1-2014

Server-Based Desktop Virtualization

Anas Raqi Shabaitah

Server-Based Desktop Virtualization

By

Anas Raqi Shabaitah

<Thesis>

Presented to the Faculty of the Graduate School of RIT University in Partial Fulfillment

Of the Requirements for the Degree of

Master of Science in Network, Systems and Security Administration (NSSA)

Rochester Institute of Technology (RIT)

Department of Networking and Systems Administration

Golisano College of Computing and Information Science

www.rit.edu

<Jan, 2014>

Advisor: Prof. Charles Border

Committee Members: Prof. Khalid Khawaja, Prof. Muhieddin Amer.

Dedication

I dedicate my thesis work to my family. A special feeling of gratitude to my wife, Reema, who never left my side and encouraged me. I also dedicate this work to my lovely daughters, Jana and Sarah.

Acknowledgements

I would like to express my deepest gratitude and appreciation to my advisor, Professor Charles Border, for his excellent guidance and caring. I would also like to thank my committee members, Professor Khalid Khawaja and Professor Muhieddin Amer. A special thanks to my family. At the end, I would like to express appreciation to my beloved wife, Reema, who was always my support in the moments when there was no one to answer my queries.

Abstract

Server-Based Desktop Virtualization

Anas Raqi Shabaitah, MSc

RIT University, 2014

Virtualization can be accomplished at different layers in the computational stack and with different goals (servers, desktops, applications, storage and network). This research focuses on server-based desktop virtualization. According to the Gartner group, the main business drivers for adopting desktop virtualization are: application compatibility, business continuity, security and compliance, mobility and improved productivity **[15]**. Despite these business drivers, desktop virtualization has not been widely adopted. According to a survey conducted by Matrix42, only 5% of desktop computers are virtualized **[37]**. The research deals with the challenges preventing the wider adoption of server-based desktop virtualization while focusing on two of the main virtualization architectures: session-based desktop virtualization (SBDV) and virtual desktop infrastructure (VDI).

The first chapter introduces some of the challenges faced by large organizations in their efforts to create a cost effective and manageable desktop computing environment. The second chapter discusses two of the main server-based desktop virtualizations (VDI and SBDV), illustrating some of the advantages and disadvantages in these different architectures. The third chapter focuses on some of the technical challenges and provides recommendations regarding server-based desktop virtualization. In the fourth chapter, measurements are conducted for the utilization and performance of SBDV on different

user profiles (light, heavy and multimedia). Data and results collected from desktop assessment and lab are used to formulate baselines and metrics for capacity planning. According to the conducted measurements, it is concluded that light and heavy profiles can be virtualized using SBDV, while for multimedia profiles, additional capacity planning and resource allocation are required. Multimedia profiles can be virtualized with VDI considering client-side rendering to avoid network bandwidth congestion.

While the research focuses on VDI and SBDV, it highlights few points related to client access devices (CADs). CADs are one of the main components in the desktop virtualization stack (OS virtualization, session virtualization, application virtualization, connection broker, CADs and user data and profiles). The latter chapter of the research focuses on conclusions and future work toward greater levels of adoption of VDI and SBDV.

Table of contents

Introduction

Desktop computers have changed and evolved over the last decade. Over years, desktops have been transformed from being large and slow machines to small and fast machines in order to meet the increasing demand. The first desktop computer kit was available in 1974 by Micro Instrumentation Telemetry Systems (MITS). At that time, computers were just kits. Nothing was assembled till the 1980s. The first versions of the assembled desktops were expensive and used to run BASIC language. The first desktop computer that became the pattern for the current home computer was the IBM PC Junior. IBM PC Junior is shown in *Figure 1* below. The number of desktop computers rapidly increased over years. According to *Forrester Research*, the total number of computers (desktops and laptops) in 2012 is around 1.4 Billion **[4]**.

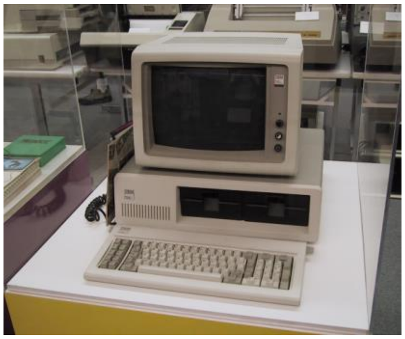


Figure 1: IBM PC Junior (Source: www.tpsoft.com)

It is not only the technology that evolved over time, the definition of desktop computers evolved as well. Up to recently, desktop computer was defined as a computer with user data and information stored on it, a screen to display information and input / output devices. Now, thanks to the mobilization of Information Technology (IT), desktops refer to whatever the user has in his hands with all of his data, configuration and settings **[1]**.

Note: The phrase “traditional desktop computers” or “fat clients” refers to physical desktop computers.

Like with other technologies, the implementation and management of desktop computers present challenges to organizations. According to VMware **[38]**, the main challenges associated with traditional desktops can be summarized in the following: management overhead, total cost of ownership (TCO), Security and inefficient utilization of resources.

In its simplest definition, desktop virtualization refers to the separation of the desktop environment (operating system, applications and user profiles) from the physical machine. This is done by hosting desktop computers as virtual machines (guests) or sessions on top of physical hosts.

Vendors promoting server virtualization are also promoting desktop virtualization. It is noticeable that the main vendors for server virtualization are also playing a major role in the desktop virtualization track. According to Gartner group **[15]**, VMware is the leader in server virtualization and is also one of the leaders in desktop virtualization. The same applies for both Citrix and Microsoft.

According to Brian Madden **[1]**:

“VMware’s success with virtualization was based on servers’ virtualizations. While this was happening, a separate group of IT Professionals were doing what is called Server-Based Computing (SBC). With SBC, instead of running applications from the local desktops, the applications run on a server centrally located and the user just sees a dynamic picture of what the application is doing. This concept of SBC was pioneered by Fort Lauderadal-based Software Company Citrix Systems in the 1990s with a product

called Citrix WinFrame. Microsoft caught wind of what Citrix was doing, and the two companies worked together to bring this SBC technology into the mainstream, creating a special Terminal Server edition of Windows NT 4.0. In the late 2000s, when Citrix and Microsoft were having a great success with SBC, VMware was enjoying great success with their server virtualization products. Around this time, it started to occur to the VMware executives, partners and investors that merely focusing on the servers in the data center was really limiting VMware's potential. If there were 50 million servers in the world that could be virtualized, the surely there must be 500 million desktops! And so the great race to virtualize desktops began".

The solution developed by VMware aims to move the resource utilization and locus of computations from the physical desktop computers to VMs hosted on a host. A VM is dedicated for each user, users connect to their VMs using remote desktop protocols. The solution developed by VMware has similarities with that developed by Citrix (SBC). The main difference is that with the VMware solution, each user has a dedicated VM. The VMware solution required more server resources to serve the same number of VMs compared to Citrix solution (SBC).

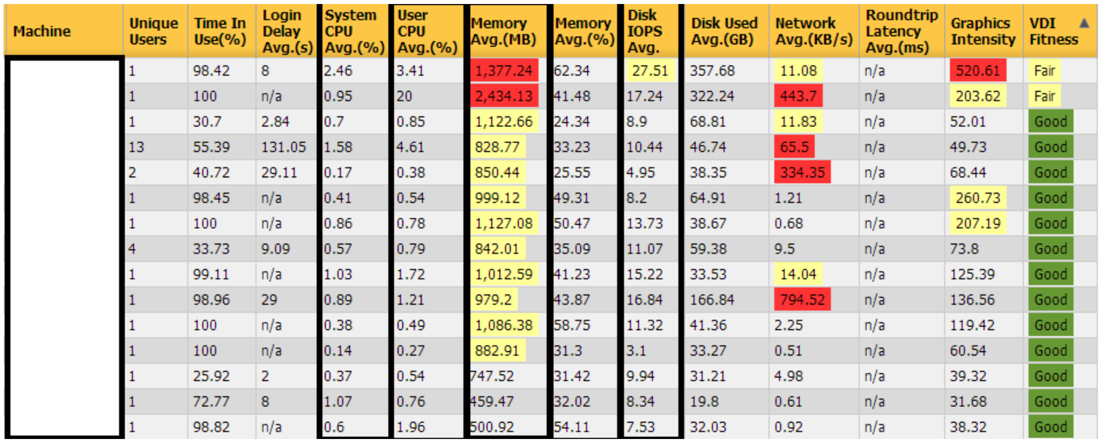
The first chapter of this research explores the challenges associated with Traditional Desktops.

Chapter1: Traditional Desktops Challenges

With the traditional deployment architecture, the execution of applications and the resource computations are done on the physical desktop computers. The traditional deployment architecture has challenges, three of the main challenges are discussed in this chapter.

1.1. Inefficient resource utilization

Desktop computers are powerful machines that are inefficiently utilized in the traditional deployment architecture of each user controlling access to their own hardware. As part of this research, a hardware and software assessment is conducted on an international company based on Dubai to better understand relevant utilization rates. Quest Desktop Virtualization assessment tool (http://www.quest.com) was run for a period of one month. *Figure 2* below shows the average resource utilization during the one month period (Desktop computers' IDs are hidden for confidentiality).



Machine	Unique Users	Time In Use(%)	Login Delay Avg.(s)	System CPU Avg.(%)	User CPU Avg.(%)	Memory Avg.(MB)	Memory Avg.(%)	Disk IOPS Avg.	Disk Used Avg.(GB)	Network Avg.(KB/s)	Roundtrip Latency Avg.(ms)	Graphics Intensity	VDI Fitness
	1	98.42	8	2.46	3.41	1,377.24	62.34	27.51	357.68	11.08	n/a	520.61	Fair
	1	100	n/a	0.95	20	2,434.13	41.48	17.24	322.24	443.7	n/a	203.62	Fair
	1	30.7	2.84	0.7	0.85	1,122.66	24.34	8.9	68.81	11.83	n/a	52.01	Good
	13	55.39	131.05	1.58	4.61	828.77	33.23	10.44	46.74	65.5	n/a	49.73	Good
	2	40.72	29.11	0.17	0.38	850.44	25.55	4.95	38.35	334.35	n/a	68.44	Good
	1	98.45	n/a	0.41	0.54	999.12	49.31	8.2	64.91	1.21	n/a	260.73	Good
	1	100	n/a	0.86	0.78	1,127.08	50.47	13.73	38.67	0.68	n/a	207.19	Good
	4	33.73	9.09	0.57	0.79	842.01	35.09	11.07	59.38	9.5	n/a	73.8	Good
	1	99.11	n/a	1.03	1.72	1,012.59	41.23	15.22	33.53	14.04	n/a	125.39	Good
	1	98.96	29	0.89	1.21	979.2	43.87	16.84	166.84	794.52	n/a	136.56	Good
	1	100	n/a	0.38	0.49	1,086.38	58.75	11.32	41.36	2.25	n/a	119.42	Good
	1	100	n/a	0.14	0.27	882.91	31.3	3.1	33.27	0.51	n/a	60.54	Good
	1	25.92	2	0.37	0.54	747.52	31.42	9.94	31.21	4.98	n/a	39.32	Good
	1	72.77	8	1.07	0.76	459.47	32.02	8.34	19.8	0.61	n/a	31.68	Good
	1	98.82	n/a	0.6	1.96	500.92	54.11	7.53	32.03	0.92	n/a	38.32	Good

Figure 2: Desktop Virtualization Assessment (Average Utilization). Tool: VDI Assessment by Quest.

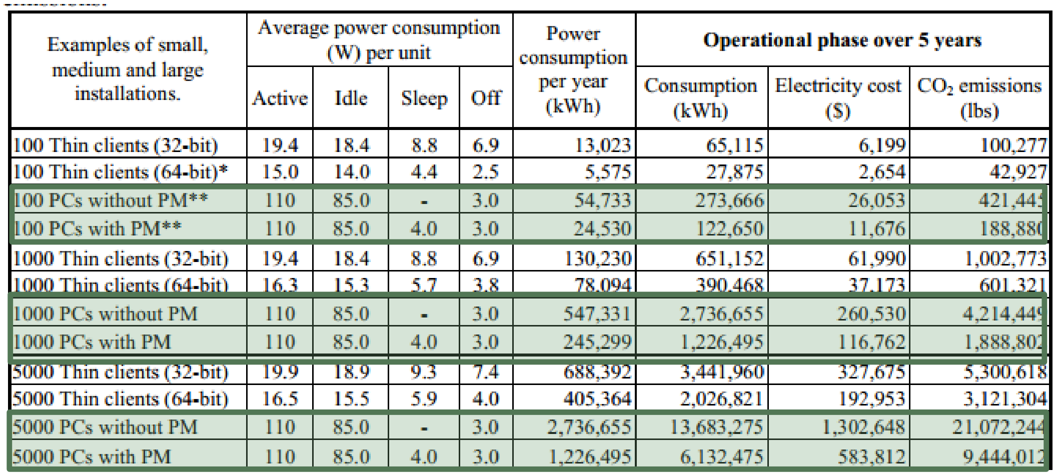
Note: more details about the conducted assessment are in the appendix section (*Appendix A: Desktop Virtualization Assessment Results*).

1.2. Power Consumption

According to Wyse Technology, the desktop traditional computing architecture is everywhere (schools, hospitals, etc...) **[2].** Desktop computers have an unseen environmental and economic impact throughout their lifecycle. Starting from their production phase and ending with their disposal. The global information and communication technology (ICT) industry accounts for approximately two percent of the global carbon dioxide (CO_2) emissions, a value equivalent to aviation according to Gartner, Inc **[40]**.

An average desktop computer with monitor requires 10 times its weight in chemicals and fossil fuels to produce **[41]**. But the main impact of desktop computers on the environment comes from their operation phase. Based on studies **[2]**, approximately 15% of organizations' energy costs and carbon footprint comes from Information Technology use, 39% of this value comes from desktop computers and servers. But it is not only the operation phase of desktop computers that effect the environment. When a desktop computer reaches the end of life (disposal phase), hazardous materials contained on it have a negative impact on the environment if not disposed of properly. Based on statistics collected from the National Recycling Coalition **[2]**, between 1997 and 2007 about 500 million personal computers were disposed of. While it is not clear what percentage of these computers were disposed of properly, the negative impact of their disposal on the environment may have been substantial.

Desktop computers consume between 85 and 110 Watts (W) **[2]**. *Figure 3* below examines the energy consumption (kWh), electricity cost ($) and CO_2 emissions for three PC installation scenarios with and without power management feature.



Examples of small, medium and large installations.	Average power consumption (W) per unit				Power consumption per year (kWh)	Operational phase over 5 years		
	Active	Idle	Sleep	Off		Consumption (kWh)	Electricity cost ($)	CO_2 emissions (lbs)
100 Thin clients (32-bit)	19.4	18.4	8.8	6.9	13,023	65,115	6,199	100,277
100 Thin clients (64-bit)*	15.0	14.0	4.4	2.5	5,575	27,875	2,654	42,927
100 PCs without PM**	110	85.0	-	3.0	54,733	273,666	26,053	421,445
100 PCs with PM**	110	85.0	4.0	3.0	24,530	122,650	11,676	188,880
1000 Thin clients (32-bit)	19.4	18.4	8.8	6.9	130,230	651,152	61,990	1,002,773
1000 Thin clients (64-bit)	16.3	15.3	5.7	3.8	78,094	390,468	37,173	601,321
1000 PCs without PM	110	85.0	-	3.0	547,331	2,736,655	260,530	4,214,449
1000 PCs with PM	110	85.0	4.0	3.0	245,299	1,226,495	116,762	1,888,802
5000 Thin clients (32-bit)	19.9	18.9	9.3	7.4	688,392	3,441,960	327,675	5,300,618
5000 Thin clients (64-bit)	16.5	15.5	5.9	4.0	405,364	2,026,821	192,953	3,121,304
5000 PCs without PM	110	85.0	-	3.0	2,736,655	13,683,275	1,302,648	21,072,244
5000 PCs with PM	110	85.0	4.0	3.0	1,226,495	6,132,475	583,812	9,444,012

Figure 3: Desktops Power Consumption, Electricity cost and CO_2 Emissions [2]

1.3. Management Overhead

Desktop computers management and administration represent a challenge in the traditional desktop deployment architecture. Some of the tasks that must be accomplished include: Desktop imaging process, updates and patches deployment, O.S migration and applications upgrade.

One way to enhance the efficiency of deployment and management include the creation of a master “Standard” image instead of manually installing OS and applications. A best practice is to use a reference desktop computer with all the standard applications and OS installed, and updated with the latest patches and drivers. The master image is captured from the reference desktop computer and saved on a network shared drive. Targeted desktop computers boot from the network using Pre-boot Execution Environment (PXE) to get the image from the network shared drive. The standard image has to be updated frequently and injected with the latest drivers. The process of “Imaging” and “Installing” can turn into a complex task especially for enterprise companies with a large number of desktop computers, and for applications that need to be updated frequently (Example: Microsoft Applications). *Figure 4* below shows the components of the imaging process.

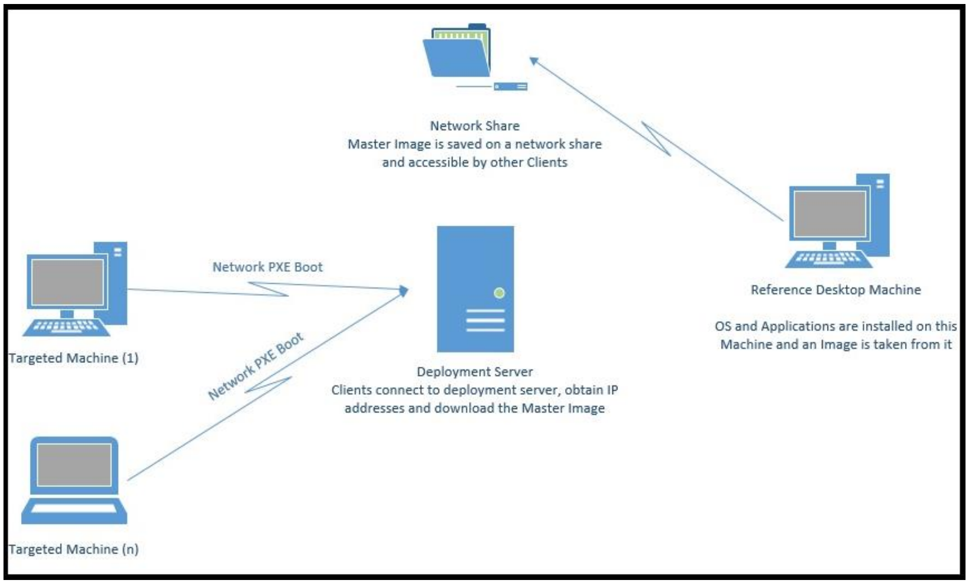

Network Share
Master Image is saved on a network share and accessible by other Clients
Reference Desktop Machine
OS and Applications are installed on this Machine and an Image is taken from it
Network PXE Boot
Targeted Machine (1)
Network PXE Boot
Deployment Server
Clients connect to deployment server, obtain IP addresses and download the Master Image
Targeted Machine (n)

Figure 4: Desktop Imaging Process

According to a report issued by Microsoft **[3]**, about 24% of computers scanned by Microsoft tools were not running real-time antimalware software or were running out-of-date antimalware software. According to BullGuard Security Centre **[42]**, only five percent of users scanned were running fully-patched Windows operating system. One reason behind this problem is the large number of desktop computers to manage. Many mid-size and enterprise companies have thousands of desktop computers to manage. While there are tools that aid in the automation and management of desktop computers tasks including patching and updating tools. Tools like Windows Server Update Service

(WSUS), a Microsoft proprietary tool to manage the deployment of windows updates from a central location. *WSUS* has a server component which connects to Microsoft update website, downloads relevant updates and distributes them to the clients based on a predefined schedule. Other tools like ManageEngine patch management tool, which works with Microsoft updates and third party updates, unlike WSUS which works only with Microsoft updates. Those tools help in automating and easing the process of patch and update deployment, but the process remains very complex and prove to failure leaving the unpatched desktop computers open for attack. Before deploying patches and updates using the patch automation tools, updates have to be tested on test environments to make sure that the newly deployed updates will not result in a downtime or interruption.

OS migration is another challenge for traditional desktop computers. Like patch deployment, there are different tools for lite-touch deployment (*deploying OS with minimal intervention from systems administrators*) and zero-touch deployment (*deploying OS with no intervention from systems administrators*). Windows Deployment Services (WDS) and Microsoft SCCM (System Center Configuration Manager) are two of the leading tools for OS deployments. Such tools help ease the deployment and migration of OS. On the other hand, while trying to make OS deployment easier, another challenge is created. Deployment systems like WDS and SCCM can turn into a complicated task that requires a huge effort to maintain and operate.

There are other challenges for traditional desktop computers like cost, but it is not a main challenge as the technology gets cheaper day after day. Another challenge is mobility, traditional desktop computers are machines that cannot mobile with the users.

Next chapter of this research, server-based desktop virtualization, is the core chapter of this research. Throughout this chapter, two of the main Server-based desktop virtualization technologies will be assessed and compared.

Chapter2: Server-Based Desktops Virtualization

2.1. Server Architectures and Locus of Computations

Server-based desktop virtualization is accomplished in different architectures. This paper will discuss the two main approaches of server-based desktop virtualization which are: virtual desktop infrastructure (VDI) and session-based desktop virtualization (SBDV).

The server-based desktop virtualization technology is evolving, and the competition between different vendors is drastically increasing, the result of this competition is the development of different architectures for server-based desktop virtualization. This research finds it crucial to have general models illustrating the server side architecture and the locus of computations. The research develops three main models:

1. Model A: persistent VM.
2. Model B: non-persistent VM.
3. Model C: session based.

All vendors' architectures fit in one or more of those general architectures. VDI can be implemented by model A and model B, while SBDV is implemented using model C. The three models are explained in the following paragraphs.

In model A, each user has a dedicated VM hosted on the host server. VMs have client OS installed on them to provide the desired user experience. The resource computations are by default performed on the host server. Thanks to the 64-bit architecture, the ratio of clients to servers is increased, resulting in more number of clients (VMs) per server. But there is one resource which is not evolving the same way that other resources do, this resource is the network bandwidth. Vendors and researchers are eager to find solutions to overcome the network challenge. One of the solutions is to perform in the graphics and multimedia rendering process on the client side.

