Hybrid Cloud Strategy

for dummies

A Wiley Brand

- Consider key cloud strategies
- Design for maximum flexibility
- Plan with an iterative approach

Ed Tittel

2nd Red Hat® and Intel® Special Edition

About Red Hat

Red Hat is the world's leading provider of enterprise open source software solutions, using a community-powered approach to deliver reliable and high-performing Linux, hybrid cloud, container, and Kubernetes technologies. Red Hat helps customers integrate new and existing IT applications; develop cloud-native applications; standardize on its industry-leading operating system; and automate, secure, and manage complex environments. Award-winning support, training, and consulting services make Red Hat a trusted adviser to the Fortune 500. As a strategic partner to cloud providers, system integrators, application vendors, customers, and open source communities, Red Hat can help organizations prepare for the digital future.

About Intel

You may know Intel for its processors. But Intel does much more. Intel invents at the boundaries of technology to make amazing experiences possible for every person in business and society. By harnessing the capability of the cloud, the ubiquity of the Internet of Things, the latest advances in memory and programmable solutions, and the promise of always-on 5G connectivity, Intel is disrupting industries and solving global challenges. Leading on policy, diversity, inclusion, education, and sustainability, Intel creates value for its stockholders, customers, and society. Contact Intel at 2200 Mission College Blvd., Santa Clara, CA 95054-1549 or call (408) 765-8080.

Hybrid Cloud Strategy

2nd Red Hat® and Intel® Special Edition

by Ed Tittel

for
dummies®
A Wiley Brand

Hybrid Cloud Strategy For Dummies®, 2nd Red Hat® and Intel® Special Edition

Publisher's Acknowledgments

Introduction

Cloud computing is now a mainstay within the world of IT. Likewise, cloud computing continues to grow ever more complex and multi-faceted. Organizations often build their own private cloud infrastructures in-house, sign up for services from public cloud providers, such as Amazon Web Services (AWS), Google Cloud Platform, IBM Cloud, or Microsoft Azure, and create hybrid environments. Other vital choices follow close behind, including Software-as-a-Service (SaaS) in many shapes and forms. Architects and engineers must deal with an array of connections, integrations, portability issues among clouds, resource options, orchestration, storage, and more. And it must all be managed and maintained, made to work for improved profitability and productivity.

You can probably see why a carefully thought-out and detailed approach to cloud computing — a strategy, in other words — is so important.

About This Book

Jumping onto the cloud is easy. Getting it right is somewhat trickier. Getting it right for the long term is a big challenge. A good cloud strategy helps you and your organization work things out, makes sure all the bits and pieces fit together well, and improves the odds of realizing your business goals.

This book's topics are laid out in a logical order. But you don't have to read chapters in order unless you want to. Even so, I think Chapter 1 is a great place to start. That said, if a topic catches your fancy, jump into (and around) this book however you like. Each chapter stands on its own, so you can chart your own course. Read it in any order you like (but it's probably not helpful to read it backwards). I hope that, if you read this book in its entirety, you'll agree that you've been handed good ingredients and a recipe to put such a strategy together.

Icons Used in This Book

I occasionally use special icons to focus attention on important items. Here's what you find:

REMEMBER This icon with the proverbial string around the finger reminds you about information that's worth recalling.

TIP Expect to find something useful or helpful by way of suggestions, advice, or observations here.

WARNING Warning icons are meant to get your attention to steer you clear of potholes, money pits, and other hazards. Soft clouds can deliver hard knocks!

TECHNICAL STUFF This icon may be taken in one of two ways: Techies will zero in on the juicy and significant details that follow; others will happily skip ahead to the next paragraph.

Beyond the Book

This book can help you discover more about the cloud and strategies for its best deployment and use, but if you want resources beyond what's offered in this book, additional reading that's chock-full of useful info can be found at the following links:

» `www.redhat.com/en/resources/managing-kubernetes-clusters-dummies-ebook`: *Managing Your Kubernetes Clusters For Dummies,* Red Hat Special Edition, explains cluster-management challenges and best practices for managing Kubernetes environments.

» `www.redhat.com/en/explore/cloud-strategy`: The Red Hat website offers all kinds of useful cloud-focused information, including a collection of resources aimed at cloud strategy.

Chapter **1**

Key Cloud Strategy Considerations

Streamlining and strengthening an IT ecosystem's foundation is essential to realizing business objectives. There's no better way to do that than formulating an effective cloud strategy. Creating a cloud strategy involves pondering some key considerations, which you discover in this chapter.

> **TIP** When building a strategy, complying with applicable policies or regulations around your data and processes is important during every step. Check each one!

Tying Your Project to Success

By tying your cloud project to clear desirable outcomes and benefits, you define your own yardstick for success. All your goals should have clear and specific key performance indicators (KPIs) and success criteria. That means making important decisions as you set those objectives. You must decide what represents success and how you'll measure its attainment. Is your desired outcome "to accelerate software delivery through adoption of containers" or is it "to automate provisioning of servers and applications over the entire life cycle"? It makes a big difference. Or, is the true goal to replace a legacy infrastructure and to modernize IT delivery?

Whatever your goals, you need KPIs and success criteria to match (and measure) them.

Overcome hurdles

For cloud computing success, projects must address organizational challenges in implementing new systems. Failure to anticipate and handle these challenges poses dangers to project success. While benefits from adopting a new cloud abound, change disturbs the status quo. That's why securing management sponsorship and buy-in is key, starting at the top of the org chart. Get all important stakeholders in your camp, and you'll have less trouble from other interests.

