VMware Infrastructure 3

VDI Server Sizing and Scaling

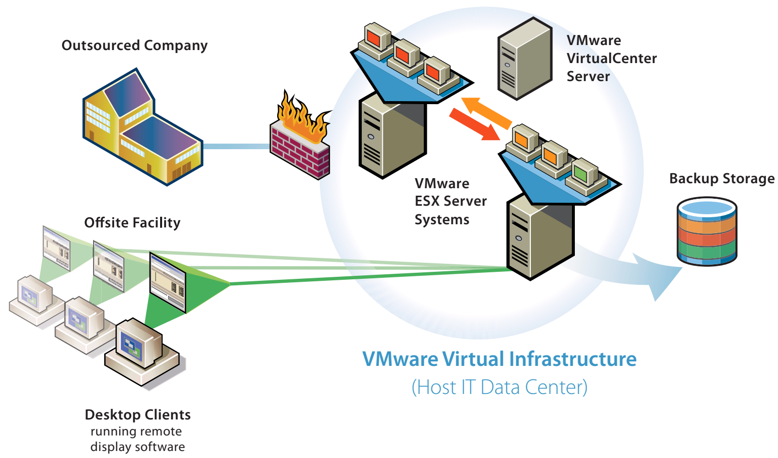
VMware Virtual Desktop Infrastructure (VDI) is a PC management solution for enterprise desktop administrators. VDI is a server-based computing offering that provides desktop environments as an enterprise hosted service. VDI allows administrators to maintain and manage all user applications, data, and environments in the centrally located data center. VDI provides a true, isolated PC environment while sharing underlying hardware resources. VDI allows administrators to provide PC environments with all the benefits of virtualization — central management, hardware consolidation, and resource flexibility.

This white paper describes the testing methodology, results, and analysis and sizing guidelines for setting up Virtual Desktop Infrastructure using VMware Infrastructure 3. It covers the following topics:

Understanding Virtual Desktop Infrastructure

A thin-client architecture enabled through VMware Virtual Desktop Infrastructure provides a new way to address enterprise desktop problems such as security, manageability, reliability, and uniformity of working environments.

VDI makes it possible to build a thin-client infrastructure that optimizes usability, manageability, total cost of ownership, and flexibility. Complete desktop environments can run in virtual machines on datacenter servers and can be accessed by end users from any PC or thin client on the corporate network. This solution provides IT with centralized control over desktop computing resources and their data as well as the ability to consolidate virtual machines and optimize resource utilization across the datacenter. Users gain the flexibility of being able to access their complete desktop environment from any location from any client.



VDI is built on VMware Infrastructure. In its most basic implementation, enterprise desktops are hosted on VMware Infrastructure and users connect to them using a remote protocol such as RDP. Other protocols such as VNC or third-party applications such as Radmin, GraphOn, and pcAnywhere can be used for remote access, as well.

VDI Benefits

Virtual Desktop Infrastructure offers the following key benefits:

- Desktop environments are isolated.
- Data is secure in the data center.
- All applications work in a virtual machine.
- Normal management tools work in a virtual machine.
- Images are managed centrally.
- Hardware can be consolidated.
- Desktops are always on and always connected.
- Users have access to their desktops from anywhere.

Going Beyond the Basics

An enterprise-class deployment requires customers to leverage additional features of the VMware Infrastructure 3 suite such as Dynamic Resource Scheduling and High Availability Services. The enterprise-level load balancing provided by Dynamic Resource Scheduling will be of particular interest.

And for many customers it will be particularly important to use a partner product or the VirtualCenter SDK to create a VDI portal that can manage user access and user sessions.

VMware has teamed up with leading technology vendors and service providers in the VDI Alliance to offer comprehensive solutions for VMware Virtual Desktop Infrastructure deployments. These partners span a range of technologies, including servers, thin terminals, user management, application provisioning, and related software. More information is available on the VMware Web site. For a link, see References on page 19.

Server Capacity Overview

The actual number of desktops that a specific configuration of servers can support varies depending on such hardware characteristics as the processor type, the amount of memory installed, the storage configuration, the network configuration, the remote protocol used, and the demands of individual users (typing speed, applications used, frequency of access, and so forth). We ran experiments to study two such workloads with clients using the RDP protocol for remote display. We ran the experiments multiple times and the same capacity results were seen. The details are described in Light Worker Workload on page 18 and Heavy Worker Workload on page 19. Unless otherwise stated, all ESX Server and guest operating system parameters were left at their default settings. The number of desktops that can be comfortably supported on ESX Server 3.0 running on an HP ProLiant DL 385 G1 server with two dual-core 2.2GHz Opteron processors under our test loads is shown in Table 1.

Workload	Number of Desktops	Desktop Activation	Desktop Configuration
Light worker	42	Resume from suspended state	Windows XP SP2 256MB guest
Heavy worker	26	Resume from suspended state	Windows XP SP2 384MB guest

Table 1

Test Environment

The key components of the test environment are shown in figure 1 and listed below.

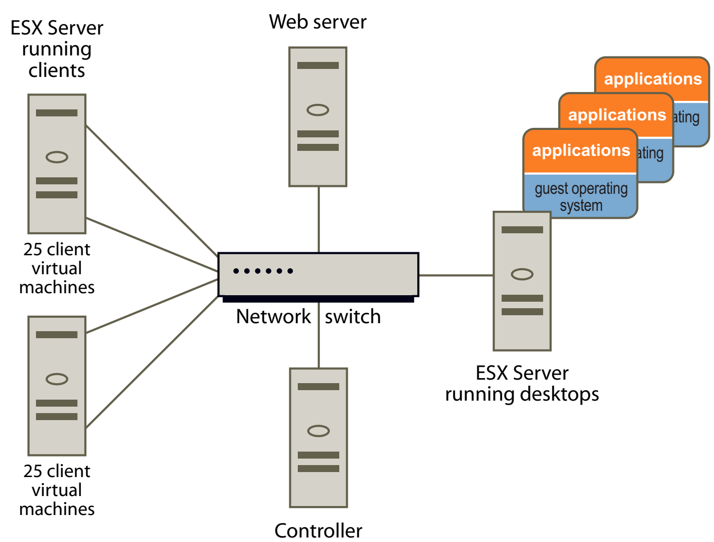


Figure 1 — Test environment

Server Running Desktops

Server Hardware

- HP ProLiant DL385 G1
- Processor (two-way, dual core): AMD Opteron 2.2GHz
- Memory: 16GB

