Note from the Author

This guide is primarily focused on helping three distinct types of audience:
1. Customers who are in the early stages of Adopting VMware Infrastructure solutions.
2. Existing IBM WebSphere customers who are considering deploying VMware Infrastructure solutions.
3. Existing VMware customers who are considering implementing new IBM WebSphere application environments.

By using this guide, customers can reduce the cost and time to deployment of an IBM WebSphere Application Server environment on the VMware Infrastructure.

This deployment guide is based on the joint IBM WebSphere and VMware solution development work performed at the IBM Innovation Center, San Mateo, CA, USA.

I would like to thank the following team members for providing valuable support to roll out this deployment guide.

IBM Innovation Center (IIC), San Mateo, CA

IBM:
David Ho, Sr. Technical Consultant, IBM WebSphere Solutions (IIC)
Ruth Willenborg, IBM Software Group

VMware:
Kristen Edwards, Sr. Global Alliance Manager, IBM Alliance
Hatem Naguib, Director, Global Partner Solutions & Marketing
Johanna Holopainen, Sr. Manager, ISV Alliances, GTM
# Table of Contents

1.0 **Introduction** .................................................................................................................. 5  
   1.1 IBM WebSphere Application Server........................................................................... 5  
   1.2 WebSphere Application Server Architecture ........................................................ 5  
       1.2.1 Stand-Alone Server Configuration ...................................................... 6  
       1.2.2 Distributed Server Configuration .................................................... 7  
   1.3 VMware Infrastructure ........................................................................................... 7  
   1.4 WebSphere-VMware Solution Architecture ........................................................ 10  
   1.5 WebSphere-VMware Solutions ........................................................................... 11  
   1.6 Deployment Approach Summary ....................................................................... 12  
   1.7 Lab Environment ................................................................................................. 14  

2.0 **Installation of VMware Infrastructure 3 Environment** ............................................. 15  
   2.1 Installing VMware VirtualCenter 2 ....................................................................... 15  
   2.2 Installing Virtual Infrastructure Client .................................................................. 21  
   2.3 Configuring VirtualCenter for Server Based Licensing ....................................... 24  
   2.4 VMware ESX Server 3 Installation...................................................................... 25  

3.0 **VMware ESX Server Management with VMware VirtualCenter** ............................. 27  
   3.1 Adding ESX Server Host to VirtualCenter........................................................... 27  
   3.2 Configuring ESX Server Host to Use Server Based Licensing ............................. 31  
   3.3 Creating a Virtual Machine .................................................................................. 32  
   3.4 Guest Operating System Installation ................................................................... 40  

4.0 **IBM WebSphere Software Installation on Virtual Machines** ................................ 48  
   4.1 Installation of WebSphere 6.0 Code Base .......................................................... 48  
   4.2 Installation of WebSphere Fixpacks.................................................................... 52  
   4.3 Installing Database Client ................................................................................... 56  

5.0 **Deployment of IBM WebSphere Application Server Virtual Machines (WASVM)** ...................................................................................................................... 61  
   5.1 Templates ............................................................................................................ 61  
   5.2 Creating Templates ............................................................................................. 61  
   5.3 Provisioning WebSphere Application Server (WAS) Virtual Machine Nodes from Templates ........................................................................................................... 65  
       5.3.1 Preparing for Guest Operating System Customization .......................... 65  
       5.3.2 Cloning a New WAS Virtual Machine from Template ....................... 66  
       5.3.3 Provisioning WAS Nodes Using Customization Specification............ 78  

6.0 **WASVM Application Customization** ....................................................................... 81  
   6.1 Creating Deployment Manager Profile on WASVM ........................................... 81  
   6.2 Creating Application Server Profile on WASVM .............................................. 89  
   6.3 Installing Load Balancing in IBM WebSphere Application Server (WAS) Virtual Machine Environment ................................................................. 98  
   6.4 Clustering and Load Balancing in WAS Environment ........................................ 106  
       6.4.1 Creating a WAS Cluster ................................................................. 106  
       6.4.2 Installing Application on WAS Cluster ........................................... 113  
       6.4.3 Configuring WAS Cluster Load Balancing ................................. 121
7.0 Upgrading IBM WebSphere Application Server Environments on Virtual Machines (WASVM) ................................................................. 123
  7.1 Snapshots in WASVM Environment........................................................................ 123
    7.1.1 Taking a WASVM Snapshot ........................................................................ 123
  7.2 Upgrading Production IBM WebSphere Application Server (WAS) Using VM Cloning ................................................................. 126
    7.2.1 Creating a New Virtual Switch .................................................................. 126
    7.2.2 Creating a Clone for Upgrade ............................................................... 130
  7.3 Upgrading WAS 6.0 to WAS 6.1 ....................................................................... 136
  7.4 Migrating WAS 6.1 VM Clone to Production ..................................................... 146
    7.4.1 Verification of the Upgrade and WAS Functionality .............................. 146

8.0 Converting Physical IBM WebSphere Application Server Instance to an IBM WebSphere Application Server Virtual Machine......................... 150
  8.1 Installing VMware P2V Assistant .................................................................... 151
  8.2 Creating a Target Virtual Machine (VM) .......................................................... 154
  8.3 Adding Target Virtual Disk to Helper VM ...................................................... 158
  8.4 Source System Preparation ............................................................................ 162
  8.5 Cloning Source Disk to Virtual Disk ............................................................... 162
  8.6 Reconfiguration of Cloned Virtual Disk ......................................................... 168
  8.7 Booting Target VM with Cloned Virtual Disk ............................................... 171

Appendix A: VMware Migration for an IBM WebSphere Application Server Virtual Machine ................................................................. 174
Appendix B: Configuring Additional Storage on VMware ESX Server Host (local or SAN) ................................................................. 185
Appendix C: References .......................................................................................... 189
1.0 Introduction

IBM and VMware have been working on joint initiatives to provide customers with proven, scalable and optimized solutions of IBM hardware and software products on VMware Infrastructure. IBM® WebSphere® software platform is one of the most widely deployed application server platforms in the IT industry today. By leveraging the power of infrastructure virtualization solutions delivered by VMware, IBM WebSphere Application Server (WAS) deployments can be consolidated and provisioning as well as change management processes improved. This deployment guide discusses how to deploy IBM WebSphere Application Server on the VMware Infrastructure platform. It also showcases the key solutions that VMware virtualization technology brings to WAS deployments.

1.1 IBM WebSphere Application Server

As the foundation of the IBM WebSphere software platform, the WebSphere Application Server (WAS) 6.0/6.1 delivers the secure, scalable, resilient application infrastructure needed for a Service Oriented Architecture (SOA).

WAS leverages the industry’s premier Java™ 2 Enterprise Edition (J2EE™) 1.4 and Web services application platform to help build, run, integrate and manage dynamic, On Demand business application solutions and helps deliver on the promise of SOA by enabling IT organizations to:

- Build and deploy application services quickly and easily
- Run services in the most secure, scalable, highly available environment
- Reuse software assets and extend their reach
- Manage applications effortlessly
- Grow as needs evolve, leveraging core assets and skills

Designed for full Java™ 2 Platform, Enterprise Edition (J2EE) V1.4 compatibility and support for Java 2 Platform, Standard Edition (J2SE) 5.0, WebSphere Application Server, V6.1 delivers a range of new features that can help you:

- Improve time to value and make the most of existing technology skills with more simple, rapid development and deployment.
- Be confident applications and data are secure, and that you can eliminate lost business opportunities with a more scalable, secure, highly available SOA runtime environment.
- Improve the flexibility of your business and increase return on investment with extensive communication services.
- Minimize the cost of managing your environment with effective application management tools.

For customers who need to develop, deploy and manage secure, portable applications that run on a multitude of server and desktop systems, WebSphere Application Server V6.1 is the premier platform of choice.

1.2 WebSphere Application Server Architecture

WebSphere Application Server is the implementation by IBM of the Java 2 Enterprise Edition (J2EE) platform. It conforms to the J2EE 1.4 specification. WebSphere Application Server is available in three unique packages that are designed to meet a wide range of client requirements.

1. IBM WAS – Express
2. IBM WAS – Base
3. IBM WAS – Network Deployment
At the heart of each member of the WebSphere Application Server family is an application server. Each family has essentially the same architectural structure. Although the application server structure for Base and Express is identical, there are differences in licensing terms, the provided development tool, and platform support. With Base and Express, you are limited to stand-alone application servers. Each stand-alone application server provides a fully functional J2EE 1.4 environment.

Network Deployment has additional elements that allow for more advanced topologies such as workload management, scalability, high availability, and central management of multiple application servers.

The IBM WAS Extended Deployment is an add-on Quality of Service extension to the WAS product set to provide high levels of scalability, manageability and performance. This feature is out of scope of this document.

1.2.1 Stand-Alone Server Configuration

Express, Base, and Network Deployment all support a single stand-alone server environment. With a stand-alone configuration, each application server acts as a unique entity. An application server runs one or more J2EE applications and provides the services required to run those applications.

Multiple stand-alone application servers can exist on a machine, either through independent installations of the WebSphere Application Server code or through multiple configuration profiles within one installation. However, WebSphere Application Server does not provide for common management or administration for multiple application servers. Stand-alone application servers do not provide workload management or failover capabilities.

Figure 1.1 shows an architectural overview of a stand-alone application server.
1.2.2 Distributed Server Configuration

With Network Deployment, you can build a distributed server configuration, which enables central administration, workload management, and failover. In this environment, you integrate one or more application servers into a cell that is managed by a deployment manager. The application servers can reside on the same machine as the deployment manager or on multiple separate machines. Administration and management are handled centrally from the administration interfaces through the deployment manager.

With this configuration, you can create multiple application servers to run unique sets of applications and then manage those applications from a central location. More importantly, you can cluster application servers to allow for workload management and failover capabilities. Applications that you install in the cluster are replicated across the application servers. When one server fails, another server in the cluster continues processing. Workload is distributed among Web containers and EJB containers in a cluster using a weighted round-robin scheme.

Figure 1.2 illustrates the basic components of an application server in a distributed server environment.

![Diagram of Distributed Server Configuration]

This deployment guide will focus on both of these configurations.

