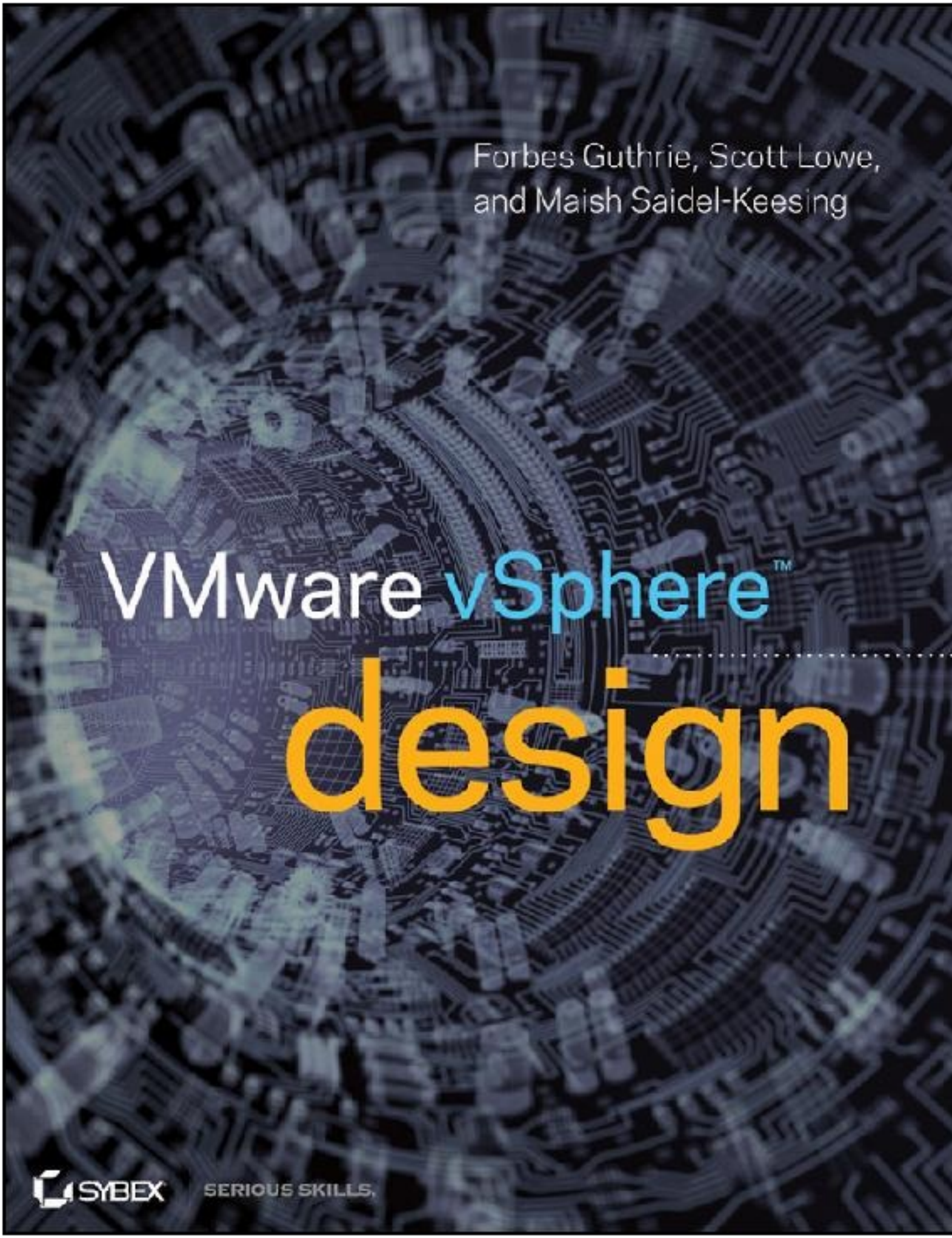

Forbes Guthrie, Scott Lowe,
and Maish Saidel-Keesing
VMware vSphere™
design
SYBEX
SERIOUS SKILLS.

Table of Contents

VMware vSphere Design

Forbes Guthrie

Scott Lowe

Maish Saidel-Keesing

Wiley Publishing, Inc.

Acquisitions Editor: Agatha Kim

Development Editor: Alexa Murphy

Technical Editor: Jason Boche

Editorial Consultant: Scott Lowe

Production Editor: Liz Britten

Copy Editor: Tiffany Taylor

Editorial Manager: Pete Gaughan

Production Manager: Tim Tate

Vice President and Executive Group Publisher: Richard Swadley

Vice President and Publisher: Neil Edde

Book Designer: Maureen Forys, Happenstance Type-O-Rama; Judy Fung

Proofreader: WordOne, New York

Indexer: Ted Laux

Project Coordinator, Cover: Katie Crocker

Cover Designer: Ryan Sneed

Chapters 1, 10, and 11 are published courtesy of EMC Corporation

Published simultaneously in Canada

ISBN: 978-0-470-92202-6

ISBN 978-1-118-06730-7 (ebk)

ISBN 978-1-118-06732-1 (ebk)

ISBN 978-1-118-06730-7 (ebk)

Library of Congress Cataloging-in-Publication Data is available from the publisher.