- Networking: 2 Broadcom dual-port gigabit Ethernet adapters
- Storage: 2 Ultra 320 SCSI (2 × 146GB disks, 15,000 rpm) configured as RAID 0

Server Software and Configuration

- ESX Server 3.0, Build 27701
- Service console memory: 384MB
- All virtual machines and service console connected to the same virtual switch uplinked to a physical Ethernet adapter

Desktop Virtual Machine Configurations

- CPU: Single virtual CPU
- Memory: 256MB (light worker case); 384MB (heavy worker case)
- Connectivity: vmxnet
- Virtual disk: 2GB with BusLogic SCSI controller
- Guest operating system: Windows XP Professional with SP2
- VMware Tools installed
- Guest operating system configured to go into standby mode after 10 minutes of inactivity

Web Server

- Dell PowerEdge 350, 851MHz Intel Celeron Processor, 768MB memory

Controller

- Dell PowerEdge 350, 851MHz Intel Celeron Processor, 512MB memory

Servers Running Clients

Server hardware

- 2 HP ProLiant DL385 G1
- Processor (two-way, single core): AMD Opteron 2.6GHz
- Memory: 8GB
- Storage: 2 Ultra 320 SCSI drives (2 × 73GB disks, 15,000 rpm) configured as RAID 0

Server Software and Configuration

- ESX Server 2.5.3, Build 22699
- Service console memory: 384MB

Client Virtual Machine Configurations

- Guest operating system: Windows XP Professional with SP2
- Virtual disk: 4GB with BusLogic SCSI controller
- Memory: 256MB

Note: Timing numbers reported within a virtual machine can be inaccurate, especially when the processor is overcommitted. Although the clients run the benchmark workloads, the timing (DU time) used in this benchmark is measured from the controller machine. Further, timing was carefully monitored during the experiments to ensure that time was kept accurately during the course of these experiments. For a discussion of timing issues in virtual machines, see the paper Timekeeping in VMware Virtual Machines. For a link, see References on page 19.

Connectivity

- 100 Mbps network connection between clients and servers
- Remote protocol: RDP

Workload Design

Terminal Server scripts are used to generate load on the desktops from the remote clients. The scripts use the `tbscript.exe` interpreter available as a part of robokit (Windows Server 2003 Resource Kit) from the Microsoft Web site. For details, see References on page 19. Two different user scenarios are tested in accordance with Gartner Group recommendations (heavy worker and light worker), though these workloads provide only a rough approximation of the workloads generated by actual users doing real work. The applications, features, and data sets in the scenarios are representative but limited. In addition, the activity is sustained at a steady pace over a period of one or two hours, with no long breaks and a repetitive set of actions using the same functions and data sets. The workloads are described in Heavy Worker Workload on page 19 and Light Worker Workload on page 18.

To capture a realistic customer scenario, each client connects to a single desktop. This is important in a benchmark environment because if a physical system hosts more than one client, the network traffic characterization will be different from that of a real customer situation. The clients run the benchmark on the desktops in a timely fashion, managed by the controller program to inject more users to generate more load on the server. The controller uses the open source STAF interface to start the workload from each client against a single virtual machine. For details on the STAF interface, see References on page 19.

Figure 2 shows a schematic representation of the benchmark behavior. The controller for the benchmark uses the VirtualCenter SDK to activate the desktop.

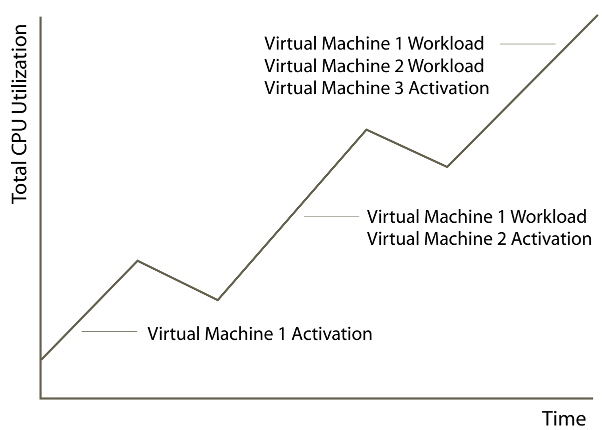


Figure 2 — Benchmark behavior

The benchmark activates the virtual machines one at a time. After the first virtual machine is resumed from suspended state, the controller program waits for two minutes before starting the load script from the first client. As seen in the graph, the CPU utilization climbs due to the resume operation. That climb is followed by a short idling phase, during which CPU utilization dips slightly. This is followed by the benchmark phase, which causes CPU utilization to climb again. Another desktop is added to the system, and the second client is used to run the workload against the newly added desktop. The workload is run in a loop in each virtual machine. Therefore the cumulative CPU Utilization in the system increases with the addition of new desktops. This process continues until the benchmark terminates.

The desktops are kept in suspended state at the Windows logon screen to avoid complete reset operations, which consume considerably more CPU resources and add more burstiness in the workload behavior. The desktops are not powered on before the test cycle begins because idling virtual machines consume CPU resources. Consequently, the results of such a test would not provide an accurate analysis of the change in end user behavior with addition of more desktops in the system.

Determining the Termination Point

The first client is treated as a canary client. The canary client runs the complete script — generating heavy or light load, depending on the workload of the current test. The time it takes the canary client to run the complete script without failures is called a desktop unit (DU) and is used in determining when to terminate the benchmark.

The controller program observes the DU time for each iteration of the workload by the canary client and determines whether the benchmark should be continued based on the termination criteria. If the time taken for the iteration is more than 10 percent above that required for the previous iteration — or if any portion of the script fails as a result of predefined timeouts — the server is considered to have reached its maximum capacity. That point marks the benchmark termination.

Note: Since the termination occurs as a result of the addition of the last desktop, the maximum number of supported desktops for a particular configuration is the number of desktops running at the termination point minus one.