1.3 VMware Infrastructure

VMware Infrastructure is the most widely deployed software suite for optimizing and managing IT environments through virtualization – from the desktop to the data center. The only production-ready virtualization suite, VMware Infrastructure is proven to deliver results at more than 20,000 customers of all sizes, used in a wide variety of environments and applications. The suite is fully optimized, rigorously tested and certified for the widest range of hardware, operating systems and software applications allowing for enterprise-wide standardization independent of operating systems and hardware. VMware Infrastructure provides built-in management, resource optimization, application availability and operational automation capabilities that deliver transformative cost savings as well as increased operational efficiency, flexibility and IT service levels.
VMware Infrastructure virtualizes and aggregates the underlying physical hardware resources across multiple systems and provides pools of virtual resources to datacenter in the virtual environment. In addition, VMware Infrastructure brings about a set of distributed services that enables fine-grain, policy-driven resource allocation, high availability, and consolidated backup of the entire virtual datacenter. These distributed services enable an IT organization to establish and meet their production Service Level Agreements with their customers in a cost effective manner.

The relationships among the various components of the VMware Infrastructure are shown in figure 1.3.

VMware Infrastructure includes the following components:
Solutions Deployment Guide

**VMware ESX Server** - A robust, production-proven virtualization layer run on physical servers that abstracts processor, memory, storage, and networking resources into multiple virtual machines.

**VirtualCenter Management Server (VirtualCenter Server)** - The central point for configuring, provisioning, and managing virtualized IT environments.

**Virtual Infrastructure Client (VI Client)** - An interface that allows users to connect remotely to the VirtualCenter Server or individual ESX Servers from any Windows PC.

**Virtual Infrastructure Web Access (VI Web Access)** - A Web interface that allows virtual machine management and access to remote consoles.


**VMware Virtual Symmetric Multi-Processing (SMP)** - Feature that enables a single virtual machine to use multiple physical processors simultaneously.

**VMware VMotion** - Feature that enables the live migration of running virtual machines from one physical server to another with zero down time, continuous service availability, and complete transaction integrity.

**VMware HA** - Feature that provides easy-to-use, cost-effective high availability for applications running in virtual machines. In the event of server failure, affected virtual machines are automatically restarted on other production servers that have spare capacity.

**VMware Distributed Resource Scheduler (DRS)** - Feature that allocates and balances computing capacity dynamically across collections of hardware resources for virtual machines.

**VMware Consolidated Backup (Consolidated Backup)** - Feature that provides an easy-to-use, centralized facility for agent-free backup of virtual machines. It simplifies backup administration and reduces the load on ESX Servers.

**VMware Infrastructure SDK** - Feature that provides a standard interface for VMware and third-party solutions to access the VMware Infrastructure.

Figure 1.4 illustrates the six basic components of virtualization infrastructure that are part of our deployment guide.

A single VirtualCenter manages multiple ESX Server hosts.

![Figure 1.4](image-url)
1.4 WebSphere–VMware Solution Architecture

Figure 1.5 represents a joint IBM Software and VMware solution architecture for running WebSphere Application Servers and different applications on VMware Infrastructure. It demonstrates the ability to run several applications on same physical server by creating virtual machine nodes containing the applications, application server instances and the operating system of choice. Using VMware Infrastructure enables deployment of virtual machines running different operating systems on the same physical server.
1.5 WebSphere-VMware Solutions

WebSphere Application Server Virtual Machine (WASVM) deployments discussed in this document will cover the following solutions that VMware Infrastructure suite brings to an enterprise WebSphere Application Server (WAS) deployment.

Server Containment

In many traditional application deployments based on WAS, there is significant server sprawl due to the use of multiple physical systems hosting the different layers of the application. This server sprawl can be seen across the development, test and production environments. For example, each developer has his own system and each test cycle uses dedicated servers assigned for the duration of the tests. This can lead to over provisioning, manageability and resource issues resulting in higher ownership and operational costs.

VMware’s virtualization technology is able to contain server sprawl by running IBM WebSphere Application Servers in virtual machines consolidated onto fewer, highly scalable, reliable enterprise-class servers leading to increased server utilization.

Key Benefits:
1. Dedicated and isolated developer environments for all developers consolidated to a few physical systems.
2. Multiple WebSphere Application Servers on same physical system providing significant consolidation and lowering total cost of ownership (TCO).
3. Multiple operating system (OS) and application server versions on the same system eliminating the need for dedicated hardware.
4. Multiple test environments sharing same physical system eliminating the need for dedicated test systems.

Customers using VMware Infrastructure have been able to achieve significant ratios of consolidation thereby drastically increasing server utilization and containing server sprawl.

Rapid Provisioning

VMware virtualization solutions significantly reduce time to provision new WebSphere (development, test or production) application environments. Typically, for a new deployment, it is required to procure hardware, install the operating system and the applications. This process takes significant time and IT resources in addition to the need for dedicated hardware. While using VMware Infrastructure, IBM WebSphere customers can take advantage of virtual machine libraries and templates to provision new pre-configured application environments in minutes on virtualized infrastructure hardware. This enables rapid WebSphere application deployments with sophisticated automation capabilities, centralized control and responsibility for hardware resources while giving business units and application owners complete control over how resources are utilized.

Key Benefits:
1. Rapid provisioning of new WebSphere Application Server instances from virtual machine (VM) templates.
2. WebSphere application development images can be passed directly to testers.
3. Testers can pass WebSphere VM images back to development for problem replication and resolution.
4. Recreate distributed WebSphere Application Server instances in production environment on a single “virtualized” physical system for test purposes.
5. Move test/QA application server instances to production in minutes.
6. Reset test images (after test completion) from templates, snapshots and virtual machine libraries cutting down on test setup and reset time.
7. Store different WebSphere applications and server versions in virtual machine libraries that can be provisioned instantly.
8. Rollback development and test images using VM snapshots during problem resolution.
Change Management

Patching and upgrading existing applications is high on the list of IT challenges facing any WebSphere customer. IT departments face two key challenges in change management – testing patches and upgrades for compatibility with standard corporate hardware, OS and application configurations and efficiently deploying critical patches and upgrades throughout the enterprise. Traditionally, IT organizations need to procure hardware (mirroring production) and create test beds mirroring the OS and application configurations of the production environment. With VMware Infrastructure, customers can clone production or create a set of virtual machine template libraries mirroring production which can then be used to provision the test environment. The latest patches and upgrades can then be tested against these virtual machines running WebSphere applications, while eliminating the need for dedicated hardware to perform these tests. These patches can then be rolled into production with minimal interruption to end users. In case of problems, the virtual machines can be instantly rolled back using snapshots.

Key Benefits:
1. Faster change management with fewer system resource requirements.
2. Patches can be tested on multiple configurations (OS, WebSphere Application Server versions etc.) concurrently while hosted on the same physical system.
3. Instant rollback of application server virtual machines using snapshots (during problem resolution).
4. Add/upgrade WebSphere applications independent of other WebSphere components in production.
5. Create a library of standard production configurations to perform change management testing and deployment.
6. Migrate WebSphere Application Server instances on virtual machines to other systems while performing maintenance or changes on the current physical system.

The key features of VMware Infrastructure discussed in this deployment guide (including templates, cloning, snapshots, P2V etc.) can be directly leveraged to provide all of the benefits discussed above.

Apart from the above described benefits, VMware Infrastructure also provides significant features for data center optimization, security, high availability, business continuity and manageability in a virtualized enterprise data center environment.

1.6 Deployment Approach Summary

The steps below show a summary of IBM WebSphere Application Server Virtual Machine (WASVM) deployment approach that has been detailed in this document.

1. Install and configure your VMware Infrastructure 3 environment including ESX Server, VirtualCenter and VMware Infrastructure client to manage ESX Server hosts.
2. Create a virtual machine and install all the WebSphere components that should be part of a base image.
   - The base image could contain the WebSphere software, fixpacks and database client.
   - Install any additional software (IT tools, anti-virus, management agents etc.) that need to be part of the base virtual machine image.
3. Create a VM template from the base virtual machine. This template will be used to provision WAS application nodes.
   - This template will act as the standard template for deployment across your WAS environment.
   - You can create additional templates if desired so you have multiple standard images for different layers and applications of the WAS environment.
4. Create WAS nodes (deployment manager, application servers etc.) from the template and perform OS and application customization specific to your WAS environment. Configure the customized load balanced WAS cluster.
   - OS customization would include changes to Windows SID, network configuration, hostname, license keys etc.
   - Application customization would include configuring the different nodes with WAS profiles (deployment manager, application server etc.) and also configuration of a load balanced WAS cluster environment.
   - You can consider building customization scripts that can be invoked as part of the cloning process.
5. If you would like to upgrade or patch your WebSphere environment, consider using
   - Snapshots while directly upgrading production WAS nodes. This will ensure prompt rollback to stable state in case of upgrade issues.
- “VM Cloning” to create a clone of production WAS nodes and migrate it to a test or development environment.
  - Create a private network or add your cloned VM to another network (test or development) that does not see the production VM so you can retain the same network attributes of the production VM.
  - Apply all upgrades and patches, test and validate functionality
  - Migrate the upgraded clone into production VM by just moving the VM network back to the production and taking your existing production VM out of network.

6. If you already have existing WAS environment on a physical system, use VMware P2V Assistant (P2V) to migrate the physical WAS nodes to virtual machines. This will help consolidate the physical WAS nodes on ESX Server(s), while retaining the integrity and validity of the configurations in your WAS environment.
   - Build a target virtual machine and create the target virtual disk
   - Recommended to create the target virtual disk to be of the same size as the source disk.
   - Install P2V on another VM (called helper VM) which has the same OS as the source (physical) system.
   - Attach the target disk created to the helper VM
   - Boot source system using P2V boot disk
   - Clone the source disk(s) to the target disk(s) using P2V
   - Reconfigure target virtual disk to make it bootable.
   - Detach the target virtual disk from helper VM
   - Boot the target virtual machine from the target virtual disk to bring up the virtual machine WAS node.

7. If you need to relocate virtual machines from one ESX Server host to another ESX Server, consider:
   - Cold migration of the virtual machines to move the VM files to a different storage location or ESX Server host.
   - Live migration (with VMware VMotion) to move online VMs to other ESX Server hosts.
1.7 Lab Environment

Find below a summary of our lab setup. The deployment guide makes references to the different virtual machines as you go through the deployment steps.

