Realtime publishers
"Leading the Conversation"

The Definitive Guide™ To

Vista Migration

sponsored by
altiris®

Danielle Ruest
and Nelson Ruest

Chapter 3: Creating the Migration Test Bed 61
Identifying Team Needs 63
Working with Different Testing Levels 65
Required Lab Environments 68
Relying on Virtual Machine Software 69
Physical versus Logical Workspaces 71
Defining Lab Requirements 72
Minimal Configurations for Lab Systems 73
Virtual Machine Configurations 75
VM User Accounts 76
Required Server and Workstation Roles 77
Requirements for each Testing Level 79
Creating the Lab Environment 81

Copyright Statement

Chapter 3: Creating the Migration Test Bed

The testing laboratory is perhaps the most important part of any technical implementation project. Testing and retesting solutions before they are deployed is the only way to ensure high quality. After all, you do not want to find yourself in a situation where you are deploying garbage into your network. Garbage aside, you'll quickly find that the lab environment is one of the most exciting aspects of the project. Things move fast, you're playing with new technologies, you're the hub of all technological requests; it's just fun to work in the lab. This is another reason why you want to have everything in place the right way.

When it comes to the deployment of Windows Vista, make sure you follow the golden rule of high quality solutions and provide a testing environment that meets and exceeds every need. But a testing lab is not only a technical solution; it also requires processes and procedures that must be followed to a 'T' if you want it to succeed.

To build the appropriate lab, you need to understand what your technical teams will require. What type of project they are in and how will the project teams be organized? In the case of PC deployments, two main tracks must be covered: the creation of the PC structure and the modification of server infrastructures to support new PC operating system (OS) features. Dividing the technical aspects of the project into these two streams will help you understand what your technical team will require and when they will require it.

As discussed in Chapter 1, the Desktop Deployment Lifecycle (DDL) involves several preparatory steps prior to deployment. Each of these steps—infrastructure modifications, application preparation, personality protection and PC image preparation—involves activities that are either concentrated on the PC or at the server level. Teams work together to prepare each aspect of the complete whole. Therefore, when it comes to the laboratory, you'll need to make several important decisions:

1. Will the laboratory involve physical space? What kind of physical space will be required?
2. Will you rely on virtual machine technology and if so, to which extent?
3. What are the minimal configurations you need to use for testing?
4. Which server and workstation roles are required in the lab?
5. Which testing levels will you want your teams to proceed through?
6. How will you create each testing environment?
7. Which strategies will you use to control the graduated testing that is required to refine the solution?

The answer to these questions will help you formulate your laboratory environment as well as the procedures it will rely on.

Begin by identifying your technical teams (see Figure 3.1). Every desktop deployment project will involve at least three teams:

- The administrative team is responsible for all project logistics as well as for coordination of all efforts.
- The PC team is responsible for the creation of the PC solution.
- The Server team is responsible for the preparation of the server environment to support the PC solution.

You should ensure a very tight collaboration between the server and the PC teams and interchange personnel between the two teams whenever possible. For example, your project may only require one security professional whose job will involve the analysis of all security features, the preparation of local security policies, the preparation of access rights on file shares in the lab, the preparation of Group Policy Objects (GPO) for production and so on. By properly scheduling activities throughout the project, you should be able to create virtual teams that intermix skills and improve communications throughout all the technical aspects of the solution you are designing.

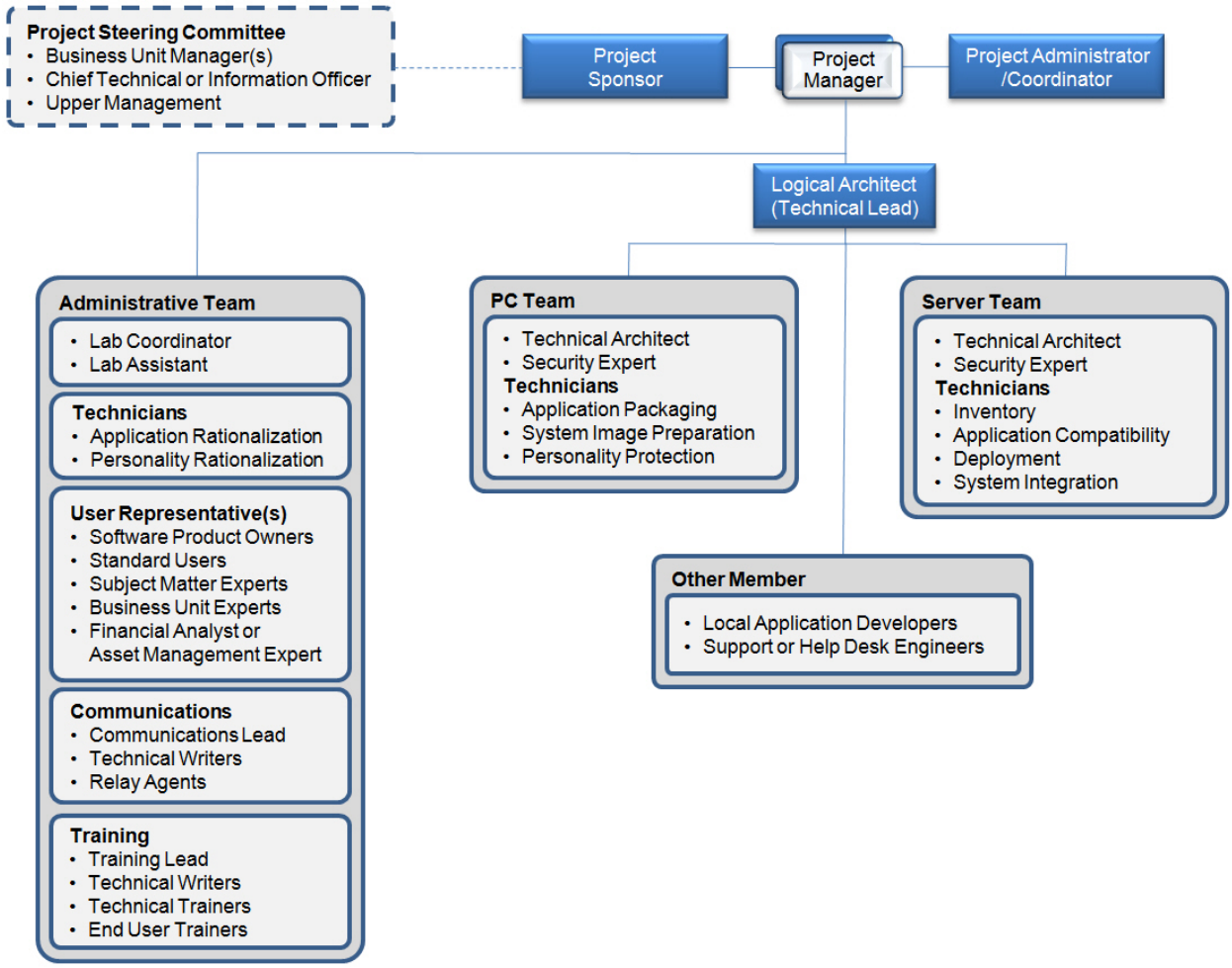

Figure 3.1. The Structure of the Technical Team

Identifying Team Needs

Every technical activity for this project needs to be tested in the lab somehow. For example, if you need to deploy a tool, script or otherwise, into production to perform your inventory or readiness assessment, then you should test it in the lab before deployment. If you need to determine which tools you will use in your deployment, then you'll need to test them. If you want to use a new method to deploy PC images, then you'll need to test it. Everything needs to be tested.