In order to determine the server profile under load, we collected statistics even after the benchmark termination point. We collected the server statistics using the ESXTOP utility available on ESX Server at one-minute intervals.

Interpreting the Results

Since VMware does not provide the remote protocol and the latencies will vary based on the protocol used and the physical distance between the clients and the desktops (LAN or WAN deployment), we do not monitor the latencies for specific GUI operations during the workload. We show the time taken to complete the DU, giving results that are helpful for capacity planning purposes. The results capture high level performance characteristics by measuring the change in latencies of the complete iterations (in minutes) but do not provide a qualitative analysis of remote performance for specific GUI operations.

Any interpretation of the results and analyses presented in this paper must take into account the nature of the test workload. Workloads that involve different applications, different data sets, and different activity patterns will yield different results. Thus, the results shown here are only suggestive and do not indicate the best or worst performance that may be seen in a real-world deployment of Virtual Desktop Infrastructure.

Performance Results

The descriptions and charts below summarize the results for the light and heavy workloads used in the tests.

Light Worker Workload

The light worker workload is a representation of a data entry worker. It uses components of Microsoft Office, including Microsoft Internet Explorer, Microsoft Word, and Microsoft Excel, in a robotic fashion. The workload details are described in Light Worker Workload on page 18

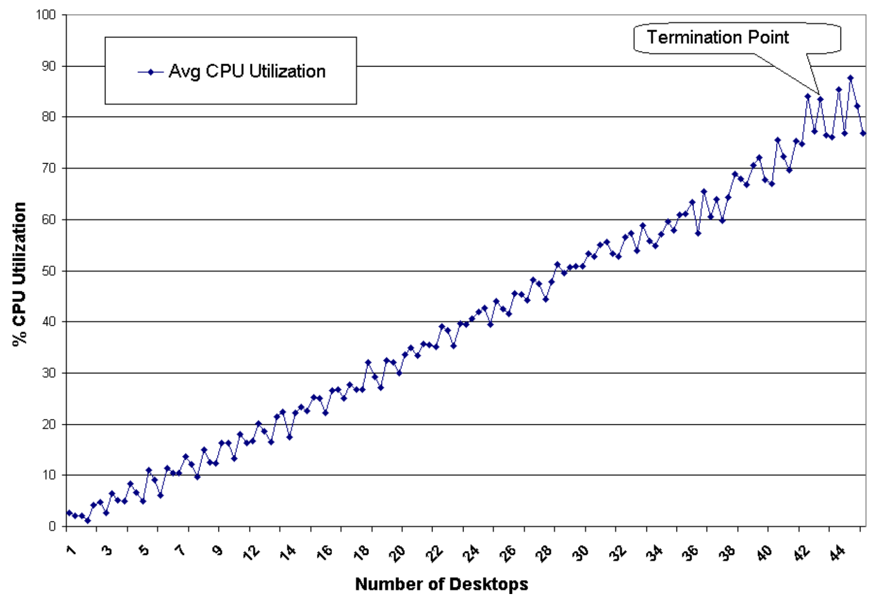


Figure 3— Light worker CPU load profile

Figure 3 shows the CPU load on the system as the number of desktops running on the ESX Server host increases. The CPU load on the system increases linearly until the termination point, which marks the conservative maximum acceptable load on the system. One of the clients failed because the latencies for some GUI operations exceeded the predefined timeouts after resuming the 43rd desktop. Following the benchmark methodology, the number of desktops supported in this configuration is therefore 42.

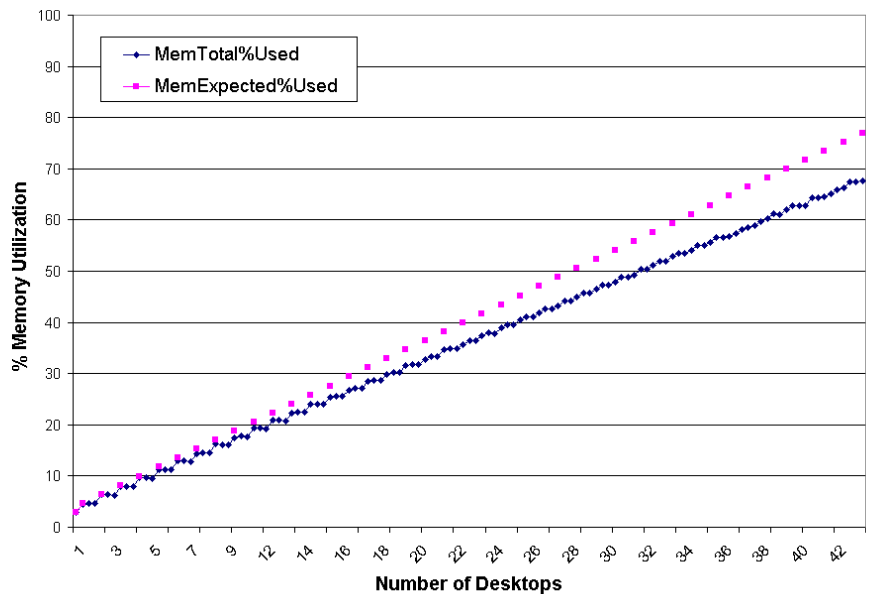


Figure 4 — Light worker memory load profile

Figure 4 shows the memory load profile on the host system. The vertical axis shows the percentage of total memory that has been used by the system.

The points showing MemExpected %Used represent the sum of the total amount of memory needed by the running virtual machines, cumulative virtualization memory overhead, memory used by the service console, and memory used by the ESX Server kernel. The points showing MemTotal %Used, in contrast, represent the aggregate amount of memory actually used by the system, including all the elements listed above.

The difference between the MemTotal %Used and the MemExpected %Used points in figure 4 is due to page sharing optimization in ESX Server.

In the above graph, the average virtualization memory overhead used for each guest is 30MB, as observed in the data collected using ESXTOP. The memory overhead indicates the current memory overhead and not the reserved memory overheads for the virtual machine. Overhead memory includes space reserved for the virtual machine frame buffer and various virtualization data structures. Overhead memory depends on the number of virtual CPUs, on the configured memory for the guest operating system, and on whether you are using a 32-bit or 64-bit guest operating system.