WASVM1 – Base virtual machine image
WASVM1_Template – Virtual machine template used to provision WAS nodes using "cloning"
WASVMND1 – WAS deployment manager node, also installed IBM HTTP server and load balancer.
WASVMND2 – Application server node
P2VVM – WAS node on physical system converted to a virtual machine
Physical WAS node – run on an IBM x336 system
ESX host IP – 10.26.38.36, 10.26.38.95

Figure 1.6
2.0 Installation of VMware Infrastructure 3 Environment

This section will focus on the installation of the following VMware Infrastructure 3 components:
1. VMware VirtualCenter 2 and license server
2. Virtual Infrastructure Client
3. VMware ESX Server 3

Prior to installing VirtualCenter you will have to decide between host based or server based licensing. ESX Server hosts have the option of having local license files (host based) or server based licensing. Since VMware Infrastructure 3 is a data center infrastructure, it is easier to manage licenses for multiple ESX Server hosts using centralized server based licensing. Our installation used server based licensing.

For details on licensing options, refer to the VMware Infrastructure 3 product documentation at www.vmware.com.

2.1 Installing VMware VirtualCenter 2

Access the CD or downloaded s/w (zip file should be unzipped).

Click “Autorun”

![VirtualCenter Installer](image)

Click on VirtualCenter Management Server
The VMware VirtualCenter version 2 default installation includes the following components:

VirtualCenter Server – A windows service to manage ESX Server hosts
.Net Framework – Software used by the VirtualCenter server and VI client.
Web Access – Web Server to allow browser-based virtual machine management.
Web Service – Software Development kit (SDK) for VMware products.
License Server – A windows service allowing all VMware products to be licensed from a central pool and managed from one console.
Take the default folder offered or provide the appropriate folder in which you want to install the VirtualCenter server.
VirtualCenter server requires a database to store and organize server data. VirtualCenter 2 supports Oracle, SQL server and Microsoft MSDE. The VirtualCenter Management Server requires administration credentials (ID and password) to log on to the database. Pls. refer to the VMware documentation for configuring your production database for VirtualCenter 2.

The lab installation used a MSDE database for VirtualCenter installation.

**Note:** MSDE is not supported for production environments.
Install the license server that will be used by VirtualCenter as well as the ESX Servers (hosts). The lab setup used “server based licensing” for the VMware Infrastructure 3 ESX Server hosts.

The license file can be downloaded from VMware or sent to the customers by email. Save the file and provide the file location information.
Accept the default ports unless you want to specifically assign your own port information.

Note: these ports might need to be opened if you need to access Virtual Center server through the firewall.
Once you click "install", the installer will install the VirtualCenter 2 server, license server, install and configure MSDE database .net Framework 1.1, VirtualCenter Repository etc.

### 2.2 Installing Virtual Infrastructure Client

Refer to your product documentation for specific requirements for the Virtual Infrastructure (VI) client.

Run "autorun.exe" from the installer CD or from the downloaded (unzipped) file.

Click on Virtual Infrastructure Client.
Take the default directory or provide the appropriate folder for VI client installation.

Exit the installer at this point. You have installed VirtualCenter 2 Management Server, Database, license server and VI client. The license server will be used to support the ESX Server hosts.
It is recommended to install the license server on the same box as the VirtualCenter server.
2.3 Configuring VirtualCenter for Server Based Licensing

Start the VI client from your start menu or from the desktop icon.

Please provide the virtual server hostname or IP.

Username: <admin user of windows environment>
Password: <password>

Go to "administration" \rightarrow "server setting" in the drop down menu.
Specify if the license server is on another system (note: **27000 is the default port used by license server**) or on the same host as VirtualCenter. The lab installation had the license server (with licenses) on the same host as VirtualCenter.

Check “change host license server settings....” option. This will help override the host’s license settings and will force the ESX Server hosts to use the license (server based) settings used by VirtualCenter.

**Note:** If you are likely to have a mix of host based and server based licensing for ESX Server, do not check mark this box.

### 2.4 VMware ESX Server 3 Installation

The VMware ESX Server Version 3 installation includes the following components:

- **VMware ESX Server** – Software to manage and serve virtual machines.
- **VMware Web Access** – Software to allow web browser access to the ESX Server host.

Put the ESX Server 3 CD in the cdrom drive. “Power on” the machine.

Wait for the ESX Server 3 Boot prompt. For Graphical mode install press “Enter”

On ESX Server 3 installer wizard screen press “next”

Take the appropriate defaults (as displayed) for Language (English), Mouse, etc.

Agree to the license agreement

Select the disk drive on which to install ESX Server.

Take the “Recommended” option unless you want to manually create all the partitions on the disks for the ESX Server install. We used a SCSI disk of 70GB size.

**Note:** The installer displays a warning if you attempt to install ESX Server software on an IDE drive or a SATA drive in ATA emulation mode. VMFS is not supported on IDE or SATA. An ESX Server host must have SCSI storage, NAS, or a SAN on which to store virtual machines. See “selecting a boot drive” in the ESX Server installation documentation for all boot disk options.

Click “yes” when prompted with “are you sure you want to remove all partitions on the drive”.

25
Solutions Deployment Guide

Take the “default” partitioning information for the disk partitions.

For ESX Server boot specification,

Take the default (from a drive – install on the MBR of the drive).

Provide the network interface to be used and configure the network address and host name. You can use DHCP or provide the network information manually.

**Note:** Initially ESX Server as well as the Virtual machines (VM) created will share this network interface. You can create additional network interfaces for the virtual machines after installing ESX Server (using the VI client).

It is recommended to use static IP for ESX Server hosts to simplify access and management of the Virtual Infrastructure through the VI client.

Provide all the appropriate network parameters.

Our lab installation used:
- IP address: 172.26.38.36
- Subnet mask: 255.255.255.0
- Gateway: 172.26.38.1
- Hostname: x336036
- VLAN ID: …

Check “create a default network for VMs”.

Enter your root password.

Click “next” on the ESX Server install summary page.

Installation will now begin.

Once installation is complete, the system will reboot and you can access the ESX Server from VI client or web browser using the ESX Server IP address. We will be using the VI client and the VirtualCenter to manage our ESX Server. Log file for the installation is located at /root/install.log.
3.0 VMware ESX Server Management with VirtualCenter

In this section, we will focus on the following steps:
1. Add ESX Server host to VirtualCenter
2. Configure ESX Server host for server based licensing
3. Create a virtual machine (VM)
4. Install the guest operating system on the VM

3.1 Adding ESX Server host to VirtualCenter

Go to VI client and login in to VirtualCenter.

Right click on “hosts & clusters” and select “new data center” (we will call it WASVMDC).

Right click on “WASVMDC” and select “Add host”.

27
Enter the IP address of the ESX Server just setup and provide the administrator account (root) and password.

The host is identified (there is a VirtualCenter host agent that is installed along with ESX Server).
This location selection adds all the virtual machines from this host under WASVMDC data center in the VirtualCenter inventory list and the default ESX Server host datastore “storage1”.

This will add the standalone ESX Server host to the VirtualCenter inventory list.
You can monitor “add host” task progress in the “Recent Tasks” window at the bottom of the VI client.

Click now on the host IP/Name to show the host details.

The “Configuration” tab can be used to set up different configuration information (including licenses, storage, network, security etc.) for the ESX Server host.
3.2 Configuring ESX Server Host to Use Server Based Licensing

Click on “configuration” tab after selecting <<ESX Server host>>

Click on “licensing features”

Click on the “Edit” button (on the right top corner) for License sources.

Provide the license server hostname or IP.

Click on the “edit” corresponding to “ESX Server license type”
Select “ESX Server Standard” (or the license type you are eligible for).

The window will now show all your license features for the host.

we are now ready to create the virtual machine for our WebSphere Deployment.

### 3.3 Creating a Virtual Machine

Login into VI Client and connect to VirtualCenter/

Right click on <<ESX host>> and click “new virtual machine”.
Create a typical virtual machine (VM) with the default device and configuration options.

The lab installation used the following:

Name of the new VM = WASVM1
Inventory Location = WASVMDC
Make sure you have enough storage to support the VMs being created. Take the default datastore “storage1”. You can create additional storage (local or SAN) later if needed.

Select the operating system you want to install. In our case, it is “MS Windows”.

Select the number of (virtual) processors you want to use for this VM. We picked 2VCPU.

Note: The actual virtual CPUs you want to use depend on your architecture and sizing considerations.
Set your memory size appropriately depending on the OS and application requirements. We picked 512MB for our installation.

Note: this can be changed later by editing the VM settings. You can pick the appropriate memory size for your VM based on your WebSphere application sizing considerations.

Provide the number of NICs you want to use for this VM. We selected “1”. Remember, we installed only one NIC (VM Network) during our ESX Server installation. You can configure additional NIC for use by VMs by using the configuration (Networking) options for the ESX Server host.
Provide size of virtual disk (storage) to host the VM. We picked 8GB for OS + Applications.

Enter the disk size in megabytes (MB) or gigabytes (GB). The default is 4GB. The available space on the selected VMFS volume is listed. You can configure a disk from as small as 1MB to as large as 2TB (2048GB).

The virtual disk should be large enough to hold the guest operating system and all of the software that you intend to install with room for data and growth.

You cannot change the virtual disk’s maximum capacity later, but you can install additional virtual disks later by using the Virtual Machine Properties dialog box.
Review the summary of parameters for the VM to be created.

Click “Finish” to create the Virtual Machine.

The “Recent Tasks” window will show the status. Wait till it shows completed.

Click on WASVM1, the new VM created.
Right click on WASVM1 and “edit settings”.

Here you can edit, remove or add different components. Click on “CD/DVD Drive 1”. This usually shows client device as the default device. You can load the OS using client or host (ESX Server) device or using an existing .iso image in the VM datastores.
We used the ESX Server host device. So select the host device and also click “connect at power on” on top.

Now you are ready to insert the Windows OS CD in the host system drive and start the guest OS install.

Refer to Guest operating system install guide for specific OS requirements.

We used Windows 2003 server SP1 as the guest OS for our VM.
3.4 Guest Operating System Installation

Insert OS CD in the CD drive on the host ESX Server system.

“Power on” the VM (Right click on WASVM1 and then select “power on”).

Click on “console” on the right side tabs.

If the boot process in VM does not recognize your OS disk, check whether it has mounted successfully on the host ESX Server.
Proceed through the regular OS setup (follow standard installation procedure for windows server 2003 SP1).
Note that disk size presented to you is the same as what you had provided as the disk size during “new VM” setup.

