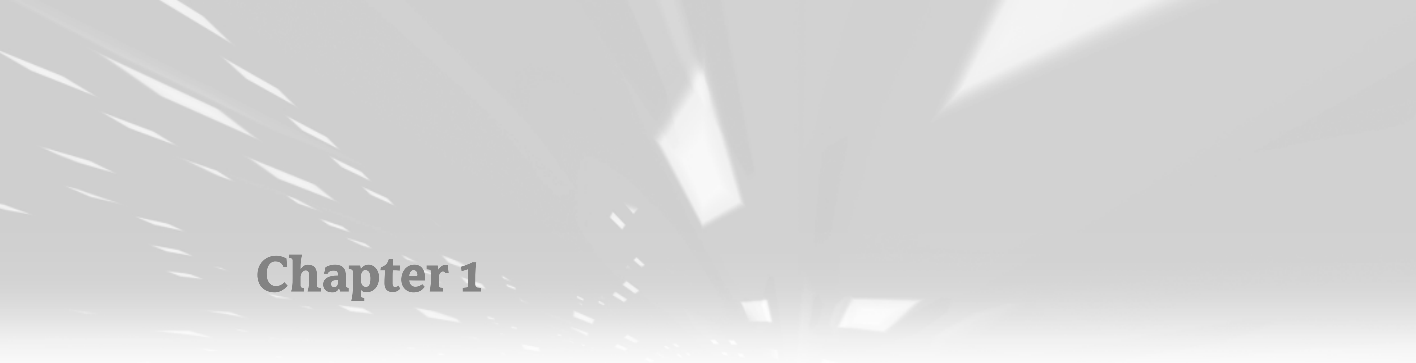
Part 1

The Basics



Chapter 1

Introducing Windows Server 2012 Hyper-V

One thing has remained constant in IT since the invention of the computer: change. Our industry has moved from highly centralized mainframes with distributed terminals, through distributed servers and PCs, and is moving back to a highly centralized model based on virtualization technologies such as Hyper-V. In this chapter, you will look at the shift that has been happening and will learn what has started to happen with cloud computing. That will lead you to Windows Server 2012 Hyper-V.

With the high level and business stuff out of the way, you'll move on to technology, looking at the requirements for Hyper-V, the scalability, and the supported guest operating systems.

You cannot successfully design, implement, manage, or troubleshoot Hyper-V without understanding the underlying architecture. This will help with understanding why you need to install or update some special software in virtual machines, why some features of virtual machines perform better than others, and why some advanced technologies such as Single-Root I/O Virtualization exist.

One subject that all techies love to hate is licensing, but it's an important subject. Correctly licensing virtualization means that you keep the company legal, but it also can save the organization money. Licensing is like a sand dune, constantly changing and moving, but in this chapter you'll look at how it works, no matter what virtualization platform you use.

We cannot pretend that VMware, the company that had uncontested domination of the virtualization market, does not exist. So this chapter presents a quick comparison of their solution and Microsoft's products. This chapter also gives those who are experienced with VMware a quick introduction to Hyper-V.

We wrap up the chapter by talking about some other important things for you to learn. The most important step of the entire project is the assessment; it's almost impossible to be successful without correct sizing and planning. Microsoft makes this possible via the free Microsoft Assessment and Planning Toolkit. One of the most important new features in Windows Server 2012 is PowerShell. This might not be a PowerShell book, but you will see a lot of PowerShell in these pages. We introduce you to PowerShell, explain why you will want to learn it, and show you how to get started.

In this chapter, you'll learn about

- Virtualization and cloud computing
- Hyper-V architecture, requirements, and supported guest operating systems
- Sizing a Hyper-V project and using PowerShell

Virtualization and Cloud Computing

You have to understand where you have come from in order to know where you are going. In this section, you are going to look at how the IT world started in the mainframe era and is now moving toward cloud computing. You'll also learn why this is relevant to Windows Server 2012 Hyper-V.

Computing of the Past: Client/Server

How computing has been done has changed—and in some ways, almost gone full circle—over the past few decades. Huge and expensive mainframes dominated the early days, providing a highly contended compute resource that a relatively small number of people used from dumb terminals. Those mainframes were a single and very expensive point of failure. Their inflexibility and cost became their downfall when the era of client/server computing started.

Cheap PCs that eventually settled mostly on the Windows operating system replaced the green-screen terminal. This gave users a more powerful device that enabled them to run many tasks locally. The lower cost and distributed computing power also enabled every office worker to use a PC, and PCs appeared in lots of unusual places in various forms, such as a touch-screen device on a factory floor, a handheld device that could be sterilized in a hospital, or a toughened and secure laptop in a military forward operating base.

The lower cost of servers allowed a few things to happen. Mainframes require lots of change control and are inflexible because of the risk of mistakes impacting all business operations. A server, or group of servers, typically runs a single application. That meant that a business could be more flexible. Need a new application? Get a new server. Need to upgrade that application? Go ahead, after the prerequisites are there on the server. Servers started to appear in huge numbers, and not just in a central computer room or datacenter. We now had server sprawl across the entire network.

In the mid-1990s, a company called Citrix Systems made famous a technology that went through many names over the years. Whether you called it WinFrame, MetaFrame, or XenApp, we saw the start of a return to the centralized computing environment. Many businesses struggled with managing PCs that were scattered around the WAN/Internet. There were also server applications that preferred the end user to be local, but those users might be located around the city, the country, or even around the world. Citrix introduced server-based computing, whereby users used a software client on a PC or terminal to log in to a shared server to get their own desktop, just as they would on a local PC. The Citrix server or farm was located in a central datacenter beside the application servers. End-user performance for those applications was improved. This technology simplified administration in some ways while complicating it in others (user settings, peripheral devices, and rich content transmission continue to be issues to this day). Over the years, server processor power improved, memory density increased on the motherboard, and more users could log in to a single Citrix server. Meanwhile, using a symbiotic relationship with Citrix, Microsoft introduced us to Terminal Services, which became Remote Desktop Services in Windows Server 2008.

Server-based computing was all the rage in the late 1990s. Many of those end-of-year predictions told us that the era of the PC was dead, and we'd all be logging into Terminal Servers or something similar in the year 2000, assuming that the Y2K (year 2000 programming bug) didn't end the world. Strangely, the world ignored these experts and continued to use the PC because of the local compute power that was more economical, more available, more flexible, and had fewer compatibility issues than datacenter compute power.

Back in the server world, we also started to see several kinds of reactions to server sprawl. Network appliance vendors created technologies to move servers back into a central datacenter, while retaining client software performance and meeting end-user expectations, by enabling better remote working and consolidation. Operating systems and applications also tried to enable centralization. Client/server computing was a reaction to the extreme centralization of the mainframe, but here the industry was fighting to get back to those heady days. Why? There were two big problems:

- There was a lot of duplication with almost identical servers in every branch office, and this increased administrative effort and costs.
- There aren't that many good server administrators, and remote servers were often poorly managed.

Every application required at least one operating system (OS) installation. Every OS required one server. Every server was slow to purchase and install, consumed rack space and power, generated heat (which required more power to cool), and was inflexible (a server hardware failure could disable an application). Making things worse, those administrators with adequate monitoring saw that their servers were hugely underutilized, barely using their CPUs, RAM, disk speed, and network bandwidth. This was an expensive way to continue providing IT services, especially when IT is not a profit-making cost center in most businesses.

Computing of the Recent Past: Virtualization

The stage was set for the return of another old-school concept. Some mainframes and high-end servers had the ability to run multiple operating systems simultaneously by sharing processor power. Virtualization is a technology whereby software will simulate the hardware of individual computers on a single computer (the host). Each of these simulated computers is called a *virtual machine* (also known as a *VM* or *guest*). Each virtual machine has a simulated hardware specification with an allocation of processor, storage, memory, and network that are consumed from the host. The host runs either a few or many virtual machines, and each virtual machine consumes a share of the resources.

A virtual machine is created instead of deploying a physical server. The virtual machine has its own guest OS that is completely isolated from the host. The virtual machine has its own MAC address(es) on the network. The guest OS has its own IPv4 and/or IPv6 address(es). The virtual machine is isolated from the host, having its own security boundary. The only things making it different from the physical server alternative are that it is a simulated machine that cannot be touched, and that it shares the host's resources with other virtual machines.

Host Resources Are Finite

Despite virtualization being around for over a decade, and being a mainstream technology that is considered a CV/résumé must-have, many people still don't understand that a host has finite resources. One unfortunate misunderstanding is the belief that virtual machines will extract processor, memory, network bandwidth, storage capacity/bandwidth out of some parallel underutilized universe.

In reality, every virtual machine consumes capacity from its host. If a virtual machine is using 500 GB of storage, it is taking 500 GB of storage from the host. If a virtual machine is going to use 75 percent of a six-core processor, that machine is going to take that processor resource from the host. Each virtual machine is competing with every other virtual machine for host resources. It is important to understand this, to size hosts adequately for their virtual machines, and to implement management systems that will load-balance virtual machines across hosts.

