IBM @server

Redbooks Paper

Massimo Re Ferre’

VMware ESX Server: Scale Up or Scale Out?

Table of contents:

Introduction

This IBM® Redpaper presents a basis for discussion about optimal and efficient implementations of ESX Server, the flagship VMware product for x86 hardware virtualization. We consider best practices regarding the scalability of an ESX Server implementation on IBM @server® xSeries® and BladeCenter™ servers.

In this document, we focus the discussion on two major scenarios. The *scale-up* implementation applies to architecture based on one or few large x86 server systems and involves adding components and resources to the system to achieve scalability. The *scale-out* implementation is based on many smaller x86 server systems means adding new server systems to your “server farm” to scale it according to your needs.

We assume that readers of this paper have an understanding of the VMware products, specifically ESX Server, VirtualCenter, and VMotion. For more information about VMware products, refer to the documentation on the VMware Web site:

http://www.vmware.com

Related IBM Redbooks and publications are available at http://www.redbooks.ibm.com.

IBM offers a wide range of products that could work with either of the scenarios we describe. Specifically, we typically leverage the vertical scalability characteristics of the xSeries 445 when implementing scale-up requirements, and the IBM BladeCenter or other low-end or mid-range Intel® servers when dealing with scale-out requirements.

As you read through this document, keep in mind that the content is a merely point of view based on our product expertise and field experience: This is not a “must-do” document, but a base for you to start building your own unique solution to fit your own unique requirements.

The information and thoughts in this document are based on availability as we write and might change in months or even weeks as new technologies, products, solutions, and experiences become available.

VMware virtual infrastructure

VMware has been promoting the concept of the *virtual infrastructure* for a while. The idea, at its essence, is to create a virtual Intel server farm from which to deploy applications and operating systems in a very dynamic way. These concepts are fully complementary with the IBM on demand strategy, in which virtualization is one of the core elements in achieving IBM goals. From this perspective, VMware ESX Server is a technology enabler that we are leveraging to implement our strategy.

One of the key attributes of the VMware virtual infrastructure (Figure 1) is that it enables administrators to manage their distributed Intel environment independently from the actual physical servers that are being used to provide computing resources such as CPU, memory, network, and disk. We achieve this by implementing two core VMware technologies:

- VMware ESX Server, the core virtualization technology, which enables you to “chunk” your real hardware in many virtual servers
- VMware VMotion, the abstracting technology, which enables you to move your running virtual machines on-the-fly from one physical server to another with near-zero service interruption

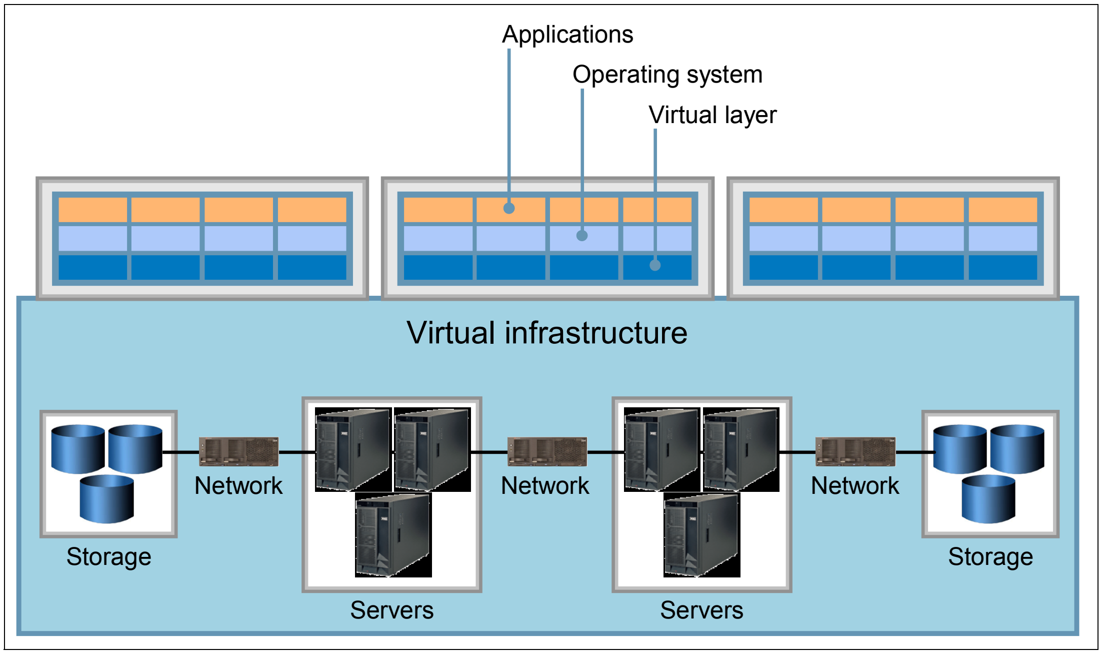


Figure 1 The VMware virtual infrastructure

The combination of these two technologies can reduce the need for the Windows® or Linux® administrator to manage at the hardware level. At that point, the physical servers being used to create the virtual infrastructure are treated simply as a bunch of hardware because the combination of ESX Server and VMotion can (potentially) abstract you from the real hardware being used. The idea now is that, from this perspective, a single physical 16-way xSeries server could be comparable to eight 2-way blades with VMotion (which, in turn, all together would become a virtual 16-way server).

There is actually a “feed the infrastructure” concept coming into place now: If the administrator realizes the need for 16-way raw computational power, a potentially option is to feed the virtual infrastructure with either one 16-way server, two 8-way servers, or even as many as eight 2-way servers. Being able to de-couple the applications and the operating systems from real server systems is, in fact, one of the key advantages of the VMware virtual infrastructure.

Figure 2 illustrates the virtual infrastructure concept.