Format using NTFS file system.
Windows setup status screens
If you want to provide static IP information, please choose custom settings.
Right click on WASVM1 and select “send Ctrl+Alt+Del”. If you do this manually even though your cursor might point to the VM console window, the VI client system will also execute a “Ctrl+Alt+Del”.

Your display and mouse movement would be a little painful because VMware tools have not been installed. You can use “Ctrl+Alt” to move in and out of the console window of VM and external host (VI client) windows. Once VMware tools are installed, the VM console display and mouse will work fine.
Once logged in, you are ready to install "VM Tools".

Note: It is important to install VM Tools immediately after the Guest OS installation and before you start other activities including application setup, template or migration.

Right click on "WASVM1" virtual machine and select "Install VMware Tools".

Select "typical" install and let the installation complete successfully.

Set the display settings when prompted and once the install is completed, reboot the system.

Navigation should be much easier when the system comes up.

The Lab VM network Information:

You can remote login (display) into the windows system, perform administration and new application installation tasks. Make sure to enable "remote desktop" option on the system (remote tab) menu.
4.0 IBM WebSphere Software Installation on Virtual Machines

Now that a basic Virtual Machine has been built with a Windows Server 2003 SP1 operating environment, we will be installing the WebSphere Application Server (WAS) components as part of the base image.

The following WAS components will be installed on the base image virtual machine:

1. IBM WebSphere V6.0 Code Base
2. WebSphere 6.0 Fixpacks
3. Database Client

The above components are the most typical of a WAS environment.

Optional: Any other custom application (including OS patches, security, anti-virus, system tools etc.) that should be deployed across the WebSphere application environment.

4.1 Installation of WebSphere 6.0 Code Base

Install from your CD or downloaded files (expanded) onto the windows virtual machine just created (WASVM1).

Click on “install.exe”

Accept license agreement
Provide directory to install WebSphere or accept the default.

If you do not intend on working with sample applications, deselect it.
Solutions Deployment Guide

VMware and IBM WebSphere Software

---

**Installation wizard**

Select IBM WebSphere Application Server Network Deployment, V8 features to install. See the INSTALLATIONihil file in the data directory for descriptions of the features:

- ProductInstallation
  - Core products files
    - Application Server Samples
    - Javadocs

---

**Installation wizard**

Installation summary for IBM WebSphere Application Server Network Deployment, V8:

Review the summary for correctness. Click Back to change values on previous panels. If the information is correct, click Next to start installing IBM WebSphere Application Server Network Deployment, V8.

IBM WebSphere Application Server Network Deployment, V8 will be installed in the following location:

**C:\WebSphereAppServer**

with the following features:

- Core product files
- Javado16

for a total size:

698.7 MB

---

**Installation wizard**

Installing IBM WebSphere Application Server Network Deployment, V8.

Please wait...

Retracting...
Uncheck “launch profile creation”. We will create WAS profiles later during virtual machine customization (WASVM Application customization section)

Uncheck “launch the first steps console” and click “finish”.

51
4.2 Installation of WebSphere Fixpacks

We should also install the various WebSphere fixpacks for version 6.0.

We downloaded the fixpacks and placed it on WASVM1 virtual machine.

Open the directory and move “update installer” to the websphere home directory where you had installed the code base – (in the lab it was c:\WebSphere\AppServer)

Go to the WebSphere home directory\updateinstaller

Run “update.exe”
Note: This part installs the JDK required by the maintenance package.

The following maintenance package requires an update to an installed JDK that is currently in use by this wizard:

- C:\WebSphere\AppServer\updates\installer\maintenance\IBM-WS-WAS-Wi
  nk12-choup0032996409.ear

In order to continue, it is needed to copy the current JDK to a new location and relaunch the wizard using the copied JDK. The current JDK will be copied from the source location to the target location:

- Source: C:\WebSphere\AppServer\jre
- Target: C:\WebSphere\AppServer\jre\sunjreinstall

After the copy has completed, this wizard will be relaunched and the update can proceed.

Click Next to begin the copy.
Click “Relaunch”.

Select the maintenance operation.
- Install maintenance package
- Uninstall maintenance package.
This could take a while, so wait till it completes.
Click “Finish” to exit.

Repeat the above procedure to install appropriate fixpacks recommended for your environment.

### 4.3 Installing Database Client

For most WAS nodes in a production environment, there will be remote JDBC connectivity to a centralized RDBMS datastore. You should include the database JDBC client configuration as part of basic WAS server image. In this case, we have selected to install the DB2 JDBC client.

Download the client or run from CD.

Click on “FP12_WR21368_RTCL.exe”
Click on “unzip”.

Go to c:\tmp\RTCL (RTC - Run Time Client)

Run setup.exe
Select "typical" install
Select the directory to install the DB2 RTC.

Accept default values above.
The actual configuration (DB connect string and username/password) should be configured after the installation of the “Run Time Client” and should be included as part of the base image of WAS that is being built.
5.0 Deployment of IBM WebSphere Application Server Virtual Machines (WASVM)

This section discusses how the WASVM Base Image can be used for deployment in Development, Test or Production environments.

We have created a base image consisting of the OS, WebSphere Application Server (WAS) environment. We will now create a template out of this virtual machine image. This template will then be used to generate (by cloning) WAS nodes which are customized with unique IP, SID, hostname and WAS application instances. This process of generating templates (or library of template images) and using them to clone multiple WAS nodes provides for rapid deployment in any WebSphere environment.

5.1 Templates

A template is a golden image of a virtual machine that can be used as a master copy to create and provision new virtual machines. This image typically includes a specified operating system and configuration that provides virtual counterparts to hardware components. Typically, a template includes an installed guest operating system and a set of applications. You can install all standard components into a VM, convert it into a template and use that image for rapid deployment across your virtual infrastructure.

Templates coexist with virtual machines at any level within the template and virtual machine domain. You can order collections of virtual machines and templates into arbitrary folders and apply a variety of permissions to both virtual machines and templates. Virtual machines can be transformed into templates without requiring a full copy of the virtual machine files and the creation of a new object.

You can use templates to create new virtual machines by cloning the template as a virtual machine. When complete, the deployed virtual machine is added to the inventory panel datacenter where the host resides. Templates are created from existing virtual machines. After a template is created, it can be deployed only to managed hosts that have access to the datastore where the template resides. On ESX Server hosts, the datastores are the VMFS, local, or NAS volumes that you configured for your ESX Server.

If a managed host with templates stored in it is removed from VirtualCenter, all the templates are also removed from the inventory. Returning the managed host to VirtualCenter does not re-register the templates because they are invisible to the host. When a host is added back to VirtualCenter, you can register the templates, making them visible.

5.2 Creating Templates

The WASVM1 virtual machine created in the previous section is the base image for our WASVM deployments and is now ready to be cloned to a template.

Use the following steps to create a template out of the WASVM1 virtual machine.

1. Shutdown the WASVM1 virtual machine.

2. From VirtualCenter, right click on WASVM1, select “shutdown guest OS”.
Right click on WASVM1, select “clone to template”.

(You can select “clone to template” or “convert to template”).

Clone to template → would keep the current VM intact and create a new VM template.

Provide the name of the template (WASVM1_Template) and also select the data center location where you want it stored (in this case: WASVMDC)

Select the “host” on which to store the template.
Provide a datastore. We had created an additional datastore by adding a second disk to our storage. See Appendix B for steps on adding storage to an ESX Server host.
Select “Normal”.

Click “Finish” to create the template. The task bar below will show you progress.

We have successfully created the template for WebSphere Application Server Virtual Machines which now will be used as a generic Windows Server 2003 SP1 / WebSphere 6.0 Image.

We will use this template to create our WAS application nodes (VMs).
5.3 Provisioning WebSphere Application Server (WAS) Virtual Machines Node from Templates

The WASVM1 image and template that were created are generic images. So while using “cloning” to provision instances of WAS nodes in a WebSphere environment, there are two distinct things that would occur:

1. WASVM Clone Creation
2. WASVM Clone Customization

Customization of the clone includes customization of the guest OS environment to make the OS image unique with unique SSID, licenses, network information etc. The application customization to create WAS instances on top of the operating system will be performed post cloning (either as a custom script or a set of applicable specific configuration – Ex: creating the WAS profile).

5.3.1 Preparing for Guest Operating System Customization

When you deploy a new virtual machine from a template or clone an existing virtual machine, you have the opportunity to customize the new guest operating system. The Guest Customization wizard guides you through the configuration options. Guest Optimization wizard will be invoked as part of the VM cloning process.

Before you run the Guest Customization wizard, if you intend to perform a guest customization, do the following:

Verify that your system meets the guest customization requirements.
Install the required components on the windows machine where VirtualCenter is installed.

Note: Our discussion in this section is focused on Windows Server 2003 SP1 used as the guest OS in our WASVM lab deployment. Refer to the VMware documentation for other guest OS requirements.

The windows guest OS environment is customized using “Sysprep” utility tools. This can be obtained either from the Microsoft website or from the Windows OS CD.

Pls. make sure to create the following directory structure on the VirtualCenter machine where you will be creating clones and templates.

Locate where VirtualCenter 2 was installed on the system.
Example: c:\ C:\Program Files\VMware\VMware VirtualCenter 2\
Under this directory, go to “resources” folder. Create (if it does not exist already) a folder called “windows”.
Under windows create folder “sysprep” and under “sysprep” create folder “svr2003.
The final destination for the “sysprep” tools is: C:\program files\VMware\VMware VirtualCenter 2\resources\windows\sysprep\svr2003.

If you are running XP or windows 2000, then the corresponding directory would be
…\resources\windows\sysprep\2k\ ...esources\windows\sysprep\xp\

Go to the OS CD.

1. Locate the DEPLOY.CAB file in the directory \support\tools
2. Open and expand the DEPLOY.CAB file using a tool such as winzip.exe or another tool capable of reading Microsoft CAB files.
3. Extract the files to the directory appropriate to your guest operating system.

EX: C:\program files\VMware\VMware VirtualCenter 2\resources\windows\sysprep\svr2003

After you have extracted the files from DEPLOY.CAB, you should see:
4. Repeat this procedure to extract sysprep files for each of the Windows guest operating systems (Windows 2000/XP/2003) you plan to customize using VirtualCenter.

We are now ready to customize a new virtual machine during the cloning the process.

Start the “cloning” process using the template (WASVM1_Template).

5.3.2 Cloning a New WAS Virtual Machine from Template

Right click on WASVM1_Template and select “deploy virtual machine from this template”.