And, since you know your teams will focus on two areas, PCs and servers, you'll need to provide functional support for each technical aspect. In addition, your testers and technicians will begin small and grow as their solution progresses. For example, at first, PC image designers will only need a workstation and perhaps some file server space to test their part of the solution, but then as their portion of the solution grows, they will require servers with special services in support of image deployment, systems to manage image servicing, systems to test image deployment results and so on.

> 🕮 For a good overview of testing activities and testing schedules, review the Test Feature Team Guide in Microsoft's BDD 2007. The guide is included with the other BDD documentation which can be found here. As you'll see, the guide runs through several testing levels and helps you identify which tests need to be performed when.

The best way to identify the needs of your teams is to sit down with them and run through the processes they intend to use to develop their portion of the solution. For the PC preparation aspect, this should include:

- PC Image Preparation
  - Workstations representative of the baseline systems you've selected, both mobile and desktop systems
  - Image management and servicing systems
  - Reference computers
  - Image storage space on a server
  - Network connectivity components to link test systems to image sources
  - Deployment mechanisms
- Application Packaging
  - Source software for all the retained applications that are moving forward to Vista
  - An electronic repository of all source software
  - A software packaging tool, including server-based workload management tools and a software repository
  - Test systems for quality assurance
  - Packaging workstations
  - Database system for storing all software packaging information

- Security Preparation
  - Test systems for the preparation of all security aspects of the workstation
  - Server-based security technologies such as Group Policy and therefore an Active Directory
  - Tools for testing the security measures put in place
  - Security utilities such as antivirus and others to harden the systems
- Personality Protection
  - Test systems for the preparation of the solution
  - Personality protection tools
  - Server-based storage for protected personalities
  - Backup technologies for personality archiving

And of course, each team will need appropriate workspaces, access to the Internet and search engines, documentation tools and so on.

You basically need to reproduce your production environment for the teams to be able to perform their work. But you also need to have central control systems to manage all of the work and activity within the lab. For example, the lab coordinator will need to have a lab schedule that will be maintained on a daily basis to help coordinate access to limited resources without negatively impacting the project schedule. In addition, the lab needs its own technical mechanisms. For example, when someone uses a workstation, it needs to be reset back to its pristine state for the next person to use it. This is only one example of the technical requirements for the lab. It also needs protection mechanisms for all the deliverables prepared in the lab—software packages, scripts, PC images and so on. Therefore the lab needs its own infrastructure as well as the more volatile infrastructure used for testing and preparation.

Ideally, your lab will be set up in a permanent space and will continue to be used once this project is over. Laboratories of this type grow and shrink according to need, but continue to meet all of the testing requirements of the organization. They are formal structures that are part and parcel of the IT infrastructure and need to be administered as such. Whether it is for training, testing or development, it's really handy to have a readily available working environment you can just jump into when you need it.

Working with Different Testing Levels

Another requirement for the lab is testing levels. Testing is performed in a graduated process which gradually evolves into more and more complicated testing levels. For desktop deployments and other IT integration projects, there are five testing levels:

- Unit
- Functional
- Integration
- Staging
- Pilot Project

Each level integrates more complexity as technical teams progress through them.

The Unit testing level is designed for information discovery. Its purpose is to let individual technical team members discover how the feature they are tasked with designing actually works. For example, the PC Image Preparation technicians should use this testing level to discover how the Windows Vista installation actually works, play with initial Unattend.XML scripts in support of installation automation, discover how the SysPrep feature of Windows has been modified for Vista and generally familiarize themselves with the entire installation process. For this, they need access to Windows machines, including bare metal systems—systems with no OS installed—as well as Windows XP systems they can upgrade.

The Functional testing level is designed for initial solution testing. Here, the technical team has determined just how a feature works and now they want to test the automation mechanisms they are preparing. They also want to demonstrate it to their peers. In the example of the PC Image Preparation task, technicians will need bare metal machines as well as upgradeable PCs.

> ☞ Once Functional testing is complete, you may decide to perform a larger proof of concept and perform an early deployment to select members of the project team. This will give them the opportunity to comment on the quality of the solution and provide additional feedback into the final configuration of your systems. Proof of concepts of this type do not need to be deployed automatically, though if you can do it, this type of test provides an additional level of assurance that the deployment project will succeed.

The Integration testing level starts bringing each individual component of the technical solution together. For example, here you would blend PC Image Preparation with Personality Capture and Restore as well as other tasks. If you remember the PC Migration Cycle presented in Chapter 2, you'll find that the first time you run through the entire cycle is when your technical teams reach the Integration testing level. The objective is to make every aspect of the cycle work smoothly with the others.

The Staging testing level is focused on getting everything right. Basically, this level will provide an environment that is very similar to your production environment. While in Integration, you blended every aspect of the solution together, in Staging, you want to make sure you can reproduce every technological aspect from A to Z without a hitch. You'll have to repeat the process until it is absolutely right. This way, you'll know exactly what to do when you move to production and you won't make any mistakes. Technical implementations are 80 percent preparation and 20 percent implementation, but you can only get there if you've fully tested each aspect.

The final testing level is the Pilot Project. While all of the other testing levels focused on technical testing, the Pilot Project focuses on the complete solution, including any administrative aspect of the process. Here you test absolutely everything: the technical solution, communications, deployment coordination, training, support and so on. This test will validate the logistics of your solution as well as the technical aspects. Make sure you've completely evaluated each aspect before you move on.

These testing levels require graduation from one to the other (see Figure 3.2). Each level will have both exit and entry criteria. For example, to leave the Unit level, technicians must prove that they have covered all of the activities for this level. To enter the Functional level, technicians must meet key requirements, and so on. You'll build your exit and entry criteria as you learn to work more fully with the lab, but basically, they should aim to make sure that technical teams are fully prepared to move from one level to another. With the high demand you'll have for the resources of the lab, you don't want one team to hog resources when they weren't ready to access them. Also, you want to make sure no damage is done to any of the high level environments—for example, Integration and Staging—causing you to have to restore them from backup or worse, recreate them from scratch. The lab is for the use of all technical team members and should never be monopolized for one single aspect of the technical solution.

> 🕮 For a view into the Integration Architecture—the architecture that is used to build and manage testing levels—as well as a sample Exit Criteria sheet, have a look at this article which is part six of a seven part series on Enterprise Architectures.

