AASCIT | **AMERICAN ASSOCIATION for Science and Technology**

A new approach for application software's and enterprise virtual machines based on virtualization

Kiranmayi Bodapati[1, *], Nurul Zulfadhli Bin Mohd[1], Nur Salha Binti Md Seth[1], Srilakshmi Vishnumukkala[2]

[1]School of Information Technology, Nirwana Education Group, Kualalumpur, Malaysia
[2]Department of Computer Science Aditya Engineering College, Andhra Pradesh, India

Email address
bkm2906@gmail.com (K. Bodapati)

Abstract

In client computing environment, there is a tight coupling between hardware and software, it makes every PC an individual managed entity. Virtualization allows managing virtual machines, application software's and data to be put into virtual layers. This approach decouples hardware, operating system, application software's and data. We propose a new approach to distributing, executing and managing the virtual machines application software's based on virtualization. This approach increases scalability and reduce the cost in managing hundreds of computers in a network. So our method use the desktop as a personal computer without modifying the hardware resource and without interfering with other applications in a safe state.

1. Introduction

In client computers, several application software's with specific versions are to be installed to perform the user's task. And some of the applications require add-ons and plugins. Users may set the configurations for their applications to satisfy the user preferences. For some of the applications are very specific to Operating system, In that case, client need to maintain multiple operating systems. Some applications do not work well under current status of user's personal computer without cause. Frequently changes of application environment like installation and uninstallation makes an application unstable. Moreover an online automated software deployment and update may threaten the performance and security. These issues are responsible for maintenance, security, and migration challenges. So the Total Cost of Ownership (TCO) and budget are increased rapidly.

Desktop virtualization, software virtualizations allows operating system, applications, and data to be put into virtual layers. Application Virtualization can make applications available to end user computers without having to install the applications directly on those computers. This is made possible through a process known as sequencing the application, which enables each application to run in its own self-contained virtual environment on the client computer. The sequenced applications are isolated from each other. This eliminates application conflicts, but the applications can still interact with the client computer.

The App-V client is the feature that lets the end user interact with the applications after they have been published to the computer. The client manages the virtual environment in

which the virtualized applications run on each computer. After the client has been installed on a computer, the applications must be made available to the computer through a process known as publishing, which enables the end user to run the virtual applications. The publishing process copies the virtual application icons and shortcuts to the computer and also copies the package definition and file type association information to the computer. Publishing also makes the application package content available to the end user's computer.

2. Related Work

Software virtualization is also known as application virtualization. It allows applications and data to be put into virtual layers instead of being installed to the base file system and registry[l]. Application virtualization is an umbrella term that describes software technologies that improve portability, manageability and compatibility of applications by encapsulating them from the underlying operating system on which they are executed [7]. It allows applications to run in environments that do not suit the native application. It also improves security, by isolating applications from the operating system. To do that, software virtualization isolates the application from the various resources such as files, registry, and so on. At execution time, software virtualization installs an application at the virtual layer.

There are some vendor-specific implementations. VMware ThinApp[2], Software Virtualization Solution[3], AppStream[4], PortableApps[5], AutoIt[6] and Microsoft Application Virtualization[8] are examples of application virtualization.

Microsoft Application virtualization known as APP-V is a solution that generates a single execution file that is isolated from the OS. When a user runs this single execution file, it performs file and registry I/O by using application level virtualization. Software Virtualization Solution[3] allows the user to install an application without modifying the configuration. To do that, Software Virtualization Solution captures all changes made during installation of a particular application in what's called a Virtual Software Package. Symantec Workspace Streaming[4] as known as AppStream[4] adds a streaming function to the software virtualization function. It enables software to be executed during on-line transmission time.PortableApps[5] provides packaged software suitable for use on portable media, without leaving settings files on the host computer using NSIS script. NSIS script is a criptdriven Windows installation system with minimal overhead backed by Null soft. Autoit [6] is a freeware ASIC-like scripting language designed for automating the Windows OU! And general scripting. It uses a combination of simulated keystrokes, mouse movement and indow/control manipulation in order to automate tasks in away. VMware Thinapp known as thin install enables organizations to respond to the management challenge by providing the capability to make applications available to end-user computers without having to install the applications directly on those computers.

Software management is one of the most time-consuming aspects of managing an enterprise network environment. As described above, there are various promising software virtualization technologies. Software virtualization technologies offer several new technologies to reduce the complexity of managing applications in enterprise environments. However, virtualized application has no rights to update itself manually; the user may have restrictions on the use of application.