![Deployment Wizard Screenshot](image-url)
Enter the name of the new VM: WASVMND1, location: WASMDC (datacenter under “Inventory”)

Select ESX Server host on which the VM will be created.

Select where the VM will be stored. (“Storage1). Remember, we are storing our templates on WASStore.
If you have completed your “preparation steps” for windows OS on the VirtualCenter system successfully, you should see the screen “select guest customization option”. If there are problems in your “sysprep” setup, you will get “do not customize” as the only option.

Click on “customize using the customization wizard”.

Note that once you customize, you have the option of saving these customization options as a specification that can repeatedly used in the future during new VM cloning (this is the 3rd option in the screen above).

Enter owner name and organization.
Specify a hostname for the computer or use the VM name (we selected to use VM name).

Leave the license key information blank. We can provide the keys during the system boot after cloning.
Enter the password for the Windows Administrator account.

Enter scripts and commands you would like to run when the system comes up and user logs in first time.
If you want to use DHCP to automatically assign network information, select “typical” settings. We picked customer settings to provide static IP address.

Click “customize” and select “Use following IP address”
Click on the drop box next to IP address.

Prompt the user for the address during the cloning process. This option will help you provide static IP to your WAS virtual machines during the cloning process.
Solutions Deployment Guide

Static IP Address Selection

Choose a method for selecting a static IP address

- Use a fixed IP address:
- Prompt the user for an address when the specification is used
- Use an application configured on the VirtualCenter server to generate the IP address

Argument:

[Buttons: OK, Cancel]

Fill in the rest of the information.

Network Properties

General DNS WINS

IP Address

- Use DHCP to obtain an IP address automatically
- Use the following IP address:

<table>
<thead>
<tr>
<th>IP Address:</th>
<th>Default gateway:</th>
<th>Alternate gateway:</th>
</tr>
</thead>
</table>

Subnet mask: 255.255.255.128

DNS Server

- Use DHCP to obtain DNS address automatically
- Use the following DNS server addresses:

<table>
<thead>
<tr>
<th>Preferred DNS server:</th>
<th>Alternate DNS server:</th>
</tr>
</thead>
<tbody>
<tr>
<td>172.26.1.1</td>
<td>172.26.1.2</td>
</tr>
</tbody>
</table>

[Buttons: OK, Cancel]

Click Ok to return to the previous screen.
Click "next"

Provide your domain name. We used "workgroup" in the lab.

Check "Generate New Security ID" box. This will generate new and unique SID for the WASVM windows environment.
Save all the customization parameters as a custom specification (WASVMND). Customization specifications help set up standard (parameters) information for guest OS and application customizations.
Click “Finish”. A customization specification by name “WASVMND Specification” will be created as part of the cloning process and can be used for future cloning from the template.

Now prior to the start of the new VM cloning process, you are prompted for the IP address of this VM.

Provide your IP address (note that the computer name field is not enabled as we have already provided the “VM name” as the host name)

Click “Finish” to start the cloning process.

The “recent tasks” bar will show you the status.

Once completed, right click the new WASVMND1 and “power on”
Enter license keys when prompted.

The system should configure and reboot again to come to the login prompt. You should now be able to remote desktop into the system.

Login and check network settings. This should be the IP address and network information you provided during the cloning process. Your computer name should be set to the “VM name”.

The final customization steps occur when the new virtual machine boots for the first time. After powering on for the first time, a customized Windows virtual machine automatically reboots twice to finalize the configuration process. It becomes operational when the logon page appears after the second reboot. This process can take several minutes, depending on the speed and load of the host. If any errors occur during the final configuration process, events are logged to the guest operating system’s event database. To view these errors, choose Start > Program > Administrative Tools > Event Viewer from the Windows Start menu.

If any of the information required in the configuration finalization process is not correct, the guest operating system pauses when the new virtual machine boots and waits for you to enter the correct information.

Incorrect information might include computer name, product key, domain name, network (DHCP) settings etc.
5.3.3 Provisioning WAS Nodes Using Customization Specification

We can use the customization specification created in the previous cloning steps to create additional WAS nodes.

Click on the WASVM template (WASVM1_Template), select “deploy a new virtual machine from this template”. We will deploy a second WAS node (WASVMND2) using the custom specifications.

Just follow the steps as outlined in the earlier section.

When you get to “Guest Customization Option”, select “Customize using an existing specification” (remember, in the last cloning we selected “customize using customization wizard”).
Select the specification (WASVMND was the specification we created in the last cloning process) to be used.

Check the “Use Customization Wizard to adjust specification …” box below to make adjustments to the specification prior to deployment, if you want to change some of the specification parameter values. We took the default (unchecked).

This will take you directly to the cloning process.

Supply the IP address of the clone.
Click “finish” to start the cloning process.

Once the cloning is completed, you see the following

Now “power on” WASVMND2.

The system should come up and ask for the licensing keys. Once the keys are entered the system should configure itself and come up to login prompt. Login to verify that the configurations worked correctly and made the 2nd WAS clone unique.

Now we are ready to customize the WebSphere Application Environment on these VM nodes (WASVMND1 and WASVMND2) to create a running WAS application environment.
6.0 WASVM Application Customization

In this section, we will go through the steps to configure the two VM nodes we have created into a functional WAS environment. The following steps will be performed as part of the customization.

1. A WebSphere Network Deployment Cell will be created
   a. WASVMND1 will be configured as the Deployment Manager Node
   b. WASVMND2 will be configured as the Application Server Node
2. Install HTTP Server and Web Plugin Components for load balancing
3. Create a WAS Cluster

At the end of this process, you will have a fully functional WebSphere Application Server Cluster.

6.1 Creating Deployment Manager Profile on WASVM

Login to WASVMND1 and go to start → IBM WebSphere → WAS Network Deployment 6.0 → Profile Creation Wizard

Click “next”
Select “create deployment manager profile”.

There are two distinct profile components. The deployment manager profile is responsible for managing the WAS cells and the application server profile will create the application servers. The application servers are stand alone servers, i.e. they are not aware of the deployment manager or WAS cells or clusters. We will, later in this section, add these application servers to the WAS cell(s) managed by the deployment manager, and then create a load balanced cluster.

Take the default.
Solutions Deployment Guide

VMware and IBM WebSphere Software

Take the default directory location.

Make sure the hostname is the correct hostname for your virtual machine. The rest can be customized.
Take the default ports

You can choose each of these settings just like you would on a physical installation. We have chosen to run the application server as a windows service with administrator as the user. Startup can be manual (default) or automatic.
Click “Next”.

Solutions Deployment Guide
Launch the first steps console.

Run the installation verification to confirm installation.
The last few lines in the window above will show that the installation is successful and the deployment manager will be started.

Now you can bring up the administrative console.
Just type any user name (we used “Raj”).

This shows that the deployment manager is running successfully.

Now we will install an application server on the 2nd WASVM node (WASVMND2).
6.2 Creating Application Server profile on WASVM

Login to WASVMND2 and go to Start → IBM WebSphere → Application Server Network Deployment V6 → Profile Creation Wizard

Click on “Create an Application Server profile”

Provide application server profile name or select the default.

Enter profile directory
You can provide the node name (we took default). Make sure the hostname is correct.

Take the default values.
Make sure to provide the administrator password.

Launch the first steps console

Perform the installation verification.
The application server is started after verification.

On the command screen, go to the bin directory as shown below
Run command "add node <<IP address of Deployment Mgr node>>"
In the lab, it was - add node 172.26.38.90

If you want to include default applications,
# add node <IP of deployment Mgr> -includeapps

We included the default applications for our setup validation.
So far, we have successfully created a deployment manager managing a cell and added a stand alone application server instance as a node to the cell. You can use the steps above to add multiple nodes of VM based WAS instances to the cell.

To validate that the above cell of deployment manager and application server were configured successfully, on the VM running the deployment manager, bring up the administrative console for WAS deployment manager.

Login into WASVMND1 and bring up the first steps console.

Click on “administrative console” and login (user: Raj)
Click on "servers" → "application servers"

From this point on, your WAS custom configurations can be performed very similar to any WAS physical environment.
Select the server and click "start:"
We validated our setup using a sample application deployed with our installation on the WAS virtual machines.

Now go to the web browser and go to http://<<application server ip>:9080/snoop

You can also invoke another servlet http://<< application server ip>>:9080/hitcount
At this point we have completed verification that the application server is running by execution of sample servlets and java code.

6.3 Installing Load Balancing in IBM WebSphere Application Server (WAS) Virtual Machine Environment

(The steps below are environment, or application, specific and are presented here for illustration purposes only. Refer to WebSphere documentation for customer specific configuration).

Note that load balancing component needs to be installed post cloning or as part of the customization process.

Note: In an enterprise WebSphere environment the HTTP server and load balancing is configured as a separate system (or VM) and not on the deployment manager.

On the system or virtual machine to host the HTTP server, navigate to the WAS software distribution, click "launchpad.bat" under "code".
Click launch installation wizard for IBM HTTP server.
Provide the directory name.
Enter the user password.

Select “launch the WAS – Plugin Install”.
Select "HTTP Server V6"
Point to the httpd.conf file where we just installed your HTTP server.
Webserver name: take the default
For the Web server plugin-cfg.xml file: point the above path to the deployment manager profile path – c:\WebSphere\AppServer\profiles\Dmgr01\config\cells\plugin-cfg.xml for our lab setup. This is the path that the necessary plugin-cfg.xml file is generated for WebSphere’s HTTP loadbalancer.

We need to specify this path so that the plug-in will be able to find this file, read it, and generate the rules for load balancing. This file will be created by the deployment manager after your application is installed, configured, and deployed on a cluster.
We will now configure a WAS cluster environment, a sample application and then enable load balancing across the WAS nodes.

6.4.1 Creating a WAS Cluster

Go to admin console of deployment manager and click on “cluster”.
Click "New"
Provide the cluster name (WASCluster) and select existing servers to be part of the cluster.

Click “next”
Click “next”.
If you want to add more members (application server nodes), use the above screen.

Click "next".

Click "Finish"
Make sure to click “save” to apply changes to master configuration.

Click “save” to synchronize changes with the WAS Nodes.
At this point, we are ready to install the application(s) on the cluster.
6.4.2 Installing Application on WAS Cluster

In the lab, for illustration purposes, we used a sample application called HelloUser called within an Servlet application (VMware.war). This application is a simple servlet that generates an input field in an html, and returns the field in a dialog box. The following are the steps to deploy the application war file (VMware.war is the name we used) and configure it on the WASCluster object configured in the previous steps.