Dear Reader,

Thank you for choosing *VMware vSphere Design*. This book is part of a family of premium-quali Sybex books, all of which are written by outstanding authors who combine practical experience with gift for teaching.

Sybex was founded in 1976. More than 30 years later, we're still committed to producin consistently exceptional books. With each of our titles, we're working hard to set a new standard f the industry. From the paper we print on, to the authors we work with, our goal is to bring you the be books available.

I hope you see all that reflected in these pages. I'd be very interested to hear your comments and g your feedback on how we're doing. Feel free to let me know what you think about this or any oth Sybex book by sending me an email at nedde@wiley.com. If you think you've found a technical err in this book, please visit http://sybex.custhelp.com. Customer feedback is critical to our efforts Sybex.

Best regards,

Neil Edde
Vice President and Publisher
Sybex, an Imprint of Wiley

This book is dedicated to my beautiful wife Tarn. I grow and find inspiration from those around me; n one is closer. Ever yours.

—Forbes Guthrie

I'd like to dedicate this book to my Lord, who is the source of all knowledge (Proverbs 2:6). Thank you, Lord, for always giving me the knowledge I need, and help me give this knowledge to others. Thi book is also dedicated to my wife Crystal and our kids, especially Sean and Cameron. Guys, thanks fo hanging in there!

—Scott Lowe

For my Njoesh: you are and will always be my biggest fan. Thank you.

—Maish Saidel-Keesing

Acknowledgments

This book has been a considerable part of my life for the last six months. It was only possible
devote so much time to it with the help and support of my wife, Tarn. Without her encouragemen
this book would never have had my contribution. I can't overstate the part she played; without h
considerable *mateship*, I would never have gotten through it. Thank you.

I would like to acknowledge the book's coauthors, Scott and Maish. Their expertise and knowledg
shine through in their chapters and enhance the depth of entire book. Scott created the outline for t
book back at the beginning of 2010, with input from Duncan Epping of Yellow-Bricks' fame. Mais
jumped on board, and finally myself. For a while, Maish and I forged ahead with the writing, un
Scott agreed to join as an author.

This being my first published book, I was amazed at the number of publishing house staff involv
in a single project. First and foremost I would like to thank Agatha Kim, the acquisitions editor, f
her project steerage and the encouragement she gave to all the authors. We were all incredibly blesse
to have Jason Boche (the Virtualization Evangelist) as our technical editor to check the subject matt
and make suggestions so that every area was covered appropriately and was technically correct.

I grew up and was educated in Scotland, and I've lived and worked across the UK and in sever
English speaking nations including New Zealand, Australia, and subsequently Canada. M
interpretation of country-specific English lexicon, mixed with unique colloquialisms, makes for
frankly weird concoction of vernacular English. The Sybex editors' ability to decipher and transla
this into something representing a sane American English dialect was undoubtedly no easy tas
(However, I still maintain that the Queen's English is the only true authority, and virtualization shou
really be spelt with an *s*.) Alexa Murphy as the development editor probably bore the brunt of this a
was always central to the smooth passage of the editing process. The Sybex team of Pete Gaugha
Connor O'Brien, and Jenni Housh kept a close guard on standards and made sure things were tickin
along. The production team, headed up by Christine O'Connor and Liz Britten, did a tremendous jo
with Tiffany Taylor as the copy editor tidying the grammar into something my Mother might be prou
of. Proofreader Jen Larsen's rapier-like eye for detail helped spot all the little mistakes that the rest
us managed to miss along the way, and Ted Laux had the unenviable but crucial task of indexing th
text—thanks, guys.

From a technical perspective, the vast collection of resources from the VMware community, th
bloggers, the book writers, the podcasters, the VMworld presenters, the instructors, and the foru
members, all helped immensely. My knowledge and understanding of the vSphere product line
directly attributable to all of you. There are unfortunately too many people who deserve rich thank
but for fear of this turning into an Oscar speech, I can only say a huge *thank you*. You all know wh
you are. Here is a big virtual pat on the back from me.

Finally, I'd like to thank the wonderful baristas of Caffè Artigiano and Waves Coffee House on Wes
Broadway, Vancouver, for their delicious highly caffeinated beverages and working refuge.

—Forbes Guthr

As with any book, many people deserve credit for the book you're now reading. First and foremos
I'd like to thank coauthors Forbes and Maish for their outstanding work and unwavering dedication
getting this book completed, and for allowing me to be part of it. It's been a blast.

I'd also like to thank the team at Sybex: Agatha Kim, Alexa Murphy, Neil Edde, and the rest of th Sybex/Wiley team that worked so hard to bring this book to print. As with the previous books I'v done with Sybex, it's been a pleasure, and I'm looking forward to more books in the future.

My thanks go to our technical editor, Jason Boche, for his efforts on this book. Jason, thank you f your honest feedback; I do believe this book is better as a result of your input.

Finally, I'd like to thank my family for putting up with me as I raced to meet deadlines while tryin to balance work and home life. You guys are the greatest!