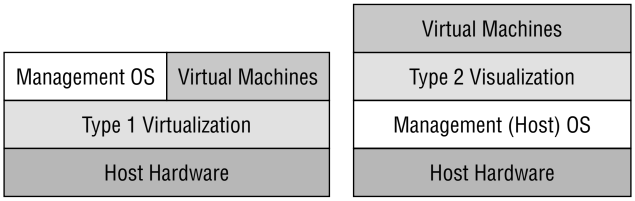
There are two types of virtualization software for machine virtualization, shown in Figure 1.1:

Type 1 Also known as a *hypervisor*, a Type 1 virtualization solution runs directly on the hardware.

Type 2 A Type 2 virtualization solution is installed on an operating system and relies on that operating system to function.

VMware's ESX (and then ESXi, a component of vSphere) is a Type 1 virtualization product. Microsoft's virtual server virtualization solution, Virtual Server, was a Type 2 product, and was installed on top of Windows Server 2003 and Windows Server 2003 R2. Type 2 virtualization did have some limited deployment but was limited in scale and performance and was dependent on its host operating system. Type 1 hypervisors have gone on to be widely deployed because of their superior scalability, performance, and stability. Microsoft released Hyper-V with Windows Server 2008. Hyper-V is a true Type 1 product, even though you do install Windows Server first to enable it.

Figure 1.1
Comparing Type 1 and Type 2 virtualization



The early goal of virtualization was to take all of those underutilized servers and run them as virtual machines on fewer hosts. This would reduce the costs of purchasing, rack space, power, licensing, and cooling. Back in 2007, an ideal goal was to have 10 virtual machines on every host. Few would have considered running database servers, or heavy-duty or critical workloads, on virtual machines. Virtualization was just for lightweight and/or low-importance applications.

The IT world began to get a better understanding of virtualization and started to take advantage of some of its traits. A virtual machine is usually just a collection of files. Simulated hard disks are files that contain a file system, operating system, application installations, and data. Machine configurations are just a few small files. Files are easy to back up. Files are easy to replicate. Files are easy to move. Virtual machines are usually just a few files, and that makes them relatively easy to move from host to host, either with no downtime or as an automated reaction to host failure. Virtualization had much more to offer than cost reduction. It could increase flexibility, and that meant the business had to pay attention to this potential asset:

- Virtual machines can be rapidly deployed as a reaction to requests from the business.
- Services can have previously impossible levels of availability despite preventative maintenance, failure, or resource contention.
- Backup of machines can be made easier because virtual machines are just files (usually).
- Business continuity, or disaster recovery, should be a business issue and not just an IT one; virtualization can make replication of services and data easier than traditional servers because a few files are easier to replicate than a physical installation.

Intel and AMD improved processor power and core densities. Memory manufacturers made bigger DIMMs. Server manufacturers recognized that virtualization was now the norm, and servers should be designed to be hosts instead of following the traditional model of one server equals one OS. Servers also could have more compute power and more memory. Networks started the jump from 1 GbE to 10 GbE. And all this means that hosts could run much more than just 10 lightweight virtual machines.

Businesses want all the benefits of virtualization, particularly flexibility, for all their services. They want to dispense with physical server installations and run as many virtual machines as possible on fewer hosts. This means that hosts are bigger, virtualization is more capable, the 10:1 ratio is considered ancient, and bigger and critical workloads are running as virtual machines when the host hardware and virtualization can live up to the requirements of the services.

Virtualization wasn't just for the server. Technologies such as Remote Desktop Services had proven that a remote user could get a good experience while logging in to a desktop on a server. One of the challenges with that kind of server-based computing was that users were logging in to a shared server, where they ran applications that were provided by the IT department. A failure on a single server could impact dozens of users. Change control procedures could delay responses to requests for help. What some businesses wanted was the isolation and flexibility of the PC combined with the centralization of Remote Desktop Services. This was made possible with virtual desktop infrastructure (VDI). The remote connection client, installed on terminal or PC, connected to a broker when the user started work. The broker would forward the user's connection to a waiting virtual machine (on a host in the datacenter) where they would log in. This virtual machine wasn't running a server guest OS; it was running a desktop OS such as Windows Vista or Windows 7, and that guest OS had all of the user's required applications installed on it. Each user had their own virtual machine and their own independent working environment.

The end-of-year predictions from the analysts declared it the year of VDI, for about five years running. Each year was to be the end of the PC as we switched over to VDI. Some businesses did make a switch, but they tended to be smaller. In reality, the PC continues to dominate, with Remote Desktop Services (now often running as virtual machines) and VDI playing roles to solve specific problems for some users or offices.

Computing of the Present: Cloud Computing

We could argue quite successfully that the smartphone and the tablet computer changed how businesses view IT. Users, managers, and directors bought devices for themselves and learned that they could install apps on their new toys without involving the IT department, which always has something more important to do and is often perceived as slowing business responsiveness to threats and opportunities. OK, IT still has a place; someone has to build services, integrate them, manage networks, guarantee levels of service, secure the environment, and implement regulatory compliance.

What if the business could deploy services in some similar fashion to the app on the smartphone? When we say *the business*, we mean application developers, testers, and managers; no one expects the accountant who struggles with their username every Monday to deploy a complex IT service. With this self-service, the business could deploy services when they need them. This is where cloud computing becomes relevant.

Cloud computing is a term that started to become well-known in 2007. The cloud can confuse, and even scare, those who are unfamiliar with it. Most consider cloud computing to mean outsourcing, a term that sends shivers down the spine of any employee. This is just one way that *the cloud* can be used. The National Institute of Standards and Technology (NIST), an agency of the United States Department of Commerce, published The NIST Definition of Cloud Computing (`http://csrc.nist.gov/publications/nistpubs/800-145/SP800-145.pdf`) that has become generally accepted and is recommended reading.

There are several traits of a cloud:

Self-Service Users can deploy the service when they need it without an intervention by IT.

Broad Network Access There is a wide range of network connectivity for the service.

Resource Pooling There is a centralized and reusable collection of compute power and resources.

Rapid Elasticity There is ample compute power and resources available if more is required, enabling the user to consume resources as required with no long-term commitment.

Measured Service Resource utilization can be measured, and the information can be used for reporting or cross-charging.

Nothing in the traits of a cloud says that cloud computing is outsourcing. In reality, outsourcing is just one deployment model of possible clouds, each of which must have all of the traits of a cloud:

Public A public cloud is one that is run by a service provider in its own facility. The resources are shared by the tenants (customers).

Private A private cloud comes in two forms. It could be a cloud that is run by a service provider but is dedicated to a single customer. Or a private cloud could be one that is run internally by an organization, with absolutely no outsourcing. The private cloud is the ultimate in server centralization.

Hybrid This is where there is a connection between a private cloud and a public cloud, and the user can choose the best location for the new service, which could even be to span both clouds.

Community In a community cloud, numerous organizations work together to combine their compute resources. This will be a rare deployment in private enterprise, but could be useful in collaborative research environments.

Microsoft's Windows Azure and Office 365, Amazon Elastic Compute Cloud (EC2), Google Docs, Salesforce, and even Facebook are all variations of a public cloud. Microsoft also has a private cloud solution that is based on server virtualization (see *Microsoft Private Cloud Computing*, Sybex 2012). These are all very different service models that fall into one of three categories:

Software as a Service A customer can subscribe to a Software as a Service (SaaS) product instead of deploying a service in their datacenter. This gives them rapid access to a new application. Office 365 and Salesforce are examples of SaaS.

Platform as a Service A developer can deploy a database and/or application on a Platform as a Service (PaaS) instead of on a server or a virtual machine's guest OS. This removes the need to manage a guest OS. Facebook is a PaaS for game developers, and Windows Azure offers PaaS.

Infrastructure as a Service Infrastructure as a Service (IaaS) provides machine virtualization through one of the deployment models and complying with the traits of a cloud. This offers a familiar working environment with maximized flexibility and mobility between clouds.

Windows Server 2012 Hyper-V can be used to create the compute resources of an IaaS cloud of any deployment type that complies with the traits of a cloud. To complete the solution, you will have to use System Center 2012 with Service Pack 1, which can also include VMware vSphere and Citrix XenServer as compute resources in the cloud.

Cloud computing has emerged as the preferred way to deploy services in an infrastructure, particularly for medium to large enterprises. This is because those organizations usually have different teams or divisions for managing infrastructure and applications, and the self-service nature of a cloud empowers the application developers or managers to deploy new services as required, while the IT staff manage, improve, and secure the infrastructure.

The cloud might not be for everyone. If the same team is responsible for infrastructure and applications, self-service makes no sense! What they need is automation. Small to medium enterprises may like some aspects of cloud computing such as self-service or resource metering, but the entire solution might be a bit much for the scale of their infrastructure.

Windows Server 2012: Beyond Virtualization

Microsoft was late to the machine virtualization competition when they released Hyper-V with Windows Server 2008. Subsequent versions of Hyper-V were released with Windows Server 2008 R2 and Service Pack 1 for Windows Server 2008 R2. After that, Microsoft spent a year talking to customers (hosting companies, corporations, industry experts, and so on) and planning the next version of Windows. Microsoft wasn't satisfied with having a competitive or even the best virtualization product. Microsoft wanted to take Hyper-V beyond virtualization—and to steal their marketing tag line, they built Windows Server 2012 "from the cloud up."