Identify major challenges

Start with easy apps and identify your implementation challenges. Learn about and consider various options to solve them. Can a public cloud handle your needs and concerns, or must you use a private cloud? Or do you need a hybrid cloud (mix of both)? Whatever you decide, be sure it matches your organization's goals for automation, management, and scaling.

Think about things from business users' perspectives. Ask yourself, "How does this cloud project serve them?" and "By opting to host my apps on public cloud providers, what are the risks of locking the company into to them?" Also ask, "Does it make more sense to build cloud-ready core applications so they can run on or migrate into any cloud infrastructure, to avoid vendor lock-in?" Emphasizing benefits helps gain buy-in from decision makers and makes a project an easier "sell."

REMEMBER

When choosing cloud technologies, it's not always either private or public. You can create a hybrid cloud approach and combine private and public clouds together.

Plan workloads

Think about the time and work it will take to bring your idea to life. That means learning about the runtime environment to better understand your IT landscape and its current limits. Find out which of your applications can — possibly through some refactoring — and which ones can't be moved to the cloud. Through analysis, and perhaps even a pilot project, determine how best to support current and planned workloads.

TIP

Migrate easy (that is, application web tier) applications and then work into the harder stuff (like databases or message bus hosting). Keep compliance in mind, as well.

Brace for impact

Your new project impacts your operations team's workflows, automation, and management policies. Bring them into your process early and often. If you focus on operational knowledge and skills, you help align your organization with new or re-engineered processes. The more Ops is prepared, the smoother your transition becomes. This makes inclusive awareness and documentation vital because it captures changes during the transformation process and keeps folks informed.

Assembling Strategy Ingredients

An open, hybrid, business-oriented cloud lets organizations focus on digital transformation, and it shortens time-to-market windows. To do this right, identify common private cloud use cases and then create architectures that fit their needs and requirements. For example, an IT organization may need a cloud strategy focused on security and compliance requirements, as well as regulatory or financial considerations. It may also want to transition to a hybrid environment in which applications run across multiple clouds. The use case would be the point from which it picks appropriate architectures. In the same vein, a telco organization needs to understand its service offerings and integration and interoperability requirements to help client organizations make the most of what they offer.

REMEMBER

Get all key questions answered before starting a cloud architecture. Don't zoom ahead with a lot of issues open.

Putting Cloud in Perspective

To put the cloud in perspective for strategy purposes, check out these two definitions:

» *Hybrid cloud* describes a mixture of public and private cloud resources. Such a mix offers potential for portability among cloud elements but requires oodles of connections and

integration points. Hybrid clouds normally employ elements such as containers, container orchestration, common operating systems, runtime environments, flexible storage, and universal developer frameworks and tools. A hybrid cloud may include dynamic resource allocation and migration among clouds (called *cloudbursting*). Hybrid clouds designed for portability make it easier to orchestrate cloud workloads via unified management.

» *Multicloud* is a combination of multiple cloud resources. Thus, it may even be applied to Software-as-a-Service (SaaS) or to cloud-specific tools, such as those used for data analytics, machine learning, or database functions.

Why Cloud Native?

Whether the application level or the infrastructure level, cloud native means that applications and services are completely at home in the cloud. Organizations adopt cloud computing to increase scalability and availability of apps. A cloud-native infrastructure makes this feasible and provides self service and on-demand provisioning for cloud-based resources. It also helps automate the application life cycle from development and into production.

Why Open Source?

Linux is often the heart (and soul and brains) for cloud-native infrastructures. In fact, open source code is everywhere in today's cloud environments and underlies cloud platforms that deliver business applications and environments.

Cloud-native software is often developed, deployed, and managed using DevOps practices, serving the need for highly flexible and agile development environments with reduced IT complexity. Developers increasingly package application components in Linux containers that run as microservices across many different types of clouds.

Open source provides a reasonable assurance of global connectivity and interoperability among applications and services. It also provides a firm foundation to avoid vendor lock-in and achieve easy workload positioning and movement.

Chapter **2**

Architecting to Maximize Flexibility

Cloud computing's growth curve is expected to steepen. Trends such as containerization, serverless architectures, and the extension of the edge into the cloud, are essential cloud computing usage trends. As the world gets more connected, users want everything to be software-defined. The growth of the Internet of Things (IoT) and cloud services should ramp up the cloud even further. Increasing popularity and reliance on the cloud means any good cloud strategy must be flexible to meet new (and possibly unforeseen) demands.

Planning for "Max Flexibility"

If you want to plan for maximum cloud flexibility, you need to ponder a variety of different considerations:

» **Portability and manageability:** The biggest enabler for flexibility comes from adopting an open and extensible architecture. It also helps avoid vendor lock-in and proprietary implementations.

>> **Regulatory requirements:** The private versus public decision often comes from the data itself, where regulatory concerns may specify that data can't leave a certain country or be subject to certain audit requirements. Such data may best reside in a private cloud under tight control, while other parts of an application (such as stateless web servers) not subject to restrictions could reside in a public cloud.

>> **IT security:** Security is emerging as an overriding concern and focus for investment and build-out in enterprise networks, with a special focus on securing the cloud.

