Making Everything Easier!™

Nutanix Special Edition

Software-Defined Storage

FOR DUMMIES®

A Wiley Brand

Learn:

- **What software-defined storage is and what it means for you**
- **About the technologies that make new storage strategies possible**
- **Why software-defined storage is poised to permanently change data centers**

Compliments of

NUTANIX®
The Virtual Computing Platform

Scott D. Lowe
Consultant and industry veteran

Software-Defined Storage

FOR DUMMIES®

A Wiley Brand

Nutanix Special Edition

by Scott D. Lowe

With contributions from:

Greg Smith
Sr. Director of Product Marketing
Nutanix

FOR DUMMIES®
A Wiley Brand

Software-Defined Storage For Dummies®, Nutanix Special Edition

Published by
John Wiley & Sons, Inc.
111 River St. Hoboken,
NJ 07030-5774
www.wiley.com

LIMIT OF LIABILITY/DISCLAIMER OF WARRANTY: THE PUBLISHER AND THE AUTHOR MAKE NO REPRESENTATIONS OR WARRANTIES WITH RESPECT TO THE ACCURACY OR COMPLETENESS OF THE CONTENTS OF THIS WORK AND SPECIFICALLY DISCLAIM ALL WARRANTIES, INCLUDING WITHOUT LIMITATION WARRANTIES OF FITNESS FOR A PARTICULAR PURPOSE. NO WARRANTY MAY BE CREATED OR EXTENDED BY SALES OR PROMOTIONAL MATERIALS. THE ADVICE AND STRATEGIES CONTAINED HEREIN MAY NOT BE SUITABLE FOR EVERY SITUATION. THIS WORK IS SOLD WITH THE UNDERSTANDING THAT THE PUBLISHER IS NOT ENGAGED IN RENDERING LEGAL, ACCOUNTING, OR OTHER PROFESSIONAL SERVICES. IF PROFESSIONAL ASSISTANCE IS REQUIRED, THE SERVICES OF A COMPETENT PROFESSIONAL PERSON SHOULD BE SOUGHT. NEITHER THE PUBLISHER NOR THE AUTHOR SHALL BE LIABLE FOR DAMAGES ARISING HEREFROM. THE FACT THAT AN ORGANIZATION OR WEBSITE IS REFERRED TO IN THIS WORK AS A CITATION AND/OR A POTENTIAL SOURCE OF FURTHER INFORMATION DOES NOT MEAN THAT THE AUTHOR OR THE PUBLISHER ENDORSES THE INFORMATION THE ORGANIZATION OR WEBSITE MAY PROVIDE OR RECOMMENDATIONS IT MAY MAKE. FURTHER, READERS SHOULD BE AWARE THAT INTERNET WEBSITES LISTED IN THIS WORK MAY HAVE CHANGED OR DISAPPEARED BETWEEN WHEN THIS WORK WAS WRITTEN AND WHEN IT IS READ.

For general information on our other products and services, or how to create a custom *For Dummies* book for your business or organization, please contact our Business Development Department in the U.S. at 877-409-4177, contact info@dummies.biz, or visit www.wiley.com/go/custompub. For information about licensing the *For Dummies* brand for products or services, contact BrandedRights&Licenses@Wiley.com.

ISBN 978-1-118- 85630-7 (pbk); ISBN 978-1-118- 85628-4 (ebk)

Manufactured in the United States of America

10 9 8 7 6 5 4 3 2 1

Publisher's Acknowledgments

Some of the people who helped bring this book to market include the following:

Development Editor: Kathy Simpson

Project Editor: Jennifer Bingham

Acquisitions Editor: Connie Santisteban

Editorial Manager: Rev Mengle

Business Development Representative: Kimberley Schumacker

Custom Publishing Project Specialist: Michael Sullivan

Project Coordinator: Melissa Cossell

Introduction

The way data centers operate is changing. Although virtualization brought major gains and improvements in data center economics and operations, it also introduced new challenges, particularly in regard to storage. As new and more varied workloads have been introduced into existing environments, storage devices have struggled to keep pace. Many of the problems arise from the legacy nature of today's most common storage systems, which are based on outdated, complex, and proprietary technologies.

But many storage problems can be solved by using software-based infrastructure based on commodity hardware. Complex software tools make it possible to move storage from proprietary hardware devices to software-led services that simply fit into the rest of the software-defined data center. In other words, the answer is software-defined storage (SDS).

Right now, SDS is a hot topic in the IT world. When any promising new technology emerges, the vendor community often goes into overdrive, with everyone working to gain either a piece of the market or to demonstrate leadership in the emerging space by using new descriptors for the technology.

This is simply a part of the hype cycle endemic to IT. When multibillion dollar markets are on the line, this isn't surprising. Fortunately, over time, the real benefits wrought from the new technologies generally become more widely known and understood. This book seeks to explain the real benefits of SDS while also making clear what SDS isn't.

About This Book

This book showcases the possibilities of a software-led world in which powerful software replaces proprietary hardware services as a key driver of data center and IT efficiency. Today's businesses demand simple, less complex technology services that don't require legions of IT staff members to operate. Instead, businesses want their internal IT functions to operate with the cloudlike efficiency of companies like Google and Facebook.

Cloud companies didn't achieve their success by maintaining the status quo: They traded traditional hardware for cheap commodity hardware and wrapped powerful software constructs around it. This setup gives the resulting environment all the availability, data protection, and administration benefits of hardware-led environments at a fraction of the cost and complexity.

You don't need an MBA or a PhD in storage to understand the concepts presented in this book. Regardless of your background, you simply need to have an open mind about the new software-led systems that are coming to data centers. (Kids these days and their crazy ideas!)

This book was written with and for Nutanix.

How This Book Is Organized

This book is organized into five chapters, which you can read in order or jump into wherever you like. That's the great thing about *For Dummies* books. You can read the parts you need or want to and don't have to read the rest. Or you can start at page one and press on through to the end.

- **Chapter 1: The Current State of Storage:** This chapter introduces you to the challenges of legacy storage.
- **Chapter 2: Software-Defined Storage Basics:** What is software-defined storage? Also, what isn't it?
- **Chapter 3: Key SDS Principles and Enablers:** Find out what technologies make SDS possible.
- **Chapter 4: How SDS Benefits Business:** This chapter explains the things that your Chief Financial Officer wants to know about SDS. Which is to say, how to make it more flexible and more efficient!
- **Chapter 5: Ten Key SDS Facts:** This chapter goes over some important facts to keep in mind as you explore SDS.

Foolish Assumptions

I assume that you understand general data center concepts, including the various resource silos, and that you have a passing knowledge of virtualization. I also assume that you have an interest in storage and IT efficiency, and in administering the whole kit and caboodle.

Icons Used in This Book

The margins of this book sport several helpful icons that can help guide you through the content.

TIP

When I present something that can save you time and effort, I toss in this icon to highlight it.

REMEMBER

Take note of anything marked with a Remember icon, because it's important to keep in mind.

WARNING!

This icon flags information to take note of, because it could cause problems.

TECHNICAL STUFF

You don't have to read Technical Stuff material, but you can gain deeper understanding of a topic if you do.

Chapter 1

The Current State of Storage

In This Chapter

- Outdated, expensive, or overly complex technology
- Isolation of storage from other resources
- Storage must embrace virtualization

Even when existing solutions meet market needs, innovation never stops. Although legacy systems met the storage challenges of yesterday, they've proved to be inadequate and inflexible to meet emerging demands. And who knows what may come down the road to present further challenges?