Microsoft has arguably more experience at running mission-critical and huge clouds than any organization. Hotmail (since the mid-1990s) and Office 365 are SaaS public clouds. Azure started out as a PaaS public cloud but has started to include IaaS as well. Microsoft has been doing cloud computing longer, bigger, and across more services than anyone else. They understood cloud computing a decade before the term was invented. And that gave Microsoft a unique advantage when redesigning Hyper-V to be their strategic foundation of the Microsoft cloud (public, private, and hybrid).

Several strategic areas were targeted with the release of Windows Server 2012 and the newest version of Hyper-V:

Automation A cloud requires automation. Microsoft built their scripting and administration language, PowerShell, into Windows Server 2012. The operating system has over 2,500 cmdlets (pronounced *command-lets*) that manage Windows Server functionality. There are over 160 PowerShell cmdlets for Hyper-V.

Using PowerShell, an administrator can quickly make a configuration to lots of virtual machines. An engineer can put together a script to deploy complex networking on a host. A consultant can write a script to build a cluster. A cloud can use PowerShell to automate complex tasks that enable self-service deployment or configuration.

Networking One of the traits of a cloud is broad network access. This can mean many things to many people. It appears that Microsoft started with a blank sheet with Windows Server 2012 and redeveloped networking for the cloud. Performance was increased, availability was boosted with built-in NIC teaming, the limit of VLAN scalability in the datacenter was eliminated by introducing network virtualization and software-defined networking, partner extensibility was added to the heart of Hyper-V networking, and the boundary of subnets for service mobility was removed.

Storage It became clear to Microsoft that customers and service providers were struggling with storage. It was difficult to manage (a problem for self-service), it was expensive (a major problem for service providers), and customers wanted to make the most of their existing investments.

Some of the advances in networking enabled Microsoft to introduce the file server as a new, supported, economical, scalable, and continuously available platform for storing virtual machines. Industry standards were added to support management of storage and to increase the performance of storage.

Worker Mobility It's one thing to have great services, but they are pretty useless if users cannot access them the way that users want to. Previous releases introduced some new features to Windows Server, but Microsoft didn't rest.

Direct Access is Microsoft's seamless VPN alternative that is not used that much. In Windows Server 2012, the deployment of Direct Access was simplified (to a few mouse clicks in Server Manager), the requirements were reduced (you no longer need IPv6 in the datacenter or Forefront User Access Gateway), and performance was increased at the client end in Windows 8 Enterprise.

Microsoft's VDI solution in Windows Server 2008 R2 was mind-boggling, with many moving pieces in the puzzle. Microsoft simplified the architecture of their VDI to be a scenario wizard in Server Manager. The Remote Desktop Protocol (RDP), the protocol used to connect users to remote desktops such as VDI virtual machines, was improved so much that Microsoft had to rename it RemoteFX. Microsoft has tackled the challenges of peripherals being used on the client, streaming rich media, and quality of service over long-distance connections such as WANs and the Internet.

The Cloud Pretty much every improvement made in Windows Server 2012 Hyper-V plays a role in a public, private, or hybrid cloud. A number of cloud-specific technologies were put in place specifically for cloud deployments, such as Resource Metering. This new feature records the resource utilization of individual virtual machines, giving you one of the NIST traits of a cloud.

We could argue that in the past Microsoft's Hyper-V competed with VMware's ESXi on a price verus required functionality basis. If you license your virtual machines correctly (and that means legally and in the most economical way), Hyper-V is free. Microsoft's enterprise management, automation, and cloud package, System Center, was Microsoft's differentiator, providing an all-in-one, deeply integrated, end-to-end deployment, management, and service-delivery package. The release of Windows Server 2012 Hyper-V is different. This is a release of Hyper-V that is more scalable than the competition, is more flexible than the competition, and does things that the competition cannot do (at the time of writing this book). Being able to compete both on price and functionality and being designed to be a cloud compute resource makes Hyper-V very interesting for the small and medium enterprise (SME), the large enterprise, and the service provider.

Windows Server 2012 Hyper-V

In this section, you will start to look at the technical aspects of Windows Server 2012 Hyper-V.

The Technical Requirements of Hyper-V

The technology requirements of Windows Server 2012 Hyper-V are pretty simple:

Windows Server 2012 Logo To get support from Microsoft, you should ensure that your hardware (including optional components) have been successfully Windows Server 2012 logo tested. You can check with the manufacturer and on the Microsoft Hardware Compatibility List (HCL) for Windows Server (`www.windowsservercatalog.com`).

If you're just going to be testing, the logo isn't a requirement but will be helpful. There is a very strong chance that if your machine will run Windows Server 2008 x64 or Windows Vista (this includes PCs and laptops), it will run Windows Server 2012. You should check with the hardware manufacturer for support.

64-Bit Processor Microsoft is releasing only 64-bit versions of Windows Server, and Hyper-V requires an x64 processor.

32-Bit and 64-Bit Guest Operating Systems

You can run both x86 and x64 guest operating systems in Hyper-V virtual machines.

CPU-Assisted Virtualization The processor must support CPU-assisted virtualization, and this feature must be turned on in the settings of the host machine. Intel refers to this as VT-x, and AMD calls it AMD-V.

Data Execution Prevention In a buffer overrun attack, a hacker writes an instruction into data memory with the deliberate intention of getting the processor to execute malicious code. With Data Execution Protection (DEP) enabled, memory with data is tagged so that it can never be executed by the processor. This prevents the attack from succeeding. DEP must be available in the server's BIOS and must be enabled in the host machine's settings for Hyper-V to install or start up. This protects the inner workings of Hyper-V from malicious attacks by someone who has logged in to a virtual machine on the host. Intel refers to DEP as the XD bit (Execute Disable bit), and AMD calls it the NX bit (No Execute bit). See your hardware manufacturer's documentation for more information. Every server from a major manufacturer should have this support. Usually issues occur only on consumer-grade PCs and laptops.

Second Level Address Translation

There was some confusion when Microsoft announced that the desktop version of Windows, Windows 8 (Pro and Enterprise editions), would support Client Hyper-V. This is the same Hyper-V as on Windows Server 2012, but without server functionality such as clustering, Live Migration, NIC teaming, and so on.

Windows 8 Client Hyper-V is great for administrators who want to use a virtual machine for their administrative functions with a different user account, consultants who want a portable demo environment, or testers/developers who want a local or mobile lab.

Client Hyper-V has all of the same requirements as Windows Server 2012—plus one more, and that's what caused the aforementioned confusion. Second Level Address Translation (SLAT) is required on a Windows 8 computer to enable Hyper-V. SLAT is a processor feature that allows Hyper-V to offload the mapping of virtual machine memory to the host's physical memory. This reduces the pressure on the host's processor and improves virtual machine memory performance. Intel refers to SLAT as Extended Page Tables (EPT), and AMD refers to it as Rapid Virtualization Indexing (RVI), which was previously known as Nested Page Tables (NPT). Outside the server space, SLAT is a relatively new feature. For example, Intel Core Duo processors don't have EPT, but Core i processors (i5 and so on) do support it.

Windows Server 2012 Hyper-V does not require SLAT, despite what some misinformed bloggers might state. Having host processors with SLAT does greatly improve the performance of memory-intensive workloads such as SQL Server or Remote Desktop Services session hosts. Note that SLAT has been around in server processors for quite a while—for example, in Intel Xeon X5500 and later.

The Architecture of Hyper-V

Understanding the architecture of Hyper-V is of great value when you are trying to troubleshoot problems or figure out why or how Microsoft accomplishes certain things.

Figure 1.2 shows the architecture of Hyper-V. A common misunderstanding is that Hyper-V is a Type 2 virtualization product. This is because you must install Windows Server 2012 (and the required drivers) to enable Hyper-V, and some people mistakenly believe that Hyper-V must therefore run on top of the operating system. When you enable Hyper-V, the host will reboot twice. During this process, Hyper-V is slipped underneath the Windows Server 2012 installation to run on the hardware at ring –1 on the processor. At this point, the Windows Server installation becomes known as the *Management OS*. Older terms such as *parent* or *root partition* are no longer used for the Management OS. The kernel of the Management OS is running at ring 0 on the host's processor.