—Scott Lov

First and foremost, I would like to thank Hashem for giving me guidance and direction in th writing of this book.

I would to thank my wife Danja and my daughters Noa, Michal, and Avital for bearing with m throughout this adventure. Without your support and encouragement, I would not have gone on th journey and I would never have been able to complete this book.

To my Mom and my Dad: I hope this gives you great Naches.

I would also like to thank Agatha and the whole Sybex team for the opportunity to participate in th project. I appreciate your patience and understanding.

To my coauthors, Forbes and Scott, and to Jason, our technical editor, thank you for all the ideas an for your help with the late-night questions.

And last but not least, a huge *thank you* to all of you in the virtualization community. Without yo we would not have so much to talk about and would not be able to bounce ideas off each othe Together, we improve each other's knowledge; and by sharing that knowledge with others, we mak the world a better place.

—Maish Saidel-Keesin

About the Authors

Forbes Guthrie is a systems engineer and infrastructure architect who specializes in virtualizatio and storage. He has worked in variety of technical roles for over 12 years and achieved sever industry certifications including VMware's VCP2, VCP3, and VCP4. His experience spans mar different industries, and he has worked in Europe, Asia-Pacific, and North America. He holds bachelor's degree in mathematics and business analysis and is a former Captain in the British Army.

Forbes' blog, www.vReference.com, is well regarded in the virtualization field and is aggregated o VMware's Planet V12n web site. He is probably best known for his collection of free reference card long revered by those studying for their VMware qualifications. Forbes was awarded the lumina designation of vExpert by VMware for his contribution to the virtualization community. His passio and knowledge have also been rewarded with the peer-reviewed top virtualization bloggers listing f the last two years running.

Scott Lowe is an author, blogger, and consultant focusing on virtualization, networking, storage, ar other enterprise technologies. Scott is currently the team CTO for the vSpecialist team at EM Corporation. In this role, Scott provides support, technical leadership, and training to the vSpeciali team worldwide.

Scott's technical expertise extends into several areas. He holds industry certifications from Cisc EMC, Microsoft, NetApp, and others. He's also one of the few people who have achieved the status VMware Certified Design Expert (VCDX); Scott is VCDX #39. For his leadership and contributions support of the VMware community, Scott was awarded the VMware vExpert award in both 2009 ar 2010.

Scott has published numerous articles on virtualization and VMware with a number of differe online magazines, and has been a featured speaker at VMworld as well as other virtualizatio conferences. He has two other published books, *Mastering VMware vSphere 4* and *VMware vSphere Administration Instant Reference* (with Jase McCarty and Matthew Johnson), both by Sybex.

Scott is perhaps best known for his acclaimed blog at http://blog.scottlowe.org, where he regular posts technical articles on a wide variety of topics. Scott's weblog is one of the oldest virtualizatio centric weblogs that is still active; he's been blogging since early 2005.

Scott lives in the Raleigh-Durham, NC, area with his wife Crystal, the two youngest of their seve children, and their two dogs, Bo and Zach.

Maish Saidel-Keesing is a virtualization and systems architect working in Israel. He first starte playing with computers when the Commodore 64 and ZX Spectrum were around, and he's been at ever since. He has been working in IT for 12 years with Microsoft infrastructures and specifically wi VMware environments for the last 5 years.

Maish was awarded the VMware vExpert award for 2010, for his contribution to the virtualizatio community, the only Israeli to receive this award in that year.

Maish currently holds certifications from several international vendors, including VMwar Microsoft, IBM, Red Hat, and Novell. He is a member of the Server Virtualization Advisory Board o http://searchservervirtualization.techtarget.com, where he provides regular insight into ar contributions to the virtualization industry.

On his popular blog Technodrone at http://technodrone.blogspot.com, Maish regularly writes abo

virtualization, Windows, PowerShell, PowerCLI scripting, and how to go virtual in the physical worl

When he has free time, he likes to listen to music and spend time with his family. In general, h spends too much of his time on the computer.

Introduction

When we were first approached about contributing to this book it stood out as a particular interesting project. A multitude of vSphere textbooks are available, explaining every facet configuring ESX, ESXi, and vCenter. If you want to know how to do something in vSphere, you' literally spoiled for choice. However, in our minds, there are few resources properly encompassing th design process. They exist for very specific features, and some older pre-vSphere works are st available; but little cover the entire design of a vSphere implementation in sufficient depth.

vSphere is the leading industry standard for hypervisors. It's simply the best enterprise soluti available today. It has become this popular largely because of its wide range of features, efficienc and flexibility. But for it to perform effectively in your datacenter, you must have a suitab architecture in place. This book is written to help you achieve that.

Above all, this is a technical book about a very complex subject. But it's not concerned with th minutiae of every command-line tool, but rather the underlying concepts. As vSphere has evolve from the early ESX days, it has grown in size to the point that every detail can't be covered in a sing tome. But we sincerely believe this book covers its intended purpose better than anything el available. We'll dive into some areas not traditionally covered in such depth.

