VMware® Infrastructure 3

Advanced Technical Design Guide

~and~

Advanced Operations Guide

Two books in one!

Ron Oglesby
Scott Herold
Mike Laverick

Chapter 7 - VMs and VM Selection

So far, most of this book has focused on the architecture and mechanics behind VMware ESX Server/VI3. We've described (sometimes in excruciating detail) the smallest components that make up ESX and the virtual environment. In contrast, this chapter focuses on the guest environment and VM strategies. This is somewhat important seeing how it's the whole reason anyone uses VMware in the first place.

In this chapter we'll look at how to create a Virtual Machines strategy for your organization and how to determine which servers in your environment will become guests in the ESX Server farms you create. Then, we'll get into VM standards, VM configurations and P2V migration decisions that need to be made.

Creating a Virtualization Strategy

Until a few years ago, the only people in an IT organization who had used (or even heard of) VMware were generally the developers and engineers. These original users of the VMware products used VMware Workstation and in some cases GSX Server to implement virtual machines for testing and lab development.

VMware allowed these IT pros to create quick test beds for new scripts, applications, or even to try out a new OS within their organization. Eventually, some IT departments started using VMware to host large portions of their test environments. However, even in these cases, most CIOs or CTOs still had not heard of the product. Those that had only had a passing knowledge of it.

As the practice of using virtual machines for testing grew, people started to notice (and so did VMware) that most of their other production hardware was underutilized. Most servers were over engineered from the start to save on upgrades later. In addition, more and more servers were required for everyday business, and it seemed like each new application on the network required its own dedicated server—even if all it did was run a batch job a couple of times a day.

In addition to this general production server underutilization, IT organizations also started to notice other benefits. The first thing benefit noticed was that the time they were spending provisioning, putting together, racking, configuring, and managing servers was becoming a larger part of their day. It used to be that organizations ran the same servers for long periods of time with very little change. Most management was on the application side or simple server maintenance and monitoring. Now it seems that almost every day brings a new application requiring any number of new servers—even while the old servers on the network never seem to be completely decommissioned.

As companies and datacenters grow larger, they begin to notice that the vast majority of their production servers are underutilized. Try this little exercise if you don't believe this: Walk into your datacenter and pick a rack of Windows servers. Out of the rack pick three of the servers at random. Open up the console on these servers and run performance monitor for 20 minutes on each. Look at Overall Processor Utilization and Physical memory in use. If you want, throw in a network and a disk counter. I'll bet that not one of the three is at or near 90% utilization on any one of those counters. In most cases, when I have done this (not including Citrix environments), the servers are all less than 20% utilized on all of those "Core Four" resources except memory, and in that case it is usually less than 50%.

Now we as engineers and architects used to think this was a good thing. This is because we were taught that if the servers are not near capacity, then the performance is as good as it can be on the server and there will be few if any complaints from the user about performance for those applications. The problem here lies in the fact that you may have a datacenter with several thousand processors in it all at an average of 10% utilization. Ten percent! Think of that. That's like buying a car you can only drive less than a day a week, then having to buy three or maybe four more of those cars to fill up the rest of the days of the week.

One thing to remember about processing is this: Any server that is not running with 100% CPU utilization is "wasting" time executing idle threads. Whether you have 30% utilization or 80% utilization, there are still available CPU cycles.

Every idle tick that is processed is time when the processor is doing nothing and is waiting for you to send it more instructions. Why not use this capacity? A server at 80% is just as fast as a server at 30%—it's just waiting to be used.

Once you understand that, it's easy to understand that virtualization and specifically VMware can help with this. The real question is how can it help your specific organization? The answer depends on where your organization's pain points are. There are basically a few different strategies for VMware. Some of these can combine and morph into smaller tactical solutions, but the basic concepts can be found in one of these three situations:

- Using VMware for test and development environments only
- Using VMware for underutilized production servers in addition to Test and Dev
- Using VMware for ALL servers in your environment

Let's examine each of these strategies, what they entail, and the benefits and drawbacks of each.

Using VMware for Test and Development Servers

This is kind of like saying, "let's use our old hardware for test servers." Almost no one can argue with this strategy. Using VMware for a test environment is really a no-brainer. Within any size organization a single VMware ESX server can be a huge benefit to the test environment.

VMware's ability to host a large number of underutilized servers on a single piece of hardware can reduce hardware costs for your test environment by 50-75% or more. In addition, the ability to make a Virtual Machine disk un-doable is a huge benefit. Imagine the following scenario:

A developer in your company wishes to test a new application install he has created. It works on his workstation but he hasn't tested it on any server builds yet. The catch is that each time he runs his test he has to see if the install was successful or not, and if not, he needs to start with a fresh build. Using a VMware test environment, the developer can test his install and if it has failed, it can be back to a fresh image in under a minute.