Today's Storage Is Yesterday's Technology

Many of the most well-known storage technologies on the market today were designed during a different era in computing. They were built with the idea that applications and associated workloads would always run on dedicated servers, which hasn't been the case since the rise of virtualization and ubiquity of virtual machines (VMs). Unfortunately, the development and administration of legacy storage systems has also failed to keep pace with emerging needs, such as cloud-based provider principles, VM-centric data center services, scale-out architectures, and on-demand provisioning features.

Until fairly recently, legacy storage systems were also unable to seamlessly leverage a major advancement in storage: solid-state (flash) storage. Flash storage was the first major development in storage hardware for many years. Sure, the market rejoiced when 15,000 RPM SAS disks hit the market in response to the need for increased storage performance, but developments in the early to mid-2000s were evolutionary in nature rather than revolutionary.

Today's Storage Is Overly Complex

Many practitioners laud the rise of virtualization as a turning point in the history of the data center, and they're absolutely correct. Virtualization only tamed the compute beast, however; it did little to address storage challenges in the data center. In fact, it likely exacerbated the situation.

At times, it seems that you need a PhD in storage systems to tune storage for the workload demands of various applications. Over the years, many constructs and features have been introduced, but most of them have simply added more complexity. Consider the following storage constructs, which storage administrators commonly manage:

- **LUNs:** Logical Unit Numbers (LUNs) are virtual constructs comprised of underlying physical storage resources. LUNs allow storage administrators to slice up physical storage in ways that make sense for applications running in the data center. To ensure that LUNs are seen only by authorized hosts, organizations must then use LUN masking.
- **RAID groups:** A redundant array of independent disks (RAID) group is a set of physical disks assigned to a group and operating at a RAID level, such as RAID 1, RAID 5, or RAID 10. The chosen RAID level significantly affects how well data is protected and how well storage performs. Intricate knowledge of RAID levels is essential to tuning legacy storage for diverse workload performance needs.

- **Tiering:** Not all workloads are created equal. By using tiering, organizations create storage layers of varying performance and protection. Storage intended to support a large data analytics application, for example, might use a LUN that was created in the performance tier and built on high-speed 15K RPM SAS disks.

The terms and overall concepts themselves aren't all that difficult to understand, but behind these terms are reams of information that only deep experience in storage characteristics can penetrate. Storage experts have no problem deciphering this information, but understanding may be bit more challenging for IT generalists in small companies, who need to spend a lot of time researching how to manage storage while their organizations are left with unused or untuned storage.

And that's exactly the problem. At one time, businesses could be patient while IT departments figured out the latest technologies. Today, chief information officers (CIOs) must react quickly to new and changing business needs, so they can no longer assign staff members to figure out key systems and hope for the best.

Moreover, the tools used to manage legacy storage don't always align with what has become the common unit of modern data center architectures: VMs. The vast array of VMs running in a typical data center demand different storage services and, frequently, different service levels. Storage management must reorient itself to the new reality that services need to be managed at a per-VM level. (I discuss VMs again in "The Hypervisor Rules the Data Center" later in this chapter.)

Finally, legacy storage systems are commonly architected around a scale up model. In a scale up environment, when more capacity is needed, customers must deploy entire new shelves of disks even if only a small amount of additional storage is necessary. Legacy storage environments create scenarios in which customers must make inefficient technology decisions, such as overprovisioning of storage, due to limitations in the legacy architecture.

Today's Storage Is Built with Proprietary Technology

In legacy systems, hardware rules the day. Storage arrays have often been filled with specialized hardware intended to help the array do the best job possible on a few key features. In fact, many arrays are built from the ground up with custom and proprietary hardware.

Many legacy systems use custom field-programmable gate arrays (FPGAs) and application-specific integrated circuits (ASICs) to achieve their goals. To provide inline deduplication services without significantly affecting the overall performance of a storage array, for example, a vendor may implement a custom ASIC to perform this one function. These FPGAs and ASICs are placed on custom controller modules, which are designed to work with the custom-designed chassis used by many vendors of legacy arrays. With each new array generation and model, hardware components — including FPGAs, ASICs, controller boards, and chassis — are redesigned to meet the goals of the new units.

This constant hardware engineering drives up research and development costs, which eventually must be passed on to customers in the form of higher prices. In addition, the hardware-centric approach creates an inflexible system that can't always be refitted to meet new needs. Applications dependent upon custom hardware can force premature rip and replace operations, creating expensive economic losses. In short, the legacy approach is costly and inadaptable. It prevents easy integration among vendors, drives up long-term storage costs, and ultimately slows innovation.

All these issues raise the need for a storage layer that could be built using commodity hardware and without the proprietary components that currently enable advanced storage features. See Chapter 2 for more on this.

Today's Storage Is a Resource Island

Virtualization enabled IT to take a centralized, top-down look at the entire computing environment as a single entity. Along with new levels of visibility into the environment came an awareness between the operating platform — the hypervisor — and the workloads that were not present before. A single hypervisor now supported all workloads rather than individual units of hardware doing so individually. This single platform yielded the ability for the hypervisor to be aware of all applications running across the entire environment, and an opportunity to tailor services for individual VMs.

Even though the situation has improved over time, storage has largely remained a resource island. It's been kept in the dark about the status of other resources in the environment — most significantly, networking and the virtualization tier. As a result, storage remains blissfully unaware of what kinds of workloads it can expect unless an administrator explicitly defines those workloads. Even then, existing storage technologies make it very difficult, if not impossible, to provide the right services for each workload in the environment. This lack of integration with and awareness of other data center components increases overall project risks and makes it difficult to introduce simplicity to the organization.

All these problems raised the need for a storage layer that could automatically adapt to changing needs and take a proactive role in ensuring ongoing application performance. See Chapter 3 for more on this.

Today's Storage Is Expensive to Scale

In a legacy storage environment, scaling can require IT to make costly decisions. The solution generally involves adding a shelf of disks, but it can be costly due to the proprietary nature of legacy storage and the often-high cost per gigabyte of additional storage.

On the performance front, the decision is even more difficult, particularly in systems that don't or can't seamlessly leverage flash storage. To increase performance capabilities in these systems — that is, to add enough input/output operations per second (IOPS) capacity to support key workloads — means adding more disk spindles even when total storage capacity isn't an issue. This solution makes legacy storage environments expensive to scale. Balancing capacity and performance needs becomes practically impossible, which makes it difficult to predict future storage needs.

Further, when storage services are delivered in a one-size-fits-all manner, the overall cost of storage is difficult to contain. Legacy storage systems apply computationally expensive and capacity-intensive services to all hosts in an environment, without regard to the requirements of each guest VM. The inability to discriminate among individual workloads means that virtualization teams have no choice but to absorb the full cost of delivering the highest level of availability; for example, to data that is ephemeral or is less critical to the business.

All these issues drove creation of storage that could be implemented in a way that met ongoing needs while also providing a predictable IT cost model. See Chapter 4 for more on this.

The Hypervisor Rules the Data Center

The hypervisor has become the de facto standard platform on which new workloads operate in the data center. Even though some storage vendors have taken steps to address this reality, they've done so by using constructs provided by the hypervisor vendors — most notably, VMware.