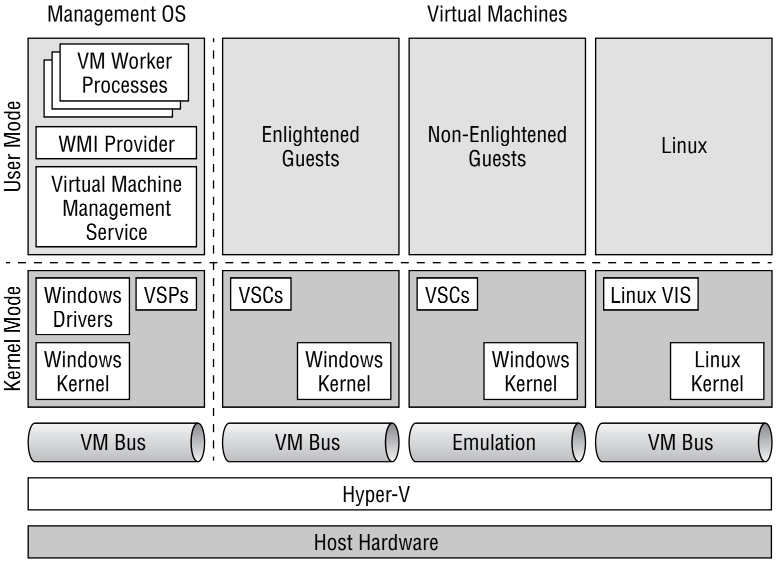


Figure 1.2
The architecture of Hyper-V

In user mode, you can find the Virtual Machine Management Service (VMMS). This process, called `VMMS.EXE`, can be found in Control Panel ➢ Services as Hyper-V Virtual Machine Management. This is the service that manages Hyper-V on this host. The Hyper-V-VMMS logs in Event Viewer are a great place to start troubleshooting a problem on a host. A Windows Management Instrumentation (WMI) provider provides a gateway to the VMMS; this is used by tools such as Hyper-V Manager and agents such as those used by System Center.

There is one worker process for every virtual machine that is running on the host. This worker process is used to manage the virtual machine. When you perform Live Migration on a virtual machine, this is managed by the worker process of that virtual machine. If you enable Dynamic Memory for a virtual machine, the worker process is involved in allocating memory to the virtual machine.

And this leads us to the virtual machines. There are three kinds of virtual machines in the world of Hyper-V:

Enlightened Windows Guests This is a virtual machine with a Windows guest OS that has the Hyper-V integration components installed. The Hyper-V integration components are like drivers. They make the guest OS that is installed in a virtual machine aware that it is running in a Hyper-V virtual machine. The integration components add driver support for virtual devices that Hyper-V can offer, such as the SCSI controller (with support for hot add/removal of storage) or the synthetic network adapter. Additional functionality can also be made possible with the installation of integration components, such as Dynamic Memory for supported guest operating systems.

The integration components are referred to as virtualization service clients (VSCs), which Microsoft documentation also calls virtual service clients and virtualization service consumers. VSCs in the virtual machines (in kernel mode) cooperate with virtualization service providers (VSPs) in kernel mode in the Management OS. This pairing is made possible by a communications channel called the VMBus. There is one VMBus channel for every virtual machine that is running. The VMBus is protected by DEP. This means that if an attacker does successfully gain control of the guest OS of a virtual machine, that attacker cannot send instructions to the Management OS via the VMBus to perform a buffer overrun attack.

Enlightened guests are capable of some additional management functions. You can initiate a clean shutdown of a guest OS from a management tool such as the Hyper-V Manager console. This will initiate a shutdown of the guest OS from inside the virtual machine instead of crudely turning the virtual machine off.

Another feature is Key Value Pairs (KVPs). A KVP allows the guest OS to share information with the Management OS. For example, a heartbeat will let the Management OS know that a virtual machine's guest OS is running. A KVP might even return information such as the guest OS's version or computer name.

Emulated Guests Some organizations need to run the legacy operating systems Windows NT and Windows 2000 Server to keep old applications operating. They might want to upgrade the version of Windows Server but can't because the application vendor has gone out of business. This can be a major problem, because you cannot get hardware to support these operating systems. This means that these installations are potentially great candidates to run as virtual machines where hardware support won't be an issue because the hardware is abstracted by Hyper-V.

The Hyper-V integration components can be installed only in supported guest operating systems. Microsoft does not support legacy operating systems such as Windows NT and Windows 2000 Server. This does not prevent you from running those legacy operating systems in Hyper-V virtual machines. The integration components won't be installed, so we refer to these virtual machines as being emulated. There are no VSCs in these virtual machines. That means there is no support for the virtual SCSI controller, the synthetic network adapter, or Dynamic Memory.

Emulated devices in these virtual machines require extra context switches on the host processor to move between kernel and user mode. This reduces the performance of the virtual machine compared to the potential performance of an equivalent enlightened virtual machine. This is not ideal, but it does allow for legacy operating systems to run as virtual machines on hardware that is still supported by the manufacturer (replacement parts are still available).

Enlightened Linux Guests Years after the first release of Hyper-V, there are still people who say that Hyper-V does not support Linux being installed in virtual machines. This is untrue; several Linux distributions and versions are supported as guest operating systems by Microsoft.

Linux can be enlightened, much like Windows, by Linux Integration Services. Microsoft has developed these integration services to offer almost all of the same functionality for Linux virtual machines, with some exceptions being Dynamic Memory and support for Volume Shadow Copy Service (a Windows service used for application-consistent backups). There is support for multiple processors, the virtual SCSI controller, clean shutdown, guest OS clock synchronization from the host (KVP), and heartbeat (KVP).

At the time of this writing, the latest version of Linux Integration Services was 3.4. You should use your favorite search engine to check whether that's the latest version, and keep it updated to ensure the best performance and feature availability.

For Microsoft, the word *support* is important. Some companies use the word *support* to say that something works. For Microsoft to support something means that they need to be able to do engineering or have a relationship with another company that can do that engineering. Microsoft supports several Linux distributions and versions. However, Linux Integration Services works very well in a wide range of Linux distributions without support from Microsoft.

Microsoft did something very interesting with Linux Integration Services. The Hyper-V Linux Integration Services are built into the Linux kernel in version 3.3 and later. That means that any Linux distribution/edition that has Linux kernel v3.3 or later has Linux Integration Services for Hyper-V already installed.

Maximum Scalability

One of the goals of virtualization is to increase the density of virtual machines that are running on a host. For larger organizations, this can mean that they have fewer hosts. This reduces space consumption, and the costs incurred by power, cooling, licensing, rack space, support, and so on. Hardware capabilities have increased significantly in recent years, and Moore's law would have us believe that this will only continue. Microsoft did a lot of engineering to increase the maximum specifications of a Windows Server 2012 Hyper-V host, as you can see in Table 1.1.

A logical processor is a thread of execution. A 10-core processor that doesn't have Hyper-Threading available or enabled has 10 threads of execution and therefore has 10 logical processors. If you were to enable Hyper-Threading on that processor, you would have 20 threads (10 cores, each with two threads) or logical processors. A single Windows Server 2012 Hyper-V host can support up to 320 logical processors. That could be configured as follows:

- Sixteen 10-core processors with Hyper-Threading enabled
- Thirty-two 10-core processors with no Hyper-Threading

My Logical Processors Are Missing!

If you are lucky enough to have a host with more than 64 logical processors, you might panic if you start Task Manager in the Management OS and can see only 64 logical processors. This is because the Management OS can see a maximum of 64 logical processors. All of the logical processors in the host are available to the virtual machines on that host, so don't worry.

This isn't a problem for the Management OS either. If your Management OS needs to use 64 logical processors, then you have a problem. The Management OS should be nothing but a Management OS for Hyper-V. Don't make it a domain controller, don't install SQL Server, don't turn it into a Remote Desktop Services session host; Microsoft's support services won't like it. Instead, install those applications in virtual machines that are running on the host.

A single host can also have up to 4 TB of RAM. A single host can also support up to 1,024 running virtual machines, and up to 2,048 virtual processors in virtual machines. That is a lot of memory and compute power that you can allocate.

For almost everyone, these numbers are pretty crazy. Most of us will not see a server of this scale in the next few years. But never say never; who wants to be remembered for saying something like 640 KB of RAM will always be enough for people! These numbers do have a real impact. If you make the decision to deploy Windows Server 2012 Hyper-V, then you know that it will scale, probably more than your hosts will ever need to.

Table 1.1: Maximum host scalability

Item	Maximum Number
Logical processors per host	320
Physical RAM in a single host	4 TB
Running virtual machines on a single host	1,024
Virtual processors running on a single host	2,048

Does this mean that you should run your entire business on a few gigantic hosts? No; there is a trade-off. You should try to consolidate and reduce your host numbers, but balance this with the risk of having too many eggs in one basket. Failover clusters, if deployed, will give us high availability, but they give us minimized downtime and not zero downtime in the event of host failure.

Prior to the release of Windows Server 2012, a Hyper-V virtual machine was limited to four virtual processors and 64 GB RAM, assuming that the host had four logical processors and 64 GB RAM to provide. That did limit the sorts of workloads that you could run as virtual machines, and left organizations with some applications running physical servers for performance reasons.

Windows Server 2012 aims to make it possible to virtualize as many workloads as possible. To accomplish this, Microsoft has hugely increased the maximum specifications of a virtual machine, as shown in Table 1.2. A virtual machine can now scale out to 64 virtual CPUs and 1 TB RAM. That should make it possible for just about every workload to run in a virtual machine on Windows Server 2012, assuming that the guest OS supports this scalability (it varies). Now the “we didn’t virtualize it because it was too resource hungry” excuse cannot be used. Size your hosts (including networking and storage) correctly, virtualize the workloads, and take advantage of the hardware abstraction and flexibility that Hyper-V can provide.