To that end, this book isn't a how-to manual with endless bullet-point instructions, but one that ain to make you think a little. It's for those of us who plan, design, implement, and optimize vSphe solutions. We hope it will challenge some of your preconceptions regarding the norm or what yo consider best practice. Just because you designed a particular configuration one way in the pas doesn't mean it's a best fit for the next rollout. Here we try to question that prescriptive bias. Usuall that choice exists because different situations call for different answers. If there was one best soluti for every case, then frankly no one would consider it a design choice at all.

This book isn't just for consultants who week by week deliver architectural solutions (although hope you guys are here for the ride, too); it's for anyone who runs vSphere in their environment. should make you question why things are set up the way are, and encourage you to examine how improve your environment even further.

There are constant advances in hardware, and vSphere is an ever-evolving tool, so it's always wor considering your existing deployments. Even if the hardware and software remain static in yo environment, you can bet that new VMs will continue to appear. Nothing stands still for long, so yo design should also be constantly growing to embrace those changes.

Each design decision has its own impact, and often these have knock-on effects on many oth elements. vSphere involves many disparate skills, such as guest OSs, server hardware, storage, an networking, and that's before you begin to consider the actual hypervisor. One of the hardest parts of creating a viable design is that normally, no individual choice can be made in isolation. Although th book is naturally split into chapters, sections, and subsections, it's only when the design is considere as a complete solution that it can truly succeed.

The book employs several techniques to understand how you can approach design. The critic requirements and constraints; the impacts, benefits, and drawbacks of each choice; and how decipher what will be best for you.

Who Should Read This Book

This book focuses on the design aspects of vSphere. It isn't primarily intended to teach you how complete certain vSphere tasks, but rather to make you think about the *why* behind your differe architectural decisions. We expect this book will be most useful for the following readers:

- Infrastructure architects designing new vSphere environments
- Engineers and architects charged with maintaining existing vSphere deployments, who wish further optimize their setup
- Anyone who appreciates the basics of vSphere but wants to learn more by understanding in dep why things are the way they are
- Long-time experts who are always searching for that extra nugget of hidden information

Ways to Read the Book

There are several ways to approach this book. Clearly, you can read it from cover to cover, and w certainly encourage anyone wanting the fullest understanding of vSphere design to do s Alternatively, if you need to brush up your knowledge on one key area, you can read each chapter isolation. Or, if you need a specific answer to a key design decision, you should be able to jump in an use this as a reference book. *VMware vSphere Design* has been written so each section stands on i own, if that is all you need from it, but it should also be a jolly good read if you want to sit down an immerse yourself.

Other Resources Available

We're often asked for good sources of vSphere information, for those seeking *absolute knowledg* Fortunately, there is a plethora of good places to look. The first stop for anyone (beyond this boo obviously) is VMware's own library of technical product documentation, which you can find www.vmware.com/support/pubs. Along with the standard PDFs, the site also offers a wide variety whitepapers, best practices, case studies, and knowledge-based articles.

Sybex has a number of excellent vSphere-focused books, such as *Mastering VMware vSphere 4*, VCP Study Guide and a Review Guide, and *Instant Reference for Administration*, among others.

A strong community of VMware users share knowledge through a number of different channels. Th VMware forums at http://communities.vmware.com/community/vmtn are an excellent source information and support for specific queries. There are a good number of vSphere-oriented blogs, th best of which tend to be aggregated on the popular Planet V12n site www.vmware.com/vmtn/planet/v12n. Finally, if you want something a little closer to home, us groups are available in many places (see http://vmware.com/vmug), where you have the chance meet other VMware users face to face to discuss and learn more about vSphere.

What You Need

To get started with *VMware vSphere Design*, you should have a basic understanding of virtualizatio vSphere itself, and the associated VMware products. Both networking and storage concepts a discussed, because they're integral to any vSphere architecture, so a basic knowledge of them assumed. The more hands-on experience you have with vSphere, the more you're likely to get out

this book. However, you don't need to be an expert beforehand.

No specific hardware or software is required while following this book, as long as you've seen th product before. But a lab is always useful to test some of many concepts we discuss. A simple neste VM lab run on a single platform should be sufficient to practice and explore most of the book content.

What's Inside

Here is a glance at each chapter:

Chapter 1: An Introduction to Designing vSphere Environments

We begin by introducing you to the design process for vSphere delivery. This chapter explains ho to understand the basic requirements, and how to assess and then design a successful, val implementation.

Chapter 2: ESX vs. ESXi

We explain the fundamental differences between VMware's two enterprise hypervisors: how t design around ESX and how to design around ESXi. This chapter compares the merits of each ar provides advice for those wishing to migrate an existing ESX deployment across to ESXi.

Chapter 3: Designing the Management Layer

In this chapter, we look at many of the software management pieces and how best to use them different design configurations.

Chapter 4: Server Hardware

This chapter provides an in-depth examination of the components that make up a server, and ho each one affects the performance of vSphere. You need to consider many factors when selectin server hardware, and we look at them, including scaling up versus scaling out approaches. We als debate the merits of blade and rack servers.

Chapter 5: Designing Your Network

This chapter covers the complex decisions you need to make to ensure that network traff provides sufficient throughput, redundancy, and security. We look how different vSpher components can affect those designs and provide some example configurations.