Among the 1,000-plus IT professionals that Red Hat surveyed for its 2021 global tech outlook report, nearly half (45 percent) mentioned IT security as the organization's top IT funding priority. Close behind: IT/cloud management (39 percent) and cloud infrastructure (34 percent). The nexus of security and cloud is top of mind at present and for the foreseeable future. Get more details at `www.redhat.com/en/global-tech-outlook-report/2021`.

>> **Geo-redundancy:** Applications that need strong resilience and high availability can benefit from a hybrid cloud model. This allows you to divide services and endpoints between multiple private and public clouds. In this kind of scenario, if a private cloud fails, you could elect to recover the service in a public cloud. Similarly, a hybrid cloud reduces risks of data loss or inaccessibility.

>> **Best practices:** Considering software life cycles in the cloud is vital as you test, develop, and eventually shift application environments from development into production. The same notion applies to private cloud infrastructures. After all, a private cloud infrastructure supports software applications that must be maintained and tested through their life cycle, too. This process lets a production environment run unhampered and unhindered, ready for workaday use, with development and testing (on different versions) in parallel.

As you work your way through this list, consider your own goals and priorities, and adjust accordingly. You undoubtedly need to be flexible (pun intended) when it comes to maximizing flexibility.

REMEMBER Conversations around cloud flexibility often default to the hybrid option. Creating the right mix between public and private clouds requires creating a roadmap. It begins with understanding overall goals for cloud computing. Bring in all your business stakeholders (key decision makers from business units that benefit from cloud computing) and start laying things out. Such a roadmap need not look years and years ahead, but it must be built to handle change with sufficient flexibility to adapt to and accommodate new services and deliverables. A forward-looking approach lets you take advantage of new capabilities and functionality from the cloud as it continues to innovate and expand.

Getting from Idea to Production

After your roadmap is laid out, what's next? Good question! The next steps involve discovery and gap analysis. That's because making cloud decisions involves a lot of choices, both technical and non-technical. Making choices must begin with a careful analysis of your organizational constraints and the options ahead. In turn, this comes from thorough discovery and analysis of current processes and outcomes. Along the way, you'll have to observe (and later on, manage) how the options you choose (or experiment with) affect interactions between operational and development teams along with other stakeholders in the organization.

Selection is also closely tied to proof of concept (POC) stages. That's because a POC usually provides the best way to inspect and experiment with service choices that follow from an initial evaluation. A POC provides a great opportunity to evaluate technologies and outcomes as they apply to specific business requirements. At the same time, you'll gain insight into possible or useful integrations.

The next step on the road to production is architecture design. Thoughtful technology and organization design is important because it impacts the future direction for IT for a long time to come. It's never easy to go back and rework things if the design isn't right. Here again, you must work closely with the organization's stakeholders and with vendor partners to assess your workload and applications.

WARNING

Architecture design is an exercise where redos are insanely expensive and can even be career-limiting (or -ending) maneuvers. This situation is clearly one where you *don't* want to follow that old maxim: "There's never enough time to do it right, but there's always enough time to do it over." Not so in this case, so don't even think about it. That said, design is when it's essential to ponder (and choose) architectures that are open and flexible enough to accommodate as-yet-unknown business requirements, new and emerging technologies, and open-ended development environments and tools.

TIP

The best way to future-proof an IT infrastructure is to design flexibility and openness in from the get-go. Make this a priority.

Circling Back to Cloud Strategy

As you make choices — particularly, those that involve choosing specific platforms, tools, or technologies — you can't help but notice their impacts on your organization, your processes, your policies, and even your business goals. It's important to recognize that not all proofs of concept succeed. Often, such failures require circling back to and adjusting strategy to reflect an improved and more realistic understanding of what's really needed. Don't be afraid to make adjustments as you go.

In fact, many experts argue that developing a proper cloud strategy unavoidably includes at least a little bit of a "two steps forward, and one step back" motion. That's because you have to move forward to select possible options, and implement POC experiments, while recognizing that not all such concepts actually prove themselves in practice — however attractive they may be in theory. Therefore, it's safest to neither assume that your progress will always be straightforward — nor that a chart of such progress will necessarily be a straight line from the starting point to the finish line. This is another reason why it's so important to be flexible when working with cloud strategy and related platforms, tools, and technologies.

Chapter **3**

Enabling Agile IT in the Cloud

Flexibility is key to any workable cloud strategy. It allows many different technologies to run across various cloud architectures, whether private, public, or hybrid. For example, do you want to run storage services on–premises and in the public cloud for backup? Or across clouds? You can do that. Better yet, technologies such as containers and Kubernetes/OpenShift open the door to truly portable workloads because they work the same wherever they run.

Automation makes cloud infrastructure and applications more streamlined and less complex and helps staff meet business objectives within organizational and regulatory constraints. In this chapter, you look at how the cloud and automation go together, and how managing infrastructure like you manage code builds in added flexibility and capability.

Living in the Cloud Age

Back in the day, getting a new server up and running was time intensive. An administrator had to work with a vendor to buy a physical server, configure it, connect it to the network, install applications, test it, and image it. If something went wrong, it

was time to start over. Weeks or months could pass before a server went live. Then, consider the time involved when many servers across different environments are affected.

No more! In the Cloud Age, admins and users can provision a server on demand. This takes mere minutes (not days, weeks, or months) by selecting parameters on a friendly user interface or via an API call. This enormous reduction in delivery time has caused the number of servers across enterprises to skyrocket. Today, the real challenge is to find a flexible, robust solution to manage all of them.