Table 1.2: Maximum virtual machine scalability

Item	Maximum Number
Virtual processors per virtual machine	64
Maximum RAM per virtual machine	1 TB

An example of a massive workload is online transaction processing (OLTP), a database service that runs at the back end of a massive web application. Microsoft has blogged about the results of a test comparing a virtual machine that has 64 virtual CPUs with an equivalent physical installation (`http://blogs.technet.com/b/server-cloud/archive/2012/11/08/windows-server-2012-hyper-v-delivers-on-scalability-and-performance-for-virtualized-enterprise-applications.aspx`). With a database supporting 75,000 simulated customers in a stock broker scenario, the virtual machine had nearly 94 percent of the transaction-processing capacity of the physical alternative. That is impressive, because it enables an organization to virtualize these workloads onto fewer physical servers (hosts) and obtain the benefits of virtualization such as cost reduction, better flexibility, easier backup, and more-reliable disaster recovery. In addition, migrating the workloads will be much easier when the hardware becomes obsolete.

Failover Clustering gives us high availability, a feature that caters for unplanned host failure. The virtual machines on the host will stop because the host has stopped, but they will automatically fail over to another host in the same cluster and start up without any human intervention. Some further improvements even allow you to order this process to reflect application or server dependencies. Microsoft has increased the scalability of a cluster of Hyper-V hosts. Table 1.3 shows the cluster scalability of Windows Server 2012 Hyper-V.

A Windows Server 2012 Hyper-V cluster can have 64 nodes (what a physical server is called in a cluster). The cluster may also scale out to contain 8,000 highly available virtual machines.

Note that you must always stay within the limits of a single host while scaling out the cluster, and vice versa. For example, you could have eight hosts, each running 1,000 virtual machines. You cannot scale that cluster out any more because you have reached the virtual machines/cluster limit of 8,000.

This increase to 64 nodes and 8,000 virtual machines enables very large infrastructures, such as those in a public cloud (hosting companies) or huge enterprises, to have fewer clusters and therefore have fewer units of administration. A cluster can start small, and whenever monitoring detects that resource contention is approaching, operators can keep adding hosts, and administrators can add them to the storage and cluster.

Table 1.3: Maximum virtual machine scalability

Item	Maximum Number
Maximum hosts in a single cluster	64
Maximum number of virtual machines in a single cluster	8,000

Beware of Older Information

Microsoft started making announcements about the scalability of Windows Server 2012 (previously Windows Server 8) in 2011. The figures increased with every public test release, right up to the announcement of general availability in August 2012. You might find many old blog posts and documents containing information that has become obsolete.

Supported Guest Operating Systems

We could consume several pages of text in this book telling you which guest operating systems are supported by Microsoft. But that would be foolish. The list of guest operating systems is fluid, changing (increasing) over time—particularly in the case of the supported Linux guest operating systems, which has gone from including just SUSE Linux Enterprise Server to all of the commonly used distributions. The latest version (at the time of this writing) could be found at `http://technet.microsoft.com/library/hh831531`.

In the case of Windows Server 2008 R2 Hyper-V, we found that the list of supported guest operating systems on TechNet was not updated to match the actual list of guest operating systems that Microsoft supported. The updated list was published on the TechNet Wiki site.

There are some things to note:

Integration Components/Services The latest version of the Window Hyper-V integration components, as they were at the time, were included in Windows 8 and Windows Server 2012. The integration components for Windows guest operating systems get updated from time to time, sometimes by service packs and sometimes by hotfixes. You might need to update the integration components in the guest OS to the latest version. Note that a Windows update released in November 2012 required an upgrade of the integration components of virtual machines with a guest OS of either Windows 8 or Window Server 2012.

Maximum Number of Virtual Processors A virtual machine will have from 1 to 64 virtual processors. Each virtual processor gives the virtual machine access to a logical processor on the host for increased parallel processing. Some guest operating systems support a smaller number of virtual processors. In the past, some have deliberately exceeded the maximum number of virtual processors for a given guest OS and have usually experienced guest OS crashes very quickly. Remember, these are referred to as maximum supported numbers for a reason; don't expect help from Microsoft other than "reduce the number of virtual processors to a supported number."

Converting Machines to Hyper-V Virtual Machines

Be careful if you are converting a physical machine with lots of cores that is running a legacy operating system. The conversion tool will probably create a virtual processor for every core, and the resulting number of virtual processors might exceed the maximum number of virtual processors for the guest OS.

Linux Is Supported Microsoft does support the major distributions of Linux as guest operating systems on Windows Server 2012 Hyper-V.

Licensing Windows Server 2012 in Virtualization

Stop! Don't turn the page to the next topic! The natural reaction of a techie is to tune out when the topic of licensing is brought up, but the techie plays an important role in getting licensing right, because some technical input is required. Get it right, and not only do you stay legal, but you also save money, and that's more budget for doing more cool things. That should be a good enough reason to keep reading.

Always Ask a Qualified Expert for Licensing Advice

This book does not provide licensing expertise. The information contained within this section is intended to introduce you to the topic of Windows Server licensing in virtualization. If you need advice about a specific scenario you are working on, you should contact your reseller (if you are a small/medium customer), your Large Account Reseller (if you are a large customer), or your distributor (if you are a reseller).

The information in this section is subject to change, so you should always check to see that you have the latest information.

Common Misunderstandings in Licensing

People commonly get a few things wrong in terms of licensing Windows Server virtual machines. Here are the corrections to the most common misconceptions:

There Is No Such Thing as Hyper-V Licensing Some people think that you have to buy licensing for Hyper-V. Actually, there is no Hyper-V licensing. Large Account Resellers (LARs), distributors, and Microsoft all make exactly $0 from Hyper-V. We can question why Microsoft has made so much effort with Hyper-V, but it is probably to protect the presence of the Windows Server brand in the datacenter and to encourage more System Center sales.

Microsoft licenses Windows Server for virtualization. You will purchase the same Windows Server licensing no matter what virtualization product you install on the host.

You Never Assign Licenses to Virtual Machines Some organizations have gotten into a habit of purchasing one copy of Windows Server for every virtual machine that is created with that guest OS. In their heads, these people think that they have assigned a license to the virtual machine. In fact, according to the licensing rules of Windows Server, they have assigned the license to the host and taken advantage of the virtualization benefits of Windows Server to license the virtual machine.

Licensing Is Mobile but Only under Certain Conditions There are two general types of Windows Server license that you can purchase. An original equipment manufacturer (OEM) license (`www.microsoft.com/oem/en/licensing/sblicensing`) is tied to a physical machine when you purchase it. The license cannot be reassigned to another server. That license dies when the server is disposed of. That's why OEM licenses are the least expensive way to license Window Server. You can add mobility to an OEM license by attaching Software Assurance (`www.microsoft.com/Licensing/software-assurance/Default.aspx`), which enables you to move the license (as well as a number of other benefits). Then, if you dispose of the physical machine originally attached to the OEM license, the license can survive.

A volume license, or VL (`www.microsoft.com/licensing/`), is one that is bought as part of a Microsoft licensing contract such as Open, Open Value Subscription, Select, or Enterprise Agreement. These programs provide easier licensing management through a portal, and easier operating system deployment through a single-product Multiple Activation Key (MAK) or automated product activation using Key Management Services (KMS), as well as other benefits that a licensing expert can brief you on. A volume license that is assigned to a physical machine can be moved to another physical machine, but cannot be moved again for 90 days.

MSDN, TechNet, and SPLA Licensing

MSDN (`http://msdn.microsoft.com/subscriptions/cc150618.aspx`) subscriptions are intended to give the licensed person rights to use Microsoft software to design, develop, test, or demonstrate his or her programs on any number of devices.

A TechNet (`http://technet.microsoft.com/subscriptions/ms772427.aspx`) subscription is intended to give IT pros easy access to Microsoft software for evaluation purposes only.

MSDN and TechNet licensing cannot be used for production systems. This is a mistake that is commonly made by Microsoft partners; those partners can get cheap access to Microsoft software for internal production usage through the Partner Action Pack (`https://partner.microsoft.com/40016455`) or free access by achieving one or more partner competencies (`www.microsoft.com/partner/licensingcalculator/default.aspx`).

We do not cover Microsoft's licensing program for hosting companies in this chapter. Services Provider License Agreement (SPLA) is a fluid program in which the only constant is change. SPLA has some unique rules that are very strange when compared to other production licensing programs. Check with your SPLA LAR or distributor for details on virtualization licensing.

Windows Server 2012 Licensing

The licensing for Windows Server had remained relatively unchanged for many years. Microsoft decided to simplify the licensing and align it with that of System Center with the release of Windows Server 2012.

In the past, there were three core editions (nothing to do with the Core Installation type) of Windows Server:

Standard was intended for basic physical servers or small businesses.

Enterprise was used for physical workload high availability (Failover Clustering) and light virtualization.

Datacenter was to be used in dense virtualization.

Table 1.4 compares these three core editions of Windows Server before Windows Server 2012. Some notes about the table's details:

The Indicated Cost The costs shown are Open NL, the most expensive Microsoft volume licensing program, and were correct at the time of writing this book. The U.S. dollar price is used for easy comparison.