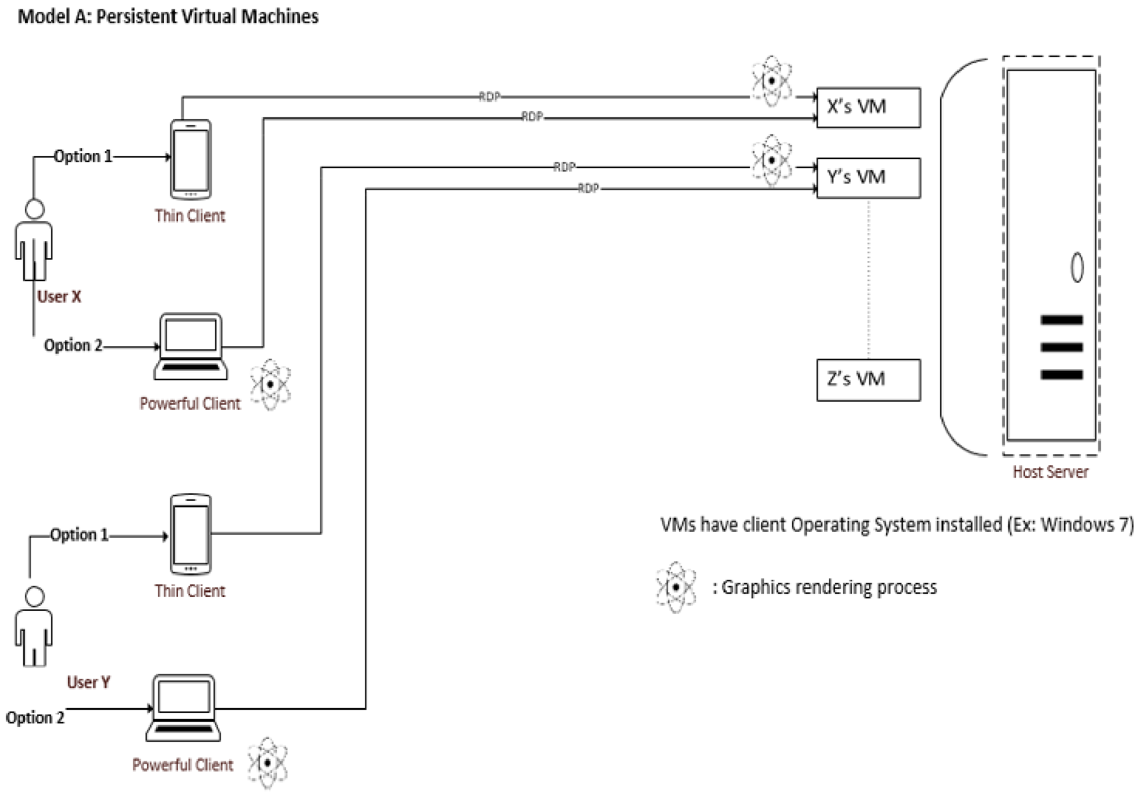


Figure 5: Model A (persistent VMs).

As shown in the *figure* above (model A), users can connect to their dedicated VMs using a thin client or a fat client. If a thin client is used, hence thin clients don't have powerful graphics adapter to perform the rendering process, the rendering process can be accomplished on the server side (server-side rendering). If a fat client is used, then the rendering process can be accomplished on the client sider (client-side rendering).

In model B, both client-side and server-side rendering are achievable depending on the client and technology used. With model B, VMs are shared between users. Each time the user connect to the server, a new VM is created from the VM resource pool. The creation process can be either on-fly or static depending on the technology used. For instance, Microsoft VDI supports static type of creation, while integrating Microsoft VDI with other products like Microsoft System Center allows on-fly creation of VMs. Model B architecture is illustrated on the *Figure* below (model B).

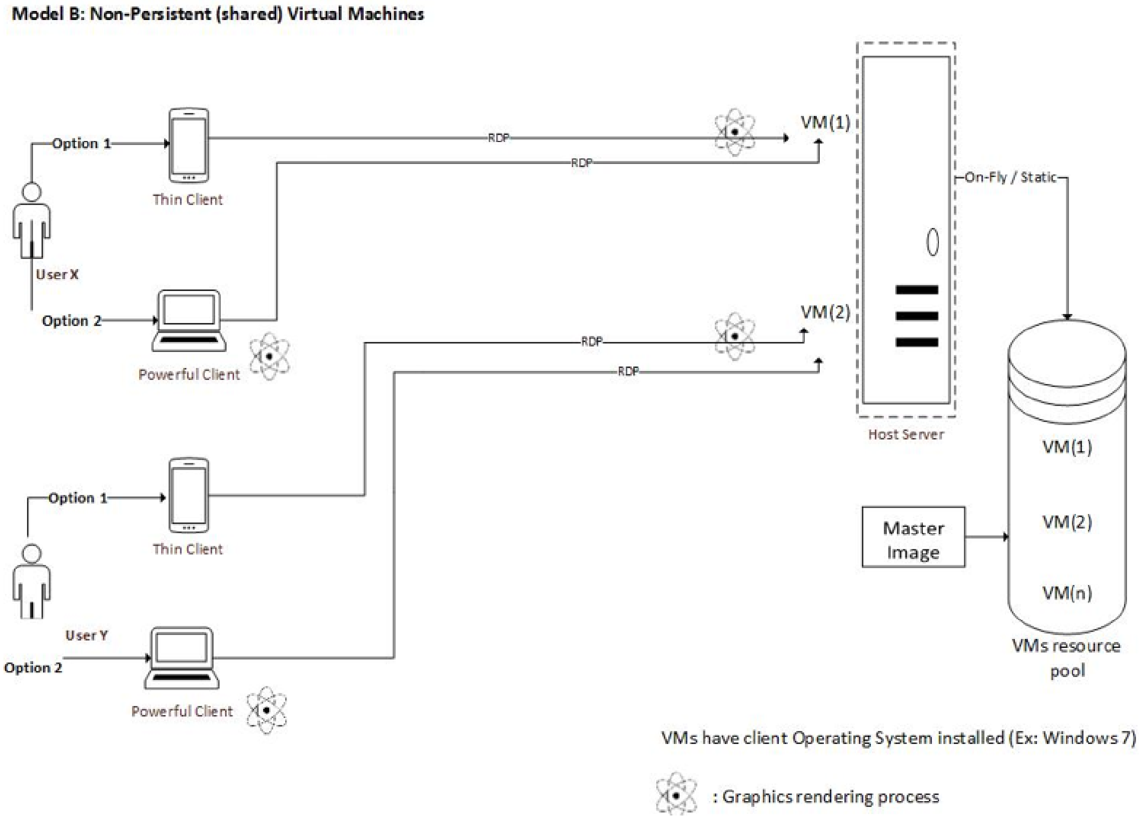


Figure 6: Model B (non-persistent VMs).

In model C, VMs are replaced with sessions. Sessions are created from a server version of OS which can be installed on a physical or a virtual machine. As with both model A and model B, both client-side rendering and server-side rendering are achievable depending on the client and technology used. Model C architecture is illustrate on the *Figure* below (model C).

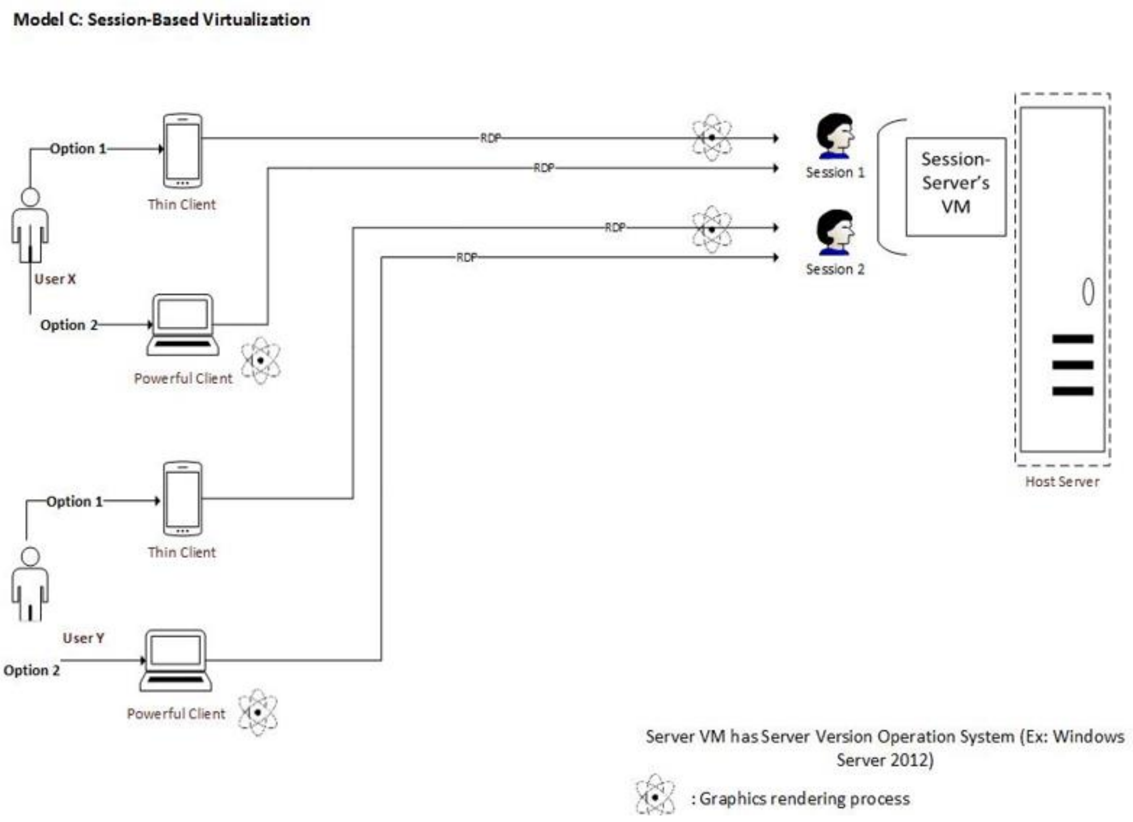


Figure 7: Model C (session-based).

The next two sections of this chapter explores the two main approaches of session-based desktop virtualization: VDI and SBDV.

2.2. Virtual Desktop Infrastructure (VDI)

Desktop virtualization consists of six components: OS virtualization, session virtualization, application virtualization, connection broker, client access devices and user data and profiles. The research focuses on three of the six components: OS virtualization, session virtualization and connection broker. Although client access devices (CADs) are an essential component of the desktop virtualization, the research will only briefly discuss the client access devices. Other components of desktop virtualization (application virtualization and user data and profiles) are outside of this research scope.

2.2.1. VDI Definition

VDI refers to the virtualized desktop computers running on top of a hypervisor. VMs can be either distributed where each user has one desktop VM (guest) or it can be a pool of VMs where VMs are shared between users, users get random VMs from the pool upon establishing the connection. Users access the VMs using Remote Protocols (RP) like Microsoft Remote Desktop Protocol (RDP), Citrix High Definition Experience (HDX)) or VMware Personal Computing over Internet Protocol (PCoIP).

2.2.2. VDI Components

The main components of a VDI architecture are:

- A protocol used to connect the users to the guest VMs. Previously, the protocol was called “the display protocol” since it was mainly used to handle the updating of the display, passing the keyboard and mouse input. The display protocols evolved and is renamed to desktop remoting protocols (DRPs). The function of DRPs is more than just the displaying part. Remoting protocols handle different functions including the following:
  - Full keyboard, mouse and touch redirection from the local client to the guest VMs.
  - Support for multi monitors.
  - Multimedia redirection. This feature allows multimedia files to be rendered on the client side (this feature is explored in details in chapter 3 & 4).
  - Bi-directional audio, which allows audio to be sent to and from the local client.
- A virtual management platform. This platform manages the servers hosting the guest VMs, it also helps in provisioning and managing the clients’ VMs.
- A connection (session) broker. It is responsible for load balancing the traffic from the clients, it is also the connection point for clients connecting from the local network.
- Application virtualization. It enables fast access to applications from end user devices. VDI allows the manual installation of applications, but with application

virtualization the deployment is faster (no local installation of application, virtualized application uses the server computational resources).

Note: This component is out of this research scope, it is mentioned since it is a main component of VDI.

- Profile and data redirection: with traditional desktop computers, users customize their desktops (background, documents, settings, etc.) and have these customizations preserved whenever they login to their machines. With VDI, users either have their persistent VMs or non-persistent VMs. With persistent VMs, users have their dedicated VMs. Like with traditional desktop computers, persistent VMs allow users to customize their desktops and have their customization available whenever they login to their VMs. But with non-persistent VMs, users connect to random VMs from a pool of VMs, which brings a new challenge for VDI, which is to save the users customization and redirect their data whenever they connect. The profile and data redirection component of VDI is used to ensure that the customization is preserved for non-persistent type of VDI.
- VDI gateway: It is used to receive requests from clients connecting from internet. For security purposes, the connection between the remote clients and the VDI gateway is encrypted, VDI gateway converts the encrypted traffic to a format that can be read by other components of VDI.

2.2.3. VDI Architecture

Figure (8) below illustrates the basic architecture for VDI.

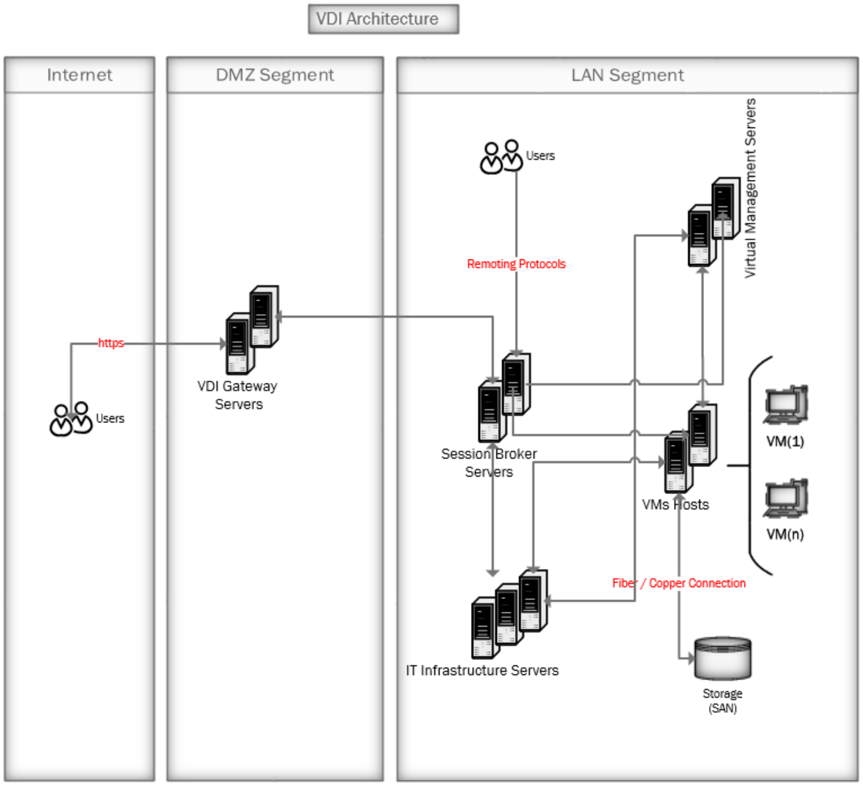


Figure 8: VDI Architecture

The architecture divides VDI components into three segments: Local area network (LAN), demilitarized zone (DMZ) and the internet. The VDI gateway servers are hosted in the DMZ zone to provide access for remote users. VDI gateway is an optional component which is used if remote access is required. For both remote users and internal users the remote desktop client (RDC) is required to establish the connection with the VDI gateway or the connection broker servers. The RDC contains information about the server to which the users connects, information includes: Server IP address, server name, screen resolution settings, connected devices and connection quality.

For remote clients, the connection with the VDI gateway server is encrypted. VDI gateway encapsulates the RDP traffic for security purposes.

The remote client initiates the connection by executing the RDC file, RDC connects to the VDI gateway. The VDI gateway receives the traffic from the remote clients then it extracts the RDP traffic and forwards it to the connection broker server. The connection broker consults the infrastructure servers and retrieves the user profile type (persistent or non-persistent VMs).

The connection broker communicates with the virtual management servers. If the type of the VM is persistent (determined from the user's parameters), then the connection broker connects to the user's VM hosted on the VMs Host. The connection broker passes the information to the VDI gateway, which encapsulates the traffic then sends it back to the remote client. If the type of VM is non-persistent, then the connection broker consults the virtual management server. The virtual management server informs the host and gets a random VM from the VMs pool. If the VMs pool has no VMs available, then the virtual management server instructs the host to create a VM from the master copy.

The same scenario applies with the internal users except that they connect directly to the connection broker instead of the VDI gateway. The VMs virtual hard disks (VHDs) are hosted on a storage area network (SAN) to achieve higher performance.

2.2.4. Microsoft VDI

Microsoft VDI shares the same components of the VDI architecture illustrated in Figure 8 above. But Microsoft has additional components added to its VDI: Microsoft licensing server and remote desktop (RD) web access. Microsoft VDI components are: RD Gateway, RD web access, RD session host, RD connection broker, licensing server and hyper-V servers to host the VMs. There are other optional components that help managing VDI infrastructure like: Microsoft System Center Virtual Machine Manager (SCVMM), Microsoft System Center Operations Manager (SCOM) and Microsoft System Center Configuration Manager (SCCM).

SCVMM is a management solution for the virtualized datacenter, it enables companies to configure and manage virtualization host, networking, and storage **[43]**. SCCM is used to provide more effective IT services by enabling secure and scalable software and applications deployment, compliance settings management, and comprehensive asset management for servers and desktops **[44]**. SCOM provides deep visibility into the health, performance and availability across applications, OSs, hypervisors and hardware **[45]**.

Figure 9 below illustrates Microsoft VDI architecture **[8]**.

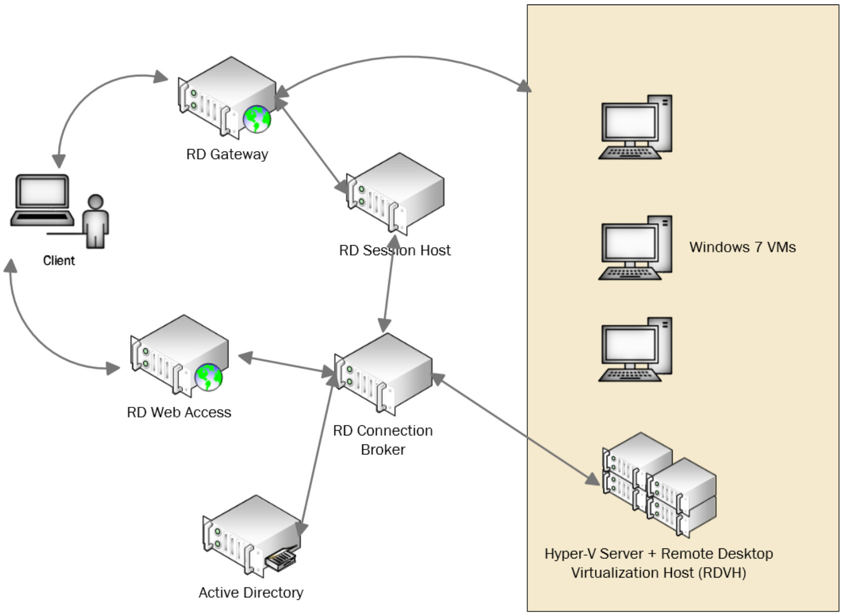


Figure 9: Microsoft VDI Architecture. [8]

Out of the box, Microsoft provides static type of VDI. This means that virtual machines must be created prior to the assignment of VMs for the users. To have dynamic type of VDI, other management tools like SCVMM are required. As per Microsoft recommendations, a Master image should be created and other VMs are cloned from the master image. When the master image is installed and configured, a tool called “sysprep” is used. Sysprep is used to prepare VM for duplication by removing unique installation information like security identifiers (SIDs). The master image holds the OS with the required applications and customization. If application virtualization is integrated with the VDI, then the applications are not installed locally on the VMs, instead they are hosted on the application virtualization servers). The average size for the master image is (8-10 G.B) **[11]**. The storage requirements for VDI can turn into a challenge if the number of VMs is high. In addition to the storage requirements, disk input and output operations (IOPS) may turn into a bottleneck for VDI. Consider having dozens of VMs hosted on a physical server where all VMs are accessed at the same time. Even if the VMs are hosted on a SAN, this will cause a lot of load on the Disks. Those challenges and others are discussed on Chapter 3 “Server-Based Desktop Virtualization Challenges and Recommendations”.

The licensing part for Microsoft VDI is complicated. At the time of writing this research, Microsoft licensing for VDI works as follow **[12] [1]**:

- Software assurance (SA) and virtual desktop access (VDA): if SA is not available, then VDA is required. VDA is a yearly subscription of 100 $ per device. Unlike SA, VDA doesn’t allow free upgrade from one version to another.
- Even if SA or VDA is available, end users are not allowed to access their VMs from thin clients without another SA or VDA. An exception for this is for home users’ scenarios where extended roaming rights (ERR) is used. ERR allows users to access their VMs without additional SA or VDA as long as their primary device at the office has SA or VDA. If a user is using a tablet at home to access his office VM then no additional license is required, but if the user brings the tablet device to the office then VDA or SA is required.

Microsoft RDP is a multi-channel protocol that allows users of a remote client to connect to a server or a VM over a network. This multi-channel capability enables the use of separate channels, called virtual channels, to carry different types of data including graphics, keyboard and mouse inputs, device communication, and file system, audio and video and licensing information. RDP is used to initialize connections, negotiate capabilities (example: security) and transfer input between the RDC and the server **[14]**.

RDP facilitates the user interaction with the VM by transferring graphics display information from the VM to the local client, as well, transferring and encrypting inputs from the local client to the VM. The following *figure* shows the connection sequence for RDP **[13]**.

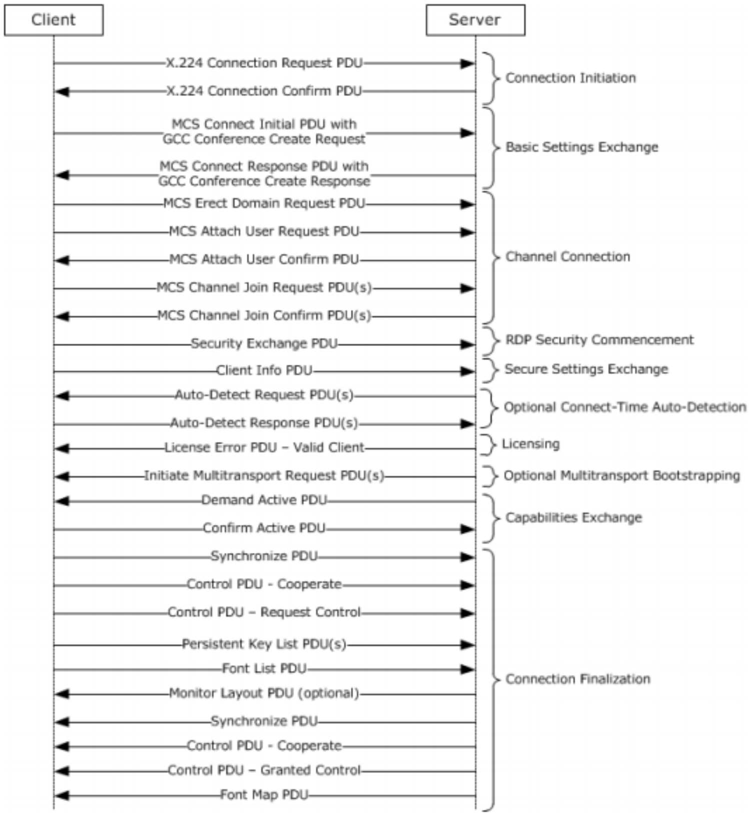

Client
Server
X.224 Connection Request PDU
X.224 Connection Confirm PDU
Connection Initiation
MCS Connect Initial PDU with GCC Conference Create Request
MCS Connect Response PDU with GCC Conference Create Response
Basic Settings Exchange
MCS Erect Domain Request PDU
MCS Attach User Request PDU
MCS Attach User Confirm PDU
MCS Channel Join Request PDU(s)
MCS Channel Join Confirm PDU(s)
Channel Connection
Security Exchange PDU
RDP Security Commencement
Client Info PDU
Secure Settings Exchange
Auto-Detect Request PDU(s)
Auto-Detect Response PDU(s)
Optional Connect-Time Auto-Detection
License Error PDU – Valid Client
Licensing
Initiate Multitransport Request PDU(s)
Optional Multitransport Bootstrapping
Demand Active PDU
Confirm Active PDU
Capabilities Exchange
Synchronize PDU
Control PDU - Cooperate
Control PDU – Request Control
Persistent Key List PDU(s)
Font List PDU
Monitor Layout PDU (optional)
Synchronize PDU
Control PDU - Cooperate
Control PDU – Granted Control
Font Map PDU
Connection Finalization

Figure 10: RDP connection sequence [13]

On the server side, Microsoft RDP uses its own video driver to render the display output. The output is transformed into network packets and sent over the network to the remote client. RDC on the remote client receives the packets and transform them into Microsoft

Windows graphics devices interface (GDI). For input devices like mouse and keyboard, inputs are redirected from the client to the server. On the server, RDP uses its own on-screen keyboard and mouse drivers to receive the inputs data **[13]**.