For additional details about memory resource management, see *Resource Management Guide: ESX Server 3.0 and VirtualCenter 2.0* , page 121 (Understanding Memory Overhead). See References on page 19 for a link.

As seen in the graph, there is enough memory in the system to support more guests, but the benchmark terminates before the system is CPU- or memory-saturated. Generally for remote access workloads, due to the bursty nature of desktop applications, the latencies for operations increase drastically after the CPU load on the system is 80 to 90 percent. It is therefore recommended to keep 80 percent CPU utilization as the upper bound for maximum capacity purposes.

The benefits of page sharing are conspicuous only after the number of running desktops reaches a certain level. The difference between the percentage of memory expected to be used and the actual percentage of total memory used in the system increases with the number of active desktops, which indicates the rise in page sharing gains as more desktops are added.

The page sharing benefit seen in figure 4 can be attributed to the fact that all the desktops are Windows XP virtual machines. The results cannot be generalized to a mix of guest operating systems. The page sharing benefits also depend on the kinds of applications used by the guests. Further benefits from sharing zero pages will be reduced if the guests run many applications and therefore use all the allocated physical memory.

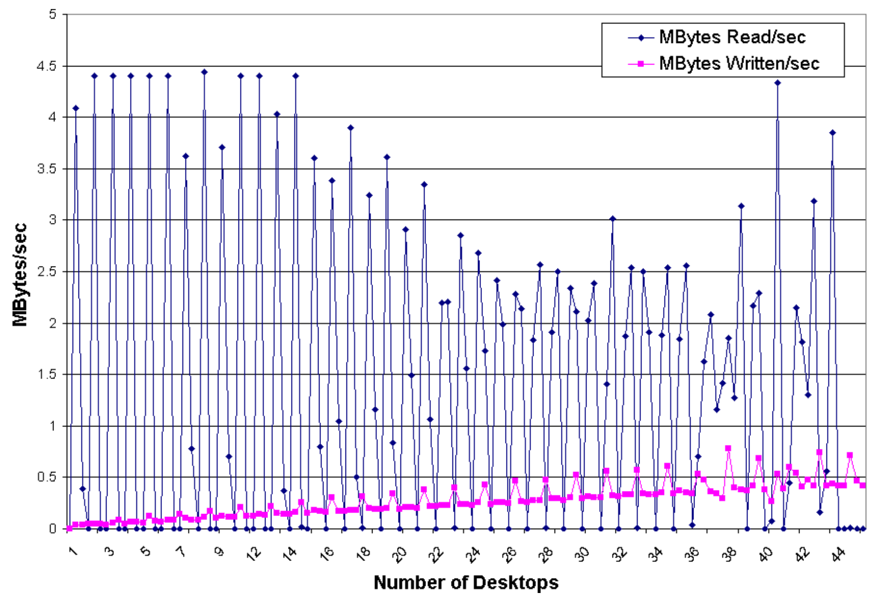


Figure 5— Light worker disk I/O load profile

Figure 5 shows the disk I/O profile during the workload. The spikes seen in the graph indicate the operations to resume a virtual machine from suspended state. The other disk reads and writes can be attributed to the office applications running inside the guests, such as reading and writing Excel sheets and writing a Word document to disk. The disk I/O and network I/O behavior are mostly reflections of the kinds of activity going on in the guest once the guest is powered on.

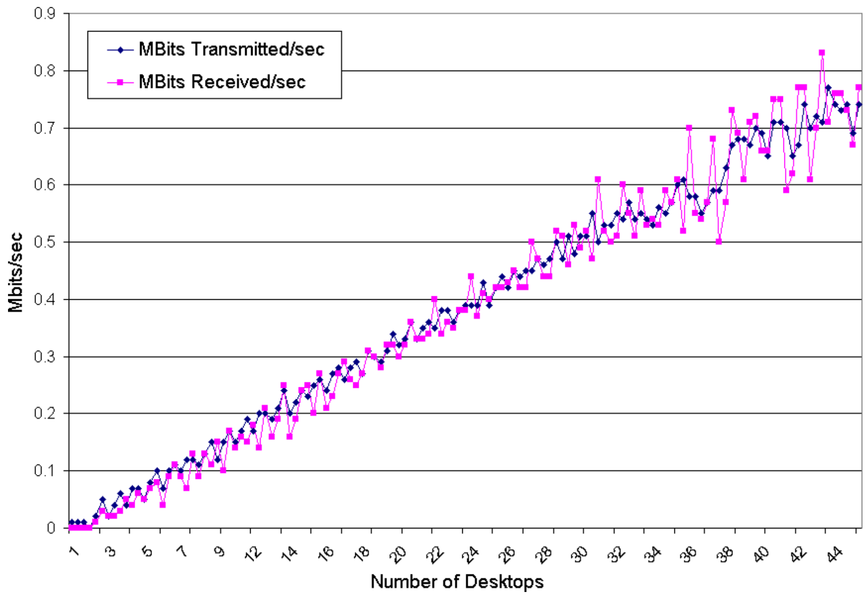


Figure 6— Light worker network I/O load profile

Figure 6 shows the network I/O behavior during the workload. The light worker workload consists of visiting some Web pages with graphics, and therefore the network I/O profile shows a good load of receive and transmit traffic from the ESX Server host. The majority of receive traffic is Web pages requested by the guests. The transmit traffic increases mostly because of the screen updates going to the RDP clients connected to the desktops running on the ESX Server.

Note: The traffic shown above is the total traffic going out on the wire from the ESX Server host. It includes the bidirectional traffic to the clients and the bidirectional traffic to the Web server. The traffic will vary based on the remote protocol used and the optimizations in the remote protocol for different connection speeds.