Licensing Basis The Standard and Enterprise editions were licensed per server. The Datacenter edition was licensed per processor, and you had to purchase at least two Datacenter licenses, even if the host had only one processor.

Features and Scalability All of the features and scalability of Windows Server were included in the Enterprise and Datacenter editions. The Standard edition supported a smaller number of processors and less RAM and did not have all of the features such as Failover Clustering and Live Migration.

Virtualization Rights A Virtual Operating System Environment (VOSE) is a licensing term for a guest OS, the OS that you install in a virtual machine. You license physical machines for Windows Server. You license virtual machines based on the virtualization rights of the host.

For example, if you purchased VMware vSphere for a host, and wanted to run four virtual machines that would have Windows Server as their guest OS, you would have licensed the host for Windows Server Enterprise and used the host's virtualization rights. This kept the licensing simpler (you license a single host instead of multiple servers) and it was cheaper (one copy of Enterprise at $2,358 was $546 cheaper than four copies of Standard at $726 each). The customer had the choice of installing Windows Server Enterprise on the host too, but they were using VMware vSphere, which was an additional cost, so vSphere was installed on the host instead of Window Server, and Windows Server was installed in the four virtual machines.

TABLE 1.4: Core Windows Server 2008 R2 editions

EDITION	LICENSE BASED ON	FEATURES	SCALABILITY	VIRTUALIZATION RIGHTS	COST
Standard	Per server	Limited	Limited	One free VOSE	$726
Enterprise	Per server	All	All	Four free VOSEs	$2,358
Datacenter	Per processor, minimum of two licenses per server	All	All	Unlimited free VOSEs	$4,809

Note that the indicated Datacenter cost is for two licenses.

PER PROCESSOR IS PER SOCKET IN MICROSOFT LICENSING

In Microsoft licensing, when you see the phrase *per processor*, it means *per socket*. It does not mean that you license each core or logical processor within a CPU. For example, a server with two 10-core processors has two processors or sockets. This server would be licensed for two processors.

Microsoft decided to simplify Windows Server licensing with the release of Windows Server 2012, as you can see in Table 1.5. There are a number of changes:

Windows Server 2012 Standard Edition The Standard edition of Windows Server 2012 now has all of the same features and scalability of the Datacenter edition. The only difference is the number of free VOSEs, or free installations of Windows Server, that are deployed on a licensed host.

Windows Server 2012 Standard has gone from one free guest OS installation of Windows Server to two.

The price of Windows Server 2012 did increase slightly, but this comes with the ability to create a Failover Cluster, and with double the number of VOSEs.

You now have to count physical processors (sockets, not cores or logical processors) when licensing a server for Windows Server 2012. The Standard license covers two processors or sockets. If a server will have four processors, you must purchase two copies of Windows Server 2012 Standard for it; there is no double-install or need to enter two product keys because this is simply an accounting operation.

Windows Server 2012 Standard is now the edition intended for use in light virtualization with optional high availability, such as a large host with a few virtual machines, a host in a small business, or for physical application clusters.

There Is No Enterprise Edition of Windows Server 2012 You can no longer purchase an Enterprise edition of Windows Server. There were two reasons to have the Enterprise edition in the past. The first was to take advantage of features (such as Failover Clustering) or scalability (processors and memory) that the Standard edition did not have. Windows Server 2012 Standard now has those features and scalability at a fraction of the cost of the Enterprise edition.

The other reason was light virtualization. The Enterprise edition entitled you to four free VOSEs on a licensed host. Now if you want four free guest OS installations of Windows Server 2012 Standard, you can purchase two copies of Windows Server 2012 Standard for that host. This stacks the virtualization rights (two copies, each with two free VOSEs) and gives you the same solution. The Enterprise edition of Windows Server 2008 R2 was $2,358, which made each VOSE cost $589.50. Two copies of Windows Server 2012 Standard will cost $1,764, which means each VOSE will cost $441, saving you $148.50 per guest OS installation. You read that right: You can pay less for a Microsoft solution! This will be especially good news for those folks who use a subscription-based volume licensing scheme such as Open Value Subscription.

Windows Server 2012 Datacenter Edition The Datacenter edition has gone from being a per-single-processor license with a minimum purchase of two per host, to a per-processor license that covers two processors.

Table 1.5: Core Windows Server 2012 editions

Edition	License Based On	Features	Scalability	Virtualization Rights	Cost
Standard	Per processor, each license covers two processors	All	All	Two free VOSEs	$882
Datacenter	Per processor, each license covers two processors	All	All	Unlimited free VOSEs	$4,809

VOSEs Guest OS Installations

The free virtualization rights of Windows Server enable you to install Windows Server (not the desktop OS) on the host that you are licensing. This includes downgrade rights; for example, you can purchase Windows Server 2012 for a host and deploy Windows Server 2008 R2 virtual machines on it.

The VOSE rights are nontransferrable. This means that if you license a host, you cannot use its VOSE rights on another host. You must license that host for the maximum possible number of virtual machines that will run Windows Server, even if those virtual machines are on that host for just 1 second.

The free virtualization rights are not a feature limit; they simply limit how many virtual machines you can install Windows Server into. For example, you could license a host with a single Windows Server 2012 Standard edition, and legitimately deploy 500 Linux virtual machines on it. How you license your Linux guest operating systems is between your employer/customer and the Linux publishers.

The unlimited virtualization rights of the Datacenter edition really are unlimited. There is no "fair usage" clause. Microsoft really is encouraging you to create as many virtual machines on a licensed host as your host and virtualization solution can support.

Hyper-V Server 2012

As usual, Microsoft has released a version of Windows Server that is dedicated to being used as a Hyper-V host. Hyper-V Server 2012 has all of the same Hyper-V and Failover Clustering features and functionality as the Standard and Datacenter editions of Windows Server 2012. There are some major differences:

Virtualization Rights There are no free virtualization rights for installing Windows Server as a guest OS with Hyper-V Server 2012. At first, you might think that this makes Hyper-V Server 2012 seem pretty useless, but that is not the case. There are a number of uses for this free product:

- A customer might have licensed their Hyper-V hosts using Windows Server 2008 R2 and have no Software Assurance (a benefit of which is free upgrades) or budget to afford new licensing. They can upgrade the hosts to Hyper-V Server 2012 to take advantage of the new features while continuing to use their Windows Server 2008 R2 licensing for their virtual machines' guest operating systems.
- A student might want to use Hyper-V Server on a machine to run virtual machines with time-limited trial editions to learn Microsoft's products and pass certification exams.
- A company might want to designate one or more hosts to run only virtual machines with Linux guest operating systems.
- The virtualization rights of Windows Server would be irrelevant to a VDI deployment, so Hyper-V Server would be suitable for the host.

Administration Hyper-V Server 2012 does not have a GUI; it is very similar to a Core installation of Windows Server 2012. You are given a basic text wizard to configure the host, but Hyper-V Server 2012 is supposed to be remotely managed and configured using PowerShell, the Remote Server Administration Toolkit (RSAT) for Windows Server 2012 for Windows 8, or enterprise management suites such as System Center 2012 with Service Pack 1.

Virtualization Scenarios

This section describes a few common virtualization scenarios to demonstrate how to use Windows Server 2012 licensing.

Single Host with Light Virtualization

You have a host with a single processor. You want to deploy three virtual machines on this host. One of the virtual machines will run Windows Server 2012, a second will run Windows Server 2008, and a third will run CentOS 6.3 (a Linux distribution).

This host can be licensed with a single copy of Windows Server 2012. The Linux distribution is irrelevant to Windows licensing, and there is no licensing cost to creating virtual machines. You simply license the host for the number of virtual machines that will run Windows Server. There will be two virtual machines with Windows Server as their guest OS. Windows Server 2012 Standard edition will cover two VOSEs, including downgrade rights for the guest operating systems.

What would happen if you wanted to have four 8-core processors with Hyper-Threading into that host? That's 64 logical processors. In that case, you will not need 32 copies of Windows Server 2012 Standard; you have four processors, and you'll need two copies of Windows Server 2012 Standard.

Maybe you want to have a host with six virtual machines running Windows Server. If that's the case, you don't need to spend $5,292 on six copies of Windows Server 2012 Standard. You can stack virtualization rights by assigning three copies of Windows Server Standard to the host, each giving you two VOSEs, and resulting in a total of six VOSEs. This will cost $2,646; licensing per host will save you money.

What Will You Do with These Savings?

As you might guess, Microsoft has a few suggestions on how to spend the money that you save by efficiently licensing your Windows Server virtual machines.

The first way to spend your budget is to buy System Center for each of your hosts to improve your end-to-end management and build a Microsoft public or private cloud. There are three ways to do that:

- Buy the System Center Server Management License (SML), licensing all of the System Center components the exact same way as Windows Server 2012.

- For anyone buying fewer than 25 hosts, you can purchase the Core Infrastructure Suite (CIS). This comes in Standard or Datacenter editions, and includes both Windows Server and the System Center SML for a host and virtual machines on the host, at a small discount.
- If you are buying 25 or more hosts, you can purchase Enrollment for Core Infrastructure (ECI). ECI has identical licensing to CIS but is intended for larger enterprises and has a much bigger discount.