Let's take this development environment a little further. If this same developer needs to test his install against various types of server operating systems and configurations, he would ordinarily require one server of each type or else he

would be forced to rebuild the server using the OS or configurations for the next test. In a VMware environment, this developer can have numerous servers all running on a single piece of hardware provisioned for just these tests. He can finish his testing sooner, utilize less of the IT staff's time, and in turn can de-commission or delete the servers he used for testing if he no longer needs them. (Try doing that with a $6000 asset assigned to a specific cost center!)

Let's look at a real world example of this from one of our clients. Before VMware, this client struggled to keep up with all of their development and test servers for their core business application. They had close to 100 test and development servers alone. After we took a look at these servers, we realized that 95% of them sat completely idle doing nothing at all on a given day.

After some doing some discovery on these servers we determined that they could repurpose about 80 of these servers if we brought in two servers (with 8 cores each) for VMware. This client was also undergoing a major expansion in their Citrix farm, and the 80 "extra" servers were a huge benefit to have from a cost savings perspective.

You're probably wondering what happened to the other 20 servers. Those 20 were used for system stress testing for new applications. Since those 20 were identical to the existing production Citrix servers, we decided that we would leave these physical servers alone in order to be able to accurately test system changes (as far as performance goes) on duplicates of the production hardware that was still utilizing physical servers.

When deciding to use VMware in your test and development environments you first must answer a few basic questions and do some discovery in your environment:

What will your production systems look like?

A test environment for a company whose production environment is hosted completely on VMware will look completely different than a test environment for a production environment consisting entirely of physical servers.

For now we won't get into the advantages and disadvantages of hosting production servers on ESX. (That comes later in this chapter.) Instead just take a look at your strategy for your production environment.

If you're introducing VMware into a test environment and your production environment will be built out of completely physical servers, then you should start by determining how many of the test/development servers are used or are going to be used for performance and stress testing. Once this number is known, you can basically rule these servers out of the VMware environment. When stress testing or gauging the performance of an application or server, you want to exactly duplicate the production environment. A performance test against a VM for an application being hosted on a physical server is basically useless.

However, the remaining servers are your VMware candidates. These servers are generally used for testing the functionality of new application code, service packs, OS upgrades, etc. These are called the "functionality" test servers because they'll be used to test the functionality of the system after a change.

These servers have just become your targeted guest operating systems. Once you have identified them, you can create a quick cost/benefit analysis showing the costs of physical servers for this environment versus virtual servers. We generally find that in test environments, the cost of virtual servers is about one third to one quarter the cost of physical servers. If the majority of your servers are idle 90% of the time, this ratio may even be higher.

Advantages of using VMware for your Test/Dev Environment

- Cost effective use of hardware
- Large portions of your production environment can be replicated on a few servers
- Lower cost of hardware for the entire test environment
- Faster rollback during testing
- Faster deployment of a new test platform
- Test VMs can be decommissioned and even deleted after they are not needed

Disadvantages of using VMware for your Test/Dev Environment

- Requires that your staff have (or learn) some basic VMware skills
- VMs are not good for load /stress testing if your production environment is completely physical

As you can see, the advantages of using VMware in your development or test environment easily outweigh the disadvantages. On top of that, the second disadvantage can be completely mitigate by simply keeping an few pieces of hardware for load testing if you plan on maintaining physical production servers.

Using VMware for Underutilized Production Servers

VMware is also often used for production servers that have very little in the way of hardware requirements. In the last year or two the use of VMware ESX in production environments has grown enormously. The initial "fear" of the new technology is gone, and the reality that a VM is just as good as a physical server in most cases has become IT's reality.

Most organizations are beginning to realize that the strategy of application isolation (giving each application in the environment its own unique server) has left them with numerous servers that utilize very little of the physical hardware available. Often times you can look at a production server that hosts file shares for users or even the backend data for a specific department and will notice that the server is barely being used.

In this scenario, the IT department would utilize ESX Server to host multiple production servers on ESX hosts. This allows them to retain unique servers for their applications while reducing the overall hardware requirements across the board.

Not using virtualization technology in your environment for this use case could be career shortening. I mean apply this type of thinking in any other aspect of business. It is unthinkable. Let's look at the following scenario to see if it sounds familiar:

The accounting department in your company requests a new server for an application they've decided to implement. The application vendor states they re-

quire their own server (typical!) but that resource requirements are not very high for this server. Well even if testing shows that the application only needed 300Mhz of processing power, you can't even buy that processor anymore. Instead you have to buy a machine with a 3GHz processor and since you have certain company standards you throw in mirrored SCSI drives and a gigabyte or two of memory. Now you have a server that you paid $6000 for that you're using 10% of its processor and it's associated rack space, power, KVM, etc.

In most cases even production servers are underutilized. Since we as IT professionals have been trained to make sure that the system is responsive to the end user, we've been conditioned to think that less utilization equals better performance for the user. To us it is a "good thing" when a production server runs at 10, 20 or 30% utilization and is only using half of its physical memory. The problem here is that we (or the businesses we work for) have basically been paying for lots and lots of hardware that isn't being used. Imagine if every server in your environment ran at 50% utilization on all its resources. Now imagine that you could magically cut the number of servers in half by moving processing time, memory, etc. from one group of servers to another. You would be seen as the IT genius of the year. Why? Because it makes business sense.

