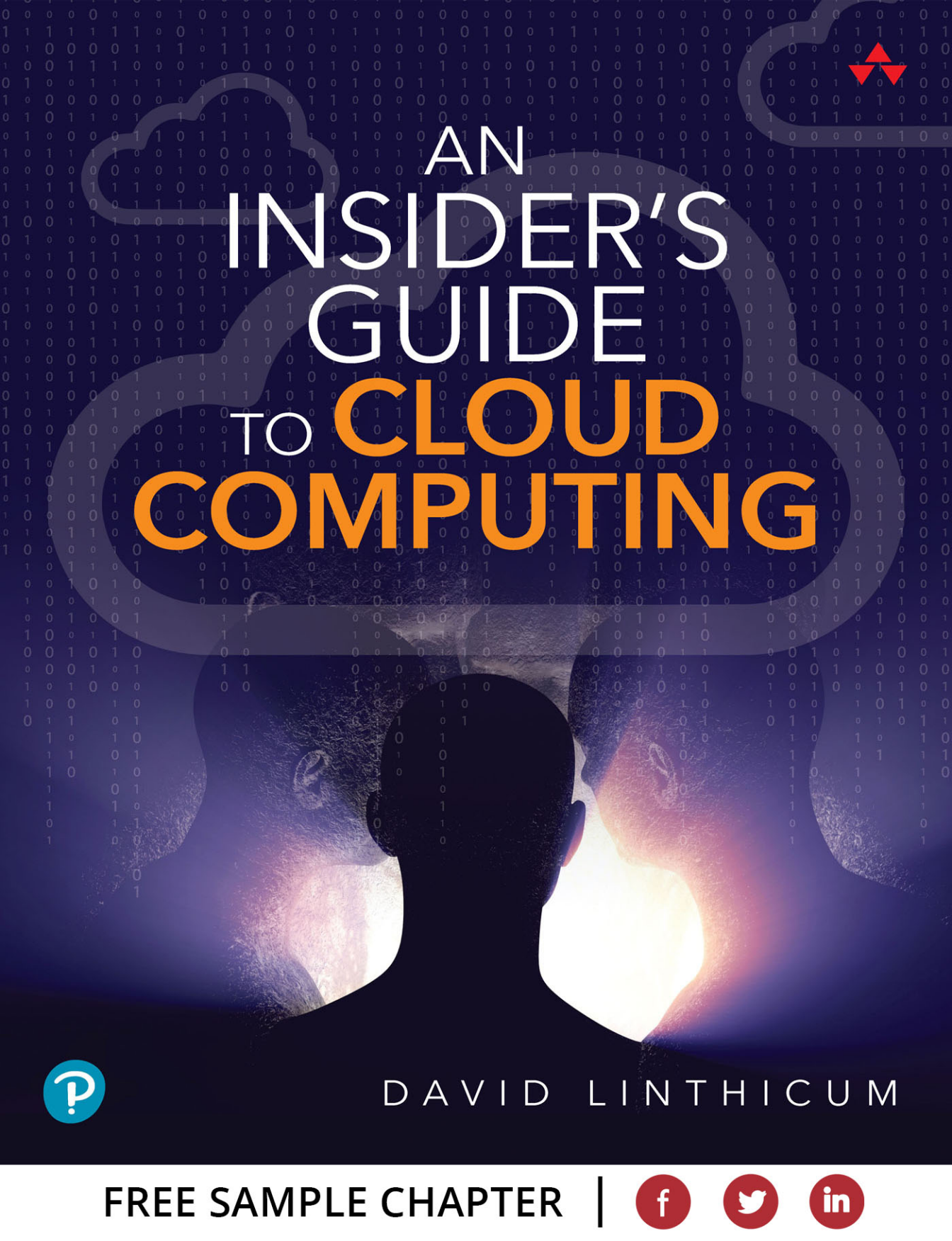

FREE SAMPLE CHAPTER |

An Insider's Guide to Cloud Computing

David Linthicum

PEARSON

An Insider's Guide to Cloud Computing

ISBN-13: 978-0-13-793569-7
ISBN-10: 0-13-793569-2

Library of Congress Control Number: 2022923586

ScoutAutomatedPrintCode

Trademarks

Warning and Disclaimer

Vice President, IT Professional
Mark Taub

Director, ITP Product Management
Brett Bartow

Executive Editor
Nancy Davis

Development Editor
Ellie Bru

Managing Editor
Sandra Schroeder

Project Editor
Mandie Frank

Copy Editor
Chuck Hutchinson

Indexer
Tim Wright

Proofreader
Barbara Mack

Technical Reviewer
Jo Peterson

Editorial Assistant
Cindy Teeters

Designer
Chuti Prasertsith

Compositor
codeMantra

Special Sales

For information about buying this title in bulk quantities, or for special sales opportunities (which may include electronic versions; custom cover designs; and content particular to your business, training goals, marketing focus, or branding interests), please contact our corporate sales department at corpsales@pearsoned.com or (800) 382-3419.

For government sales inquiries, please contact governmentsales@pearsoned.com.

For questions about sales outside the U.S., please contact intlcs@pearson.com.

Cover image credit: Space Wind/Shutterstock

Pearson's Commitment to Diversity, Equity, and Inclusion

Pearson is dedicated to creating bias-free content that reflects the diversity of all learners. We embrace the many dimensions of diversity, including but not limited to race, ethnicity, gender, socioeconomic status, ability, age, sexual orientation, and religious or political beliefs.

Education is a powerful force for equity and change in our world. It has the potential to deliver opportunities that improve lives and enable economic mobility. As we work with authors to create content for every product and service, we acknowledge our responsibility to demonstrate inclusivity and incorporate diverse scholarship so that everyone can achieve their potential through learning. As the world's leading learning company, we have a duty to help drive change and live up to our purpose to help more people create a better life for themselves and to create a better world.

Our ambition is to purposefully contribute to a world where:

- Everyone has an equitable and lifelong opportunity to succeed through learning.
- Our educational products and services are inclusive and represent the rich diversity of learners.
- Our educational content accurately reflects the histories and experiences of the learners we serve.
- Our educational content prompts deeper discussions with learners and motivates them to expand their own learning (and worldview).

While we work hard to present unbiased content, we want to hear from you about any concerns or needs with this Pearson product so that we can investigate and address them.

- Please contact us with concerns about any potential bias at https://www.pearson.com/report-bias.html.

Contents at a Glance

Table of Contents

About the Author

David Linthicum is on most top-10 lists of technology innovators and influencers, including cloud computing, edge computing, AI, and security technology. David is a best-selling author of more than 15 books and more than 7,000 published articles. He is also the originator of many business-related technology concepts, including enterprise application integration (EAI). He's an innovator within service-oriented architecture (SOA), and now cloud computing and the use of cloud computing for digital transformations.

With his remarkable ability to design, explain, and implement technology solutions to solve existing business problems and create new opportunities, David rose rapidly through the corporate ranks from programmer to CEO, with stops in between that helped inform his holistic view of enterprises. Based in Washington, DC, David currently serves Global 2,000 clients as Deloitte's Chief Cloud Strategy Officer, where he drives new innovations and market offerings, and leads people and projects, as well as thought leadership outreach. This includes on the "Deloitte On Cloud Podcast," as well as several *Forbes* and *WSJ* articles.

David's 60+ courses on LinkedIn Learning consistently appear on the "Popular Courses" list and provide course content on cloud computing, cloud architecture, cloud security, cloud governance, cloud operations, AI, DevOps, and many other concepts related to cloud computing and enterprise technology in general. He's also an adjunct professor for Louisiana State University (LSU), where he's created courses on DevOps, Cloud Computing, Cloud Architecture, and other courses that are in demand by the LSU student body. David has done over 1,000 conference presentations in the U.S. and abroad, often as a keynote speaker at conferences related to enterprise technology. He has hosted over 2,000 Webinars on the correct use of enterprise technology, including cloud computing, edge computing, AI, DevOps, and data science.

To my father,
Ronald Gerald Linthicum,
April 7, 1939 - March 27, 2014.
None of my accomplishments have been made without your encouragement and guidance. We miss you, Dad.