Understanding Infrastructure as Code

The answer to managing large numbers of on-demand servers is *Infrastructure as Code* (IaC). Using IaC operators can provision and manage IT systems programmatically with a template defined in software code. If you're thinking this sounds a lot like application development, you're right.

The crux of IaC is a definition file, from which the entire-infrastructure — networks, storage, virtual machines, and so on — is built. Such a playbook specifies not only infrastructure elements but also how they should be configured and sets the stage for consistency. With a definitive file, automation, and configuration management software, such as Ansible, Chef, or Puppet, use it to configure and provision cloud infrastructure. And it can do so programmatically, in response to demand and usage changes, or user requests.

Best IaC Practices

As IaC makes its way into more cloud environments, ensure your team follows best practices as a standard for execution. The following best practices apply to every IaC environment:

» **Use definition files as documentation:** Definition files are the essence of IaC — your blueprints for whatever type of resource you're creating — and they serve as documentation. In short, your code becomes your doc.

» **Your documentation is built into your code:** Because everything that describes infrastructure is in code, it's vital that code be crystal clear and self-explanatory.

- » **Version everything:** Use a version control system, such as Git (see the nearby sidebar for details), to track changes, conduct peer reviews, and provide good governance. Versioning is especially handy for audits and compliance.
- » **Validate templates before putting them to use:** Validating a template before creating or updating resources lets you identify syntax errors, misconfigurations, and dependency problems.
- » **Perform continuous testing:** This invokes the big picture to validate templates. Constantly test systems and processes to correct mistakes quickly and ensure changes don't create instability or unexpected performance issues.
- » **Make small, incremental changes:** Making a small change allows you to quickly see the effect and roll it back if needed with minimal effort. Batch changes may seem faster but are more difficult to troubleshoot.
- » **Keep services available:** If a server fails, another should be ready to take over. The goal is to deliver uninterrupted services per service-level agreements (SLAs).

THE GITOPS VISION

GitOps is a way to develop IaC that uses Git repositories as a sole source of truth. Submitted code triggers a standard pipeline for continuous integration/continuous development (CI/CD) in DevOps fashion. This ensures that well-documented, policy-driven requirements for security, IaC itself, or boundaries in the application framework are met or exceeded. All code changes get tracked in such an environment. This makes updates easy to move through the life cycle and provides version control should rollbacks prove necessary.

GitOps delivers the following benefits:

- Standardized workflow for application management
- Increased security through baking in application requirements
- Improved reliability with Git-supplied visibility and version control
- Consistency across clusters, clouds, and on-prem environments

Tools may be combined to build a usable GitOps framework, such as Git repositories, Kubernetes, plus CI/CD and configuration management tools.

The Six-Phase Iterative Approach to Cloud Infrastructure

IaC is what provides real payoffs from using the cloud. If you're ready to incorporate IaC into your IT organization, follow a structured approach to planning and design for rollout. To flesh out and implement your cloud strategy using IaC principles and practices, follow this checklist:

» **Discover:** Nail down and sort out short-term and long-term requirements with IT and business partners. This process includes identifying challenges, articulating business objectives, and determining workloads to move to the cloud.

» **Design and build:** Make sure the design fits your specific business strategies and use cases for the delivery of a Minimally Viable Product (MVP) that can scale as your company grows.

» **Testing/Validation:** Validate your technical design and try service features, assess their life cycle (for example, how they're updated/upgraded), see how they work, and assess organization fit and impact.

» **Migration plan:** Set a strategy for how best to migrate your selected applications. Standardize and establish automation mechanisms and operational processes.

» **Operationalize:** Tune your cloud infrastructure to meet your performance needs and add/integrate the operational tools you need to detect and respond to failures at lightning speed. The question isn't whether your infrastructure or application will fail; it's all about when and how fast you detect and recover from faults. The real goal is for no failure to be visible or noticeable to your end-users.

» **Iterate:** Revisit your initial MVP design for further improvements. Design changes to accommodate new business needs. Don't try to "boil the ocean" in your first attempt. Define increments and sprints with clear achievable outcomes that are time bound.

This process helps you look at all possible technologies and solutions and to come up with a solid action plan. It creates the roadmap for seeing your cloud project through from beginning to end.

Chapter **4**

Adapting to a Hybrid World

Check any cloud usage survey you like. Notice that enterprises (and other organizations) think "more is better" when it comes to cloud adoption and consumption. In fact, *2021 Global Tech Outlook: A Red Hat report* found 27 percent of the enterprises it surveyed have an explicit hybrid cloud strategy, while 11 percent call their strategies multicloud (as in more than one). Put them together, and 38 percent have a cloud strategy that involves two or more clouds. The same study found nearly all enterprises surveyed plan to increase the number of clouds they use. Today, using two is typical: 60 percent of respondents report using two or more cloud platforms. You can get more info about this report at www.redhat.com/en/global-tech-outlook-report/2021.

It's hard to argue against the assertion that deploying a hybrid cloud makes good sense to more organizations, given that going hybrid enables cloud-based and on-premises resources and assets to work together better.

Using Hybrid Infrastructures

Many modern organizations already use hybrid IT infrastructures today. Given that a hybrid cloud is an IT architecture that provides some degree of workload portability, orchestration, and management across two or more environments, you may find it in these scenarios:

>> At least one each of private and public clouds

>> Two or more private clouds, and/or public clouds

>> A bare-metal or virtual environment connected to one or more public and/or private clouds

Today, it makes more sense to define hybrid cloud computing by what it *does*, instead of what kind of or how many clouds it uses. In fact, modern hybrid clouds should

>> Connect multiple computers over a network.

