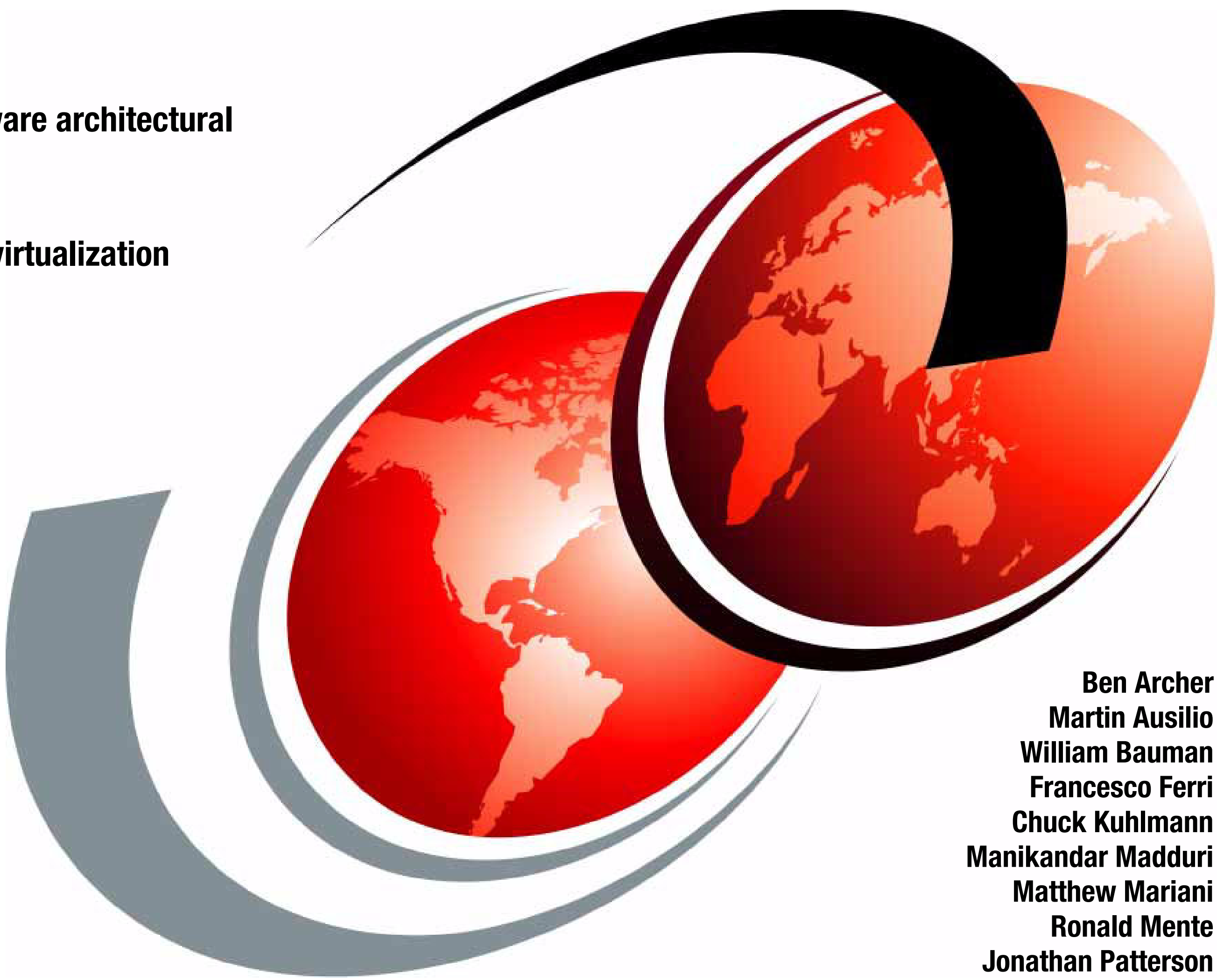
System x Virtualization Strategies

Enterprise class virtualization solutions

eX4 hardware architectural features

Software virtualization platforms



Ben Archer
Martin Ausilio
William Bauman
Francesco Ferri
Chuck Kuhlmann
Manikandar Madduri
Matthew Mariani
Ronald Mente
Jonathan Patterson

Redpaper

ibm.com/redbooks

International Technical Support Organization

System x Virtualization Strategies

February 2010

REDP-4480-00

Note: Before using this information and the product it supports, read the information in “Notices” on page v.

First edition (February 2010)

This edition applies to Model eX4 of IBM System x.

Contents

Notices

This information was developed for products and services offered in the U.S.A.

IBM may not offer the products, services, or features discussed in this document in other countries. Consult your local IBM representative for information on the products and services currently available in your area. Any reference to an IBM product, program, or service is not intended to state or imply that only that IBM product, program, or service may be used. Any functionally equivalent product, program, or service that does not infringe any IBM intellectual property right may be used instead. However, it is the user's responsibility to evaluate and verify the operation of any non-IBM product, program, or service.

IBM may have patents or pending patent applications covering subject matter described in this document. The furnishing of this document does not give you any license to these patents. You can send license inquiries, in writing, to:
IBM Director of Licensing, IBM Corporation, North Castle Drive, Armonk, NY 10504-1785 U.S.A.

The following paragraph does not apply to the United Kingdom or any other country where such provisions are inconsistent with local law: INTERNATIONAL BUSINESS MACHINES CORPORATION PROVIDES THIS PUBLICATION "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESS OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF NON-INFRINGEMENT, MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE. Some states do not allow disclaimer of express or implied warranties in certain transactions, therefore, this statement may not apply to you.

This information could include technical inaccuracies or typographical errors. Changes are periodically made to the information herein; these changes will be incorporated in new editions of the publication. IBM may make improvements and/or changes in the product(s) and/or the program(s) described in this publication at any time without notice.

Any references in this information to non-IBM Web sites are provided for convenience only and do not in any manner serve as an endorsement of those Web sites. The materials at those Web sites are not part of the materials for this IBM product and use of those Web sites is at your own risk.

IBM may use or distribute any of the information you supply in any way it believes appropriate without incurring any obligation to you.

Information concerning non-IBM products was obtained from the suppliers of those products, their published announcements or other publicly available sources. IBM has not tested those products and cannot confirm the accuracy of performance, compatibility or any other claims related to non-IBM products. Questions on the capabilities of non-IBM products should be addressed to the suppliers of those products.

This information contains examples of data and reports used in daily business operations. To illustrate them as completely as possible, the examples include the names of individuals, companies, brands, and products. All of these names are fictitious and any similarity to the names and addresses used by an actual business enterprise is entirely coincidental.

COPYRIGHT LICENSE:

This information contains sample application programs in source language, which illustrate programming techniques on various operating platforms. You may copy, modify, and distribute these sample programs in any form without payment to IBM, for the purposes of developing, using, marketing or distributing application programs conforming to the application programming interface for the operating platform for which the sample programs are written. These examples have not been thoroughly tested under all conditions. IBM, therefore, cannot guarantee or imply reliability, serviceability, or function of these programs.

Trademarks

Preface

IBM® is the global leader in enterprise class virtualization solutions. IBM offers a full range of virtualization hardware, including the IBM Enterprise X Architectures (eX4 and eX5), as well as software. As more modern computing is becoming open and standards-based, IBM is also a global partner with both hardware and software vendors of virtualization technologies, such as VMware's ESX.

This IBM Redpaper™ publication discusses many of the major virtualization environments that run on IBM System x®, particularly the eX4 architecture. The audience for this paper includes decision-makers, consultants, architects, and planners who want to consider the benefits of virtualization on System x.

The team who wrote this paper

This paper was produced by a team of specialists from around the world working with the International Technical Support Organization, Poughkeepsie Center.

Ben Archer, Australia
Martin Ausilio, Argentina
William Bauman, United States
Francesco Ferri, Italy
Chuck Kuhlmann, United States
Manikandar Madduri, India
Matthew Mariani, United States
Ronald Mente, United States
Jonathan Patterson, United States

Thanks to the following people for their contributions to this project:

Mike Ebbers
International Technical Support Organization, Poughkeepsie Center

Thanks to the authors of the draft edition:

Joshua Blumert
Michael Goodenough
Jonathan Patterson
Sami Raunila

Thanks to the contributors to the draft edition:

Chris Almond
Steve Ashton
Vy Phan

Now you can become a published author, too!

Here's an opportunity to spotlight your skills, grow your career, and become a published author - all at the same time! Join an ITSO residency project and help write a book in your area of expertise, while honing your experience using leading-edge technologies. Your efforts will help to increase product acceptance and customer satisfaction, as you expand your network of technical contacts and relationships. Residencies run from two to six weeks in length, and you can participate either in person or as a remote resident working from your home base.

Learn more about the residency program, browse the residency index, and apply online at:

ibm.com/redbooks/residencies.html

Comments welcome

Your comments are important to us!

We want our papers to be as helpful as possible. Send us your comments about this paper or other IBM Redbooks® publications in one of the following ways:

- Use the online **Contact us** review Redbooks form found at:

  ibm.com/redbooks

- Send your comments in an e-mail to:

  redbooks@us.ibm.com

- Mail your comments to:

  IBM Corporation, International Technical Support Organization
  Dept. HYTD Mail Station P099
  2455 South Road
  Poughkeepsie, NY 12601-5400

1

System x virtualization strategies

IBM System x technology provides a platform for achieving increased workload consolidation and performance, while reducing the total cost of hardware. The shift toward virtualized computing involves new approaches, new ideas, and a shift in thinking from disparate, physical resources to unified, logical resources. With virtualization, economies of scale and utilization of system resources such as processors, storage, and networks are significantly increased.

IBM is the global leader in enterprise class virtualization solutions. IBM offers a full range of virtualization hardware, including the eX4 and upcoming eX5 Architectures, as well as software. As more modern computing is becoming open and standards-based, IBM is also a global partner with both hardware and software vendors of virtualization technologies, such as VMware, Red Hat, and Microsoft®.

1.1 What is virtualization

Virtualization is the process of presenting computing resources in ways that users and applications can easily get value out of them, rather than presenting them in a way dictated by their implementation, geographic location, or physical packaging. In other words, virtualization provides a logical rather than physical view of data, computing power, storage capacity, and other resources. Virtualization is normally implemented on servers to:

- Create virtual resources within single physical device or combining multiple devices.
- Reach beyond the box to see and manage many virtual resources as one.
- Dynamically change and adjust across the infrastructure.

1.2 Applying virtualization to your needs

Using virtualization can reduce costs in a variety of ways, such as hardware, software and management costs. A number of areas in which virtualization can be used include:

- Server consolidation and server containment

  This is one of the most important benefits and is one of the key reasons to use software virtualization technologies on high-end System x servers. Server containment is the next phase after server consolidation. After you have consolidated your existing production servers, you do not want requests for new Windows® and Linux® systems deployments compromising the streamlining that you have achieved in your data center.

- Flexible development and test environments

  The ease and flexibility of creating and reconfiguring guest operating systems (known as virtual machines or VMs) means that development and test environments get significant benefit from virtualization. In fact, this is where most of these x86 virtualization technologies were used when first available.

- Disaster recovery made much easier

  Disaster recovery (DR) is another key reason to implement virtualization technologies. Because the whole virtual machine running on any virtualization technology is typically self-contained in a single file, it becomes very easy to manipulate a Windows machine.

- Faster application deployment

  Products such as VMware ESX Server and Microsoft Virtual Server store the entire virtual machines in a single large file. As a result, operating system and application provisioning can be achieved by simply copying a standard template virtual machine file to a new file, which becomes the new virtual machine. As a result, you can activate your new VM in a matter of minutes or hours.

1.3 Desktops

Virtual Desktop solutions that offer access any time and anywhere. IBM Virtual Infrastructure Access Services uses a connection broker, seamlessly linking end-user devices with centrally hosted applications. The solution acts as a universal connection plug that enables users to access the applications and information they require from virtually anywhere. Users log on over a security-rich connection through the Internet, and connect to applications that they would normally access in a Microsoft Windows environment.

For example, if academic institutions enable *remote* access to applications, the institutions can help to:

- Facilitate mobility by decoupling students and faculty from buildings, buildings from IT infrastructure, and infrastructure from applications.
- Significantly reduce the need for laptops and PCs.
- Quickly respond to new or changing requirements.
- Simplify and speed rollout of new educational applications.
- Measure, monitor, and meter software application usage.

2

The IBM Enterprise X Architectures

IBM X-Architecture® is not just about performance, it is about innovation, scalability, and reliability.