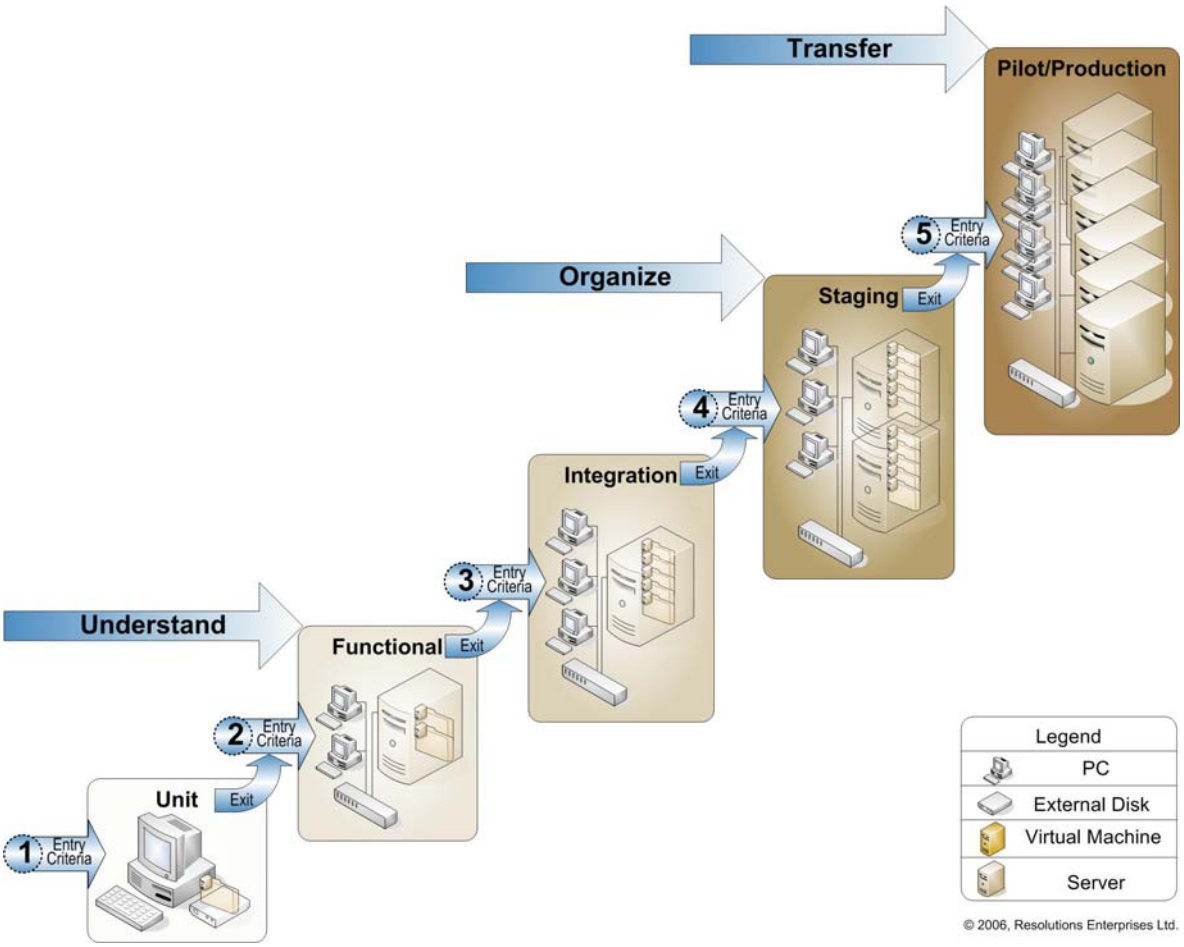

Figure 3.2. The Five Different Testing Levels

> ☞ Remember that building the lab is part of the Understand phase of the QUOTE System. Once the initial components of the lab are ready, technicians can undertake Unit and Functional testing. Then, when you move on to the Organize phase, testing will progress to Integration and Staging. The final testing level, Pilot, is part of the Transfer phase.

Required Lab Environments

Besides providing different testing levels, you'll find that your lab will also have to provide multiple environments. A minimum of three environments are required:

- The Certification Center
- The Conversion Center
- The Acceptance Center

The first, the Certification Center, is the core environment required for technical testing. It includes the five testing levels discussed earlier and is designed to support the preparation and certification of the technical solution. Certified solutions help protect your production environments by validating that each aspect of the solution passed all of your testing requirements and is ready for prime time. Unlike the other centers, the Certification Center is virtual and does not require a specific physical location.

> ☞ Certification is one of the most important processes in IT implementation preparation. It is closely tied to two other concepts: standardization and rationalization. Together, these three concepts help you maintain a dynamic infrastructure that can quickly meet any business requirement. For its part, certification focuses on validating technical solutions. When solutions have been certified before delivery, you'll find that you will have fewer support issues. In some of the projects we have performed, certified solutions were often delivered with zero defects and not one single support issue. That's why we believe that certification is worth its weight in gold for all IT environments.

The second, the Conversion Center, is required once the solution has begun to gel into a cohesive whole which is often after it has entered the Integration testing level. At this stage, you begin to understand just what your solution will look like and how it will affect both your in-house applications and the data in your organization. For example, if your only moving to Vista, you might find that you don't have a lot of data to convert since it does maintain the same data types as previous versions of Windows. But, if you decide to include Microsoft Office Professional 2007 with your Vista deployment, you'll have to provide a space for your users to build a conversion program. This means converting Word or Excel templates as well as document repositories. You might also need to develop an interim strategy, deploying conversion filters to people that don't have the new solution yet so they can continue to exchange documents with their peers.

In terms of applications, the Conversion Center can be used by in-house developers to upgrade the applications they are responsible for. Depending on the scale of the applications to be converted, you may find that you have to build up the scale of the Conversion Center. For example, if you are converting a line of business application that is supported by a team of developers, you might create a Conversion Center space just for them so that their work does not impact others. But, if you are looking to convert custom applications build by user developers, for example, Access applications, a single Conversion Center may be the right ticket. At the same time, you should look to restructure the way your organization uses Access and other user development tools.

> 🕮 If you need to convert Microsoft Access applications, you should take a look at how to gain control of decentralized application proliferation through Decentralized Deployment Strategies.

> ☞ If you do need to convert data and applications because you are moving to Microsoft Office Professional 2007, then you should definitely get a hold of the Microsoft Office Compatibility Pack as well as the Microsoft Office 2007 Resource Kit.

The Conversion Center can be opened as soon as you have a working solution for the creation of new PCs. The solution doesn't have to be deployable in an automated way. It all depends on whether you are using virtual machine technology or physical machines for testing and conversion. If you are using virtual machine technology, then you can just reset each virtual machine after each test, but if you use physical machines, then you'll have to at least have some system images of the machines so you can easily reset them after each test. More on virtual machine technology and its role in the laboratory is covered later in this chapter.

Finally, the Acceptance Center also needs to be maintained and supplied by the laboratory. Acceptance doesn't occur until components are ready for it so this part of the lab rarely comes until at least the Integration testing level has been reached. Exceptions are software packages. Each software package must be accepted by the appropriate software product owner—a subject matter expert who is also responsible for the evolution of a product in your network—when it has been prepared. Several packages need to be prepared before the complete solution needs to be integrated. As each package is finished, it should be passed through acceptance testing as soon as possible. This allows you to run parallel streams of activities and reduce the overall project timeline.

For full acceptance of the entire solution, tests must be done from the Staging testing level and must include all aspects of the solution. In this case, acceptance is performed by various IT stakeholders—stakeholders who will be responsible for the solution once it is in production.

> ☞ At the expense of repeating ourselves, we strongly recommend that if you are bringing in outside help for this project, you make sure that the bulk of the help you get is to replace your staff in their day to day tasks. This liberates them to work on this project. Its more 'fun' than normal day to day work and it gets them to learn new tricks and techniques. What's better is that in the end, you get to keep all the knowledge in house, letting you move forward at a faster pace.

Relying on Virtual Machine Software

Virtual machine (VM) software—software that emulates a physical machine and lets you run other instances of an operating system—is an ideal complement to the laboratory. There are lots of reasons why this is the case. First of all, in almost all instances, you can virtualize most of the servers in the solution. Ideally servers will be running Windows Server 2003 R2 (WS03) to ensure that they offer the most up to date server capabilities. In most of our tests, we've been able to run file server roles with as little as 256 MB of RAM allocated to the VM. Of course, you may need to increase the amount of RAM when you add roles to the server, but if you have the appropriate host, you should easily be able to run any server role you need.

Windows Vista client machines are a bit more problematic. Vista requires a minimum of 512 MB of RAM. You can cut it down to lower RAM levels, but performance will seriously decrease. Remember though that some physical PCs will still be required to test Vista image deployment. Several aspects of this test cannot be driven through virtual machines—driver tests, hardware abstraction layers (HAL) and the Aero user interface—at least not yet, so you'll need to have access to come physical PCs for testing.