3. Application Software's Based on Application Virtualization

Application Virtualization enables you to deploy, update, and support applications as services in real time, on an as-needed basis. When you use App-V, you transform individual applications from locally installed products into centrally managed services. Applications become available everywhere they need to be—no computer pre-configuration or changes to operating system settings are required. The follow fig:1 shows Microsoft Application Virtualization architecture.
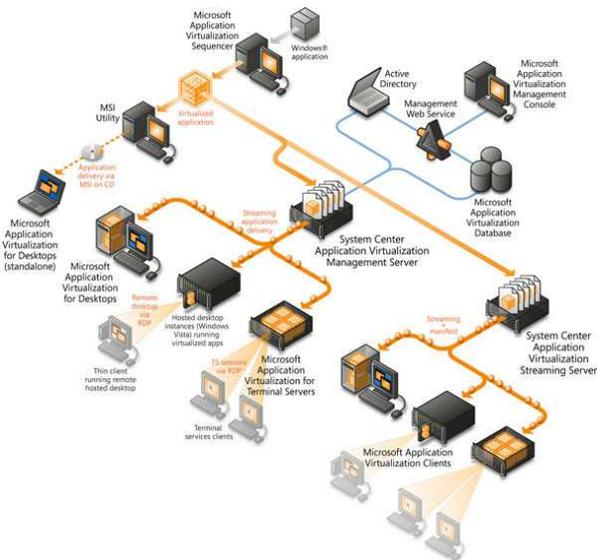

Fig. 1. *Architecture of Microsoft Application Virtualization*

3.1. Application Virtualization Management Server

The App-V Management Server both publishes and delivers sequenced applications on-demand to the App-V Desktop Client and the Remote Desktop Services Client. The App-V Management Server uses Microsoft SQL Server® for its data store, and one or more App-V Management servers can share a single SQL data store. The App-V Management Server authenticates requests and provides the security, metering, monitoring, and data gathering that you need. The

server uses Active Directory and supporting tools to manage users and applications.

The App-V System also includes the App-V Management Console and Management Web Service. Administrators use the App-V Management Console to configure the App-V environment. Using the App-V Management Console, administrators can add and remove applications, change File Type Associations (FTAs), and assign access permissions and licenses to users and groups. The App-V Management Web Service is the communication conduit between the App-V Management Console and the SQL data store.

3.2. Application Virtualization Desktop Client

The App-V Windows Desktop Client automatically sets up and manages virtual environments for App-Vsequenced applicationsand publishes the applications to the user's desktop. The App-V Client stores user-specific virtual application settings in each user's profile—for example, registry and file changes. The following Fig: 2 shows the architecture of the APP-V Client.
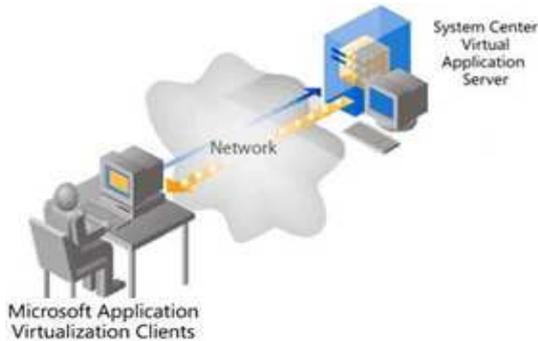

Fig. 2. APP-V Client architecture

3.3. Application Virtualization Sequencer

The App-V Sequencer is a wizard-based tool that administrators use to transform traditional applications into virtual applications. The Sequencer produces the application "package," which consists of several files. These files include a sequenced application (.sft) file, one or more Open Software Description (.osd) application configuration files, one or more icon (.ico) files, a manifest xml file that can be used to distribute sequenced applications with electronic software delivery (ESD) systems, and a project (.sprj) file. Optionally, the Sequencer can generate a Windows Installer file (.msi) that can be deployed to clients configured for standalone operation. All files are stored in a shared content folder on the Management and/or Streaming Server and are used by the App-V Client to access and run sequenced applications.

3.4. Application Virtualization Streaming Server

This App-V Streaming Server has streaming capabilities, including active/package upgrade without the Active Directory or SQL Server requirements. However, it does not have a publishing service, or licensing or metering capabilities. The publishing service of the App-V Management Server is used in conjunction with the App-V Streaming Server, so the Management Server configures the application but the Streaming Server delivers it (usually in branch offices). The following Fig: 3 shows the architecture of streaming server.