Chapter 6: Storage

In this chapter, we analyze the different factors that influence a complete virtualization storag strategy, comparing availability, performance, and capacity. We contrast different storag protocols and, finally, explain how to configure multipathing in different setups.

Chapter 7: Virtual Machines

In this chapter, we describe each VM component in turn, to help you understand how VMs shou be designed to make the most efficient solution for you. We look at how to optimize the OS an the applications within VMs, and then explain different methods of efficiently replicating the V design though the use of clones and templates. Additionally, we look at some of the techniques t protect those VMs with clustering solutions.

Chapter 8: Datacenter Design

This chapter examines in detail each element of a vSphere inventory's hierarchy. It looks at th importance of clusters in the design and how to successfully implement the resource-manageme and redundancy features of a cluster. We discuss resource pools, DRS, DPM, HA, and FT, and wh

interdependencies exist when they're used in combination.

Chapter 9: Designing with Security in Mind

Chapter 9 highlights some of the areas that security-conscious environments can use to ensure th vSphere is suitably strengthened. It explains the different security measures included in t hypervisor and the management tools, and how best to tighten that security as required.

Chapter 10: Monitoring and Capacity Planning

This chapter explains the concepts of monitoring and capacity planning. Monitoring relates to t present or recent past, whereas capacity planning looks to the future. The chapter also examin some of the common tools used for both and how to involve them in your design.

Chapter 11: Bringing It All Together

In this chapter, we return to the overall design strategy by looking at a specific example through design for a fictitious company. We look at several of the decisions made during the desig examine the justifications behind those decisions, and consider alternative choices that could hav been made.

How to Get in Touch with the Authors

We welcome feedback from you about this book or about books you'd like to see from us in the futur You can reach Forbes Guthrie by writing to forbesguthrie@vReference.com or by visiting his blog at www.vReference.com. You can reach Sco Lowe at scott.lowe@scottlowe.org or visit his blog at http://blog.scottlowe.org. You can reach Maish Saide Keesing at maishsk@gmail.com or on Twitter at @maishsk. Or, visit his web sit http://technodrone.blogspot.com.

Sybex strives to keep you supplied with the latest tools and information you need for your wor Please check their web site at www.sybex.com, where we'll post additional content and updates th supplement this book should the need arise. Enter **vSphere** in the Search box (or type the book's ISB —**978-0470922026**), and click Go to get to the book's update page.

Chapter 1

An Introduction to Designing VMware Environments

Designing VMware vSphere environments can be a complex topic, one that means many differe things to many different people. In this chapter, we'll provide an introduction to designing VMwa vSphere implementations. This introduction will give a preview of some of the more detail discussions that take place in later chapters, and will provide a framework for how all the oth chapters fit into the overall process.

This chapter will cover the following topics:

- The importance of functional requirements in VMware vSphere design
- The what, who, and how questions involved in VMware vSphere design and why they' important
- An overview of the VMware vSphere design process

What Is Design?

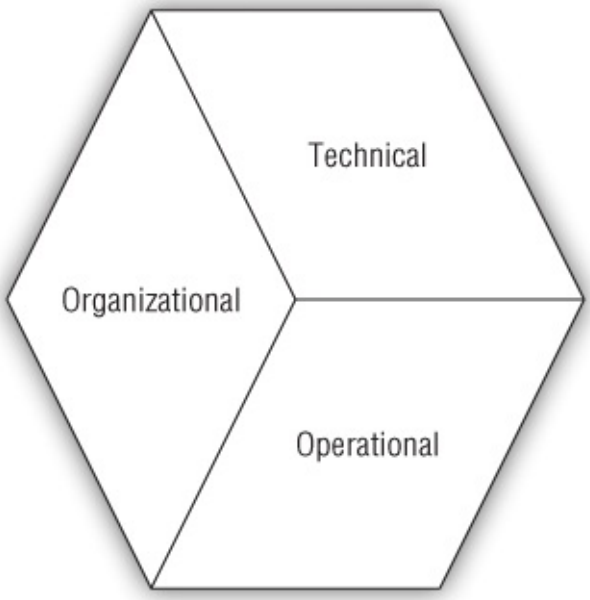
When we talk about “designing your VMware vSphere environment,” what exactly does that mean? the context of VMware vSphere, what is design? What does design entail? These are excelle questions—questions that we intend to answer in this chapter and the coming chapters throughout th book.

In our definition, *design* is the process of determining the way in which the different elements th make up a VMware vSphere environment should be assembled and configured to create a virtu infrastructure that is strong yet flexible. Design also includes the process of determining how th virtual infrastructure will integrate with existing infrastructure as well as how the virtu infrastructure will be operated after the implementation is complete.

That's a reasonable definition; but for someone who is new to VMware vSphere design, does th really describe what design is? Does it help understand the nature of design, or what makes up design?

In looking at a VMware vSphere design, you can say that VMware vSphere design has three ke facets: the technical or structural facet, the organizational facet, and the operational facet. Figure 1 shows how these three facets are all part of the larger entity that we refer to as *design*.