Nevertheless, working with both client PCs and servers through VMs will seriously cut down the cost of building the laboratory. You can rely on virtual technology from either Microsoft or VMware as both offer free copies of their tools, once again reducing lab costs. Both offer full support for running Windows servers or PCs. In addition, you may want to obtain tools for the conversion of physical machines to virtual instances. This saves a lot of time as you simply point to the physical machine you want to capture, and easily transform it into a virtual instance.

☞ Get your free copies of the various products you need to support virtualization in your lab:
- Microsoft Virtual Server 2005 R2
- Microsoft Virtual PC 2004 SP1
- VMware Server
- Microsoft Virtual Server 2005 Migration Toolkit
- VMware Converter 3.0

In addition, working with virtual machines, you can 'virtualize' the laboratory. Many of our clients buy some very large host machines to run their laboratories. The ideal machine will be an x64 server running several multicore processors, lots of RAM and lots of disk space. Use Windows Server 2003 R2 x64 Enterprise Edition as the host OS for this server, install the virtualization technology you've chosen and you're off and running. When you consider the cost of these machines compared to the cost of having a complete physical lab, they really cut down the overall cost of the lab.

☞ For example, one customer was able to build an entire collaboration testing environment in less than 32 hours with only three people, two of which could not touch the keyboard since they were foreign nationals. Think of it: less than four days to build three physical hosts and more than 10 virtual machines playing roles as various as Active Directory, Exchange, SharePoint Portal Server, Content Management Server, SQL Server, Live Communications Server and more. This also included all of the standards for the install, all of the documentation for installation and configuration procedures, and of course, the virtual machines themselves including source machines for WS03 Standard Edition, Enterprise Edition and Windows XP Pro. In addition, they are now able to reuse and even duplicate this environment for other testing purposes. There is no doubt that this level of ROI is simply not available with physical laboratory environments.

There are a whole series of operations you can perform with virtual machines that you just can't perform with physical machines. For example, you can very easily create a source machine. Install a first instance of the OS into a VM, customize its configuration, update its default user profile, update it in terms of patches and once it is ready, copy it and SysPrep the copy. Voila! You now have a source machine that can be copied and reused to seed any machine role you need. That's a lot easier than working with a physical machine.

Another major benefit is the use of Volume Shadow Copies (VSC) on WS03. Since virtual machines are nothing more than a series of files on a hard disk, you can enable automatic backup protection for them by enabling VSC and then relying on the Previous Versions client to restore any damaged machine almost instantly. VSC automatically takes two snapshots per day and can store up to 64 different snapshots which provides a very adequate level of protection for your VMs. This doesn't replace proper backups, but it is at least a first line of defense that costs very little if anything.

Physical versus Logical Workspaces

Another advantage of virtual machine technology in the testing lab is that you don't need the physical space a conventional lab usually requires. If you create the lab space in a datacenter by hosting a series of virtual machines on a given set of servers, the host servers can easily be located in the normal datacenter and profit from the standard operations applied to any production server—backup, patch updates, antivirus updates and so on. Then, your technical team can connect to the VMs these servers host through the normal network. There may be reason for you to provide your teams with a separate network segment to isolate their network traffic, but if everything happens in the datacenter on the same server hosts, then network traffic is not really an issue.

You can create a single workspace for both technical and administrative project team members; that is, if they are in the same physical location. A workspace of this type will really help foster team building as well as ensure that there is constant communications between all team members. This type of work environment can seriously cut down project timelines and increase team bonding, one of the major factors in project success (see Figure 3.3).

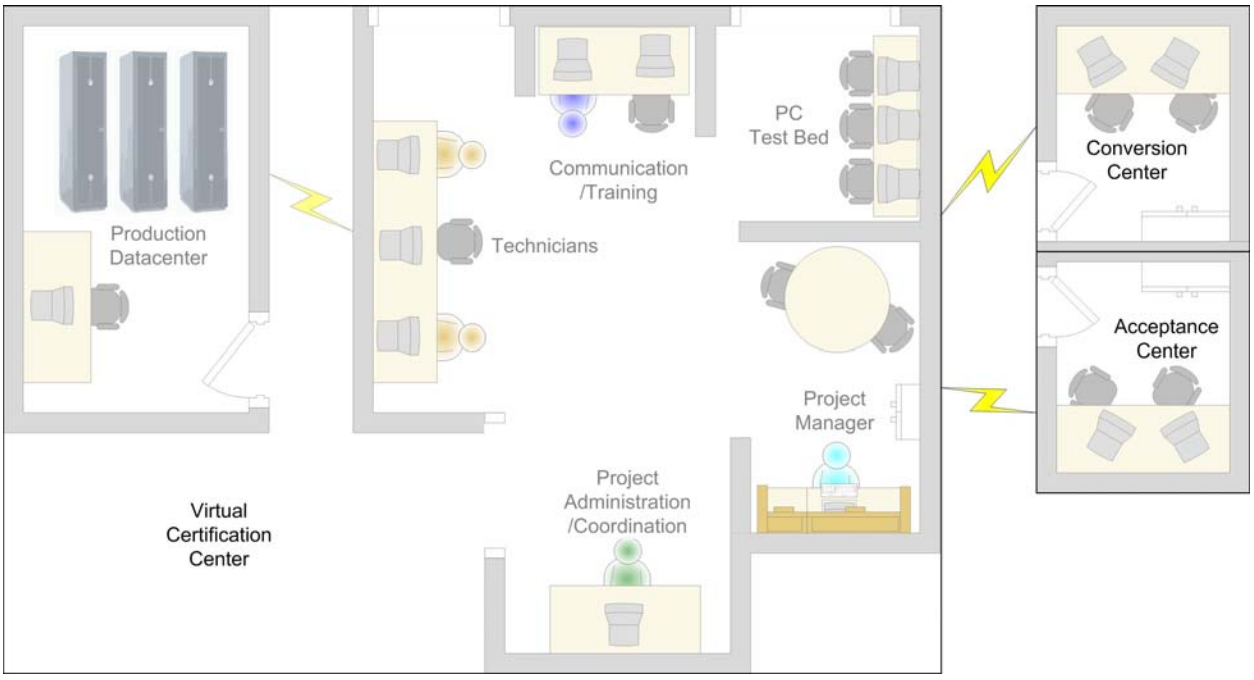

Figure 3.3. The Layout of the Project Workspaces

In the best projects we have run, a central workspace was created for team members. If other locations were involved in the project, this central workspace was duplicated in each location. Within the central workspace, each team member had their own PC for day to day work. Technical team members were given more powerful PCs with lots of RAM so that they could run local VMs if required. In addition, examples of each of the baseline PCs retained for the solution were available. All of these systems had a connection to the VMs in the datacenter either through the VM client or through Remote Desktop Connections. This forms a 'virtual' Certification Center.

As far as the other centers were concerned—the Conversion and Acceptance Centers—they were located in separate rooms, not too far from the central workspace. Separate rooms were used because of the nature of the work conducted in each center and because we needed to provide support to the users working at both conversion and acceptance. Each location was linked to the datacenter and to each other. Using a central workspace allowed team leaders to conduct ad hoc meetings whenever issues came up. These issues were often resolved before they became problems. This strategy provided the very best results and supported every aspect of the project. In addition, it fostered team building as all team members, even administrative members, grew excited as they saw the progress of the solution through instant demonstrations by technical staff.