2.1 The Enterprise X4 architecture

In large, scalable servers, the importance of the memory, I/O, and disk subsystems are paramount. To efficiently utilize multiple execution engines, whether they are cores, threads, or physical processors, you have to achieve two things. First, your software must have the ability to distribute workload across the execution engines, this applies to the hypervisor, the operating system, and the application. Second, your memory and I/O subsystems must have the physical ability to feed the execution engines with enough data to keep them fully utilized.

These two requirements are easier said than done; indeed the average processor utilization in an x86 computing platform is 5 - 10%. Software vendors have previously been able to fall back on the increasing clock speed of the processors for performance gains, however the recent shift to multi-core processing requires intelligence coded into the applications to support parallel processing. The applications can eventually be recoded, but this takes time and a finite limit always exists to the performance benefits that can be achieved by adding processor cores. Using software to virtualize your servers offers a way to use existing applications, not written for parallel processing, and still scale up in performance because the hypervisor can distribute multiple virtual machines across all of the processor cores in the system.

Virtualization brings many other benefits such as increased flexibility for server fault tolerance, disaster recovery, and power savings that reduce costs and support environmental responsibility. It is however a software solution and can therefore do little to alleviate the problem of physically feeding data into the processing units fast enough to maximize utilization. The performance of the processor, memory and I/O subsystems inside an x86 server must complement each other to allow virtualization software to get the most use out of the machine.

The System x data center is built to face current industry challenges. Figure 2-1 shows the eX4 chipset.

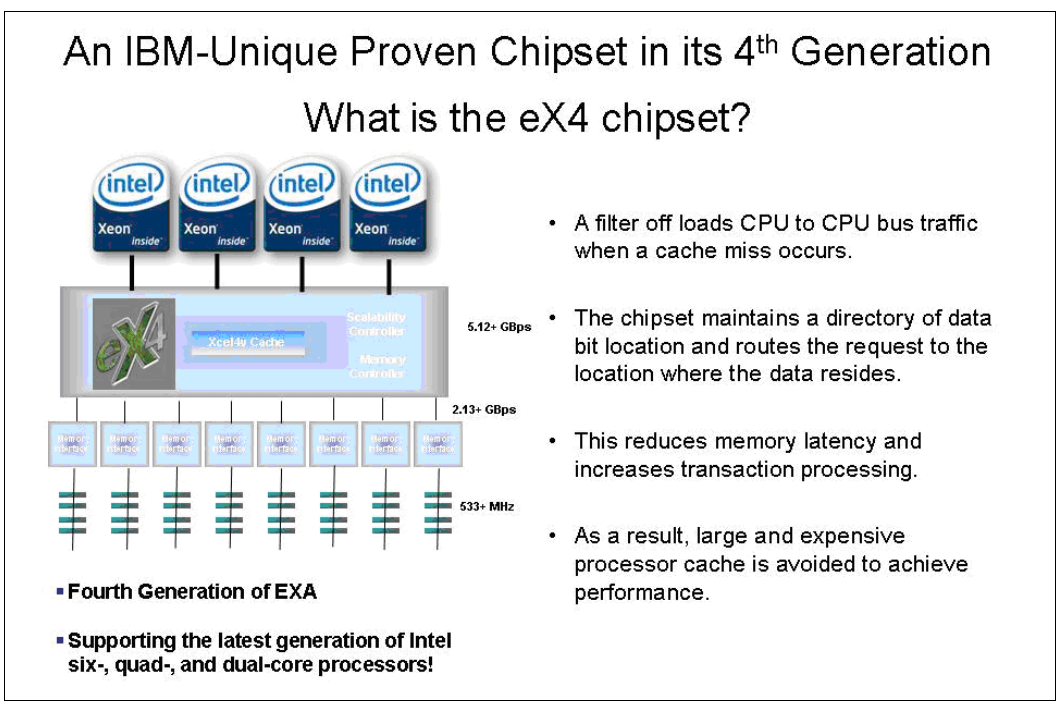


Figure 2-1 The eX4 chipset

2.1.1 x3850 M2 at a glance

Figure 2-2 shows the x3850 M2 unit.

Figure 2-2 3850 M2

The x3850 M2 and x3950 M2 have the following key features:

- A 4-socket-capable server
- The eX4 Architecture featuring the XA-64e fourth-generation chipset
- Two standard Intel® Xeon MP dual, quad or hex core processors, upgradable to four. These processors support 64-bit addressing with the Intel 64 Technology architecture.
- Support for Intel Virtualization Technology (VT)
- Support for an internal removable flash drive installed in a dedicated USB connector on the system board. It can be used to boot an embedded hypervisor, such as ESX3i.
- Memory standard of 4 GB or 8 GB, expandable to 256 GB (using 8 GB DIMMs), using high-performance PC2-5300 ECC DDR2 DIMMs
- Active Memory™ with Memory ProteXion, memory mirroring, memory hot-swap and hot-add, and Chipkill
- Seven half-length 64-bit PCI Express x8 slots, two of which are hot-swap
- Integrated LSI 1078 serial-attached SCSI (SAS) controller.
- Support for RAID-0 and the RAID-1 standards. To enable additional RAID features and a 256 MB battery-backed cache, an optional ServeRAID-MR10k RAID controller is available.
- Four internal hot-swap drive bays for up to 584 GB of internal storage (using 146 GB disks)
- Integrated dual-port broadcom 5709C PCI Express Gigabit Ethernet controller
- Onboard Baseboard Management Controller and Remote Supervisor Adapter II adapter as standard
- Three-year warranty onsite, nine hours per day, five days per week, with a next business day response

Integrated Hypervisor ESX 3i Models

Certain x3850 M2 models have VMware ESXi embedded into the server. This is implemented through a USB stick that plugs straight into the motherboard. When the server is powered on, it boots directly to the set up screen for ESXi and enables you to set a root password and various simple networking parameters. You can then point a vSphere client at the server to start managing the server and populating it with virtual machines. Alternatively you can manage the system through the VMware Virtual Center. For more Information see Chapter 3: VMware Virtual Infrastructure.

2.1.2 Scalability options with the x3950 M2

An x3950 M2 can be connected to a maximum of three other similar units to form a 4-node, 16-socket complex. This scaling can be done in a linear fashion, such as adding processors, memory, and I/O connections in the same proportions; or you can use the scaling to balance system performance. For example, in a non-CPU-intensive workload that requires very large amounts of memory, you can have two x3950 M2s with two processors in each, resulting in the processors being able to access the full 64 DIMM slots. This balancing of resources prevents you from having to buy processors simply to get extra memory and can therefore save on licensing costs and power. Figure 2-3 shows the combinations of x3950 M2s that are currently supported.

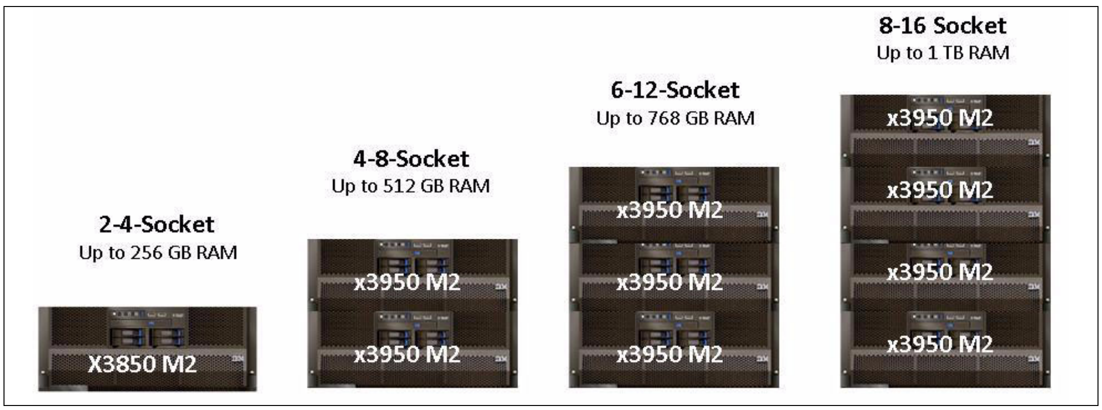


Figure 2-3 x3950 scalability

2.1.3 Advantages of a scale-up strategy

As mentioned in 2.1.2, “Scalability options with the x3950 M2” on page 8, a popular strategy is to standardize on a particular model from which to build an organization's farm of virtual host servers. This approach is good for a number of reasons, described in the following list. For example a VM might not be able to live migrate between two dissimilar hosts, which can create barriers, preventing VMs from running in an optimized way.

- Large hosts: Advantages of the eX4 Architecture as an organization's virtualization strategy.

  The number of VMs is not altered but the number of hypervisor images and physical assets to manage will be reduced as well. Server management costs are much higher than acquisition costs and having fewer physical systems to manage will no doubt reduce this. There is also a clear upgrade path when using the eX4 Architecture. Instead of buying and installing more systems and more hypervisor layers, one can simply schedule down-time to upgrade the host server from a 4-socket to an 8-socket system, and then start adding workload to the new server immediately. This argument is countered in smaller consolidations if you only have a small number of physical servers in the first place. They can be potentially virtualized onto a single large system, but this approach is not advisable for redundancy reasons.

- Larger VMs run more efficiently on larger hosts.

  For a VM with four virtual CPUs to do any work, the hypervisor must find and schedule four physical CPU cores. However, putting four vCPU VMS on a 4-core system is not advisable. For this machine to do any work, the scheduler has to clear out all other virtual machines and put their work on hold. This step reduces performance of both the large VM and the other VMs on the system. There are more permutations of ways to schedule multiple VMs with more physical CPU cores to choose from. This relationship is based on the formula $n \ choose \ r$, where n is the number of CPU cores available to the scheduler and r is the number of vCPUs in the VM. The $n \ choose \ r$ formula is not a linear relationship. If we set r constant at 2 and look at the scheduling opportunities on a 4, 8, 16, and 32-core system, we get the graph shown in Figure 2-4.

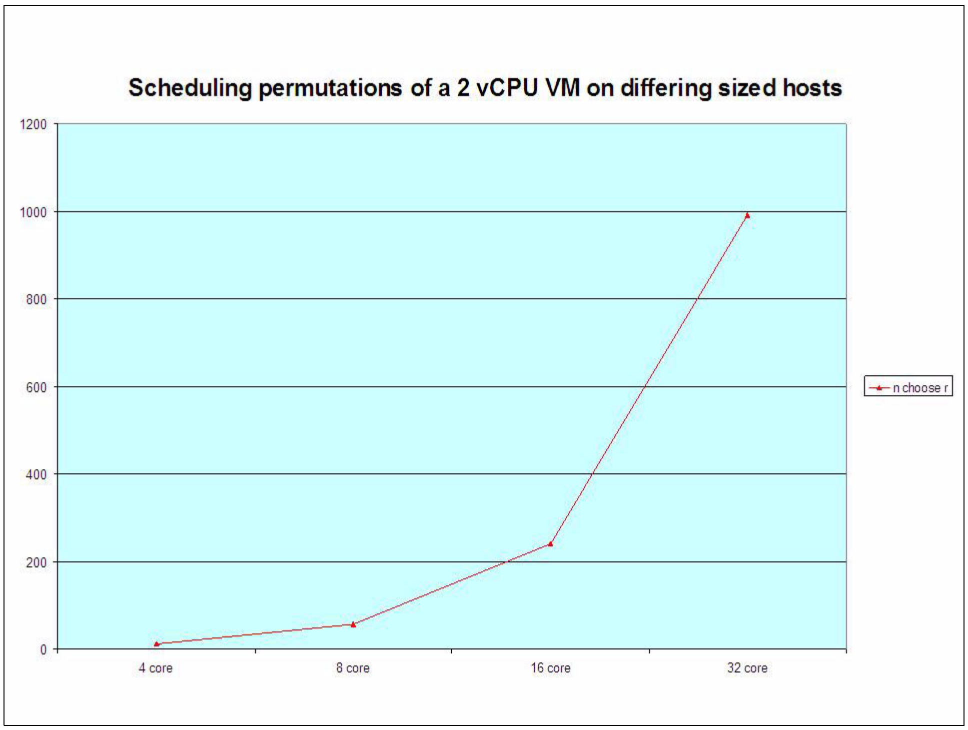


Figure 2-4 Scheduling virtual CPs

  Figure 2-4 shows that the opportunities to fit VMs onto a host in any particular time-slice grow exponentially. A 16-core system has more permutations than 2- or 8- core machine. Likewise, a 32-core system has many times more permutations than 4- or 8-core machines. Quantifying how much this phenomenon will effect performance of a particular environment is very difficult. Too many variables are involved and would require complex statistical analysis.

