GCP, and Azure, you often have to *turn off* monitoring. This should be something you spend significant time on.

So first, make sure you and your team learn where monitoring and alerts are deposited. It's often slightly different for different services like a database, or an API gateway, or a storage bucket for unordered data. Second, spend time building consolidated dashboards that relay critical information to you quickly. In the event of a problem, you don't want to have to look in 20 different places to track down what's gone wrong.

Finally, ensure that alerting goes beyond your cloud provider and to your operations and support team proactively. There's nothing worse than an alert triggering but nobody receiving it.

Keep Architecture Current

There's a certain finality to the idea of modernization: You modernize, and then you're finished! Of course, that's not accurate. Modern in the year 2022 will be legacy in 2024, especially with the blazingly fast advances being made in cloud services, data, and machine learning. You should undertake modernization with a commitment to continued improvement.

This begins with building a modular, or loosely coupled, architecture. You can also employ paradigms like microservices to allow you to upgrade, enhance, or transform portions of your architecture without having to update everything at once.

More importantly, though, just keep reapplying your modernization approach. Segment your digital portfolio, assess where older components are costing you value, and then modernize. This

Avoid Chasing Shiny Objects

A shiny object represents a technology advancement that doesn't provide *business value*. Using new technology is a good use of resources, as long as there's something that adds to your business.

A simple example is moving from a REST API to a GraphQL API. GraphQL is popular, very trendy, and offers a lot of flexibility. The deciding factor, though, shouldn't be whether GraphQL is "good" or "bad." It should be: "what does it offer the business?" If you have partners that use applications that can send to and consume GraphQL, this could be a worthy investment. But if you have 100 partners and 95 of them (when you ask them, because you should ask them) prefer REST, then GraphQL, for *your organization,* is a shiny object.

Look for the Next Big Thing

You should be actively looking for areas where technology can add additional value. In fact, if you're getting and keeping the right people, they'll do this for you. Good architects and engineering leaders are always checking out the latest developments in React, object-oriented databases, or container technology.

TECHNICAL STUFF React is a JavaScript-based framework for building dynamic front-ends and communicating with APIs and other back-end systems. Keeping up with new technologies like React will keep your options open as your evolve your technology stack.

You should establish channels for these curiosities to become organizational learnings. Regular lunch-and-learns where new ideas and technologies are presented will push information out. Then, leaders should be carefully listening. Perhaps a new transfer protocol could reduce latency because of how your React-based front-ends structure API requests. In those cases, an article that one architect read on medium.com could end up being part of a new technology initiative that significantly improves your business and modernization efforts.

Modernize your organization's digital core

This book contains everything you need to begin modernizing your own organization's digital footprint and strategy. You learn how to build a clear, holistic, cross-organizational plan that addresses your data and technology as critical parts of advancing your organization. You also gain focus around driving business outcomes through migrating and modernizing your applications as well as your data. You build a plan for disruption to achieve scaling and growth without losing sight of clear goals.

Inside…

- Building value in cloud modernization
- Adapting to a cloud perspective
- Modernizing to optimize
- Managing multi-cloud environments
- Planning for volatility
- Monetizing data and technology
- Avoiding common modernization pitfalls

HITACHI
Inspire the Next

Go to Dummies.com™
for videos, step-by-step photos,
how-to articles, or to shop!

for dummies®
A Wiley Brand

WILEY END USER LICENSE AGREEMENT

Go to www.wiley.com/go/eula to access Wiley's ebook EULA.