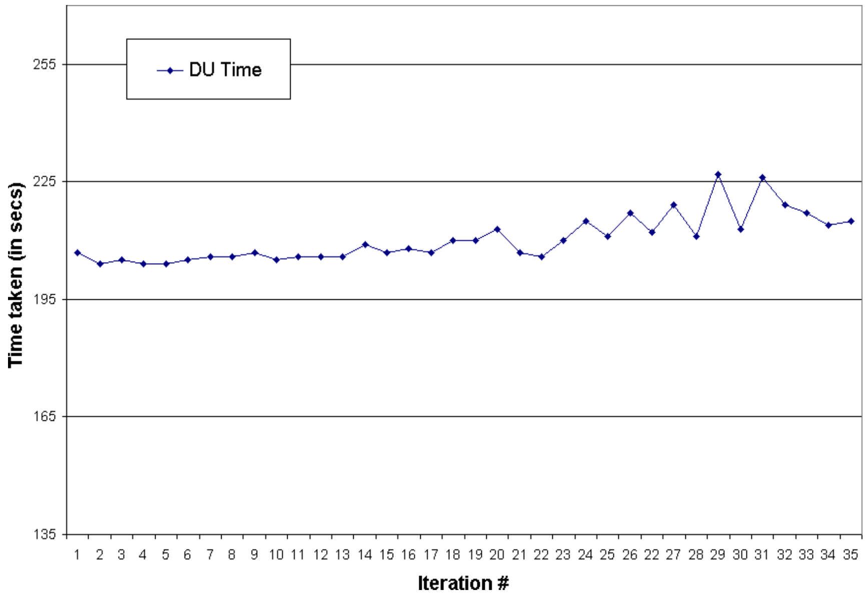


Figure 7 — Light worker canary client profile

Figure 7 shows the canary client profile. The time taken for each DU remains the same despite the increasing CPU load on the system. As shown in the figure, the DU time stays flat with some variation per iteration, which suggests that even with the increasing CPU load in the system, the ESX Server scheduler is fair to the canary guest and to the other guests. The benchmark terminated because a script on one of the clients failed to run to completion, not because of an increase in DU time.

Light Worker Workload with Cold Reset

For the results shown in the preceding graphs, the benchmark script activates the desktop for a user by resuming it from suspended state. The desktops are suspended at the Windows log-on screen.

Another way of activating the desktop for a remote user is to use the power-on operation that corresponds to a cold reset for a physical machine.

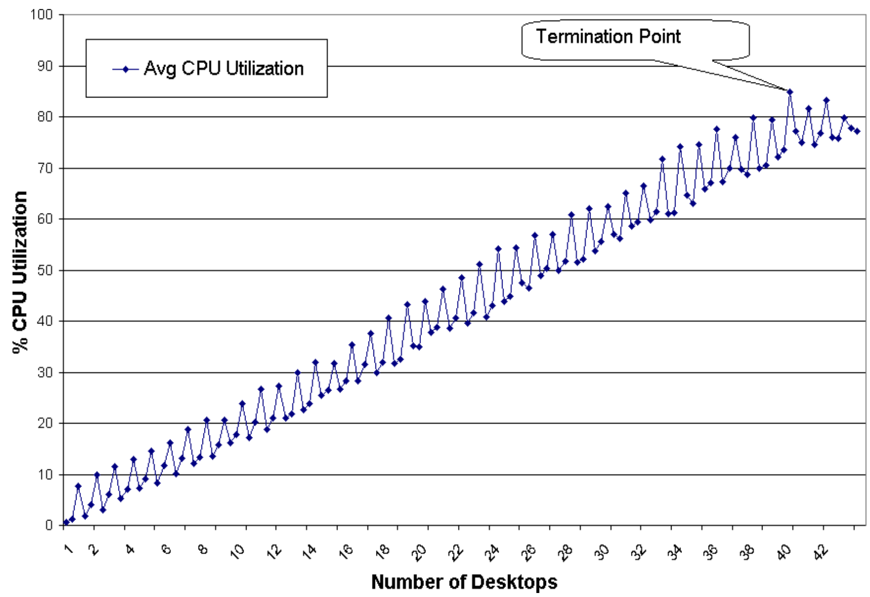


Figure 8 — Light worker cold-reset activation CPU load profile

Figure 8 shows the CPU load profile for a case using a cold reset. Compared to resuming from suspended state, the power-on operation is more CPU intensive. As a result, the CPU utilization of every desktop is very high for a brief period, shown by the spikes and dips in the graph. This situation is highlighted to demonstrate the fact that different desktop activation schemes may result in different behavior. One of the clients failed after powering on the 41st desktop, therefore this configuration supports 40 desktops. The other characteristics — such as memory load, disk I/O, and network I/O behavior— remain the same and are not shown here.

As seen in the graph, desktop activation policy also affects the capacity of the system. Therefore, consider keeping virtual machines in suspended state or in standby mode if all users activate their desktops in a short period of time. For details on standby mode, see Heavy Worker Workload with ESX Server 3.0 ACPI S1 Feature on page 16.

Heavy Worker Workload

The heavy worker workload is a representation of a knowledge worker. It uses components of Microsoft Office, including Microsoft PowerPoint, Microsoft Word, and Microsoft Excel, in a robotic fashion. The workload details are described in Heavy Worker Workload on page 19.