Here are the primary sets of application programming interfaces (APIs) that VMware created to help storage vendors adapt their storage arrays for the VM-centric nature of modern data centers:

- **vSphere API for Array Integration (VAAI):** To increase VM density on host machines and improve the performance of common storage operations, such as block zeroing and file/block copies, VMware created a set of APIs that storage vendors can use to provide better integration of the virtualization and storage layers in a data center.
- **vSphere APIs for Data Protection (VADP):** Rather than running backup operations from individual VMs and disrupting the workloads running inside those machines, vendors can use VADP to enable backup software to perform centralized VM backups.
- **vSphere API for Storage Awareness (VASA):** With VASA, administrators can gain insight into array capabilities, enabling tighter integration between the hypervisor and the storage layer. Information such as details on storage virtualization, configuration, capacity, and thin provisioning can be gathered and passed up through vCenter Server to the user.

All these advancements in integrating the hypervisor with storage are software-based. An underlying premise of software-defined storage (SDS) — the very topic of this book — is that the hypervisor is the new bare metal of the data center. In a software-defined data center (SDDC), all services are built on the virtualization layer, which not only explicitly decouples the data and control planes, but also allows storage functionality to extend to the time of creation (late binding). Instead of depending on rigid hardware constructs to meet the needs of all your workloads, features and policies can be applied via the hypervisor. The hypervisor and SDS work together by providing a menu of services (APIs) to learn about the capabilities of various hardware devices and apply the right capabilities and properties as needed, on a per-VM basis.

As vendors build services on the virtualization layer, development must be done independently of the actual hypervisor running in the environment. That is, the services can't be tightly coupled to, or embedded within, any one vendor's hypervisor; services must be built in an agnostic manner or else enterprises will lose the inherent flexibility that comes from virtualization.

Today, although many data centers continue to use legacy storage devices, a virtualization-centric software layer is showing signs of correcting storage issues.

Chapter 2

Software-Defined Storage Basics

In This Chapter

- Seeing how the SDDC works
- Understanding requirements for virtualization
- Declaring what SDS isn't

Software-defined infrastructure concepts are bringing innovation and new possibilities to the data center. Software is poised to completely transform all aspects of the data center, of which storage is but one component. In this chapter, I provide an overview of software-defined storage (SDS) in data centers and discuss what SDS is — as well as what it isn't.

In its 2013 Storage Taxonomy Report, analyst group IDC describes SDS as any storage software stack that can be installed on any commodity resources (x86 hardware, hypervisors, or cloud) and/or off-the-shelf computing hardware. Further, such solutions are used to offer a full suite of storage services and federation between the underlying persistent data placement resources to enable data mobility between these resources.

The Anatomy of the Software-Defined Data Center

Although the term *software-defined data center (SDDC)* only came into existence in 2012, the concept behind SDDC got its start back when VMware brought x86 virtualization to market, an introduction that has revolutionized the way that business-critical workloads are deployed, managed, and protected (see Chapter 3). Because the hypervisor-based software layer became the glue that binds together workloads and brings efficiencies to the server tier, administrators have many new workload management options. To move a running workload, for example, they can move it from one host to another, from one data center to another, or even from a private data center to one operated by a cloud provider.

Led by hypervisor-based *software-defined servers* (virtual machines) provided by vendors such as VMware, Microsoft, and Oracle, other components of the data center are getting a software-based makeover as well, with the eventual final product being the SDDC. One of the foundational tenets of SDDC is that all data center services are delivered via software, running on commodity hardware.

Layers

In order to provide critical business services, the final product — a fully software-defined data center — will consist of a number of distinct layers. This book focuses on the storage layer, but as an integrated system, an understanding of all of the layers in context is necessary.

Compute layer

The rise of software-defined servers via the hypervisor has mainstreamed the idea that software running on commodity hardware might ultimately supplant proprietary hardware platforms. The end result is likely a data center comprised of rack after rack of identical x86 servers.

Networking layer

One of today's hottest topics is the "softwareitization" of the network layer, a transformation that yields software-defined networking (SDN). SDN presents new opportunities for flexibility, cost avoidance, and reduction of administrative overhead. The network's control and data planes, for example, which have traditionally been tightly coupled and distributed across every networking device in the environment, are decoupled (see "Decoupled control and data planes" later in this chapter).

The data plane remains widely distributed to enable data transmission to every device, but the control plane becomes a centralized, software-enabled service that has insight across the entire environment. As a result, networks based on SDN can leverage lower cost devices, more easily work around physical faults, and reduce the need for administrative intervention when application needs change.

Storage layer

Today, the storage layer is gaining attention among vendors as they strive to build products that reduce or eliminate the challenges inherent in traditional storage. The SDS vision has yet to permeate the wider IT community, although it is gaining momentum.

As is the case with SDN, SDS seeks to separate the physical storage hardware from the storage logic that determines data placement and what services are applied during read and write operations. The end result is a storage layer that is very flexible, which can adjust to meet shifting application needs. It also creates a unified and coherent data fabric that maintains full visibility of each virtual machine.

Services layer

Even though getting to the full promise of the SDDC is a laudable goal for any organization, this book concentrates on the software portion of that goal: SDS. To that end, the focus in the following sections is on how the SDS services layer provides storage features.

Storage services

Storage services are still important, and just because they've been moved into a software layer doesn't mean that the customer should have to compromise and give up enterprise-class features. In fact, delivering such features via software allows them to be expanded and improved over time. These features include:

- **Dynamic tiering:** Because emerging storage systems generally support a powerful combination of high-performance flash storage and slower, but larger capacity, hard disk drives, software-based dynamic tiering enables the storage layer to shift data between tiers of storage automatically to optimize performance. No administrator is required to deal with complex rule sets.
- **Caching:** Caching has become an increasingly important feature, primarily due to cost reductions for flash storage, and has enabled new classes of storage, such as server-side cache devices and hybrid storage arrays.

But even though flash storage costs have come down, it's still much more expensive than hard disk drives. As such, storage vendors are carefully leveraging flash in ways that enable faster performance by placing the most in-demand data in a higher speed cache.

- **Replication:** Storage replication creates a variety of data protection (local replication) and disaster recovery (replication across different locations) opportunities, which is why many organizations consider it to be an essential storage feature. With replication, organizations can copy production data between different storage systems located in geographically disparate data centers.
- **Quality of service (QoS):** The goal of QoS is to ensure predictable, high performance for each application. Historically, IT teams have avoided mixing different types of workloads (for example, Microsoft Exchange, SQL databases, and VDI) on a common platform because they compete for resources, thus jeopardizing performance SLAs. Having a distributed control plane, however, enables data to be stored locally to any specific VM so that performance is protected and performance characteristics can be easily observed and analyzed
- **Snapshots:** Snapshots provide a point-in-time copy of the storage system, from which they can be recovered. Snapshots don't replace proper backups, but their ability to deliver a low recovery point objective has made them the go-to method for most recoveries.
- **Deduplication:** Although storage capacity costs continue to come down, even for flash storage, most people don't want to waste capacity when reasonable conservation opportunities are available. Deduplication is a popular way to achieve this goal. Deduplication can save organizations money, thanks to reduced capacity needs.
- **Compression:** Data compression is another feature in the data reduction portfolio that helps organizations maximize storage capacity. Although data deduplication works at the block level, compression works at the file level and can crunch files down to a fraction of their original size.
- **Cloning:** In certain projects —such as virtual desktop projects — cloning is a sought-after feature because it helps administrators streamline and improve overall service.