Go to admin console for deployment manager,
Click application → install new applications

Specify path name for the application
Solutions Deployment Guide

VMware and IBM WebSphere Software

Take defaults

[Image of a web interface showing the installation of an application.

The interface includes a tree menu on the left with options such as "Welcome", "Guided Activities", "Sensors", "Applications", "Security", "Environment", "System Administration", "Monitoring and Tuning", "Troubleshooting", and "Service Integration".

The main area of the interface shows the installation process with steps and options.

- Step 1: Select installation options
  - Step 2: Map modules to servers
  - Step 3: Map virtual hosts for Web modules
  - Step 4: Summary

Options include:
- Pre-compilation
- Directory to install application
- Distribution application
- Use binary configuration
- Display enterprise beans
- Application name
- VMware xar
- Create MBeans for resources
- Enable class reloading
- Restart interval in seconds
- Display Web services
- Validate input file/namespace

The interface also includes a note:
"NOTE: Syntax errors in the policy files will cause the enterprise application to fail to start. It is advised to use the policies provided by the IBM for setting the policy files. It is also advised to use the policies provided by the IBM for setting the policy files."

The interface also includes a message box with the following text:
"The contents of the vcap policy file are:
H \ Template policy file for enterprise application. Permissions can be added if required by the enterprise application. H // NOTE: Errors in the policy files will cause the enterprise application to fail to start. It is advised to use the policies provided by the IBM for setting the policy files."

The message box contains a list of permissions and settings.

Overall, the interface provides a guided installation process for an enterprise application within the WebSphere environment, offering a step-by-step guide to configuring and deploying the application.
In the above screen, select your application(s) and also select the clusters and click “apply”.
Solutions Deployment Guide

Click ‘finish’.

Make sure to click “save to master configuration”
Make sure to check “synchronize changes with Nodes”, then click “save”.

Now go back to the admin console screen and go to applications  →  enterprise applications.

Select (check box) application name (VMware.war) and start the application.
Now you are ready to run the application.
Go to any browser and type http://<application server IP>:9080/<Enterprise Application Name>/<Component APPNAME>

In the lab, it was http://172.26.38.91:9080/VMware/HelloUser

This completes the validation of our clustered application server environment. You can run any sample application or your own custom application in your WAS environment to validate the functionality of the WAS cluster.
6.4.3 Configuring WAS Cluster Load Balancing

These steps would be performed on your deployment manager VM (WASVMND1 in our lab setup).

1. We will manually generate the plugin-cfg.xml file, and start the load balancer. The plugin-cfg.xml file will need to be regenerated whenever new servers are added to the cluster (WASCluster) or whenever an application is added.

Run the GenPlugInCfg.bat file in $WAS_HOME\bin on deployment manager. This will generate the file in the deployment manager profile home under config\cells. This should be the same directory path provided during the load balancing (HTTP Server and Web-Plugin) Installation.

```
C:\WebSphere\AppServer\bin>GenPlugInCfg
IBM WebSphere Application Server, Release 6.0
WebSphere Plugin Configuration Generator
Copyright IBM Corp., 1997-2001
PLGC0012I: The plug-in is generating a server plug-in configuration file using the cluster definition WASVMNDICell101.
PLGC0055I: Plugin configuration file = C:\WebSphere\AppServer\profiles\Dmgr01\config\cells\plugin-cfg.xml
```

To verify the correct path, open the httpd.conf file in the IBM HTTP Server “conf” directory from the WAS HTTP loadbalancer system or VM. At the end of this .conf file you should see a path similar to the following, C:\WebSphere\AppServer\profiles\Dmgr01\config\cells\plugin-cfg.xml.

If the WAS HTTP loadbalancer system or VM is on another system or VM as mentioned before, your path may look like the following, C:\IBM\WebSphere\Plugins. If this is the case, then you will need to generate the plugin file as above, then copy it, either through FTP or SCP or etc…, to this system and into the directory found in the httpd.conf file, e.g. C:\IBM\WebSphere\Plugins

2. Start the load balancer. Run the following command from the prompt in the IBM HTTP Server bin directory.

```
$apache –k start
```
3. To verify that the webserver is up,

On a browser type: http://<< HTTP server IP>>/

4. To verify that you can access the application through the load balancing web server,


We now have a load balanced WAS cluster environment running on Virtual Machines. We can add more WASVM nodes to the cluster while running on the same physical system (ESX Server host).
7.0 Upgrading IBM WebSphere Application Server Environments on Virtual Machines (WASVM)

There are 2 distinct VMware Infrastructure 3 features that can simplify the upgrade (or patching) of WebSphere Application Server environments.

1. Snapshots: Create a snapshot prior to upgrade. If the upgrade or patching fails, you can revert back to snapshot instantly. This provides for quick problem resolution in case of challenges during patching or upgrades.

2. VM cloning: Clone the WASVM node that needs to be upgraded and make sure to configure it on another virtual NIC network. Make sure this network is private or on a different network from the production environment. You can create the cloned VM on the same ESX Server host or on another ESX Server host and configure it to be on a separate network while retaining all the configurations deployed in production. This will ensure that the cloned VM and the original VM are not talking to each other. Now you can perform the upgrade in isolation and test it. While you are performing the upgrade, the production VMs are not impacted and continue to run. Once the upgrade is successful and tested, you can just enable this cloned VM to be the production WAS node and shutdown or delete the original VM.

We will look at the steps involved in using “Snapshots” or “VM cloning” to successfully upgrade our WAS cluster environment.

7.1 Snapshots in WASVM environment

A snapshot captures the entire state of the virtual machine at the time you take the snapshot. This includes the memory state, disk state and the VM settings. When you revert to a snapshot, you return all these items to the state they were in at the time you took that snapshot.

Snapshots are useful when you need to revert repeatedly to the same state but you don’t want to create multiple virtual machines. With snapshots, you create backup and restore positions in a linear process. Snapshots can be used as restoration points during a linear or iterative process, such as installing update packages, or during a branching process, such as installing and testing different versions of a program or patches. You can take a snapshot while a virtual machine is powered on, powered off, or suspended.

Snapshots can bring unique value to a WebSphere environment. You can use snapshot for the following:

1. Create point in time copies for recovery from problems.
2. Use snapshots during upgrades and patching. If there are problems during change management processes, you can instantly revert back to the original image.
3. Snapshots can be useful during application failures to revert back to stable image of the application environment.

7.1.1 Taking a WASVM Snapshot

Go to VirtualCenter, click on “WASVMND1” virtual machine.
Right click on WASVMND1 → Snapshot → “take snapshot”.

Enter the name for the snapshot that makes logical sense so that you can roll back to this WAS system state when needed.

The “recent tasks” window shows the status of the snapshot.
Right click on WASVMND1 ➔ Snapshot ➔ Snapshot Manager.

This shows you the snapshots available for the VM.

We can use the snapshot feature during the WAS upgrade process where the snapshots can help to rollback to the earlier stable WAS image in case of patch or upgrade problems.

**Note:** It is recommended to take a snapshot before any major upgrade or patching when executed directly on the production WAS virtual machine.
Create a second virtual network switch (private – not exposed to the public) on the ESX Server host. Do not associate this virtual switch with any physical NIC. This will ensure that the VM created on this virtual switch can only communicate with other VMs on the same switch and cannot talk to the public network or other VMs on the virtual switch (VM network) already created and associated with the physical Ethernet interface.

### 7.2.1 Creating a New Virtual Switch

Login to VI client and select “host”

Click on “configuration” tab on the right side and go to “networking”
Click on “add networking” on the right hand corner (top)

Select “virtual machine”
Check “create a virtual switch” but make sure to uncheck vmnic1. Remember, we are creating a virtual switch without associating with any physical network interface. You can always associate with a physical NIC as long as the NIC and virtual switch or configured to be a separate network and will not co-exist with the original VM on the production network. Also, it is important to note that we are creating an exact mirror image of the production WAS VM so all the network settings will remain the same.

Take the default name VM Network 2.
Click “finish”

This will create the new virtual switch and virtual interface that can be used for our upgrade VM that will be cloned from production VM (WASVMND1). (WASVMclone1 that you see in screen above was created as a test clone using this 2nd virtual network).
7.2.2 Creating a Clone for Upgrade

Shutdown your WASVM node (we used WASVMND1).

On the VI client, right click on WASVMND1 → Clone

Provide a new VM name (WASVMND1clone).
If you want to store the clone in a separate inventory location (different datacenter location), you can provide that above.

If you wanted to create this clone on a different ESX Server host, you can provide that here. In the lab, we created it on the same ESX Server host.
Select datastore (storage1).

Select “do not customize”. Remember, we want to keep the clone an exact mirror of our production (original) VM – WASVMND1. This will help us to make sure that all the OS and application settings are retained during upgrade and help quickly re-provision this system as the new production VM (post upgrade).
Click “finish”

You can follow the progress in the “recent tasks” window.

Once the cloning is completed,

Edit the network settings for the new clone.

Right click on WASVMND1Clone → Edit Settings

Highlight network adapters,
Click “add” to add a new network interface.

Select “Ethernet adapter”
Select “named network with specific label” and select VM Network 2 that we had created recently.

Click “finish”.

Go to network adapter 1 (VM Network) and click “remove”.
Click “ok”. This will delete the VM network for the new clone while making VM Network 2 the new virtual switch network for WASVMND1Clone.

If you go back to edit settings, you should see the following screen.

Now this is in an isolated network that we created within the host and you can retain all attributes (including hostname, IP etc.) and start the upgrade.

Now “power on” the cloned VM.
7.3 Upgrading WAS 6.0 to WAS 6.1

In this section, we will illustrate a migration path for the VM node running deployment manager. The deployment manager may be running, but it is a good idea to stop the server. The load balancer and the application servers in the cell may still be available and running during this migration. To begin the process, we will need to install the v6.1 code base on the deployment manager node.

Go to your download (or CD image) of the WAS 6.1 upgrade files.

Click “install.exe”
Give the path name for the WAS6.1 deployment. This should be different from the current WAS 6.0 installation directory.
Click “yes”.

Uncheck the “create a new WAS profile” and click “Finish”.

Go the windows “Start” menu.