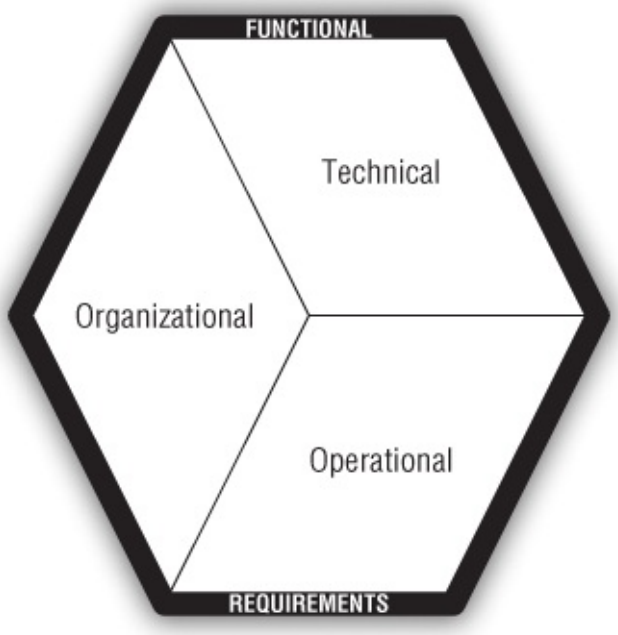
Figure 1.1 The different parts of VMware vSphere design are merely facets of a larger entity.



These three facets serve to organize the design in a way that is logical to us, grouping togeth information, decisions, criteria, constraints, and standards. We'll explore these facets in more deta later in this chapter in the section titled “The Facets of vSphere Design.”

When defined or described in this way, VMware vSphere design seems simple. As you'll see in th book—or perhaps as you've already seen, depending on your experience—it can be complex. Even the most complex of designs, however, there is a single unifying element that brings the differe facets together. What is this single unifying element that ties everything together, as illustrated Figure 1.2? This element is the functional requirements of the design.

Figure 1.2 The functional requirements unify the different facets of the design.



Functional requirements are incredibly important. In fact, we can't stress enough the key role th functional requirements play in VMware vSphere design (or any IT design task, for that matte Functional requirements are important because they answer the question “What *things* should th design *do*?”

It's important to remember that companies implement VMware vSphere for a reason, not just for t sake of having vSphere installed. As much as VMware would love for that to be the case, it's not.

every instance, there's a driving factor, a force, a purpose behind the implementation. There's a reaso the company or organization is implementing VMware vSphere. That reason, naturally, varies fro customer to customer and organization to organization.

Here are some example reasons taken from our own experience in the virtualization industry:

Consolidation The company or organization has too many physical servers and needs to reduce th number of physical servers. The need to reduce the number of physical servers can be driven b any number of reasons, including a need to reduce data-center space usage, a need to cut power an cooling costs, or an attempt to reduce hardware refresh costs.

New Application Rollout The company or organization is deploying a new application or a ne service in its data center, and it has chosen to use virtualization as the vehicle to accomplish th deployment. This may be a deployment of a new version of an application; for example, a company currently using Exchange 2007 may decide to roll out Exchange 2010 in a virtualize environment on VMware vSphere. As another example, a company deploying SAP may choose t do on VMware vSphere. The reasons for choosing to deploy on a virtualized environment are to numerous to list here, but they can include increased utilization, simplified deployment, and bett support for a disaster recovery/business continuity (DR/BC) solution.

Disaster Recovery/Business Continuity (DR/BC) The company or organization is in the midst developing or enhancing its DR/BC solution and has chosen to use virtualization as a ke component of that solution. Perhaps the company is using array-based replication and wishes use VMware vSphere and VMware Site Recovery Manager (SRM) to provide a more automate DR/BC solution. The choice to use virtualization as a component of a DR/BC solution is almo always a financial one; the company or organization wishes to reduce the amount of downtim (thus minimizing losses due to downtime) or reduce the cost of implementing the solution.

Virtual Desktop Infrastructure The company or organization wishes to deploy a virtual deskto infrastructure (VDI) in order to gain desktop mobility, a better remote-access solution, increase security, or reduced desktop-management costs. Whatever the motivation, the reason for th VMware vSphere environment is to support that VDI deployment.

As you can see, the reasons for adopting virtualization are as varied as the companies an organizations. There is no one reason a company will adopt virtualization, but there will be a reaso There will often be multiple reasons. These reasons become the basis for the functional requiremen of the design. The reasons are the *things* the design must *do*. Functional requirements formalize th reasons why the company or organization is adopting VMware vSphere and turn them into actionab items that you'll use to drive all the other decisions in the design.

Think about some of the examples we just provided. Does the organization plan to virtualize a ne rollout of Microsoft Exchange Server 2010? If so, then the VMware vSphere design had bett accommodate that functional requirement. The design must specifically accommodate Microso Exchange Server 2010 and its configuration needs, supportability requirements, and resour constraints. If you fail to properly account for the fact that Microsoft Exchange Server 2010 will ru in this virtualized environment, then you've failed to consider one of the design's function requirements—and, in all likelihood, the implementation will be a failure. The design will fail to d the *thing* the company needs it to do: run Microsoft Exchange Server 2010.