Acknowledgments

There are a bunch of people to thank for this book, including Nancy Davis and Ellie C. Bru at Pearson. Also, many people behind the scenes who do the copy editing, layouts, graphics, and even the physical printing. Also, the audio engineers who assisted with the audio version of this book.

I also want to thank Thomas Erl, a major tech author and friend, who was instrumental in getting me back in touch with Pearson, after 13 years since my last book, and myself swearing off book authoring in favor of blogs and podcasts. I'm back as a book author. Let's see how this goes.

Of course, the book would not be as readable were it not for Linda Crippes, my editor and writing advisor, who has taken many paragraphs to something that's enjoyable to read. We've worked together for so long that, at this point, I'm not sure when we started. Just glad to have somebody to assist in that department since my brain works better in speaking mode than when I write.

I also want to thank the reviewers, including Jo Peterson, who provided some great feedback on this "very different type of book." She provided many of the better ideas as we break new ground here. It's always good to have a team of smart people who make sure your ideas are not too crazy.

Keep in mind that my opinions here, which is most of the book, are mine and mine alone and not necessarily that of my current, past, or future employer. That said, this book was carefully written as not to create conflicts of interest, or cause concerns with any independence issues.

Finally, I want to thank my wonderful clients, who provided me with the experiences required to write a book like this. I'm in business to make them successful, and I feel privileged to say their successes reflect my own.

We Want to Hear from You!

As the reader of this book, *you* are our most important critic and commentator. We value your opinion and want to know what we're doing right, what we could do better, what areas you'd like to see us publish in, and any other words of wisdom you're willing to pass our way.

We welcome your comments. You can email or write to let us know what you did or didn't like about this book—as well as what we can do to make our books better.

Please note that we cannot help you with technical problems related to the topic of this book.

When you write, please be sure to include this book's title and author as well as your name and email address. We will carefully review your comments and share them with the author and editors who worked on the book.

Email: feedback@community@informit.com

Reader Services

Register your copy of *An Insider's Guide to Cloud Computing* at www.pearsonitcertification.com for convenient access to downloads, updates, and corrections as they become available. To start the registration process, go to www.pearsonitcertification.com/register and log in or create an account.* Enter the product ISBN 9780137935697 and click Submit. When the process is complete, you will find any available bonus content under Registered Products.

*Be sure to check the box indicating that you would like to hear from us to receive exclusive discounts on future editions of this product.

Preface

This book blows open the cloud computing industry's secret doors and finally says the quiet parts out loud.

Many of us know about or have experienced cloud computing outcomes that did not live up to expectations. Some outcomes were outright cloud failures that made no sense. Learning through trial and error is an expensive way to solve the problems.

There are thousands of secrets in the cloud computing industry. If revealed, these secrets could help more enterprises succeed with cloud projects their first time and avoid expensive do-overs or the ongoing expenses of a cloud system that "more or less" works, but not really and not ideally.

Cloud technology providers spend billions in marketing dollars to keep you unaware of certain secrets. Wouldn't it be nice to know cost-optimized ways to leverage cloud computing services that can cut your cloud bill in half? Or, how to leverage cloud service providers using multicloud and hybrid cloud configurations that allow you to shop in a larger pool of technology and get the best price for best-of-breed technology?

Don't feel singled out or alone. The current cloud skills shortage almost guarantees you will have a blind spot or experience gap in some or many sections of your cloud project. Providers realize there is a certain lack of knowledge within most enterprises. They also recognize that there are two general types of organizations that leverage or are about to leverage cloud.

The first type spends millions more than required on cloud computing solutions, largely due to a lack of staff familiarity with the minutiae of the cloud computing industry; thus, the enterprise remains oblivious to the pitfalls. It's not in a provider's best interest to educate away their profits.

The second type comprises a smaller minority: those enterprises that are in-the-know and understand what others don't. This includes knowledge of long-held cloud secrets that were once whispered about in conference rooms or spilled over drinks at cloud conference happy hours. They might even know more than their providers about certain aspects of cloud computing.

This book reveals many of those secrets, as well as the secrets that swirl around the hidden values of emerging technologies such as artificial intelligence, containers, no-code, and serverless computing. There is information about what works, what doesn't, and how to pick your best resources for migration or net-new cloud-based application development.

Finally, and most important, this book reveals why some workloads and data sets don't belong in the cloud…for now. We also review the true value of cloud computing in general.

Other secrets will spill, such as the actual value of cloud computing when it's applied to your carbon footprint, and the folly of some cloud technologies that were hyped just a few years ago that are now worthless and should be avoided. Also included is a discussion of "game changer" technologies that have small marketing budgets and should be examined more closely.

Regardless of how much you think you know, it's always best to start (or continue) a project with the most relevant information. Some of the secrets revealed in this book will change your odds of success with cloud computing by a little or by a lot…often by a lot.

Why I Wrote This Book

For this book to have any value, it must provide information in a candid way and not hold back on issues that many editors and book publishers would find a bit risky. For example, discussing the real value of cloud computing regardless of how the larger cloud providers now define it. Or, looking at the claims of sustainability in the cloud computing and technology markets, and letting you know the current expectations and core benefits with comparisons to what we're seeing in real-world projects.

This book's primary purpose is to provide the positive realities of what cloud technology can bring to today's enterprises and to reveal some often-unexpected downsides. The surprises tend to happen when organizations approach cloud computing incorrectly, or when the public cloud providers themselves don't have a good handle on how their clients should leverage their technology.

The misuse of and misinformation about cloud computing technology are the two most common problems. Yes, the technology does have value, but what you use and how it's used determine its value. Every day we learn more about what works, what doesn't, and how to make sure cloud computing works best for an enterprise. Cloud providers, technology providers, and those charged with selecting and configuring this technology must work together. It's also good to know if and where your providers themselves might still have some blind spots.

What struck me about the cloud computing market in general is that the billions of dollars spent on cloud marketing seems to spin cloud technology as something that can't fail. Marketing has a big impact on how organizations leverage the technology. It can also obscure some important information if potential clients don't know the right questions to ask prior to selecting cloud providers or cloud technology. And yet, how can you get the answers if you don't know the questions to ask?

This lack of knowledge leads to a few probable outcomes:

First, and the most helpful thing that could occur, would be failure. Although nobody likes to fail, at least we understand that what we did was wrong, and we back up to try a different approach with another cloud technology selection and configuration. Although this effort costs time and money, if you apply the lessons learned, the movement to the correct, near-optimized solution will be a win.

Second, and the most negative of these outcomes, is to select a cloud technology solution that's under-optimized. It costs way more than it should and does not bring the ultimate business value back to the enterprise. The issue here is that the solution works, which is the only metric that many cloud architects and developers use to measure their success. It doesn't seem to matter that it costs the enterprise up to a half a billion dollars in lost revenue. Who has the knowledge to consider the cost savings of a near-optimized configuration and the business value that ultimately gets left on the table with an under-optimized system? I would strongly question the opinions of the cloud architects and developers who installed and still believe that their underoptimized system "works."

More often, underoptimized situations go undetected. The bad solution continues to hurt the business ongoing. Chances are that you have an underoptimized example in your own career. If you don't, reading this book will help keep you in that rare air.

The outcome of trial and error is success. A success on the first try requires selection of the right technology with a configuration that gets as close as possible to full optimization, in terms of cost efficiency and value that's returned to the business. Luck doesn't get you there. It's a matter of being open-minded from the very beginning of a cloud project. What is most likely to work best for a particular project, given the requirements and current state of the technology? Look beyond the hype and noise to see what the technology can do as well as what it can't do.

Cloud migration and/or development projects are not considered fully successful unless they approach optimized efficiency and return optimal value to the business. That criterion makes fully successful projects rare. Not because they are hard to do, but because of existing biases, and a lack of related experiences and skills. In other words, few cloud architects and developers can see beyond the hype, misinformation, and what's popular to make potentially unpopular decisions, even if they're the right ones.

How to Read This Book

If the chapter topics seem to jump around a bit, that's on purpose. So many secrets exist around the use of cloud technology that they can't be organized into neat categories. This book organizes what I think is important to understand about the cloud computing industry into chapters that obviously go together. These chapters include knowledge and information that we see lead to successful cloud projects.