Virtualization and per-VM service delivery

Some people may contend that *virtualization* and *software-defined storage* are interchangeable terms, but nothing could be further from the truth. SDS is a superset technology, and virtualization is one of its primary components.

At its most basic level, virtualization is about abstraction (see the next section). Abstraction effectively decouples the workloads from the underlying hardware. The workloads now operate inside a software-led construct called a virtual machine (VM).

For a software-defined storage solution, all of the previously mentioned services must operate at a per-VM level, so resources are used wisely and VM mobility is preserved across the SDDC.

Architecting solutions at the VM level provides more flexibility in responding to business needs. If a virtualized workload has new components added, for example, it becomes easy to protect the investment using replication for just that VM. In contrast, by using coarse-grained, hardware-based solutions like RAID, IT teams may have to replicate hundreds of unimportant workloads, or redesign the entire infrastructure to meet the new requirements.

Just the facts on abstraction

Like virtualization and SDN, abstraction is a key element of SDS. Without virtualization, in fact, SDS wouldn't be possible, because SDS depends on the fact that storage resources are abstracted (or decoupled) away from hardware. When they're abstracted, those services are extended through additional software mechanisms. These attributes determine, for example, where to store data based on application need; they also provide important storage services, such as deduplication and thin provisioning. But, these mechanisms aren't part of the virtualization stack; they're additional services that provide additional functionality.

Another architectural attribute that's increasingly common in SDS is software that enables linear scale-out capabilities. This feature relies on virtualized storage, but it enables the storage environment to meet customers' ongoing capacity and performance needs.

Using commodity hardware in abstraction

By abstracting physical resources, organizations can begin to leverage different kinds of underlying hardware. The abstraction portion of SDS enables organizations to move to less-expensive commodity hardware while still maintaining the complete set of storage services needed in today's enterprise.

Many traditional storage constructs, such as RAID groups and LUNs (see Chapter 1), were created to provide usable administrative objects to manipulate. Although they may have succeeded in their efforts, they also added a lot of complexity to the storage equation. Through abstraction, many of these constructs can be eliminated or hidden from data center management teams, instead allowing virtualization teams to provision and manage storage on a purely VM-centric basis so that storage workflows align with application management.

How can this be? With SDS, the management layer knows exactly what's happening in the storage system and in other areas of the data center, including the virtualization tier that is running a variety of applications and workloads. Administrators don't need to perform as much initial configuration of storage because the storage resource can create its own constructs based on specific application needs.

An SDDC not only delivers all services via software, but it separates the control and data planes for each individual service or application. In such a loosely coupled, distributed model no one component will bottleneck the system. To illustrate, consider the implications if the storage logic depends on using a hardware device like a RAID controller to provide availability. In that case, all future flexibility becomes chained to that hardware device and the system moves away from being software-defined. Further, separation of the storage intelligence from the underlying hardware provides multiple benefits. For instance, if a drive fails in the hardware layer, distributed compute resources can all participate in the data rebuild and recovery process.

In an SDS system, the control plane enables policy-based management of storage services that is decoupled from the hardware environment. The importance of this decoupling becomes more apparent as multiple hardware systems are added to the environment. Given visibility into the full spectrum of hardware, IT can extend data placement beyond a single system or single location, leveraging cloud and even off-premises storage systems. An SDS system can have hardware present in many data centers — both public and private — and still be managed and controlled via a single, distributed control plane.

Open APIs

Besides being open from a hardware perspective, a key characteristic of the SDDC and SDS is interoperability facilitated by open application programming interfaces (APIs). Such APIs enable ongoing automation beyond direct storage management and enable provisioning that can be controlled by third-party extensions to the environment. A specific application may need a particular kind of storage, for example, and could use an SDS vendor's open APIs to provision that storage by using the knowledge of the environment that has been gleaned by the SDS management layer. Today, the most common API standard is called the representational state transfer (REST) API.

What SDS Isn't

SDS isn't just marketing hype, although some legacy vendors may see it as that. That's not surprising after all, as legacy vendors have a vested interest in maintaining the status quo — expensive hardware, expensive software add-ons as options, proprietary hardware, and high margins — for as long as possible. In fact, some vendors lash out at SDDC and SDS to generate fear, uncertainty, and doubt among their customers.

In reality, though, as more SDS solutions enter the market, they'll become more compelling and powerful, and they'll be able to compete with the industry behemoths, most of which are working on products that adhere to their own visions of SDS.

Not just virtualization

Tempting as it may be to label some familiar constructs as SDS, many products that are marketed as such aren't SDS in the true sense of the concept.

Although virtualization is a critical component of SDS, it's only one small part of the broader SDS picture. Imagine an archery practice target. If you were using a similar image to describe SDS, virtualization would be the innermost circle; the next circle would represent the software layer that enables the advanced functionality provided by SDS, and so on. Although storage virtualization is important in that it provides the abstraction that SDS needs to operate, a solution is just virtualized storage without the accompanying management layer.

Not just federated storage

Likewise, SDS isn't simply federated storage, although with the right management framework, it would get closer. *Federated storage* — a collection of disparate storage resources governed by a common management system — doesn't equate directly to SDS. The underlying storage systems may still have their own hardware constructs to enable scale and data protection and may not be built on commodity hardware. A common federated storage construct on the market today is the virtual storage appliance, or VSA for short. Although federated storage systems may provide some of the workload placement and enterprise feature sets of SDS, they may use proprietary hardware to do so, so they're not truly SDS systems.

Not just software

SDS isn't a software-only solution; it's a software-directed solution that sits atop commodity hardware — for both compute and storage. This hardware should still undergo some form of selection process to ensure that it can support enterprise-level workloads. After all, an organization that seeks to save a few bucks by buying refurbished 5,400-RPM hard drives shouldn't expect to enjoy long-term stability or significant performance from its storage system, whether it's software-defined or not.

This is why many vendors that sell what they claim to be commodity-based SDS systems actually sell bundles of hardware *and* software. In the real world of the enterprise, ongoing support for infrastructure solutions is a must-have feature. Having the ability to support a product means the difference between ultimate success and failure. Just because a vendor sells hardware along with software, one should not discount the software-based nature of the solution. This is why companies such as Nutanix sell what many consider to be SDS — or at least partially SDS — even though they are sold as a part of a hardware/software bundle.

There are other vendors that provide software only SDS, but they're only addressing the first "S" — the software. However, to their credit, these other vendors also make available a complete hardware compatibility list with varying levels of testing per device. The difference between an SDS vendor that bundles hardware and one that doesn't is that the software-only vendor doesn't provide support for the hardware solution. They simply sell and support the software side of the SDS house.

Not just about capacity and performance

Although SDS can improve both the capacity and performance characteristics of a data center, SDS is also a way to rethink the entire storage paradigm to achieve many benefits, including these:

- **Lowered costs:** IT departments need to find ways to lower technology-focused capital and operational costs and redirect savings to activities that provide more value to the organization. Traditionally, storage has been a very expensive part of the data center.
- **Automation opportunities:** With a complete management layer (control plane) sitting atop the storage hardware (data plane), and with a full set of APIs, SDS can be a full partner in data center automation and orchestration opportunities. These kinds of projects are being undertaken by forward-thinking IT departments that want to eliminate non-value-adding operational support processes in favor of automated services. Such endeavors can help IT take an exception-based approach to operations rather than having to be involved throughout the process.
- **Application enablement:** Not all applications are terribly smart, but as the data center becomes more software-led, applications will gain more intelligence. This intelligence will enable them to instruct individual data center components to perform particular tasks, as well as to inform control planes about the features and performance levels required to operate the application. For example, imagine an application that knows a peak processing period is about to begin. This application

> can proactively notify the storage layer that it will need higher performing disks during this period and, due to the nature of the surge, it also needs to request a higher replication factor to ensure the highest levels of replication. In response, the software layer could even dial down data protection on less critical workloads for the duration of this surge. In a future world, all of this could happen without an administrator ever getting involved.