With this in mind, you can look back at Figure 1.2 and better understand how the function requirements both surround and unify the facets of VMware vSphere design. Continuing in ou example of an organization that is deploying Exchange Server 2010 in a virtualized environment, th

functional requirements that derive from that reason will affect a number of different areas:

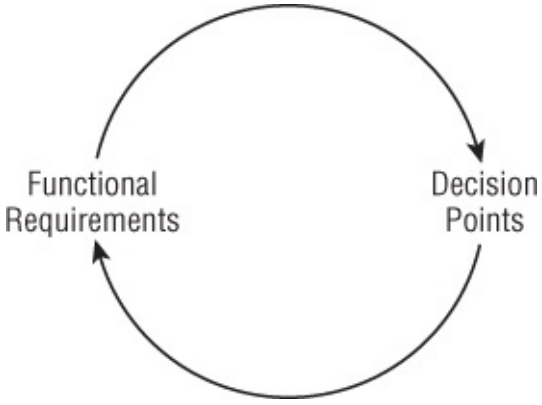
- The server hardware selected needs to be capable of running the virtual machines configured wi enough resources to run Microsoft Exchange Server 2010.
- The virtual machines that will run Exchange will, most likely, need to be configured with mo RAM, more virtual CPUs (vCPUs), and more available disk space.
- The configuration of Exchange Server 2010 will affect cluster configurations like the use VMware High Availability (HA), VMwareDistributed Resource Scheduler (DRS), ar VMwareFault Tolerance (FT).
- The cluster configuration, such as the ability (or inability) to use VMware FT, will in turn affe the networking configuration of the VMware ESX/ESXi hosts in the environment.
- Operational procedures need to be included in the design as a result of the use (or lack of use) features like VMwareHA, VMwareDRS, and VMwareFT.

This list can go on and on, but at this point you should get the idea. The functional requiremen affect almost every decision point in every facet of the design; as a result, they lie at the core creating a VMware vSphere design. Any design that doesn't directly address the organization functional requirements is a poor design, and the implementation won't be a success. Any consulta or VMware vSphere architect who attempts to design a vSphere environment without knowledge the functional requirements will fail. After all, the functional requirements are the targets the design aiming to hit; how can the design hit those targets if the targets aren't known and understood?

Interestingly, although the functional requirements directly affect the decision points—things lik what servers to use, the form factor of the servers, the number and type of network interface car (NICs), and so on—these decision points also affect the functional requirements. An inhere interdependency exists between the functional requirements and the decisions, as shown in Figure 1.3

> **Note**
>
> You'll likely see the term *design constraints* used in formal VMware design documentation. A design constraint is a decision point—such as the type of server you'll use, the type of storage you'll use, or the way in which you'll connect to an existing network—that has already been made and can't be changed. Because this decision point can't be changed, it constrains your design.

Figure 1.3 Functional requirements and design decision points are interdependent.



As a result of this interdependency, you'll find that creating a design is often an iterative proces Based on the functional requirements, you make a decision. Then, based on that decision, you ensu that the decision is capable of supporting the functional requirements. If so, you proceed with oth decision points. If not, you revise the decision point based on the functional requirements. Th

iterative process again underscores the importance of the functional requirements in the creation the design.

At the beginning of this section, we told you that design is the process of determining the way which the different elements that make up a VMware vSphere environment should be assembled ar configured to create a virtual infrastructure that is strong yet flexible. When we factor in the key ro that functional requirements play in unifying the technical, organizational, and operational facets of design, perhaps a better definition is that design is the process of determining the way in which th different elements that make up a VMware vSphere environment should be assembled and configure in order to satisfy the functional requirements. Or, in simpler terms, design is making the VMwa vSphere environment *do* the *things* it needs to do.

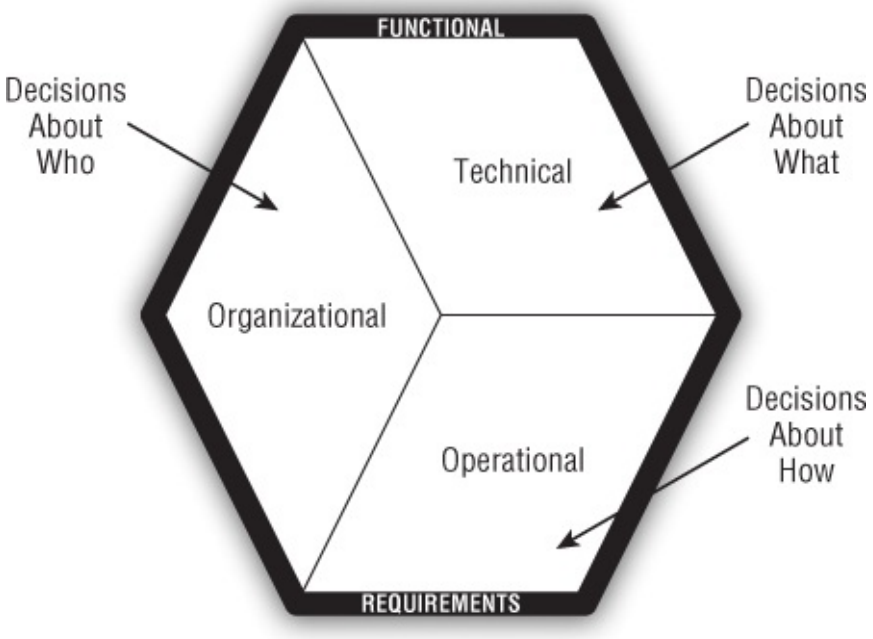
Now that you have a better understanding of what VMware vSphere design is and why it's importan in the next section we'll take a closer look at the facets of design.