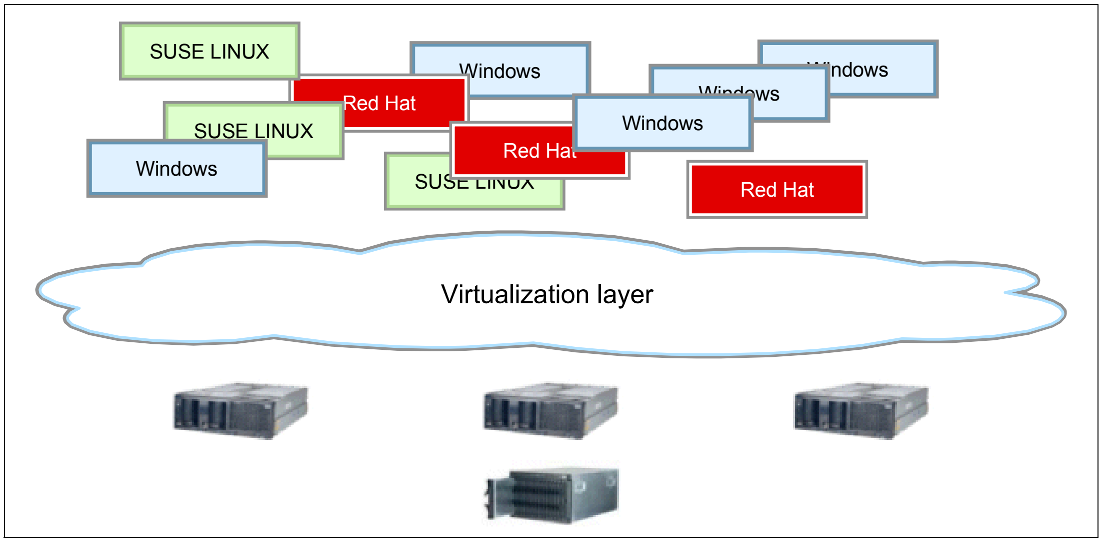


Figure 2 The Virtual Infrastructure physical implementation

If this infrastructure requires an upgrade because you need more resources (CPUs, memory, I/O) to support new applications and new environments, you no longer need to map these new applications and services to new real hardware; instead, you can add raw physical resources to this virtual infrastructure and use them via the virtualization layer.

Readers who are familiar with mainframe systems will realize that this approach is similar to what they do when they buy mips (millions of instructions per second, historically a mainframe term indicating processor capacity) to upgrade their systems: A mainframe customer usually does not buy a set of mainframes for a given project to support new applications and services, but typically upgrades the mainframe infrastructure with more resources. Therefore, a hardware upgrade to the infrastructure above would look similar to Figure 3 on page 4.

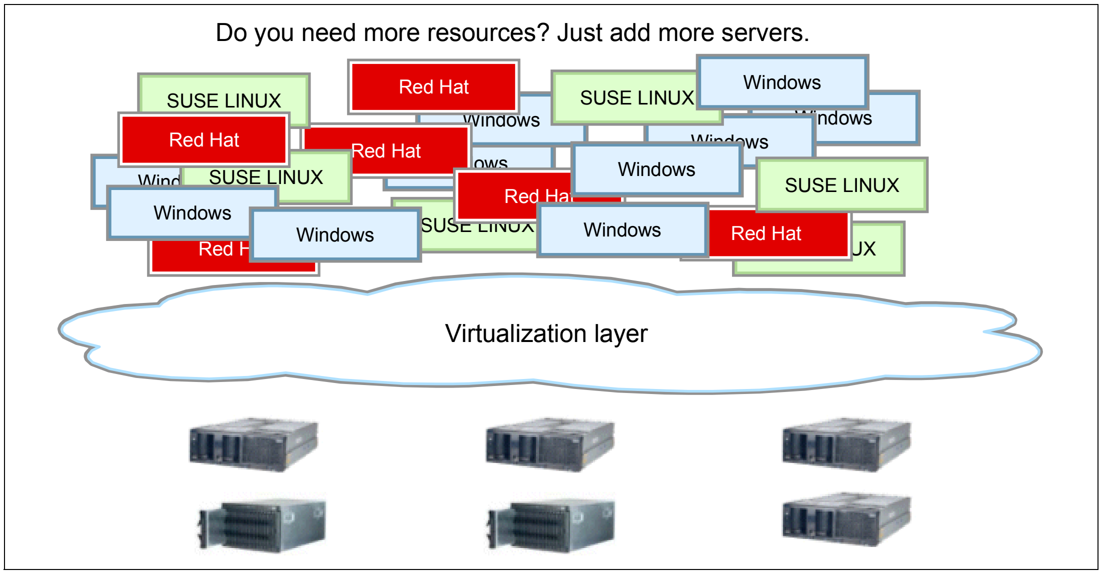


Figure 3 The Virtual Infrastructure expands on demand

Readers might be concerned that, when using virtualization software to streamline and keep hardware resources under strict control, they could risk losing control over the applications and operating systems layers. This is partially true in that pure virtualization is about consolidating at the hardware level (that is, you keep all of your operating system images unchanged in configuration and number). Other consolidation techniques may take into account a rationalization of the operating system layer (thus reducing the number of OS images), but they either are not feasible in most cases or are difficult to achieve.

In an ideal world, you might prefer a single enterprise server with a single operating system that runs all of your applications and services. The Intel world is quite different, as it tends to be distributed. This means that one server (hardware and operating system) typically maps to a single application or service. Virtualization technologies are useful for addressing the hardware proliferation problem but typically do not address the problem of maintaining many operating system images. To achieve this, you need a radical change in how an x86 environment is operated and how ISV develops applications on it. ISVs tend to develop applications with the "one server, one application" paradigm in mind, which is unlikely to change any time soon.

For more information about this topic and for other potential alternatives to hardware virtualization, refer to *IBM @server xSeries Server Consolidation: an Introduction* at:

`http://www.redbooks.ibm.com/abstracts/redp3785.html`

VMware VMotion

VMware VMotion is a technology that enables system administrators to move running virtual machines from one physical server to another without having to take any virtual machines offline. We assume that the reader has some familiarity with this technology; if not, refer to the proper VMware documentation available on their Web site for more information.

VMotion provides two macro-functions to ESX Server users:

- It allows for planned maintenance of a physical server without service interruption. In this case, you simply "move away," on-the-fly, VMs that are running on the physical server that will be taken offline for regular maintenance (hardware or software upgrade, for example).

- It (potentially) enables the building of a transparent virtual infrastructure that goes beyond the physical boundaries of the hardware being used.

Consider the example of a 16-way server configuration with a single image of ESX Server (a scale-up scenario). If you are loading 40 virtual machines on this server, those processes will be spread across the 16 processors by the ESX Server scheduler. Nothing else is required.

Or you can consider using eight independent 2-way servers (each with an ESX Server image) to host those 40 virtual machines. In this case, using the raw ESX Server technology, you have to deal with physical boundaries, so if a VM must be brought online on a different server with spare resources, that VM has to be restarted on that specific server.

VMotion provides the level of transparency that you can find on the 16-way server transparently moving a running VM from, for example, physical CPU number 4 to physical CPU number 13. Similarly, VMotion enables a system administrator to transparently move a VM from physical server number 2 to physical server number 7. You could say that VMotion is the abstraction layer that enables you to overcome physical server boundaries as if your whole infrastructure was a single server image.

In this paper, we assume that the requirement to have VMotion for balancing hardware resources and for regular maintenance of a blade infrastructure is far greater than the need to have VMotion for regular maintenance of an infrastructure built with few high-end servers.