Other chapters are decoupled from the theme of the previous chapter or the ones that follow, but they build on information introduced in previous chapters and present new information on an obscured or "secret" topic. These chapters include secrets of cloud computing that most people in the industry are not willing to share.

Thus, although we talk about cloud storage and cloud computing services in the first few chapters, which are both considered foundational infrastructure services, we move quickly to more advanced cloud services such as artificial intelligence and machine learning that seem to dominate today's technology press. Then, we talk about other trends such as multicloud, and how enterprises are failing and succeeding with those technology configurations. We include what's being hyped and what works, which are two very different things.

Next, we get into a few cloud topics that seem to be making technology press headlines: cloud's ability to support sustainability and to reduce our carbon footprint and more. Finally, we discuss the likely future of cloud computing. Here we focus on what's most likely to occur versus what is predicted.

We end with a discussion of skills: how to find them and how to build your own. To build net-new cloud-based systems, it's critical that project leaders know how and where to build new skill sets into the enterprise, as well as how to find, attract, and keep the right skills. We also look at how to manage your own skills to become someone with talents that enterprises will pay top dollar to attract and/or retain.

What Benefits Can You Expect?

The core benefit to reading this book is that you'll be "in the know" about things that most of the cloud providers, technology vendors, consultants, and other players would rather you not know. Armed with this information, you'll be able to ask specific questions around the use and cost of the technology under consideration and many other factors to question that you'll learn about here.

To use a very simple example, let's say a new business will soon launch and you're in charge of finding a system to manage inventory. You want to determine the specific type of cloud storage that will best support a new inventory system for the enterprise's possibly unique type of inventory. You need to consider different types of storage, such as object, block, and file storage, and understand the implications of how each operate as well as the different price points of each. Moreover, you need to understand how particulars such as deals for goods can be obtained when the enterprise buys ahead of need, and when this will likely work to the enterprise's benefit and when it will not.

There will also be opportunities and challenges with cloud heterogeneity. Multicloud provides you with the ability to select best-of-breed technology and services. However, the price of this choice is complexity. The number of different system types and brands that you must operate over the years to come will require different skill sets and interfaces.

The complexity of it all becomes the challenge. Always keep an eye on the number of moving parts and configurations required to become successful. Determining the best balance of choice and complexity for each project, as well as for the enterprise, will result in a near-optimized system that brings the most value to the business.

Information presented in this book will not tell you exactly what to do step by step, one-size-fits-all. It *will* arm you with the right knowledge to make your ventures into the world of cloud computing more likely to succeed.

This page intentionally left blank

Chapter 3

The Realities and Opportunities of Cloud-Based Compute Services That Your Cloud Provider Will Not Tell You About

There are no secrets to success. It is the result of preparation, hard work, and learning from failure.

— Colin Powell

Now that we've covered storage, let's cover compute. This topic is a bit more complex. In addition to choosing public cloud computing processors, which are usually referred to as central processing units (CPUs), we must pick the platforms needed to operate those processors. The platforms themselves require choices, including operating systems, memory configurations, network connections, and even the need to define some physical storage that needs to exist.

In this chapter, we focus on what these services are, how to understand them, how to pick them, and how to obtain the best value from your cloud provider. We also look at some concepts you need to understand before you pick compute services. These concepts include how cloud hosting works, and how to understand your own requirements so you don't under- or overbuy. Additionally, we discuss how to find the best deals with known discount approaches, such as leveraging reserved instances or leveraging processors that are considered "off-brand." Finally, we review some traditional options that we often forget about in the haze of cloud computing migrations. Again, the focus is to provide you with inside information on what to leverage and how to think pragmatically about cloud-based compute.

The Trade-offs of Multitenancy

When you leverage CPUs via a cloud service, you typically share those CPUs with other cloud computing tenants. You get the services that you contract for, but you're not the only client leveraging that physical resource. You could even be multiplexed across many virtual compute servers and be serviced by several CPUs, memory configurations, and input/output management subsystems.

The problem is that you really don't know what goes on behind the scenes in terms of how a public cloud provider deals with its tenancy. Providers debated extensively about multitenancy approaches when cloud computing first started. Multitenancy choices in those days were—and still are—"share all," "share some," and "share nothing" approaches.

My first cloud consulting firm focused on creating all three types of multitenancy architectures for new and existing software companies looking to become SaaS cloud providers. They needed the ability to deliver multiuser types of services, as well as the ability to deal with resource sharing that includes CPUs, memory, storage, and so on.

It was not unusual during the early days of cloud computing to have traditional software companies approach multitenancy in rather unorthodox ways—especially when they received requests from new or existing clients that wanted to be on a SaaS or other type of cloud. Rather than leverage a true multitenant automated platform, many traditional software providers simply bolted a new server onto the rack of their so-called cloud provider services. They would assign a name and IP, and the client leveraged only that server while leveraging the provider's "cloud." This is a classic example of a traditional service disguised to look like a cloud service.

Today, most of us recognize this "bolt-on" model as silly and wasteful. At the time, most clients never understood this was happening. While it's not a true multitenant architecture, it was an approach that many used to break into cloud computing, and it formed many cloud users' understanding of a "share nothing" approach.

Although we could pick apart multitenant approaches as we did during the early years of cloud computing, they worked for the most part, and the issues that arose are long since resolved. Most public cloud providers now offer solid multitenant services that optimize performance, even though they multiplex your use of a public cloud across many different physical resources. There are still some multitenant trade-offs to consider:

- Performance can be "bursty." In many cases this is the bursty nature of most Internet connections, in that performance will speed up and slow down based on the demand on the network resources. You can see the same type of performance patterns by looking at the performance of a cloud platform. When asked to do processor-intensive tasks, certainly when there's not enough memory and CPU power allocated, you'll see performance that seems as if it's starting and stopping.
- It's difficult to predict the performance of deeper types of platform use, such as threading. A thread is a flow of execution of tasks found in processes, also called a thread of execution or thread of control. Most advanced operating systems support threading. Here you will see many

of the same issues occur as previously discussed, including bursty performance. Also, launched threads may not succeed if the processor was busy for some reason, or if the application expected the thread to return faster than it did.

- Many applications that are ported to the clouds using a lift-and-shift approach where a minimum amount of redevelopment occurs. Because the application was not developed and tested on a multitenant platform, its performance can be unpredictable.

Of course, the severity of these trade-offs depends on the cloud provider you use and how the provider specifically approaches multitenancy. These issues need to be worked out beforehand, worked around, or properly fixed after the migration. Many of these trade-offs result in more migration costs for enterprises.

The Realities of Resource Sharing

When it comes to understanding multitenancy, it's important to understand that all public cloud providers have a different approach to tenancy. What one public cloud provider says may not apply to other providers—even if they say basically the same thing. Public cloud providers consider their approach to multitenancy proprietary to their IP. Although you get some overviews around how a provider approaches tenancy, what occurs in the background or in the native public cloud system that you can't see is really what determines how the provider carries out multitenancy.

In some cases, you can insist that the public cloud provider reveal how multitenancy is carried out, typically around compliance audits that you need to support. For example, say you're a government contractor who plans to place government systems on a public cloud provider. These systems may, as part of an audit, insist that the contractor and provider reveal the mechanisms behind multitenancy for a specific cloud they will leverage. You'll find that some cloud providers are candid, having had to address this issue before. Others are not as forthcoming. If the provider won't reveal the specific details and mechanisms behind how it approaches multitenancy, and if this won't meet the needs of the audit, you need to determine that up front or face having to move off that cloud because it was later determined to be noncompliant with an audit.

The moral of this story is that you determine these details up front. Understand what the requirements are for your specific situation and if you need to have a detailed understanding of how the multitenant system manages resources. Figure that out before moving assets to that cloud.

The good news is that, for most use cases, high-level explanations will suffice. In the cases with special audit requirements, it's usually best to leave these systems in the enterprise data center and save yourself the hassle.