>> Consolidate IT resources.

>> Scale out easily and quickly provision new resources.

>> Facilitate moving workloads between environments.

>> Run a single, unified management tool (and interface).

>> Orchestrate processes using automation.

The infrastructure that supports a hybrid cloud made of private and public cloud components works the same as standalone private and public clouds. Therefore, it includes

>> Networks, such as local area networks (LANs), wide area networks (WANs), virtual private networks (VPNs), and/or application programming interfaces (APIs), that connect multiple computers

>> Virtualization, containers, and data services abstract resources that may then be pooled together into data lakes

>> Management software that allocates resources into environments where applications and services run, provisioned on demand via an authentication service

Individual clouds become hybrid when application or service environments interconnect seamlessly. Interconnectivity is what makes hybrid clouds work and explains why they're the foundation for edge computing.

Edge computing describes a computing model that distributes compute resources out at the "edge" of a network — where the users and Internet of Things (IoT) devices and sensors reside — as needed and also centralizes resources in a cloud model whenever possible. Edge computing can quickly provide actionable insights derived from time- (and location-) sensitive data. Edge computing use cases often involve IoT, mobile technologies, and telecommunications.

Interconnectivity and interoperability in a hybrid cloud explain how workloads move around and how management gets unified across multiple clouds, all of which use APIs to interact with a single set of management tools and interfaces. And finally, this details how orchestration runs processes in the "right" cloud, depending on service-level agreements (SLAs), security, compliance requirements, and so on.

Choosing Hybrid-Happy Platforms

A handful of basic principles describe two general ways to build a hybrid cloud environment. One is traditional; the other is more modern (and represents the best way to achieve future-proof, flexible business outcomes). Traditional methods mean interconnecting private and public cloud environments using massive, complex middleware. One may build a private cloud from scratch or use a prepackaged cloud infrastructure. Linking private and public elements may use middleware or a special VPN that public cloud providers include in subscriptions, such as AWS: Direct Connect, Azure: ExpressRoute, Google Cloud: Dedicated Interconnect, and OpenStack: Public Cloud Passport.

Using such connections to tie clouds together takes time and effort. That's why modern hybrid clouds focus instead on portability for apps that run inside them. Focusing on apps means building them as collections of small, independent, loosely coupled services. With the same operating system in every IT environment, and managing everything on a unified platform, apps run everywhere with equal ease and facility. They can also move easily as well.

> **TIP**
>
> Think of a hybrid cloud as resulting from running the open source Linux OS everywhere, building and deploying cloud-native apps, and managing environments and apps using an orchestration tool like Kubernetes or Red Hat OpenShift.

Using the same OS abstracts all hardware requirements, just as using an application platform abstracts app requirements. This creates an interconnected, consistent computing environment where apps can move from one place to another quickly and easily. There's no need to set up and manage APIs for each different connection. Better yet, you don't need to worry about connections breaking when apps get updates or move from one cloud to another.

Embracing Kubernetes

Kubernetes is an open source container-orchestration platform designed to automate how containerized applications are deployed, how they scale up and down, and how they're managed. Kubernetes works across a wide range of infrastructure environments. Most cloud services offer Kubernetes-based platforms as a service — that is, as explicit Platform-as-a-Service (PaaS) or Infrastructure-as-a-Service (IaaS) environments. This lets Kubernetes function as the platform on which container-based applications can run, scale, and be managed.

> **REMEMBER**
>
> Kubernetes is a de facto operating system for modern hybrid cloud environments. That is, if you build an application to run in a Kubernetes-managed container, you can be sure it will run in any cloud or datacenter environment that supports Kubernetes. Because nearly all of them do, it's like getting a free pass for interoperability, scaling and orchestration, and management wherever the app runs.

The Kubernetes platform makes it easy to provide consistency across different cloud platforms because applications and their data reside within containers that come prepackaged with all inputs and outputs, capabilities, and connections needed, regardless of the runtime environment (which may be private or public cloud-based, on-premises or off).

> **REMEMBER**
>
> Because Kubernetes and containers are built from Linux, it's best to use the same Linux distribution all over. That includes the Kubernetes node hosts and within containers themselves.

Getting Past Plain Vanilla

TIP

Maximizing Kubernetes is more than simply grabbing code and putting it to work. Some distributions include additional features that confer amazing value. Look for, or insist on, certain features from your toolset to create a flexible and future-proof hybrid cloud environment. Ditto for containerized applications in that environment. The features include

>> **Application services:** Applications in Kubernetes can use generic services that work across all containers in a cluster. Services follow a standard TCP/IP model, using TCP or UDP port addresses. A good Kubernetes implementation offers a large library of predefined and well-known services and also makes it easy for developers to build their own services so they can easily leverage common efforts and create reusable building blocks.

>> **Data services:** OpenShift Container Storage Interface (CSI) snapshot functionality offers customizable, point-in-time snapshots of persistent data volumes. These work with enterprise backup solutions to support Kubernetes apps. They use APIs that provide application-consistent backup images, complete with metadata to describe cluster resources and persistent data volume contents.