Microsoft has enhanced its RDP protocol starting from the first version of RDP, version four. Version four is part of Windows NT 4.0 terminal server edition. RDP version 4 was mainly focused on connecting multi login sessions to the server. RDP has evolved over time, the latest version of RDP, known as RDP 8.1 which is part of Windows 8.1 and Windows server 2012 R2.

<u>Note</u>: RDP 8.1 is the latest version of RDP released by Microsoft at the time of writing this research.

Prior to RDP 8.1, RDP used to work well with LAN (10 Mbps and above). But when it comes to WAN, RDP isn't performing well. Different versions of RDP protocol (RDP 7 and RDP 8) are tested and assessed as part of this research, test results and scenarios are illustrated in *Chapter 4 "Lab Experiments and Capacity Planning".*

Prior to RDP 8, graphics rendering is performed using the graphics driver of RDP on the server side (software-based rendering). Software-based rendering is poor in performance compared to hardware-based rendering. With RDP 8 (supported in Windows 8 and Windows server 2012), Microsoft introduced a new technology known as RemoteFX. It aims to provide a better experience and performance for remote devices. The first version of RemoteFX (released with Windows 2008) supports LAN scenarios only, while with the second release (Windows 2012), Microsoft RemoteFX supports WAN scenarios.

Note: WAN scenario for RemoteFX is tested, results are illustrated on *Chapter 4 "Lab Experiments and Capacity Planning"*.

RemoteFX consists of three main components: graphical processor unit (GPU) virtualization, enhanced codec and USB port-level redirection. GPU is a graphics card processor designed to perform complex mathematical calculations that are required for graphics rendering. With GPU virtualization, a physical GPU can be virtualized and assigned to multiple VMs. With Microsoft VDI, one physical GPU can be assigned to twelve virtual GPUs. Graphics card are usually equipped with one or two GPUs, which provides up to twenty four virtual GPUs. As a result, one physical server can provide GPU capabilities to twenty four VMs. As illustrated earlier, Microsoft RDP renders graphics on the server side. The integration with physical GPU increases the performance for the rendering process. The challenge is that most of the servers don't have powerful graphics card. To overcome this challenge, Microsoft introduced a software rasterizer that allows using RemoteFX for environments that don't have physical GPUs.

RemoteFX introduces enhanced codec that is designed to improve the encoding and decoding of remote display. A major improvement on RemoteFX is the RemoteFX USB redirection. Prior to RemoteFX, RDP supports basic redirection such as keyboard, mouse, disk and other devices. RDP basic redirection is done by redirecting devices to one of the supported RDP redirection devices types. As a result, some of the device-specific functionalities are not functional with RDP.

RemoteFX is not a replacement for the basic RDP redirection used with the RDP versions prior to RemoteFX. It is used as a supplementary to support devices that are not compatible with basic RDP redirection. At the time of writing this research, RemoteFX is not widely adopted due to the challenges it has. With RemoteFX, optimization cannot be achieved since the redirection is done at the port level. In order to use RemoteFX, the driver for the devices must be installed on the host. RemoteFX USB redirection cannot be used for WAN Scenarios, and the major challenge for RemoteFX is that only one session from the remote client can access the redirected device at a time, that includes the local client is self.

RDPs are a crucial components for any VDI design. Vendors and researchers focus their efforts to enhance the RDPs. With traditional desktop computers, the desktop computer case is connected to the monitor by a video graphics adapter (VGA) or digital visual interface (DVI) cable. With VDI, the network media between the host and remote clients is equivalent to VGA or DVI cable of the traditional desktop computers.

More details about RDPs including Microsoft RDP & RemoteFX are in *Chapter3 "Desktop Virtualization Challenges and Recommendations"*.

2.2.5. VMware VDI

Figure 12 below shows the high-level architecture of VMware VDI **[17]**.

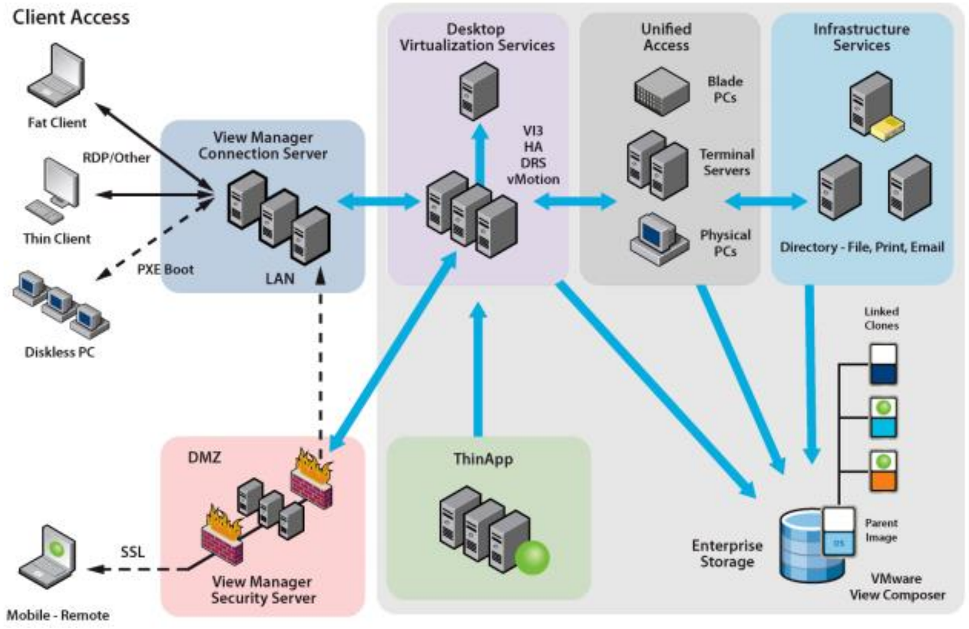


Figure 12: VMware View high-level architecture [17].

The VMware VDI solution is VMware View **[17]**. The first component of VMware View is the CADs. This component includes the clients used to access the VMs hosted on VMware hypervisor (VMware has its owned hypervisor known as VMware vSphere). CADs can be either fat, thin, repurposed desktop computers or mobile devices. With repurposed desktop computers, companies lock down the OS installed and allow only the RDC (VMware View Client) to run from the desktop computer. Another approach for using repurposed desktop computers with VDI is by network boot (PXE boot), where clients load a light version of Linux OS or VMware View web access to access the VMs.

This approach (network boot) is preferable since no OS is installed on the client side resulting in reduced management overhead and enhanced security.

Access infrastructure is the second component of VMware View VDI solution. Access infrastructure aims to provide access from remote clients to the hosted VMs. LAN, WAN and VMware View Manager (connection broker) are part of the Access Infrastructure layer.

Virtual infrastructure is the third component. It defines the technologies used to host the VMs. The fourth component is View desktops which has the configuration of the VMs accessed by remote clients. Session management is the last component of VMware View. This component plays the role of deployment and management for VMs, and the integration with the infrastructure services like AD. Desktop provisioning, pool management, session monitoring and virtual printing are parts of the session management component.

Personal Computer over Internet Protocol (PCoIP) is the RDP used by VMware. PCoIP was developed by Teradici **[18]**. PCoIP uses UDP instead of TCP to deliver bitmaps by encoding them on the host server then streaming them over the network to the remote clients. It transmits only regions of the screen that change from frame to the next frame in order to optimize the bandwidth utilization.

From an architectural standpoint, PCoIP is more like Microsoft RemoteFX. Both of them are designed for general purposes. But unlike RemoteFX, PCoIP doesn’t support GPUs on the server side which means that PCoIP doesn’t work with applications that require GPU on the server side.

2.2.6. Citrix VDI

XenDesktop is the VDI solution from Citrix. The core components for XenDesktop are: Desktop Delivery Controller (DDC), Provisioning Servers (PVS), Virtual Desktop Agent (VDA) and Secure Delivery and Desktop Receiver **[19]**. The architecture is illustrated in *Figure 13* below.

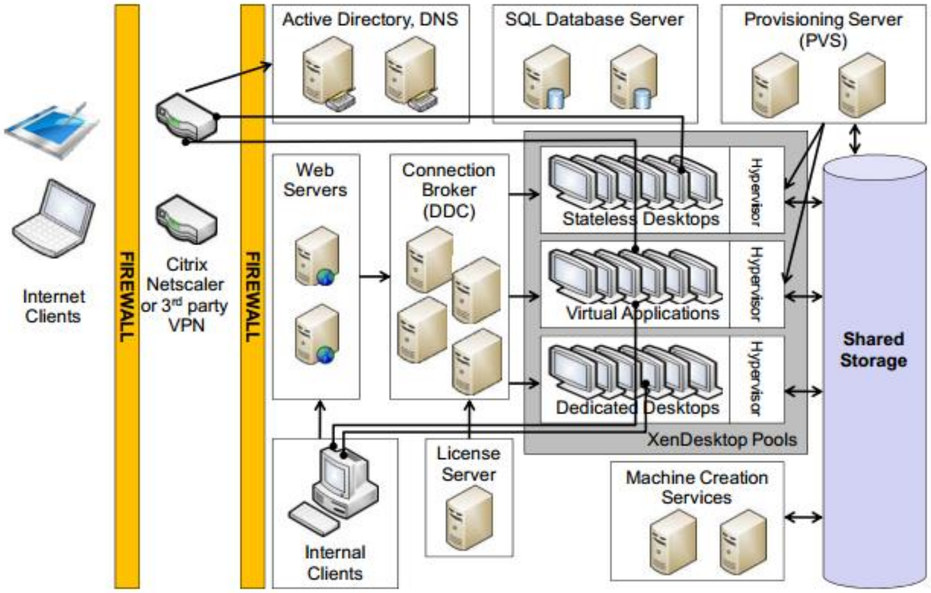


Figure 13: Citrix XenDesktop Architecture [19].

DDC is the infrastructure component installed on servers in order to authenticate users and manage virtual desktop environment. The connection broker is part of the DDC component. VDA component resides on VMs that are hosted on the physical servers and is used to establish connection between the VMs and the remote clients using Citrix RDP (High Definition Experience Protocol (HDX)). HDX is the new version of Independent Computing Architecture (ICA) protocol.

The desktop receiver component is the RDC for Citrix. It is installed on the remote clients to initiate the connection with the VMs. To host the VMs, Citrix has its own hypervisor (XenServer), Citrix also supports other hypervisors like Microsoft hyper-V and VMware vSphere ESX (i) hypervisors. PVS are based on software-streaming technology (PVS is used for OS streaming virtualization, a client-based desktop virtualization that is outside the research scope).

HDX is built on top of ICA, HDX uses both TCP and UDP. HDX uses three principals to optimize and deliver applications over network with best effort. Intelligent redirection, adaptive compression and data de-duplication are the three principals used by HDX. With intelligent redirection, activities like screen updates, applications, and object rendering are monitored to determine where to redirect the computational part of the rendering process. HDX supports both client-side and server-side rendering, so if the rendering requires a high percentage of the available bandwidth, then HDX instructs the rendering process to be done on the client side if the client supports rendering (in this case, the client should have a powerful VGA to support the rendering process. This is against the VDI strategy of replacing powerful computers with thin clients). Client-side rendering is performed by sending few commands from the server side to the remote client instead of sending the whole rendered data, this saves bandwidth and delivers multimedia objects with high quality. One of the great features with HDX intelligent redirection is that devices like webcam, printers and scanners can be terminated locally on the remote client instead of redirecting them from the server. The local termination allows the interaction with the devices at the native USB port speed and reduces the load on the network. *Figure 14* below shows how the intelligent redirection of HDX protocol works **[20]**.

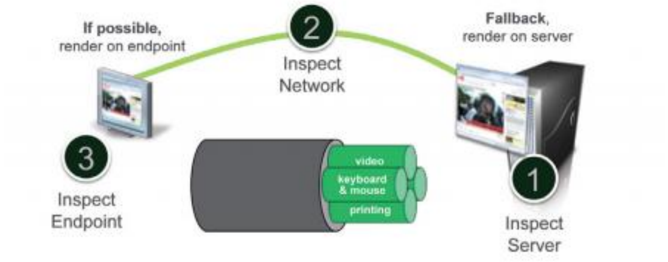


Figure 14: HDX Intelligent Redirection Principal [20].

Adaptive Compression is a core intellectual property of the HDX protocol. It is used to set and decide which codecs to use based on different network conditions and also optimize the utilization of CPU and GPU resources. Data de-duplication is implemented with the help of multicasting and caching techniques. HDX supports multicasting of multimedia in one-to-many scenarios where multiple remote clients request the same multimedia file from the server. In this case, HDX sends only one copy of the requested file to the remote clients' site (recommended for WAN scenarios). HDX caching duplicates commonly access data (print jobs, images and video files) **[20]**.

More details about the three RDPs (HDX, RemoteFX and PCoIP) are available in the next chapter (*Chapter3 – Server-based Desktop Virtualization Challenges and Recommendations*). The next section of this chapter explores the second server-based desktop virtualization approach (session-based desktop virtualization).

2.3. Session-Based Desktop Virtualization

Session virtualization, presentation virtualization, terminal services and remote desktop session host (RDSH) refer to the session-based desktop virtualization (SBDV). SBDV is the oldest server-based desktop virtualization approach but still the most used with more than 100 million users worldwide [1].

2.3.1. Definition

Session is defined as a list of processes that provides services to a logical entity like a logged in user [8]. Both VDI and SBDV aim to place user sessions / VMs on a single server or few number of servers. As well, both solutions use the same RDPs. The difference between VDI and SBDV is on the architecture.

Session virtualization is similar to multiple users sharing the same room where each user has his / her own desk (session), but since they are sharing the same room they are not allowed to talk loudly or change anything in the office layout. But at the same time they can customize their own desk. While with VDI, it is similar to multiple rooms or offices where every employee has his/ her own room.

2.3.2. Architecture

With SBDV, virtual management server is not required as virtualization is done on the sessions not on the OS itself. SBDV allows multiple users to connect and receive sessions from the session host servers, resulting in more users connecting to the same server compared to VDI. Due to that, SBDV is considered much cheaper and easier to implement compared with VDI.

A high-level architecture of SBDV is illustrated in *Figure 15* below.

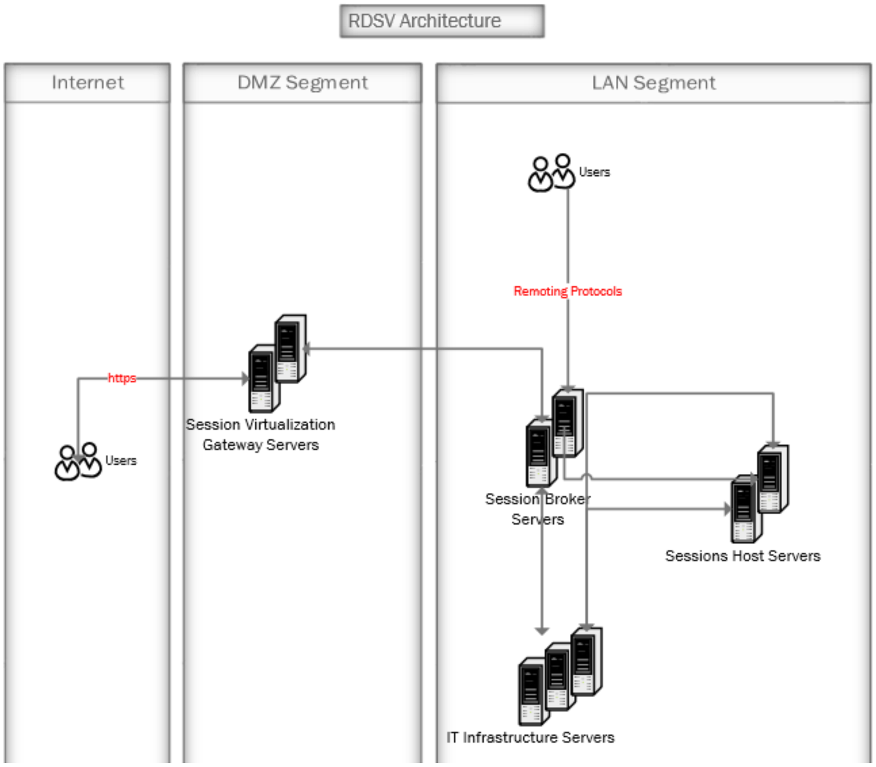


Figure 15: SBDV Architecture.

The function of sessions host server is to host the user' sessions. Users connect to the session host and get their sessions with the required applications. For the applications to function, they should be installed on the session host server. Since session host server runs a server version of the OS, application compatibility issues are possible. Challenges of SBDV are elaborated in *Chapter 3* of this research.

The following two sections of this chapter explores two of the main SBDV technologies, Microsoft session-based desktop virtualization (remote desktop services – RDS) and Linux terminal server project (LTSP).

2.3.3. Microsoft RDS

Microsoft terminal services (TS) is the foundation for different vendors and technologies. For instance, Citrix built its protocol on top of Microsoft TS and added more functionality like applications publishing. TS evolved since its development (in 1998). Microsoft RDS (previously known as TS) is a full solution for session virtualization. *Figure 16* below shows Microsoft RDS architecture **[8]**.

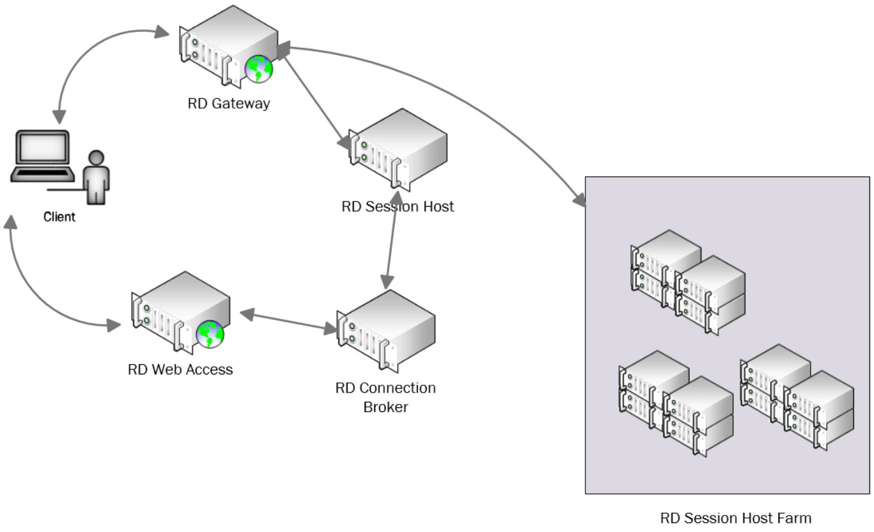


Figure 16: Microsoft RDS Architecture [8].

In Microsoft SBDV approach, security and protection are guaranteed by isolating users' sessions. For each session, the majority of the running processes and services are owned by the login user. Only two of the processes running under the user session are owned by the system, winlogon.exe and csrss.exe. Those services are created for each user session but maintained under system as they are too critical to run under the end users' credentials.

Microsoft enhanced its session virtualization solution since TS. One of the enhancements is the application virtualization (RemoteApp), a feature that allows publishing specific applications instead of the full desktop.

Another enhancement is the driverless printing. Prior to this feature, printing while using Microsoft session virtualization was a big challenge as it was required to install local printer drivers on the session host server. In addition to that, print jobs were rendered on the server side, causing high utilization on both the server resources and network bandwidth. Microsoft solved this problem by rendering the print jobs using extensible markup language (XML) paper specifications (XPS) file, then sending the print jobs using Microsoft RDP to the remote client where they get printed using the local printer of the remote client **[8]**.

2.3.4. Linux Terminal Server Project (LTSP)

LTSP is an open source project. It was born out of the need to have a terminal that could communicate using TCP/IP protocol between IBM AS/400 and UNIX application server. In addition to allowing the communication between the two servers, the terminal should allow users to perform their daily job like accessing email, documents editing and browsing the internet **[22]**.

According to Jim McQuillan, the author of The Linux Terminal Server Project: Thin clients and Linux article, LTSP is considered a project because it came out to the public as a result of a project implemented to address a customer needs. To respond to the customer needs of having a terminal to communicate between IBM AS/400 and The

UNIX Server, the decision was made to use a diskless workstations (diskless workstation is a workstation that doesn’t require a CD-ROM, hard disk or floppy for booting purposes) running Linux kernel and X-Windows (X-Window is the RDP used by Linux terminal server and clients). This is how the project started. The first implementation contained a server and 11 workstations. (The latest update is on June 2000 where the customer had more than 100 workstations using the LTSP). As a result of this successful and low-cost project, the team decided to share their solution with the rest of the world.

The LTSP is combined of four core components: clients (thin clients, diskless clients or traditional desktop computers), a dedicated switch backplane, a Linux server running the LTSP package and X-Window protocol to display the sessions on the client side. Thin clients are preferred over traditional desktop computers for two mean reasons. The first reason is that thin clients are cheaper than traditional desktops computers and have a longer life time (*for more details about thin clients, refer to Appendix D – Thin Clients Overview)*. The second reason is that thin clients provide more security and easier management.

LTSP is architected based on the network size. For small networks, thin clients, LTSP and other computers can share the same network. While all components can share the same network for small-size networks, for large networks or networks with bandwidth intensive applications, a separate network subnet should be created for the LTSP. On this subnet, thin clients and the session host are connected to a dedicated switch (switch port speed is decided based on capacity planning). Other desktop computers (Desktop computers that are not part of LTSP) are connected to another subnet and network switch that can be shared with other network resources. *Figure 17 & 18* illustrates the high-level architectures for both approaches **[23]**.