All of this basically means that businesses have x% of available server resources not being used and still continue to purchase more of these same resources. It's like a company owning two buildings in downtown Chicago, both of which are only 50% full. Sure they plan to grow but they only plan to add 10 or 20% more people over all. Should they go ahead and buy a third building? Of course not. This (in business) makes no sense, but for some reason we as IT professionals have been taught this is how we do things in the Wintel world. (Part of the blame must go to the hardware vendors here, but that's another story.)

Combining numerous logical servers onto a physical piece of hardware is no longer a good idea, it's a "must do". Environments of any significant size can see a major cost savings in hardware by just putting some underutilized servers onto a single ESX server. In most large environments, the pay back for ESX is about 6 months depending on the rate of server deployment per month. For one client, the ESX payback was seen in 12 weeks, and that included factoring in consulting costs to architect the environment.

So how is it done? You can begin by looking at existing servers in your environment and even servers that are being requested. Estimates are made about

the server's utilization and that helps us determine the number of guests per processor. Later in this section we'll dive into the detailed process of evaluating potential guests but for now you should understand that you will have to estimate requirements for new servers and evaluate existing servers' utilization in your environment.

Often your "newer" servers can be re-used as ESX hosts for a number of your other servers. In this strategy existing dual and quad processor servers can be re-purposed as ESX hosts. Their original operating system is migrated to a VM and other VMs are then loaded onto the "new" ESX server. This allows you to implement a server consolidation strategy with a reduced outlay in new hardware costs.

Advantages of using VMware for Under Utilized Production Servers

- Cost effective use of hardware
- Existing hardware can often be reused as ESX hosts
- Lower cost of hardware for the entire environment
- Faster deployment of new VM servers
- Fewer production network ports required
- Fewer production SAN ports required
- Less time spent provisioning and tracking hardware in the environment
- Less hardware to manage and support in the environment
- Ability to move VMs from host to host as performance dictates
- Ability to do maintenance on physical hardware without interruption to VMs

Disadvantages of using VMware for Under Utilized Production Servers

- Requires that your staff have (or learn) VMware skills
- Some software vendors do not officially support VMs
- Can require in-depth knowledge of VMware to troubleshoot performance problems

- May require you to invest in some SAN infrastructure to provide space for VMs

Using VMware for all of your systems

While this may sound like a leap for some of us, it's really not that farfetched. VMware offers a lot of benefits aside from its major one of being able to share hardware. In fact I know personally of 3 Fortune companies that have a mandate of "Virtual First". Meaning, if you want a server, their standard is a VM. If you want to deviate from that standard you need approval from on high, and a solid business or technical reason not to use a Virtual Machine.

When a business decides to use VMware for "all" their production servers, they rarely actually get every server on the network hosted on VMware. But the theory for such a deployment (or attempting such a deployment) is as follows:

Since very few servers on the network are truly utilized 100%, this means that a dual processor web server with 65-70% utilization still has cycles left. While in a consolidation model (underutilized servers only) this web server may not fit your ideal profile for migration into ESX, it still has hardware resources available and could be hosted to gain other benefits. Let's assume that you wish to move all servers into the ESX environment. If this is a true business goal, then you could move the web server onto a quad or eight-way ESX host and still have room for other VMs. You can then select VMs for coexistence with this server that require very little in the way of resources, or allow VMw'ares DRS to automatically shift load as needed. This will allow you to get your maximum number of VMs per processor even though one of the VMs uses a high amount number of resources.

The general idea behind using VMware for all of your production servers is twofold:

- You can use your hardware to its maximum extent and realize all of the benefits from that
- Disaster recovery (DR) becomes extremely simple for the entire environment

Generally when people move all (or a vast majority) of their servers to VMware, it's because most of their servers are underutilized or they have a DR plan that relies heavily on the portability of virtual machines. In either case, they begin to truly use VMware to its fullest potential. Determining if this works for your environment will depend on whether most of your servers are underutilized or if you need that VM portability for DR or other purposes.

Advantages of using VMware for All or a Majority of Production Servers

- Cost effective use of hardware
- Existing hardware can often be reused as ESX hosts
- Lower cost of hardware for the entire environment
- Faster deployment of new servers
- Fewer production network ports required
- Fewer production SAN ports required
- Less time spent provisioning and tracking hardware in the environment
- Less hardware to manage and support in the environment
- Ability to move VMs from host to host as performance dictates
- Ability to do maintenance on physical hardware without interruption to VMs
- Facilitates DR for the entire Environment
- Decreases recovery time from hardware failures for all systems

Disadvantages of using VMware for All or a Majority of Production Servers

- Requires that your staff have (or learn) VMware skills
- Some software vendors do not officially support VMs
- Can require in-depth knowledge of VMware to troubleshoot performance problems
- Requires a more complex and expensive SAN infrastructure

Candidate Identification

Once you have identified your overall VMware strategy you can move on to identifying candidates in your environment. First, you should understand that there are two types of identification:

- Identification of candidates from existing servers
- Identification of candidates for new servers being provisioned

Identification of candidates from your existing environment is fairly simple and really just a simple exercise in resource utilization. Identification of candidates for new servers is a little trickier and generally is a process that requires you test each new deployment and move on from there. Let's go into each of the scenarios separately.