Defining Lab Requirements

Now that you understand the various needs for the lab, you can begin to prepare for it. This may require some initial acquisitions in order to properly populate the lab. Too many organizations populate labs as afterthoughts, bringing in the oldest machines in the network and leaving lab technicians with having to cannibalize various machines to try to put something decent together. Don't make this mistake! This lab is the epicenter of the project so make sure you populate it appropriately. Here's an example of what you might need. The scale of the requirement will depend on the scale of your project. Keep in mind that you can run to five or six virtual machines per processor if all of the other requirements—RAM, Hard Disk Space—are available.

Minimal Configurations for Lab Systems

The minimal configurations required to support the lab should resemble the following list.

1. Host Server(s)
   - Dual x64 dual-core SMP Server
   - 512 MB RAM for the host OS
   - 256 to 512 MB RAM for each VM running on the host
   - At least 2 disks for RAID 1 (mirroring)
   - Preferably, 3 or more disks for RAID 5 (stripe with parity)
   - Use the largest disk size you can, currently about 300 GB
   - Retain about 30 GB for the system drive
   - Assign the bulk of the space to a data drive which will store the VMs
   - Retain about 30 GB for a third drive which will host the shadow copies produced by VSC
   - Dual network interface cards (NIC) at a minimum speed of 100 Mbps to support multiple connections to the virtual machines hosted by the server; verify with the virtualization manufacturer to see if you can team the NICs

> ☞ Use 64-bit processors because they break the memory barrier and can handle significantly more network throughput. 32-bit processors are limited to 4 GB of RAM and have to use special procedures to access any RAM above this limit. 64-bit processors can natively access up to 32 GB of RAM. There's just no comparison when it comes to running VMs.
>
> Also, make sure you acquire the proper versions of 64-bit processors because you will want to run x64 VMs. Currently, all AMD 64-bit processors support 64-bit VMs, but only VT-enabled processors from Intel will do so.
>
> Finally, you might consider selecting AMD processor-based servers since AMD currently guarantees that it will use the same socket footprint for quad-core processors when they are released later in 2007. To upgrade your server, simply pop out the existing processor and pop in a quad-core.
>
> 🕮 For more information on x64 servers, view the Move to the Power of x64 webcast from Resolutions.

2. Vista Premium Ready PC(s)
   - 1 GHz processor, 32- or 64-bit
   - 1 GB of RAM minimum
   - DirectX 9 support with at least 128 MB of dedicated graphics memory
   - DVD-ROM drive
   - Audio Output

> ☞ A Premium Ready PC configuration is listed here, but if you opt to keep some Vista Capable PCs in your network, you should also include examples in the lab. In fact, you need to have examples of each machine configuration you will retain after the migration to Vista.

3. Technician Workstation(s)
   - 1 GHz processor, 64-bit if possible
   - 2 GB of RAM minimum
   - DirectX 9 support with at least 128 MB of dedicated graphics memory
   - DVD-ROM drive
   - Audio Output

> ☞ Technicians should have as powerful a PC as possible. They will be running VMs both locally and remotely and may need to run more than one VM at a time.
>
> Only technician workstations are listed here, but you might consider giving each staff member, even administrative members, PCs that meet at least the baseline systems you have selected for your project. The entire project team will act as guinea pigs when you perform the proof of concept and get the entire project team to test the solution. This is another strategy that fosters excitement within the project team.

4. External Hard Drive(s)
   - External drive of 80 GB at 7200 RPM with USB 2.0 or Firewire (IEEE 1394)

> ☞ For Unit and Functional testing levels, the lab can deliver canned VMs on external hard disk drives. Technicians can then work from their own PCs with their own VMs without impacting any other person in the team. Running these VMs on high-speed external drives provides much better performance than running them on the system disk. In addition, because of the average size of a VM—from 4 to 20 GB per VM—delivering them on external hard disks makes a lot more sense than trying to place them on spanned DVDs. Using external hard disks, you will also be able to deliver VMs to remote offices where technical staff may perform activities in support of the project.

Virtual Machine Configurations

Virtual machines should be configured as follows:

1. **Standard Server VM**

   RAM: ..... 512 MB of RAM minimum
   OS: ..... WS03 Standard Edition
   Service Packs: ..... All applicable service packs and hotfixes should be installed
   Number of Disks: ..... Depends on the role; can be from 1 to 3
   Disk Size: ..... Drive C: 20 GB expandable disk (leaves room for upgrades)
   Drive D: 50 GB expandable disk (optional based on server role)
   Drive E: 10 GB expandable disk
   Network Cards: ..... At least one NIC per VM
   CD/DVD Drive: ..... When providing VMs for use either locally or remotely, you should include ISO files for the installation media; this lets technicians add roles to the machine and generally control its feature set

2. **Enterprise Server VM**

   RAM: ..... 512 MB of RAM minimum
   OS: ..... WS03 Enterprise Edition
   Service Packs: ..... All applicable service packs and hotfixes should be installed
   Disk Size: ..... Drive C: 20 GB expandable disk (leaves room for upgrades)
   Drive D: 50 GB expandable disk (optional based on server role)
   Drive E: 10 GB expandable disk
   Network Cards: ..... At least one NIC per VM
   CD/DVD Drive: ..... When providing VMs for use either locally or remotely, you should include ISO files for the installation media; this lets technicians add roles to the machine and generally control its feature set

3. **Bare Metal PC VMs**

   RAM: ..... 512 MB of RAM minimum
   OS: ..... No OS
   Service Packs: ..... No fixes or service packs
   Disk Size: ..... Drive C: 20 GB expandable disk (leaves room for Vista installation)
   Network Cards: ..... At least one NIC per VM
   CD/DVD Drive: ..... When providing VMs for use either locally or remotely, you should include ISO files for the installation media; this lets technicians add roles to the machine and generally control its feature set

4. **Vista PC VMs**

   RAM: ..... 512 MB of RAM minimum
   OS: ..... Windows Vista based on the editions you decide to deploy
   Service Packs: ..... All applicable service packs and hotfixes should be installed
   Disk Size: ..... Drive C: 20 GB expandable disk
   Network Cards: ..... At least one NIC per VM
   CD/DVD Drive: ..... When providing VMs for use either locally or remotely, you should include ISO files for the installation media; this lets technicians add roles to the machine and generally control its feature set

5. **Windows XP PC VMs**

RAM: ..... 512 MB of RAM minimum
OS: ..... Windows Vista based on the editions you decide to deploy
Service Packs: ..... All applicable service packs and hotfixes should be installed
Disk Size: ..... Drive C: 20 GB expandable disk (leaves room for the upgrade)
Network Cards: ..... At least one NIC per VM
CD/DVD Drive: ..... When providing VMs for use either locally or remotely, you should include ISO files for the installation media; this lets technicians add roles to the machine and generally control its feature set

> ☞ When creating client VMs, keep in mind that Windows Vista requires 15 GB of disk space to perform an installation or an upgrade. Stay on the safe side and make the disks even bigger. Since you're using expandable disks, the actual space won't be used until the system needs it.
>
> Also note that no Windows 2000 machine is required since you cannot upgrade from 2000 to Vista. You need to perform a clean installation.
>
> 🕮 Also, if you are working with VMware, check out this blog on the Realtime Publishers' Vista Community Web site for this on how to make it work right.

VM User Accounts

User accounts are also critical when setting up VMs for distribution to the technical team. With the Unit and Functional testing levels, it is safe to give administrative access to the technicians on your project team in both server and workstation VMs because they are standalone environments. But as you proceed through the testing levels, you need to tighten down change control and grant access to high privileged accounts only to the lab administrator. After all, capturing the changes required to the infrastructure is the purpose of these environments.

When you deliver standalone machines for either the Unit or Functional environment, you should endeavor to make servers domain controllers. Their behavior will be different than member servers, but it will be easier to assign different roles to the accounts the technicians will require. In many cases, these testing levels will require either a single PC VM or a PC VM coupled with a server VM where the server is playing a series of different roles. Technicians need to have appropriate access rights to discover how the Vista installation process works or to work with any utilities which will be required to support the migration.