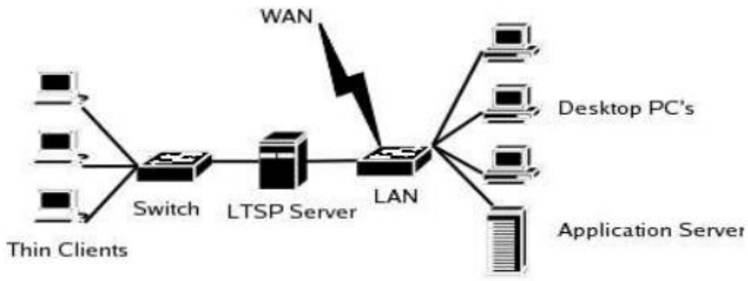


Figure 17: Separate LTSP Sub network [23]

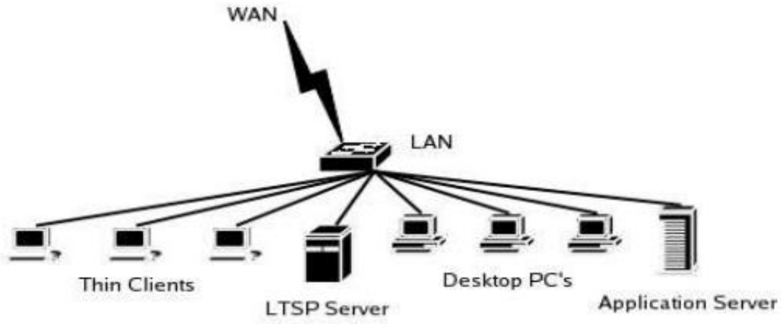


Figure 18: Shared LTSP Sub Network [23]

Thin client boots from the network using PXE and requests an IP address from the DHCP server. The DHCP server passes additional information to the thin client and downloads a Linux initramfs file system (initramfs is a complete set of compressed directories, initramfs is used to mount a root file system). The Linux initramfs is downloaded via TFTP into the memory of the thin client. After that, the thin client connects to the LTSP Server's X session. When the connection is established with the server, all operations and computations are done on the server side and displayed into the client monitor using the X-window protocol.

The X window system, known as X, is an open source client-server system used to manage the graphical user interface (GUI) for single or multiple computers **[24]**. X is separated from the OS, which means that if GUI is not required, then installing the X system is not needed, resulting in a better performance and a higher security.

The next chapter deals with some of the main challenges of server-based desktop virtualization and provides recommendations to overcome those challenges.

Chapter3:

Challenges of Server-Based Desktop Virtualization & Recommendations

The objective of server-based desktop virtualization is to move the locus of computations from the end users' desktop computers to the servers in order to reduce the cost, enhance the security and efficiently utilize the available resources. There are different challenges associated with server-based desktop virtualization. One of the main challenges is that the administration and management overhead is moved from the desktop computers to the servers, another challenge is to have a proper capacity planning for the server's resources and network bandwidth to accommodate the required computations. There are other challenges that are specific to VDI and SBDV. The main four technical challenges and the recommendations for those challenges are elaborated in the following sections of this chapter.

3.1. Administration and Management Overhead

Moving desktop computers from the client side to the servers reduces the administration overhead from the client side but increases it on datacenter. In order to manage the hosted VMs / sessions, the three elements of administration (people, process and technology) should be on place.

With traditional environments, the support team members are categorized on three levels: level1, level2 and level3. Level1 is the tier with majority of staff, staff in level1 are entry level staff. Level3 is the "expert tier" with less number of staff. While with desktop virtualization, end users receive a new VM each time they login (except for persistence setup), the issues related to applications and OS is expected to be reduced. Requests like password reset and customization for OS are not expected to reduce. So it is expected that number of staff in level1 is slightly reduced. Since all desktop computers are centralized, if a user faces a login failure it is most probably not a single user issue, it is an issue that affect all users. With desktop virtualization, most of the incidents will be tackled at the datacenter level and not at the end user desk, due to that Level2 and Level3 staff will get increased **[7]**.

Virtualizing desktop computers increases the administration and management overhead. Companies can reduce the management overheads by carefully planning changes in the administration elements to adapt with desktop virtualization. Starting with people, the support pyramid (level1, 2 and 3) should be changed, adding more staff to level 2 and 3 while slightly reducing level1 staff.

Companies should have clear processes in place to ease the management and administration. For instance, the procurement strategy for desktops should be changed. In most cases, no more super-desktop computers or fat desktop computers are required. Companies should have standard procedures and processes for buying thin clients. Standardizing hardware helps reducing the administration overhead.

Technology plays an important role in reducing administration overheads. Tools that automate and proactively act to incidents will definitely reduce the administration and management overhead.

3.2. IT Infrastructure Requirements

IT infrastructure challenges are summarized in the following:

- Storage capacity requirements.
- Storage Performance.
- Server Resources.
- Network Latency.

VDI demands more storage capacity compared to SBDV. For large deployments, storage is a bottleneck with VDI approach. As outlined earlier, VMs can be either persistent or non-persistent. With persistent approach, users have their dedicated VMs. In average, the disk requirement for one image (OS, applications and data) is 30-40 G.B **[1]**. For large deployments, example of 4000 desktop computers, the required space is 160 TB. While the storage cost is drastically reduced, still server storage cost is higher than desktop computers storage. According to Gartner group, over 40-60% of desktop virtualization budget is spent on storage **[46]**.

One of the options to reduce the disk space requirements is to use a master image. With the master image, the required disk space is reduced since VMs uses the same master image. Since VMs use the same master image, the settings and applications are identical for all VMs. It is recommended for companies seeking to standardize their applications and settings. If the users have the freedom to customize their applications, then this is not the right approach. Application virtualization (Microsoft App-v, VMware Thin App and Citrix XenApp) is an option to overcome this challenge. By integrating application virtualization with the master image, clients use the master image to load the OS and then their customized applications are streamed using the application virtualization solution.

Vendors and Researchers are continuously developing new solutions to overcome the space requirements of VMs. For instance, Microsoft introduced a new feature called "differencing disk" [8]. This feature allows a child-parent relationship between the child (differencing disks) and the parent (master image) VHDs. Differencing disk holds the changes from the parent VHD, while parent VHD contains the OS plus the standard applications and settings. Applications like Antivirus Agents, Anti-spam and standard applications are included within the parent VHD. This approach reduces the space requirements for the VMs under the condition explained in the following paragraph.

To host 100 VMs, the solution is to create a master image and then prepare 100 VMs in the form of differencing disks. This approach reduces the disk space if the majority of the disk read operations are done on the master image, this is not always true. Having the majority of the disk read operations performed on the master image means that users are only using the applications installed on the master image. What if the users install their own application, then the majority of the disk read and write operations will be performed on the child disks, which means that the size of the child disks will increase. In some cases, the size of the child disk get increased to a level where it exceeds the size of the master image. As a conclusion, this solution can be implemented for cases where differences between the master image and the child disks are minimum [1].

If the majority of the disk read operations are performed on the master image, then it is important to make sure that the master image is hosted on a disk with very high I/O to handle the disk read requests from the VMs. If the number of read requests to the master image is high (which is the case for VDI since multiple VMs are reading from the same master image), then the master image disk performance can be turned into a bottleneck.

The objective of VDI is to host multiple VMs on a single host in order to reduce the cost and to increase the efficiency and security. With the persistent type of VMs where users have their own VMs (no master image or image sharing), the performance is better compared to the shared VM approach, but it has drawbacks including disk space requirements and administration overhead. This approach is illustrated in *Figure 19* below.

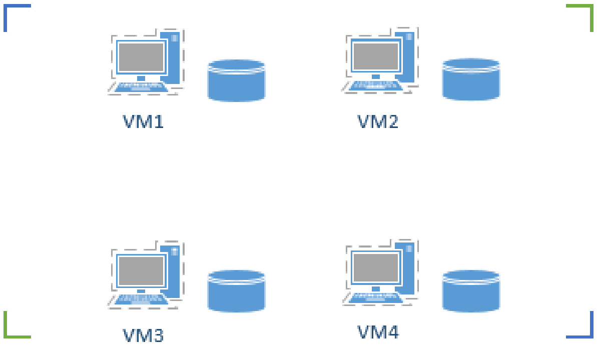


Figure 19: Persistent VMs hosted on a physical server.

With the master image approach where multiple VMs share the same image (*Figure 20*), the challenge is with the disk write operations on the master image. Technically, it is not possible for two VMs to share the same disk since the disk is locked by the VM accessing it. VMs have huge number of transaction to write including logs, temp files and page files. So to solve the problem, the master image is used for disk read operations, while VMs perform their write operations on an alternative location “child disks” **[1]**.

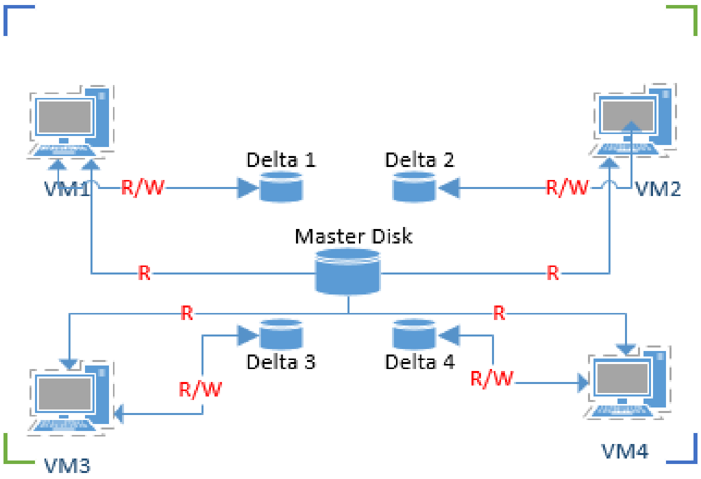


Figure 20: VMs with Shared Master Image. Read-Only Operations are performed to the Master Image. Read/ Write Operations are performed on the differential / delta disks.

As illustrated in *Figure 20*, VMs have two VHDs. The master image for the read operations, and the child VHDs for the read/ write operation. When a VM writes on differential disk and it needs to read what it wrote earlier, then the read operation is performed on the child disk and not on the master image VHD. This approach reduces the required disk capacity on the SAN or storage resources. It is expected that huge number of read hits will be performed on the master image which may result in a performance bottleneck. To overcome this challenge, read operations can be cached for better performance. It is recommended to cache the disk read operations on a storage with high IOPS, solid state disk (SSD) is a good choice for caching read operations **[1]**. New servers are equipped with SSD, which makes the implementation of the caching option easier.

As outlined earlier, the parent-child approach reduces the disk requirements, but this is not applicable in all cases. When the VMs are newly created, they have minor changes from the master image which makes the differential disks very small in size. But with time, VMs perform disk write operations on their child disks, operations like installing new hotfixes, installing applications and service packs increase the size of the child disks. And over time, the size of the child disks may exceed the size of the master image. Not only that, as outlined earlier, in order to overcome the performance bottleneck of the master image, storage disks like SSD are recommended. But since the write operations are done on the child VHDs and VMs will read what it wrote before, then most of the disk read operations are shifted from the master image to the child disks which are hosted on a storage disks like the SAN. With multiple machines reading and writing to the SAN, there is a possibility for performance bottleneck. But unlike with master image, using SSD with the child disks will be a challenge due to the high cost of SSD, as well, SSD doesn't come in large capacity. A normal hard disk costs around $0.075 per GB while an SSD costs around $1 per GB (SSD is almost 13 times more expensive that normal HDD) **[47]**.

To overcome the performance challenge for the child disks, VDI vendors recommend recomposing VMs. The recomposing process involves creating an updated master image with all the hotfixes, service packs and common applications. In order to apply the updated master image, the "dirty" child disks must be deleted. By doing so, the difference between the differential images and the master VHDs is minimum and majority of read operations are again performed on the master image. The problem with this approach is the deletion of the child disks which contain all users' customized settings, applications and data, this is an approach that will not be accepted for some scenarios **[8]**.

To overcome the problem of child disks deletion, a different partition for the user data can be created. For example, C drive is used for the system and D drive for the users' data and the users are restricted to save their data only to the designated partition (D).

The problem with this approach is that if the user installs applications on C drive, then things are getting back to square one again (the size increment of the differential disk) [1].

There is no one solution that fits all requirements, different requirements can be addressed with different solutions. The parent-child approach fits for standard applications. The persistent VMs approach with no master image can solve the problem for the users who need administrative permissions and customized applications. Other solutions involve using SBDV, more details about the best solution based on needs are in *section 3.3 "Session Virtualization vs VDI"*.

The server resources requirements is another challenge for server-based desktop virtualization. Moving desktop computers from the client side to the datacenter requires investment on the datacenter side, it also requires proper capacity planning for IT infrastructure resources (memory, disk, CPU, GPU and network) to accommodate the computational needs of the desktop computers. Hardware and software assessment are a crucial element for the planning phase, it gives a clear idea about different parameters like applications compatibility, resources utilization, applications execution latency, peak hours for applications usage and others.

Network is another crucial component for any desktop computing architecture, let it be a traditional desktop computers environment or a virtualized one. But when it comes to virtualized desktop environment, more care should be placed on the network part since it is the artery of the virtual desktop environment. For LAN scenarios with a network speed of 100 Mbps and above, all types of RDPs are expected to function with an acceptable performance. But the concern is with WAN scenarios where high speed bandwidth is not always available. WAN optimization, WAN acceleration and Quality of Service (QoS) help optimizing the WAN. Solutions like Riverbed (*a leader in WAN optimization*) cache the common traffic and apply different compression algorithms to optimize the traffic. Different RDPs utilize the available bandwidth in different ways. More details about the RDPs are in *Section 3.6 "Remote Desktop Protocols"*.

3.3. Session Virtualization and VDI

As outlined earlier, there is no one solution that fits all requirements. VDI is recommended for some scenarios where SBDV doesn’t fit and vice versa. While for other scenarios, integrating both solutions gives the required outcome. This section explores the challenges for both solution and denotes where each solution fits the best.

With SBDV, the host “session host server” provides sessions to the clients, the clients connect to the server (server OS) and not to a client OS. SBDV is the most used server-based desktop virtualization solution with more 100 million users worldwide [1]. SBDV simplifies the administration and installation of applications but it has several limitations. Application compatibility is one of the challenges for SBDV. Some of the applications do not support multiple sessions (instances) of the same application running simultaneously on a single OS. While other application will not function when running in session-based virtualization since all sessions have the same IP address (which means all clients use the same IP Address which belongs the server).

Note: the application compatibility issue related to using a single IP address is tackled by having virtual IPs to provide each session with its own IP address.

Other issues with SBDV relates to availability. Since all sessions run on the same server, any failure on the server side will affect all sessions hosted on that server. In addition to that, any maintenance or upgrade on the server should be performed out of peak hours. While high availability (HA) and load balancing (LB) help increasing the server up time, there are cases where such solutions will not be helpful. For instance, if the sessions are load balanced into two or more servers and one the servers malfunctioned, then the active session on the malfunctioned server will not be automatically redirected to the other server in the cluster, users session are disconnected and they have to initiate the connection again. Other challenges are related to access level assigned to users. Since users share the same server, it is highly recommended not to provide administrative or power privileges to end users, any change performed by users with administrative permissions affects all users session hosted on the same server. Hence, SBDV is not recommended for scenarios where administrative privileges is needed by the end users.

Performance is another challenge for SBDV, the server resources at not equally distributed among sessions. A user with intensive access to resources will degrade the performance for other users. Vendors introduced different solution to control the distribution of the server resources between the sessions. Solutions like session throttling and resource quotas help assigning different quotas for the users based on their profiles and requirements.

SBDV is still the most used approach among server-based desktop virtualization solutions. This is due to the simplicity of management and administration, low-cost and other factors. While SBDV is not the right solution for power users, it fits for normal users. Users using desktop computers for word processing, internet access and other light applications can be hosted under the SBDV umbrella.

VDI is a complex solution that involves integration between different components. VDI administration overhead is higher compared to SBDV. For instance, while installing a new application under SBDV is performed on a single location (the host server), with VDI persistent VM approach, one has to manage those tasks manual or using central management tools like Microsoft SCCM. With VDI non-persistent VM approach, the process is easier as the applications installation is performed on a single location (master image). Another challenge for VDI is the cost. The VDI cost is derived from high-end servers, licenses and operational costs.

According to Forrester **[25]**:

"When considering hosted virtual desktops (also known as virtual desktop infrastructure, or VDI) as an alternative or in addition to physical PCs, it's difficult to find a reliable information about the differences in costs between physical PCs and virtual desktops, especially in the operational costs. Forrester has reviewed various total-cost-of-ownership (TCO) calculators and guides from several vendors, and all have a predictable pattern: They typically describe the lifecycle costs of traditional PC environments in details while overlooking the true operational costs of the supporting VDI infrastructure components".

Note: the cost of server-based desktop virtualization is based on hardware cost for the servers, licenses and operational costs. Servers are getting cheaper and more powerful, which increases the ratio of clients to servers and reduces the cost. Thin clients are another factor that helps reducing the cost of virtual desktops (*more details about thin clients are available in Appendix D: Thin Clients Overview*).

Both VDI and RDSH have common challenges illustrated in the following points.

- No offline mode: since VMs and sessions are hosted on the datacenter servers. LAN connection is required for on-premises access, while internet connection is required for remote users.

- Multimedia and graphical application limitations: both RDPs and network bandwidth have evolved since 1998 (terminal services), and so did the user expectations and requirements. While users where using only word processing and light applications before, now users require graphics intensive and heavy applications [1].

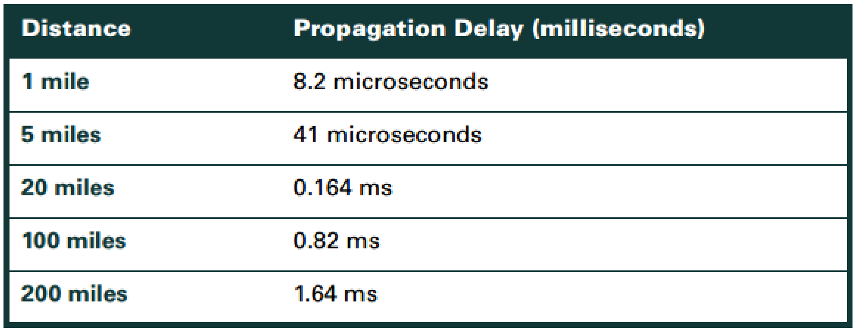
Server-based desktop virtualization is an emerging technology, it has limitations that should be tackled. There are scenarios where server-based desktop virtualization (VDI and SBDV) fit, examples: disaster recover (DR), Bring Your Own Device (BYOD), high security requirements where data should not leave the datacenter, scenarios where connection between the applications and their database servers requires high speed connection and for scenarios where desktop computers' availability is critical.

The cost component of the server-based desktop virtualization should be clarified and cleared out. Desktop virtualization reduces the cost of hardware since traditional desktop computers are replaced with thin clients, but when it comes to the licensing cost, the cost is not clear. According to Forrester Research [25], few firms have a clear understanding of what is involved in operating a VDI environment, which causes them to significantly underestimate the true costs. To avoid the underestimation of cost, vendors should clearly clarify the cost for their desktop virtualization solution, while companies should carefully calculate the initial cost for desktop virtualization (hardware, software and consultation) and the cost for operating the solution (running cost). The cost is an important factor for any decision taking scenarios, but there are some cases where other factors (security, risk mitigation, and business requirements) play an important role as well. For instance, if resilience for desktops is required, VDI is the recommended solution even if it is not cheaper than traditional desktop computers.

3.4. Remote Desktop Protocols

As outlined earlier, all RDPs work well on LAN scenarios with a connection speed of 100 Mbps and above. The challenge is when it comes to WAN scenarios. Before going through different developments on RDPS to support WAN scenarios, it is important to highlight the limitation of the network. Whether using desktop virtualization or traditional desktop computers, the network latency is a fact especially for high distance communications.

Propagation delay is the primary source for network delay. The propagation delay measures the time required for traffic to travel at the speed of light in the communication media (copper, fiber) from source to destination. In free space, the speed of light is approximately 3x105 km/ second. This theoretical speed is reduced in the communication media like copper and fiber **[26]**. *Figure 21* below shows the propagation delay over distance **[27]**.



Distance	Propagation Delay (milliseconds)
1 mile	8.2 microseconds
5 miles	41 microseconds
20 miles	0.164 ms
100 miles	0.82 ms
200 miles	1.64 ms

Figure 21: Propagation Delay and Distance [27]

RDPs are continuously evolving to address the needs of end users. Different vendors have different developments and solutions to overcome the limitations of RDPs. Different developments are outlined in *Chapter2*. This section aims to focus on the factors affecting the quality delivered by RDPs.

Graphics rendering can be performed on the server side and on the client side. Server-side rendering is suitable for still images and light profiles *(more details about the different profiles categorizations are in chapter 4 – Lab Measurements and Capacity Planning).* With the introduction of GPU and its integration with hypervisors, VMs are able to perform intensive graphics rendering on the server side. But the challenge is the network bandwidth, unlike still images, multimedia requires higher frame rates per second and higher network bandwidth. Server-side rendered multimedia is compressed and encrypted before being transferred to the client side. The client then decrypts and decompresses it. Three approaches are available for compression: lossy, lossless and hybrid. Lossy compression is used with UDP protocol which allows dropping packets in case of network saturation (recommended for video streaming). While lossless compression is used if packet lose is not acceptable, lossless compression uses TCP protocol. Hybrid approach combines both, it is configurable to use lossy or lossless compression, Citrix HDX Protocol is an example for the hybrid approach.

Client-side rendered multimedia is more efficient than server-side rendering. With client-side rendering, the server sends few commands to the clients to render content on the client side rather than rendering it on the server side, then the display updates are sent over the network medium. For client-side rendering to function, the client should understand all graphics format natively. In addition to that, client side should have proper graphic cards to perform the rendering process. This is against the objective of server-based desktop virtualization to move the computation from the client side to the server side and replace fat clients with thin clients.

As outlined earlier, vendors have their own RDPs. Different researches were conducted to develop open standard RDPs. Net2Display VESA standard is one of open standard developments. The development of Net2Display protocol took more three years (2006-2009), by that time all RDPs were matured **[29]**.

The objective of Net2Display protocol is to create an open standard that allows servers to send display data over high speed data channels. Typical channels include but not limited to: Ethernet, USB, etc. Other objectives are to reduce the complexity of clients and to support both long distances and high quality video with low response time. Net2Display remotes the display from the Net2Display host to the Net2Display client and optionally redirects I/O devices (keyboard, mouse, scanners, printers, etc.) attached to the client using USB **[29]**. *Figure 22* illustrates the block diagram for Net2Display Protocol.

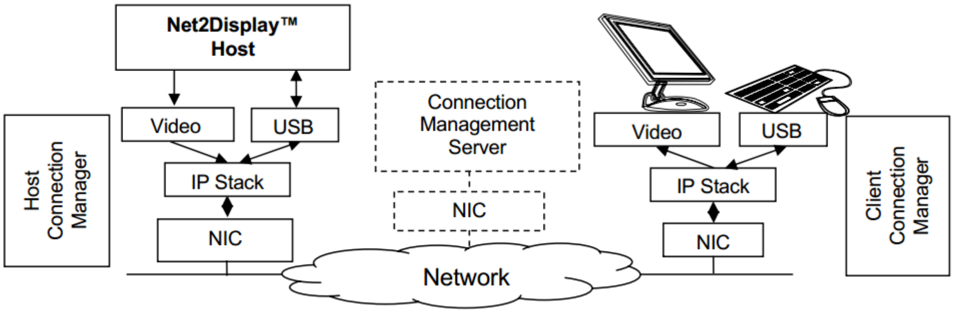


Figure 22: Net2Display Bock Diagram [29]

The connection management block is presented at the client network to manage the connection between the Net2Dispaly host and the Net2Dispaly clients. The function of connection management block is to allocate host resources to the Net2Display client

Net2Display protocol could have had tangible advantages that comes with real adoption. One advantage is simplified clients since one protocol and one RDC are used. Other advantages are optimization and better performance for video traffic, this is achievable by adhering to a standard video compression approach that is supported in hardware.