Existing Candidate Identification

To identify VMware candidates from your existing servers you need to have performance data from each server, an understanding of your strategy, and inventory data from there (including peripheral devices if any).

To collect this data there are a number of tools available. IBM has their CDAT tool. UNISYS has some tool they tout, but it is really UNIX tool that they ported (in a poor way) for Windows. VMware has their capacity planner application, and another major player in this area is Platspin's PowerRecon. Since I really don't like the CDAT or UNISYS tool, here we will focus on the raw counters you need and the two available tools.

Performance Counters that Matter

Let's start with some basics on candidate identification. Candidate identification of existing servers is really simple math, and the understanding of peripheral devices. To understand this let's look at each of the core-four resources individually and their needs from a metric and inventory perspective (this next section is Windows focused but can easily be done on Linux, for simplification we use Windows metrics only).

% of Processor Time and Speed and Number of Processors

When the Total % of processor time is combined with the number and speed of processors in the target system you come up with a PERFECT number for determining CPU utilization on a potential guest. A number of people try to use the % of CPU utilization only; this only leads to false identification, and more often than not, the exclusion of older candidates that are perfect fits for VMware.

When using % of processor only you get only a piece of the story. A server with a single 1Ghz processor running at 20% is different than a server with four 3GHz processors running at 20%. To make this counter valuable you need to take the % of the overall available processor throughput. So in our examples this would look like the following:

(Speed * Number of Procs) * (%utilization / 100)= MHz in use

(1000 * 1) * (20/100) = 200 MHz

(3000 * 4) * (20/100) = 2400 MHz

Those numbers look different don't they? Exactly. In candidate identification you need to rationalize the relative counters to come up with a valuable number. Now when you dump these into a spreadsheet you can come up with a MHz in use (Average and peak if you can get both) to determine the real CPU needs for a VM.

Typically we will take into account the speed of the processors and the number of cores when doing candidate identification. If you are looking for only underutilized servers you are probably going to want to see about a 4 or 5 to 1 ratio of VMs to Cores. So if a core gets you about the total throughput of 2800 MHz (not complete throughput of the processor's rated speed) you need to do the math and calculate in peaks and average run time.

To do this lets take the 2800 MHz per core. Assume we don't want to run at more than 80% normally (80% of 2800 MHz) 2240MHz. Now let's assume that you are more conservative and want to leave 20% of the CPU for peaks, failures etc (see cluster design sections to understand failover capacity better). In that

case you take 60% of the 2800MHz and end up with 1680MHz. Divide this number by 4 or 5 and you come up with 420 MHz or 336MHz. So your low hanging fruit (CPU wise) should average 420 MHz or less while still allowing for spikes of up to 40% more.

Now remember this is an average. So if 420 MHz is the average of your candidates for low hanging fruit you can have some that average 800 MHz and some 200 MHz. If you are going more aggressive and want to use all but the highest utilization physical servers you could take that average target up to 1 or 2 GHz. Our average rule of thumb is to calculate the point at which a server REQUIRES a physical server (generally averaging a full processor use or more) and identify them as potential candidates but not low hanging fruit.

Memory in Use

Memory in use is big for two reasons. The first is cost. Memory (in large amounts) is expensive. So if you have a host with 16GB of memory and put a VM on that require 8GB but little processor you have effectively used ½ of that server (by creating a bottleneck) and reduced your overall consolidation ratio. The second is that there is a limit to the amount of memory you can use in a VM. Using ESX you have a 16GB memory limit for any single VM. This means large boxes using that much memory don't often fit into "classic" consolidation models.

Much like processor the memory targets in VMs is heavily reliant on the consolidation ratio you are looking to achieve. If you have looked into the processor specs and determined you want 5 VMs per core, you then have to see if you can support that many VMs based on memory constrains. If you have 2GB per core installed (dual processor dual core server) for 4 cores and your VMs need 1GB each you have a memory bottleneck. Now if you have 4GBs per core and you need 1GB average per VM, you have a 4:1 ratio (close to your processor count of 5:1). So with memory you want to determine the amount of physical memory the servers use and find the average that works for you.

In our example we would be looking for VMs that physically USE a little less than a GB of memory. Then when the VMs are built you can assign them the next 256 MB increment up from what they already use.

To find this one of the best ways (available in almost all tools) is to grab the free memory counter (MB or Bytes) and subtract the free memory from the memory installed. So if I have a physical server with 4096 MB installed and free MB is 3275, my physical memory in use is 821. For the VM you would count this as a 1024 MB server when being migrated, but from a candidate identification standpoint it falls right into your strike zone.