- Memory deduplication ratios are higher if you have more VMs in the same address space.

  In VMware ESX and Red Hat Enterprise Virtualization, this memory deduplication is also known as transparent page memory sharing. The kernel of the hypervisor scans pages in all of its VMs for similar pages. If it finds two pages that are exactly the same, it will deduplicate and point both VMs at the same physical address space. If one of the VMs tries to write to the page, the memory sharing is forced to end. If all of the VMs on a host run the same O/S, deduplication will unlikely be possible, therefore, you can have many VMS all reading from the same physical address space and your deduplication advantage will be higher.

- Fewer host servers means fewer unused pools of resources.

  One of the problems and reasons that SANs have become so popular is that they avoid the silos of unused disk space that occurs when you attach disk locally. The same argument can be said of using smaller virtual hosts systems. Any CPU time or memory space that is not being used on one host cannot be used by a VM struggling to find resource on another host. A live migration may be instigated to rebalance the VM (but this also requires CPU resources and uses up network bandwidth). Having fewer large hosts reduces the probability that a VM would need to perform a live migration to another host simply to find more resources, which in turn reduces live-migration network traffic and CPU overhead.

- Fewer network ports are needed, and networks are integrated

  Fibre Channel switches, HBAs, cables, SFPs[1], and port licenses are expensive, so be sure to use them to their full capacity. Having fewer physical hosts helps to keep down the costs of these components and helps to increase their utilization. Many hypervisors allow you to create integrated networking objects, such as virtual switches within the kernel of the hypervisor. This means that inter-VM traffic does not have to be sent out over the network. Inter-server IP traffic incurs minimal latency and helps reduce the load on the external Ethernet switches.

[1] Host bus adapter (HBA); small form-factor pluggable (SFP)

3

Virtualization strategies architectural overview

This chapter illustrates the architectural differences in x86 hypervisors available in the market today.

3.1 Background

The three primary categories of x86 architecture include:

- Containers: Virtualization code that runs as an application within a traditional operating system. The OS creates encapsulated, isolated virtual instances that are pointers to the underlying host OS on which it is executed. This type is also known as *operating system virtualization*.
- Type 2: Virtualization code that runs as an application within a traditional operating system (OS) environment and that creates fully emulated instances of the hardware made available to it by the traditional OS on which it is executed. This type is also known as *hosted* hypervisors.
- Type 1: Virtualization code that runs directly on the system hardware and that creates fully emulated instances of the hardware on which it is executed. This type is also known as *native* or *bare metal*.

Figure 3-1 illustrates the high-level software *stack* used in each architecture category. Note that in each model, the virtualization layer is implemented at a separate level, providing varying costs and benefits to the intended application.

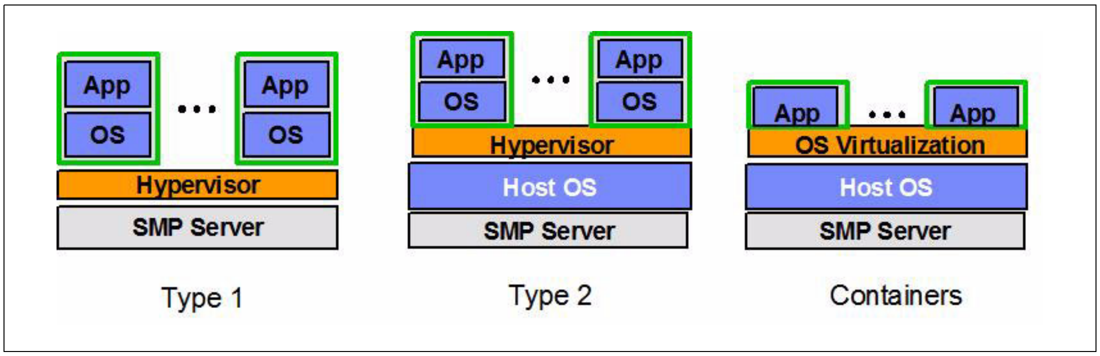


Figure 3-1 Architecture stacks

In addition to the architectural categories, also be aware of certain fundamental concepts of a hypervisor, including:

- Virtual Machine Monitor (VMM): This is the technical term for the code that creates, manages, and destroys the virtualized hardware.
- Paravirtualization: This is the modification of software to make it aware that it is running in a virtualized environment. For a given hypervisor, this can include one or both of the following elements:
  - Kernel Paravirtualization: This refers to modification of the OS kernel, which is becoming standard and requires guest OS and hypervisor compatibility
  - Driver Paravirtualization: This refers to modification of guest OS I/O drivers (network, storage, and others), which is common. Examples include VMware Tools, MS Integration Components.

These concepts are referenced in the remainder of this chapter; they relate to the specific hypervisors available today.

3.2 Operating system virtualization: containers

In the container model category, the virtualization layer is implemented by creating virtual operating system (OS) instances that are merely pointers back to the key system files of the root OS, as shown in Figure 3-2. These pointers, which reside in protected memory within the OS container, provide for low memory overhead and, therefore, a large density of virtual instances. Such densities are one of the key advantages of the container architecture over Type 1 and 2 models, which traditionally require a full instantiation of the guest operating system for each virtual machine.

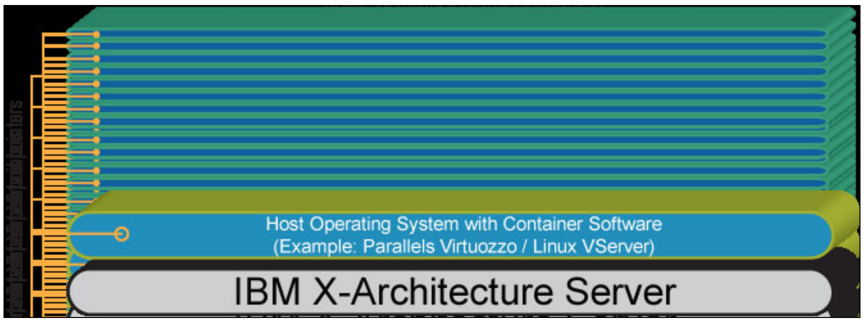


Figure 3-2 Virtual instances (pointers)

By virtue of shared system files, all containers may only provide guests based on the root OS. As a simple example, a base Windows Server 2003 OS may also be used to create Windows Server 2003 containers. Similarly, any patches or updates applied to the root OS system files are inherited by the child containers. This approach provides a convenient *path once* scenario; however, this can also be a detriment because corruption to the root OS can also affect guest functionality.

Within the container, users may customize the guest instance with specific applications, fixes (but not service packs, which modify the shared system files of the kernel), and OS service components. For situations where certain services or applications are used in a majority of the guest containers, the needed functionality may be installed to the root operating system and then acquired *automatically* by the guest instances in a template-like methodology.

In most cases, the number of containers is limited only by host OS resources available, and each guest may be configured up to the maximum (virtual) hardware and up to the specified limits of the root OS. These scalability characteristics, combined with ease of guest management, make the container approach a strong candidate for applications requiring large virtual machine densities, such as virtual desktop.

Parallels Virtuozzo Containers is the leading OS virtualization product available today. In addition to the features we have listed, Virtuozzo provides high availability and migration of guests across physical hosts (assuming identical root operating systems and patch levels). Architecturally, Virtuozzo implements a proprietary Kernel Service Abstract Layer (KSAL) that protects the host OS files from corruption and a secure copy-on-write (COW) file system that enables independent guest modifications. Similar to the parent partition in hybrid hypervisors (as described later in this document), the first virtual instance created is a simple management container that provides virtual machine monitoring functionality. Parallels Virtuozzo Containers, v4.5, includes support for nested implementations of Virtuozzo within Hyper-V, both within the parent partition and guests. See Figure 3-3 on page 14.

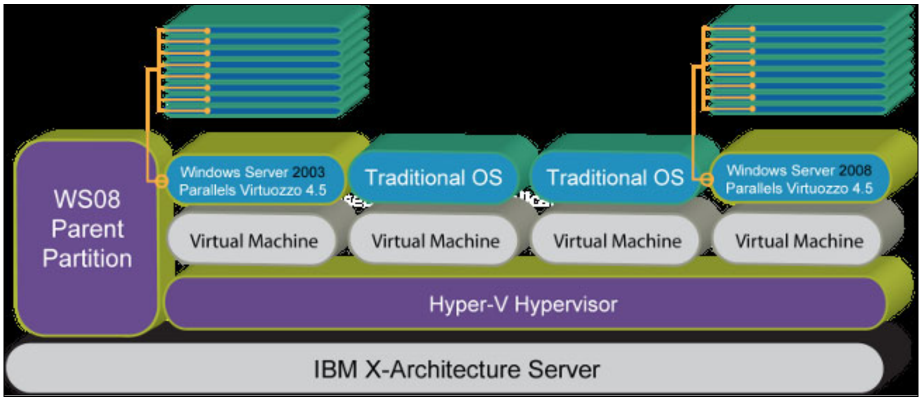


Figure 3-3 Virtuozzo nested implementations

This style of implementation, although complex, is an excellent example of the flexibility of the Virtuozzo architecture to provide significant virtual machine densities. For more information, wee the Parallels Virtuozzo Containers Web site:

http://www.parallels.com/products/pvc45/

3.3 Type 2 hypervisors

Type 2 (*hosted*) hypervisors implement hardware virtualization through a software layer, which is implemented on top of an existing operating system. Unlike the container architecture, Type 2 guests provide a complete, isolated, and independent copy of the guest operating system and typically use paravirtualized drivers for network and I/O to achieve improved guest performance. However, because virtualization functions must pass through the host OS, performance characteristics of guests are considerably lower than bare-metal hypervisors (which is discussed in more depth in the following sections). In addition, few features are available for high availability and enterprise management functions. For these reasons, Type 2 hypervisors are most often used for development and test or desktop-class applications.

Popular examples of Type 2 hypervisors include:

http://www.vmware.com/products/workstation/index.html
http://www.virtualbox.org/
http://www.microsoft.com/windowsserversystem/virtualserver/

All of these products are functionally proven and commonly used today. This document does not discuss them further because they are not intended for enterprise class workloads.

3.4 Type 1 hypervisors

Type 1 (*bare-metal*) hypervisors include the majority of enterprise virtualization offerings available today. This type of hypervisor runs directly on the system hardware and provides the greatest available guest performance. Through enhancements in processor virtualization extensions included in Intel VT and AMD-V technologies, Type 1 hypervisors can achieve beyond native, bare-metal OS performance (such as the equivalent guest performance running directly on dedicated hardware) under certain circumstances.

This class of hypervisors has several sub-types, which we discuss in more detail:

- Stand-alone: VMware vSphere
- Hybrid: Citrix XenServer, Microsoft Hyper-V, OracleVM
- Mixed: Kernel Virtual Machines: Linux KVM

3.5 Type 1 stand-alone: VMware vSphere

In a stand-alone hypervisor, all hardware virtualization and virtual machine monitor (VMM) functions are provided by a single, tightly integrated set of code. This architecture is synonymous with the construct of VMware vSphere and previous generations of the ESX hypervisor. Figure 3-4 is a sample diagram of the architectural overview of VMware vSphere 4.0 (also referred to as ESX 4). Contrary to common belief, VMware is *not* a Linux-based hypervisor. Rather, ESX consists of a strictly proprietary, highly complex OS called VMkernel, providing all virtual machine monitor and hardware virtualization functions. The full version of ESX does provide a Linux-based service console (shown to the left in the diagram). As described in Chapter 4., "VMware vSphere" on page 19, ESXi (the embedded version of the hypervisor) does not contain this service console instance.