Net2Display didn't gain the expected adoption due to different factors. Firstly, the protocol took long time to develop. As per VESA in 2006 **[29]**, the protocol was expected to develop in a year time but it took more than three years to release the first version, such period gave other RDPs enough time to become mature. Secondly, Net2Display protocol is just a specification or a framework without a code or a real product.

Wyse corporate released the following statement as a feedback about Net2Display **[48]**

"This protocol, while good, is like any protocol. It must have an ecosystem of support, or no matter how good it is, it won't be adopted. Wyse's strategy is to assist in the development and optimization of protocols, using our significant technical depth and IP portfolio, and this is why we're part of Net2Display, just as we have also added features to RDP, ICA/HDX, and PCoIP. However, without the support of one of our major partners (C/M/V), we won't be including the protocol in our support strategy. This, of course, can change as our partners review and test the protocol, if they choose to".

As a conclusion, the approach of having a standard RDP is required. The proposed standard should combine both benefits of current RDPs and should be a real product to use.

Chapter4: Lab Measurements and Capacity Planning

The objective of the lab measurements is to provide an estimation for the server resources' utilization and the network bandwidth requirements for the server-based desktop virtualization. Measured resources in this chapter are: memory, disk I/O, network and CPU. The measurements are conducted on two versions of Microsoft RDP, RDP 7.0 and RDP 8.0. The same scenarios detailed in this chapter can be conducted for other RDPs like VMware PCoIP and Citrix HDX.

The research divides the users into three profiles: light users, heavy users and multimedia users (detailed in next section). The defined parameters (applications and settings) for each profile are defined based on an assessment conducted on an international company based in Dubai (*more detailed about the assessment in Appendix A*).

The capacity planning section represents the results collected and observed from the lab measurements. Capacity planning is a crucial component in the planning phase of any ICT related project, in particular, server-based desktop virtualization. Capacity planning helps developing an estimation for the resources required to meet the current demand while reserving a room for the future demand (scalability).

4.1. Lab Measurements

4.1.1. Lab Design & Components

The lab measurements are conducted using the following components:

- Remote Desktop Session Host server: RDSH Server provides sessions for the three profiles (light, heavy and multimedia). (Appendix B).
- CADs: Four types of devices are used: thin client (Raspberry Pi), a laptop (Dell), a tablet (iPad) and a VM. (Appendix B).
- Microsoft Remote Desktop Load Simulation Tool: a toolset used to for server capacity planning and performance/ scalability analysis (Appendix B).
- Test Controller Server: used to coordinate the test execution between the client devices and the RDSH server (Appendix B).

The Lab environment is illustrated in *Figure 23* below.

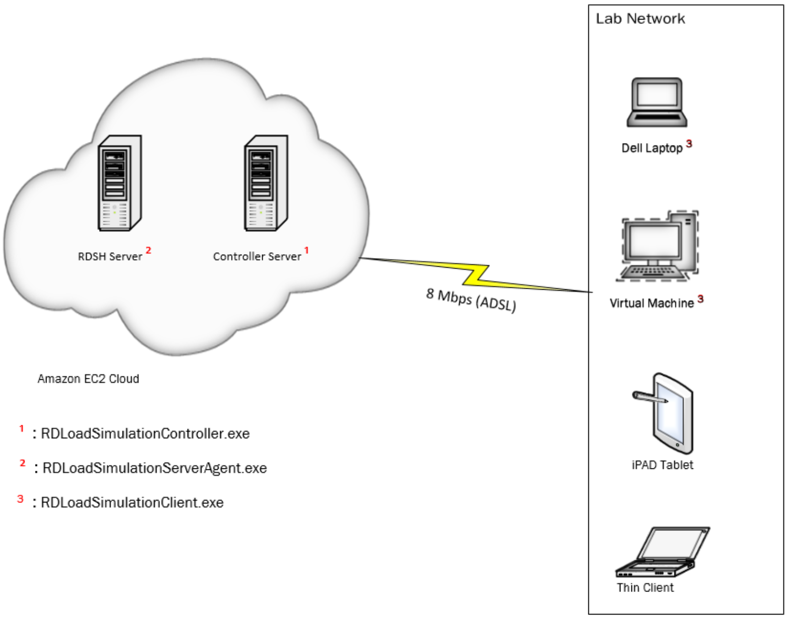


Figure 23: Lab - High Level Architecture

4.1.2. User Profiles

Users are categorized into three profiles based on their requirements and the applications used. The profiles are: light, heavy and multimedia. Below sections illustrate each profile in details.

4.1.2.1 Light Users

<u>Light Profile Parameters:</u>

- Applications used: Microsoft Internet Explorer V10, Microsoft Word 2013, Microsoft Excel 2013, a home folder.
- Screen resolution: 800 × 600.
- Bitmap quality: 32 bit.
- Typing rate: 80 words per minute (WPM).
- Printer redirection: enabled.
- Test duration: 5 minutes and 20 seconds.

<u>Light Profile Scenario:</u>

Below scenario is automated using a customized script, the script content is provided in *Appendix C – Light Profile Script.*

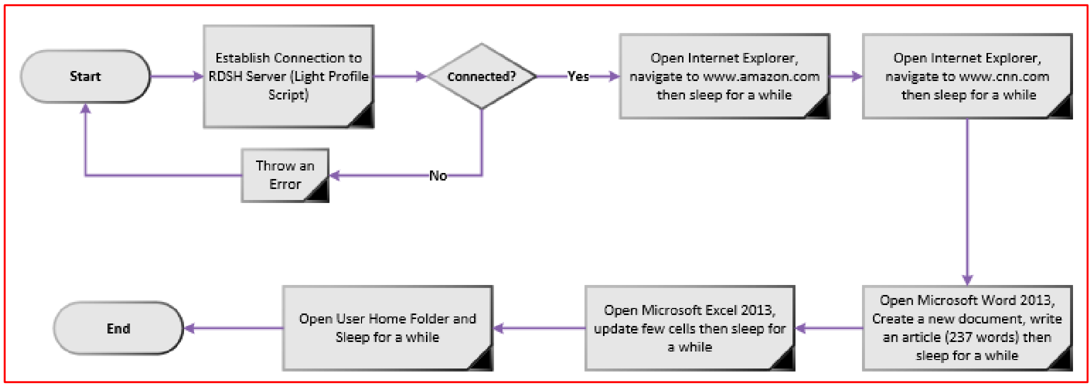

Start
Establish Connection to RDSH Server (Light Profile Script)
Connected?
Yes
No
Throw an Error
Open Internet Explorer, navigate to www.amazon.com then sleep for a while
Open Internet Explorer, navigate to www.cnn.com then sleep for a while
Open Microsoft Word 2013, Create a new document, write an article (237 words) then sleep for a while
Open Microsoft Excel 2013, update few cells then sleep for a while
Open User Home Folder and Sleep for a while
End

Figure 24: Light Profile - Flow Chart

4.1.2.2 Heavy Users

Heavy Profile Parameters:

- Applications used: Microsoft Internet Explorer V10, Microsoft Word 2013, Microsoft Excel 2013, Microsoft PowerPoint 2013, Microsoft Outlook 2013, Windows calculator and Adobe Acrobat Reader 9.0.
- Screen resolution: 1024 × 768.
- Bitmap quality: 32 bit.
- Typing rate: 160 WPM.
- Printer redirection: Enabled.
- Test duration: 3 minutes and 25 seconds.

Heavy Profile Scenario:

Below scenario is automated using a customized script, script content is provided in *Appendix C – Heavy Profile Script.*

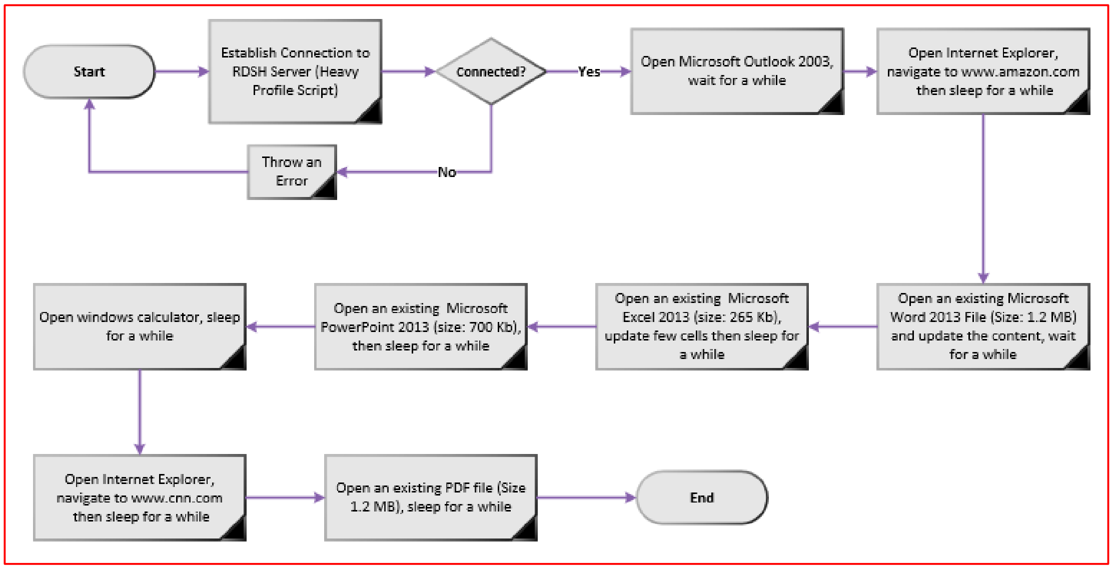

Start
Establish Connection to RDSH Server (Heavy Profile Script)
Connected?
Yes
No
Throw an Error
Open Microsoft Outlook 2003, wait for a while
Open Internet Explorer, navigate to www.amazon.com then sleep for a while
Open an existing Microsoft Word 2013 File (Size: 1.2 MB) and update the content, wait for a while
Open an existing Microsoft Excel 2013 (size: 265 Kb), update few cells then sleep for a while
Open an existing Microsoft PowerPoint 2013 (size: 700 Kb), then sleep for a while
Open windows calculator, sleep for a while
Open Internet Explorer, navigate to www.cnn.com then sleep for a while
Open an existing PDF file (Size 1.2 MB), sleep for a while
End

Figure 25: Heavy Profile – Flow Chart

4.1.2.3 Multimedia Users

<u>Multimedia Profile Parameters:</u>

- Applications used: Microsoft Internet Explorer V10, Microsoft Outlook 2013 and VLC Media Player.
- Screen resolution: 1024 × 768.
- Bitmap quality: 32 bit.
- Typing rate: 160 WPM.
- Printer redirection: enabled.
- Audio redirection: enabled.
- Test duration: 4 minutes and 30 seconds.

Multimedia Profile Scenario:

Below scenario is automated using a customized script, script content is provided in *Appendix C – Multimedia Profile Script.*

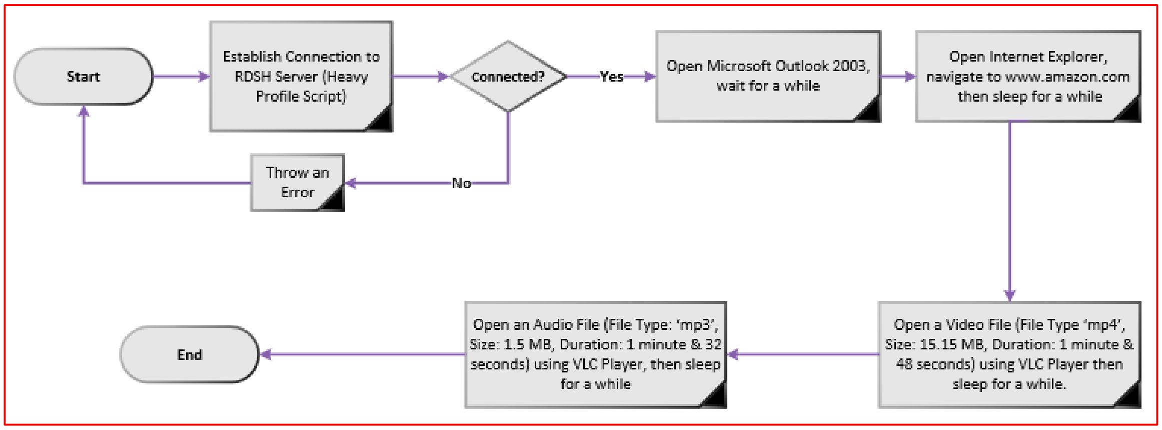

Start
Establish Connection to RDSH Server (Heavy Profile Script)
Connected?
Yes
Open Microsoft Outlook 2003, wait for a while
Open Internet Explorer, navigate to www.amazon.com then sleep for a while
No
Throw an Error
Open a Video File (File Type 'mp4', Size: 15.15 MB, Duration: 1 minute & 48 seconds) using VLC Player then sleep for a while.
Open an Audio File (File Type: 'mp3', Size: 1.5 MB, Duration: 1 minute & 32 seconds) using VLC Player, then sleep for a while
End

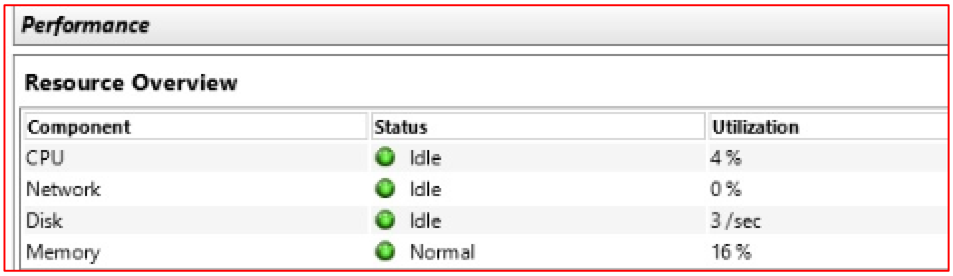
Figure 26: Multimedia Profile – Flow Chart

4.1.3. Lab Measurements

The RDSH server shown in *Figure 23* has different applications and services installed (Microsoft Active Directory (AD), DNS, and Remote Desktop Services). In order to get accurate results for the required resources per session / VM, the server utilized resources are captured prior to the test.

4.1.3.1. Light Profile

When the RDSH server is in idle mode (no sessions connected), the resources utilization is as illustrated in *figure 27* below:



Performance

Resource Overview

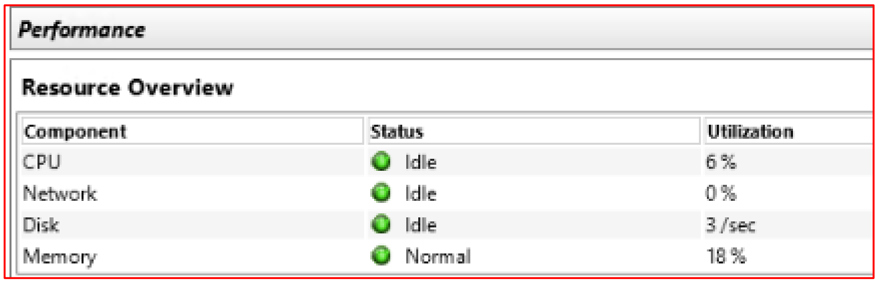
Component	Status	Utilization
CPU	● Idle	4 %
Network	● Idle	0 %
Disk	● Idle	3 /sec
Memory	● Normal	16 %

Figure 27: RDSH server resource utilization – Idle mode (light profile)

Note1: RDSH server memory: 7.5 G.B, only 7.0 G.B is available, rest are used for page files.

Note2: RDSH server CPU: 2 Cores (Intel Xeon CPU E5-2650 @ 2.00 GHz). CPU utilization in *figure 27* combines both cores.

Test1: The first test is conducted by connecting one user (session) to the RDSH server. Results are illustrated in below *figure*.



Performance

Resource Overview

Component	Status	Utilization
CPU	● Idle	6 %
Network	● Idle	0 %
Disk	● Idle	3 /sec
Memory	● Normal	18 %

Figure 28: RDSH server resource utilization – light profile (1 session)

The resource utilization for *one session* is calculated as below:

- CPU utilization percentage = 6% - 4% = **2%.**
- CPU utilization value =   2% × 2.00 GHz = **40 MHz**.
- Memory utilization percentage = 18%-16% = **2%.**
- Memory utilization value = 2% × 7.0 G.B = **140 MB**.

The next step is to perform a stress test on the RDSH server to check the maximum number of sessions that can be hosted.

Firstly, the research calculates the maximum number of sessions theoretically:

- 1 session requires 40 MHz of CPU and 140 MB of memory.
- CPU:

$$\text{Max number of sessions} = \frac{2000\text{ MHz} - 80\text{ MHz}}{40\text{ MHz}} = \mathbf{48\ Sessions}.$$

⇨ Memory:

$$\text{Max number of sessions} = \frac{7.0\text{ G.B} - 1.12\text{ G.B}}{140\text{ MB}} = \mathbf{42\ Sessions}.$$

Note: calculation formula:

$$\text{Max number of sessions} = \frac{\text{Total Available Resource} - \text{Resource Utilization in Idle Mode}}{\text{Resource Utilization for one session}}$$

The stress test is conducted using the following three components **[32]**:

1. RDLoadSimulationController.exe: installed on the controller server. It is a central control point for load simulation testing. It controls all test parameters and defines the progression of the simulated user load, it communicates with the Client and RDSH server to synchronize and drive the client-server remote desktop automation.

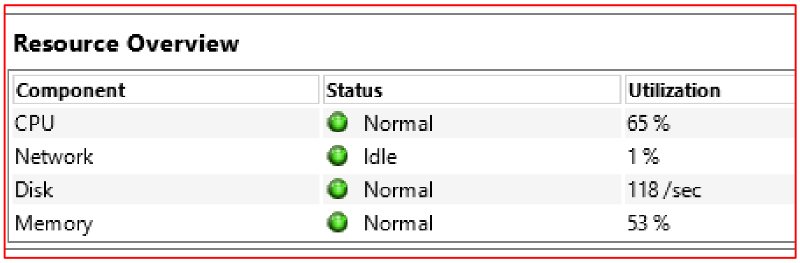
2. RDLoadSimulationClient.exe: a client agent that controls the client side of the load simulation testing.

3. RDLoadSimulationServerAgent.exe: runs on the RDSH server for test synchronization with the load server.

Test2: The second stress test is performed on 25 sessions. The theoretical load is:

⇨ CPU = 40 MHz × 25 = 1 GHz. (53% of available CPU).
⇨ Memory = 140 MB × 25 = 3.5 G.B. (59% of available memory).

The results collected from the performance counters of the RDSH server are shown in *figure 29* below.



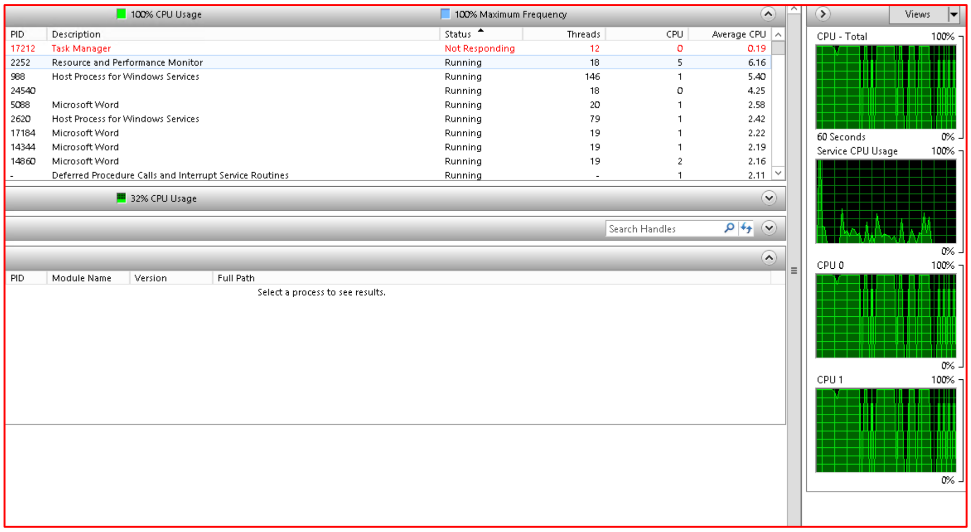
Resource Overview

Component	Status	Utilization
CPU	● Normal	65 %
Network	● Idle	1 %
Disk	● Normal	118 /sec
Memory	● Normal	53 %

Figure 29: RDSH server utilization – light profile (25 sessions).

Test3: The third stress test is conducted on 45 users:

The results collected from the performance counters of the RDSH server are shown in *figure 30* and *figure 31* below.



🟩 100% CPU Usage 🟦 100% Maximum Frequency

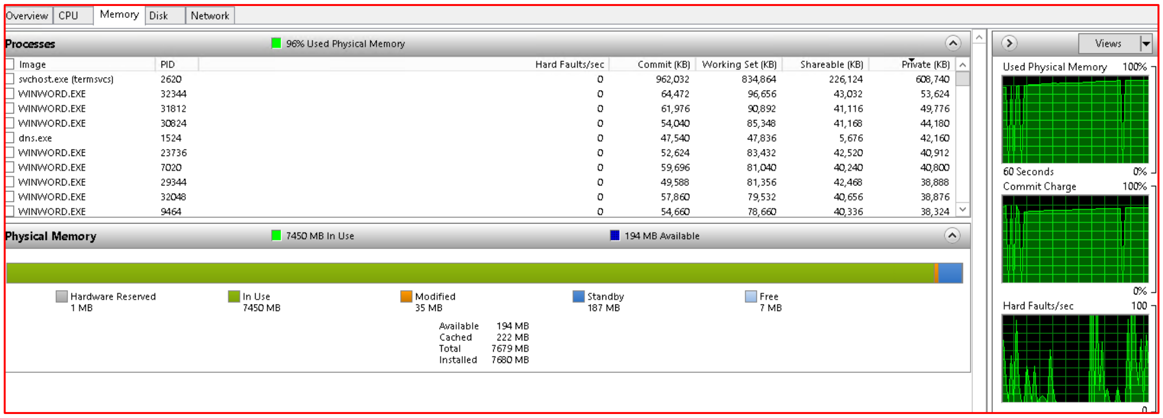
PID	Description	Status	Threads	CPU	Average CPU
17212	Task Manager	Not Responding	12	0	0.19
2252	Resource and Performance Monitor	Running	18	5	6.16
988	Host Process for Windows Services	Running	146	1	5.40
24540		Running	18	0	4.25
5088	Microsoft Word	Running	20	1	2.58
2620	Host Process for Windows Services	Running	79	1	2.42
17184	Microsoft Word	Running	19	1	2.22
14344	Microsoft Word	Running	19	1	2.19
14860	Microsoft Word	Running	19	2	2.16
-	Deferred Procedure Calls and Interrupt Service Routines	Running	-	1	2.11

🟩 32% CPU Usage

PID	Module Name	Version	Full Path

Select a process to see results.