Putting the legal issues aside for now, it's helpful to understand the basic approaches and mechanisms that public cloud providers leverage for their public clouds, especially compute. These include

- **Shared** is often the default and thus the approach that most cloud customers use, no matter whether they understand the concept or not. Shared means several public cloud computing accounts may share the same physical hardware. For most uses, this type of multitenancy

approach is fine, and you won't even know you're sharing resources. However, performance issues such as the trade-offs we've described previously will come into play if you have special high-end requirements. As you may have guessed, a Shared model is typically the least expensive because you use fewer physical resources, all of which are shared. Figure 3-1 shows what the high-level architecture looks like, where a single physical resource is shared by many tenants. Note: The way you share resources—and how they are shared—is specific to the cloud provider's multitenant services.

- **Dedicated Instance** is when your instance (that is, running your applications) runs on single-tenant hardware. This approach typically works better for special higher-end requirements, such as applications that tend to saturate the CPUs and other resources. For all practical purposes, it's as if you host your application on your own server that nobody else uses. Of course, this approach is more costly than the Shared model because you leverage physical hardware that others cannot leverage. Also, you rarely know the physical location of hardware. The provider allocates different servers for your use, and you are typically not assigned a dedicated server.
- **Dedicated Host** is when your instance runs on a physical server with all its capacity dedicated to your use. You have complete control over this isolated server. Functionally, it's as if you purchased a server, installed it within your data center, and now have exclusive use of it. In many instances, the cloud provider may even let you know where it's physically located, although those policies differ from cloud provider to cloud provider. This is the costliest of all the sharing options listed here.

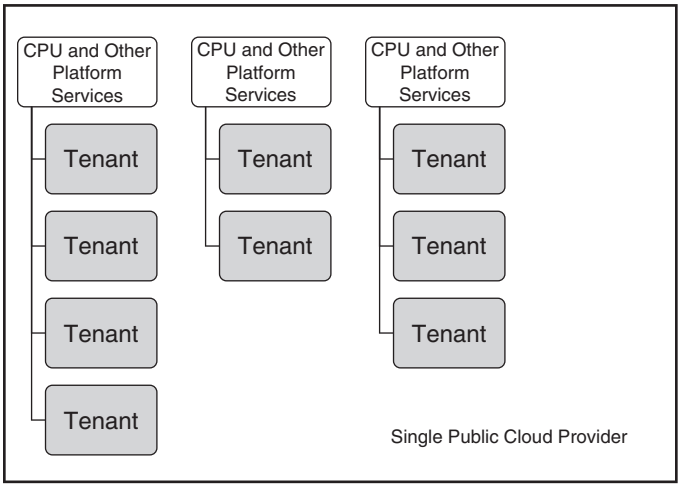

FIGURE 3-1 Resource sharing for most public clouds means that you share physical resources with many tenants or companies like yourself that need to access compute services. This is often the least expensive approach because the cloud provider can share a few resources with many tenants. As far as the tenants are concerned, they leverage a virtual resource that functions much like a dedicated server.

Costs Versus Consistency

The trade-off with compute models comes down to cost versus consistency. You either pay less to leverage physical servers that are shared with others, or you pay more for the luxury of dedicated hardware to ensure greater consistency. Figure 3-2 depicts the typical cost curve that most enterprises will encounter when purchasing compute resources from a large public cloud provider. None of this should be shocking, considering that as public cloud providers' costs go up, they will pass the cost for the additional hardware resources, such as storage and compute, on to their customers.

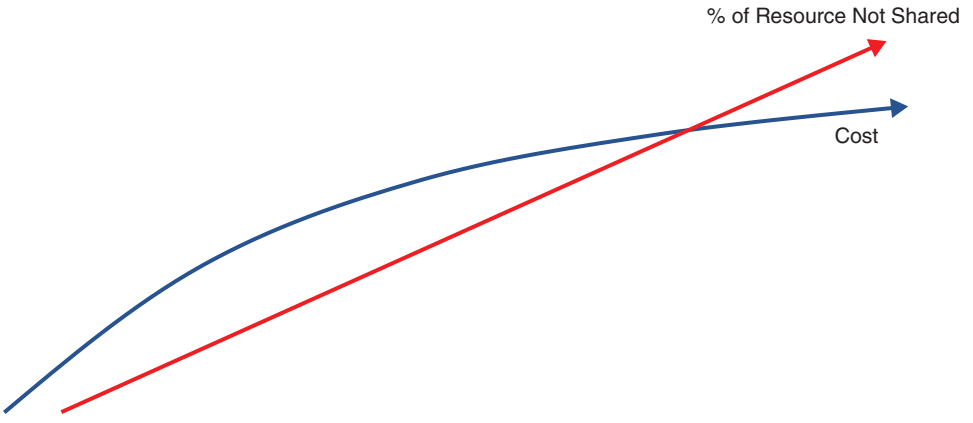

FIGURE 3-2 As a rule, costs rise as you share fewer resources in the cloud. However, at some point, the cost begins to level out but never becomes completely flat, or diminishing.

The real question then becomes, What would compel you to use nonshared resources, and how would it even justify the extra costs? In truth, many of those who insist on dedicated instances, and even those who reserve a physical server in a public cloud, typically have no need for them. The extra cost is a classic waste of money.

In many instances, customers pay much more to use a public cloud provider's dedicated resources than if they owned their own physical server, even with data center rental and maintenance factored into the equation.

In too many instances, I see this become a kind of ego play, in that someone wants to elevate the importance of their workloads to a point where they lose sight of the real goal, which is to get to the true value of cloud computing. The objective is to pay less to securely share most resources and still gain the soft values of cloud computing.

There are a few instances when it's the right move to elevate workloads and data, when the needs for consistency and control outweigh the cost factors. However, most arguments for uninterrupted performance or security concerns with shared resources prove unfounded or, more often, available within a shared model option. We're well into our 15th year of leveraging IaaS clouds and pretty much understand what shared and nonshared resources can do, and the cost trade-offs.

Despite the history and provable facts, I still see enterprises demand and deploy nonshared resources for the consistency and control reasons just mentioned. This will lead to a sizable reduction in the value that cloud computing brings to the business. In many instances, it will then lead to cloud computing having negative value that is completely avoidable. Carefully consider the business case and the technical realities before deciding that your workloads are much too important to mingle with others.

Cross-Partition and Cross-Tenant Hacks

It was not long after cloud computing became a thing that alarms began to sound around the possibility that other tenants could reach into your compute instance and grab your data or interrupt your processing. This is called a *cross-channel attack,* a *cross-partition attack,* or a *cross-tenant attack.* For our purposes, we'll use the terms interchangeably.

These "cross" concerns came from some valid realities, including the fact that you are indeed running on the same physical servers. It's logical to assume that if something were left misconfigured or a vulnerability overlooked that could be exploited, then someone or some program could purposefully reach through a logical partition and cause problems.

Experiments proved that this scenario could potentially occur, but only if there were a vulnerability on the multitenant system that was known to black hats, and thus could subsequently be exploited. The chances of that occurring, even though there was a chance, were much less than other security risks that typical applications and/or data sets encounter on a traditional platform. So, by moving to cloud, you lower your security risk. The chances that you'll be hacked tenant-to-tenant are pretty much nonexistent.

The real issue here is that multitenant just seems scary. Your application that processes sensitive, often proprietary data, runs within the same memory set and CPU as other companies' applications and data. Your cotenants could be a competitor, an illegal business, or the government.

Figure 3-3 depicts the types of attacks that tenants find of most concern, namely, a tenant being able to reach into the workspace of another tenant running on the same physical server. Or a tenant reaching across to another tenant running on another physical compute server.

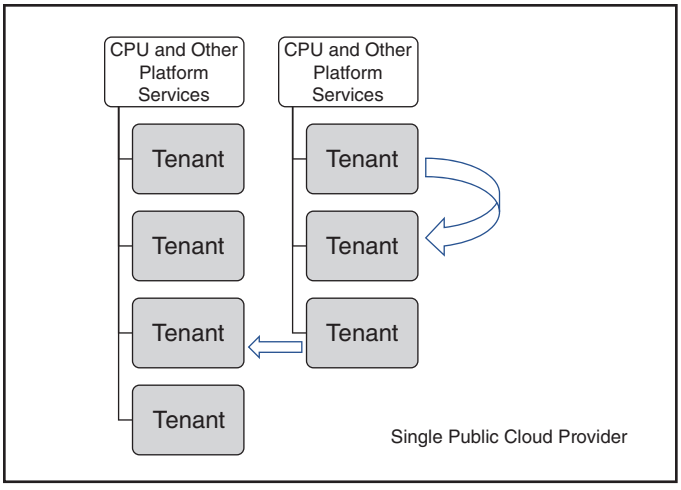

FIGURE 3-3 In the early days of cloud computing, many feared cross-channel, cross-tenant, and cross-partition hacks, or reaching to compute and storage spaces controlled by another tenant—either within the same physical CPU and platform services, or between physical CPUs and other platform services. Much of this concern has been eliminated, but it's still wise to ask your public cloud provider how this will be managed.