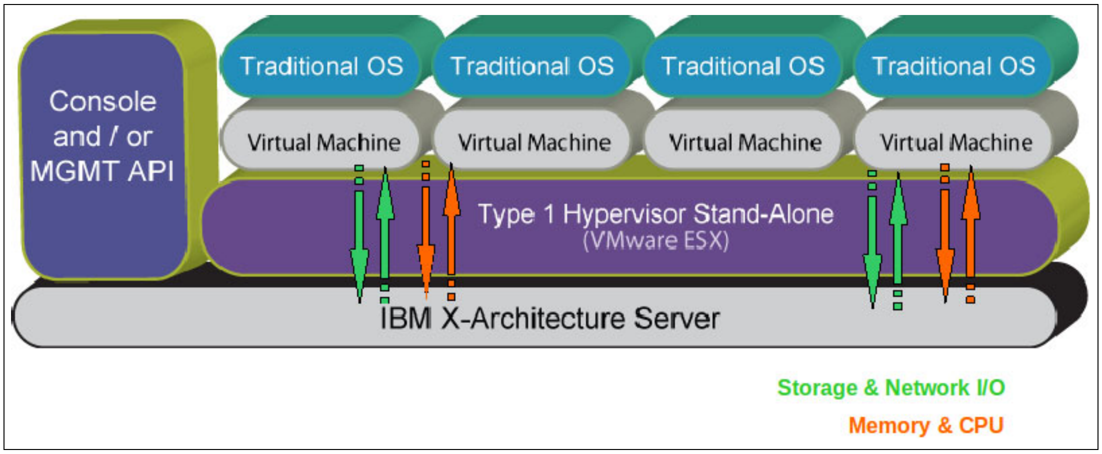


Figure 3-4 ESXi layer

This model is often referred to as a *fat* or *thick* hypervisor, however that is not an accurate description. Although the ESX architecture has grown complex through longstanding development of hardware virtualization using full binary translation, VMware has maintained an industry-leading position in hypervisor performance and feature set. It has furthered that leadership position through recent implementation of processor virtualization extensions.

As is true with most Type 1 and 2 hypervisors, VMware does require network and I/O driver paravirtualization within the guest. These drivers are commonly known as VMware Tools.

3.6 Type 1 hybrid: Citrix XenServer, Microsoft Hyper-V, OracleVM

The *hybrid* Type 1 architecture includes a split software model where a *thin* hypervisor provides hardware virtualization in conjunction with a parent partition (privileged virtual machine), which provides virtual machine monitor (VMM) functionality. This model is associated primarily with Microsoft Hyper-V and Xen-based hypervisors including Citrix XenServer and OracleVM. See Figure 3-5.

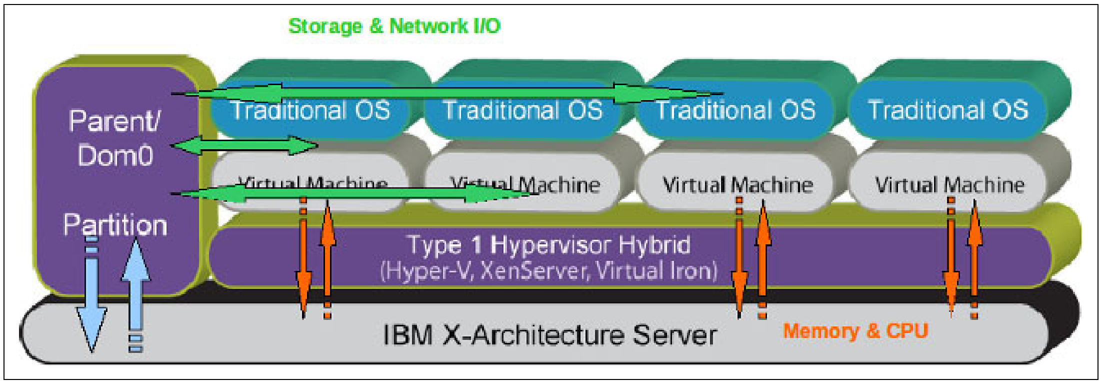


Figure 3-5 Hybrid architecture

The parent partition, also called “Domain 0” (Dom0), is typically a virtual machine that runs a full version of the native operating system with root authority. For example, Dom0 for Xen, enabled and executed within Novell SUSE Linux Enterprise Server (SLES), would execute as a full instance of SLES, providing the management layer of VM creation, modification, deletion, and other similar configuration tasks. At system boot, the Xen-enabled kernel loads initially followed by the parent partition which runs with VMM privileges, serves as the interface for VM management, and manages the I/O stack.

Similar to VMware, all hybrid products available today provide paravirtualized drivers for guests, enabling improved performance to network and I/O resources. Guests that are not implementing paravirtualized drivers must traverse the I/O stack in the parent partition, degrading guest performance. OS paravirtualization is becoming increasingly common in order to achieve optimal guest performance and improved interoperability across hypervisors. For example, Microsoft Hyper-V/Windows Server 2008 R2 provides full OS paravirtualization support (also known as *enlightened* guest support) for Windows Server 2008 and SUSE Enterprise Linux guests.

Although OS paravirtualization is a growing trend in OS development, note that OS paravirtualized is currently not a requirement in the hybrid model. Certain vendors, such as Virtual Iron (Oracle), are targeted at small to medium business workloads and can achieve acceptable guest performance, using full hardware virtualization. The hybrid architecture, originating from the Xen project, provides a range of vendor options in terms of price performance, benefits from contributions in the open-source community (Xen), and is well positioned to accomodate future support of paravirtualized operating systems.

3.7 Type 1 mixed: Linux-based Kernel Virtual Machines

The Linux-based Kernel Virtual Machine (KVM) hypervisor model provides a unique approach to Type 1 architecture. Rather than executing a proprietary hypervisor on bare-metal, the KVM approach leverages open-source Linux (including RHEL, SUSE, Ubuntu, and others) as the base operating system and provides a kernel-integrated module (named KVM) that provides hardware virtualization. The KVM module is executed in user-mode (unlike stand-alone and hybrid hypervisors, which run in kernel/root mode), but enables virtual machines to execute with kernel-level authority by using a new instruction execution context called *guest mode*. See Figure 3-6.

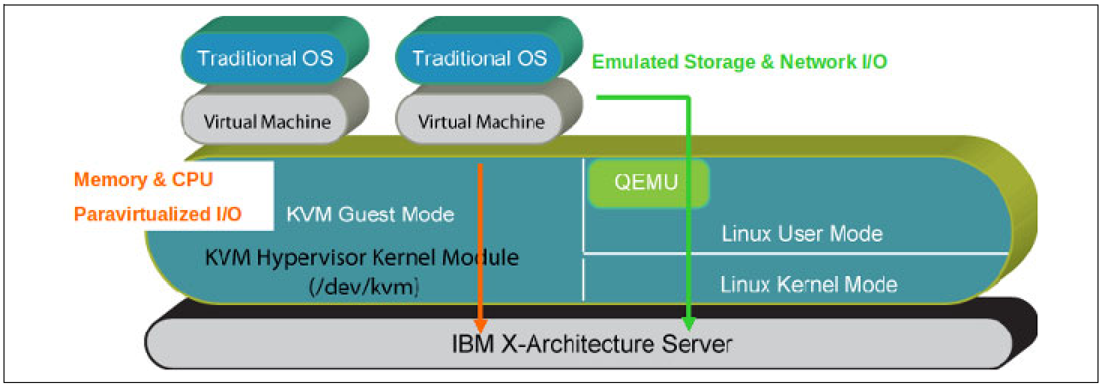


Figure 3-6 KVM layer

KVM provides full hardware virtualization by using a modified version of the open source QEMU hardware emulator package.

http://wiki.qemu.org/Main_Page

This implies that guest operating systems have no requirement for OS paravirtualization, similar to VMware. Linux KVM uses VirtIO as a framework for the implementation of I/O paravirtualization, which utilizes user mode VirtIO drivers inherent in Kernel/QEMU for enhanced performance. KVM is now available as a standard module in current Linux distributions including, but not limited to, Red Hat Enterprise Linux and SUSE Enterprise Linux, as well as desktop class builds such as Ubuntu. The ability to use the existing Linux code base as host OS, combined with the *bare-metal* performance characteristics achieved through guest mode has made Linux KVM an increasingly popular hypervisor alternative.

4

VMware vSphere

Since its original release in 2001, VMware has maintained a market leadership position with its bare metal hypervisor, ESX, and centralized management software, vCenter. With its most recent product release, vSphere 4.0 (ESX 4), VMware continues this tradition with enhancements in scalability, manageability, and reliability. These enhancements include differentiating features such as fault tolerance and distributed virtual switch technology. VMware has achieved these capabilities through a long standing development heritage that has yielded the most mature, most complex hypervisor that is available today.

To further this technology leadership, VMware has taken an approach of aligning vSphere 4.0 (formerly referred to as Virtual Infrastructure) and vCenter Server 4.0 (formerly referred to as Virtual Center Server) under the concept of a *cloud operating system* in order to portray the notion of a more flexible, dynamic environment for deploying virtual resources. Users must pay a cost premium to use the full range of features available in the vSphere product line. Therefore, although VMware has become the standard to which other hypervisors are compared, ESX is currently and will continue to be challenged by competitors with capabilities that are *good enough*.

4.1 VMware product ecosystem

VMware's hypervisor is available in either a full ESX version (ESX Classic) or thin version known as ESXi. ESXi can be pre-installed by the server hardware vendor on either local or bootable USB media, or can be installed by the user on local hard disk drives (ESXi Installable edition). Both full ESX and ESXi versions are functionally equivalent; however, use of advanced features such as VMotion and high availability (HA) require the appropriate license for ESX/ESXi and vCenter Server for centralized management. ESXi is also available in a feature limited *free* downloaded version. The vCenter Server enables features including:

- High availability (HA): Automatic restart of VMs within a host cluster after a host node failure
- Dynamic resource scheduling (DRS): Load-balancing of VMs across multiple hosts in a cluster
- Fault tolerance (FT): The mirroring of a VMs real-time state across two active hosts, enabling zero downtime in the event of a host outage

  Note: There is a vCenter limitation on the number of VMs that can be enabled with this feature; therefore, only the most critical VMs in a system should be configured in this way.
- Storage vMotion: The online movement of an underlying VM file structure from one disk subsystem to another.

  An example is moving from a slow performing disk system to a newer, faster one without.
- Update Manager: Automatic deployment of OS updates to either ESX or ESXi hosts and guests.
- vCenter Heartbeat: Provides clustering of multiple vCenter Server installations in support of higher availability.

VMware has maintained Standard, Advanced, and Enterprise tiers of licensing for advanced features and has added a fourth tier, Enterprise Plus, with the release of vSphere 4.0. In order to have all cluster-related functions, such as DRS, available to a production environment, customers commonly use an Enterprise or Enterprise Plus license.

Figure 4-1 shows the licensing tiers.

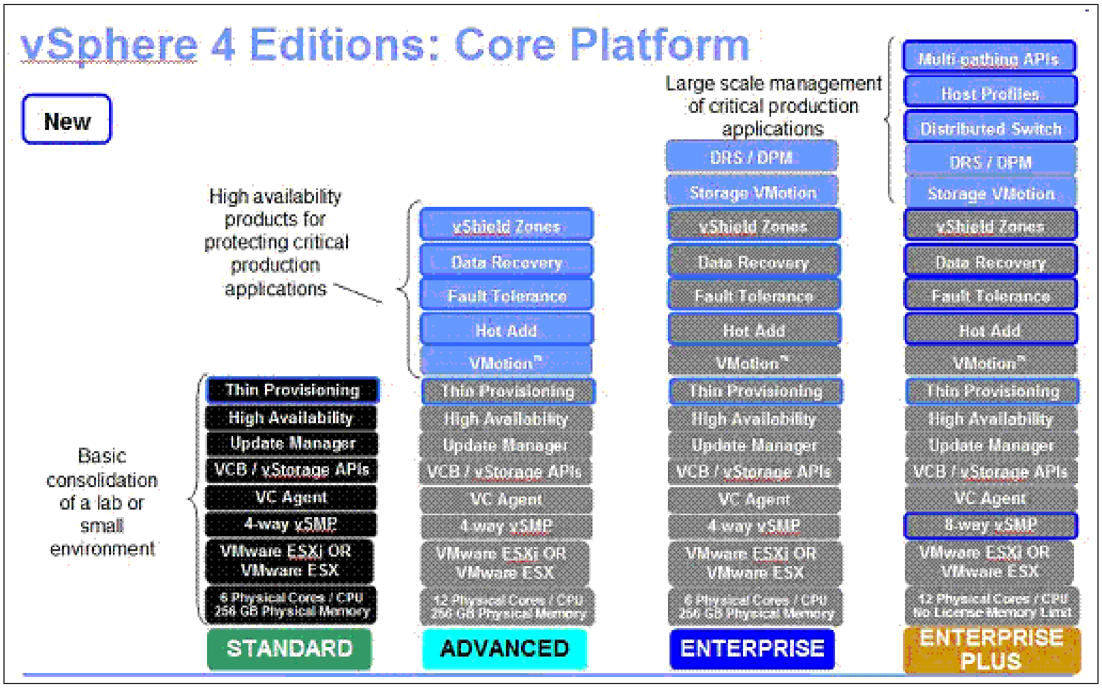


Figure 4-1 VMWare licensing tiers

Additional technologies offered by VMware include:

- VMware Data Recovery/vStorage API for Data Protection: Is a proxy service for performing online backup of complete VM file structures. This technology is also included in most versions of ESX and vCenter.
- VMware Site Recovery Manager (SRM): This technology offers replication of all VMware data center content from one site to another to enable automatic VM infrastructure failover in support of disaster recovery. This requires vCenter interaction as well as support of SAN-to-SAN mirroring at the disk subsystem level, such as Enhanced Remote Mirroring (ERM) on IBM DS4000® and DS5000 products.
- VMware Lab Manager: Provides a resource management layer for allowing interactive, on-demand creation of VMs in team or lab-based environments.
- VMware View: Is VMware's connection broker and virtual desktop infrastructure product.
- VMware Workstation: Is a Type 2 hypervisor software for running VMs on a workstation.
- VMware Player: Is similar to VMware Workstation, but does not allow VM modification.
- VMware ACE: Deploys and manages secure, portable PC environments.
- VMware ThinApp: Provides simplified encapsulation and deployment of end-user applications, for use particularly in virtual desktop environments.