Figure 30: RDSH server – CPU utilization – light users (45 sessions)



Overview CPU Memory Disk Network

Processes 🟩 96% Used Physical Memory

Image	PID	Hard Faults/sec	Commit (KB)	Working Set (KB)	Shareable (KB)	Private (KB)
svchost.exe (termsvcs)	2620	0	962,032	834,864	226,124	608,740
WINWORD.EXE	32344	0	64,472	96,656	43,032	53,624
WINWORD.EXE	31812	0	61,976	90,892	41,116	49,776
WINWORD.EXE	30824	0	54,040	85,348	41,168	44,180
dns.exe	1524	0	47,540	47,836	5,676	42,160
WINWORD.EXE	23736	0	52,624	83,432	42,520	40,912
WINWORD.EXE	7020	0	59,696	81,040	40,240	40,800
WINWORD.EXE	29344	0	49,588	81,356	42,468	38,888
WINWORD.EXE	32048	0	57,860	79,532	40,656	38,876
WINWORD.EXE	9464	0	54,660	78,660	40,336	38,324

Physical Memory 🟩 7450 MB In Use 🟦 194 MB Available

Hardware Reserved	In Use	Modified	Standby	Free
1 MB	7450 MB	35 MB	187 MB	7 MB

Available	194 MB
Cached	222 MB
Total	7679 MB
Installed	7680 MB

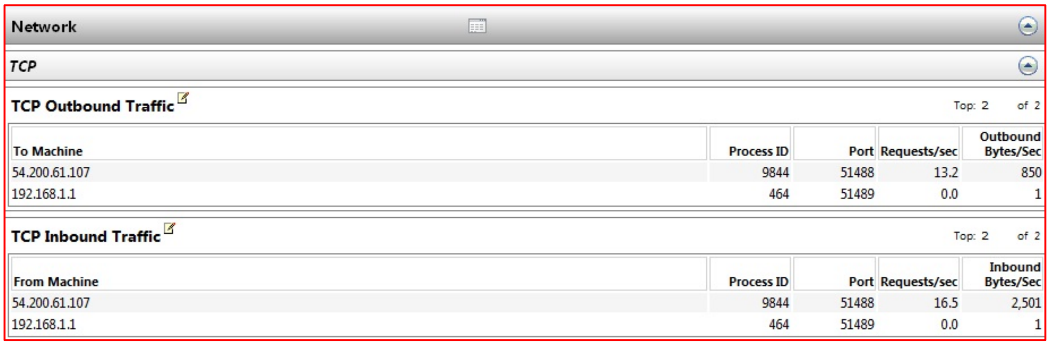
Figure 31: RDSH server- memory utilization – light users (45 sessions)

With 45 sessions, the server's CPU and Memory are overloaded. It is recommended to utilize 70-80% of the Server Resources to avoid any bottleneck.

Note: The stress test assumes that all the sessions are logged in to the server simultaneously. This is not the case in production environment but the stress test aims to provide capacity planning for the worst cases scenarios.

Test4: The required WAN bandwidth is measured for Microsoft RDP 7.0.

The required bandwidth for one session is shown in *figure 32* below:



TCP Outbound Traffic

Top: 2 of 2

To Machine	Process ID	Port	Requests/sec	Outbound Bytes/Sec
54.200.61.107	9844	51488	13.2	850
192.168.1.1	464	51489	0.0	1

TCP Inbound Traffic

Top: 2 of 2

From Machine	Process ID	Port	Requests/sec	Inbound Bytes/Sec
54.200.61.107	9844	51488	16.5	2,501
192.168.1.1	464	51489	0.0	1

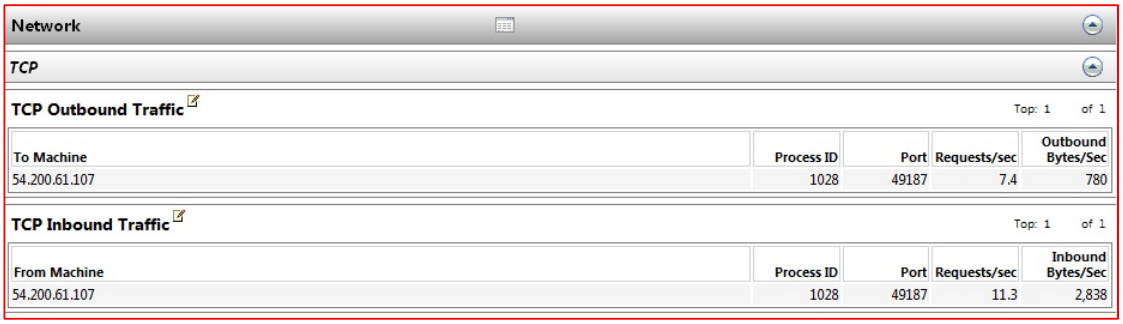
Figure 32: RDSH server – WAN bandwidth (RDP 7.0) – light users (1 session)

The total required bandwidth for one session (inbound traffic + outbound traffic):

850 + 2,501 = 3,351 bytes/sec ≈ **27 Kbps**.

Test5: The required WAN bandwidth is measured for Microsoft RDP 8.0.

The required bandwidth for one session is shown in *figure 33* below:



TCP Outbound Traffic

Top: 1 of 1

To Machine	Process ID	Port	Requests/sec	Outbound Bytes/Sec
54.200.61.107	1028	49187	7.4	780

TCP Inbound Traffic

Top: 1 of 1

From Machine	Process ID	Port	Requests/sec	Inbound Bytes/Sec
54.200.61.107	1028	49187	11.3	2,838

Figure 33: RDSH server – WAN bandwidth (RDP 8.0) – light users (1 session)

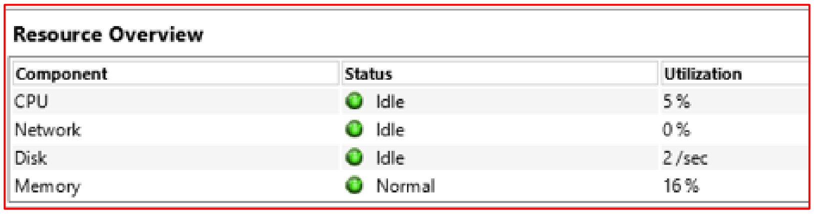
The total required bandwidth for one session (inbound traffic + outbound traffic):

$$750 + 2{,}838 = 3{,}351 \text{ bytes/sec} \approx \mathbf{28.7\ Kbps}.$$

Note: RDP 7.0 and RDP 8.0 approximately have the same bandwidth requirements for light users.

4.1.3.2. Heavy Profile

The performance counters were captured prior to the stress test. The resources utilization is shown below.



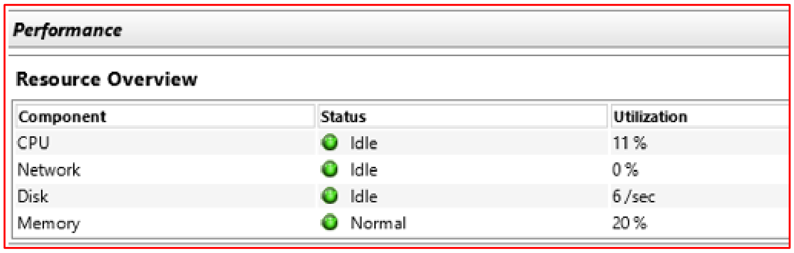
Resource Overview

Component	Status	Utilization
CPU	● Idle	5 %
Network	● Idle	0 %
Disk	● Idle	2 /sec
Memory	● Normal	16 %

Figure 34: RDSH server resource utilization – idle mode (heavy profile)

Note: The usable memory is **6.42 G.B**, rest is reserved for the OS and installed applications.

Test1: The first test is conducted by connecting one user (session) to the RDSH Server. Results are illustrated in below *figure*.



Performance

Resource Overview

Component	Status	Utilization
CPU	● Idle	11 %
Network	● Idle	0 %
Disk	● Idle	6 /sec
Memory	● Normal	20 %

Figure 35: RDSH server resource utilization – heavy profile (1 session)

The resource utilization for *one session* is calculated below:

⇨ CPU Utilization Percentage = 11% - 5% = **6%.**
⇨ CPU Utilization Value = 6% × 2.00 GHz = **120 MHz**.
⇨ Memory Utilization Percentage = 20%-16% = **4%.**
⇨ Memory Utilization Value = 2% × 6.420 G.B = **257 MB**.

Next step is to calculate the maximum number of sessions that can be hosted on the RDSH Server.

- ⇨ 1 session requires 120 MHz of CPU and 257 MB of memory.
- ⇨ CPU:

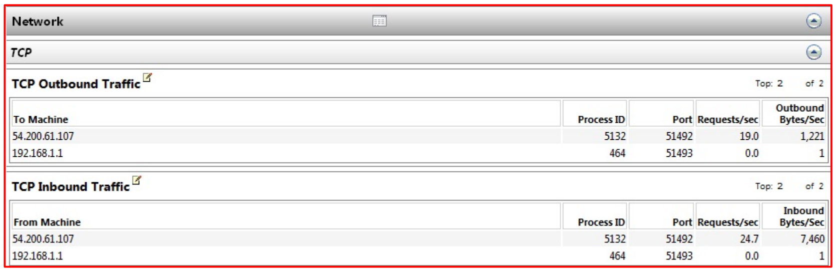
$$\text{Max number of sessions} = \frac{2000\ \text{MHz} - 120\ \text{MHz}}{120\ \text{MHz}} \approx \mathbf{16\ Sessions}.$$

- ⇨ Memory:

$$\text{Max number of sessions} = \frac{6.42\ \text{G.B} - 1.0272\ \text{G.B}}{257\ \text{MB}} \approx \mathbf{21\ Sessions}.$$

Test2: The required WAN bandwidth is measured for Microsoft RDP 7.0.

The required bandwidth for one session is shown in *figure 36* below:



Network

TCP

TCP Outbound Traffic — Top: 2 of 2

To Machine	Process ID	Port	Requests/sec	Outbound Bytes/Sec
54.200.61.107	5132	51492	19.0	1,221
192.168.1.1	464	51493	0.0	1

TCP Inbound Traffic — Top: 2 of 2

From Machine	Process ID	Port	Requests/sec	Inbound Bytes/Sec
54.200.61.107	5132	51492	24.7	7,460
192.168.1.1	464	51493	0.0	1

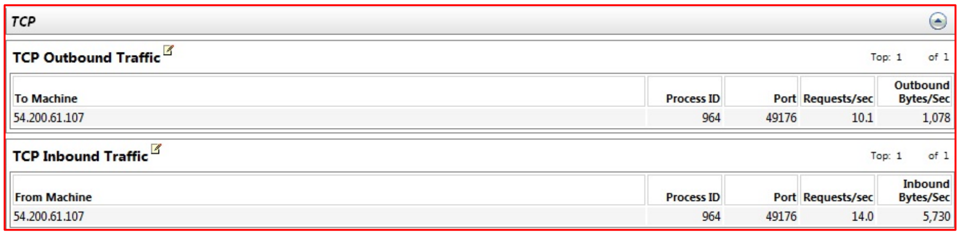
Figure 36: RDSH server – WAN bandwidth (RDP 7.0) – heavy users (1 session)

The total required bandwidth for one session (inbound traffic + outbound traffic):

1,221 + 7,460 = 8,681 bytes/sec ≈ **69.5 Kbps**.

Test3: The required WAN bandwidth is measured for Microsoft RDP 8.0.

The required bandwidth for one session is shown in *figure 37* below:



TCP

TCP Outbound Traffic — Top: 1 of 1

To Machine	Process ID	Port	Requests/sec	Outbound Bytes/Sec
54.200.61.107	964	49176	10.1	1,078

TCP Inbound Traffic — Top: 1 of 1

From Machine	Process ID	Port	Requests/sec	Inbound Bytes/Sec
54.200.61.107	964	49176	14.0	5,730

Figure 37: RDSH server – WAN bandwidth (RDP 8.0) – heavy profile (1 session)

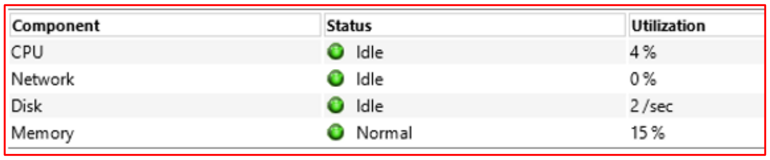
The total required bandwidth for one session (inbound traffic + outbound traffic):

1,078 + 5,730 = 6,808 bytes/sec ≈ **54.5 Kbps.**

Note: RDP 8.0 optimizes the required bandwidth for heavy profile.

4.1.3.3. Multimedia Profile

When the RDSH server is in idle mode (no sessions connected), the resources utilization is as illustrated in *figure 38* below:



Component	Status	Utilization
CPU	● Idle	4 %
Network	● Idle	0 %
Disk	● Idle	2 /sec
Memory	● Normal	15 %

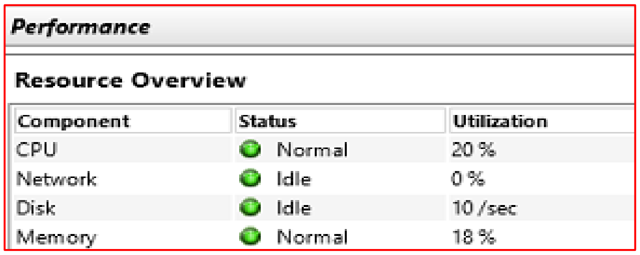
Figure 38: RDSH server resource utilization – idle mode (multimedia profile)

Memory specifications: 7.5 G.B (6.52 G.B available, rest are reserved for page files and other files).

Note: RDP is configured to redirect the audio from the server to the remote client.

Test1: The first test is conducted by connecting one user (session) to the RDSH server.

Results are illustrated in below *figure*.



Performance

Resource Overview

Component	Status	Utilization
CPU	● Normal	20 %
Network	● Idle	0 %
Disk	● Idle	10 /sec
Memory	● Normal	18 %

Figure 39: RDSH server resource utilization – multimedia profile (1 session)

The resource utilization for *one session* is calculated below:

- CPU Utilization Percentage = 20% - 4% = **16%.**
- CPU Utilization Value = 16% × 2.00 GHz = **320 MHz.**
- Memory Utilization Percentage = 18%-15% = **3%.**
- Memory Utilization Value = 3% × 6.52 G.B ≈ **196 MB**.

Next step is to calculate the maximum number of sessions that can be hosted on the RDSH server.

- 1 session requires 200 MHz of CPU and 522 MB of memory.
- CPU:

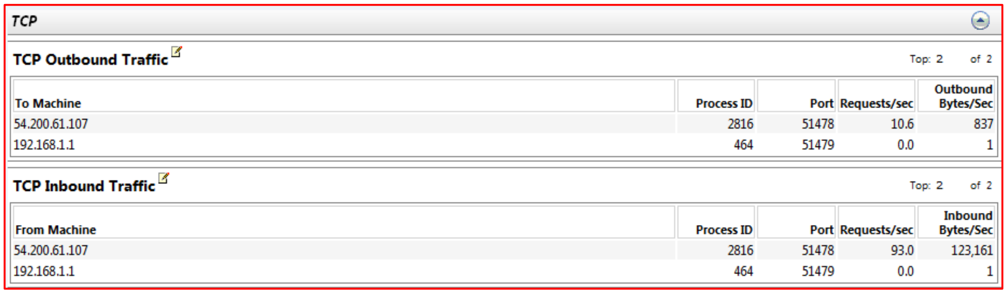
$$\text{Max nummber of sessions} = \frac{2000\ \text{MHz} - 80\ \text{MHz}}{320\ \text{MHz}} \approx \mathbf{6\ sessions.}$$

⇨ Memory:

$$\text{Max number of sessions} = \frac{6.52\ \text{G.B} - 0.978\ \text{G.B}}{196\ \text{MB}} \approx \mathbf{28\ sessions.}$$

Test2: The required WAN bandwidth is measured for Microsoft RDP 7.0.

The required bandwidth for one session is shown in *figure 40* below:



TCP

TCP Outbound Traffic — Top: 2 of 2

To Machine	Process ID	Port	Requests/sec	Outbound Bytes/Sec
54.200.61.107	2816	51478	10.6	837
192.168.1.1	464	51479	0.0	1

TCP Inbound Traffic — Top: 2 of 2

From Machine	Process ID	Port	Requests/sec	Inbound Bytes/Sec
54.200.61.107	2816	51478	93.0	123,161
192.168.1.1	464	51479	0.0	1

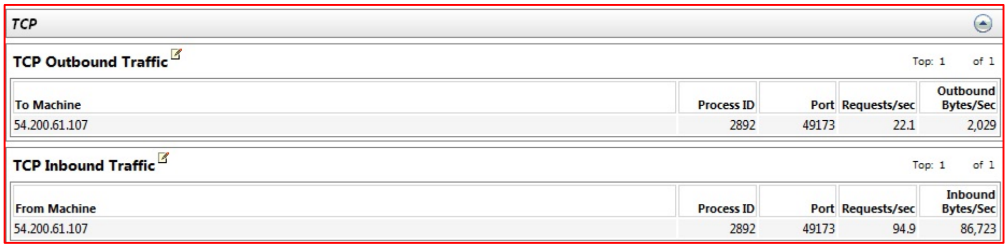
Figure 40: RDSH server – WAN bandwidth (RDP 7.0) – multimedia profile (1 session)

The total required bandwidth for one session (inbound traffic + outbound traffic):

837 + 123,161 = 123,998 bytes/sec ≈ **992 Kbps.**

Test3: The required WAN bandwidth is measured for Microsoft RDP 8.0.

The required bandwidth for one session is shown in *figure 41* below:



TCP

TCP Outbound Traffic — Top: 1 of 1

To Machine	Process ID	Port	Requests/sec	Outbound Bytes/Sec
54.200.61.107	2892	49173	22.1	2,029

TCP Inbound Traffic — Top: 1 of 1

From Machine	Process ID	Port	Requests/sec	Inbound Bytes/Sec
54.200.61.107	2892	49173	94.9	86,723

Figure 41: RDSH server – WAN bandwidth (RDP 8.0) – multimedia profile (1 session)

The total required bandwidth for one session (inbound traffic + outbound traffic):

2,029 + 86,723 = 88,752 bytes/sec ≈ **710 Kbps.**

<u>Note</u>: RDP 8.0 optimizes the required bandwidth for multimedia profile.

4.1.3.4. Disk Utilization

Light Profile

The average disk read and write hits for one session is not considered for the capacity planning since the value is very small.

Heavy Profile

The average disk read and write hits for one session = **4 hits / second** (refer to *figure 34* and *figure 35*).

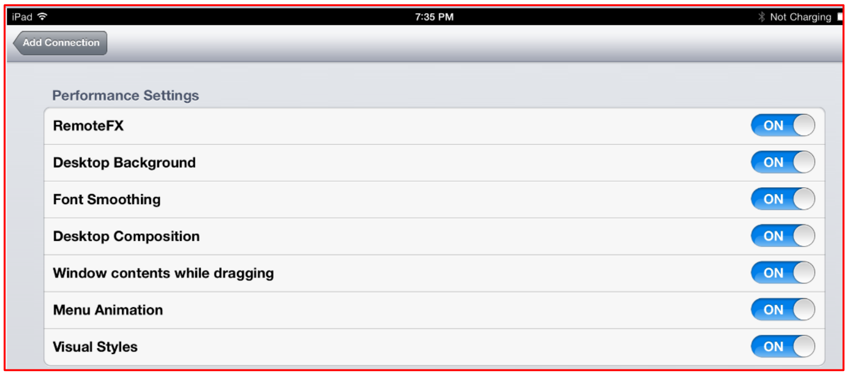
Multimedia Profile

The average disk read and write hits for one session = **8 hits / second** (refer to *figure 37* and *figure 38*).

Due to the rapid growth in the number of mobile devices, it is expected in the near feature for mobile devices (Tablets and Smartphones) to become the primary devices (Desktops) for end users **[4]**. It is important to review and test if mobile devices are compatible with virtual desktop environment and provide the expected user experience. One of the lab scenarios is to test the user experience on mobile devices. The test is conducted on an Apple iPad.

The following parameters are configured for the test:

- Bitmap: 32 bit (highest quality).
- Screen resolution: 800x600.
- User experience: Configured for the highest performance (Figure 42 below).



Performance Settings

RemoteFX ON

Desktop Background ON

Font Smoothing ON

Desktop Composition ON

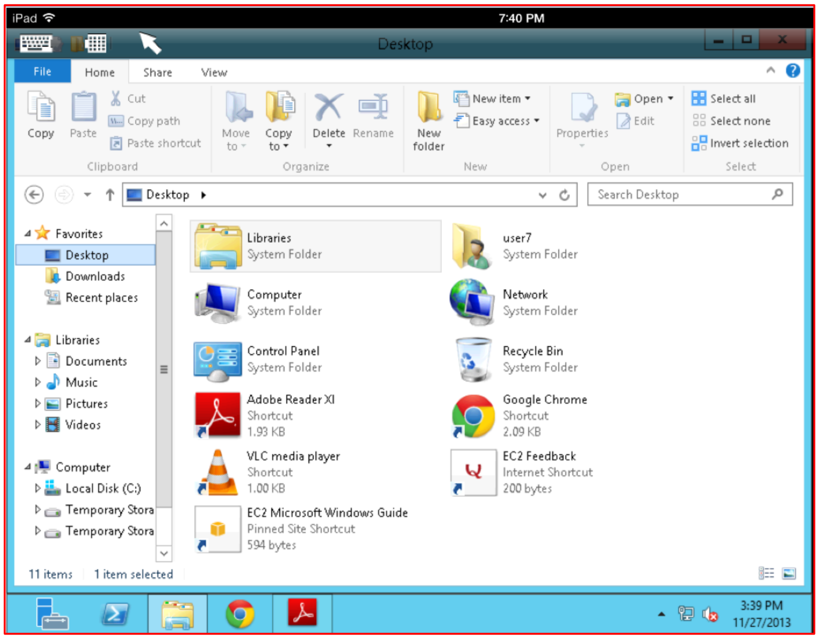
Window contents while dragging ON

Menu Animation ON

Visual Styles ON

Figure 42: User experience settings for RDP connection from an iPad

The connection experience is acceptable from the iPad for both light and heavy profiles. While for the multimedia profile, the video and audio quality is very low. It is also observed that the user experience while working with normal version of applications from a touch screen interface is not the same compared with traditional desktop computers. More details about this observation is in the conclusions and future work section. *Figure 43* below shows the RDP session from an iPad tablet.



Libraries System Folder

user7 System Folder

Computer System Folder

Network System Folder

Control Panel System Folder

Recycle Bin System Folder

Adobe Reader XI Shortcut 1.93 KB

Google Chrome Shortcut 2.09 KB

VLC media player Shortcut 1.00 KB

EC2 Feedback Internet Shortcut 200 bytes

EC2 Microsoft Windows Guide Pinned Site Shortcut 594 bytes

Figure 43: Virtual desktop session from an iPad tablet.

Thin clients reduce the implementation and operational cost of VDI and SBDV. With thin clients a longer life time, less power consumption, and less procurement cost are the main advantages of thin clients over fat clients. RDP connection is tested from a thin client (Raspberry Pi). The following command is used to access a full screen session from Raspberry Pi device:

```
rdesktop  RDSH Server Name (IP Address) -f
```

Raspberry Pi is a cheap mini computer that can be used as a thin client for light and heavy profiles but not for multimedia profiles. More details about thin clients are in *Appendix D "Thin Client Overview".*

4.2. Capacity Planning

Below *figure* summarizes the required server and WAN resources requirements for each of the profiles tested in the Lab.