When you grant users access to the VMs that make up either the Integration or Staging testing levels, you give them accounts with appropriate access rights as outlined by the information the technical team will provide you.

> ☞ Remember that in Windows Vista, the default administrator account is disabled at installation. Your technicians will require their own administrative accounts. You should make sure that whenever you provide access to machines in a shared testing level, that the administrative account includes a strong secret password.

Required Server and Workstation Roles

Within each testing level, you'll need to assign or create several different machine roles. As mentioned previously, in the Unit and Functional testing levels, server roles can be integrated into one single VM, but as you move up the testing ladder, you'll want to separate different roles to represent the production environment you're running. For example, your Integration level may still join several roles together into a few machines, but when you get to Staging, it should be as similar as possible to the production environment. Staging is that last technical test before you start making changes in production itself so you need to get it right (see Figure 3.4).

> ☞ You can even use VMs to simulate remote sites. Windows Server 2003 includes routing capabilities that are very similar to those of Cisco's devices. You can enable Routing and Remote Access (RRAS) on two different VMs and use them to simulate the routing of data from a local to a remote site. Then you can add branch office server roles to the site located behind the remote routing server.

In the end your lab will need to provision several different types of levels and several different environments. Remember that when you build the more permanent testing levels or environments, you'll want to build in redundancy into the lab infrastructure. For example, the Staging testing level should have at least two domain controllers for redundancy. File servers will need to have several different file shares to provide storage for PC images, software repositories, and user data repositories. Deployment systems will need to support several features such as inventory, PC image deployment, software deployment, status reporting and perhaps software metering if you need it. Each level will also require some form of database server because several deployment components require it. And if you simulate remote sites, you'll also need to include replication engines into your solution to replicate objects such as software packages, PC system images, and so on.

> ☞ Rely on WS03 R2 for all replication scenarios. WS03 R2 includes the Distributed File System Replication System (DFSRS), a new delta compression replication engine that is a snap to configure and replicates only changes between files. It is much better and easier to use than either the traditional File Replication Service (FRS) or scripts based on the Robocopy utility found in the WS03 resource kit. Try it, you'll love it!

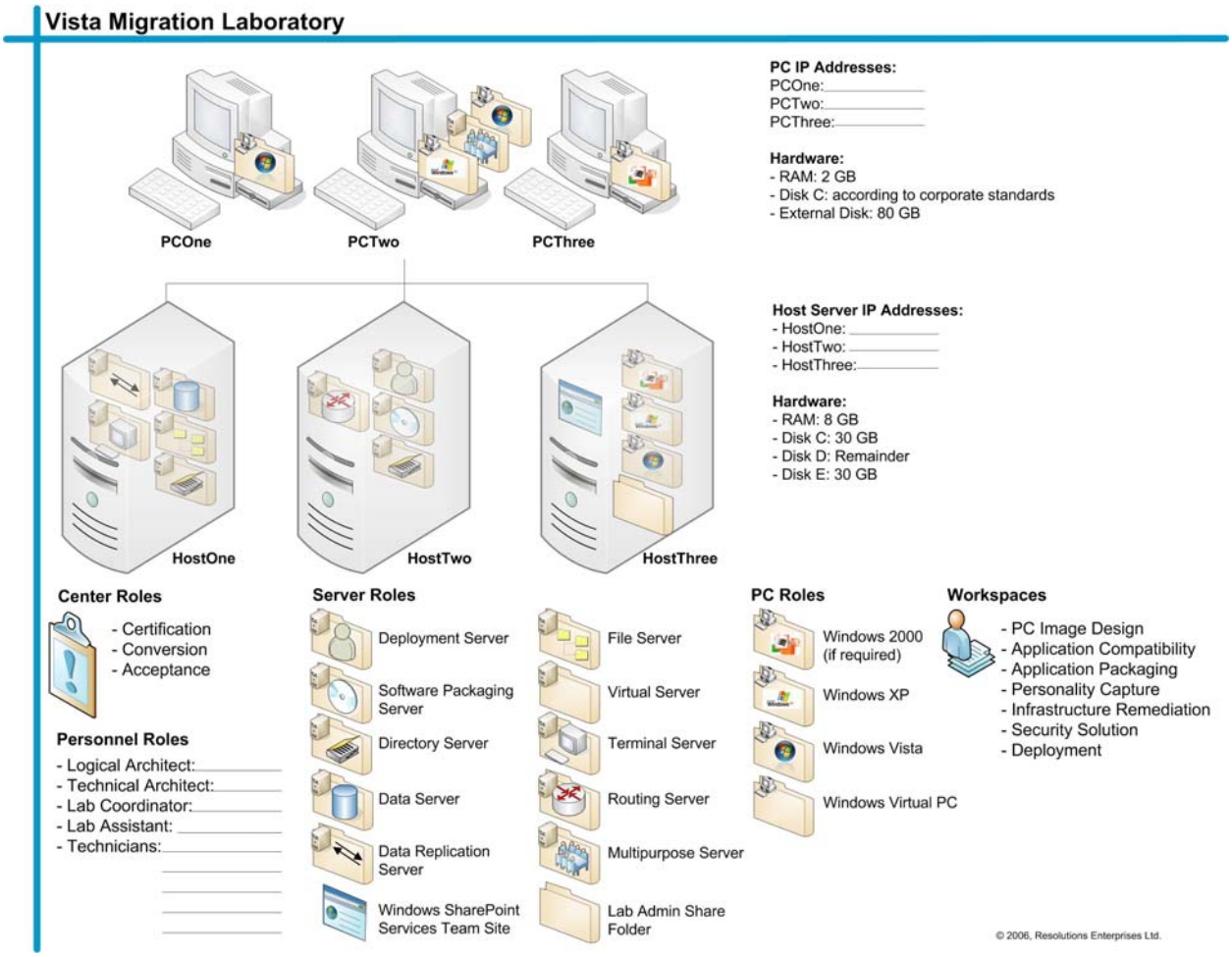

Figure 3.4. The Different Roles required in the Virtual Lab

> ☞ **Reproducing Business Critical Applications**
>
> In some instances it is not possible to reproduce business critical applications—applications such as SAP databases—within the lab environment. There are several strategies you can use (see Figure 3.5) depending on the size of your organization. Small organizations will want to link testing PCs to the production SAP database and will need to proceed with utmost care. Medium organizations may be able to generate a duplicate SAP database with scrubbed data. Large organizations will already have development versions of their SAP databases and can link to those to perform application and compatibility testing. In all cases, testing systems should be isolated from the production network as much as possible.
>
> In any event, make sure you fully back up all critical systems before you begin any form of testing against them.

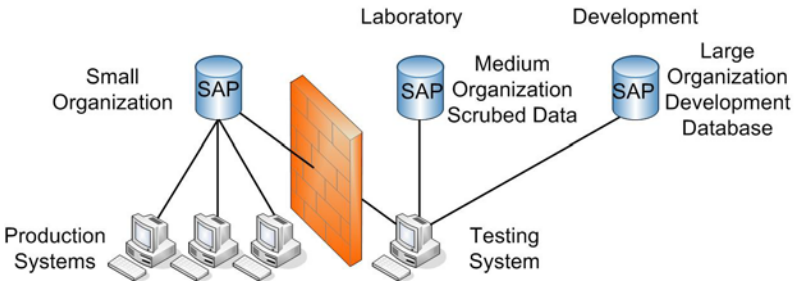

Figure 3.5. Complex Business Critical Testing Scenarios based on Organization Size

Requirements for each Testing Level

Table 3.1 outlines guidelines for the requirements for each testing level. Adjust them according to your own needs.