The capabilities of the entire VMware virtualization ecosystem is implied for the purposes of this paper. The intent of this paper is to address virtualization strategies on IBM System x, BladeCenter®, and iDataPlex™ platforms, therefore, the authors assume a certain level of familiarity with VMware's virtualization strategy and ecosystem. We do not provide details of the exact functions of each of the technologies. For more information, go to:

http://www.vmware.com/products/

4.2 VMware vSphere scalability

VMware currently maintains two generally available releases of ESX in order to allow customer transition to the latest version. With each release, VMware has maintained industry technology leadership and has added additional physical and virtual scalability with each subsequent release. ESX 3.5 provides support for up to 128 VMs, 32 cores, and 256 GB of physical memory per host. Guest support includes up to four vCPUs with 64 GB of memory per VM. For vSphere 4.0, VMware has increased these maximums to 320 VMs, 64 logical processors (considers threads/core), and up to 1 TB of physical memory per host. Guest support includes up to eight vCPUs with 256 GB of memory per VM. For the latest support information, visit the VMware maximum supported configurations Web site:

http://www.vmware.com/support/pubs/vs_pages/vsp_pubs_esx40_vc40.html

VMware vSphere is a Type 1 hypervisor. As with other Type 1 hypervisors, vSphere is an excellent candidate for any of IBM x86-based hardware offerings. Each tier (according to Figure 4-1 on page 21) enables additional levels of hardware virtualization support. To fully enable server memory capability beyond 256 GB, purchase the Enterprise Plus edition, with a maximum memory capability of 1TB (at the time of writing). This high level of memory scalability makes ESX an excellent fit for the IBM flagship offering, the X4-based, x3950 M2, as well as the X5 architecture (to be released in 2010). For more details about the scalability of vSphere, visit the vSphere 4.0 announcement page:

http://www.vmware.com/support/vsphere4/doc/vsp_40_new_feat.html

Because Intel and AMD are offering significantly increased processor core-count density per socket, processor utilization becomes less of a concern. Therefore, the scalability of virtualized systems is becoming increasingly dependent on memory capacity. This dependency leaves memory capacity at a disadvantage on many modern servers. The x3950 M2 is the most scalable, reliable server in the x86 world. With the ability to address up to 1 TB of physical RAM, it is a natural fit for VMware's ESX platform.

Table 4-1 lists several possible memory configurations of an ESX server. The table illustrates both high virtual machine density and high memory density. The scalability of the X4 platform is absolutely essential in the high memory density scenarios outlined in the table. Because all production workloads become virtualized, typical database, e-mail, or ERP applications will require this sort of memory density in order to gain return on investment (ROI) of modern processor architectures in modern x86 servers.

Table 4-1 Memory configurations

Total RAM (GB)	Memory overhead (GB)	Total VMs	VM RAM 64	VM RAM 32	VM RAM 24	VM RAM 16	VM RAM 12	VM RAM 8	VM RAM 4	VM RAM 2	VM RAM 1	VM RAM .5
588	1.23	90	2	4	2	6	4	4	8	25	15	20
1011	1.21	88	6	8	4	8	4	4	4	10	20	20
388	3.02	320	-	-	-	-	-	5	25	40	80	170
1021	1.04	66	10	6	4	2	2	2	-	-	-	40

VMware vSphere 4.0 is capable of scaling to 512 virtual host processors and 320 total virtual machines. Table 4-2 details the processor scalability of ESX. The first three rows demonstrate maximum virtual processor per VM limitations, and the next three rows are samples of possible configuration scenarios before reaching a maximum.

Table 4-2 ESX processor scalability

Total Virtual Host Processors Max: 512	Total VMs Max: 384	Virtual CPUs Per VM	Virtual CPUs Per VM	Virtual CPUs Per VM
		1	2	4
512	256	0	256	0
512	128	0	0	128
320	320	320	0	0
460	320	220	80	20
512	262	112	100	50
400	320	260	50	10
Host Virtual Processor Limit Reached				
Total Virtual Machine Limit Reached				
Virtual CPU Per VM Limit Reached				

VMware scalability and feature set together have made vSphere (ESX) the premier hypervisor available today. As mentioned in Chapter 2, “The IBM Enterprise X Architectures” on page 5, attaining a maximum possible utilization of your server, while at the same time maintaining headroom for workload bursts, is important. VMvware vSphere's CPU scalability, in combination with its extraordinary x86-memory scalability and IBM X4 Architecture, make 70 - 80% processor utilization quite attainable. Especially when considering cost premiums for advanced VMware features, no other x86 server platform can achieve such extraordinary return on investment (ROI).

4.3 Considerations

The scalability of vSphere 4.0 enables the highest guest VM density available today and the highest virtual hardware support for resource intensive applications. While providing the highest level of scalability, vSphere 4.0 also provides the highest levels of availability, through industry-proven features including HA, VMotion, and fault tolerance. Therefore, ESX should be a first consideration for general purpose server consolidation, especially when considering the virtualization of mission critical applications such as MS SQL or Exchange. VMware provides multiple tiers of package licensing so that customers can acquire feature support according to their individual needs. However, as described in the previous section, ROI for VMware is best achieved through high density of virtualized guests. Along these lines, if you are considering advanced versions of ESX, select a server platform, such as IBM X4 and X5 based systems, with hardware scalability and availability features that are consistent with that of ESX. IBM X-Architecture provides this scalability along with industry leading reliability, availability, and serviceability (RAS) hardware enhancements that is a best-fit for server consolidation needs.

The ability of VMware to provide high virtual machine densities also makes it a good fit for virtual desktop implementations. VMware, today, provides a near end-to-end solution including hypervisor (ESX) and connection broker infrastructure (VMware View) in support of virtual desktop infrastructure (VDI). VMware also has plans to release advanced multimedia acceleration in the near future for the vSphere 4.0 platform. Virtual machine densities, if sized properly for VDI, typically scale to much greater levels than general several consolidation environments. This means that hypervisor availability can become even MORE important, because having one system down can result in the work stoppage of hundreds of users.

Therefore, every fraction of system availability matters. Again, RAS features of IBM X4 and X5-based systems provide the needed levels of system uptime for VDI.

VMware also holds industry leadership status for the management of virtual environments. The vCenter Server 4.0 provides hypervisor-level high availability, and has also matured into a unique *control center* of the virtual environment. The vCenter Server 4.0 can enable advanced failover scenarios (such as Site Recovery Manager) and a number of vendor offerings to enhance a virtual environment. vCenter Server additionally provides HA at the systems management level through its ability to be clustered (for example, vCenter HeartBeat). Such features add to VMware's appeal for mission-critical applications.

In addition to scalability and availability enhancements, the vSphere 4.0 release also brings to the table the latest in implementation of processor virtualization extensions and host OS support of new hardware features. For example, vSphere 4.0 includes virtual hardware version 7, with features such as hot-add of memory and CPU for supporting guest operating systems (HW v7 also requires vSphere 4.0 or later as the host operating system). Another example is the implementation of VMDirectPath for virtual machines, a software implementation of Intel VT-d/AMD-Vi, which gives a VM *direct* access to hardware for increased I/O performance. The enablement of greater guest OS function, a demonstration of the maturity of ESX, is yet another differentiator of VMware.

With such key characteristics of performance, availability, scalability in addition to a broad portfolio of product offerings and support for guests and hosts, VMware provides a virtualization solution to address nearly all scenarios, particularly those at the high end. Advanced features might require a significant investment in hypervisor and management infrastructure; therefore, a supporting hardware system such as those based on IBM X Architecture should be used in order to best maintain availability of the virtual environment and achieve maximum ROI through high virtual machine density.

5

Microsoft Hyper-V R2 and System Center

Microsoft's Hyper-V R2 Hypervisor is available as a component of Microsoft Windows Server 2008 R2 as well as a stand-alone product named Microsoft Hyper-V Server 2008 R2. Hyper-V poses an interesting value proposition to Microsoft customers in that it is included with the cost of the Standard, Enterprise, and Datacenter offerings of Windows Server 2008. As many companies are already familiar with or standardize on VMware's Infrastructure 3 or vSphere offerings, to complete their own virtualization offering, Microsoft leverages their System Center technologies to enhance the management and functionality of their Hyper-V platforms. Since its initial release in 2008, Hyper-V has been both a stable and solid performing hypervisor. The two biggest concerns with Hyper-V have been addressed in the R2 release: It fully supports failover clustering and it now includes live migration, the ability to move a virtual machine from one physical host to another without service interruption.

5.1 The Microsoft virtualization ecosystem

Simplified, the Microsoft System Center Server Management Suite Enterprise (SMSE) or Datacenter (SMSD) is required, along with Microsoft Windows Server 2008 R2, in order to achieve the highest level of functionality within the Microsoft virtualization ecosystem. Breaking it down a little more, in order to leverage all of the capabilities of Hyper-V R2, it will be necessary to install and configure several components of Microsoft System Center. In an effort to better align System Center with Windows Server licensing, Microsoft has created a Datacenter licensing model for System Center that is probably the most likely selection for most customers. This is not unlike VMware's Infrastructure/vSphere offerings that enhance the capabilities of the base VMware ESX product. One significant advantage to Microsoft's approach is that SMSE is not limited to managing virtual environments, as it was designed to manage all systems, including physical and virtual.