Cross-channel attacks are a false concern for the following reasons:

- The public cloud providers became aware of this concern years ago and built specific security measures into their systems to isolate resources used by specific tenants. Of course, providers all do this differently, but they all have their own mechanisms to detect any cross-channel intrusions.
- The data that resides within each tenant partition is encrypted at rest and in flight. If, for some reason, a tenant could reach into another partition, the data is worthless given that it's encrypted. Most public cloud providers encrypt everything to remove this security vulnerability. You maintain your private key, and even the cloud provider can't see your data without it.
- The nail in the cross-channel risk coffin is that tenants can't launch a machine instance on a specific machine, say, a specific machine where their target runs their data. Lacking the ability to choose the physical server that the tenant will run on, at least in a shared multitenant deployment, means no tenant can leverage standard approaches to access other tenant instances, such as making use of Level 3 memory cache exploitations.

Nothing I say here pushes these possibilities off as a nonthreat, only that there are many other things that should rank higher on your list of concerns in terms of cloud security. In this instance, it's okay to say, "Nothing to see here."

Keep in mind that when you play the cloud security game, it's not about creating an invulnerable security layer. Ultimately, nothing is invulnerable. It's about removing as much risk as you can, and then prioritizing and paying attention to the more likely threats.

CPU Performance, Meet the Internet

Most of those who leverage cloud computing do so using public cloud providers. Although some companies can afford dedicated connections into a cloud provider, most of us will leverage cloud services over the open Internet. That causes a few problems when it comes to CPU performance, as well as data consumption and transmission.

As you can see in Figure 3-4, applications that are network-bound cause an issue. This means that the applications leverage the Internet to transmit and receive inter-process communications (IPCs) to share data and application message traffic between applications. This does not affect applications that are designed not to carry out IPCs and data exchange over the open Internet and may be sharing data only via the much faster network that exists internal to a cloud provider. By avoiding use of the Internet and its bursty latency that often occurs, your application won't be bound to the speed of the Internet when considering overall performance.

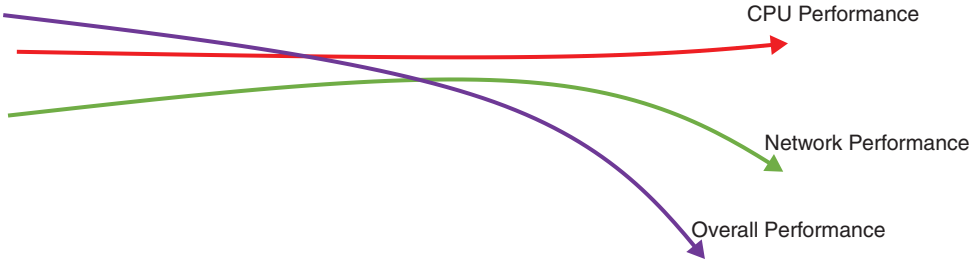

FIGURE 3-4 Although you might leverage the fastest CPUs, if you move data or inter-process communications over the open Internet, the speed of network communications will become the bottleneck. This is something that many using the public cloud attempt, but often redesign or rehost the applications and data back onsite, so that network latency is diminished.

Yes, most of us are aware of performance issues when you carry out more heavy-duty processing and data communications over the Internet. What we often overlook are changes to the criteria you should use when selecting compute platforms, including CPU speed and type, memory size, and even the operating systems you plan to leverage, such as Linux and Windows NT.

Figure 3-4 shows performance at various levels of network performance, or an increase and a decrease in bandwidth, with a significant drop-off at the end. Note that the processor speed is about the same, and even rises slightly as it moves to the right. However, as network performance decreases, it really does not matter which processor you picked because that's not what determines overall performance.

For example, let's say you have a saturated Internet router or even a denial-of-service attack. If the application or applications are bound to Internet speeds due to dependencies on IPCs or data exchanged that's needed to drive the application(s), the CPU speed and performance become irrelevant.

This is not a call to rewrite all your migrated applications to remove or reduce IPCs or data exchanged over the Internet, or communications that once existed only on your corporate network. Instead, consider the cost you'll pay for the platform, including CPU, memory, and so on.

The moral of this story: If you don't get the extra benefit of faster CPU processors, why spend the extra money?

The Slowest Components Determine Performance

Here's another way to think about buying compute. If your applications are processor bound, meaning they consistently wait for a process to complete or finish a compute cycle to continue, then you'll likely get real value out of deploying the fastest and most expensive CPUs. This includes high-performance computing (HPC) platforms that are now available, which clients often leverage in response to slow-running applications.

However, for many applications, the CPU, memory size, and speed have little to do with overall performance. It's the application design that's at fault; trying to fix the problem on the provider side just wastes money. Sometimes it costs tens of thousands of dollars extra per month to run network-bottlenecked applications on expensive CPUs and HPCs, which removes any value gained by using public clouds.

Many enterprises go wrong here when they pick cloud computing services and components, which often includes the compute platform. With an incomplete understanding of what determines overall application performance, the typical response is to upgrade and up-spend the cloud provider's CPUs. After all, it's a simple click of a mouse to do the upgrade; you don't have to visit a data center to integrate a new compute server with the other physical servers and the network. The process is so easy that you can get into real trouble with cloud spending before you realize the root causes of application performance problems.

A detailed discussion of typical application performance problems, how they are diagnosed, and how to design an application for good performance is beyond the scope of this book. However, it's an area of focus that many cloud computing and cloud application architects should better understand.

Remember: The slowest running component determines overall performance. The most frequent culprit is the network. However, storage I/O delays, omnibus latency, and yes, the CPU could also be factors.

This point brings up more complexities when you pick components to build your application(s) or system. It's not about what you spend for better-performing cloud services. It's more about the design of your application(s) or system and where the performance bottlenecks will likely exist. The kneejerk reaction to a poorly performing application is to toss money at the problem and hope that fixes it. Instead, that approach introduces a whole new set of problems. The result is that you'll spend too much for the application's infrastructure that will not solve the root cause of the performance issue.

For example, let's say a client of mine migrated an inventory control application from an existing LAMP stack (Linux, Apache, MySQL, PHP/Perl/Python) that ran in their data center to a public cloud provider. Minimum changes were made to the application, choosing instead to take a lift-and-shift approach to minimize the costs of migration.

After the migration was complete, performance issues were noted during acceptance testing. The user interface ran 40 percent slower than it ran on the traditional platform. Without truly diagnosing the problem, the application owners decided to leverage a more powerful and more costly platform, meaning a higher-end CPU cluster and more memory. The result was a 5 percent increase in performance, with the resulting performance determined to be unacceptable.

In this example, it could be any number of components or cloud services that caused the performance problem or problems. Without a sound diagnosis of what's at the root of the performance issues, you're just taking shots in the dark by renting better components and cloud services and hoping for the best.

Upon detailed diagnoses, it was determined that a combination of the database and network was the root cause of the problems. The database was fixed by changing a few tunable parameters, in this case, significantly increasing the data cache size. A failing Internet router in the department that used that application the most caused the network issue. Spending more money on CPU and memory resources did not help and only confused the matter more. By reviewing the application's overall design and usage, we identified the actual issues and fixed them for a minimal amount of money.

How to Speed Things Up Through Design

The lesson here is that the key to selecting the best compute configuration (covered next) is to first understand the design of the application that will leverage the compute instance. Many of us hate the "it depends" answer, but here it really depends on how the application was structured, and thus how it leverages compute, storage, memory, and the network.