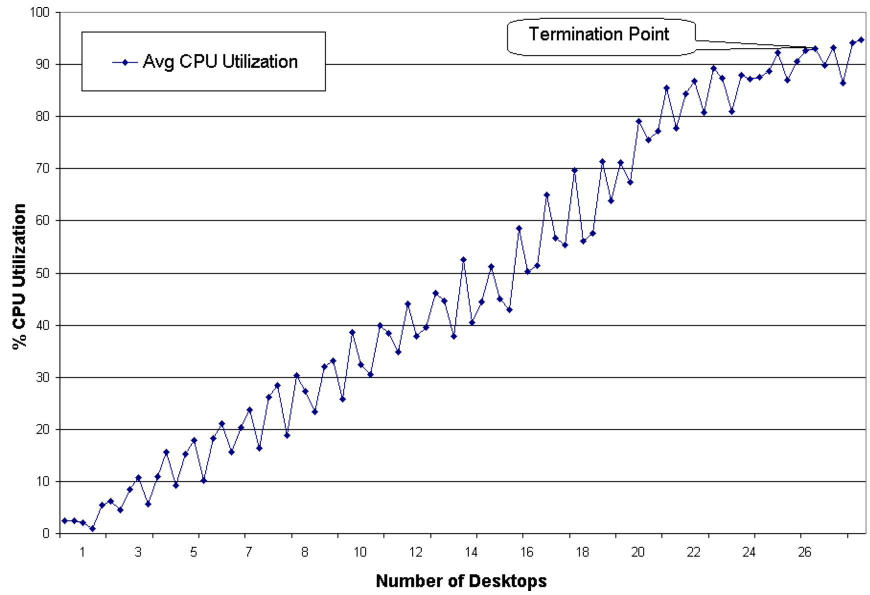


Figure 9 — Heavy worker CPU load profile

Figure 9 shows the CPU load on the system as the number of desktops running on the ESX Server host increases. The CPU load on the system increases linearly until the termination point, which marks the conservative maximum acceptable load on the system. The termination criteria are met after the 27th desktop is powered on. One of the clients failed because the latencies for some GUI operations exceeded the predefined timeouts after resuming the 27th desktop, therefore this configuration supports 26 desktops.

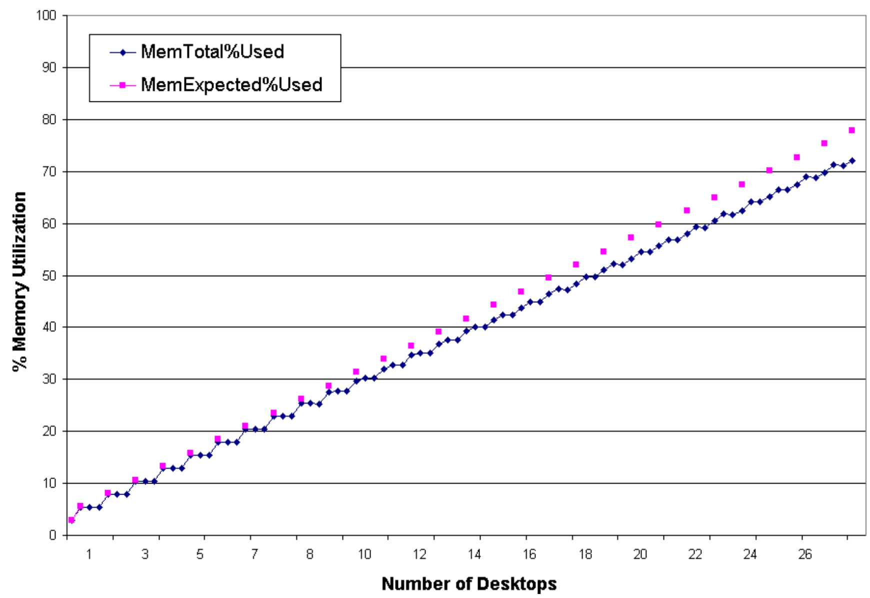


Figure 10 — Heavy worker memory load profile

Figure 10 shows the memory profile during the workload. The graph shows the actual memory percentage use reported by ESX Server.

The points showing MemExpected %Used represent the sum of the total amount of memory needed by the running virtual machines, cumulative virtualization memory overhead, memory used by the service console, and memory used by the ESX Server kernel. The points showing MemTotal %Used, in contrast, represent the aggregate amount of memory actually used by the system, including all the elements listed above.

The difference between the MemTotal %Used and the MemExpected %Used points in figure 10 is due to page sharing optimization in ESX Server.

In the above graph, the average virtualization memory overhead used for each guest is 35MB, as observed in the data collected using ESXTOP. During the heavy worker workload, as in the light workload case, the benefits of page sharing are conspicuous only after the number of running desktops reaches a certain level.