A not on paging activity. Some people place a huge amount importance on paging activity. Personally I believe that if your candidate is paging (and using low amounts of memory) you can go ahead and give it more memory in its VM configuration. Having been part of projects with hundreds and thousands of physical to virtual migrations I have NEVER seen paging as a performance issue once the server was made a VM. Sure it may page, but give it some more memory. It doesn't kill the server and should not be relied on for candidate identification.

Disk IO and Throughput

A lot of time and space could be given to this discussion. But before we go into any of it I want to preface this section with the following statement:

While Disk IO and Throughput is important, it is rarely a constraint. I just finished a review of an environment with about 1800 servers in it. In this environment only about 20 servers had what we considered "high IO" and moved them into the "potential candidate" instead of prime candidate. 20 servers of the 1800 is only about 1% of the servers. And in every one of these 20 server they also have high memory and processor usage already moving them out of the low hanging fruit.

Now that I am off of my soap box let's talk about disk IO. In Windows you can grab a decent counter as Disk Transfers/Sec. This counter is the combined number of read and write operations on the disk. We use this number as the "IO" number and tend to rule out or move to "potential candidates" when servers get about 250 or 300 IOs per second. Now this is not a hard fast rule, but again we are looking for averages, if you all your VMs averaged more than 300 IOps on a single box, and you had 20 or 30 VMs you would have a lot of IO for a single server. But to have a few of these mixed in is not going to really hurt performance.

On the throughput side a good counter to look at is the Disk Bytes/sec counter. This counter is the combination of data from both reads and writes for the disk. This average is what you are looking for (later in this section we will discuss time frames but for now let's use overall average). For this counter we often find that servers fall into the 50,000 to 100,000 range. Sounds impressive right? Well, remember this is in BYTES. Converted to MB of throughput 100,000 Bytes is about 97K or less than a MB of throughput on the disk. From a candidate perspective we would try to keep this average below 20 or 30 MB. The reality is that if you are pushing 50MB or more of throughput you are more than likely also seeing high memory and processor usage.

Network Throughput

Much like disk you want to work out the averages for IO and throughput and assign accordingly. Also much like disk very few servers on your network use those gigabit connections you have connected to them. In most cases peak network and processor usage are seen during backups and tends to be 4 -10 times as high as normal load for the server.

From a counter perspective we like to look at the Bytes\Total Sec. The trick with this is that the counter is in Bytes and our Nics and Networks are rated in Bits. So you will need to do a little math here. Much like disk anything that I have seen ruled out based on the network (and I have only seen a couple) also had processor and memory constraints.

From a numbers perspective you need to be realistic. Most environments will have at least 2 NICs per server for a VM. If these are GB NICs and you have 20 VMs, then each VM should average less than 100 MB. Easy to do because in 90% of servers you will find they average less than 10MB.

Time Frames for Counters

The past few pages have went over the basic counters you need to look at for candidate identification. But it then comes down to being able to get data from the "important" times. VMware breaks this down with their tool by breaking usage data into an overall average, a peak hour average, and a business hours average (8am-5pm). In most cases simple perfmon cannot give you this info,

and a tool may be required or you may need to mine this out of your existing performance monitoring database.

In any case the important numbers are from the overall average, the peak hour average, and an identification of the peak hours. The overall average gives you numbers you need to scale and plan your environment, but the peak hour allows you to determine if peak hours are important or if the average is more important.

When we say "peak hour identification" we mean what hour (on the clock) do the servers peak. You will want to make a chart of this based on your candidates to determine if a majority of them all peak during the same hours (a 1 to 2 hour time slot) or if the peaks are spread almost across the 24 hour clock. If 50% of the server peak during a 1 or 2 hour period, then you need to plan on that from a utilization perspective. If your servers peak all around the clock and somewhat evenly, then the overall average can be used since servers peaking will be a small % of the VMs with the rest in off-peak hours.

A perfect example of this is a client I had that their backup jobs all kicked off simultaneously, as in not staggered at all. Basically at 7 pm every night almost every server in the environment went from 5% utilization to 50-100% utilization. The solution to this was to either build the clusters to take 50% utilization for each VM at the same time, or stagger the backups over night starting at 7, 8 , 9, 10 o'clock, etc. Now this is an extreme case, but something to be aware off. If most of your servers (let's say 60%) peak during the 8am to 11am window, you may have to scale your farm based on their peak numbers.

New Server Candidate Identification

New server identification is a little tougher. The best process for this is to implement a good introduction process into the farm that includes some testing and a virtual to physical migration process should you need it.

The most successful companies with VMware have implemented a "VM First" policy. This policy simply states any new servers for test or production are deployed as VMs unless a technical (or sometimes business) reason can be found not to. This requires that new servers be spec'd as VMs and tested. This is easy for organizations that manage application life cycle through a Dev, UAT and

Production cycle. During UAT you will be able to get a good read on the resource usage and performance for the end user.