Fortunately, you can run the application within an application profiler to understand how the application leverages infrastructure resources, such as I/O, storage, compute, memory, and network. In many instances, this is done prior to migrating the application to the cloud so that you can make a more educated determination of how to set up the target cloud's infrastructure to better support the application and data storage.

You can also understand the design of the application in other ways. A good old-fashioned review of programming code and database structure comes to mind. A review of the documentation left behind from the original design is another sound idea, as is talking to those who originally designed and built the application. During speedy migration projects, you'll find that most of those who migrate the application don't go to this level of due diligence and end up having performance problems. Reminder: Tossing more resources at the application after the fact is the costliest way to bandage over performance problems.

Enterprises consistently under- or overestimate the amount and configuration of cloud-based resources needed for a single application, or many applications. This is more than an application-level problem. It's now a holistic migration problem.

The key to understanding the resulting performance problems is to first understand the design of each application, how it leverages different resources, and thus how the target cloud compute instances should be configured, along with other services that the application may need, such as storage.

Application design needs to be considered when building net-new cloud applications or when migrating an existing application to the cloud. Almost all performance issues I encounter, in terms of applications being migrated to the cloud or built on the cloud, end up being issues that were fixed by changing the application's design.

Examples include leveraging new models for utilizing memory more efficiently, reducing calls across the network, and even performing foundational tasks such as leveraging a database caching system to reduce disk I/O and network utilization. Some of these are tweaks, such as tuning your database. Others are major surgery, such as leveraging a new and more efficient sorting approach. Potential design patterns pretty much number in the millions. It's important that you have some visibility into these patterns to make the most of your cloud deployments.

Picking the Most Optimized Compute Configuration

So, let's say we do most things right, in terms of understanding the design of our application workloads and data storage requirements. That means we pretty much know what we need for CPU, memory, and other platform requirements such as operating systems. How do we pick the most optimized compute configuration? Keep in mind that we can make mistakes here in two different directions.

As you can see in Figure 3-5, the value delivered is at its lowest points when we leverage too few resources or too many resources. If we leverage too few resources, this will save money on cloud resource usage, but application performance and resiliency will suffer, leading to a reduction in the business value the application will deliver to the business.

It's the same case for using too many resources. Although application performance and resiliency should be good, we pay too much for unnecessary resources. Thus, the extra cost reduces the value delivered to the business by overspending. We maximize the value delivered to the business when the value delivered to the business is about centered where both curves meet, and the number of resources is as fully optimized as value delivered.

Keep in mind that we may be facing a mindset issue here. Customers build in the cloud like they used to build data centers. They build like they are retailers trying to build for the Christmas rush. This is largely because it's so easy. The best analogy that I have is the current rise of food delivery services. The ease in which we can obtain our favorite foods, with pretty much no effort or pain, means that we will have the mindset to buy more, and thus get fat (or fatter). The mindset around cloud computing means that we're getting fat with cloud services that we most likely don't need.

What's interesting about this chart is that the current number of fully optimized enterprise cloud applications is pretty much zero. Most of those charged with picking cloud resources, including the CPU and memory resources, over- and underestimate the number of required resources. They end up on either end of the resource and value curves and almost never near the center. One problem is that few cloud architects don't suffer the immediate consequences of resource allocation mistakes and remain blissfully unaware of the ripple effect of their choices.

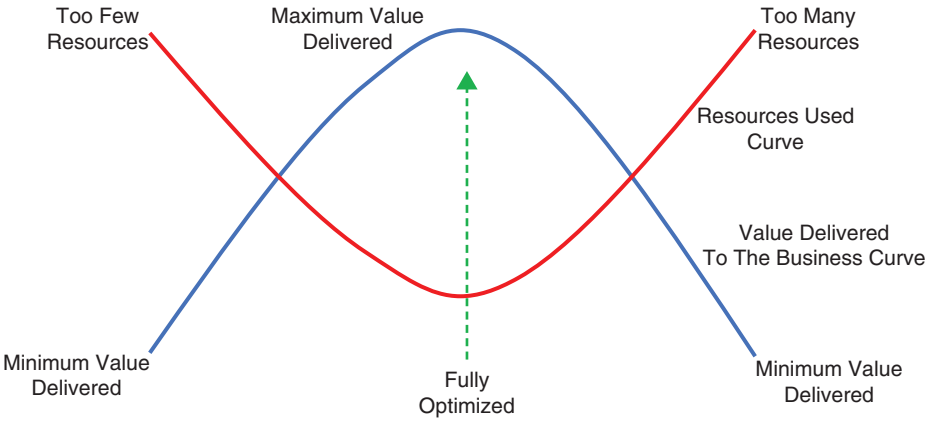

FIGURE 3-5 Here's what happens when you leverage too few or too many resources, and the corresponding effect on value delivered to the business. The objective here is a fully optimized system that delivers the most value to the business. You will find that just tossing money at problems fails to deliver the value. Also, not spending enough on the resources that you need also removes value. It's a balancing act.

Although overallocated resources result in higher cloud bills, the cloud-based application typically performs just fine, which is the metric most application owners use. Even those who underallocate resources may not know they have a value delivery problem unless they measure productivity, which may suffer due to application latency and outages. It will take time to go back and identify where a system failed to live up to expectations, but these ghosts in the machines will eventually come back to haunt you.

Again, the ability to optimize systems comes down to understanding your requirements before selecting the cloud resources you should leverage. The process should never be about guessing, nor trial and error. It should be about mathematically understanding the requirements around processor speed and memory use to get as close as you can to full optimization. This is a bit like horseshoes and darts in that you'll win this game only if you get close. Very few will obtain full optimization when it comes to business value; what's important is that you get as close as possible.

Figure 3-6 looks at your options when it comes to selecting a cloud compute platform. Selecting and configuring a compute instance sound simple; just select the CPU type (such as x86), including brand (Intel, AMD, and so on) and processor speed. However, you must also consider the number of processors configured and the size and speed of the memory. And then there are different types of processors, such as a microprocessor, microcontroller, embedded processor, and digital signal processor. Also, different processor generations, brands, and standards.

Yes, you can configure and deploy some pretty powerful compute platforms. However, it's not only about what you pick to align the platform to the requirements of the application, but what goes on behind those choices, in terms of power and cost of the compute resource.

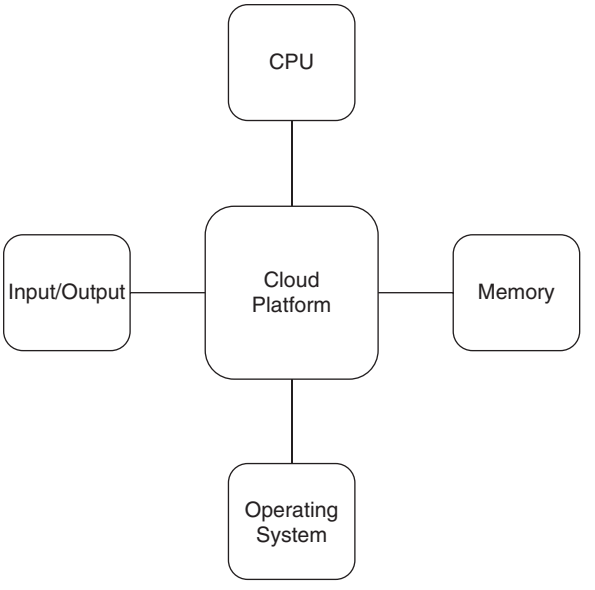

FIGURE 3-6 When picking a compute platform, you must decide the power and type of processor, number of processors, memory configuration and size, and operating system. Input/output usually needs to be configured as well.

The operating system the platform will employ focuses on the type and brand. For example, if your application runs on Linux on its traditional platform, you'll need to pick a Linux-compatible brand for cloud migration such as Red Hat, or perhaps a version created by the cloud provider such as AWS Linux. You might also pick Windows NT and other operating systems that you need to configure.

You'll find that the types, brands, and even the versions of operating systems vary a great deal from public cloud provider to public cloud provider. It would be nice to have a list here by platform. However, there could be a dozen more operating system types, brands, and configurations available by the time of publication. But I've found that most of them do the same things, and today arguing about what operating system is best has diminished returns. Don't get caught up in the silliness of becoming a believer around one specific operating system or another. Certainly, not the public clouds where changing operating systems takes a very short period of time and does not require touching physical hardware.