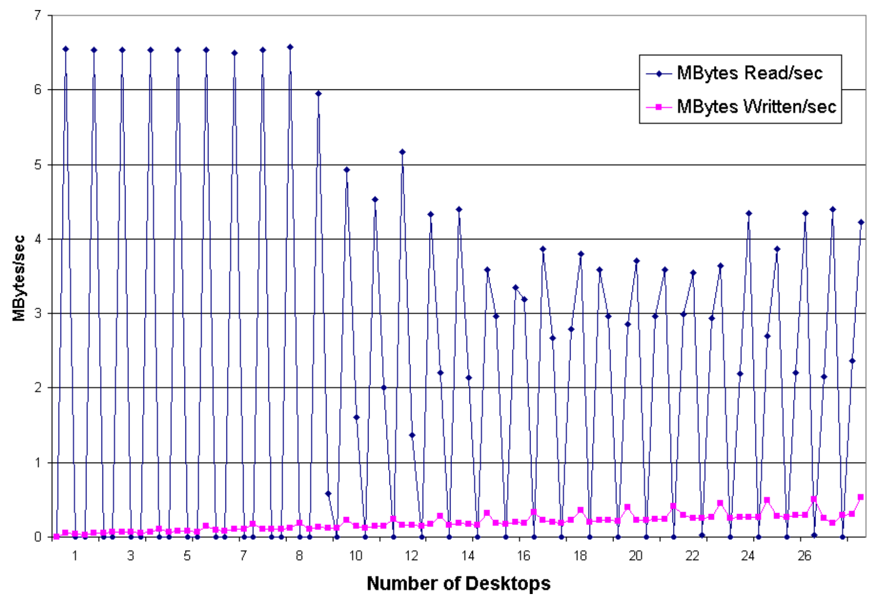


Figure 11 — Heavy worker disk I/O profile

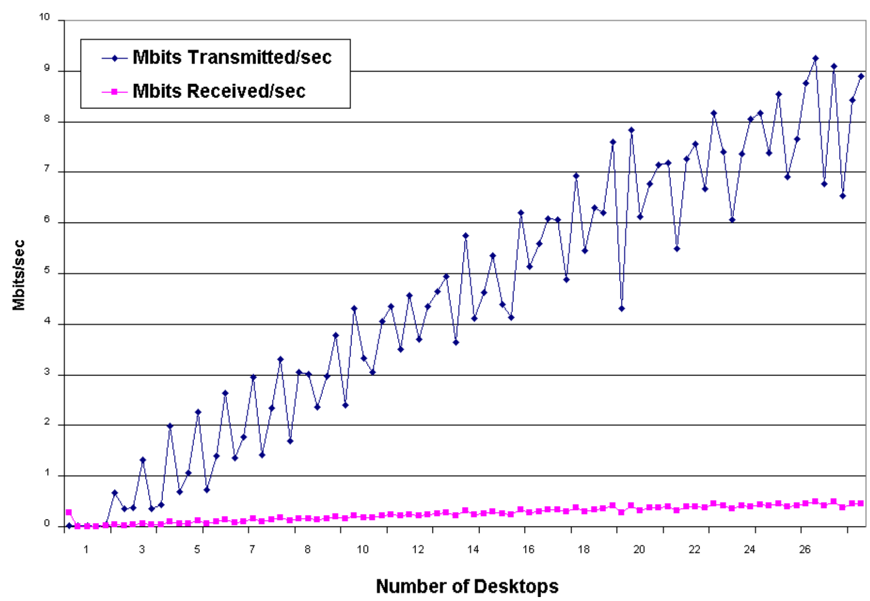


Figure 12 — Heavy worker network I/O profile

Figures 11 and 12 show the disk I/O and network I/O behavior of the host system during the workload. As mentioned before, the disk and network I/O profiles are dependent on the guest activity and therefore will vary with users' activity. In this case, the guest workload runs through a PowerPoint presentation, then loads and saves an Excel sheet, so there is some disk activity. In figure 6, the disk read spikes indicate the operations that resume desktops from the suspended state. Similarly, since the workload streams a PowerPoint presentation across the network, the network throughput is more aggressive than it would be if driven by the expected work pattern of an average user. In the test load, network transmit activity totally dominates the network traffic. In this graph, the network receive throughput can be attributed to incoming RDP requests.

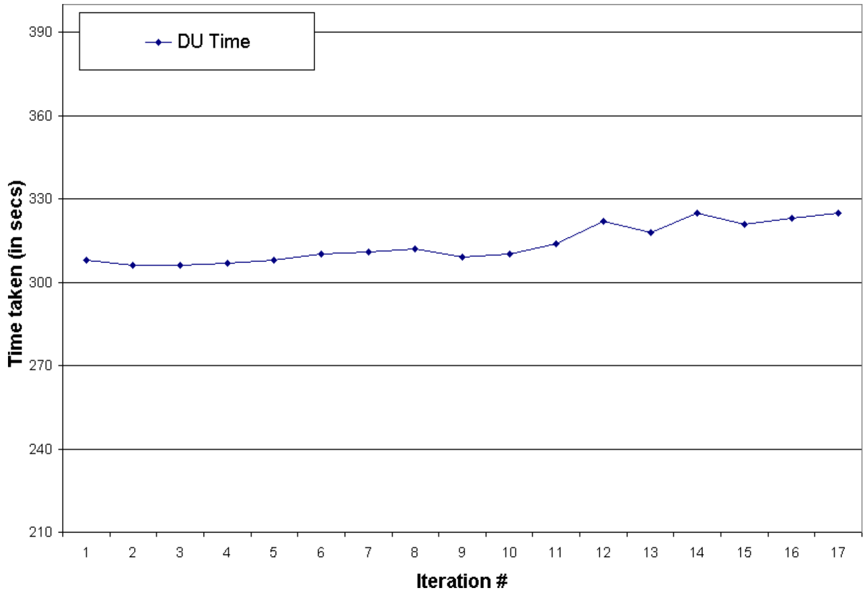


Figure 13 — Heavy worker canary client profile

Figure 13 shows the time the canary client takes to run each desktop unit. The canary client completes the batch of operations that constitutes the heavy worker workload and runs in a loop. As seen in the case of the light worker workload, the DU time stays flat with little variation from iteration to the next. The benchmark terminated because a script on one of the clients failed to run to completion, not because of an increase in DU time.

Heavy Worker Workload with ESX Server 3.0 ACPI S1 Feature

For the results shown in the preceding graphs, the benchmark script activates the desktop for a user by resuming it from suspended state. The desktops are suspended at the Windows log-on screen.

Another way of of activating the virtual machines is to power on approximately the number of desktop virtual machines you estimate the host can support, with the virtual machines configured to go into standby mode after a certain fixed interval. ESX Server 3.0 introduces new power management options to determine how the virtual machine responds when the guest operating system is placed on standby. In this case the Windows XP desktops are configured to Wake on LAN for virtual machine traffic on the default virtual machine network. When any network traffic arrives for a virtual machine, it seamlessly wakes up from standby mode.

In this case we powered on 28 desktops and waited for approximately 15 minutes before starting the benchmark on the virtual machines. This allowed enough time for the virtual machines to go into standby mode. The effect is shown in the early phase of figure 14.

The controller program waits for three minutes between the time it starts the workload against one client and the time it starts the workload on the next client. The figure shows the increase in CPU utilization due to the increase in the load from the clients.

As seen in the previous case, the termination criteria are met after the 27th desktop is powered on. One of the clients failed because the latencies for some GUI operations exceeded the predefined timeouts after resuming the 27th desktop, therefore this configuration supports 26 desktops.