In short, we assume that it is acceptable to restart virtual machines running on a high-end server in order to move them onto spare resources of other big servers for regular maintenance. We also assume that restarting virtual machines every time you need to rebalance resource utilization in your server farm is not desirable. If you also take into account that low-end server modules such as single blades might require more downtime for regular maintenance when compared to high-end redundant and scalable servers, this makes the choice of using VMotion not an option but a requirement. Table 1 shows a schematic summary of this concept.

Table 1 Scale-out versus scale-up characteristics

	Server blade	**High-end server**
Downtime for regular maintenance	More frequent	Less frequent
Requirement to cross physical boundaries to find spare resources	Required	Not required

From this table, we assume that VMotion is a must-have technology when using low-end servers as building blocks of your virtual infrastructure. We consider the use of VMotion as complementary or nice-to-have in environments where highly scalable servers are being used as building blocks of the virtual infrastructure.

High availability considerations

High availability (HA) is a means to provide a certain level of resiliency in the virtual infrastructure, if a compute node fails. (A compute node is a server that in turn hosts an ESX Server Console OS image.) It is important to note that although VMotion can address the guest OS resiliency during a planned downtime, the same technology may be useless during unplanned events.

The current VirtualCenter architecture actually exacerbates these HA issues because, assuming the best case of having the virtual machines' dsk files on a shared SAN, the virtual machines' vmx files are usually stored on the local ext3 volumes of the Console OS and these become unavailable along with the failing physical server they are hosted on.

Assuming that you decided to store those files on a central repository such as an NFS share that is visible to all ESX Server systems, you still have to deal with the problem of detecting a virtual machine (or an entire ESX Server system) failure and restarting it on another available server that comprises the virtual infrastructure. We are not even taking into account in this case that the NFS share would be yet another single point of failure for your virtual infrastructure and hence would require the application of its own HA technologies.

So, assuming again the best case of having the entire set of dsk files shared on the SAN, we can think of at least three ways to provide a certain level of resiliency to the infrastructure:

- The virtual machines are configured and registered manually on other available servers and are restarted manually with some downtime, which depends on how promptly the operators can respond to the crash event.
- The virtual machines are configured and registered via custom scripts that run on other available servers and are restarted automatically with minimal downtime.
- HA software is configured either at the guest OS level or at the Console OS level and can automatically trap a crash then fail over either an application or the whole virtual machine to, respectively, a different guest OS or a different ESX Server system.

As always, there are advantages and disadvantages. The third implementation might seem at first to be the best (for example, because potentially it could detect application failures) but it is more complex to set up. HA software implementation at the Console OS level might introduce incompatibilities with other features such as VMotion, and these implementations usually require strict certifications from both the ISV and OEMs that are difficult to match in an ESX Server environment.

As a result, we have seen that the first and second approaches are most widely used, although there are implementations of the third approach. It is important to note that as VirtualCenter matures and new features are included, it could be wise to expect a certain level of fail-over automation to be integrated and provided by the virtual infrastructure itself. Basically, with what most people are doing via scripting (as in the second approach), we should expect to see it built in as a feature of future VMware VirtualCenter releases. In fact, VMware announced at VMworld 2004 that they will introduce in 2005 such a feature under the name of DAS (Distributed Availability Services) as a plug-in to VirtualCenter.

Defining scale-out and scale-up

This is a challenging section; not specifically technically challenging, but because it is about interpretations. Historically, the concept of scaling out versus scaling up has been introduced to describe the use of many small low-cost servers versus fewer big expensive servers, with low-cost and expensive being the keywords. There are certainly technical differences between the two implementations but generally hardware cost is one of the key reasons to choose a scale-out approach because, computing power being equal, a single big SMP server usually costs more than many small servers.

It is easy to agree that when using 2-way blades you are implementing a scale-out approach, and when using a single 16-way x86 server you are implementing a scale-up approach. However, it is not easy to agree on which approach we are using if we compare 4-way to 8-way servers. We do not pretend to be able to give you this definition, as this is largely determined by the size of your whole infrastructure, your company strategies, and your own attitudes.

For this discussion, we will always refer to the extremes: 2-way blade servers for the scale-out approach and 16-way x445 for the scale-up approach. Although 4-way servers should fall into the scale-out category and 8-way configurations should fall into the scale-up approach, this will vary depending on your company's characteristics and infrastructure size.

It is possible to segment this market (scale-out servers versus scale-up servers) by costs and not by technical characteristics. CPU vendors, in this case Intel and AMD, tend to charge more for CPUs that are used in 4-way (and higher) servers (Xeon MP for Intel). So a 4-way system is built with the same Intel CPUs that a 16-way server is built with. Because, for the most part, CPU costs usually drive server cost, the investment in four 4-way servers is closer to one 16-way than to eight 2-way servers.

An exception is the IBM xSeries 445 Entry server, which uses Xeon DP processors in a 4-way configuration. Because of the modular NUMA design, we have specific 4-way xSeries 445 models that use the low-cost Xeon DP processors that Intel positions to use on 2-way Intel systems only.

Figure 4 summarizes this position/

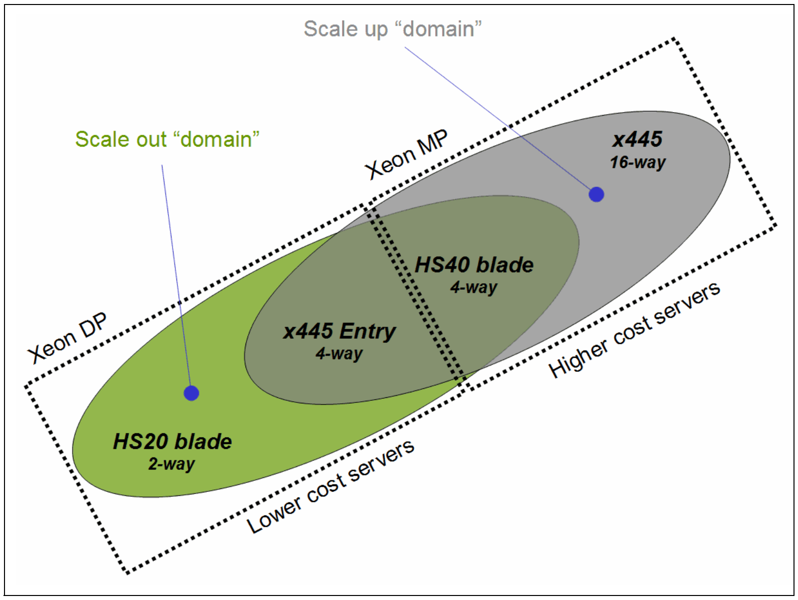


Figure 4 Server and processor positioning

Scale-up (fewer, larger servers)

In this section, we discuss the characteristics (including advantages and disadvantages) of a scale-up implementation approach in order to create or "feed" the infrastructure.