Start → All Programs → IBM WebSphere → Application Server Network Deployment 6.1 → Migration Wizard
It should identify the existing versions of the product from the list. Select the product and click “next”.
Select the profile you want to migrate (in this case Dmgr01 is the only one we used)

Take the default above.
Specify a new name or use the existing profile name (Dmgr01)

Provide the backup directory for the existing configuration.
Take the default
Disable the previous version of the deployment manager (unchecked box)

Take the default
The migration process will start when you click “next”. This will start the migration process and the status window will display progress.
This completes the migration of WAS 6.0 to WAS 6.1. Now you can test the upgraded 6.1 clone to make sure the upgraded WAS instance works successfully prior to moving this instance to production.
7.4 Migrating WAS 6.1 VM Clone to Production

Shutdown the production VM (WASVMND1) and make the clone (WASVMND1Clone) as the new production system. Since all the attributes (including IP address) are retained, you can move this clone directly into production.

Go to VirtualCenter Console ➔ right click on “WASVMND1Clone” ➔ edit settings ➔ network adapters.

Change VM Network 2 back to VM Network. Make sure you have taken the original WASVMND1 (WS 6.0 VM) offline before doing this.

Now your new WAS 6.1 VM is online.

Currently we have a configuration of two virtual machines running the WAS environment:

WASVMND1Clone: Deployment Manager (WAS 6.1) and load balancer
WASVMND2: Application Server (WAS 6.0).

Upgrade the other WAS node(s) in the same manner as shown above and refer to the WebSphere documentation for details on specific application upgrade steps. If you are likely to have a mixed environment (6.0 and 6.1 instances), please ensure that the configuration is supported.

7.4.1 Verification of the Upgrade and WAS functionality:

Start the deployment manager from the application server network deployment V6.1 menu.

Now go to admin console [http://localhost:9060](http://localhost:9060)
You should be able to see the new version number 6.1.0.0 (earlier it was 6.0.2)

Go to servers ➔ application servers.
Here, you will see that the server1 version is still 6.0.2.0 (this is the application server WAS node we had created called WASVMND2).

verify that your sample application (HelloUser) still works.

http://<< deploy Mgr IP >>/VMware/HelloUser (or whatever your test application is).
The WASVM clone (WASVMND1Clone) has been successfully upgraded.

For more information on WebSphere Upgrade, go to
8.0 Converting Physical IBM WebSphere Application Server Instance to an IBM WebSphere Application Server Virtual Machine

VMware P2V Assistant is an enterprise-class migration tool that transforms an image of an existing physical system into a VMware virtual machine.

VMware P2V Assistant allows you to:

- Non-intrusively copy and transform physical systems into VMware virtual machines
- Migrate legacy servers to new hardware with no need to reinstall operating systems or application software
- Perform migrations across heterogeneous hardware
- Proactively readjust disk sizes, types and partitions to maximize utilization of storage resources

P2V will help migrate existing WAS instances on physical servers to WASVM instances running on the VMware Infrastructure.

For the purpose of demonstrating P2V, we created a WAS 6.1 server instance on a physical system running on Windows Server 2003 SP1 and added it to the WAS cluster.

Before installing the P2V tool, check the P2V Platform compatibility guide documentation to ensure that your hardware and operating system are supported for P2V.

The basic P2V Assistant processes are:

Cloning — The process of creating a cloned disk, where the cloned disk is a virtual disk that is an exact copy of the source physical disk. This involves copying the data on a physical source machine’s hard disk and transferring that data to a target virtual disk (the new cloned disk). This process can be performed using the P2V Assistant installed on a helper machine (physical or virtual) or using a third-party disk imaging or backup/restore tool running in a helper virtual machine. The helper machine is an intermediary machine that allows the P2V Assistant to interact with the target machine’s virtual disks.

System Reconfiguration — The process of adjusting the migrated operating system to enable it to function on virtual hardware. This adjustment is performed on the target virtual disk after the cloning and enables the target virtual disk to function as a bootable system disk in a virtual machine. System reconfiguration is not required if the target virtual disk will not be used as a bootable system disk. This process is performed using the P2V Assistant installed on a helper machine (physical or virtual). A system reconfiguration does not change the identity of the new virtual machine.

Creating the Virtual Machine — Using the new virtual disks in a new virtual machine. If the new virtual machine disk had an operating system and you performed a system reconfiguration on it, the new virtual machine retains the identity and settings of the original physical machine. These include: machine name, SID, hard disks (partitions), data files, application and user settings. This process does not require the P2V Assistant or a helper machine (physical or virtual). It does require that the cloned disk be detached from the helper virtual machine, if that was used.
If you plan to run the new virtual machine on the same network as the original source machine, you need to modify the identity (name, IP address, and networking) of the virtual machine so the machines can co-exist on the same network.

Note: While the P2V Assistant simplifies the physical-to-virtual migration process by automating several steps, it is not intended to serve as an unattended migration tool for casual use. Migration is complex — and whether using the P2V Assistant or not, only technical system administrators with a good understanding of and familiarity with operating systems, drivers, imaging tools, and VMware products should attempt the physical-to-virtual migration process.

Fundamentally, you need the following components to perform P2V:

1. The source system – physical system with the OS and WAS applications configured
2. A Virtual Machine (also known as helper VM) on the current ESX Server host running the same OS as the source system (recommended). P2V assistant will be installed on this helper virtual machine.
3. A new Virtual Machine that will be the final destination for the post P2V virtual environment.

8.1 Installing VMware P2V Assistant

Create a new VM with Windows 2003 SP1 loaded on it. This VM is called the helper VM and P2V assistant will be installed on this VM. (In our lab, we decided to use our base image VM (WASVM1) for this purpose)

Download the P2V assistant software executable. Start the installation.
VMware and IBM WebSphere Software

Solutions Deployment Guide

END USER LICENSE AGREEMENT
FOR VMWARE® P2V ASSISTANT SOFTWARE PRODUCT

VMWARE, INC. LICENSES THIS P2V ASSISTANT SOFTWARE PRODUCT TO YOU SUBJECT TO THE TERMS CONTAINED IN THIS END USER LICENSE AGREEMENT ("EULA"). READ THE TERMS OF THIS EULA CAREFULLY. BY INSTALLING, COPYING OR OTHERWISE USING THE SOFTWARE (AS DEFINED BELOW), YOU AGREE TO BE BOUND BY THE TERMS OF THIS EULA.

1. Accept the terms in the license agreement:
2. I do not accept the terms in the license agreement:

Install this application for:

- Anyone who uses this computer (all users)
- Only for me (WASWin)

Customer Information
Please enter your information.

User Name:
Organization:

Install Wizard
< Back Next > Cancel

Install Wizard
< Back Next > Cancel
Click "install" to begin installation process.
We will now create a VM which will eventually host the migrated virtual WAS environment (virtual disk)

8.2 Creating a Target Virtual Machine (VM)

Go to VirtualCenter console and right click on the ESX Server host.

Click on “new Virtual Machines”.

We will create a new virtual machine named (P2VVM) to host our WAS instance (post P2V).
Size of our physical (source) system disk was 68GB out of which it uses 14GB.

Note that P2V cloning copies only the files (actual data) and not the entire disk. So we will be creating a 20GB virtual disk to accommodate our physical image. It is a recommended best practice to keep the size of the target disk(s) the same as the source disk(s).

Select the right guest operating system and version for the virtual machine. This needs to match with the source system OS and version.
Select the number of virtual processors for the VM.
Shutdown the target VM (P2VVM) you just created. Note down the path of the blank virtual disk created during P2VVM creation.
8.3 Adding Target Virtual Disk to Helper VM

Add it to the helper VM (in our case, WASVM1) as a second virtual disk as shown below.

Shutdown helper VM (WASVM1).

Right click on WASVM1 → edit settings → Hard Disk1

Click “Add” to add a new virtual disk to the VM.

Select “hard disk”
Select "use an existing virtual disk" and browse to the datastore directory where the P2VVM disk is located.
Select "independent" mode with persistence.
Click ok.

*Power on* the helper VM (WASVM1).
8.4 Source System Preparation

Shutdown the physical system hosting the WAS server and reboot the system using the P2V Assistant 2.1.2 boot CD.

You should see the boot prompt. Press “enter”

Press “ok” on the welcome screen.

Select the source disk(s) involved and the network interface used to access the system.

If the disks are connected by Fibre Channel, select “yes” when prompted for loading FC drivers.

You have the option of providing DHCP or Static IP.

Provide the required network information.
Ex: our IP was 172.26.38.89 and TCP Port: 7000

Press OK.

You should see a summary of the network settings. Leave the source system at this state.

8.5 Cloning Source Disk to Virtual Disk

Start the P2V assistant on the helper VM.

Note: if you have not added licenses for P2V, you will be prompted at this point.
Give the IP address you just configured on the source machine

Select the hard disk that needs to be cloned (we were using only c:\).
We will perform “reconfiguration” later. So, just select “clone disk without modifications”
We want to use the 2nd virtual disk of 20GB we attached to the helper VM as our target disk. So select “use a direct disk device”.

Then click “select” to choose the disk drive.
Make sure you select the right disk (disk number 1 of 20GB capacity)

Click "No".

Note: it is recommended to create a virtual disk of the same size as the physical disk that is being cloned even though only the data is copied over during P2V clone process.
The P2V Assistant wizard has collected all the necessary information. Click Next to commit the operation.

Operation in progress

Formatting volume 1

Operation in progress

Copying files from volume 1 [FileSystem=NTFS]
MB Copied=49 Remaining=976 Rate=5104 KB/sec
Time Elapsed=00:00:10 Remaining=00:33:22

Operation in progress

Copying files from volume 1 [FileSystem=NTFS]
MB Copied=9861 Remaining=145 Rate=4578 KB/sec
Time Elapsed=00:33:45 Remaining=00:00:32
8.6 Reconfiguration of Cloned Virtual Disk

Use the VMware P2V Assistant to reconfigure a cloned disk so it can be used as a bootable system disk in a virtual machine.

Start P2V Assistant.
Select “perform a system reconfiguration of existing virtual disk”

Select the direct disk device that has the cloned physical image.

Select “ESX 2.1.x, 2.5.x or 3.x (2VCPU)”. Also select the system reconfiguration options.
Click "next". This will update the registry and reconfigure the operating system on the virtual disk.
You have completed reconfiguring a system on the cloned virtual disk using VMware P2V Assistant

8.7 Booting Target VM with Cloned Virtual Disk

Shutdown the helper virtual machine from the VI client.