Normally, cloud providers give you a few choices. First, you can select one of their prebuilt configurations with the processor, memory, and operating system preconfigured for you. These "packages" are often the easy choice because these components are known to work with the provider's systems, or you can select the components a la carte from a list of prebuilt configurations provided by the public cloud provider. Or, you can select just the memory, operating system, and processor(s) that you need. So, what's the best path?

I never really liked the idea of selecting the packages that the public cloud providers put together. Not that there is anything wrong with prebuilt bundles that are known to work together, but the odds are against a prebuilt bundle meeting the exact needs of your application workload. This is especially true if you're only guessing about application requirements without having a detailed profile, and little thought is given to your exact and specific needs.

Picking the correct compute configuration is an often-overlooked art form, certainly in the world of cloud computing. Let's look again at Figure 3-6, where there are three to four components to choose. You'll also need to attach storage (perhaps a database), configure the network, and even attach special services such as machine learning and data analytics. For now, let's just keep the discussion to compute.

For most migrated applications, you'll need to profile the application workload using an automated profiling tool to determine the exact processor and memory needs. Or, you can use formulas to determine the compute needs of the application based on what it's built to do and how it should use processor and memory. Guessing is only a last resort that pretty much ensures a wrong answer.

Paying Too Much for Cloud Compute? Here's Why

A common theme in this book is that most who leverage public clouds pay too much. You'll find study upon study that proves this statement correct, but what's at the heart of the issues that break cloud budgets? If we just focus on compute issues for the moment, a few issues seem to tip the scales.

As covered previously, most of those who configure and allocate cloud computing compute resources do so without a good understanding of what the application does and how it does it. When the choices are based on uneducated guesses, most of those tasked with picking a compute configuration will select more resources than the application needs. Or, the budget-conscious could pick too few resources and then end up with an application that fails as it runs out of CPU horsepower or perhaps memory. Over-buying typically won't be identified as a problem until cloud costs finally come under audit. Eighty percent of the applications I see have typically overpurchased compute resources, spending as much as three to four times the money they should have on resources that won't be fully utilized.

Another issue? The cost tracking that cloud providers make available are not set up to let you know when or if you're overspending. To their credit, most cloud providers do offer tools to determine the size of the resources that you need. However, the tools have limited value. Typically, they're just questionnaires that those about to migrate and build net-new applications will need to fill out. In many cases, the new clients don't know what to tell the questionnaire.

Many of the mistakes being made today are by IT leaders who are just learning to understand cloud compute. How eager would you be to point out your own mistakes, ones that could be costing the business millions of dollars in unnecessary cloud resources?

The solution to this problem is obvious, but one that's rarely considered due to cost concerns. The better option is to implement a complete FinOps (Financial Operations) system that includes people, processes, and tooling that can automate the ability to determine when and if you're spending more cloud dollars than needed.

A FinOps system should also suggest changes that could reduce spending without impacting application performance and reliability. A FinOps approach and tooling will pay for itself compared to the costs involved to do a migration wrong the first time and then go back and fix it later. Implementing a FinOps system can provide a larger and ongoing return on its initial investment—this considering that ongoing processes and automation allow us to monitor costs, create policies around managing costs, and optimize the use of cloud resources with costs in mind.

Here are a few more ways to ensure that your cloud compute costs are more in line:

- **Accept outside help.** Hire people who can provide you with an objective opinion as to what you're spending on compute and where it's in line with what the workloads need. Consultants often provide this service for several businesses and thus have experience in what's optimized and what's not.
- **Deploy FinOps.** This is a subject that we cover heavily in this book. Basically, it's the ability to track cloud costs by usage, manage cloud negotiations to obtain the best prices, and do cloud cost forecasts to understand what's spent now and what will be spent in the near and far future.
- **Gain a wider understanding of cloud costs.** In many instances, the focus is on a single cloud provider. Rather than just work within their walled garden, it's a good idea to understand what the other cloud providers offer in terms of costs and compute configurations. You may find that you can pay half the cost for the same configuration on one provider versus another. We cover multicloud later as it relates to the consideration of compute configurations from other providers.

Keep in mind this all goes to the benefits of cloud cost optimization. This is both an approach as well as sets of software that allow us to optimize costs using automated systems. This means that we really don't have to think about it, it's carried out through a magical process. This is good, considering that many cloud geeks, like myself, don't seem to have an active part of our brains that deal with cost issues. This protects us…well…from us. Some of these things include better forecasting, a single pane of glass for multicloud, row-level access controls, cost tracking and forecasting, through the use of historical cost data, in-depth optimization recommendations, and automated remediation.

Picking the Right Operating Systems

For many developers, operating systems are a personal choice. I remember the operating system wars in the '90s when I was attacked by one side or the other, simply for picking one operating system over another in a review article. I learned that, in many instances, picking an OS is an emotional decision rather than an objective, technical one. If you're going to get emotional, get emotional about being true to your workload requirements and minimizing costs.

With that said, all operating systems are not created equal. The operating system you pick will affect how your applications and data leverage the processor, storage, memory, and even the network and user

interface management, to cover just a few of the top attributes affected by the operating system. You'll see different performance profiles for the same basic workload on different operating systems. This includes open-source systems such as Linux, where many flavors and builds are put out by different vendors. Even the cloud providers themselves have their own versions that they sell within their cloud.

In many instances, the operating system you leverage will depend on the type, brand, and size of the processor you run. Some operating systems support only a small group of processors, typically those that support a similar architecture. This means they are not binary compatible with other processors and thus won't work with another processor. The topic of processor and operating system relationships is also beyond the scope of this book. However, it's a good idea to brush up on the subject before diving into designing specific solutions for your cloud compute configuration.

Picking the Right Memory Configurations

There are many kinds of computer memory. For our purposes, let's focus on the most important basic attributes required to select the right memory configuration for a specific workload. These attributes include size of memory, speed of memory, and type and amount of memory cache.

Memory cache provides temporary storage for frequently used instructions and data for quicker processing by the processor. The cache is an extension of a compute platform's main memory. Its use is a given in most cases. You might look for more cache to speed up processor requests that are more routine, which can be held in cache and thus are not required to be reloaded at the cost of performance. However, you can allocate too much cache that won't be used, or not enough, and that will also impact processor performance.

The speed of the memory is just that. It's the amount of time it takes memory to receive a request from the processor and then read or write data. RAM speed or frequency is measured in megahertz (MHz), and that's often how cloud providers will tell you about the speed of the memory for your compute configuration.

Finally, there's the type and size of the memory, or how much memory will be available to the application. Typically, memory size is measured in gigabytes (GBs). If, for some reason, you don't allocate enough memory, your application won't just stop processing. Instead, the operating system will begin to write to disk storage instead of writing to memory. Your application performance will suffer greatly when this occurs. Memory size is also a tunable parameter in the operating system, as to when and how the operating system leverages storage instead of actual memory.

The Concept of Reserved Instances

Did you ever get an offer to purchase something ahead of need at a good discount, something like a vacation timeshare or even a cemetery plot? Reserved instances offered by public cloud providers are the same. They are compute configurations and/or storage instances that are purchased in advance of actual need, and the provider offers a discount to pay in advance and reserve the purchases for your use.

There are a few things to discuss here, good and bad. First, let's cover the good. If you're good at planning ahead and understand exactly when and how you will need a compute or storage instance from a cloud provider, this is a good option to consider. However, based on my experiences, chances are you're overconfident about your planning abilities, most especially if this is your first cloud project. But let's assume that some of you have this superpower, and then the reserved instances may be a way to get more for less.

If you have not guessed yet, the bad around reserved instances is that most of the people planning for the use of cloud resources in the future do not have enough experience or luck to make the right choices. When they stock up on reserved instances, most overpurchase. Unlike that giant cooler the big box store took back a month after you bought it, all sales are final with cloud providers.

Figure 3-7 depicts what happens most often. Instances are purchased from the cloud provider for a predetermined cost, which is now a sunk cost. The instances typically expire after a specific amount of time, usually one to three years, and end up being largely unused. Thus, you may see a cost layout as shown in Table 3-1.