The necessary technologies to implement a complete Microsoft virtualized ecosystem are:

- Microsoft Windows Server 2008 R2 with the Hyper-V Role enabled
- Microsoft Windows Server 2008 R2 Failover Clustering Feature installed and Cluster Shared Volumes enabled
- Microsoft System Center Virtual Machine Manager 2008 R2
- Microsoft Operations Manager 2007 R2
- Microsoft Data Protection Manager

The capabilities of the entire Microsoft virtualization ecosystem is implied for the purposes of this paper. As the intent of the paper is to address virtualization strategies on IBM System x, BladeCenter and iDataPlex platforms, the authors assume a certain level of familiarity with Microsoft's virtualization strategy and ecosystem. Although we do not go into detail on the exact functions of each of the above mentioned technologies, more information can be found at the Microsoft Virtualization page and the Microsoft System Center page:

http://www.microsoft.com/virtualization/default.mspx
http://www.microsoft.com/systemcenter/en/us/default.aspx

5.2 Hyper-V architecture overview

Figure 5-1 on page 27 is a sample diagram of the architectural overview of Microsoft Hyper-V. The Child Partition Guest Operating Systems presented in green are Enlightened (Microsoft's term for paravirtualized kernels). The Child Partitions in purple are traditional guest operating system kernels. The purpose of the diagram, on its own, is merely for broad reference of the Hyper-V Architecture. For more information about the architecture of Hyper-V, consider watching the following demonstration video by Ben Armstrong, Hyper-V Product Manager:

http://cid-17e0c6128f544f9e.skydrive.live.com/self.aspx/Public/Hyper-V%20Architecture%20Demonstration%20Video%20Streaming.wmv

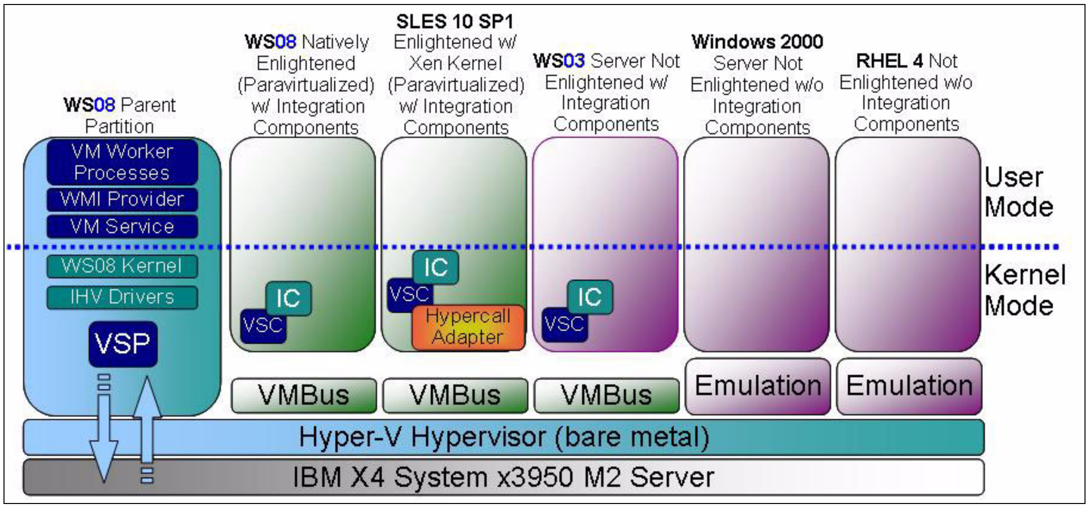


Figure 5-1 Architectural overview of Hyper-V

5.3 Hyper-V scalability

Microsoft Hyper-V is a Type 1 hypervisor, and is therefore an excellent candidate for any of IBM's x86-based hardware offerings. Hyper-V is capable of leveraging the entire physical address space of the Windows Server 2008 R2 kernel, which is presently 1TB with Enterprise or Datacenter Editions. This sort of memory scalability makes Hyper-V an excellent fit for the IBM flagship offering, the X4-based, x3950 M2, as well as the X5 architecture to be released in 2010. For details about the scalability of Hyper-V R2, visit the Windows Virtualization Team Blog:

http://blogs.technet.com/virtualization/archive/2009/05/12/tech-ed-windows-server-2008-r2-hyper-v-news.aspx

The scalability of virtualized systems is becoming increasingly dependent on memory capacity. As Intel and AMD are offering significantly increased processor core count density per socket, processor utilization becomes less of a concern. This does leave memory capacity at a disadvantage on many modern servers. The x3950 M2 is the most scalable, reliable server in the x86 world. With the ability to address up to 1TB of physical RAM, it is a natural fit for Microsoft's Hyper-V platform.

In Table 5-1 we detail some of the possible memory configurations of Hyper-V server. The table illustrates both high virtual machine density as well as high memory density. The scalability of the X4 platform is absolutely essential in the high memory density scenarios outlined in Table 5-1. As all production workloads become virtualized, typical database, e-mail, and ERP applications will require this sort of memory density in order to gain return on investment of modern processor architectures in modern x86 servers.

Table 5-1 Hyper-V memory configurations

Total Physical RAM Used (GB)	Memory Overhead (GB)	Total VMs	VM RAM 64	VM RAM 32	VM RAM 24	VM RAM 16	VM RAM 12	VM RAM 8	VM RAM 4	VM RAM 2	VM RAM 1	VM RAM 0.5
588	1.23	90	2	4	2	6	4	4	8	25	15	20
1011	1.21	88	6	8	4	8	4	4	4	10	20	20
476	3.52	384						9	25	50	100	200
1021	1.04	66	10	6	4	2	2	2				40

Hyper-V is capable of scaling to 512 Virtual Host Processors and 384 Total Virtual Machines. Any combination of these limits is possible. Table 5-2 details the processor scalability of Hyper-V. The table outlines various maximums, the first three rows demonstrate maximum Virtual Processor Per VM limitations, and the next three rows are samples of possible configuration scenarios before reaching a maximum.

Table 5-2 Hyper-V CP scalability

Total Virtual Host Processors Max: 512	Total VMs Max: 384	Virtual CPUs Per VM	Virtual CPUs Per VM	Virtual CPUs Per VM
		1	2	4
512	256	0	**256**	0
512	128	0	0	**128**
384	**384**	**384**	0	0
512	372	272	80	20
512	262	112	100	50
476	**384**	320	50	14
Host Virtual Processor Limit Reached				
Total Virtual Machine Limit Reached				
Virtual CPU Per VM Limit Reached				

As mentioned in Chapter 2, "The IBM Enterprise X Architectures" on page 5, attaining a maximum possible utilization of your server (for ROI purposes), while at the same time maintaining headroom for workload bursts, is important. Hyper-V's CPU scalability, in combination with its extraordinary x86-memory scalability and IBM X4 Architecture, make 70 - 80% processor utilization quite attainable. No other x86 server platform can achieve such extraordinary ROI.

6

Red Hat Enterprise Virtualization (KVM)

The Red Hat Enterprise Virtualization (RHEV) offering is based on its acquisition of the Qumranet project. The Qumranet project was initially developed to address the shortcomings of Remote Desktop Protocol (RDP) for virtualized desktops. The project developed its own method of instantiating virtual machines. The method, now known as Linux Kernel-based Virtual Machines (KVM), was designed to simplify the management of virtualized instances. It was also to provide a robust framework with which to enhance virtualized instances capabilities from the underlying architecture.

The developers of the Qumrant project released the KVM portion of it to the open source community early in its development. This release accelerated adoption and development of the virtualization technology. KVM was quickly accepted by the community and was integrated into the Linux 2.6.20 kernel.

The Red Hat Enterprise Virtualization offering is available in two licensed versions: RHEV for Servers and RHEV for Desktops. For more information about the technologies, and license models provided by each, see the Red Hat Enterprise Virtualization page:

`http://www.redhat.com/virtualization/rhev/`

For the purposes of our discussion, all features of the RHEV ecosystem are assumed.

6.1 Product ecosystem

KVM is available in the open-source Linux kernel (as of v2.6.20), and therefore can be obtained and implemented through a variety of sources, including both server and desktop-class operating systems. Red Hat is the furthest along in enterprise deployment capabilities, offering full support in RHEL 5.4. This lead is not surprising, given Red Hat's recent acquisition of KVM author Qumranet. KVM is also available through SUSE; however, as of the date of authoring provides only experimental code in their enterprise OS release, which is SLES 11:

http://www.novell.com/linux/releasenotes/x86_64/SUSE-SLES/11/

KVM can also be used for desktop-class Linux distributions, such as those listed at:

https://help.ubuntu.com/community/KVM

The most notably commercialized offering using KVM is Qumranet/RedHat's virtual desktop offering. This package provides a comprehensive set of tools, including a connection broker and centralized management utility, for managing a KVM-based virtual machine environment. A key differentiator of this virtual desktop offering is the inclusion of a high-speed remote display protocol called SPICE. See Figure 6-1.

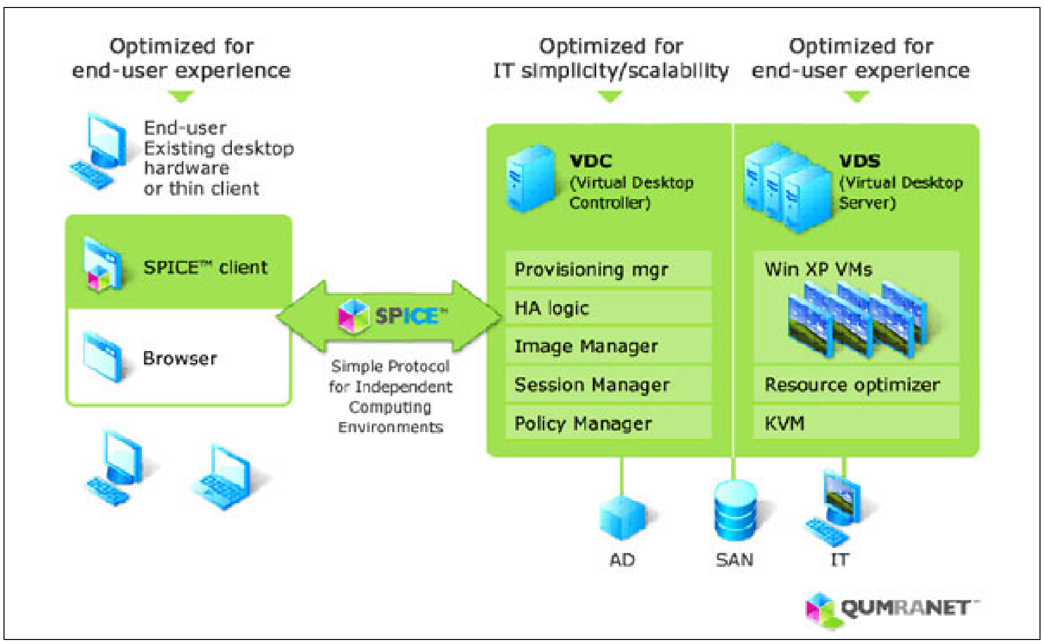


Figure 6-1 Virtual desktop package

KVM supports a full range of guest operating systems including Windows and Linux and is also supported by large list of open source hypervisor management utilities. More detailed support information can be found on the KVM community Web site:

http://www.linux-kvm.org/page/Main_Page

6.2 Scalability

The scalability and capabilities of KVM-based virtualization solutions have the most potential of any virtualization platform, but as of the time of this writing, are generally the most limited. The potential comes from the uniqueness of the KVM architecture. KVM can take advantage of the robustness and feature-rich option of the Linux kernel, while maintaining the performance of a dedicated hypervisor.

At this time, RHEV for Servers supports up to 256 cores and 1 TB of RAM per host. The guest virtual machines can support up to 16 virtual CPUs and 256 GB of RAM. See the data sheet:

http://www.redhat.com/f/pdf/rhev/DOC049R4-RHEV-Overall-Datasheet.pdf

For high availability and resource distribution, up to 50 hosts per cluster are supported. RHEV for Servers includes support for advanced features like memory page sharing, disk thin provisioning and single image clones. Support is also being written for VM fault tolerance, or lock-step virtual machines. For more information about fault tolerance for KVM, see the Kemari project at:

http://www.osrg.net/kemari/

6.3 Considerations

Although KVM is early in enterprise level adoption, it is a readily available, open-source virtualization alternative and is easily accessible to existing Linux users. KVM will benefit and most likely expand in use because of tight integration with the standard Linux kernel through near term releases of major Linux vendors including Novell SUSE and Red Hat. It is easily viable today for development and test virtual server environments, and can be considered for HA and enhanced management requirements using Open Source and certain vendor-specific utilities (such as RHEL 5.4, which includes formal support of KVM). With wide support of both Linux and Windows guests, KVM might also be considered for server virtualization and will be increasingly feasible as enterprise Linux vendors improve management tool offerings.

The Qumranet/Red Hat software stack offers an attractive alternative for virtual desktop. It provides a complete solution, including the SPICE protocol as a differentiator for virtual desktop experiences that require accelerated graphics performance.

KVM will continue to grow in adoption and, because of its integration with the standard Linux kernel, is being investigated for a variety of applications. One example is used as a deployment target for cloud environments, such as being done for the cloud toolkit:

http://xcat.sourceforge.net/

A key implication of KVM's integration with the standard Linux kernel is that it provides a simple transition to a virtualized environment for users with existing Linux skills or existing Linux server deployments. In addition, use of standardized virtualization modules enables simpler validation on the part of both hardware and OS vendors. This ease of use provided by the integration with Linux kernel, in addition to Type 1 performance, positions KVM a strong hypervisor competitor and cost effective solution for x86 virtualization.

7

Citrix XenServer

The Xen hypervisor is a unique open source technology, developed collaboratively by the Xen community spearheaded by XenSource and engineers at over 20 of the most innovative data center solution vendors, with Citrix and IBM as the most active contributors.

7.1 Introduction

Originally a research project about x86 virtualization at the University of Cambridge Computer Laboratory, Xen quickly outgrew its research roots. A private company, XenSource, was founded to coordinate development, promote the Open Source project, and release a commercial edition of Xen.