The IBM building block for a scale-up approach is the xSeries 445 which is a modular, super-scalable server that can drive as many as 16 processors in a single system image. In our case, "single system image" means a single instance of VMware ESX Server with a single Console OS (put simply: one ESX Server system).

Figure 5 on page 8 illustrates a 16-way configuration using xSeries 445 modules in a single system image.

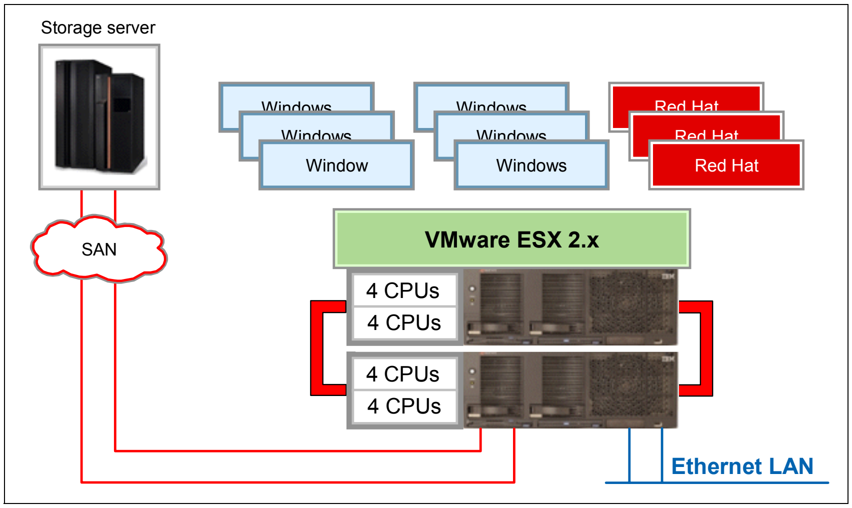
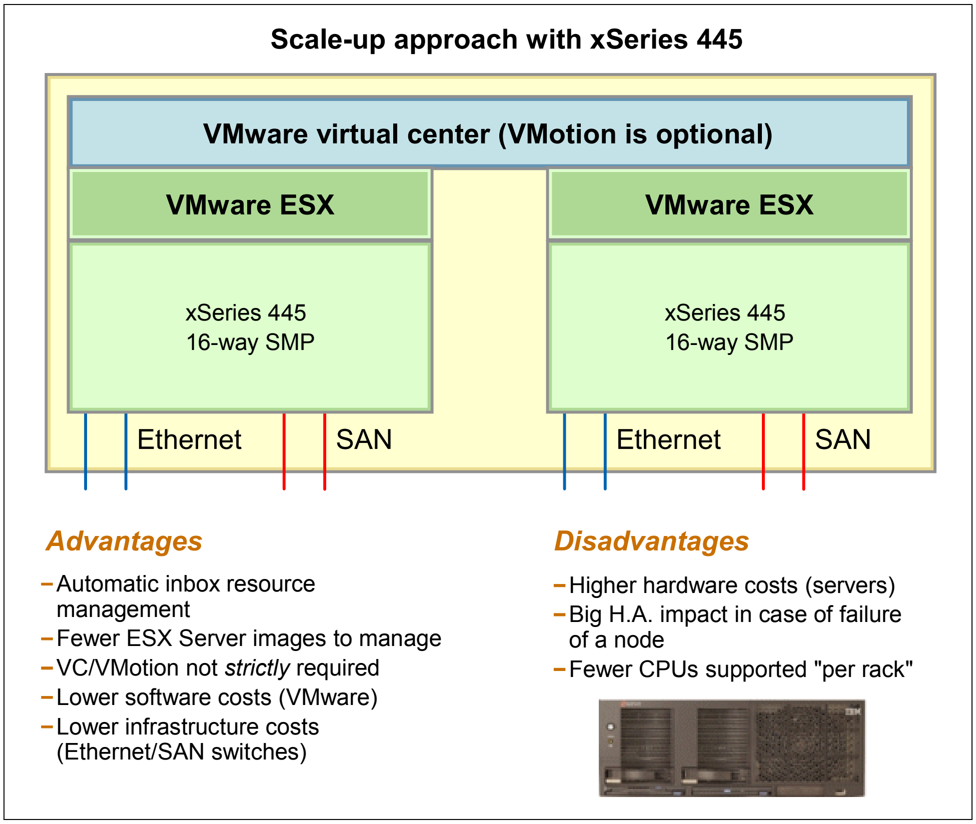


Figure 5   Scale-up implementation

In this scenario, all 16 processors are configured so that ESX Server can be installed once and drive all of the available resources. (Note that the number of SAN/Ethernet connections in this picture is merely an example).

Figure 6 shows how a (simplified) scale-up approach would be implemented.



Scale-up approach with xSeries 445

VMware virtual center (VMotion is optional)

VMware ESX	VMware ESX
xSeries 445 16-way SMP	xSeries 445 16-way SMP
Ethernet SAN	Ethernet SAN

Advantages

- Automatic inbox resource management
- Fewer ESX Server images to manage
- VC/VMotion not *strictly* required
- Lower software costs (VMware)
- Lower infrastructure costs (Ethernet/SAN switches)

Disadvantages

- Higher hardware costs (servers)
- Big H.A. impact in case of failure of a node
- Fewer CPUs supported "per rack"

Figure 6   Scale-up characteristics

Figure 6 lists advantages and disadvantages upon which the industry generally agrees. Of course, one reader might rate fewer ESX Server images to manage as a key advantage while another reader might not be concerned about having to deal with 30 ESX Server system images. Even if we could try to give each of these a weight based on our experience, we realize that this will be different from customer to customer.

A typical concern regarding this kind of high-end configuration is performance and scalability. This is a real concern in most of the scenarios where a single application tries to benefit from all available resources. Usually there is a scaling penalty because some applications are written in a way that limits their ability to generate more workload to keep the server busy.

With ESX Server implementations, this is not usually a concern. In fact, we are not dealing here with a single application running on the high-end system but rather with multiple applications (and OS images) running on the same high-end systems. Because the virtualization layer is very efficient, if server is not being fully utilized you can add more virtual machines to result in more workload for the server. This has proven to scale—if not linearly, then very close to that.