>> **DevOps tooling:** Kubernetes is about running apps in containers. Modern app development adheres to a combination of development and operations philosophies known as DevOps. Key to this approach is for continuous integration and continuous deployment (CI/CD). In practice, this means your Kubernetes environment should integrate easily and seamlessly with your development and deployment tools, and support CI/CD.

>> **ISV marketplace support:** Enterprises can be comfortable buying tools and technologies for Kubernetes application development, deployment, and management if prospective purchases are certified for Kubernetes. Red Hat operates an open marketplace specifically for such software from independent software vendors (ISVs) to simplify buying and deploying container-based software across clouds. Find it at `marketplace.redhat.com`.

>> **Cloud services integration:** Kubernetes and Red Hat OpenShift work with the major cloud platforms, including AWS, Azure, IBM,

and Google. Cloud interoperability and access are essential in any modern hybrid cloud infrastructure and provide the foundations for workload migration, easy deployment and management, and more.

Making Much of Management

In a dynamic and distributed environment, with containers and clusters in multiple clouds, management and monitoring gain extra importance and value. A single view of core capabilities is key. Centralized management translates into

» Control of public clouds, including AWS, Azure, Google Cloud Platform, IBM Cloud, and others

» Management of key technologies used in the cloud and on-premises, including virtualization (VMWare, OpenStack, and more), varied operating systems (Linux, Windows, and more)

» End-to-end cluster management, using Infrastructure as Code (IaC) best practices and design principles to deliver reliable consistent management at-scale, cover clusters across multiple datacenters and public cloud services, illuminate health across all clusters and pods, and provide troubleshooting insight across all clusters

» Baked-in security that lets developer and operations teams work from the same playbook and account for compliance and regulatory requirements through the entire life cycle

REMEMBER

A strong management solution offers policy-based governance, risk, and compliance controls.

» Centralized life-cycle management for containerized apps that allows for intelligent placement rules, uses channel and subscription definitions for deployment, provides usable views of service endpoints and pods, and facilitates moving workloads across clusters — even across public clouds

» Better results as integrated dashboards and tooling bring everybody together to get the work done consistently

REMEMBER

A strong management solution helps break down departmental boundaries and encourages collaboration and cooperation across the organization.

Chapter **5**

Cloud Strategy for Telcos

For telecommunications companies — telcos, for short — digital transformation isn't just an idle dream. It's a make-or-break proposition. Service providers that can't lead the way into modern, progressive technologies must fall by the wayside because telcos, first and foremost, connect us all — companies, people, processes, and devices. In turn, many companies rely on telcos to provide solutions to help them better serve their own clients and customers. Today, clouds are where IT and applications live, so if telco infrastructure isn't modern (moving toward cloud-native capabilities) and won't or can't support hybrid clouds easily or very well, everybody comes up short. That's simply unacceptable.

REMEMBER

Digital transformation describes what happens to companies — including telcos — as they fundamentally change the way they do business by adopting innovative ways to offer products and services built on new digital tools and technology. For success, though, organizational culture change must coincide because transformation requires new ways of thinking and doing to improve or replace what came before.

Telcos are at the forefront of digital transformation because their services and infrastructure form the foundation on top of which digital transformation rests. It's fair to characterize them as digital service providers (DSPs). That means clouds, mobile apps, and everything-as-a-service require new kinds of storage, analytics,

automation, networking functions, and management. Savvy DSPs must supply and support all this, so their customers can build their futures.

Many telco CEOs are looking to accelerate the transition of their companies into DSPs. But they must execute on a holistic strategy so it doesn't fall prey to short-term budget limitations or fail to deliver on over-ambitious, poorly conceived implementation plans. Instead, telcos must carefully weigh the business benefits for each element they build into their plans.

Finding Opportunity in a Paradigm Shift

When the rules change enough that you find yourself asking, "Are we still playing the same game?" that's a paradigm shift. Digital transformation — especially for telcos — represents a clear paradigm shift because it literally "changes everything." Early adopters of new tools and technologies can capture more of the opportunities created. At a minimum, digital transformation for DSPs involves a cluster of key ingredients:

» **Network functions virtualization (NFV):** Virtualizes networking functions (routing, filtering, prioritizing, orchestrating, and organizing network traffic and structures) used as the building blocks that create fully fledged communication services

REMEMBER

Telcos helped create NFV and have pushed implementations to meet telco-grade standards. *Telco-grade* designates systems and equipment that support the levels of capability, reliability, stability, quality, and availability required by telcos to provide the infrastructures that everybody relies on and that customers can bundle with their own services.

» **Cloud-native applications and services:** Depend on using open source, standardized containers and Kubernetes, and associated infrastructures, development tools, environments, and more

TIP

Telcos should seek out a cloud-native platform that allows the same operation across most, if not all, cloud infrastructures supporting containers.

» **Automation:** Covers the use of programming or scripting to capture commands and configurations and the use of management and control software to orchestrate its use

Automation is key to scalability because it responds to events, requests for service, or incidents without human intervention or error. Automation, once proven and tested, is accurate, works at machine (not human) speeds, and may be repeated. Automation usually works within something like open source Ansible, which supports software, service and network provisioning, configuration management, and deployment tools. It enables infrastructure as code (IaC) — see Chapter 3 for more information.