Citrix Systems completed the purchase of XenSource in 2007. Development of Xen has been extremely rapid, with Citrix following a twice yearly release schedule.

Although Xen is owned by XenSource, the nature of Open Source software ensures there are multiple forks and distributions, many released by other vendors. The most commonly seen versions are:

- Xen: The Open Source version released by Xen Source
- Commercial releases by:
  - Citrix Systems: Citrix Xen Server
  - Oracle: Oracle VM (Virtual Iron)
  - Oracle: Sun xVM

Xen is also featured as the default hypervisor in most Linux distributions, although certain vendors (most notably Red Hat) have recently shifted their focus to the KVM hypervisor. KVM is featured in Chapter 6, “Red Hat Enterprise Virtualization (KVM)” on page 29.

XenSource has had a close relationship with Microsoft to drive industry standards within virtualization. This extends to interoperability between Microsoft and Xen guests, allowing the best choice of hypervisor to be used as a host while being virtual.

At the time of writing, Xen Server v5.5 is the most widely supported version, with the release of V6 expected in 2010.

7.2 Technical

Xen is an open source *bare metal* (native) hypervisor, released under the GPL2 license by XenSource. In its early releases, the Xen hypervisor introduced the concept of paravirtualization to the x86 platform. Initially a feature of the IBM System/360 in the 1970s, paravirtualization requires the modification of the guest OS so that it is aware of the hypervisor below it.

Since the release of Xen Server 3.0, hardware-assisted virtualization has been supported through the use of Intel's VTx and AMD's AMD-V, which are integrated into modern x86/x64 processors. Although significant differences exist in the instruction sets and therefore the implementation in both technologies, the use of a common Xen abstraction layer permits the use of unmodified hosts. Hardware assisted virtualization simplifies the deployment of hosts running proprietary operating systems such as MS Windows where modification of the kernel to support paravirtulisation may be problematic.

Microsoft Hyper-V and XenServer share similar architectural designs. The key difference is that within Xen, Dom0 (the privileged domain that runs the Virtual Machine Monitor and manages I/O) is an optimized Linux kernel; Microsoft utilizes the kernel or Microsoft Windows Server 2008 for their Parent Partition (similar to Dom0 in Xen). These similarities and adherence to open standards allows the optional Citrix Essentials package to enable dynamic virtual machine migration between Microsoft Hyper-V and Citrix XenServer.

7.3 Citrix XenServer 5.5

Citrix released XenServer v5.5 in June 2009 with a large number of optimizations. Most notable are significant upgrades to the management and storage infrastructure. The rapid release schedule of upgrades since the Citrix purchase of XenSource is increasing Citrix XenServer's market acceptance. The schedule also allows Citrix to incorporate the latest technologies in the upstream Xen Kernel (future versions) from the open source community on a timely basis. This is an important strategy in a market as rapidly evolving as the virtualization market.

Although Citrix XenServer offers a competitive virtualization solution for today's enterprise customers, a notable omission from XenServer v5.5 is the ability to over commit memory resources. Memory overcommitment (Memory Ballooning) enables higher levels of consolidation, especially with respect to virtual desktops. Without the ability to overcommit memory resources Citrix XenServer is unable to consistently achieve the consolidation ratios seen on similar products, like VMware's vSphere and Red Hat's Enterprise Virtualization (KVM) offerings. The plan is for Citrix to add memory overcommitment in an upcoming version of XenServer. At present, the Xen Kernel 3.4, which is incorporated within the current open source Xen Hypervisor, has the following new features (all expected to ship in upcoming versions of Citrix XenServer):

- Device pass-through improvements: particular emphasis on support for client devices (further support is available as part of the XCI project)
- RAS features: CPU and memory offlining
- Power management: improved frequency/voltage controls and deep-sleep support; scheduler and timers optimized for peak power savings
- Support for the Viridian (Hyper-V) enlightenment interface

Additionally, VM Fault Tolerance (FT) is in release candidate at the time of this writing. FT is the ability to run a virtual machine instance on one physical host and a separate (hidden) instance, lock-step, on a separate physical host. In the event of a failure on the primary host, the secondary host virtual machine instance takes over, with no downtime to the virtual machine or user services. This technology was first incorporated in VMware's latest vSphere offering.

7.4 Licensing

Citrix has a simple and attractive licensing plan for XenServer. The core hypervisor and management console are available free of charge from the Citrix Web site. Customers requiring more advanced functionality can purchase one of two *Essentials* packages, which add significant functionality to XenServer.

The Essentials packages are licensed per server, not per core or processor. This approach makes them particularly compelling from a scale-up perspective when hosted on scalable architectures such as IBM eX4 and future systems.

7.5 Citrix XenServer versions

Citrix has released XenServer in three editions, with the core elements being free of charge and the additional functionality being available within two Essentials packs. This availability allows organizations to tailor their licensing and, if required, mix and match between the three tiers for differing requirements, while centrally managing all hosts within a single instance of XenCenter. See Table 7-1.

Table 7-1 XenServer versions

Feature	**XenServer (Free of charge)**	**Essentials for XenServer, Enterprise Edition**	**Essentials for XenServer, Platinum Edition**
Native 64-bit Xen hypervisor, supporting unlimited Windows and Linux guests VMs, and CPUs	✓	✓	✓
XenCenter management console	✓	✓	✓
XenMotion live migration	✓	✓	✓
Historical performance reporting, with e-mail alerting for performance and errors	-	✓	✓
High availability	-	✓	✓
Integrated storage management with StorageLink	-	✓	✓
Workflow Orchestration with Workflow Studio	-	✓	✓
Dynamic provisioning services (virtual only)	-	✓	✓
Dynamic provisioning services (physical and virtual)	-	-	✓
Automated lab management	-	-	✓
Stage Management	-	-	✓

7.6 The Citrix Xen product stack

The Xen hypervisor is the core of XenServer. It runs at the lowest and most privileged level and is the basic abstraction layer of software that sits directly on the hardware below any operating systems. It is responsible for CPU scheduling and memory partitioning of the various virtual machines running on the hardware device. The hypervisor abstracts the hardware for the virtual machines and also controls the execution of virtual machines as they share the common processing environment.

Above the hypervisor, run one or more guest operating systems. At installation time, the first guest VM is created. Known as Domain 0 (Dom0), this VM has privileged management access to the hypervisor and direct access to the physical hardware. Its primary role is to act as a secure management interface to the hypervisor.

Recent additions have significantly improved the Xen hypervisor. However one of the most important additions was full support for Microsoft's Hyper-V interface. As discussed earlier in this chapter, Hyper-V and XenServer share a common architecture. Cooperation between Microsoft and Citrix allows a Xen-enabled Linux guest running on Hyper-V to use the native

Xen hypercalls to access virtualization functions. A Microsoft-supplied software component called the Hypercall Adapter coverts the XenServer hypercalls to Hyper-V hypercalls so that the Linux guest can run with full kernel paravirtualization support. With the addition of the Hyper-V edition of Citrix Essentials there is even support for XenMotion between running XenServer hosts and Hyper-V.

Citrix XenServer v5.5 runs the 3.3 Xen kernel, which bought many significant improvements to the Xen Hypervisor:

- Power management (P & C states) in the hypervisor
- Hardware-assisted paging enhancements: Offers 2 MB page support for better TLB locality.
- CPUID feature leveling: Allows safe domain migration across systems with various CPU models.
- PVSCSI drivers for SCSI access direct into paravirtualized guests
- Full x86 real-mode emulation for HVM guests on Intel VT: Supports a much wider range of guest operating systems that are already available.

The next Citrix release of XenServer is expected to incorporate the new 3.4 version of the Xen Kernel.

As discussed earlier, further enhancements are already in the works for future versions of the Xen Kernel.

Within the proposed Xen 4.0 kernel are:

- RDMA live migration support
- Dom0 kernel in Linux 2.6.30 or later
- Dom0 support for Marvell 6480 disk driver
- Pass through USB controllers/devices for PV guests
- Fault tolerance: Project Remus, Kemari, or both
- Monitor, limit, control network traffic coming at DomUs
- Internationalization, Unicode support
- Configure virtual bridge like real switch (such as control VLAN, port status)
- VLAN tagging per NIC in the VM configuration file
- Virtual Ethernet Switch
- Physical Xen boot/install support from native UEFI (pUEFI); virtual UEFI (vUEFI) support
- Limit I/O for individual disks of VM (similar to credit scheduler weight)
- Dynamic Memory Management for overcommiting RAM
- PCI CGA pass-through for VT-d (vendor cards such as Nvidia, AIT, and so on)
- Full AMD IOMMU Support
- Online resizing of DomU Disks
- Cross compiling Xen and Modular builds

Citrix Xen Server 5.5 requirements

Citrix Xen Server 5.5 has the requirements, in Table 7-2, for the XenServer host. However configurations beyond these limits might be supported by XenSource in the open source editions, as shown in the table.

Table 7-2 Open source editions

Component	Requirements
CPU	One or more 64-bit x86 CPU(s), 1.5 GHz minimum, 2 GHz or faster multicore CPU recommended. To support VMs running Windows, an Intel VT or AMD-V 64-bit x86-based system with one or more CPUs is required (up to 32 CPU cores supported).
Memory	1 GB minimum, 2 GB or more recommended (up to 128 GB supported)
Disk	16 GB of disk space minimum, 60 GB of disk space recommended
Network	100 Mb or faster, 1000/T connection recommended (up to 6 NICs supported)

Citrix XenServer v5.5 supports all current core x86 operating systems. However it is aimed predominately at the Microsoft Windows market. If support is required for paravirtualized Linux guest operating systems, the optional Linux support packages must be installed. These allow Linux base operating systems to operate at Dom0. Without this Linux pack, you must run all operating systems in a hardware-assisted mode. At the time of installation, you are given the option to install the optional Linux Pack from a second CD, as shown in Figure 7-1.

Figure 7-1 Installing the optional Linux Pack

If you choose not to install the Linux pack at this stage, you can install it at a later time from the Linux Pack installation CD or ISO image on the XenServer host and running the `install.sh` script either from a command line within XenCenter or directly on the host itself.

Although Table 7-3 shows the supported operating systems list for Citrix Xen Server, many other operating systems can successfully run as guests and are supported within the open source version releases.

Table 7-3 Supported guest operating systems

	Supported Linux guest operating systems	**Supported Microsoft Windows guest operating systems**
32 bit	**Linux 32-bit:** ► Red Hat Enterprise Linux 3.5, 3.6, 3.7, 4.1, 4.2, 4.3, 4.4, 4.5, 4.6, 4.7, 5.0, 5.1, 5.2 ► CentOS 4.1, 4.2, 4.3, 4.4, 4.5, 4.6, 4.7, 5.0, 5.1, 5.2 ► Oracle Enterprise Linux 5.0, 5.1, 5.2 ► Novell SUSE Linux Enterprise Server 9SP2, 9SP3, 9SP4, 10SP1 ► Debian sarge (3.1), etch (4.0)	**Microsoft Windows 32-bit:** ► Windows Server 2008 ► Windows Server 2003 Web, Standard, Enterprise, Datacenter SP0/ SP1/SP2/R2 ► Windows Small Business Server 2003 SP0/SP1/SP2/R2 ► Windows XP SP2, SP3 ► Windows 2000 SP4 ► Windows Vista original and SP1
64 bit	**Linux 64-bit:** ► Red Hat Enterprise Linux 5.0, 5.1, 5.2 ► CentOS 5.0, 5.1, 5.2 ► Oracle Enterprise Linux 5.0, 5.1, 5.2 ► Novell SUSE Enterprise Linux 10SP1, 10SP	**Microsoft Windows 64-bit:** ► Windows Server 2008 ► Windows Server 2003 Standard, Enterprise, Datacenter Edition SP

7.7 XenCenter

XenCenter is the powerful centralized multi-server management console for Citrix XenServer. It provides a powerful multi-server administrative console for all your XenServer resource pools, hosts and virtual machines, with integrated storage management capabilities. Unlike other virtualization management servers, XenCenter is a lightweight *pane of glass* Microsoft Windows application that runs on any desktop, with configuration and performance data replicated across all managed servers in a pool. This approach eliminates potential single points-of-failure and cost from the virtualization infrastructure. If the server being used to manage a pool fails, the function can be restarted on any other server in the pool, either manually with free XenServer or automatically with the high availability functions of Citrix Essentials for XenServer. XenCenter can also deliver performance information in an easy-to-track graphical form, and enables flexible views of the infrastructure through searching, sorting, and tagging.