Test Level	Objective	Virtual Machines	Physical Machines
Unit	Discovery of new features and feature operation	• PC Team: Typical Windows XP PC and Bare Metal • Server Team: Multi-purpose Server plus Windows XP PC and Bare Metal	• None
Functional	Automate features and obtain peer review	• All Teams: Same as Unit	• None
Integration	Link all aspects of the solution together	• All Teams: Several Single-purpose Servers plus Windows XP PC, Bare Metal and Vista	• Testing Stations: all models of retained configurations plus all external hardware devices
Staging	Finalize all technical procedures and prepare for final acceptance testing	• All Teams: Servers represent small-scale production environment plus Windows XP PC, Bare Metal and Vista	• Same as Integration
Pilot	Finalize all technical and administrative procedures	• Few VMs if any	• Use production systems in preparation of massive deployment

Table 3.1. Requirements for each Testing Level

Each of the five environments has its own requirements, but fortunately, you can reuse many of the requirements of a previous level for those of the next. Here's the breakdown of machine reuse.

- Unit level: individual technicians work with their own machines stored on an external disk and linked to their own PC.
- Functional level: as team members graduate from Unit to Functional, they reuse the original individual machines they had in Unit since they are no longer needed at that level.
- Integration level: all team members begin to share the same set of virtual and physical machines as they begin to integrate their part of the solution into a single environment. Change control is activated to capture all practices.
- Staging level: all technical team members share an environment which is a scaled down version of the production environment. Change control is absolute and no change can be performed without being tracked. Procedures are completely finalized and are tested from end to end.
- Pilot level: All team members, including administrative staff, use production systems to target a deployment to about 10% of the population.

In addition to the requirements for testing levels, you'll also need to understand the requirements for the three environments—Certification, Conversion and Acceptance—the lab will support.

The Certification requirements are listed in Table 3.1 because this environment comprises each of the testing levels. The requirements for the other two environments are listed in Table 3.2.

Environment	Objective	Virtual Machines	Physical Machines
Conversion	Convert data and applications, both end-user developed and corporate	• Data Conversion: Possibly a Vista PC linked to the Staging test level • End-User Apps: A Vista PC linked to either Integration or Staging • Corporate Apps: A Vista PC linked to either Integration or Staging	• All: PCs with access to the virtual environments • Corporate Apps: Possibly Vista PCs with Aero interface linked to either Integration or Staging
Acceptance	Provide acceptance of software packages and of the overall solution	• Software Owners: Vista PC linked to Integration • IT Stakeholders: Reuse Staging	• All: PCs with access to the virtual environments • IT Stakeholders: All baseline systems retained to support Vista

Table 3.2. Requirements for Conversion and Acceptance Environments

Creating the Lab Environment

The actual creation of the laboratory environment is simpler once you understand how it should work. If you want to get your technical team members going early, all you need to do is prepare the virtual machines that are required for Unit and Functional testing. These are fairly quick to prepare, but you should still proceed with care. Ideally, you will already have initiated the acquisitions required for these VMs—external drives and perhaps high performance PCs—and have determined which virtualization technology you want to use as well as obtain the virtualization software. This will be the first of many source installation files that the lab will be responsible for maintaining. You will also need to obtain source installation media for all of the operating systems you will be working with. To make them easier to work with, transform the install media into ISO files because these act as CD drives in virtual machines.

Once you're ready, prepare these two environments in the following order. Rely on an existing workstation with sufficient capabilities to perform these steps.

1. Install the virtualization technology on a workstation with sufficient capabilities to create virtual machines.
2. Create your first virtual machine, assigning RAM, disk space and network interface cards. It is best to create a new folder with the machine name and store all VM component files into this folder.
3. Connect the VM's CD/DVD drive to the ISO file for the OS you want to install.
4. Perform the installation, running through your standard procedures for OS installation in your organization.

> ☞ If you also elected to obtain a physical to virtual conversion tool, you can also create this machine by converting an existing physical machine into a VM.

5. Customize the machine as you normally would, update the default user and apply any patches or updates that are required.
6. For any copy of Windows prior to Windows Vista, copy the following files from the Deploy.cab file located on the appropriate Windows installation CD to a new folder called SysPrep on the %systemroot% drive—usually the C: drive:
    a. Setupmgr.exe
    b. SysPrep.exe
    c. Factory.exe
    d. Setupcl.exe
7. Run Setupmgr.exe to generate a SysPrep.inf file. Use your organizational standards to provide the answers required for this file. Close Setupmgr.exe.
8. Copy the VM's entire folder, rename the folder and the VM files to "machinename SysPrepped" and open the new machine in your virtual machine tool.
9. Run SysPrep.exe on the machine to select the Reseal option, depersonalize it, and shut it down. You now have a source machine from which you can generate a number of copies.

10. Since Unit and Functional levels require at least three source machines: WS03 Standard, WS03 Enterprise and Windows XP, repeat this process until each one is created.
11. Create a fourth source machine with no OS. This will become the bare metal machine testers will use for the Vista installation.
12. Document each machine, listing user accounts and capabilities, copy them onto the external disks and provide the disks to your technical team members.

That's it. Now that your team members are ready to proceed with their own work, you can move on to create and prepare the working environment for the lab as well as preparing the Integration and Staging environments.

Virtual Machines and Software Licensing

Even though you're working with virtual machines, you still have to be conscious of licensing issues, especially if you're building a laboratory to last. For this reason, we don't recommend using evaluation copies of software or operating systems. Here are some general guidelines on how you should license virtual machines. You should verify with your vendor to make sure these guidelines meet their licensing requirements.

- SysPrep machine: A SysPrep machine does not require a license because it is a machine that is used only to seed other machines and doesn't actually get used as is. Once you've copied the SysPrep machine and start personalizing it, you need a license for the machine.
- Running virtual machines: Each machine that is named and is running on a constant basis needs to have its own license.
- Copied virtual machines: Each copy of a virtual machine does not need its own license so long as they are not running at the same time.
- Copied and renamed virtual machines: Each time you copy a virtual machine and rename it, you need to assign a license to it. A renamed machine is treated as a completely different machine and therefore needs a license.

Using either the Microsoft Developer Network (MSDN) or TechNet Plus subscriptions, you have access to ten licenses of each product, though each license needs activation. Both subscriptions support the legal reuse of virtual machines.

Note: If the host operating system is WS03 Enterprise Edition, then you can run four server VMs at no additional cost. This makes a good argument for making the host systems run this OS.

More information on virtual machine licensing for Microsoft Windows Server 2003 can be found here: http://www.microsoft.com/licensing/highlights/virtualization/faq.mspx.

When you're ready to build the lab itself as well as Integration and Staging, you'll need to perform the following activities:

- Begin with the base server installation(s) for the host servers
- Create file shares for the lab repositories
- Install virtual machine (VM) software on host servers
- Create the VMs that simulate production servers and other aspects of the production environment

You'll find that you need a great deal of storage space. You must plan for storage space, keeping in mind the following requirements:

- Space for storing all lab data
- Space for the installation media for all OSs and for all retained software products
- Space for the images built during testing—images of both VMs and physical systems
- Space for backing up virtual hard disks and duplicating entire virtual environments

On average, a minimum of 200 GB of space is required, but this number is affected by the number of disk images, VMs, and application packages your project covers and it does not include the storage space required on the host servers themselves.