Note: The values provided below are per session.

Profile type	**CPU quota**	**Memory quota**	**Disk IOPS**	**Network bandwidth (RDP 7.0)**	**Network bandwidth (RDP 8.0)**
Light profile	40 MHz	140 MB	---	27 Kbps	28.7 Kbps
Heavy profile	120 MHz	257 MB	4	68.5 Kbps	54.5 Kbps
Multimedia profile	320 MHz	196 MB	8	992 Kbps	710 Kbps

Figure 44: Capacity planning parameters and values for light, heavy and multimedia profiles.

In the capacity planning phase, assigning 70-80% of the server resources to clients avoids overload on the server side and reserves a room for scalability.

For VDI, the resource calculation is different than SBDV. VDI uses VMs instead of sessions which means that more resources are required. The resource allocation for VDI depends on the approach used. For persistent type of VDI, users have their own dedicated VMs, while the calculations differ for non-persistent VDI. VDI capacity planning is not in the scope of this research.

Conclusions and Future Work

Conclusions

This research has different objectives. The first objective is to deal with the different challenges and obstacles preventing the wider adoption of server-based desktop virtualization. The second objective is to provide recommendations to overcome those challenges. In order for server-based desktop virtualization to be implemented, it must deliver a user experience that is greater or at least equal the user experience offered by traditional desktop computers, while this is not achievable (at least for the time being) for all types of profiles, it is applicable for the majority of the profiles, in particular, heavy and light profiles. Since user experience is important, a complete chapter of this research focuses on capacity planning and server sizing. The following paragraphs summarize the conclusions of this research.

Desktop virtualization assessment is the first step for implementing server-based desktop virtualization. Desktop virtualization assessment is important for both the technical and the business decision making. From the technical perspective, data collected from the assessment is used to create technical design and resource capacity planning. From business perspective, assessment information among with the design analysis provide an understanding of the management and operations changes associated with virtualizing desktops [7].

No one solution fits all requirements. While SBDV works well with light and heavy profiles under predefined assumptions like application compatibility, SBDV doesn’t deliver the expected user experience for multimedia profiles for WAN scenario. VDI overcomes some of the limitations that SBDV have, like application compatibility and performance issues. But VDI has its limitations and challenges as well, which make its wider adoption unpredictable for the time being. One of the keys for the success of server-based desktop virtualization is choosing the right solution(s) based on the requirements.

As outlined earlier, VDI did not gain the expected adoption. The poor adoption has different reasons. One of the reasons is that VDI and desktop virtualization in general are considered to be similar to server virtualization. While this is true from the core technology perspective, VDI and desktop virtualization in general have other considerations. This research refers to those considerations as "psychological factor". End users are the main players in desktop virtualization. Another reason for the poor adoption is the network constraint, providing the same user experience provided by traditional desktops is difficult, especially for WAN scenarios.

This research focuses on server-based desktop virtualization for large-size and mid-size companies. The question is what about small companies and home users? Small companies and home users represent a worth-mentioning percentage of desktop computers worldwide. One of the proposed solutions for small companies and home users is Desktop as a Service (DaaS). With DaaS, desktop virtualization infrastructure is hosted at a Cloud Service Provider (CSP) datacenter(s), the CSP is responsible for managing the backend components (security, storage, backup, etc.). DaaS is not in the scope of this research, it is an emerging concept and a good topic for other students and researchers to conduct their researches on.

Future Work

Mobile Devices and Server-Based Desktop Virtualization

Mobile devices are widely adopted. This growth aims to serve the users need to “balance work and play” in a single device. Companies started adopting the concept of Bring Your Own Device (BYOD). BYOD increases the employees’ productivity, innovation and satisfaction while lowering the cost. While BYOD brings added values, it has challenges as well. The main challenge associated with BYOD is security. Leaving mobile devices unmanaged can lead to loss of control (governance and compliance), data leakage and other vulnerabilities.

Control and security for company-owned devices (desktop computers and laptops) are guaranteed by applying security and policies like antivirus solutions and Data Leakage Prevention (DLP). While for mobile devices (both company-owned and employee-owned), one option to achieve the required security and control is by Mobile Device Management (MDM) and Mobile Application Management (MAM). Containerization is one of the main features provided by MDM and MAM, its objective is to separate between personal and corporate information. Corporate information resides in a container where security controls like encryption and DLP are applied.

The question is where does Server-Based Desktop Virtualization fit in this evolution (mobile devices)? And what is the impact of this evolution of Server-Based Desktop Virtualization? From desktop virtualization perspective, mobile devices are one type of CADs. Using MDM and MAM, remote desktop clients can be published to mobile devices in a secure way. But unlike other types of CADs, the majority of mobile devices are with touch screen interfaces. Virtual desktops are intended for traditional access (keyboard and mouse), such access is not convenient for mobile devices.

Desktop virtualization vendors and are eager to enhance their infrastructure to provide better user experience of accessing virtual desktops from mobile devices. For instance, Microsoft redesigned its client OS (Windows 8) to support touch-capable devices.

Obviously, companies will end up having a mixed set of devices including mobile devices, thin clients and fat clients. Managing those devices will be complicated. This research proposes modifying the desktop virtualization infrastructure components and design to manage the mixed device complexity. The proposed modification are illustrated in the following paragraph.

CADs use RDCs to connect to the virtual machine or virtual session. The research proposes that the remote desktop client should be modified to pass information about the thin client to the connection broker and virtualization management servers. Information includes the client model and client operating system. If the client has a full version of OS, then the traditional virtual desktop interface will be displayed on the CADs. If the client is a tablet for instance, then it a touch-compatible virtual desktop interface will be displayed on the CADs. Such modification helps enhancing the user experience of accessing virtual desktops from different types of CADs.

Open Standard

The objective of standards is to guide the technology into the right direction and make it more responsive to the demand.

VESA proposed Net2Display as an open standard for RDP, as outlined earlier, the protocol was late in development and did not gain the expected adoption. A new standard with open specifications will help in making desktop virtualization more responsive to the end users' demands.

One of the challenges of VDI is the ambiguity of cost associated with implementing and operating it. Having a standard cost model will help in understanding and estimating the cost of implementing VDI.

Finally, having a standard benchmarking for the different desktop virtualization components (hypervisor, RDPs, etc.) will help in sizing, planning and choosing the right solution. Several researchers have built their own strategies and methodologies to test desktop virtualization components. For instance, *Benny Tritsch (www.drtritsch.com)* and *Shawn Bass (www.shawnbass.com)* have built their own methodologies to test and evaluate different RDPs.

HTML5

HTML5 is the fifth version of HyperText Markup Language (HTML), HTML is used for structuring and presenting web contents. HTML5 has several enhancements compared to its predecessor versions. From desktop virtualization stand point, the main feature of HTML5 is the canvas. Canvas is a resolution-independent bitmap canvas that can be used to render multimedia and visual files on fly. HTML5 compatible browsers can be used as a RDC to access virtual desktop. Vendors like Citrix and Ericom built an HTML5-based RDCs.

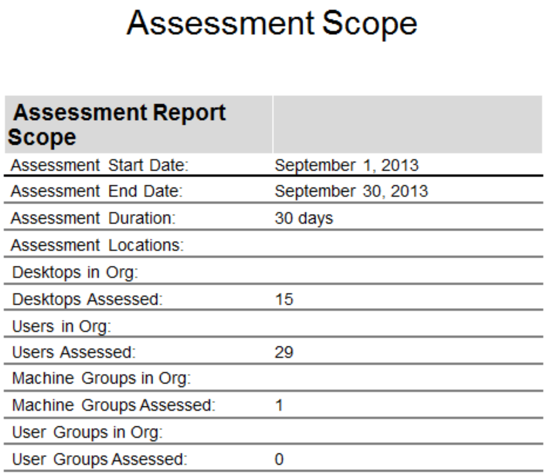
HTML5 has challenges to be tackled. The first challenge is that the old and legacy applications are not compatible with HTML5. Another challenge is that web browsers do not have full access to native devices (keyboard, mouse, printers, etc.), which makes the redirection and interaction with client side devices another challenge.

Appendixes

Appendix A: Desktop Virtualization Assessment Results.

The assessment is conducted on an international company based on Dubai. Data gathered from this assessment is used to feed the test profiles with realistic and accurate data. The assessment is conducted using a powerful tool from Quest (Quest VDI Assessment). This tool is chosen due to the detailed information that can be gathered from it like user login delay, application launch delay, application most used by time, desktop age and key metrics during average workday. Such information helps creating an accurate capacity plan and server sizing data. For example, the user experience (login delay and application launch delay) provides information about current delays in order to plan for a better performance or at least equivalent performance when implementing server-based desktop virtualization. While desktop age information helps defining the desktop procurement and disposal strategy (desktops that should be disposed and desktops that can be used with server-based desktop virtualization).

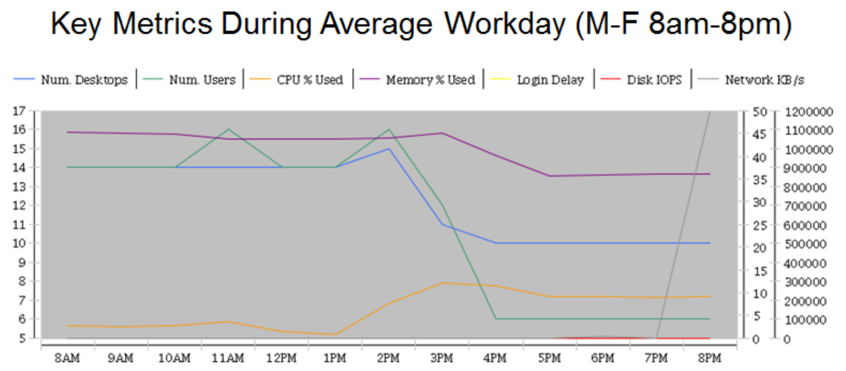
The assessment is conducted for a period of 1 month to gather as much data as possible (for large number of desktops with critical applications, the longer the assessment is conducted, the more accurate results are collected). The assessment targeted 15 desktop computers and 29 users. Desktop computers and users are selected from different departments (account, IT, HR, management and commercial). *Figures* below provides sample reports generated by Quest VDI Assessment tool.

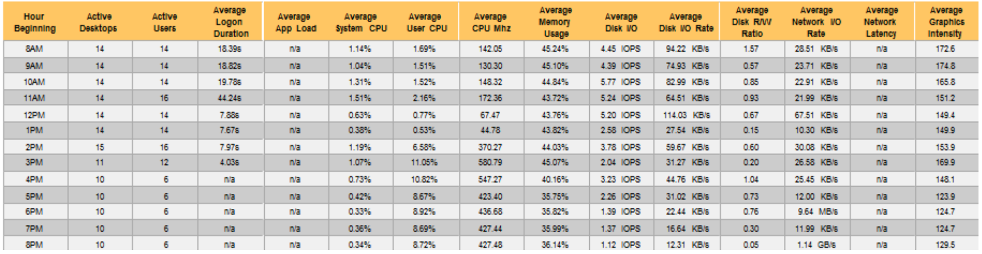


Assessment Scope

Assessment Report Scope	
Assessment Start Date:	September 1, 2013
Assessment End Date:	September 30, 2013
Assessment Duration:	30 days
Assessment Locations:	
Desktops in Org:	
Desktops Assessed:	15
Users in Org:	
Users Assessed:	29
Machine Groups in Org:	
Machine Groups Assessed:	1
User Groups in Org:	
User Groups Assessed:	0

Figure 45: Assessment Scope.





Hour Beginning	Active Desktops	Active Users	Average Logon Duration	Average App Load	Average System CPU	Average User CPU	Average CPU Mhz	Average Memory Usage	Average Disk I/O	Average Disk I/O Rate	Average Disk R/W Ratio	Average Network I/O Rate	Average Network Latency	Average Graphics Intensity
8AM	14	14	18.39s	n/a	1.14%	1.69%	142.05	45.24%	4.45 IOPS	94.22 KB/s	1.57	28.51 KB/s	n/a	172.6
9AM	14	14	18.82s	n/a	1.04%	1.51%	130.30	45.10%	4.39 IOPS	74.93 KB/s	0.57	23.71 KB/s	n/a	174.8
10AM	14	14	19.78s	n/a	1.31%	1.52%	148.32	44.84%	5.77 IOPS	82.99 KB/s	0.85	22.91 KB/s	n/a	165.8
11AM	14	16	44.24s	n/a	1.51%	2.16%	172.36	43.72%	5.24 IOPS	64.51 KB/s	0.93	21.99 KB/s	n/a	151.2
12PM	14	14	7.88s	n/a	0.63%	0.77%	67.47	43.76%	5.20 IOPS	114.03 KB/s	0.67	67.51 KB/s	n/a	149.4
1PM	14	14	7.67s	n/a	0.38%	0.53%	44.78	43.82%	2.58 IOPS	27.54 KB/s	0.15	10.30 KB/s	n/a	149.9
2PM	15	16	7.97s	n/a	1.19%	6.58%	370.27	44.03%	3.78 IOPS	59.67 KB/s	0.60	30.08 KB/s	n/a	153.9
3PM	11	12	4.03s	n/a	1.07%	11.05%	580.79	45.07%	2.04 IOPS	31.27 KB/s	0.20	26.58 KB/s	n/a	169.9
4PM	10	6	n/a	n/a	0.73%	10.82%	547.27	40.16%	3.23 IOPS	44.76 KB/s	1.04	25.45 KB/s	n/a	148.1
5PM	10	6	n/a	n/a	0.42%	8.67%	423.40	35.75%	2.26 IOPS	31.02 KB/s	0.73	12.00 KB/s	n/a	123.9
6PM	10	6	n/a	n/a	0.33%	8.92%	436.68	35.82%	1.39 IOPS	22.44 KB/s	0.76	9.64 MB/s	n/a	124.7
7PM	10	6	n/a	n/a	0.36%	8.69%	427.44	35.99%	1.37 IOPS	16.64 KB/s	0.30	11.99 KB/s	n/a	124.7
8PM	10	6	n/a	n/a	0.34%	8.72%	427.48	36.14%	1.12 IOPS	12.31 KB/s	0.05	1.14 GB/s	n/a	129.5

Figure 46: Key Metrics during Average Workday.

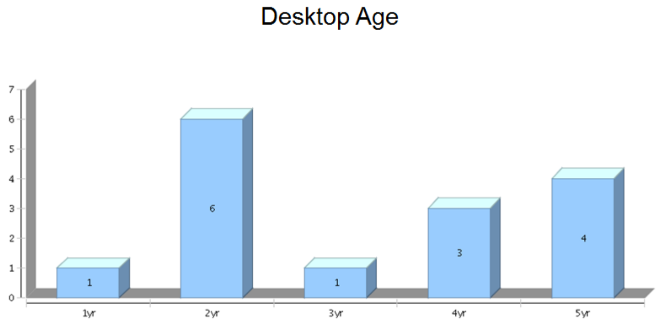


Figure 47: Desktop Age.

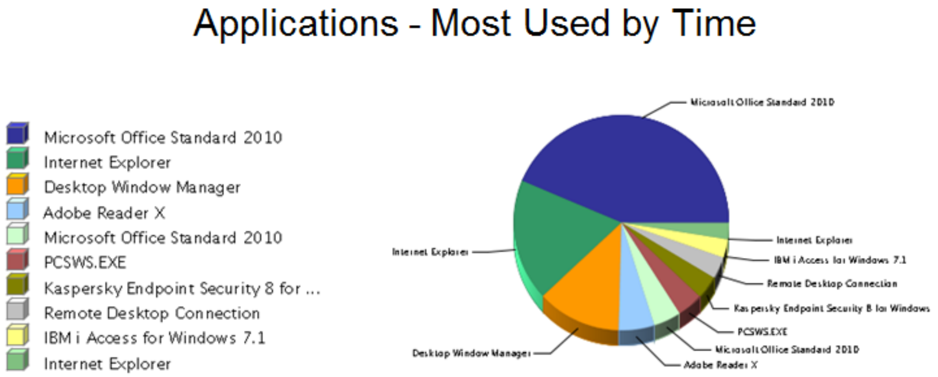


Figure 48: Applications, Most Used by Time.

Appendix B: Lab Measurements Components.

(1) Remote Desktop Session Host server: Windows Server 2012 with Remote Desktop Services role installed. Hosted on Amazon EC2 cloud (instance type: M1 large. Intel Xeon CPU E5-2650 @ 2.00 GHz, 7.5 G.B memory).

(2) CADs:

- Raspberry Pi: a credit-card sized computer. Raspberry Pi *Model B* is tested and used as a thin client.

  (Model B specs: 512 MB SDRAM, 8 G.B Micro SD hard disk, onboard 10/100 Ethernet, HDMI video output, Raspbian OS (based on Debian) and wireless 802.11 n USB adapter).

- Dell Laptop: Dell Latitude E5500, Intel Core 2 Duo / 2.26 GHz, 2 G.B RAM and 160 G.B hard disk and gigabit Ethernet, Windows 7 with RDP 7.0.

- Virtual machine: 2.26 GHz, 1 G.B RAM, gigabit Ethernet, Windows 8 with RDP 8.0.

(3) Test Controller server: the central control point for the load simulation testing. It controls the test parameters and defines the progression of the simulated user load. **[32].** Controller server is hosted on Amazon EC2 Cloud (instance type: M1 small. 1 Core CPU and 1.7 G.B RAM).

Appendix C: Customized Scripts.

Light Profile Script:

```
***************************************************************************************
'// Light User Profile Script
***************************************************************************************
'Global Settings
VK_RETURN   = 13
VK_LWIN     = 91
WINDOW_EVENT   = 1
MENU_EVENT     = 2
OBJECTSHOW_EVENT = 3
OBJECTFOCUS_EVENT = 4
VKeyFlag = 1
AltFlag = 2
CtrlFlag = 4
ShiftFlag = 8
'RDSH Server Name
Server = "RDSH"
'Username to connect to the RDSH Server. This is a variable name that is defined in the Remote Desktop Simulation tool
User = "Variable"
Password = "P@ssw0rd"
'Domain name for the RDSH Server
Domain = "RDSH"
'End of global settings
'// instantiate the RUIDCOM object
Set RUIDCOM = CreateObject ("RUIDCOM.RUI")
'// set connection properties
RUIDCOM.DesktopWidth = 800
RUIDCOM.DesktopHeight = 600
RUIDCOM.DesktopBpp = 32
RUIDCOM.TypingRate = 80
RUIDCOM.RedirectPrinters =1
'// Connect to Server
RUIDCOM.TSConnect Server, User, Password, Domain
WScript.Sleep (1000)
'--------------------------------------------------------------------------------------------
'// Open Run for Internet Explorer
RUIDCOM.VKeyDown VK_LWIN
RUIDCOM.PressKeyAndWaitForEvent "Open Run Dialog", asc("r"), 0, "Run", OBJECTSHOW_EVENT
RUIDCOM.VKeyUp VK_LWIN
WScript.Sleep (2000)
'Navigate to www.amazon.com
RUIDCOM.SendKey "iexplore.exe www.amazon.com"
RUIDCOM.PressKeyAndWaitForEvent "Open Explorer", VK_RETURN, VKeyFlag, "Explorer", WINDOW_EVENT
WScript.Sleep (5000)
RUIDCOM.VKeyDown VK_LWIN
RUIDCOM.PressKeyAndWaitForEvent "Open Run Dialog", asc ("r"), 0, "Run", OBJECTSHOW_EVENT
RUIDCOM.VKeyUp VK_LWIN
WScript.Sleep (5000)
'Navigate to www.cnn.com
RUIDCOM.SendKey "iexplore.exe www.cnn.com"
RUIDCOM.PressKeyAndWaitForEvent "Open Explorer", VK_RETURN, VKeyFlag, "Explorer", WINDOW_EVENT
WScript.Sleep (5000)
'--------------------------------------------------------------------------------------------
'// Open Microsoft Word 2013
RUIDCOM.VKeyDown VK_LWIN
RUIDCOM.PressKeyAndWaitForEvent "Open Run Dialog", asc("r"), 0, "Run", OBJECTSHOW_EVENT
RUIDCOM.VKeyUp VK_LWIN
WScript.Sleep (2000)
RUIDCOM.SendKey "winword.exe"
RUIDCOM.PressKeyAndWaitForEvent "Open word", VK_RETURN, VKeyFlag, "Word", WINDOW_EVENT
WScript.Sleep (5000)
RUIDCOM.SendKey " the text comes here "
WScript.Sleep (5000)
'--------------------------------------------------------------------------------------------
'//Open Microsoft Excel 2013
RUIDCOM.VKeyDown VK_LWIN
RUIDCOM.PressKeyAndWaitForEvent "Open Run Dialog", asc ("r"), 0, "Run", OBJECTSHOW_EVENT
RUIDCOM.VKeyUp VK_LWIN
WScript.Sleep (1000)
RUIDCOM.SendKey "excel.exe"
RUIDCOM.PressKeyAndWaitForEvent "Open Excel", VK_RETURN, VKeyFlag, "Excel", WINDOW_EVENT
WScript.Sleep (5000)
RUIDCOM.SendKey "Light Users"
```

```
WScript.Sleep (5000)
RUIDCOM.SendKey "Light Users"
WScript.Sleep (5000)
RUIDCOM.SendKey "Light Users from an excel sheet"
WScript.Sleep (5000)
RUIDCOM.SendKey "Light Users"
WScript.Sleep (5000)
RUIDCOM.SendKey "Light Users"
WScript.Sleep (1000)
'---------------------------------------------------------------------------------------------
'//Open home folder
RUIDCOM.VKeyDown VK_LWIN
RUIDCOM.PressKeyAndWaitForEvent "Open Run Dialog", asc ("r"), 0, "Run", OBJECTSHOW_EVENT
RUIDCOM.VKeyUp VK_LWIN
WScript.Sleep (5000)
RUIDCOM.SendKey "explorer.exe"
RUIDCOM.PressKeyAndWaitForEvent "Open Excel", VK_RETURN, VKeyFlag, "Excel", WINDOW_EVENT
WScript.Sleep (5000)
'---------------------------------------------------------------------------------------------
RUIDCOM.SendKey "y"
'End
*******************************************************************************
```