The xSeries 445 leverages the NUMA architecture, which enables administrators to add CPU-memory bandwidth as they add CPUs to the configuration. ESX Server fully exploits the NUMA architecture of the xSeries 445. VMware has confirmed that the x445 is the only NUMA topology they recognize and for which they activate the NUM-aware algorithms in the VMkernel scheduler. Other NUMA topologies and implementations such as the Opteron and its HyperTransport, as of today, are not recognized and optimized for that. (Of course, this might change over time as ESX Server matures.)

It is important to note that, although the core ESX Server product is licensed per CPU (so that there is no difference between licensing one 8-CPU server or four 2-CPU servers), in this scenario you can, potentially, save money in buying fewer software modules because you can opt not to use VMotion (which we defined as a nice-to-have).

Scale-out (more, smaller servers)

This section describes the advantages and disadvantages of implementing the VMware infrastructure with many small servers, each running ESX Server. Although many options exist for implementing a farm comprised of small low-end servers, we consider the use of the IBM BladeCenter the most viable alternative when discussing this requirement. BladeCenter is the name of the powerful integrated chassis that, along with many other infrastructure components such Ethernet and SAN switches, contains the IBM HS20 and IBM HS40 blades (the 2-way and 4-way Intel-based blade servers).

All of our discussions about the IBM blades can apply to other 2-way (and 4-way) traditional servers, either rack or tower, but we think that the use of blades makes more sense in a scale-out implementation such as this.

Figure 7 on page 10 shows a typical ESX Server deployment on an IBM BladeCenter.

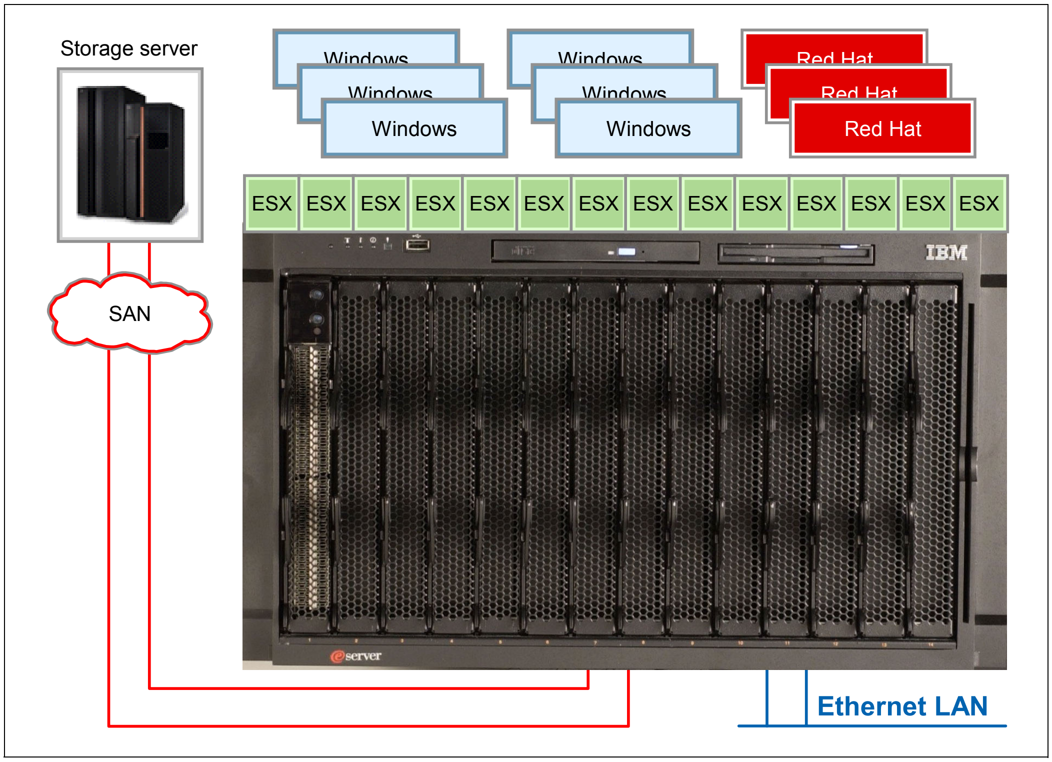


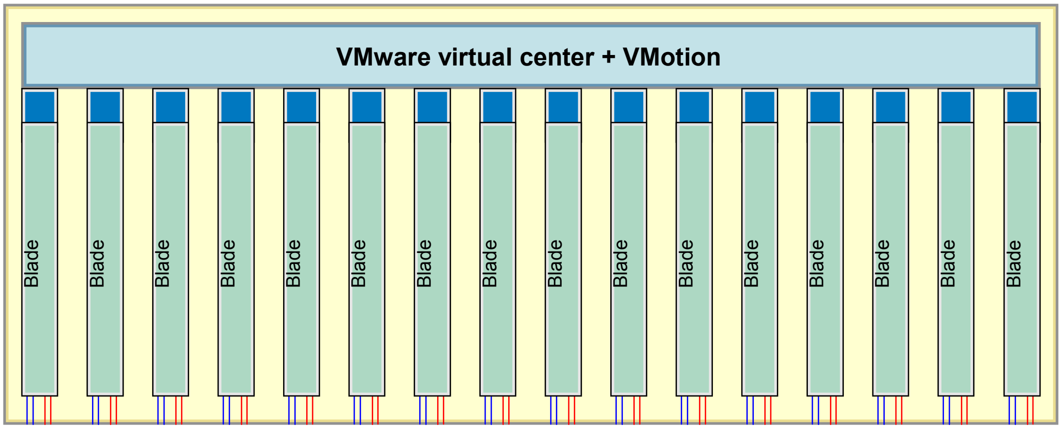
Figure 7 Scale-out implementation

Although from a graphical perspective the two solutions (xSeries 445 and BladeCenter) look similar, there is one key differentiator between the two: The 16-way x445 is operated by a single ESX Server image, but in this scenario every blade must be driven by its own ESX Server image, resulting in 14 ESX Server installations to drive 14 independent HS20 blades.

As anticipated, this would not make much difference if we were to use 14 standard 2-CPU servers instead of blades. As a result, we would have had 14 ESX Server installations to operate 14 independent tower or rack servers.

Figure 8 on page 11 shows how a (simplified) scale-out approach would be implemented.

Scale-out approach with IBM BladeCenter



Advantages

- Lower hardware costs (servers)
- Low H.A. impact in case of failure of a node
- More CPUs supported "per rack"

Disadvantages

- Manual across-boxes resource management
- VC/VMotion mandatory for reasonable management
- Many hypervisor (ESX) images to maintain
- Higher software costs (VMware)
- Higher infrastructure costs (Ethernet/SAN switches)

Figure 8 Scale-out characteristics

This approach has a number of advantages, including the high availability and resiliency of the infrastructure. If a single “module” fails (be it blade hardware or the ESX Server image), you lose only a small part of your virtual infrastructure as opposed to what would happen if you lost an entire 16-way server (which is supposed to run eight times the number of virtual machines that the 2-way blade supports). Another easy advantage to notice is server costs: Eight 2-way blades usually cost less than a single 16-way server due to the premium price associated with big SMP servers.