When you build the Integration and Staging test levels, remember that they need to be as similar to production as possible. Also remember that they don't need to be identical. For example, there is no reason to use the same Active Directory forest or domain names since they do not affect the PC deployment process. Ideally, you'll use proper names for AD components, names that make sense and that you can keep on a permanent basis. In our opinion, it is always best to actually acquire proper domain names for this purpose because they are permanent and this low cost is not a burden on the project's budget.

When you work on the client computer portion of the lab, design it to test the same functions and features currently in use or planned use in the production environment. Include the same types of hardware, applications, and network configurations.

> ☞ Remember, the hardware components of the lab can be limited to the actual computers targeted for deployment and the host servers for VMs. In addition, the host servers can double as file servers for lab data. There is no reason for the actual production server hardware to be duplicated so long as all of the services provided in production are duplicated in VMs.
>
> With the use of the right hardware, the lab can also be transformed into a portable environment. Several manufacturers are now releasing very powerful portable hardware. For example, NextComputing LLC offers a portable lab server through its NextDimension system. This system can include multiple dual-core CPUs, loads of RAM, multiple serial ATA (SATA) drives with lots of space and even multiple monitors, all in a transportable format. For more information on the NextDimension, go to http://www.nextcomputing.com/products/nextdim.shtml.

Once you've built out the entire lab including machines for use in each of the testing levels, you can build a custom Microsoft Management Console (MMC) using the Remote Desktops snap-in to create a single console that will give you access to all machines, virtual and physical (see Figure 3.6). Add multiple copies of the Remote Desktops snap-in—one for each of the different testing levels and one for the physical hosts—to differentiate between each level. Also, use significant names to easily identify which machine you are working with. Make sure Remote Desktop connections are enabled in each machine and you can use this console to manage each and every environment.

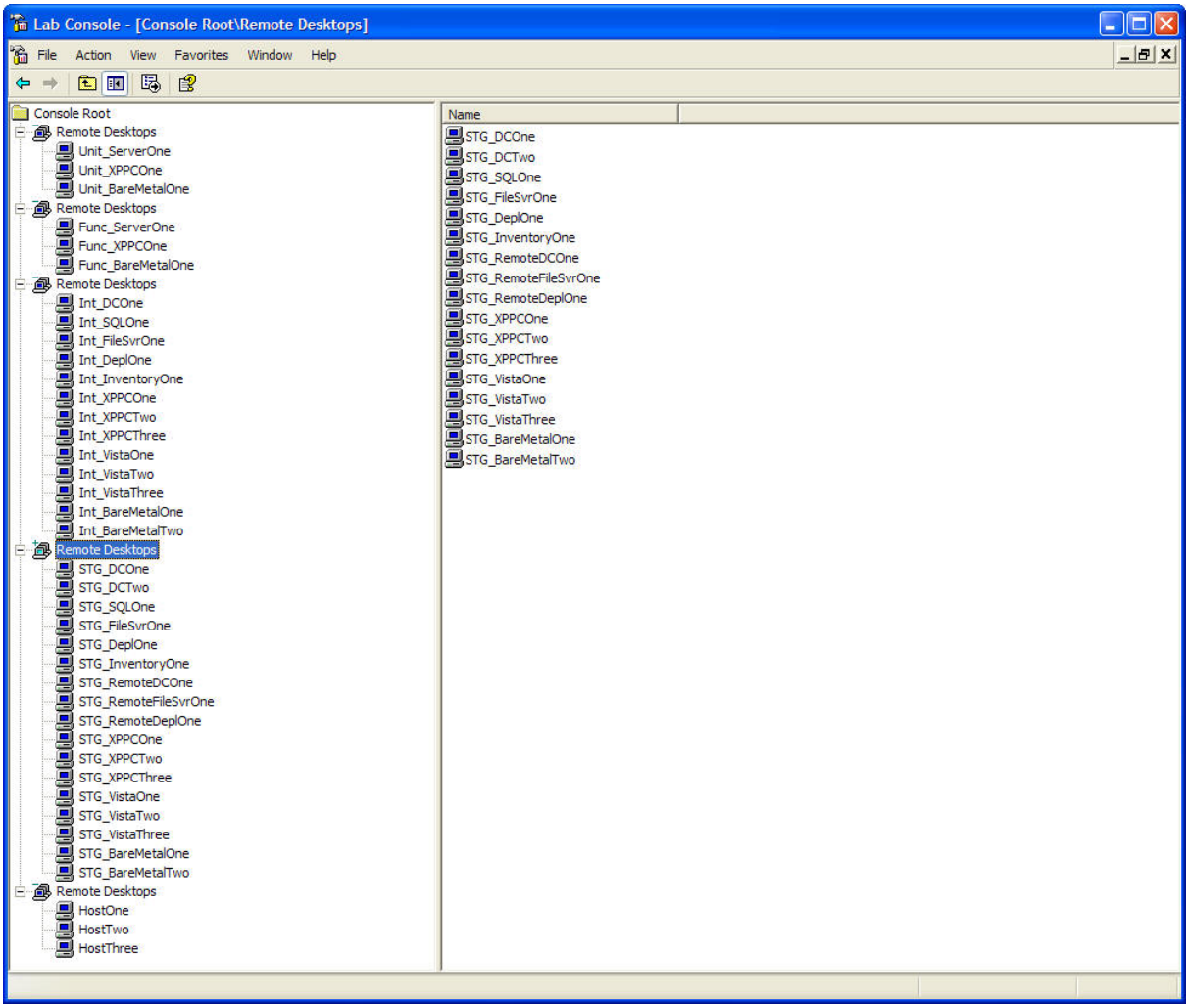

Figure 3.6. Multiple copies of the Remote Desktops snap-in in an MMC give you access to all testing levels

The lab coordinator and the lab assistant are responsible for the construction of the lab. They will draw upon some technical resources for this construction, and then need to put in processes for the reuse of VMs and other lab components. The lab coordinator will also be responsible for the creation of a lab schedule and perhaps a team workspace. Microsoft SharePoint Portal Server is ideal for the support of both of these tasks. You can use a copy from your MSDN or TechNet licenses to put it in place. SharePoint is also ideal for the creation and maintenance of the deviance registry—the tool you will use to track any issue, both technical and administrative, the project will encounter as it creates the solution for deployment. Once again, the lab manager will be responsible for the maintenance of this registry.

> ☞ You can use either Windows SharePoint Services (free with a license of WS03) or SharePoint Portal Server. SharePoint Services are easier to implement, but both require a back end database. In addition, you can find a template for a project team site here. More templates for SharePoint can be found here.

The lab includes project deliverables, the lab itself being one of them. Other deliverables include:

- Laboratory description: this outlines the strategy you will use to create and implement the environments supported by the lab.
- Technical laboratory deliverables: this helps identify how the deliverables from the laboratory can be used in support of other testing or development scenarios. With virtual laboratories in particular, it's really easy to include pre-constructed machines as deliverables to other projects.
- Laboratory management practices: the practices you're going to use for the management and operation of the laboratory.
- Future plans and projected growth**:** to look beyond the immediate and cover both best practices and recommendations for future lab usage as well as for the creation and management of a distributed virtual laboratory structure as more and more members of the organization require access to virtual technologies.

When the technical builds of the test levels are complete, they should be fully tested to ensure that they do indeed reproduce the production environment and that they will support testing of the entire deployment solution, including image and software deployment.

Then, once this is complete, your lab will be ready for prime time and will be able to support all aspects of the solution preparation.

> 🕮 You can build your own lab, or if you want to fast-track this aspect of your project, you can rely on virtual appliances—pre-built VMs that include all of the required features to support each testing level—that have been prepared by third parties. Click here to find out more about virtual appliances in support of the Desktop Deployment Lifecycle.