If a server is going right to production, the policy would state that the VM will be deployed and performance will be monitored. If performance issues arise fallback options using a V2P tool will move the server to a physical chassis. This doesn't happen often, but the ability to move back to physical is what allows it to be successful.

In either case the VM should be deployed and its resource utilization reviewed. If enormous resource needs are identified, the VM can be upsized fairly easily or moved to physical.

Candidate Assessment Tools

The two major tools that we have talked about here are Platespin's Powerrecon and VMware's Capacity Planner. We will look at an overview of each one here and then compare and contrast their benefits and drawbacks. You could pull the data you need from internal monitoring tools (if they are configured to gather what you need) but assuming you don't have those tools we are only focusing on the two industry standard external tools.

VMware's Capacity Planning

A few years ago VMware purchased a company called AOG. This company was the original vendor for Capacity planner and gained some notoriety even before the purchase for being able to collect performance and inventory data without clients on the target servers.

Capacity Planner's architecture is pretty simple; one of their collectors (a Windows Server) is installed local to within the targeted environment. Their collector software uses WMI and remote registry calls to gather inventory information on up to about 500 servers. Personally I limit these collectors to about 300 or 350 servers total. Anyway, after the inventory for hardware and applications is done a new job is started that collects performance metrics from the targeted servers. This job goes out to each server once an hour (in Windows remote Perfmon calls are made in Linux shell access is made and data from the proc file

system is grabbed) and gathers four samples from the server for key performance metrics.

Over a 30 day period this data is collected. About once an hour the data (stored on the collector in CSV format) is uploaded to VMware's capacity planning website over SSL. It's this point in the process that gives some engineers pause… The data going out is inventory and performance data, which for most organizations this is no big deal, for some it is a showstopper.

The reason the data is moved to VMware is so they can process the inventory, marry it up to the performance data and even run some default reports. The idea being that VMware's partners can leverage pre-canned reports and tools without having to develop their own. The issue with this is that everyone using this tool tends to give the same results, and it's only available through VMware partners. That's right; you can't get the tool and use it, you have to find someone to sell you the licenses and do the assessment.

Advantages of using Capacity Planner

- Cost effective – it's the cheapest tool around
- Agentless, requires no change on the target systems
- Data and reports and generally reviewed by someone with experience in the VMware space

Disadvantages of using Capacity Planner

- Data is sent off-site
- Must be purchased from a partner, no self analysis
- Is only available in time chunks (30 or 90 days) no long term use
- Collectors are clunky and sometimes need to some baby-sitting to make sure data is being sent up to VMware
- Only works on Windows and Linux

Platespin's PowerRecon

PowerRecon is really starting to become a decent product. Platespin's first version of this product pretty much fell on its face. But with their latest release they have taken aim at VMware targeting some of their "bad points".

Much like Cap Planner PowerRecon will gather inventory and performance information remotely. But unlike Cap Planner, PowerRecon keeps the data in-house on a server in your environment. They then have a localized web-interface that installs with the product that allows you to run the reports and do identification without a outside consultants.

In addition they support other operating systems (like Sun) on non-x86 hardware. This could be advantageous for mixed environments that don't focus on Windows only. Also they allow you to buy this on a project bases (30-45-60 days etc) or on a permanent basis.

As you can see Platespin's product is very similar to VMware's, with the difference being that PlateSpin targeted Cap Planner's weaknesses. Solving their issues does not come cheap though, the price of PowerRecon is (at time of this writing) is about twice that of VMware's tool.

Advantages of using PowerRecon

- Agentless, requires no change on the target systems
- Data is not sent off-site
- Self use is available
- Permanent licenses can be purchased
- Works with some non-x86 systems

Disadvantages of using PowerRecon

- More expensive tool than VMware's

VM Standards

VM standards are possibly just as important as candidate identification. Much like physical hardware you need to define your "base" VM build. Now remember VMs are hardware. Your OS and other items that go into it are important, but here we are talking about the standard hardware that you will provision to VMs.

Typically one of two things happen in a new VMware environment. Either the administrator and other staff go 'hog wild' and provision VMs with dual and quad processors and 2,4 and 6 GB of memory, or they under provision VMs

and become the "VM Nazi" dictating that any VM only gets a single proc and 256 MB of RAM.

The reality is that the sweet spot is somewhere in between these two extremes. In cases where companies over provision (dual procs for everything) they didn't understand the impact of this on their environment (essentially doubling their scheduling load). Conversely organizations limiting VMs to only all single procs with almost no ram are limiting their field of play and possibly hurting performance of some of the VMs

The sweet spot comes when you develop a realistic VM standard that allows you to cover the needs of 80% of your VM audience with the ability to "upgrade" VMs to cover the other 20%.

Typical items that need to be address in your standard VM include:

- Number of Virtual Processors assigned
- Standard amount of memory
- Standard number of NICs and Virtual Switches connect to
- Standard Virtual Disk configuration (type, size and data disk size)
- Logical partitioning standards within the virtual disks

The standards you create are important and give you a baseline configuration for the majority of your VMs. This will help to limit support, create an environment that is easy to understand and troubleshoot, and is a best practice.