One drawback to the proposed solution is the management of all of those Console OS images. For example, think of a minor or major upgrade to an ESX Server: You would have to perform that on every blade. Or, at the moment, a system administrator has to invoke VMotion manually, and software to automate the whole process is at the very early stages. (See “Appendix B. IBM Virtual Machine Manager” on page 14.)

Another important drawback of this approach is that, although 2-way server costs are much lower than those of high-end SMP servers, we need to look at the scenario as a whole: Every 2-way server must have a (typically redundant) network connection and a (typically redundant) SAN connection. So, to the raw costs of the servers being used, you have to add the costs of things like:

- Ethernet ports on the departmental network switch
- Fibre Channel ports on the SAN switch
- The Ethernet and Fibre Channel adapters (host bus adapters or HBAs) per each server
- Pure costs and management of the cabling
- Maintenance for the above infrastructure

We are not implying that a single 16-way or 8-way server can always be configured to use only a pair of redundant HBAs. We have customers running 16-way xSeries 445 with only two HBAs, but because the dozens of virtual machines running on those are typically CPU-bound and memory-bound and do not require specific disk bandwidth, we understand that for scalability reasons more HBAs might be required for those high-end configurations. However, when using low-end servers, you would have to configure eight 2-way servers—so, for a total eight HBAs (16 if redundancy is a requirement), this also means eight (or 16) SAN switch ports that (likely) would be underutilized. This waste of money and resources is exactly what we are trying to avoid with virtualization technologies, which are all about simplifying the whole infrastructure through better resource utilization.

Although the IBM BladeCenter offering is very appealing from a management perspective because all of these infrastructure modules get collapsed into the chassis, this does not mean that they do not exist. No matter how these switches get deployed (racked traditionally or into the BladeCenter chassis), they still have to be purchased.

Conclusions

Unfortunately there is no generic solution for this complex matter. We have had instances of serious issues in finding a common agreement regarding the size of the servers that can be treated as modules or components of a scale-up or a scale-out approach, so how can we find agreement about when to use one approach and when to use the other?

For example, we have had customers implementing Oracle RAC in a scale-out approach using four 2-way Intel-based systems instead of a single high-end (scale-up) 8-way Intel-based system, and we have met other customers reporting implementing a scale-out approach using four 8-way Intel-based systems instead of a single high-end (scale-up) proprietary UNIX® platform. We need to look at these cases in perspective and, as you can see, the scope of your project and the size of the infrastructure is one of the key elements to acknowledge.

As we have said earlier, this depends on many factors and every situation is different. We can only make here general comments and conclusions that should not be intended as rules but mostly as high-level thoughts regarding a complex matter like this. We believe that we are in a position to make an agnostic analysis because we are basically the only vendor with a complete portfolio of x86-based systems available, from 2-way servers up to 16-way servers (and more in the near future). Vendors providing only 2-way and 4-way servers are probably stating that 8-way and 16-way solutions would not fit your needs.

We want to make two major comments here. One is related to the size and the scope of the virtualization project. The other comment relates to the readiness of the add-on tools to build the VMware virtual infrastructure (namely VirtualCenter and VMotion) on top of the ESX Server core capabilities.

Regarding the size and scope of the project, we tend to hear absolutes when talking to customers regarding this topic, such as “2-way servers are always the best solution” or “4-way servers are always better.” It would be better to think instead in terms of the percentage of computational power that each single server/node brings to a virtual infrastructure design.

To clarify this concept: Say your company needs to consolidate 30 servers onto a VMware virtual infrastructure. For the sake of the discussion, they would all run on eight physical CPUs based on preliminary capacity planning. What would your options be? You could install a single 8-way server but that probably would not be the proper solution, mainly for redundancy reasons.

You could install two 4-way servers but this would cause a 50% reduction in computing power if a hardware or software failure occurs on either of the two servers. However, we know many customers who find themselves in that situation because they thought that the value of such aggressive consolidation is worth the risks associated with the failure of an entire system. Many other customers would think that a good trade-off is to install four 2-way servers, as a node failure really provides a 25% deficiency of computing power in the infrastructure.

Here is a second example: Say that your company needs to consolidate 300 servers. For the sake of the discussion, they would all run on 80 physical CPUs based on a similar preliminary capacity planning. You can deduce from this second example that the absolute numbers that have been used in the first example might have a very different meaning here. In this exact situation, for example, the use of 10 8-way ESX Server systems would cause a 10% reduction of computing power in case of a single node failure, which is usually acceptable given the RAS (reliability, availability, serviceability) features of such servers and the chance of failure. (Remember that in the first example the failure of a single 2-way server brings a much larger deficiency of 25%). However, the use of 2-way server blocks means that the administrator has to set up and administer as many as 40 ESX Server systems, which could be a problem regarding regular maintenance and associated infrastructure costs.

We should also take into account that VMware has already announced that they will introduce 4-way SMP-capable virtual machines in 2005. By definition, a physical server cannot be “smaller” than the virtual machine that can run on top of it, so this might lead administrators to think, at least, that some of their nodes cannot even be 2-way servers (assuming that these administrators are going to run enterprise 4-way and above SMP virtual machines).

Two comments:

- Do not think about absolute numbers regarding the configuration of the servers (that is, virtual infrastructure building blocks); put them into *your* perspective.
- Regarding the readiness of the tools that comprise the VMware virtual infrastructure value proposition: Even if the grand view of VMware regarding the use of modular low-cost hardware tied together with management software and technologies (such as VirtualCenter and VMotion) is appealing, we must also realize that these components are not as mature, solid and feature-rich as the ESX Server core virtualization capabilities (for now, at least).

For example, we still lack enhanced and mature tools that enable the migration of virtual machines across the infrastructure based on complex policies to hide the complexity and modularity of the infrastructure itself and transform it into the single pool of resources that VMware referred to at VMworld 2004 in its vision of the virtual infrastructure. We are also missing the single, mature, and complete point of control for the high number of ESX Server images that a scale-out approach requires. Think of the amount of work that, as of today, either a minor upgrade of ESX Server version or a parameter tuning would require if your virtual infrastructure were based on hundreds of low-end servers that require these day-by-day routines. As of this writing, this is a manual process that may or may not be made semi-automatic at the cost of developing custom scripts (which might or might not always be the best option for a customer).