The ability to move virtual machines between members of a pool live and online with XenMotion enables IT staff to update and manage servers without interrupting user access to virtual machines, essentially eliminating interruptions because of planned downtime. Virtual machines in shared storage can be migrated live to any other server in the resource pool with sufficient capacity. Unlike most competing solutions, Citrix XenCenter with XenMotion functionality is available free of charge. With the addition of Citrix Essentials for XenServer, XenCenter becomes even more powerful by the addition of the advanced feature sets shown in Table 7-1 on page 36.

The minimum system requirements for Citrix XenServer are:

- x86-based system
- Microsoft Windows 2000, Windows XP, Windows Server 2003, Windows Server 2008, or Windows Vista
- .NET Framework 2.0 or later
- CPU speed: 750 MHz minimum, 1 GHz or faster recommended
- RAM: 512 MB minimum
- Disk space: 100 MB minimum
- Network interface card

Figure 7-2 shows the components that can be controlled by the XenCenter console.

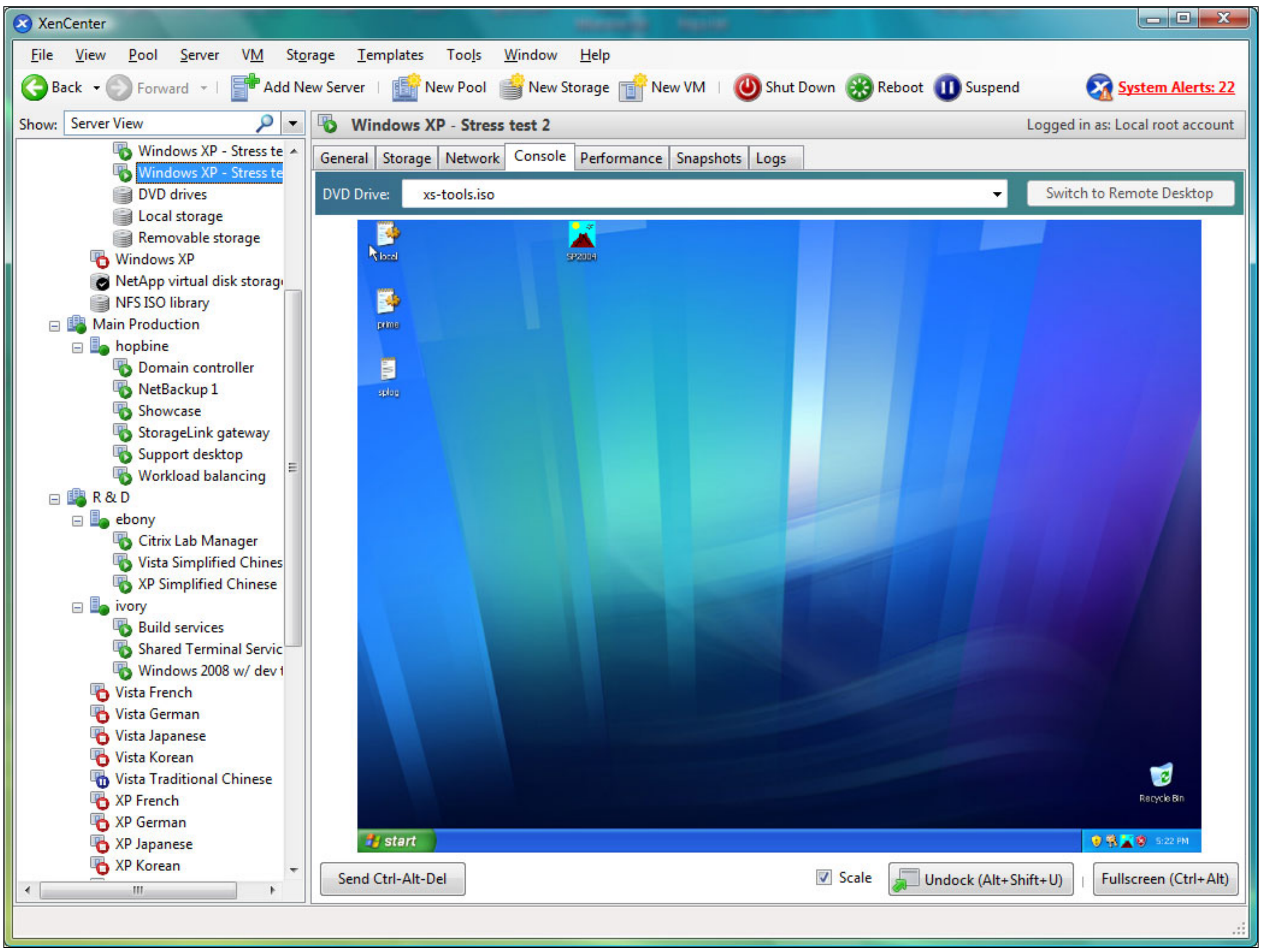


Figure 7-2 XenCenter console

8

IBM Systems Director VMControl

IBM Systems Director VMControl Version 2.1 is a progressive set of capabilities that simplify the management of virtual environments across multiple virtualization technologies and physical platforms. IBM Systems Director VMControl delivers leadership heterogeneous virtualization management and enables an infrastructure for cloud computing.

8.1 Overview

IBM Systems Director VMControl V2.1 represents a transformation from managing virtualization to using virtualization to better manage your IT infrastructure. IBM Systems Director V6.1 and IBM Systems Director VMControl V2.1 help reduce the total cost of ownership in a virtual environment by decreasing management costs, increasing asset utilization, and linking infrastructure performance to business goals.

IBM Systems Director VMControl Version 2.1 is a progressive set of capabilities to simplify the management of virtual environments across multiple virtualization technologies and physical platforms. IBM Systems Director VMControl delivers leadership heterogeneous virtualization management through IBM Systems Director VMControl Express and Standard Editions. Table 8-1 shows the features contained in each version.

Table 8-1 Features comparison by version

IBM Systems Director VMControl features	Express	Standard
Create and manage virtual machines	✓	✓
Virtual machine relocation	✓	✓
Virtual machine relocation	-	✓
Deploy virtual images	-	✓
Maintain virtual images in a centralized library	-	✓

8.2 IBM Systems Director VMControl 2.1 Express Edition

IBM Systems Director VMControl 2.1 Express Edition encompasses virtual server life cycle management as a no charge plug-in to IBM Systems Director. IBM Systems Director VMControl helps to better manage your IT infrastructure by providing:

- A single and consistent solution for all IBM Systems
- Radically improved time to value for new solutions
- Repeatable accuracy and consistency with automation

IBM Systems Director VMControl Express Edition for life-cycle management features enable you to:

- Discover virtual resources.
- Display inventory and topology.
- Monitor virtual resource health.
- Relocate virtual resources.
- Create and manage virtual servers.

8.3 IBM Systems Director VMControl 2.1 Standard Edition

IBM Systems Director VMControl Standard Edition (Image Manager V2.1), a priced feature of IBM Systems Director VMControl, simplifies the management of virtual environments for multiple virtualization technologies and physical platforms, helping to reduce the cost of managing a virtualized environment. IBM Systems Director VMControl Standard Edition captures information from active systems and stores the information in a repository as reusable system images, also referred to as *virtual appliances*. IBM Systems Director VMControl Standard Edition provides support to manage virtual appliances and automate deployment of virtual appliances from a centralized location.

Benefits of IBM Systems Director VMControl Standard Edition, help you to:

- Eliminate the installation, configuration, and maintenance costs that are associated with running complex stacks of software.
- Reduce operational and infrastructure costs because of increased efficiency in using IT resources.
- Manage a library of ready-to-deploy system templates that meet specific hardware and software requirements.
- Store, copy, and customize existing images to reuse them within system templates for creating virtual servers.
- Create a system template that is based on an existing virtual server and that includes an image in addition to running applications.
- Reuse the system template to create new similar virtual servers.

An overview of features includes:

- Simplified virtual server provisioning
- Self-contained images that decrease the cost of managing dependency problems
- Rapid deployment
- Common model for physical and virtual servers
- Bring image-based provisioning to physical servers

Value gained from improved system image management includes:

- Application can be up and running quickly
- Enhanced tooling to lower the test costs
- Centralized management of images versus management by individual system administrators
- Improved recoverability
- Updates are applied once to an image, and propagated to all its instances
- Reduced disruption of provisioning
- Reduced deployment errors and troubleshooting

For more information about IBM Systems Director VMControl, go to:

http://www.ibm.com/systems/management/director/plugins/

9

Conclusion

Within all new and emerging technologies there is a point of reflection as the market matures, when significant consolidation of the technologies occurs, and leading players emerge. This point-of-reflection often coincides with the standardization of one or more technologies. Within the IBM System x86 virtualization market, VMware has become that standard. However, unlike the video recording battles between VHS and Beta, or more recently BlueRay and HD-DVD, there is room for more than one technology in the market, each serving a different requirement.

The latest virtualization offerings from Red Hat, Microsoft, Citrix, and others are beginning to take shape. Most large companies have some amount of virtualization in place today, and are rapidly increasing that footprint. VMware is still the dominant market leader, but the other offerings are carving out niches. Red Hat's Enterprise Virtualization offering has already become popular as the basis for many cloud solutions; Microsoft is gaining significant traction with the small, medium business (SMB) market space; and Citrix is a leading alternative for companies that want the features, reliability, and backing of a stable company but lower costs for an enterprise virtualization solution.

Related publications

The resources listed in this section are provided for a more detailed discussion of the topics covered in this paper.

Online resources

These Web sites are relevant as further information sources:

- Parallels Virtuozzo Containers

  http://www.parallels.com/products/pvc45/

- Popular examples of Type 2 hypervisors

  http://www.vmware.com/products/workstation/index.html
  http://www.virtualbox.org/
  http://www.microsoft.com/windowsserversystem/virtualserver/

- VMware products

  http://www.vmware.com/products/

- VMware maximum supported configurations

  http://www.vmware.com/support/pubs/vs_pages/vsp_pubs_esx40_vc40.html

- The vSphere 4.0 announcement

  http://www.vmware.com/support/vsphere4/doc/vsp_40_new_feat.html

- Microsoft Virtualization

  http://www.microsoft.com/virtualization/default.mspx

- Microsoft System Center

  http://www.microsoft.com/systemcenter/en/us/default.aspx

- Hyper-V Architecture Demonstration video

  http://cid-17e0c6128f544f9e.skydrive.live.com/self.aspx/Public/Hyper-V%20Architecture%20Demonstration%20Video%20Streaming.wmv

- Windows Virtualization Team Blog

  http://blogs.technet.com/virtualization/archive/2009/05/12/tech-ed-windows-server-2008-r2-hyper-v-news.aspx

- Red Hat Enterprise Virtualization

  http://www.redhat.com/virtualization/rhev/

- SLES 11

  http://www.novell.com/linux/releasenotes/x86_64/SUSE-SLES/11/

- Linux distributions

  https://help.ubuntu.com/community/KVM

- KVM community

  http://www.linux-kvm.org/page/Main_Page
- RHEV data sheet

  http://www.redhat.com/f/pdf/rhev/DOC049R4-RHEV-Overall-Datasheet.pdf
- Cloud toolkit

  http://xcat.sourceforge.net/
- IBM Systems Director VMControl

  http://www.ibm.com/systems/management/director/plugins/
- Kemari project

  http://www.osrg.net/kemari/

How to get Redbooks

You can search for, view, or download Redbooks, Redpapers, Technotes, draft publications and Additional materials, as well as order hardcopy Redbooks publications, at this Web site:

ibm.com/redbooks

Help from IBM

IBM Support and downloads

ibm.com/support

IBM Global Services

ibm.com/services

System x Virtualization Strategies

Redpaper™

Enterprise class virtualization solutions

eX4 hardware architectural features

Software virtualization platforms

IBM is the global leader in enterprise class virtualization solutions. IBM offers a full range of virtualization hardware, including the IBM Enterprise X Architectures (eX4 and eX5), as well as software. As more modern computing is becoming open and standards-based, IBM is also a global partner with both hardware and software vendors of virtualization technologies, such as VMware's ESX.

This IBM Redpaper publication discusses many of the major virtualization environments that run on IBM System x, particularly the eX4 architecture. The audience for this paper includes decision-makers, consultants, architects, and planners who want to consider the benefits of virtualization on System x.

REDP-4480-00