Heavy Profile Script:

```
,*******************************************************************************
, Heavy Profile Script
,*******************************************************************************
'// global settings
VK_RETURN   = 13
VK_LWIN     = 91
WINDOW_EVENT   = 1
MENU_EVENT     = 2
OBJECTSHOW_EVENT = 3
OBJECTFOCUS_EVENT = 4
VKeyFlag = 1
AltFlag = 2
CtrlFlag = 4
ShiftFlag = 8
Server = "RDSH"
User = "user10"
Password = "P@ssw0rd"
Domain = "RDSH"
'// End of global settings
'// instantiate the RUIDCOM object
Set RUIDCOM = CreateObject ("RUIDCOM.RUI")
'Resolution for heavy users profile: 1024x768
RUIDCOM.DesktopWidth = 1024
RUIDCOM.DesktopHeight = 768
'--------------------------------------------
RUIDCOM.DesktopBpp = 32
RUIDCOM.TypingRate = 160
RUIDCOM.RedirectPrinters =1
'// Connect to Server
RUIDCOM.TSConnect Server, User, Password, Domain
WScript.Sleep (1000)

'-----------------------------------------------------------------------------------------------
'// Open Run for Microsoft Outlook 2013
RUIDCOM.VKeyDown VK_LWIN
RUIDCOM.PressKeyAndWaitForEvent "Open Run Dialog", asc("r"), 0, "Run", OBJECTSHOW_EVENT
RUIDCOM.VKeyUp VK_LWIN
WScript.Sleep (2000)
RUIDCOM.SendKey "outlook.exe"
RUIDCOM.PressKeyAndWaitForEvent "Open Outlook", VK_RETURN, VKeyFlag, "Outlook", WINDOW_EVENT
WScript.Sleep (5000)
'-----------------------------------------------------------------------------------------------
'// Open Internet Explorer and Access www.amazon.com
RUIDCOM.VKeyDown VK_LWIN
RUIDCOM.PressKeyAndWaitForEvent "Open Run Dialog", asc("r"), 0, "Run", OBJECTSHOW_EVENT
RUIDCOM.VKeyUp VK_LWIN
WScript.Sleep (5000)
RUIDCOM.SendKey "iexplore.exe www.amazon.com"
RUIDCOM.PressKeyAndWaitForEvent "Open Explorer", VK_RETURN, VKeyFlag, "Explorer", WINDOW_EVENT
WScript.Sleep (5000)
'-----------------------------------------------------------------------------------------------
'// Open an existing word document and update it
RUIDCOM.VKeyDown VK_LWIN
RUIDCOM.PressKeyAndWaitForEvent "Open Run Dialog", asc("r"), 0, "Run", OBJECTSHOW_EVENT
RUIDCOM.VKeyUp VK_LWIN
WScript.Sleep (2000)
RUIDCOM.SendKey "C:\Files\Word.docx"
RUIDCOM.PressKeyAndWaitForEvent "Open word", VK_RETURN, VKeyFlag, "Word", WINDOW_EVENT
WScript.Sleep (5000)
RUIDCOM.SendKey "the text comes here"
WScript.Sleep (5000)
'-----------------------------------------------------------------------------------------------
```

```
'//Open an existing Microsoft Excel file and update it
RUIDCOM.VKeyDown VK_LWIN
RUIDCOM.PressKeyAndWaitForEvent "Open Run Dialog", asc("r"), 0, "Run", OBJECTSHOW_EVENT
RUIDCOM.VKeyUp VK_LWIN
WScript.Sleep (1000)
RUIDCOM.SendKey "C:\Files\Excel.xlsx"
RUIDCOM.PressKeyAndWaitForEvent "Open Excel", VK_RETURN, VKeyFlag, "Excel", WINDOW_EVENT
WScript.Sleep (5000)
RUIDCOM.SendKey "Light Users from an excel sheet"
WScript.Sleep (5000)
'---------------------------------------------------------------------------------------------
'//Open an existing Microsoft Powerpoint file and update it
RUIDCOM.VKeyDown VK_LWIN
RUIDCOM.PressKeyAndWaitForEvent "Open Run Dialog", asc("r"), 0, "Run", OBJECTSHOW_EVENT
RUIDCOM.VKeyUp VK_LWIN
WScript.Sleep (1000)
RUIDCOM.SendKey "C:\Files\PowerPoint.pptx"
RUIDCOM.PressKeyAndWaitForEvent "Open PowerPoiny", VK_RETURN, VKeyFlag, "PowerPoint", WINDOW_EVENT
WScript.Sleep (5000)
'---------------------------------------------------------------------------------------------
'//Open calculator
RUIDCOM.VKeyDown VK_LWIN
RUIDCOM.PressKeyAndWaitForEvent "Open Run Dialog", asc("r"), 0, "Run", OBJECTSHOW_EVENT
RUIDCOM.VKeyUp VK_LWIN
WScript.Sleep (1000)
RUIDCOM.SendKey "calc.exe"
RUIDCOM.PressKeyAndWaitForEvent "Open Calc", VK_RETURN, VKeyFlag, "Calc", WINDOW_EVENT
WScript.Sleep (5000)
'---------------------------------------------------------------------------------------------
'// Open Internet Explorer and Access www.amazon.com
RUIDCOM.VKeyDown VK_LWIN
RUIDCOM.PressKeyAndWaitForEvent "Open Run Dialog", asc("r"), 0, "Run", OBJECTSHOW_EVENT
RUIDCOM.VKeyUp VK_LWIN
WScript.Sleep (5000)
RUIDCOM.SendKey "iexplore.exe www.cnn.com"
RUIDCOM.PressKeyAndWaitForEvent "Open Explorer", VK_RETURN, VKeyFlag, "Explorer", WINDOW_EVENT
WScript.Sleep (5000)
'---------------------------------------------------------------------------------------------
'//Open an existing PDF File
RUIDCOM.VKeyDown VK_LWIN
RUIDCOM.PressKeyAndWaitForEvent "Open Run Dialog", asc("r"), 0, "Run", OBJECTSHOW_EVENT
RUIDCOM.VKeyUp VK_LWIN
WScript.Sleep (1000)
```

```
RUIDCOM.SendKey "C:\Files\pdf.pdf"
RUIDCOM.PressKeyAndWaitForEvent "Open PDF", VK_RETURN, VKeyFlag, "PDF", WINDOW_EVENT
WScript.Sleep (5000)
'-------------------------------------------------------------------------------------------
RUIDCOM.SendKey "y"
, end
*****************************************************************************************************************
```

Multimedia Profile Script:

```
'***************************************************************************************************
' Multimedia Profile Script
'***************************************************************************************************
'// global settings
VK_RETURN   = 13
VK_LWIN     = 91
WINDOW_EVENT   = 1
MENU_EVENT     = 2
OBJECTSHOW_EVENT = 3
OBJECTFOCUS_EVENT = 4
VKeyFlag = 1
AltFlag = 2
CtrlFlag = 4
ShiftFlag = 8
Server = "RDSH"
User = "variable"
Password = "P@ssw0rd"
Domain = "RDSH"
'// End of global settings
'// instantiate the RUIDCOM object
Set RUIDCOM = CreateObject ("RUIDCOM.RUI")
'Resolution for heavy users profile: 1024x768
RUIDCOM.DesktopWidth = 1024
RUIDCOM.DesktopHeight = 768
'-------------------------------------------
RUIDCOM.DesktopBpp = 32
RUIDCOM.TypingRate = 160
RUIDCOM.RedirectPrinters =1
RUIDCOM.RedirectDevices = 1
'// Connect to Server
RUIDCOM.TSConnect Server, User, Password, Domain
WScript.Sleep (1000)
```

```
'----------------------------------------------------------------------------------------------
'// Open Run for Microsoft Outlook 2013
RUIDCOM.VKeyDown VK_LWIN
RUIDCOM.PressKeyAndWaitForEvent "Open Run Dialog", asc("r"), 0, "Run", OBJECTSHOW_EVENT
RUIDCOM.VKeyUp VK_LWIN
WScript.Sleep (2000)
RUIDCOM.SendKey "outlook.exe"
RUIDCOM.PressKeyAndWaitForEvent "Open Outlook", VK_RETURN, VKeyFlag, "Outlook", WINDOW_EVENT
WScript.Sleep (5000)
'----------------------------------------------------------------------------------------------
'// Open Internet Explorer and Access www.amazon.com
RUIDCOM.VKeyDown VK_LWIN
RUIDCOM.PressKeyAndWaitForEvent "Open Run Dialog", asc("r"), 0, "Run", OBJECTSHOW_EVENT
RUIDCOM.VKeyUp VK_LWIN
WScript.Sleep (5000)
RUIDCOM.SendKey "iexplore.exe www.amazon.com"
RUIDCOM.PressKeyAndWaitForEvent "Open Explorer", VK_RETURN, VKeyFlag, "Explorer", WINDOW_EVENT
WScript.Sleep (5000)
'----------------------------------------------------------------------------------------------
'----------------------------------------------------------------------------------------------
'////video
RUIDCOM.VKeyDown VK_LWIN
RUIDCOM.PressKeyAndWaitForEvent "Open Run Dialog", asc("r"), 0, "Run", OBJECTSHOW_EVENT
RUIDCOM.VKeyUp VK_LWIN
WScript.Sleep (1000)
RUIDCOM.SendKey "C:\Files\video.mp4"
RUIDCOM.PressKeyAndWaitForEvent "Open Video", VK_RETURN, VKeyFlag, "Video", WINDOW_EVENT
WScript.Sleep (5000)
'----------------------------------------------------------------------------------------------
'// Audio
RUIDCOM.VKeyDown VK_LWIN
RUIDCOM.PressKeyAndWaitForEvent "Open Run Dialog", asc("r"), 0, "Run", OBJECTSHOW_EVENT
RUIDCOM.VKeyUp VK_LWIN
WScript.Sleep (2000)
RUIDCOM.SendKey "C:\Files\audio.mp3"
RUIDCOM.PressKeyAndWaitForEvent "Open Audio", VK_RETURN, VKeyFlag, "Audio", WINDOW_EVENT
WScript.Sleep (5000)
'----------------------------------------------------------------------------------------------
RUIDCOM.SendKey "y"
'**********************************************************************************************
'end
'**********************************************************************************************
```

Appendix D: Thin Clients Overview.

Thin client is a device that relies on the server to operate. Thin client is attached to a monitor, has different input devices (keyboard, mouse, etc.) with basic processing power to be able to interact with the server. It contains no moving parts like fans or hard disks (it may contain a flash memory to boot a light version of OS or a web access interface).

Old desktop computers can be used as thin clients, this process is known as desktop computers repurposing. Old desktop computers are repurposed by replacing the fat OS with a thin OS or completely removing the OS from the desktop computer. If a thin OS is installed on the repurposed clients, then it is recommended to apply policies to restrict the access only to the RDC installed on the thin client.

Thin clients play a crucial role in the server-based desktop virtualization. There are several advantages for thin clients illustrated in the following paragraph.

Accessing the server side from thin clients can be accomplished in different ways. One way is to have a light version of OS (thin OS) with RDC installed on it. With this approach, extra administration and management are required, this is because the thin client has an OS installed that should be patched. Another way to access the server from thin clients is by using PXE boot. With this approach, no OS is installed on the thin client, the thin client loads the packages required to establish the remote desktop (Example: LTSP), those packages are loaded from the server to the thin client memory. This approach guarantees the highest security assuming that the server is well-protected. Since the packages are loaded into the thin client memory, no residual data is stored on the thin clients. For instance, if a virus infected one of the thin clients, when the thin client reboots, a fresh copy of the boot package is downloaded to the thin client memory **[35] [36]**.

Another advantage of thin clients is cost reduction. Thin clients are cheaper than fat clients and consumes less power. A comparison between thin clients and fat clients is illustrated in the introduction of this research (*Figure 3: Desktops Power Consumption, Electricity cost and CO_2 Emissions)*.

Bibliography

[1] Brian Madden with Gabe Knuth and Jack Madden, *The VDI Delusion*, Burning Troll Productions, San Francisco, California, 2012.

[2] Wyse Technology Inc., *Environmental Benefits of Thin Computing*, CanyonSnow Consulting, Los Gatos, March 2009, 3-5.

[3] Danielle and others, *Microsoft Security Intelligent Report volume 14*, Microsoft, 2013, 5-9.

[4] Forrester Research, *375 Million Tablets Will Be Sold Globally In 2016*, Forrester, April 2012. www.forrester.com.

[5] Jack Madden, *Desktop Virtualization*, Margaret Rouse, November 2011. www.searchvirtualdesktop.techtarget.com/definition/desktop-virtualization

[6] Parallels, *Parallels VDI Solution White Paper,* Parallels Optimized Computing. April 2009.

[7] Michael Fox, *Demystifying the Virtual Desktop*, *Starting with Desktop Virtualization,* Michael Fox, USA, October 2010.

[8] John Savil, *Microsoft® Virtualization Secrets*, John Wiley & Sons, Inc. Indianapolis, 2012.

[9] John Savil, *Main components of VDI,* Windows IT Pro, December, 2008.

[10] Microsoft, *Microsoft System Center,* Microsoft TechNet. www.technet.microsoft.com/en-us/systemcenter.

[11] Paul Venezia, *Virtual Desktop Deep Dive* white paper, InfoWorld Media Group, 2013.

[12] Microsoft, *Windows Licensing for VDI, Quick Reference Guide*, Microsoft, 2010. www.partner.microsoft.com/Us/40171220.com.

[13] Microsoft, Microsoft Developer Network (MSDN), *Remote Desktop Protocol*, October 2013.

[14] Microsoft, *Remote Desktop Protocol (RDP)*, www.microsoft.com/en-us/legal/intellectualproperty/IPLicensing/Programs/RemoteDesktopProtocol.aspx.

[15] Gartner Group, *Gartner Quadrant for x86 Server Virtualization Infrastructure*, Gartner Group, June 2013.

[16] IDC Corporate, *Virtual Box Counters*, USA, www.idc.com.

[17] Warren Ponder and others, *VMware View Reference Architecture, A guide to large-scale Enterprise VMware View 3 and VMware View 4 Deployments*, VMware Inc. USA, 2010, 1-10.

[18] Alyssa Wood, *PCoIP (PC over IP),* Margaret Rouse, September 2012. www.searchvirtualdesktop.techtarget.com/definition/PCoIP-PC-over-IP.

[19] Citrix Systems Inc., *Citrix XenDesktop with Citrix Provisioning Server Best Practices,* Citrix Systems Inc., Florida, 2008, 3-8.

[20] Citrix Systems Inc., *Citrix HDX Technology white paper*, Citrix Systems Inc., Florida, 2013, 2-4.

[21] Jo Harder and Jason Maynard, *Technical Deep Dive: ICA Protocol and Acceleration*, Citrix Systems Inc., Florida.

[22] Jim McQuillan, *the Linux Terminal Server Project: Thin Clients and Linux,* USENIX Association, CA, USA, 2000.

[23] Brian Zammit, *Linux Terminal Server Project: Server Configuration Guidelines,* Systems Aligned Inc., 2004, 1-4.

[24] The Linux Information Project, *the X Window System: A Brief Introduction,* March 2006.

[25] David K. Johnson and others, *Hosted Virtual Desktops Versus Physical PCs: Understanding the Operational Cost Differences*, Forrester Research, January 2013.

[26] O3B Networks, Ltd., *What is Network Latency and Why Does it Matter*, Jersey, November 2008, 9-12.

[27] Cisco Systems, *Design Best Practices for Latency Optimization*, USA, 2007, 1-2.

[28] Shawn Bass & Bernhard Tritsch, *Microsoft RDP and RemoteFX, ICA/HDX, EOP and PCoIP: VDI Remoting Protocols Turned Inside Out*, Microsoft TechEd North America, Atlanta, 2011.

[29] Kenneth Ocheltree and others, *Net2Display™: A Proposed VESA Standard for Remoting Displays and I/O Devices over Networks*, VESA, 2006.

[30] VESA, *VESA Net2Display Remoting Standard, Version 1*, Milpitas, CA, October 2009, 32-34.

[31] Quest, *Getting Started Guide, Quest VDI Assessment*, Dell Software.

[32] Hammad Butt, *User Guide: Remote Desktop Load Simulation Tools,* Microsoft Corporation, 2010.

[33] Raspberry website, *Raspberry Pi specs*, element14 community, www.element14.com/community/community/raspberry-pi.

[34] IBM, *what is bring your own device?* , www.ibm.com/mobilefirst/us/en/bring-your-own-device/byod.html.

[35] 2X Software, *Thin Clients: Benefits and Savings using thin clients*, 2010.

[36] CDW·G, *Thin Clients Are in (Again)*, 2010, 2-3.

[37] Matrix42, *Survey at the 2011 Citrix Synergy*, Conference, San Francisco, 2011.

[38] VMware Inc., *Addressing Desktop Virtualization with VMware Virtual Desktop Infrastructure*, CA, USA, 2007.

[39] Tom Wall, *Virtualization and Thin Client: A Survey of Virtual Desktop environments*, Dublin Institute of Technology, 2009, 23-24.

[40] Christy Pettey, *Gartner Estimates ICT industry Accounts for 2 Percent of Global CO2 Emissions*, Gartner Group, Stamford, April 2007.

[41] A.T Kearney, *Crucial role of green IT*, USA, 2008.

[42] BullGuard, *PC Security*, *Computer Threats*, www.bullguard.com.

[43] Microsoft, *Microsoft System Center Virtual Machine Manager*, www.technet.microsoft.com/en-us/systemcenter.

[44] Microsoft, *Microsoft System Center Configuration Manager*, www.technet.microsoft.com/en-us/systemcenter.

[45] Microsoft, *Microsoft System Center Operations Manager*, www.technet.microsoft.com/en-us/systemcenter.

[46] Chris Wolf, *Desktop Virtualization Trends at Gartner Center*, Gartner Group, December 2012.

[47] Brian Madden, *Net2Display v1.0 spec is now released. No one has anything good to say about it*, BrianMadden.com, November 2009.

Glossary

VDI: Virtual Desktop Infrastructure. *A computing model that adds a layer of virtualization between the desktops and host servers.*

PXE: Pre-boot Execution Environment. *PXE allows workstations to boot from a networked server.*

TCO: Total Cost of Ownership. *The Cost that estimates the total direct and indirect costs of owning an asset.*

SBC: Server Based Computing. *The technology that aims to move the computing from the client side to the server side.*

MSIR: Microsoft Security Intelligent Report. *A report generated periodically by Microsoft, it analyses the threats, vulnerabilities and threats using data from the internet services and computers worldwide.*

WSUS: Windows Server Update Service. *A server application developed by Microsoft that allows administrators to manage and distribute Microsoft Products' updates and hotfixes.*

WDS: Windows Deployment Services. *A server application developed by Microsoft that enables the deployment of Windows Operating Systems over the network, eliminating the need for manual deployments using CDs and DVDs.*

SCCM: System Center Configuration Manager. *A server application developed by Microsoft that manages the deployment of applications and security of devices from a central interface across an enterprise.*

RDS: Remote Desktop Services. *Previously known as Terminal Services (TS), is a Microsoft Windows component used to access applications and sessions on a remote machine or server.*

TS: Terminal Services. *A Microsoft Windows component used to access applications and session on a remote machine or server. TS is renamed to Remote Desktop Services (RDS).*

LTSP: Linux Terminal Server Project. *LTPS is a free open source terminal server for Linux, it allows multiple users to access the same machine (server) using a terminal device known as thin client.*

VM: Virtual Machine. *It is a software computer that, like a physical computer runs an operating systems and applications, it has its own configuration files and settings and runs on top of a hypervisor.*

DRP: Desktop Remoting Protocol. *It is the protocol used to access virtualized desktops and sessions.*

ICA: Independent Computing Architecture. *It is a proprietary protocol for Citrix. ICA is used to passing the data and updates between servers and clients. ICA is the Desktop Remoting Protocol for Citrix.*

HDX: High Definition Experience. *HDX is built on top of Citrix ICA and is used to deliver high definition experience to end users.*

PCoIP: Personal Computing over Internet Protocol. *It is a proprietary protocol for VMware. PCoIP is a Desktop Remoting Protocol that was developed by a company named Teradici.*

CAD: Client Access Devices. *Any Client Device (Thin Client, Fat Client or Mobile Devices) used to access the sessions / virtual machines hosted on the server side.*

DMZ: Demilitarized Zone. *A computer host or a small network residing between the company's internal network and the internet. It is used to secure the internal network from externals by avoiding external users' direct access to the internal network.*

RDC: Remote Desktop Client. *It is a piece of software that has the configuration and parameters for the server to which the clients connect. Examples of RDC are Citrix Receiver and Microsoft Remote Desktop Connection (mstsc.exe).*

HTTPS: Hypertext Transfer Protocol Secure. *A TCP protocols which operate by default using port 443. HTTPS is used to secure the data and web traffic between the client and the server.*

SSL: Secure Socket Layer. *It is a standard security technology for establishing an encrypted channel between the web server and the client browser. It uses certificates to encrypt the traffic and verify the identity of the web server.*

TLS: Transport Layer Security. *TLS is the successor to the Secure Socket Layer (SSL).*

DHCP: Dynamic Host Configuration Protocol. *It is a client-server protocol that allows clients to obtain TCP / IP configuration information from a DHCP server.*

DNS: Domain Name Servers. *It is a standard protocol that translates internet domain and host names to internet protocol addresses.*

VHD: Virtual Hard Disk. *It is a file format that represents a virtual hard disk drive.*

SAN: Storage Area Network. *It is a high-speed network of storage devices. SAN provides a block-level storage.*

NAS: Network Attached Storage. *It is a file-level data storage, NAS is normally connected to computers network and used to provide access to different clients and computers in the network (mainly used as a file server).*

SCVMM: System Center Virtual Machine Manager. *It is a Microsoft product which is part of Microsoft System Center Suit, SCVMM is used intended to manage virtualized and cloud-based environments.*

SCOM: System Center Configuration Manager. *It is a Microsoft product which is part of Microsoft System Center Suit, SCOM is used to monitor the servers and the IT infrastructure.*

SIDs: Security Identifiers. *It is a unique value of variable length used to identify a trustee.*

IOPS: Input Output Operations per Second. *IOPS is a common performance metrics used to benchmark storage and hard disks performance and throughput.*

SA: Software Assurance. *It is a term that used to refer to the level of confidence that a software is free of vulnerabilities. Microsoft's definition of software assurance is an option (package) that allows companies and individuals to upgrade their current software or application at no additional cost.*

VDA: Virtual Desktop Access. *When it comes to Microsoft, VDA is a license option that allows accessing virtual desktop environments from software assured PCs and allows users to remotely access their virtual machines from a third-party devices such as tablets.*

ERR: Extended Roaming Rights. *When it comes to Microsoft, ERR is the ability for users to access their Virtual Desktop Infrastructure desktops from non-corporate machines outside the corporate domain.*

GDI: Graphics Device Interface. *It is a Microsoft Windows Application Programming Interface (API) which is responsible for representing graphical objects and transmitting them to output devices.*

GPU: Graphical Processor Unit. *It is a computer chip that performs complex mathematical calculations, mainly for the purpose of image rendering.*

DVI: Digital Visual Interface. *It is a common digital video cable used for desktops and LCD monitors. It has 24 pins and support both analog and digital video.*

TFTP: Trivial File Transfer Protocol. *It is a simple form of file transfer protocol (ftp), TFTP uses UDP and has no security features. TFTP is often used by diskless workstations to boot from the network, as well it is used for routers to upgrade their OS.*

SSD: Solid State Disks. *SSD refers to disk that are built entirely from semiconductors (no moving parts). The storage of SSD is handled by flash memory chips. It has less power consumption, faster access and higher reliability compared to traditional disks with moving parts.*

QoS: Quality of Service. *It is an industry standard used to ensure high quality performance for critical applications.*

AD: Active Directory. *It is Microsoft's trademark directory service. AD is a centralized and standardized used to automate network management for user data, security and resources.*

DLP: Data Leakage Prevention. *It refers to the solution used to detect potential data breach and prevent such breach by monitoring, applying different policies, and blocking sensitive data while in-use, in-motion and at-rest.*

XPS: XML Paper Specification. *It is an open specification for page description language and fixed-document format. XPS was developed by Microsoft in June 2009 as a replacement for PDF file format.*

Table of Figures