The Facets of vSphere Design

As we described in the previous section and illustrated in Figure 1.1, your design must address thr facets, or the design will be incomplete. These three facets—technical, organizational, and operation —are unified by the functional requirements; but within each facet, a wide variety of decision poin must be specified in the design.

The best way to understand how these facets differ from each other is to look at the types decisions that fall in each facet. This is graphically depicted in Figure 1.4.

Figure 1.4 Each facet of the design primarily addresses a different type of decision, such as who, what, or how.



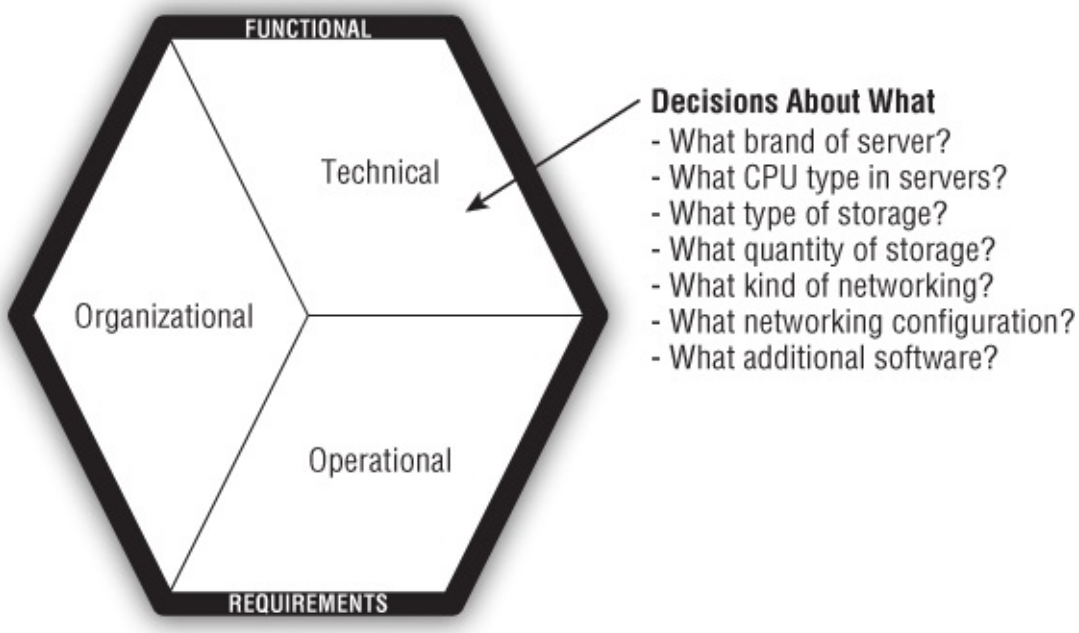
In each facet of the design, you'll make decisions based on the functional requirements, followed b an iterative review (as illustrated in Figure 1.3) to ensure that the functional requirements are still m based on the decision. In this section of this chapter, we'll take a deeper and more detailed look these facets, examining some of the decision points that are involved.

We'll start with the technical facet.

The Technical Facet

The *technical facet* is the facet that IT people most closely identify with design. It involves the piec and parts of technology that make up the final environment: things like what servers to use, wh quantity of random access memory (RAM) the servers will have, what configuration the storage arra will use for its datastores, and what the networking configuration will look like. You might also se the technical facet referred to as the *physical design*, although it incorporates certain logical aspects well. These are all decisions about what will or won't be included in the design, and all these decisio fall into the technical facet, as illustrated in Figure 1.5.

Figure 1.5 The technical facet includes the "what" decisions that are familiar to many IT professionals.



It's important to be sure the technical facet is as complete as possible, so the design should inclu —at the very least—decisions in the following technical areas:

- The number and type of servers in the environment
- The number, type, and speed of the CPUs in the servers
- The amount of RAM in the servers
- The type of connectivity to the shared storage
- The type or configuration of the shared storage
- The number of physical NIC ports available
- The manufacturer and model of the NICs in the servers
- The exact configuration of the virtual switches (vSwitches) and distributed vSwitches in t environment
- The amount of power required by the equipment
- The amount of cooling required by the equipment
- The amount of rack space or floor space required by the equipment

This is, of course, just a small list to get you started thinking about the detail you should provi when crafting a design for a VMware vSphere environment. Subsequent chapters examine each these areas in much more detail. For example, VMware vSphere networking is covered in detail Chapter 5, "Designing Your Network"; Chapter 6, "Shared Storage," discusses shared storage in mo