An alternative is to attach Software Assurance (SA) to your host licensing if the license program doesn't already include it. This add-on will entitle you to upgrade your hosts or virtual machines to a newer version of Windows Server when it is released, as well as including other benefits. Many might question the worth of SA, but it does offer some value. Imagine a time in a couple of years when you acquire a new server application that requires Windows Server "vNext." If your hosts don't have Software Assurance, you'll either have to get a new host or a new physical server, or acquire entirely new licensing for the hosts.

Software Assurance has another benefit called Cold Backups for Offsite Recovery. If you replicate virtual machines with a Windows Server guest OS to a disaster recovery (DR) site, those replica virtual machines in the DR site require Windows Server licenses, even if they are powered off. You could use Hyper-V Server 2012 for the DR site (reducing your costs) and take advantage of this benefit. In this case, you would spend less money on SA than you would spend on Windows Server licensing for the DR site hosts and have access to the many other SA benefits.

Single Host with Dense Virtualization

Imagine that you have a host with two processors, and you want to run 12 virtual machines on it with Windows Server as their guest OS. You could legally license this host with six copies of Windows Server Standard, costing $5,292.

Although that option would be legal, it would not be the most cost-effective way to license the host and virtual machines. Instead, you can buy a single copy of Windows Server 2012 Datacenter edition, costing $4,809. This will save you $483 and gives you the flexibility to deploy as many Windows Server virtual machines as your host and virtualization platform will allow.

What if you were going to license a host that has 16 10-core processors with Hyper-Threading enabled (320 logical processors), has 4 TB of RAM, and will run 1,024 virtual machines?
This host has 16 processors and therefore will require eight copies of Windows Server 2012 Datacenter edition. This will cost $38,472. That might seem wildly expensive, but it's a lot less than the $451,584 that 512 copies of Windows Server 2012 Standard would cost to license 1,024 VOSEs. If only we all had to work with this size of host and had this problem!

A Small Virtualization Cluster

In this example, we have a small, two-node virtualization cluster with a total of eight virtual machines. Each host will have one or two processors. This is a pretty common deployment in the SME space.

The wrong licensing solution would be as follows: You assume that you could get away with placing four virtual machines on each host. You would purchase two copies of Windows Server 2012 Standard for each host (a total of four licenses) to cover four VOSEs on each host. This should cover any temporary host maintenance or failover scenarios. This solution is not legitimate because it breaks the mobility rules of Windows Server licensing; moving or failing over the four virtual machines from host A to host B would require a license move. OEM licenses cannot move, and volume licenses can move only once every 90 days.

To license this two-node cluster correctly, you need to license each host for the maximum number of virtual machines that could reside on the hosts, even for 1 second. There are eight virtual machines, and therefore you need to license each host for up to eight virtual machines—that is, four copies of Windows Server Standard per host, with a total of eight copies of Window Server Standard for the cluster.

When consulted, a licensing expert should ask whether this company will be planning growth in the near future. If this could result in the company having 12 or more virtual machines, the customer should purchase one Windows Server 2012 Datacenter edition per host. This will be less expensive than stacking copies of Windows Server 2012 Standard edition and will give the customer the flexibility to deploy as many virtual machines as needed without having to purchase additional Windows Server licensing.

By the way, this is very similar to licensing nonclustered hosts that allow virtual machines to move between them with Live Migration.

Hyper-V Licensing Is Free

We have not mentioned Hyper-V since we started talking about licensing and mentioned that there is no such thing as Hyper-V licensing. You license your hosts for the number of virtual machines that will license Windows Server. If you've read the facts and studied the examples, you'll see that the following are true:

- Licensing virtual machines is much cheaper than licensing physical servers.
- There is no cost to installing Windows Server 2012 on a host. This is effectively free because you've paid for the virtual machines with per-host licensing and you have the right to install your host-assigned license on the host if you choose to.

If you don't choose Hyper-V as your virtualization platform, odds are that you'll go with vSphere. You will be licensing your Windows Server virtual machines as documented previously, no matter what virtualization you'll use. You could have elected to use the host licensing to install Windows Server on the host and enable Hyper-V, but instead, you'll have elected to spend a lot more money on an alternative virtualization solution. We'll leave the rest of the VMware vs. Microsoft debate to social media and marketing people. However, we do need to talk a little bit more about VMware.

VMware

Even the most ardent Hyper-V fan has to give VMware a lot of credit. Machine virtualization was driven by VMware, and they have forced Microsoft to create an excellent virtualization solution in Windows Server 2012 Hyper-V. The credible and competitive Windows Server 2012

Hyper-V also forced VMware to reverse an unpopular licensing change and to introduce previously unannounced features in vSphere 5.1.

Migrating from VMware

It is possible to convert VMware virtual machines into virtual machines that you can run on Windows Server 2012 Hyper-V. This is known as virtual-to-virtual (V2V) conversion. There are several ways to accomplish this:

Microsoft Virtual Machine Converter Accelerator Accelerators are free products that are released by Microsoft to assist you with specific administration tasks. The Microsoft Virtual Machine Converter (MVMC) Solution Accelerator (`http://technet.microsoft.com/library/hh967435.aspx`) converts VMware virtual machines into Windows Server 2012 Hyper-V virtual machines. It supports vSphere 4.0, 4.1, and 5.0 hosts that are managed by vCenter 4.1 or 5.0. This tool is tiny (just over 4 MB) and will not only convert the virtual machine but will also remove the VMware tools and install the Hyper-V integration components. It is limited to supporting Windows 7, Windows Vista, Windows Server 2008 R2, Windows Server 2008, Windows Server 2003 R2 with SP2, and Windows Server 2003 with SP2 guest operating systems.

System Center 2012 with Service Pack 1 System Center Virtual Machine Manager (VMM) has the ability to take control of vSphere installations and convert VMware virtual machines into Hyper-V virtual machines. The solution by itself is not as powerful as the MVMC. However, MVMC can be run using PowerShell. This means that you can use MVMC as part of a bigger solution, such as a System Center Orchestrator run book that could be used in a repeated, automated, and audited fashion to convert lots of virtual machines in a larger project.

Third-Party Solutions You could use a third-party solution, such as the free V2V Easy Converter by 5nine (`www.5nine.com/vmware-hyper-v-v2v-conversion-free.aspx`). The 5nine solution has a nice feature: It supports guests that run Windows as well as Ubuntu and CentOS, two of the most common free Linux server distributions.

Virtual Machine Migration Toolkit The Virtual Machine Migration Toolkit (VMMT) is a program that is the result of a partnership between Microsoft and Veeam Software, and is available only to specially selected Microsoft partners in a few countries. This solution uses Microsoft and Veeam products to automate the conversion of lots (perhaps thousands) of vSphere virtual machines into Hyper-V virtual machines.

Transferring Skills to Hyper-V

Although there are many similarities between ESXi or vSphere and Hyper-V, there are also many differences. There is no right way. VMware chose to do things one way, and Microsoft had the opportunity to observe and to do some things very differently. This can cause a learning curve for VMware product veterans who are trying to adapt to Hyper-V.

The biggest difference between the two platforms is the hypervisor architecture. VMware's hypervisor is referred to as a *monolithic hypervisor*. This Type 1 virtualization is bigger than Hyper-V and contains the drivers. This means that VMware writes the drivers and has a smaller HCL, even though this still includes the typical tier 1 servers that you will likely purchase.

You've already seen that Hyper-V places the drivers in kernel mode in the Management OS. This Type 1 virtualization is known as a *paravirtualized hypervisor*.

vSphere uses a file system called the Virtual Machine File System (VMFS). VMware allows you to extend VMFS volumes by using extents, but they do not recommend it unless you have no other choice (`http://blogs.vmware.com/vsphere/2012/02/vmfs-extents-are-they-bad-or-simply-misunderstood.html`). Meanwhile, Microsoft uses NTFS (with a new file system called ReFS being added) and has no such concerns with extending a file system. VMware uses the file system to orchestrate itself in a cluster. Microsoft uses a host in the cluster to orchestrate Cluster Shared Volumes (CSVs) because they found that it could give much better performance/scalability.

Some things are quite similar. VMware has Raw Device Mapping, and Hyper-V has pass-through disks. These features present raw LUNs on your physical storage directly to virtual machines. The paid-for versions of vSphere include high availability (HA) to create host clusters. All versions of Windows Server 2012 Hyper-V, including Hyper-V Server 2012, include Failover Clustering to give you HA. The non-free versions of vSphere include vMotion, a feature to move virtual machines from one host to another with no downtime to service functionality. Hyper-V includes Live Migration, a similar feature that supports a wide range of virtual machine mobility between hosts.

You cannot assume that one hypervisor is implemented the same way as another. Lots of little details are different on both platforms. Unfortunately, VMware veterans too often assume that they can implement Hyper-V without learning how the Microsoft solution works. The result is an unhappy employer or customer who has to engage with an actual Hyper-V expert to undo the damage and rescue the Hyper-V farm. That same rule can probably be safely applied to Hyper-V engineers assuming that they can deploy vSphere. If you are a VMware expert and you are reading this book to learn, we strongly recommend that you don't skip any pages or sections. You never know which little nugget of information could be the difference between being an IT hero or a villain.