» **Artificial intelligence (AI) and machine learning (ML):** Represent the use of computers to analyze enormous collections of data that modern digital devices, systems, and services generate

ML, in particular, is able to teach itself new and interesting ways to understand data, some of which may be counterintuitive or too complex for humans to handle unaided. AI and ML make sense of, protect, and enhance the value of data within complex systems. They're what lets online shopping sites suggest purchases based on observed shopping behavior, supports fraud detection for financial services providers, and helps software developers find and fix potential issues before they turn into real problems. AI and ML excel at building baselines and finding anomalies in all data. This is of great value to all organizations.

Paradigm shift opportunities go mostly to organizations that quickly recognize the value brought by major changes and that use new technologies to make it easy for clients or customers to benefit. In leading the way to digital transformation for organizations of all sizes, telcos are uniquely positioned to benefit from providing services and infrastructures that support their customers' effective use of hybrid cloud and the cloud-native applications that run there.

Interconnecting the World

A modern hybrid cloud environment creates a veritable ecosystem within which telcos — and their legions of customers — can survive (and thrive) amidst the trials of digital transformation. This

ecosystem depends on a key set of foundational ingredients that, in turn, depend on

>> Running the open source Linux OS everywhere

>> Virtualizing or containerizing network functions

>> Providing a consistent way to build and deploy cloud-native apps

>> Managing hybrid environments with the Kubernetes orchestration found in Red Hat OpenShift

Red Hat OpenShift, integrating Kubernetes and other open source technologies, takes the work out of finding and resolving interoperation issues across public and private cloud platforms. It also makes creating and running a hybrid cloud environment easier and relatively straightforward, making it particularly appealing for telcos.

Building blocks for modern telco networks

As the entire telco industry moves toward 5G adoption and deployment, new technology helps deliver more capabilities to customers faster, improve their experiences, and accelerate digital transformation. This involves rethinking the entire telco-grade environment and switching away from proprietary hardware and software to open, standards-based services and software-defined systems to accelerate and amplify new 5G capabilities, support media and entertainment streams, and more.

The distributed architecture of 5G also permits DSPs to offer services at the network edge, a priority for telco customers who want to improve data access and application responsiveness. Examples include access to complex medical imaging and diagnostic services in an ambulance or AI-driven operation of an autonomous vehicle in real traffic.

DSPs can use hybrid cloud to extend datacenter resources while maintaining control over increasing presence and capability at the edge. Edge computing is key to enabling 5G and its new service opportunities. Many depend on localized compute resources and data acquired and consumed at the network edge (especially for IoT devices and sensors, or next-gen augmented reality gaming,

COUNTING GENERATIONS: 1G, 2G, . . .

In the world of wireless, connections are often labeled using numbered generations, from 1G all the way to 5G. Here's what each of these generations represents:

- **1G:** First generation cellular telecom standards were introduced in 1979 and are no longer used since 2G came along.

- **2G:** Second generation wireless standards include GSM, introduced in 1991. It offers digital encryption, better wireless efficiency, and early data services such as Short Message Service (SMS) for texting.

- **3G:** Introduced in 1998, 3G offers faster data rates and broadband access to smartphones and mobile computer modems. As with subsequent generations, 3G takes advantage of new frequency bands.

- **4G:** First introduced in Scandinavia in 2009 and the USA in 2011, 4G improves mobile broadband for Internet access, adding IP telephony, video conferencing, and other advanced services. Another common label for 4G is Long Term Evolution (LTE).

- **5G:** Offers data rates of hundreds of megabits per second (Mbps), up to 1 gigabit per second (Gbps). With its first large-scale deployments in April 2019, 5G also supports massive scaling for Internet of Things (IoT) sensors, better transmission efficiency and coverage, and lower latency.

With each new generation, speeds have increased as has the number, type, and complexity of digital streams supported. 5G, in fact, appears poised to challenge wired and cable connections for the "last mile" (from the edge of the infrastructure to homes and offices).

or vision–controlled factory automation that demand both limited latency and intensive processing).

Indeed, open source and the hybrid cloud are emerging as the building blocks for modern telco agile networks, enhancing customer experiences, even when their needs change.

The open DSP journey

Telcos today are continuing their ongoing digital transformations. And now, compute and storage at the network edge support 5G, reduce latency and congestion, and improve cloud-native application performance. This lets telcos create new vertical business opportunities.

Greater use of open, flexible cloud and networking technologies has allowed DSPs increasingly to recognize that they shouldn't rely on a single vendor to drive network innovation, especially as networks become more virtual than physical. Open community collaboration accelerates innovation. Using an open, flexible environment also permits DSPs to add or change vendors as their business needs, cost considerations, and customer demands evolve, using the best solutions to offer top-notch customer experiences.

With a mature, stable, and open foundation for hybrid cloud infrastructures and their operations, telcos can quickly add new functions, services, tools, and technologies. By taking advantage of open technologies — APIs, standard containers, and management/automation tools — DSPs can meet customers' needs, whatever they're doing, wherever they're located, on whatever devices they're using, and whatever content they're consuming. Rapid innovation helps telcos compete more successfully in today's dynamic markets.

Modernizing OSS/BSS

Internal telco operations use operations support systems and business support systems, often abbreviated as OSS/BSS. The former addresses how telcos configure, provision, maintain, and troubleshoot network services, while the latter addresses how telcos track service orders, manage customer relationships, handle billing, and manage cross-carrier transactions. These are the nuts-and-bolts internal systems on which telcos depend, but like all other aspects of modern business, they too must change as a part of digital transformation.