In addition to your "standard" VM configuration and the upgrade process for it, you could have a two tier system where there is a standard, and an up-sell VM that has a greater number of resources. These are often used in environments that charge back or charge an initial cost for their VMs as a way to recoup the cost a VM that uses more resources. Below is a table that shows a typical standard VM configuration and a large resources VM:

Tier	Low utilization/Standard VM	High utilization VM

Range	80% of env.		20% of env. (2x cost)	
Base	Windows	Linux/UNIX	Windows	Linux/UNIX
Proc	1 vCPU	1 vCPU	2 vCPU	2 vCPU
Disk	C: 10GB (VMDK) Data: Right-sized VMDK	12GB w/single VMDK	C: 10GB (VMDK) Data: Right-sized VMDK	12GB w/single VMDK
Network	1—Backup 1—Prodnet	1—Backup 1—Prodnet	1—Backup 1—Prodnet	1—Backup 1—Prodnet
Memory	1GB	512MB	2GB	1GB

As you can see the standard is just that, a base line VM standard. If a specific VM needs more memory or processor the standard allows for that. Of note you should see that the memory gets larger as you go to the High Utilization VM but does not go crazy. Memory should really be sized properly as it is the most common bottleneck in ESX environments, and can be easily over allocated based on application owners whining about needing X amount of memory.

Physical to Virtual (P2V) Migrations

P2V migrations are a mystical and magical thing to those new to VMware. To those that have been doing them for a while they are nothing more than an imaging or file copy process followed by some driver replacement. The reality is that a P2V process, generically, is the process of getting an existing physical server into a Virtual Machine.

When spoke about in the VMware community a P2V often means a tool based process that moves the contents of an existing server's disks into a virtual machine disk file, then inserting the scsi drivers into that virtual machine file to make it bootable when attached to a virtual machine. While this is a simplification, it is really the 'gist' of the whole evolution. How you actually do this is where the decisions come into play.

There are a number of players in the space that make P2V utilities. There is the free Ultimate P2V based on using a WinPE boot CD, Ghost and some cool utilities to do the driver fixes. There is VMware's own VMware Converter, and of course Platespin's PowerConvert.

The real info you need to know is about your options. There are three basic P2V options for migrations:

- Use a P2V tool and do a "live" migration
- Use a P2V tool and do a cold migration
- Don't use a tool, and build a new VM then migrate to it

Each option has its benefits and drawbacks, but you should understand all of them BEFORE you offer up any of them.

P2V tools and Live Migrations

The term 'live migration' makes most people think of zero downtime, no interruption Physical to Virtual Migrations. The reality is that we want this type of migration, but often it does not work out very well. Early versions of these tools (starting out in about 2005/2006) were really bad. You often had to stop services (even like the Windows Server Service) just to get the live migration to work. So essentially your server was turned on during the migration, but clients couldn't access it… so why even do it then?

In the last year or so these tools have gotten better. Platespin's PwoerConvert has live migration option as does VMware's Converter. While this sounds great, you should understand that these tools have their limitation. At time of writing neither of these are recommended for actively used, transactional based servers. The risk is that you may miss a transaction in the cutover. That leaves servers

that do not change that often (like web servers and application servers) as the best target for these migrations.

The process is pretty simple really, an "agent" type of software is installed on the source machine. This agent acts a lot like a backup agent helping to move files from the source machine to the target VMDK(s). Once completed the agent can be removed or left in place.

You should not that when one of these migrations takes place you are essentially copying the contents of the source server's drives to VMDK files for the new VM. This can cause issues if you have servers with hundreds of thousands of files on them as the software has to parse through each file and copy individually as apposed a simple disk/block level copy that a cold migration does.

Advantages of using Live Migrations

- When they work they reduce the amount of downtime needed to convert to virtual
- Easier for app owners to buy into conversion with small downtime window for migrations
- Migrations move the application and entire server and configurations in whole, often reducing app owner involvement to simple testing

Disadvantages of using Live Migrations

- Is not recommended for actively used transactional systems (the ones you most want to do live migrations on).
- Can take longer than a cold migration that is a simple imaging process
- Requires that an 'agent' be installed on the source machine to facilitate the file copies

P2V tools and Cold Migrations

Cold migrations are the original form of migrations. Essentially an image is take of the server you are attempting to migrate. That image is pushed into a VMDK file just as if it was a physical server you were pushing an image to. The Actual "conversion" comes after the imaging is completed by inserting the drivers in to

file system within the VMDK. This makes the VMDK bootable so that the targe VM can boot up and complete the process (generally installing mouse and video drivers along with VMware Tools).

Cold migrations are the original migrations that were used in VMware environments. They are pretty much fool proof, and work on almost any type of server. The trick with these is that you need to be able to boot the machine to a dos or Win PE environment (and have the drivers to do that) and get an image into a target VM's disks.