No feature compromise required

Given today's powerful processors and underlying hardware, such as flash based solid-state disks, it should come as no surprise that an SDS solution can keep up with even the largest proprietary storage systems on the market in terms of both features and performance.

Considering the fact that many legacy storage systems force customers to take an a la carte approach to enterprise storage capabilities, such as deduplication, SDS may be able to give customers a more comprehensive array of out-of-the-box features and capabilities.

No public cloud required

The emerging software-led infrastructure may be based on the advanced architectures of the largest cloud players on the planet, but you don't have to use a cloud provider to use cloud solutions. As enterprise-focused companies have worked to refactor cloud-scale solutions — such as networking, compute, and storage services from Google and Facebook — into components consumable by the enterprise, they have done so with the understanding that the customers will generally deploy solutions into private data centers rather than at cloud providers.

At the same time, though, customers will demand on-premise programmability and economics that rival public cloud storage services. This is where it becomes the most obvious that emerging SDS solutions were born in the data centers of cloud providers and are now being translated into the enterprise realm. Today's SDS solutions achieve local programmability, but with cloud-scale economics and benefits, thus reducing the cost of storage while making it more flexible in the data center. Chapter 3 explains the concepts behind SDS that enable these outcomes.

Chapter 3

Key SDS Principles and Enablers

In This Chapter

- Running on x86 servers
- Automating storage decisions
- Taking advantage of flash

What makes SDS a serious contender to transform the storage market? A confluence of events and technologies has made it possible for vendors to build storage architectures in ways that they never have before. This confluence, along with a desire to reimagine the whole data center, is helping to drive SDS innovation and adoption. In this chapter, I discuss the principles and enablers that are making SDS a more popular topic.

The Dominance of x86

Compared with the Intel processors of ten years ago, today's x86 processors are beasts, even in the midrange market. These processors, with as many as ten CPU cores per socket, have more than enough horsepower to handle tasks that used to require custom application-specific integrated circuits (ASICs), which makes the resulting product a bit less complex and a bit less expensive. Virtualization enhances x86 because you can create virtual CPUs to granularly schedule and assign processing resources to virtualized workloads.

One of the key aspects of SDS is its capability to leverage the economics and ubiquity of commodity hardware. x86 is the cornerstone of this capability, having proven itself for more than a decade as the foundation of the virtualization movement. The x86 architecture has beat out just about all other architectures in the data center processor race, and as Intel continues to extend its capabilities (by adding virtualization extensions, for example), x86 has become the processor of choice.

Mainstream Intel processors are now multicore processing behemoths with capacity to spare, even in demanding circumstances. Moreover, they're relatively cheap, particularly when compared with the custom processors and ASICs used in legacy storage devices. Further, they're a known quantity, which allows vendors to focus innovation on more important areas of the storage stack.

In SDS systems, software-based constructs replace the fixed hardware elements of legacy systems, such as RAID controllers and custom ASICs. Also, x86 CPUs drive fast, capable software-centric storage controllers, which are fundamental to SDS. Implemented correctly, these processors enable systems to scale to meet enterprise needs while delivering enterprise-grade features.

Because Intel operates on a processor life cycle of less than 18 months, as opposed to the traditional storage array life cycle of four to five years or more — innovation in SDS systems can take place at a much faster rate. With just minor modifications, such as replacing or upgrading processors, customers can enjoy these innovations without having to replace their entire storage environments. For example, in Intel's latest processor, Haswell, there is added support for DDR4 RAM, which can dramatically improve performance. Ivy Bridge, Intel's prior generation processor, added support for 16 PCI Express 3.0 lanes, dramatically expanding scalability opportunities.

Automatic Storage Decisions

At one time, the major decisions about storage revolved around whether an application required block-level, file-level, or object-level storage. In the world of SDS, however, data center teams are no longer involved in these decisions. Underlying physical storage may still provide these three kinds of storage services, but the intelligent software layer decides where to store data based on application and data needs and characteristics.

Different kinds of physical storage are used for different application and data needs:

- **Block storage:** This type of storage offers lower level protocols for fine-grained data transfers, but is lacking metadata and requires a higher level file system to make it usable. VMware VMFS-based volumes are block-level volumes.
- **File storage:** This type of storage is a higher level structure that's easier to manage, but often difficult to scale. A Windows file server's storage is an example of a storage system that uses file-level storage.
- **Object storage:** This type of storage is well suited for scale out, but has a hard time supporting data that frequently changes or is transactional. Data is treated as an object and each object carries with it the data payload, any metadata associated with the object, and a globally unique identifier that enables the object to be located again.

Compatibility with Commodity Hardware

The ability to leverage commodity hardware is a pillar of the software-defined data center (SDDC) and a critical component of SDS. Vendors can use off-the-shelf components to build inexpensive storage systems, often with common server chassis, Intel x86 processors and motherboards, and available hard drives and solid-state drives. The use of Intel x86 processors means that vendors don't have to conduct research and development on processors or other hardware, so they can focus on the software side — the real power of the SDDC.

The goal of SDS is to leverage storage hardware for which there is factory-defined nothing. *Factory-defined nothing* means that the storage functionality isn't predetermined by componentry installed during the hardware manufacturing process. Instead, it has the inherent flexibility to adapt and augment functionality during its active lifecycle. In short, nothing comes hardwired and there is eminent flexibility. Customers gain the ability to simply use the same industry standard hardware they use for servers, and they don't have to worry about replacing hard-to-find custom components if they fail.

The role of hardware in SDS

A move to software-managed data center services actually bodes well for the overall hardware market, particularly for vendors that provide and build upon commodity hardware devices. Hardware's role in SDS is critical but simple. There are just a couple of rules:

- **Hardware must be interchangeable (agnostic).** What changes from a hardware perspective are any potential dependencies on custom or proprietary hardware. In SDS, there is an assumption that x86-based compute technology will be used to run software-driven storage services (control plane) and the hardware delivering the actual storage capacity (data plane). Storage hardware must remain agnostic for the solution to work. It should be possible to interchange the storage hardware with something else and still have the solution operate, as long as the new hardware doesn't include anything proprietary.

  For example, SDS-based systems, had they existed just a few years ago, would have been built in such a way as to seamlessly accommodate flash memory technologies. With more investment in storage techniques and technologies, there is every reason to believe that further innovations are to come. For example, NVMe (non-volatile memory express) is expected to be broadly available in 2014 from suppliers such as Intel.

- **ASICs and FPGAs are elements of the past.** Custom hardware constructs have no place in SDS and have been replaced by powerful software that replicates the functionality once provided by ASICs and field-programmable gate arrays (FPGAs). Today's mainstream x86 processors are more than up to the task of supporting SDS systems and all their components. In fact, Intel CPUs continue to optimize and accelerate specific tasks with new instruction sets to yield performance on par with specialized ASICs. SHA-1, which is used by some deduplication mechanisms, is now accessible directly via x86 instruction sets.