TABLE 3-1 This is an example of the business case for leveraging reserved instances: a refund.

Resources and Outcomes	Cost
10 reserved instances, x86 CPUs, 6 GB of memory, running Linux, usable for one year.	$20,000 one-time fee.
Number of instances used, 4, with the remainder returned to the cloud provider.	Considering what we paid for the reserved instances, we say that we paid $5,000 per instance.
Cost of purchasing the same instances as needed.	$3,000 per instance.
Overpaid/underpaid.	$20,000 – $12,000, or $8,000 overpaid for the same cloud resources.

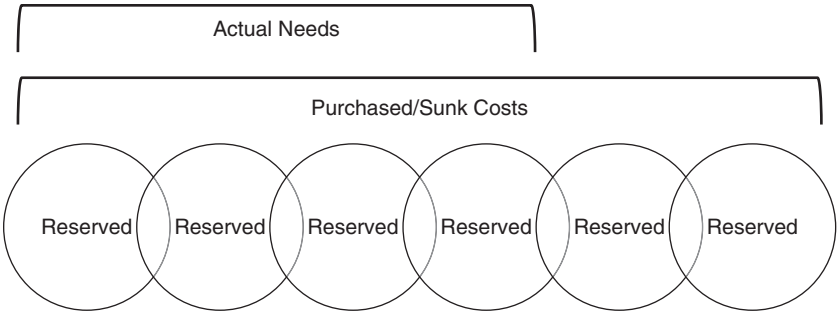

FIGURE 3-7 Although reserved instances may make sense for a small few, most overbuy and underutilize reserved instances that typically cannot be returned for a refund.

So, do reserved instances ever make sense? Sure, such as when you're relatively certain that your instance need will be X and then you use X. However, as we covered previously, most of those who do cloud resource and cost planning still don't fully understand what will be leveraged and in what amount of time.

Another scenario that would make sense is if you forecast the need for 100 compute instances over the next year. You could purchase 50 reserved instances to reduce the cost. If you fall a bit short, such as needing only 63 instances over the year, you're still ahead of the game.

In this case, you paid for 50 instances at a discount that you needed and used. With 50 instances discounted, you spent less money than if all the instances were purchased at full price as needed, even though you were off by 13 instances. Although you did not correctly predict the number of instances you planned to leverage, the strategic use of reserved instance pricing allowed you to save some money without much risk of letting the prepurchased instances go to waste. Now that more FinOps teams and tools are starting to emerge, we're seeing more enterprises take advantage of this strategy.

Enterprise Discount Program (EDP) is a cost-savings approach that most enterprises don't understand. AWS is the cloud provider known for this program, but other public cloud providers have something similar. The idea is that organizations that commit to a predetermined annual cloud spend (typically $1 million or more) can obtain automatic discounts. The discounts will vary based on spend commitment. These are functionally like reserved instances but provide more holistic cost advantages for larger enterprises that will be spending a great deal on public cloud services.

The general recommendation here is that you should avoid using reserved instances unless or until your cloud resource planning is so good and accurate that you feel it would be an advantage. For most of us, it's a gamble. Much like those subscription services you sign up for and used once or twice, they rarely deliver the value you expect.

Going Off-brand

Most people who look at cloud computing focus on the top three or four brands. After all, those are the cloud brands that most enterprises leverage, and those brands spent billions on their cloud services over the last few years to stay cutting edge and competitive. However, the big brands are not always the most optimal.

Today we can leverage reliable cloud services from any number of providers, including storage and compute services. Depending on the needs of your project, a nonmainstream brand could end up saving millions of dollars in cloud service costs over the next several years.

There also are non-cloud or less-than cloud options, such as managed services providers (MSPs). MSPs manage the services, including cloud services, for you, including major or minor brand cloud services, traditional system services, network services, or anything else that others can maintain on your behalf. Covered later, the opportunity with MSPs is to replace some cloud services with services that MSPs run while they also take care of systems that run across traditional platforms and/or any cloud platforms. Many MSPs offer built-in optimization systems, which means that they are not only managing traditional services and public cloud services on your behalf but also are able to optimize the use of these resources for you, thus reducing overall costs.

Finally, there are the opportunities and challenges around multicloud. Multicloud means you leverage heterogeneous public cloud services that allow you to deal with other services as the same. With multicloud, you can leverage many brands of storage and compute services using the best-of-breed services as well as least-cost services rather than deal with just a single cloud provider.

Leveraging Second-Tier Cloud Providers

Now that the concept of IaaS cloud computing is more than a decade old, we are beginning to see second-tier cloud providers enter the market. Of course, they only offer a fraction of the services you'll find within a major cloud provider, and they don't have as many points of presence (where the data centers are located), but a second-tier provider can offer a cost savings that makes them too compelling to ignore.

So, how much savings? Considering just compute, second-tier providers can offer the same compute configurations at prices 25–50 percent lower than those of the larger players, and you leverage the configurations in much the same manner. On one hand, they are not the premium brand, and thus you'll get some questions from those who consider the cloud brand inferior. But, if you save $10 million a year and your production workloads do not suffer, it may be worth the risk. Make sure to include testing, fully understand how you'll be billed, and do other due diligence, and the chances of success go way up.

Also, keep in mind that we now live in a cloud computing world where multicloud is now the norm. It's considered acceptable to leverage two or three major cloud brands. Add in the ability to place discount brands into your cloud services catalog to have them available for use, and this could be where the second-tier concept takes off. In this case, it's just as easy to attach a lower-cost resource such as storage and compute as it is to attach a major brand resource. Ease of implementation will push many enterprises to leverage the lower-cost resource, and perhaps send more savings to the bottom line. Also, by design, these lower-cost resources can work and play well with major brand cloud resources. Much of what is happening in the cloud world now and over the next several years will be a race to the bottom of the market.

Leveraging MSPs

MSPs differ a great deal in the services that each provides, but most will manage public cloud services for you, including provisioning, securing, and maintaining these cloud services in support of your workloads and data. They also support traditional systems, such as mainframes and traditional x85 such as LAMP-based platforms. In many instances, MSPs can be talked into managing more specialized systems such as edge computing systems and high-performance computing. MSPs often provide cloud-like services such as storage and compute that are less costly than the same services you might find within public cloud providers.

The advantage of using MSPs over traditional clouds is both cost and the ability to host several different platforms, including mainstream public clouds, and have the MSP manage those platforms. This means

you have a service running in front of your cloud service that removes you from much of the work and complexity of managing all those systems on your own. For many enterprises, working with an MSP is often a lower-cost choice when compared to the costs to manage all their compute and storage services. With hundreds of MSPs in the market today, this option will continue to grow in scope.

Multicloud by Necessity

We'll address the important topic of multicloud later in the book, so we won't get too deep into it here. However, when discussing the ability to leverage different cloud brands and cloud types to save costs, remember that multicloud is a core weapon to leverage to both reduce costs and focus on the use of best-of-breed cloud resources.

Multicloud by necessity means that we consider multicloud a tool to leverage different services to provide a choice that should lead to utilizing services that are more cost effective. Multicloud provides the ability to pick the exact right services your applications and systems need, or best-of-breed, as well as pick the services that are at a lower cost point. To achieve your optimization objectives, multicloud is usually a necessity.

Call to Action

Compute is a fundamental building block of cloud-based systems. I revealed some secrets here that you won't often hear from cloud providers or other sources. It's not because they don't want you to know the truth, but that the truth around using the cloud compute resource is still misunderstood on several levels. This is certainly the case when framed within the more holistic concept of building systems in clouds.

In this chapter, we focused on what these cloud computing services are, how to understand them, how to pick them, and how to obtain the best value from your cloud provider. The idea here is to learn the tricks and get the insider path on the current cloud reality. You should now understand compute and its related services in ways and with methods that can lower your costs of leveraging cloud compute instances and increase your productivity. Cost and quantity are often at odds, but they need not be.

This page intentionally left blank

Index

A

B

C

D

E

F

G

H

I

J-K

L

M

N

O

P

R

S

T

U

V

W-X-Y-Z