Cold migrations are nice in that they work almost every time, require no agent in the source system leaving it in a pristine state should you need to fail back to it, and generally work very predictably (since it's a block level copy of a disk you can make assumption about timing based on the amount of disk used and not work about copying hundreds of thousands of small files).

The disadvantage of cold migrations is that app owners have heard about the "live" migrations and want to use those if possible. Most customers want to have almost no downtime and think that the live migration is the key to it all…

Advantages of using Cold Migrations

- Server is shutdown during the migration so no transactions can be missed
- Can be faster than a live migration option (from overall migration time frame)
- No agent installed on the source machine
- Migrations move the application and entire server and configurations in whole, often reducing app owner involvement to simple testing

Disadvantages of using Cold Migrations

- Can be increased downtime when compared to the Live migration
- Requires that you shutdown the server for the duration of the migration

Building "new" VMs for migrations

This is the simplest of all migrations (from a VM Administrators standpoint). Essentially you build a new VM with a fresh OS installed, and allow the application owner to migrate or install his app into the VM and handle the cutover. But while it is easiest for the VMware team it is often the most impractical from a project standpoint.

When a server is migrated using a P2V tool of some type the server is moved as an image. This means that configuration changes, application tweaks, and all the little bits of supporting tools that the application teams have installed or messed with over the years are moved over with no changes. This is great from a project perspective as it requires the least number of man hours over all in most cases, and ensures a duplicate of the existing server in the new VM environment.

Conversely if you were to create a brand new VM, the application owner would have to install the application, any supporting tools, make the configuration changes needed, test the application, then try to figure out what they missed. Obviously some servers would be fairly simple, but in the current IT world where identifying an app owner and even installation instructions for every application is a huge problem, this often adds weeks or sometimes months to individual application migrations.

Advantages of Building New VMs

- Reduces workload on the VMware team
- Supplies a fresh OS build in the new environment (with no hangovers from the physical world)
- Good time for an OS upgrade if needed

Disadvantages of Building New VMs

- Creates an enormous amount of work during the migration
- Slows the migration process down

Post Migration Clean Up

No matter which P2V tool or process you use, there will be some cleanup you need to do after the migration. When you move a physical server to the virtual

world there are services and applications running (most times) that have direct ties to hardware that was once installed. So post migration you need to have a strategy to deal with these hangovers from the physical world.

Typically we will have a script that will disable items like HP SIM monitoring, or Dell OpenManage services. Also you will need to do some basic clean up of the "old" nics that are no longer showing up in the Network Properties (in Windows) but still has the old IP address assigned. Finally a major change needed in most P2V'd servers in the HAL change. Prior to migration most servers are dual or quad processor servers. Then, when moved to a VM, are setup as single or uni-processor servers. This can cause excessive CPU activity because of the mismatched HAL. Basically you need to "downgrade" the system to a uni-processor HAL. For this to work properly you would need to carry out a kernel update. In the world of Microsoft Windows this means changing from an ACPU Multi-processor to a ACPI Uni-processor HAL within the "Device Manager" tool. Again there are steps to do this on the internet and in the admin section of this book. But for now, know that it is important!

As stated, the internet (and second half of this book) has tons of lists and step by step processes covering P2Vs. This section focuses on the decisions that need to be made and knowledge needed and leaves the regurgitation of the lists to your Google searches.

VM Deployment/Templates

From a design perspective the basic decision you need to make is whether to use templates or not. Templates are essentially clones of individual VMs. Basically you can configure a single VM, install your supporting software/agents, configure the OS with specific settings for your organization, then clone that template over and over for new deployments. When you need to update the template you can simply fire it up, make your updates and resave it.

The nice thing about templates is that they essentially allow you to do disk copies of the VM, and in the case of Windows operating systems even can be integrated with Microsoft's deployment tools such as Sysprep. The major drawback we see is that this process may not match up well with your existing deployment mechanism and you could wind up just adding another tool for OS deployment to your environment. With that in mind you really have a few options:

- Use template based VM deployment
- Use existing tools (like PWE boot based tools) for VM deployment
- Hand build/other…

Using templates is not a bad thing, and in most VMware environments pretty much the norm. Using a template is a lot like using an imaging process for OS deployment, the difference here being that your hardware is never different since ESX abstracts the underlying hardware into the virtual machine's standard hardware.

The drawback is that you may already be using a tool like Altiris to do your OS builds. Or in the Linux space you may have centralized scripts that you run to ensure each new machine is built to current standards. In these cases template use would basically create two deployment mechanism instead of further simplifying your environment.

So, templates are great for VM environments, but you need to make a smart decision about them. Don't create more complexity when you don't need to. I have a client that does complete unattended installs of any OS they support. They slip stream in any patches or updates to the builds and find that deploying a VM this way is no different than their existing process. For them it makes perfect sense.

On the other side I have a client that hand builds based on a checklist whenever a server is not delivered from their vendor with their pre-loaded image. In that case templates were a perfect fit as they had no internal tool.