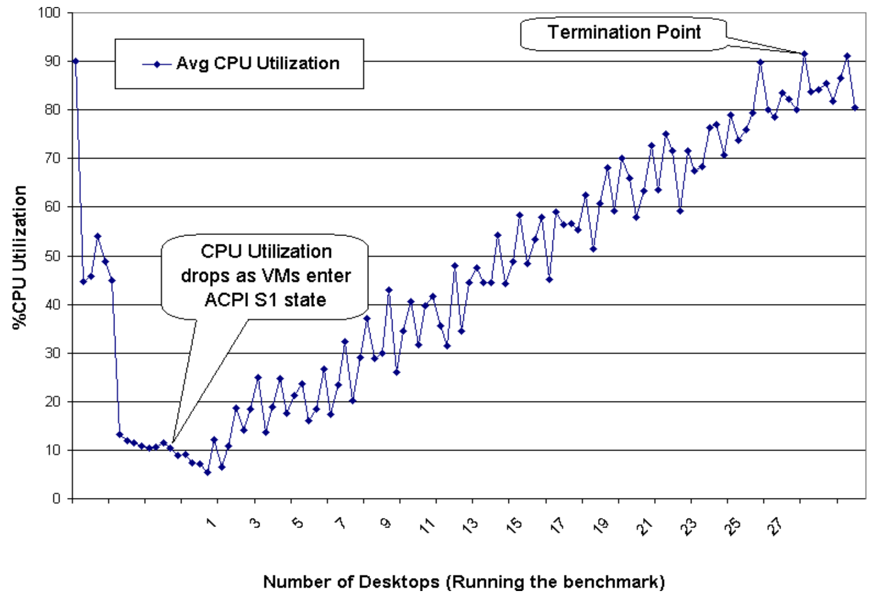


Figure 14 — Heavy worker CPU load profile using ACPI features

This illustrates the fact that keeping the virtual machines in standby mode using the Wake on LAN option is another useful activation scheme for desktop hosting. Compared to the resume from suspended state or cold reset option, this activation scheme offers very fast activation (on the order of a few seconds) as it is based on the Wake on LAN feature. The end user does not have to wait for the desktop to resume in such a deployment scheme.

Details of how to configure the virtual machines for power management are described in *Basic System Administration: ESX Server 3.0 and VirtualCenter 2.0*, p. 150. See References on page 19 for a link.

Summary of Results

Virtual Desktop Infrastructure provides a new approach to host desktops on servers using VMware Infrastructure. It provides complete isolation between different desktops running on the ESX Server along with the always-available remote access functionality.

This paper illustrates the methodology you can use when determining the server capacity needed for a VDI deployment.

We looked into two different workloads for capacity planning guidelines. For a light worker workload an HP DL 385 G1 server could support 42 Windows XP virtual machines. For a heavy worker workload the same server supported 26 Windows XP virtual machines.

The results are conservative, however, with a server considered to be at capacity when the client fails either due to a 10 percent increase in the canary time observed for the workload or when a client script fails due to predefined timeouts.

In assessing your own server needs, keep this conservative approach in mind — along with differences between the artificial test workloads and the actual workloads generated by your real-world users. Use this analysis and your own testing to determine the server capacity needed to support the number of desktops you plan to deploy, taking peak workloads into account and leaving appropriate room for growth. The best test is with your own workload, since the consolidation ratio depends on load, as this study has shown.

Setting Up the Tests

This paper presents the results of tests using the hardware configurations and workloads described below.

Hardware Configuration for Server Running Desktops

One HP ProLiant DL385 G1. Two 2.2GHz AMD Opteron dual-core processors, 16GB RAM, 2 Ultra 320 SCSI drives (2 × 146GB disks, 15,000 rpm)

Hardware Configuration for Servers Running Clients

Two HP ProLiant DL385 G1. Two 2.6GHz AMD Opteron single-core processors, 8GB RAM, 2 Ultra 320 SCSI drives (2 × 73GB disks, 15,000 rpm)

VMware Product Use and Virtual Machine Configuration

- VMware product: ESX Server 3.0, build 27701 (in server running desktops)
- VMware product: ESX Server 2.5.3, build 22699 (in servers running clients)
- Virtual Ethernet adapter: vmxnet
- Disk: SCSI, BusLogic
- Floppy: No
- CD-ROM: No
- USB: No
- Sound: No
- VMware Tools installed: Yes
- Local or remote VMware Console: No
- Guest operating system: Windows XP SP2
- Network connection between the desktops and clients: 100Mbps

Light Worker Workload

The light worker workload is designed to simulate data entry workers.

Data entry workers input data into computer systems

Example jobs include transcription, typing, order entry, clerical work, and manufacturing.

Steps in the Light Worker Workload

1. Connect to a Windows XP virtual machine and log on.
2. Do the following in a loop:
    a. Start Internet Explorer. Load a Web page with heavy graphics. Close Internet Explorer.
    b. Start Word. Type a small document. Close Word.
    c. Start Excel. Open an Excel sheet. Close Excel.
    d. Start Internet Explorer. Load a Web page with heavy graphics. Close Internet Explorer.

Heavy Worker Workload

The heavy worker workload is designed to simulate knowledge workers.

Knowledge workers are defined as workers who gather, add value to, and communicate information in a decision support process. Cost of downtime is variable but highly visible. These resources are driven by projects and ad hoc needs towards flexible tasks. These workers make their own decisions on what to work on and how to accomplish their tasks.

Example jobs include marketing, project management, sales, desktop publishing, decision support, data mining, financial analysis, executive and supervisory management, design, and authoring.

Steps in the Heavy Worker Workload

1. Connect to a Windows XP virtual machine and log on.
2. Do the following in a loop:
   a. Start PowerPoint. Load a massive presentation and browse the slides. Close PowerPoint.
   b. Start Internet Explorer. Browse three different Web pages. Close Internet Explorer.
   c. Start Command Prompt. Do a directory listing.
   d. Start PowerPoint. Load a massive presentation and browse the slides. Close PowerPoint.
   e. Start Excel. Open an Excel sheet. Close Excel.
   f. Start PowerPoint. Load a massive presentation and browse the slides. Close PowerPoint.
   g. Start Word. Type a small document. Close Word.

References

Basic System Administration: ESX Server 3.0 and VirtualCenter 2.0.
www.vmware.com/pdf/vi3_admin_guide.pdf

Microsoft Windows 2003 Resource Kit Tools (`rktools.exe`)
www.microsoft.com/downloads/details.aspx?FamilyID=9d467a69-57ff-4ae7-96ee-b18c4790cffd&DisplayLang=en

Resource Management Guide: ESX Server 3.0 and VirtualCenter 2.0
www.vmware.com/pdf/vi3_esx_resource_mgmt.pdf

STAF
www.staf.sourceforge.net.

Timekeeping in VMware Virtual Machines
www.vmware.com/pdf/vmware_timekeeping.pdf

Virtual Desktop Infrastructure Alliances
www.vmware.com/partners/alliances/solutions/

VMware, Inc. 3145 Porter Drive Palo Alto, CA 94304 www.vmware.com