Right click on the helper VM (WASVM1) → edit settings → hard disk 2

Click “remove”

Make sure to select “remove disk from virtual machine”
Go to the target virtual machine (P2VVM) that was already created earlier in the P2V exercise.

“Edit settings” to confirm that hard disk 1 is pointing to the right disk file.

“Power On” the target (P2VVM) virtual machine.

Note: make sure your physical system that was cloned is not on the network, else you will have a network address clash. You can always change the IP address of the physical (source) system later if you want to bring it up again and re-provision it for other purposes.
Note: When the new VM was booted, we received an error (screen shown above). This error is consistent with a known issue while P2V windows running on IBM hardware (x336, x346, HS20s etc.). This problem was specifically due to IBM OSA IPMI device driver. There are two ways to get rid of this problem.

1. Uninstall from the source system (add/remove programs) prior to the P2V cloning process
2. Mount the target virtual disk (post cloning) on the helper VM,
   a. Go to the windows\system32 directory on the target disk
   a. Rename the IPMI.sys to IPMI.old
   b. Remove the virtual disk from helper VM (as shown earlier in this section)
   c. Put the target virtual disk in the new VM and boot.

We followed the second option.

Make sure the virtual disk file is added as the main hard disk 1 for the target VM (P2VVM). “Power on” the target VM. The system should start up successfully. Before proceeding to login, make sure to install VMware tools.

Check the P2V documentation to confirm if any post boot up steps (cleanups, configuration etc.) are required for your specific OS version. If there are any services that fail to start during boot, check the event log viewer for details. Windows 2003 Server is quite successful in configuring all needed drivers on boot up and might request you to reboot upon startup.

Also ensure that your host name and IP address are configured correctly and is the same as the physical system. In our labs, we had used DHCP for the physical system but were using static IP for the VMware Infrastructure 3 environment. So we had to manually configure the IP address.

Now start your WAS application services and you should be part of the WAS cluster that the physical WAS instance was initially configured for.

At this point, we have successfully migrated a WAS instance from a physical server to a virtual machine. Repeat this procedure to migrate all your WAS components from physical servers to virtual machines running on VMware Infrastructure.
Appendix A:

VMware Migration for an IBM WebSphere Application Server Virtual Machine

Moving a virtual machine from one host to another is called migration. Migrating a powered-on virtual machine is called “Live migration with VMotion”. Migration with VMotion, designed to be used between compatible systems, allows you to migrate virtual machines with no downtime and zero disruption to end users but requires VMotion licensing and specific configuration.

You can perform the following types of migration of virtual machines (VM):

1. **Cold Migration**: VM Migration in powered-off or suspended state.
   a. If the VM files are on Shared storage (SAN, iSCSI or NAS), only configuration files are moved with no disk movement.
   b. Relocation: When VM files are not on SAN (Ex: local storage) and needs to be moved to another storage destination or ESX Server host.
2. **Live Migration with VMotion**: Moving a virtual machine that is powered on with online applications. The VM files should be located on shared storage and accessible by both the source and destination ESX Server hosts.

VMotion allows working processes to continue throughout the migration process. The entire state of the virtual machine as well as its configuration file, if necessary, are moved to the new host even while the data storage remains in the same location on the SAN. The associated virtual disk remains in the same location on the SAN storage that is shared between the two hosts. Once the configuration file is migrated to the alternate host, the virtual machine runs on the new host.

The state information includes the current memory content and all the information that defines and identifies the virtual machine. The memory content includes transaction data and whatever bits of the operating system and applications are in the memory. The defining and identification information stored in the state includes all the data that maps to the virtual machine hardware elements, such as BIOS, devices, CPU, MAC addresses for the Ethernet cards, chip set states, registers, and so forth.

Migration with VMotion takes place in 3 distinct stages:
1. When Migration with VMotion is requested, VirtualCenter verifies that the existing VM is in a stable state with its current host.
2. The VM state information (memory, registers, network etc.) is copied to the target host.
3. The VM resumes its activities on the new host.

If any error occurs during migration, the virtual machines revert to their original states and locations. Virtual machines can be moved between hosts within the same datacenter. Virtual machines cannot be moved between datacenters.

The VMkernel networking stack must be set up properly to accommodate VMotion.

The network services provided by the VMkernel (iSCSI, NFS, and VMotion) use a TCP/IP stack in the VMkernel. This TCP/IP stack is completely separate from the TCP/IP stack used in the service console. Each of these TCP/IP stacks accesses various networks by attaching to one or more port groups on one or more vSwitches.
VMotion Requirements

Shared Storage
Ensure that the managed hosts use shared storage. Shared storage is typically on a storage area network (SAN), but can also be implemented using iSCSI and NAS shared storage.

CPU Compatibility
Make sure that the source and destination hosts have a compatible set of processors. VMotion transfers the running architectural state of a virtual machine between underlying VMware ESX Server systems. VMotion compatibility requires that the processors of the target host be able to resume execution using the equivalent instructions that the processors of the source host were using when suspended. Processor clock speeds and cache sizes, and the number of processor cores may vary, but processors must come from the same vendor class (Intel or AMD) and same processor family (P3 or P4) to be compatible for migration with VMotion.

Networking Requirements
VMotion requires a Gigabit ethernet network to ensure rapid migration. A dedicated network is recommended to keep VM memory state secure. VMs must have access to the same subnets on the source and destination ESX Servers. VMotion automatically maps VMs to appropriate virtual NICs based on network labels.

Before migrating with VMotion, ensure the following on the source VM:
1. VM have no active connection to a local CD-ROM or floppy image.
2. VM have no access to (CD or floppy) image located in a local datastore.
3. VM is not mapped to a SAN LUN that cannot be seen by the destination ESX Server.
4. VM has no active connection to an internal virtual switch (it should be connected to the production network).

Migration benefits in a WAS environment
1. Relocate WAS instances (VMs) from one physical system to another without disrupting end user functions.
2. Migrate WAS instances from failing hardware without losing application functionality and configuration.
3. Relocate the WAS environments from one storage to another
4. Perform hardware maintenance without disrupting WAS applications running on the system.
5. Use Migration to support distributed resource management and VMware high availability solutions in a virtualized data center.

To showcase Migration, we installed ESX Server 3 on another physical (IBM X336) system.

Migration Setup
1. Add the new ESX Server host (172.26.38.95) to the WASVMDC data center on the virtual center.

Right click on WASVMDC and select “add host”
Specify Connection Settings

- Connection
  - Host name: 172.26.38.95

Authorization
- Username: root
- Password: 

Host Information

Name: 172.26.38.95
Username: root
Model: eServer iSeries 336 (9037-MAC)
Version: VMware ESX Server 3.0.2 Update 1
Solutions Deployment Guide

VMware and IBM WebSphere Software

Add Host Wizard

Virtual Machine Location
Select a location in the VirtualCenter inventory for the host’s virtual machines.

Connection Settings:
Host Summary
Virtual Machine Location
Ready to Complete

Add Host Wizard

Ready to Complete
Review the options you’ve selected and click Finish to add the host.

Connection Settings:
Host Summary
Virtual Machine Location
Ready to Complete

Please review this summary before finishing the wizard.

Host: 172.30.30.16
Version: VMware ESXi Server 3.5.6 (build-9770)
Networks: VMNetwork
Databases: Connected

Add Standalone Host
VMware 64% offline
admin
Make sure it is pointing to the license server.

Go to configuration → licensed features
Make sure it is pointing to the right license server. Click “edit” on license type to select appropriate licensing.

Now you are ready to use the 2nd ESX Server host as part of the Virtual Infrastructure.

2. **Configure the VMkernel** network configuration for VMotion.

Click on ESX host (172.26.38.95) → Configuration Tab → Networking

Click on “Add Networking”

Select “VMKernel”. Click “next”
Select “create a virtual switch” to define a new switch for VMKernel. If the lab, we selected “Use vSwitch0” as VMnic0 was the only network configured between the ESX Server hosts.

Note: It is always recommended to keep the VMKernel network on a dedicated Gigabit network connection between the ESX Server hosts.

Provide the IP information (this will be different from the ESX Server service console IP and will be used by VMKernel for VMotion). Make sure to select “use this port group for VMotion”.
Click “Finish”.

This will create the VMKernel network switch port.

Repeat the steps above to create the VMKernel configuration on the other ESX Server.
Ensure that the physical interfaces on which the vswitch for VMKernel sits are configured to see each other.

3. **Make sure the shared storage is added** and configured for both ESX Server hosts. (shown in Appendix B)

**Cold Migration**

Make sure the VM is powered off or suspended (we used WASVMND1 – our old WAS 6.0 image).

Right click on VM → Migrate

Select the destination host (the new host we just added 172.26.38.95)
Select “destination resource pool” (we did not create a separate resource pool).

Select to move all “configuration files and virtual disks” (This VM was on a local storage and is being moved to another local or shared storage on the destination host).
Click “Finish”.

View the “recent tasks” window for status.

You should see that the VM has successfully migrated to the new ESX Server host and location. You can bring up the server on the new host.

**Live Migration with VMotion**

To showcase live migration, we migrated a running WASVM (WASVMND1clone) and moved it to another ESX Server host.

Make sure to disable access to the CD/DVD drive (uncheck “connected” and “connect at power on”).

Select the destination host.
Select the "resource pool".
Select "High Priority".

Click “Finish”.

Monitor progress in the “recent tasks” window.

This should successfully migrate the running VM to the destination. All the applications and processes should continue to run as it was prior to the migration.
APPENDIX B:

Configuring Additional Storage on VMware ESX Server Host (local or SAN)

Once the SAN LUNs are created and allocated to the host ESX Server, do the following:

Click on host IP/name → configuration tab → storage

Click on “add storage”
It shows a list of available devices to be added (local or SAN)

Click “Next”
Click “Next”

Enter the datastore name (WASStore)
Once completed, the datastore is created and is available for VM use.
APPENDIX C:

References

VMware Infrastructure 3 Documentation
http://www.vmware.com/support/pubs/vi_pubs.html

IBM WebSphere V6 Documentation
http://publib.boulder.ibm.com/infocenter/wasinfo/v6r0/index.jsp

IBM-VMware Alliance Website
http://www.vmware.com/partners/alliances/technology/ibm.html

VMware Official Site
www.vmware.com

IBM Official Site
www.ibm.com

VMware Technology Network
http://www.vmware.com/vmtn/