Scaling and Availability via Software

For all its benefits, implementing SDS isn't easy. The entire modern software-defined movement has come about thanks to companies like Google and Facebook that decided legacy data center paradigms were costly, inefficient, and could not easily scale. Most specifically, these companies looked at the storage layer and made the decision that traditional SAN and NAS devices had no place in their data centers and they dedicated significant engineering resources toward eliminating them from the equation. At the same time, however, these companies needed solutions that had the ability to scale to millions — and potentially billions — of users in a way that didn't create a massive expense sink.

They succeeded in building these environments thanks to a whole lot of ingenuity in hardware platform choice — all commodity all the time — and through the strategic use of software as a scaling mechanism. The lessons learned from these efforts are making their way into enterprise-focused software-defined solutions of which SDS is a component. The primary goal — making SDS independent of proprietary hardware — is already well understood.

However, commodity x86 hardware doesn't natively include all the engineered high availability and redundancy capabilities of proprietary hardware-centric products. In fact, if one were to simply build a storage system with commodity hardware and without a software management layer, the scale shortcomings of the underlying hardware alone would become quickly apparent.

REMEMBER

This is where software comes in. In SDS, the hardware is augmented with powerful software mechanisms that enable these x86 nodes to take part in a scale-out, distributed cluster that can scale in a linear fashion without much in the way of overall limits. In this scale-out storage model, each x86 node contains direct attached hard disks and solid-state storage that can be leveraged by all nodes and all workloads. Further, the scale-out applies to not only storage capacity, but also to storage control logic — to avoid performance bottlenecks when scaling.

In SDS storage clusters, data availability is achieved through software resilience rather than through highly redundant hardware devices. In fact, the software layer is designed for a hardware infrastructure that is expected to have failures, but can immediately detect and respond to unavailability of hardware resources to maintain overall storage SLAs.

For example, rather than using expensive, slow, and increasingly unreliable RAID constructs, SDS systems may choose instead to take a replica-based approach to data protection by storing multiple copies of data at various locations in the cluster. This is exactly the approach taken by companies such as Nutanix in their offerings.

Data locality

As a way to maximize scaling opportunities and provide the best possible performance, whenever possible, Nutanix systems manage data locality as a key part of the scaling strategy. This is accomplished by attempting to keep data local to the source virtual machine node whenever possible. However, as is always the case in virtual environments, workloads can and will move to alternate hosts. In these cases, Nutanix doesn't just blast the network fabric by doing a full-on transfer of a virtual machine's storage blocks from one host to another. Instead, the virtual machine's files are retrieved from the remote node lazily. As the virtual machine makes storage calls, the retrieved blocks are moved to the new virtual machine location while the rest of the virtual machine remains on the original node. Over time, the full contents of the virtual machine will eventually make their way to the new location, but the migration will happen in a natural way and without impact on the network.

All of this scaling is achieved through the implementation of a software layer than manages every aspect of the storage environment.

Virtual Controllers and VM-Aware Software

When you understand that SDS is more than just virtualization (see Chapter 2), you see that the software management layer plays a critical role in any solution that claims to be SDS. The management service must be VM-aware — that is, integrated into the overall software fabric of the data center — to support the myriad workloads and I/O patterns of the virtual data center. Only when storage services are delivered on a per-VM level will software-defined storage be able to provide the promised efficiencies and cost savings.

In addition, SDS must be capable of spawning virtual software controllers so that storage resources (both capacity and control logic) are easily provisioned, agile, and scalable.

Further, with complete awareness of the other virtual elements in the infrastructure, the software layer can decide — based on administrator-defined policies — what storage is needed based on what it sees taking place in the environment. For example, applications that need highly responsive storage can be serviced by flash-based capacity, while less time-critical workloads will tap into slower, but less expensive, hard disk drive-based storage. The abstraction layer offered by SDS and virtualization makes it possible to see every aspect of the environment like never before, which provides automated workload management opportunities.

SDS allows the storage fabric to be adaptive, so that applications with high I/O demands are given access to storage resources with guaranteed SLAs. This ensures that the storage component of the data center is as flexible as other technologies. For example, consider the case of a VDI environment in which all desktops boot at once, resulting in a *boot storm.* A boot storm is a period of extremely high I/O. SDS can detect when a high I/O condition is beginning and automatically move the appropriate data to a higher performance storage tier.

The Emergence of Flash

Traditional hard disks are slow. Even 15K RPM Serial Attached SCSI (SAS) disks can't provide the high performance that is demanded by many of today's workloads, such as virtual desktops, big data analytics, and even basic server virtualization. Over time, the I/O needs for emerging services has far outstripped the I/O possibilities inherent in the hard disk market.

Valiant attempts have been made to squeeze every last ounce of performance out of hard disks. Unfortunately, many of the solutions have ended up being very expensive ways to address storage performance issues. Today, there's a better option: solid-state (flash) storage.

Flash-based storage has been around for quite some time, but until recently, it was too expensive for most organizations to deploy. But as its cost has plummeted, flash storage has become an important part of emerging classes of storage, such as hybrid storage arrays, all flash arrays, and SDS systems.

Flash storage adds to storage a couple of potential performance perks:

- **Fast tier:** With flash storage, administrators enjoy a really fast performance tier within their storage systems. This tier can be reserved for workloads that have particularly high input/output operations per second (IOPS) demands.
- **Caching:** By leveraging solid-state drives for read and write caching, flash storage can accelerate normally slow hard drives. The cache is automatically used to store hot data and metadata.

  Here's how it works: Bear in mind that hard drives prefer a sequential I/O pattern to maximize performance. The caching layer can be used to hold write operations for a period of time. Eventually, the software management layer will take steps to reorder this data — what would have been random writes to the hard drives — into sequential I/O, which is then written to the hard disks. This simple technique significantly speeds up random I/O operations.

Short stroking

One attempt to squeeze more performance out of hard disks is known as short stroking. *Short stroking* is a method by which a hard drive is formatted, but only the outer rings of each platter actually undergo formatting. The rest of the drive is ignored. By formatting only the outer edge of each platter, the heads aren't required to move nearly as much as they are when the full disk is formatted. Of course, as is probably obvious, this method also eliminates the potential to use the capacity from the portions of the disk that are ignored. So while it may bring down cost per IOPS, short stroking greatly increases cost per gigabyte and ends up being a very expensive way to address storage performance issues.

In short, flash storage offers in-system, software-controlled management of storage that simply wasn't available with hard drives alone.

Rapid Failure Recovery

Failure happens. Every component in the data center will eventually fail. It may not happen today; it may not happen tomorrow. But eventually, everything fails. That's not intended to be a really depressing way to start a section, but is intended to ground in reality the fact that no single component is going to work forever.

But a properly designed SDS system is designed to cope with failure when it occurs. This means that the software layer is aware of all hardware and software components and their status at all times, even across a massively scaled, highly distributed system. As the storage service scales, it becomes increasingly important to ensure that the management layer can protect against various kinds of failures.

SDS-based storage systems provide several types of protection and data availability mechanisms:

- **Intelligent data placement:** Data protection begins the second that data is written to a physical disk and acknowledged to the application workload. In an SDS storage system, data placement and protection are critical because no hardware-based RAID mechanism does the work of data protection. In SDS, data placement may happen several times.

  In Nutanix systems, a data placement algorithm ensures that copies of data are spread out among physical nodes and converged appliances so that the loss of a disk, node, or a full appliance doesn't prevent access to data stored in the cluster.