Of course, VMware (and its key partners including IBM) is making big progress in this management space, but consider this paper a point-in-time analysis for current deployments, so the next comment is:

- When planning (big) deployments and you are convinced that a low-end server approach is the best, double-check all advantages and disadvantages before committing to a given decision.

The intent of this Redpaper is not to force you to think that high-end is always the best choice, but based on field experience, we believe that the BladeCenter ESX Server offering does have big potential for now for small and mid-size virtual infrastructure projects. This accounts for dozens of (or a few hundred) virtual machines, especially in those cases where many VMs can be hosted on a single blade due to their workload patterns. In enterprise virtual infrastructure projects, which account for hundreds or thousands of virtual machines, we believe that the ESX Server ecosystem that builds the virtual infrastructure on low-end servers has to mature before these large projects could be deployed on such hardware (and we have no doubt that it will mature over time). At this moment, we believe that high-end SMP servers still have a very important role in those enterprise projects, especially from a return-on-investment (ROI) perspective, mainly because of the better manageability of the entire solution.

Appendix A. Server virtualization alternatives

This redpaper has focused extensively on VMware ESX Server technologies. It is important to note that there are other virtualization solutions in the marketplace such as Microsoft® Virtual Server 2005.

There is a lot of debate around Microsoft and VMware virtualization products but we do not want to provide a side-by-side comparison of the two. The purpose of this document is to discuss best practices and initial thoughts regarding the proper implementation of a virtual, flexible, and transparent virtualized infrastructure. As we write this, it is something that Virtual Server cannot provide due to some missing key features such as the ability to share storage among different Virtual Server systems and the ability to move running virtual machines from one host onto another on the fly (a VMotion-like feature).

Some of the discussions we went through in this Redpaper can be applied to a Microsoft Virtual Server scenario, but take into account that missing features such as those described above would cause a scale-out approach to be even less transparent than a comparable VMware ESX Server solution with VMotion.

Having said this, we are monitoring closely all of the virtualization technologies that are available in the marketplace and we are driving their adoption based on technical requirements, ROI, and customer demand, trust, and confidence in the available technologies.

Appendix B. IBM Virtual Machine Manager

IBM Director is a key component of the set of systems management features that are available on the xSeries and BladeCenter platforms. This tool is aimed at the management of the system environment that spans the hardware and critical functions of the operating systems. It provides a single consistent interface for managing the entire xSeries and BladeCenter implementation, keeping an up-to-date inventory of the current infrastructure and, more important, an engine that can respond and react when a problem occurs on the infrastructure.

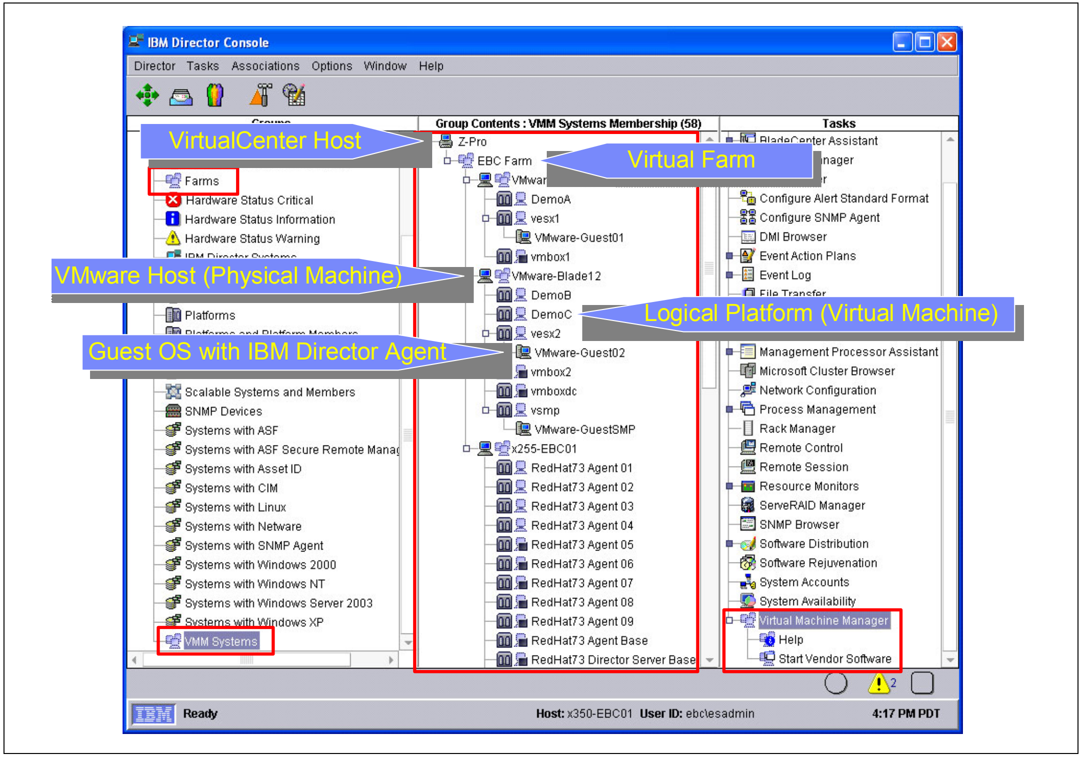
IBM has developed a plug-in to IBM Director called Virtual Machine Manager (VMM) that extends IBM Director to include “virtualization awareness.” Specifically, VMM supports both VMware deployments and Microsoft Virtual Server deployments, and it provides a detailed view of physical platforms correlated to the virtual environment that each of those supports.

One of the interesting things VMM can do in the current release is detect via the base IBM Director software an hardware alert (such as a PFA alert or another non-critical hardware issue) and migrate one or more virtual machines off that server via VMotion onto another server that is operating normally and has sufficient resources to support the additional workload.

For more information about this plug-in (and for its download) refer to:

http://www.ibm.com/pc/support/site.wss/MIGR-56914.html

Figure 9 shows the IBM Director console with VMM.



VirtualCenter Host

Virtual Farm

VMware Host (Physical Machine)

Logical Platform (Virtual Machine)

Guest OS with IBM Director Agent

Figure 9 Virtual Machine Manager

Appendix C. SAN considerations

We do not discuss storage-related concerns in this Redpaper, but it is safe to say that the same concerns we have covered regarding server systems can be compared to similar discussions related to storage (and VMFS volumes) configurations and best practices.

Many discussions are going on regarding the advantages and disadvantages of using, at one extreme, a single huge VMFS volume to accommodate all of your virtual machine files, as opposed to the other extreme, which calls for a dedicated LUN/VMFS for each virtual machine disk.