Other Essential Knowledge

This section highlights a few other topics that we strongly recommend that you learn more about.

Microsoft Assessment and Planning Toolkit

Microsoft released and has continued to improve the Microsoft Assessment and Planning Toolkit (MAP) to help engineers and consultants get off to a good start with their deployment projects. MAP (`http://technet.microsoft.com/library/bb977556.aspx`) is a free tool that is used to scan and assess an existing environment. In the context of planning a Hyper-V project, it will do the following:

- Discover physical and virtual machines (including Hyper-V, vSphere, and ESXi)
- Use this data to identify the machines that could possibly be converted into Hyper-V virtual machines (candidates)
- Gather resource utilization data of the virtual machine candidates over a predetermined time frame
- Enter the specifications of possible hosts and storage

- Use the gathered performance data to estimate how many hosts will be required to run your virtual machine candidates
- Generate a spreadsheet with lots of detailed information and a report in Microsoft Word format that can be used as the basis of a project proposal

MAP is free. It requires only a short amount of time for installation, minimal effort for configuring resource monitoring, and no time at all for entering various host specifications to size a solution. This scientific approach is much better than sticking a proverbial wet finger in the wind to guesstimate the required number of hosts and their specifications.

Sizing a Cloud

A veteran of the hosting industry can tell you that you cannot use traditional sizing techniques to size a cloud. That is because you have no existing workloads to convert into Hyper-V virtual machines or to assess. A cloud features self-service, so the service provider (the IT department in the case of a private cloud) has no idea what will be deployed on the cloud in the future.

The approach that some hosting companies use is to identify a number of different host and storage technologies, specifications, and their costs. The costs of owning (networking, rack space, licensing, power, high availability, disaster recovery, systems management, cooling, and so on) are determined, and these are used to calculate the cost to the customer of the service. For example, a megabyte of RAM might cost a certain amount per month. Find a sweet spot between minimized cost and maximum level of service, and you have yourself a possible architecture.

A public cloud service provider will usually need to be very price competitive, while dealing with the highest levels of security for its tenants. The providers, like those operating a private cloud, might want to consider different tiers of hosts and storage, with some offering cheaper service at reduced cost and availability, while others offer better performance and availability but at a higher cost. Public clouds might lean toward the lower-cost option while private clouds might focus more on availability, but this should be dictated by the strategy decision makers.

PowerShell

As you get further into this book, you might start to wonder whether you accidentally picked a PowerShell book instead of a Hyper-V one. Microsoft gave us a lot of power in PowerShell. You can rapidly deploy change to lots of virtual machines with a single line of code, or you can automate complex tasks with a script. You will see lots of examples of this throughout the rest of this book.

Although some might have believed that PowerShell was being widely used, this was not actually the case. Most people simply never had a reason or the time to invest in PowerShell. Now they might hear that PowerShell is all over Windows Server 2012 and Hyper-V, and they might wonder if it is too hard to learn. The reality is that Microsoft makes it quite easy to pick up some PowerShell administration and scripting skills.

PowerShell is made up of cmdlets. Each cmdlet does some action to something. The naming of cmdlets reflects this:

```
Verb-Noun
```

Here is an example:

```
Get-Help
```

That's the cmdlet that will provide you with help in PowerShell. You can use this to get help on another cmdlet. For example, you can start PowerShell and type in the following:

```
Get-Help Get-Command
```

`Get-Command` is used to get information on a PowerShell `cmdlet.` You can get examples of how to use it by running the following:

```
Get-Help Get-Command -Examples
```

Armed with the knowledge that the results of that instruction will give, you are now able to list all of the Hyper-V PowerShell module cmdlets:

```
Get-Command –Module Hyper-V
```

There are a number of ways that you can run PowerShell:

PowerShell Window This is like a command prompt, and is the best way to quickly make simple changes using single cmdlets.

Scripting Using Notepad You can write a script (saved as a `.PS1` file). This script runs lines in sequence and can include programming structures and decision making.

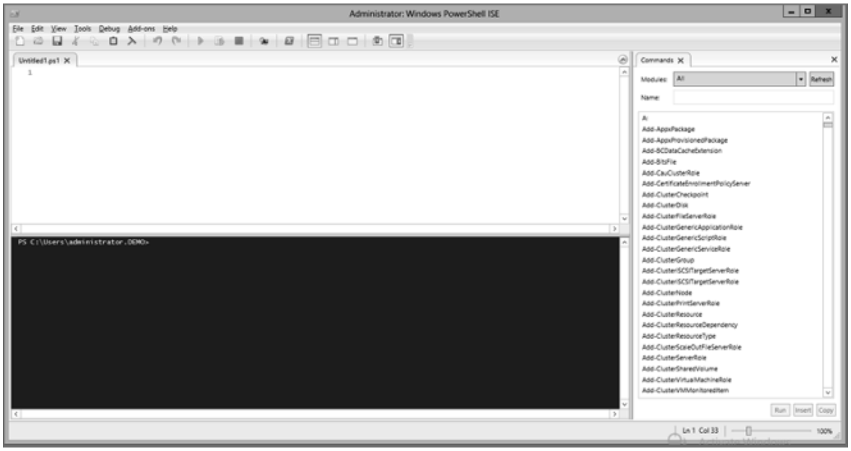
Third-Party Scripting Tools Some excellent scripting tools have been released to help administrators write and manage their PowerShell scripts. These tools can offer better script debugging and management functionality than you will get in Windows Server 2012.

You don't need to spend any money to get a PowerShell scripting tool. Windows Server 2012 includes the Integrated Scripting Environment (ISE), a very nice tool for PowerShell beginners and veterans (see Figure 1.3).

There are two ways to type PowerShell in ISE. You can open one or more scripting windows. Unlike with Notepad, each line is numbered. This is useful if you run a long script, it reports an error, and that error is on line 36. You can scroll straight to line 36 in ISE, but you have to slowly count to line 36 in Notepad.

ISE also features a PowerShell window. There is a good reason for using this window. As you type, you will notice that ISE includes the ability to suggest or auto-complete your cmdlets. Type `New-` in ISE and you'll see it offer you every possibility to complete your cmdlet. This is a nice way to discover cmdlets. If you type `New-VM` followed by a space and a hyphen, you'll see that ISE can suggest ways to use the cmdlet.

The Commands pane on the right can be used to search for cmdlets. When you find one, you can click a button called Show Details. This can reveal the different ways (if available) to use a cmdlet and the flags/parameters that can be used, with the mandatory ones being highlighted with an asterisk. You can select and complete the parameters in the Command pane and then click Run to run the configured cmdlet or click Insert to add it to the PowerShell window.



```
PS C:\Users\administrator.DEMO>
```

Modules: All

Name:

A:
Add-AppxPackage
Add-AppxProvisionedPackage
Add-BCDataCacheExtension
Add-BitsFile
Add-CauClusterRole
Add-CertificateEnrollmentPolicyServer
Add-ClusterCheckpoint
Add-ClusterDisk
Add-ClusterFileServerRole
Add-ClusterGenericApplicationRole
Add-ClusterGenericScriptRole
Add-ClusterGenericServiceRole
Add-ClusterGroup
Add-ClusteriSCSITargetServerRole
Add-ClusteriSCSITargetServerRole
Add-ClusterNode
Add-ClusterPrintServerRole
Add-ClusterResource
Add-ClusterResourceDependency
Add-ClusterResourceType
Add-ClusterScaleOutFileServerRole
Add-ClusterServerRole
Add-ClusterSharedVolume
Add-ClusterVirtualMachineRole
Add-ClusterVMMonitoredItem

FIGURE 1.3
The Integrated Scripting Environment

Other nice features in ISE include the ability to debug a script (troubleshooting by pausing a running script to inspect the code and variables) by using the options in the Debug menu, or use Start Snippets in the Edit menu to discover PowerShell snippets.

PowerShell might take some effort to learn at first, but you will quickly learn the value of it. A simple script can save you lots of time in the long term. Scripts can give you consistent results when run repeatedly. For example, a consultant can use a single script on lots of customer sites to standardize configurations and achieve the same high levels of deployment.

And here's your best reason to learn PowerShell: The most powerful features of Hyper-V are revealed and enabled only by PowerShell. You'll see many examples of this as you progress through this book.

You will never be the first person to look for help on PowerShell. There are plenty of places to find help or solutions; just go to your favorite search engine and start there. Here are a few other resources:

Scripting with Windows PowerShell Videos `http://technet.microsoft.com/scriptcenter/powershell.aspx`

This is the first of five videos teaching the basics of PowerShell scripting.

The Script Center `http://technet.microsoft.com/scriptcenter/default`

This Microsoft site has lots of example solutions.

PowerShell for Beginners `http://social.technet.microsoft.com/wiki/contents/articles/4307.powershell-for-beginners-en-us.aspx`

This is a collection of resources on the Microsoft TechNet Wiki.