- **Controllers:** In SDS, software-based controllers are responsible for making sure that data is read from and written to disk and remains available for use by applications and virtual machines. Software controllers are often redundant, helping the environment maintain a high level of availability, even in the event of a failure.

- **Software RAID:** Although SDS makes hardware-based RAID systems unnecessary, software-based RAID constructs may be used. To conform to the concepts of SDS, these RAID constructs must be fully supported by software-based controllers and must be able to scale to meet enterprise-class capacity and performance needs.

  In a Nutanix Virtual Computing Platform system, for example, each host runs a virtual machine inside of which runs a software-based storage controller. This storage controller and its place in the overall architecture of the Nutanix cluster helps the entire system work around potential failure since each one is loosely coupled to the others in the cluster. In the unlikely event of a complete failure of the storage management virtual machine, other storage virtual machines can automatically and transparently assume responsibility for data management while the failed controller recovers. The same goes for hardware failure. If an entire hardware node fails, other storage virtual machines can simply pick up where the failed node left off using replicated copies of the data. For the user, the entire process is transparent. For administrators, the process is also transparent, which is one of the primary design goals in software-designed infrastructure.

In order for SDS to be able to ultimately replace big iron storage systems, the ability to seamlessly accept failure and work around it is a critical design element.

The Rise of the SDDC

The overarching goal of software and infrastructure management is the implementation of a full SDDC. As part of this larger paradigm, SDS shares some characteristics with SDDC:

- **Decoupled control and data planes:** The decoupling of the control plane from the data plane (see Chapter 2) enables distributed decision making about what resources are provisioned and how they operate. It also enables the use of less intelligent and more affordable commodity hardware that can bring down the overall cost of the infrastructure.
- **Late binding:** A programming term known as *late binding* refers to the ability of the environment to remain flexible. In early or static binding systems, many components, and their actions and configurations, are fixed in the hardware. In late binding, these items are software delivered and managed as a part of the run-time operation, so they can be tuned for a specific environment. SDS and SDDC leverage this late-binding principle to give the environment as much flexibility as possible.

  This active systems approach to infrastructure assumes that all elements of the data center are dynamic. The data center environment, the needs that are placed on it, and the business itself are constantly changing, forcing changes at every turn. Late binding is the idea that many operations are decided at run time rather than being imposed by the vendor or via strict administrative rules during initial configuration. Creating strict rules during initial configuration means that IT has to stay vigilant to respond to changes that may occur, which is always does. By accepting that chaos is the norm and allowing the active systems and their policies to manage the environment, IT administrators can step back and focus on more value-add business activities.

Chapter 4

How SDS Benefits Business

In This Chapter

- Taking advantage of the cloud
- Making best use of IT resources
- Preparing organizations for the future

Revolutions in IT don't take place just for the benefit of IT. Generally, a technology advancement has to provide some tangible, ongoing business benefit before it becomes a trend. With SDS, several business outcomes and drivers make the technology a clear win for many companies, both large and small.

Cloud-Scale Enterprise Architecture

Even small companies understand that to grow and remain competitive, they must be able to look and act like their larger brethren. Enterprises are finding it necessary to transform their IT departments from internal, reactive structures to cloud-scale service providers that can provide (or contract for) a full range of services at a reasonable cost.

WARNING!

As cloud and service providers approach business leaders with offers to transform their workloads without involving IT, chief information officers (CIOs) must take notice and take action so that they can remain in control of their organizations' technology goals while still providing the kinds of services that the business demands.

Support of IT Goals

Expensive projects designed to improve IT systems are getting extra scrutiny these days. Increasingly skeptical financial officers are questioning the overall value of the services provided by the IT department. CIOs are being pushed to deploy systems that meet ongoing availability, scale, capacity, and performance goals while lowering operational costs.

SDS systems, particularly when they're deployed as a part of a broader software-defined data center (SDDC) initiative, give IT more flexibility to meet new workload needs as they arise. Much of this flexibility comes from eliminating storage as a separately managed resource in the data center. As a separately managed resource, storage systems require expensive personnel with specialized skills. With SDS, these resources can be redirected elsewhere. When they're part of an SDDC, storage resources can be automatically assigned to meet application needs at a per-VM level. The application tier communicates with the SDS control plane in the form of an API. The control plane dynamically assigns the physical storage resources to the requesting application to meet the performance and data protection SLAs needed. Not only does this happen automatically, but if the requirements of the application change in real time, the SDS system can adapt and make the corresponding changes to the storage layer.

In addition, because such solutions use commodity hardware, they help IT organizations reduce storage costs while improving overall performance. In other words, with SDS, both the cost per gigabyte of storage and the cost per input/output operation per second (IOPS) of storage come down, without compromising storage features. Sometimes, in fact, these features (including deduplication, compression, and metadata caching on flash storage) bring down the overall cost of the storage service, because they're delivered via software, rather than expensive proprietary hardware

Another IT trend that SDS can assist is the march toward simplification of IT systems. Over the decades, data centers have become littered with point solutions designed to meet specific needs without really linking to anything else. Silos of infrastructure have been deployed to ensure performance instead of giving control at the VM level.

This works directly against the need to simplify the infrastructure. It also adds additional data center costs in the form of space, power consumption, and cooling load. Under the SDDC umbrella, a hardware device should be able to deliver just about any data center service (or services). Even common resources have become fickle and difficult to manage. CIOs are looking for every opportunity to simplify their data centers, because simplification of the data center ultimately yields better service, reduced costs, and an ability to better focus on the business.

Efficient Use of Existing Resources

The days when corporate IT provided all of an organization's technology services are close to being over, and with good reason. Too many new vendors with compelling solutions have entered the market, allowing organizations to buy third-party services for much less than it would cost to have IT departments provide them. As a result, IT departments need to reduce personnel costs or reallocate technical resources to meet growing business and technology needs. This speaks to another trend that sees today's IT departments transforming into internal service providers and brokers of third-party technology services. Such services are often based in the cloud and may be part of an organization's hybrid cloud efforts.

Even in these "stretched" environments, in which IT is running services both on-premises and in a cloud provider's data center, SDS architectures should be able to leverage off-premises and cloud-based storage as part of a single-data fabric for the enterprise. Just as SDS uses high-speed flash SSD for the most in-demand data sets, SDS can also push archived data to less expensive offsite storage to improve the overall cost equation, but without sacrificing IT priorities. SDS helps IT achieve this goal thanks to its ability to integrate more deeply into the infrastructure, enabling new automation opportunities.

Implementing Private Cloud Functionality

A growing number of CIOs want to harness the benefits of the cloud yet keep resources firmly under the organization's control in the data center. SDS is equipped to support private cloud initiatives — specifically automated and orchestrated data center infrastructures. *Orchestration* is a set of services that provides automation, coordination, and management of complex data center services. SDS systems must be able to support these strategic initiatives by allowing storage to be a cloud-managed resource in a larger orchestrated environment. Some tools on the market — such as OpenStack — can help companies meet the full potential of orchestration.

SDS provides the cloud management stack with detailed information about storage resources being consumed. It answers such questions as:

- How much storage is needed?
- Which storage tier is required?
- What level of performance is being delivered?

The private cloud can then leverage this information to optimize the overall storage environment and charge the business for the resources being delivered.

The bottom line is this: SDS can be an important component of a private cloud environment.

REMEMBER

The goal for modern CIOs is to reduce TCO significantly and to see returns on new investments as quickly as possible. Automation that uses existing resources is a great way to get closer to this goal.