You might deduce from this that many of the discussions we have already had regarding the server models (few big nodes versus many small nodes) can be also used to discuss storage-related layouts and best practices.

Having a single huge VMFS might be optimal from a flexibility and utilization perspective, but it could be a nightmare from a high-availability perspective and, in some circumstances, from a performance perspective.

As always, *in medio stat virtus* (virtue stands in the middle). A good trade-off could be that administrators create a certain number of LUNs where each LUN supports a certain number of virtual machine disks (this number could be five, 10, 20, or even more, depending on the characteristics and the size of each infrastructure). To maximize flexibility and high availability, and to minimize the risk of VMFS corruption, a reasonable best practice could be that the virtual infrastructure is deployed in groups (clusters) of servers that each map to a group or cluster of VMFS logical units on the SAN.

Figure 10 summarizes the logical view of this concept.

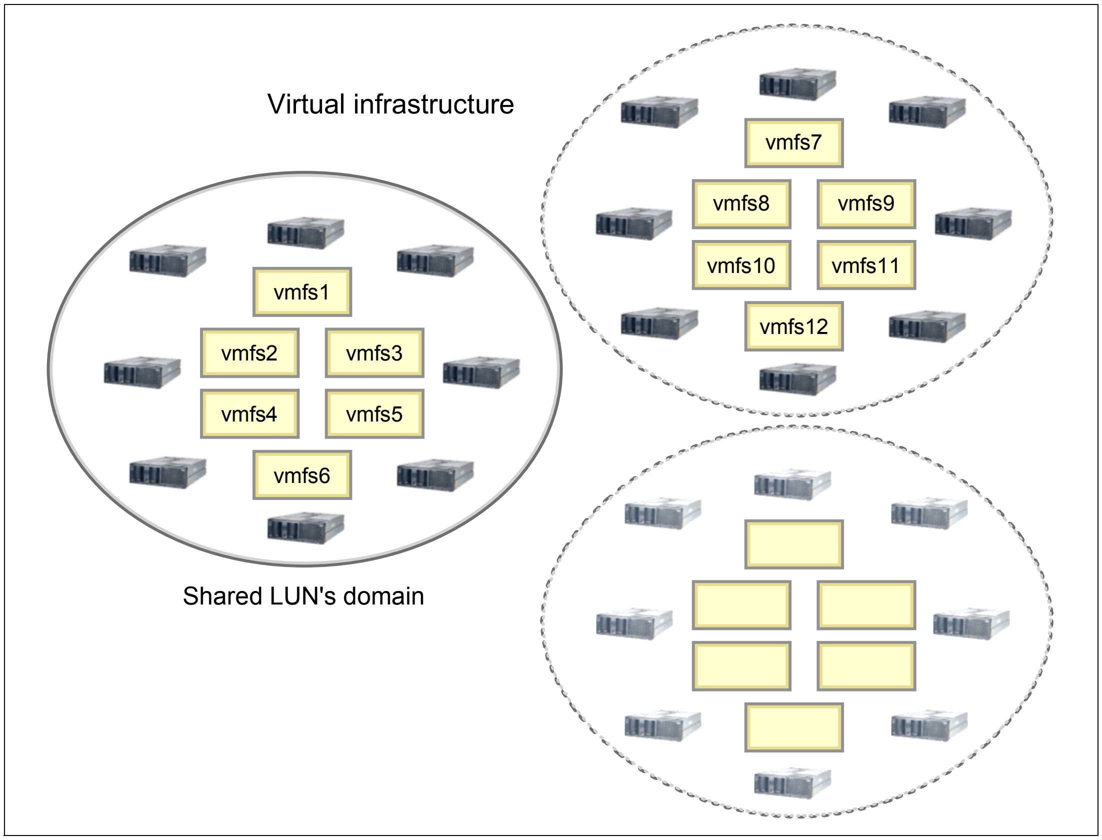


Figure 10 SAN logical view in a VMware Virtual Infrastructure context

The point here is that if any single piece of the infrastructure fails, other components can take over. So, for example, if an ESX Server system in a shared LUNs domain fails, the VMs can be brought online quickly on other nodes as they have the same SAN visibility (they all see the same dsk files in the same domain). Also, having the minidisk files belonging to a single domain spread across a number of VMFS volumes (six in this example) should alleviate the downtime and time to recover via a restore operation if one of the VMFS / LUN volumes become corrupted or unavailable for some reason.

Ideally in a virtual infrastructure (comprised of many server nodes), you could move every virtual machine on every physical server. In reality, you cannot connect (or at least it is not a best practice to connect) more than a certain number of servers to shared VMFS volumes. Hence the introduction of the concept of shared LUN domains, which is all about segmenting or zoning clusters of servers to have access to VMFS volumes in that zone.

Although it might not seem so in Figure 10, the idea behind the example is that each of those vmfs partitions hosts more than one minidisk (varying quantities depending on the situation). The reason for using more VMFS volumes in each VMotion domain ensures that the corruption of a single VMFS does not stop the whole domain. In this example, you would lose 1/6 of all of your virtual machines, and their restore from tape or other restore process will take 1/6 of the time that would take to back up a monolithic VMFS for the domain.

The team that wrote this Redpaper

This Redpaper was produced by a team of specialists from around the world working at the International Technical Support Organization, Raleigh Center.

Massimo Re Ferre' is a Certified IT Architect within the IBM Systems Group in EMEA. For more than 10 years, he has worked for IBM in Intel solutions, architectures, and related products. In the past few years, he has worked with server consolidation and server rationalization projects with key customers in the EMEA South region. Massimo is a member of the Technical Expert Council (TEC), which is the Italian affiliate of the IBM Academy of Technology, and he is a VMware Certified Professional. His e-mail address is king@it.ibm.com.

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

John Hawkins, VMware, Inc., USA
Lance Berc, VMware, Inc., USA
Andreas Groth, IBM IT Architect, UK
Roberta Marchini, IBM IT Specialist, Italy
Fabiano Matassa, IBM IT Architect, UK
Linda Robinson, International Technical Support Organization, Raleigh Center
Betsy Thaggard, International Technical Support Organization, Austin Center
David Watts, International Technical Support Organization, Raleigh Center

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 illustrates 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. 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 IBM's application programming interfaces.

This document created or updated on December 16, 2004.

Send us your comments in one of the following ways:

- Use the online **Contact us** review redbook form found at:
  ibm.com/redbooks
- Send your comments in an email to:
  redbook@us.ibm.com
- Mail your comments to:
  IBM Corporation, International Technical Support Organization
  Dept. HZ8 Building 662
  P.O. Box 12195
  Research Triangle Park, NC 27709-2195 U.S.A.

Trademarks