OSS/BSS gain many of the same benefits that other applications and services enjoy when switched to a cloud-native architecture

and hybrid cloud ecosystem. The new capability, flexibility, and efficiency they bring help modernize OSS/BSS.

The ecosystem supports a broad range of solutions, including databases, data stores, AI/ML analytics, and more, designed to enable a modern OSS/BSS environment. In turn, this makes telcos more agile, more flexible, and more able to innovate with new services, bundles, partnerships, and customer offerings.

Winning DSPs Serve Customer Success

DSPs can plan on future benefits from a hybrid cloud adoption and deployment. A hybrid cloud ecosystem makes it easier and faster to migrate workloads among public and private clouds. DSPs can read-ily shift from internal infrastructures to public clouds across mul-tiple providers, if desired, and support cloudbursting when needed.

REMEMBER

Cloudbursting refers to responding to increased demand for an application or service by placing additional workloads for it into one or more additional clouds (usually public). It supports scaling up and scaling out.

DSPs must understand the impact on customer experiences. Stra-tegic initiatives that benefit from hybrid cloud adoption include

» **Strengthening customer digital relationships:** By using data analytics and AI/ML, telcos can meet and predict customer preferences and requirements more accurately and provide them with better service. In addition, aggregat-ing customer data in a cloud-based datastore, instead of spreading it across multiple systems (and locations), lets telcos create a 360-degree, holistic view of their customers. This plays nicely into customer interactions and feeds better into models for customer behavior.

» **Delivering new, value-added B2B capabilities:** DSPs must be able to deliver customized, carefully crafted solutions to enterprises. And they need to do this quickly and affordably. To easily team up with enterprise partners, public cloud services must be equally accessible at all times to all parties involved. In a hybrid cloud ecosystem, cloud-native applica-tions are ubiquitous, and consequently, everything is easily accessible, flexible, and extensible.

>> **Gaining a competitive edge by building new revenue streams around 5G:** New core and edge capabilities are necessary for telcos to better monetize their infrastructure investments. Edge clouds offer all the benefits associated with more distant public clouds but with lower latency and more immediate access to compute and storage resources. The real trick is to identify the best opportunities that 5G and edge computing present and to deliver differentiated cloud-based services with more agility and speed, using the hybrid cloud ecosystem to obtain a first-to-market (or early-to-market) advantage.

TIP The biggest benefits of the hybrid cloud ecosystem are probably yet to be discovered, but the necessary tools and scaffolding are already available. Telcos can seek out and capitalize on the oppor–tunities as they emerge.

Chapter **6**

Ten Reasons to Develop a Cloud Strategy

Each *For Dummies* book ends with a Part of Tens chapter. This one gives you reasons to develop a cloud strategy:

» **You get a roadmap:** A cloud strategy provides a roadmap for becoming cloud native. You'll understand your goals and objectives, what you can migrate to the cloud, which challenges you face, and how to overcome them.

» **Increase your cloud flexibility:** A cloud strategy also helps you ramp up cloud services appropriately to meet both existing and new (and possibly unforeseen) demands.

» **Go open source:** An architecture based on open source software is the cornerstone of a cloud strategy. Adopting an open and extensible architecture adds flexibility and portability (and avoids vendor lock-in).

» **Meet compliance requirements:** A well-developed cloud strategy helps you ensure that you're meeting regulatory and internal compliance requirements. It also designs in policies, tests, and checks to compare and synchronize what's required and what's built.

- » **Follow a proven approach:** A cloud strategy provides a checklist to ensure nothing is overlooked. The checklist covers discovery, design and build, testing and proof of concept, and migration.

- » **Reach your goals more quickly:** Making cloud decisions involves many technical and business choices, input from operational and development teams, and proofs of concept. A solid strategy ties it all together and keeps choices on track toward achieving business outcomes.

- » **Increase agility with automation:** Infrastructure as Code (IaC) lets you provision and manage servers automatically by using a defined template. There's no need to touch a physical box. IaC is the essence of agility. Check out Chapter 3 for more information about IaC.

- » **Embrace Kubernetes:** A cloud strategy looks to replace one-off, manually configured runtimes and applications with cloud-based containerized equivalents that are standardized and automated and can be quickly created, turned off, or replaced. See Chapter 4 for more info.

- » **Unified IT management is in play:** Single-pane-of-glass management increases efficiency, makes complex environments easier to manage, and builds in reliability and scalability.

- » **Get better visibility:** Unified IT management results in highly accurate data. Plan better, spend more wisely, and lower overall IT costs. From GitOps to global management, a cloud strategy illuminates the entire IT life cycle and drives continuous improvement.

Put together your hybrid cloud strategy

Cloud computing has become a mainstay in the world of IT. Trends such as containerization and serverless architectures, and the extension of the edge into the cloud, are essential cloud computing trends. As the world gets more connected, users want everything to be software-defined. Increasing popularity and reliance on the cloud means any good cloud strategy must be flexible to meet new or unforeseen demands. With this book, you see how to put together your cloud strategy and plan for the future.

Red Hat | intel

Inside…

- Key architecture design considerations
- Infrastructure as Code best practices
- Hybrid cloud platforms and technologies
- Cloud strategy for service providers
- Ten reasons to develop a cloud strategy

Ed Tittel is a long-time IT industry writer and researcher who covers Windows, networking, security, and cloud computing topics. He's contributed to over 100 computing books, including various For Dummies titles, and writes regularly for Computer-World. For more information, visit edtittel.com.

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.