Faster Payback from IT

Server virtualization was the first hurdle that was removed in IT's quest to more rapidly deliver on new initiatives. However, the rest of the data center has remained a bit of a burden in this effort. IT departments must demonstrate rapid time to value on new projects and business initiatives while keeping costs reasonable. Many new initiatives, such as virtual desktops and data analytics, require significant power and capacity, and are dependent on well-designed storage systems. With SDS, IT departments no longer have to painstakingly design, provision, and manage storage to support business needs. The capability to scale in a linear fashion for both capacity and performance ensures that the storage system won't be a roadblock to new endeavors.

Focus on the Future

Business leaders have become accustomed to walking around with computers in their pockets that provide on-demand access to resources and information around the clock. Although IT departments may provide access to some of these information resources, access is also provided by third parties, and the entire experience is shaping executives' perceptions of what technology should be able to do and how it should be able to work. CIOs need to transform their departments to meet these new visions or risk becoming victims of outsourcing to vendors that executives view as being more capable.

TIP

At the same time, IT departments aren't being handed buckets of money to make these transformations. They're forced to continue to work on the 80/20 concept, wherein 80 percent of the budget funds existing operations and 20 percent funds new initiatives. SDS and SDDC can help IT departments take steps toward operating 24/7 without breaking the bank. Software-led infrastructure provides ample opportunities for automation and integration. Such initiatives enable IT to meet business demand without excessive cost and delay, and to prepare their organizations to operate better in the 21st-century marketplace.

Chapter 5

Ten Key SDS Facts

In This Chapter

- Seeing how virtualization and SDS align
- Recognizing the importance of hardware
- Understanding how flash storage makes things better

As IT departments begin to consider software-defined storage, considering the ten key facts in this chapter can help them make better decisions.

Virtualization Is the Basis of SDS. . .

The mainstreaming of virtualization in the early 2000s changed everything. Virtualization — an abstraction of physical hardware resources (compute, memory) that is presented to a software layer running virtual machines (VMs) — resulted in new ways of looking at the data center environment. The real innovations occurred when virtualization was coupled with robust management tools that permitted seamless migration of workloads among servers, delivering greater agility and availability. Workloads are no longer tied to a specific system; the virtualization layer provides a common platform for the underlying hardware and presents it as a whole to the applications above it. Virtualization allows IT administrators to assign resources to individual applications and workloads. Without the abstraction layer (which I discuss in Chapter 2), SDS would still be a pipe dream.

. . .But SDS Is More Than Virtualization

Although virtualization is a critical component of SDS, it's only one part. If SDS were an archery practice target, virtualization would be the innermost circle; the next circle would represent the software services layer that enables the advanced functionality provided by SDS, and so on. Although storage virtualization provides the abstraction that SDS needs to operate, a solution is just virtualized storage without the accompanying management layer. This layer manages advanced workload management services such as deduplication and compression (see Chapter 2). These services need to operate at the VM level to give fine-grained control to the application. In addition, in physical storage environments that have tiered hardware (for instance, high performance versus high capacity resources), the management layer determines which storage tier will run each workload.

Hardware Is Still Critical

It's easy to get lost in the software beauty of SDS and forget that hardware remains an important part of any solution. In SDS, features and product differentiation are delivered via the software layer, which is decoupled from the underlying hardware. This enables the hardware resources layer to be upgraded without affecting the storage services that get delivered to the virtualization host and its associated workloads.

Many vendors that sell what they claim to be commodity-based SDS systems still bundle hardware and software into a single solution. There is a good reason for this. In the real world of the enterprise, ongoing support for the full infrastructure is a must-have. Having the ability to provide end-to-end product support means the difference between ultimate success and failure. Just because a vendor sells hardware along with software, it is vitally important to understand the criticality of the software component.

SDS Must Offer No Compromises

SDS provides improved economics and better overall IT agility. But from a storage perspective, features and services can't be sacrificed.

Practical SDS solutions must still deliver full-featured capabilities including data tiering, snapshots, deduplication, and more. These services are requisite in nearly any data center. The only thing that changes with SDS is that these features get delivered via software instead of proprietary hardware.

SDS Owes a Debt to Cloud

Although SDS and cloud storage aren't necessarily linked, it's fair to assert that the inspiration of SDS traces back to Facebook, Google, and other leading cloud infrastructures. As more users took advantage of these cloud-based services, the companies needed to add more computing power and massive amounts of storage capacity. Traditional x86-based architecture (see Chapter 3) with shared SAN storage couldn't meet these companies' needs in a new era of IT economics, so they created powerful, software-driven architectures that leverage inexpensive internal server storage (flash and HDD), which they presented as aggregated storage for the entire system. As such, the cloud essentially signaled the ultimate demise of centralized, silo-based storage.

Although enterprise, midmarket, and small to medium-sized business customers don't need the scale required by cloud providers, they can still enjoy the benefits generated by those providers' techniques, such as dramatically lowered costs and architectural simplicity — and realize these for nearly any virtualized workload.

Flash Extends SDS

Regular hard disk drives are really good for one thing: storing lots of data inexpensively. To get high-speed storage performance with just traditional drives, however, customers have to buy hundreds of drives that work in parallel. The byproduct of building higher capacity systems just to yield better performance is a system where only a fraction of the capacity is actually used. Sacrificing utilization for better performance is not a winning proposition.

Flash-based storage, however, is orders of magnitude faster than disk drives. And, when integrated smartly into a system, that system can generate impressive overall performance without meaningful cost increases. In SDS, for example, converging flash and hard disk drive storage together allows IT administrators to use the right tier of storage for each application. Together, flash storage and SDS are inexpensive and complementary alternatives to traditional storage arrays.

Decoupled Control and Data Planes Add Flexibility

The decoupling of the control plane from the data plane (see Chapter 2) enables centralized decision-making about how resources operate, as well as efficient delivery of new enterprise-wide services engineered into software. It also enables IT departments to use less-expensive off-the-shelf hardware to bring down the overall cost of the infrastructure. The brains of the storage system are centralized, while the data plane responsible for handling the lower-level data placement is distributed, but remains under the auspices of central intelligence.

x86 Hardware Becomes the Norm

The Intel x86 architecture (see Chapter 3) has become the standard data center platform for today's data centers. It not only provides high performance without the cost of proprietary hardware, but it delivers a historically proven model that significantly increases compute power in compressed product generations. Further, the addition of virtualization-focused *hooks* into the CPU architecture make this low-cost technology just about ubiquitous in enterprise and cloud data centers. As a result, nearly all SDS solutions are built on some kind of x86-based hardware. Its low cost and flexibility provide the obvious platform to meet the needs of today, as well as tomorrow.

SDS Is for Real

It's well known that new technologies go through a hype cycle, which often begins with overinflated claims about both the cost and the benefits of a technology. SDS may stand as a rare exception to the rule, and deliver on many of the vendor promises being made by established and start-up vendors alike.

SDS Improves the Business

A software-defined data center (SDDC), with SDS as a cornerstone, allows businesses to use technologies that are simple to configure, inexpensive to manage, and operationally more efficient. When IT becomes a service that can be rapidly delivered anywhere in the organization, the business can move faster. This is true with storage as well, of course. When storage simply becomes a service — decoupled from the hardware substrate — the business gets the applications and data they want. But with SDS, it's just faster and lower cost than current infrastructures.