Fig. 3. Streaming server architecture

4. Enterprise Virtual Machines Based on Virtualization

When upgrading to a new version of Operating systems, enterprises must first inventory and test line of business applications on the new operating system and your organization may have some applications that are not yet officially supported by your vendor, or might not work at all despite all efforts.

This whole process—testing, fixing the application, upgrading to a new version that supports Windows 8, or finding an alternative application can be time-consuming. Meanwhile, users are unable to take advantage of the operating system's new capabilities and enhancements, and IT departments have to delay upgrade plans. Microsoft Enterprise Desktop Virtualization (MED-V) can help ease these challenges.

MED-V removes the barriers to Windows upgrades by resolving application incompatibility with Windows 8 and delivering applications in a Windows XP-based application compatibility workspace. Upgrades can proceed on schedule, and users can take advantage of the power of Windows 8 right away without losing access to applications they need while IT departments can remediate incompatible applications.

MED-V enables users to seamlessly start the legacy applications right from the Windows Start. Applications appear and operate as if they were installed on the desktop. They can even be pinned to the Windows 8 taskbar! In MED-V 2.0 legacy applications share seamless access to users' documents and network-available printers, and even USB devices such as flash storage or Smartcard readers.

MED-V is easy to deploy and manage. MED-V 2.0 integrates with existing management and deployment systems, such as System Center Configuration Manager, for

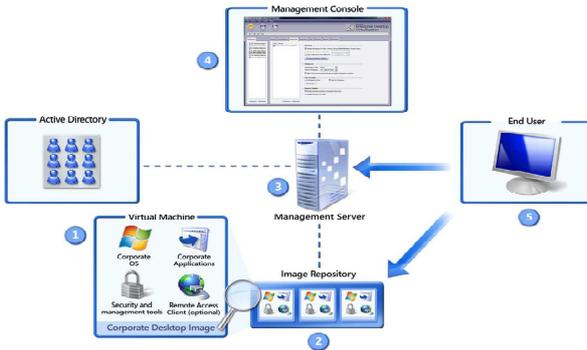easy, scalable, enterprise deployment. The following Fig: 4 shows the High level architecture of MED-V.

Fig. 4. *High level architecure of MED-V*

4.1. Microsoft Enterprise Desktop Virtualization Design Process

The following decisions and activities that must occur in planning the design for Microsoft Enterprise Desktop Virtualization (MED-V). The four steps that follow represent the most critical design elements in a well-planned MED-V design.

The decision flow below provides a graphic overview of the steps involved in designing a MED-V infrastructure (Fig: 4.1)
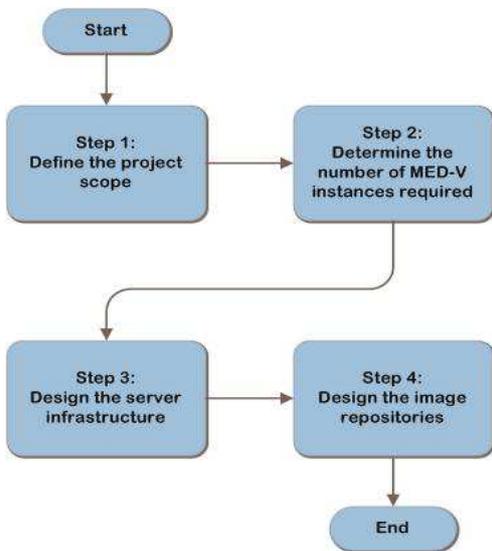

Fig. 4.1. *Steps involved in designing a MED-V infrastructure*

- Step 1: Define the Project Scope
- Step 2: Determine the Number of MED-V Instances Required
- Step 3: Design the Server Infrastructure
- Step 4: Design the Image Repositories

Some of these items represent decisions that must be made. Where this is the case, a corresponding list of common response options will be presented. Other items in this list represent tasks that must be carried out. These types of items are addressed because their presence is significant in order to complete the infrastructure design.

4.2. Enterprise VMs Management Server

The MED-V Management Console (Management.exe) allows you to manage the MED-V Client Policy, Image Distribution, and Reporting. The MED-V Management Console is bundled with the MED-V Client and can be installed with or without the MED-V Client.

The operating system requirements for the console align with the client requirements and this is what prevents the MED-V management console (the management.exe utility) from being installed on a server-based operating system.

I am often asked why this is the case. The philosophy behind this is the fact that MED-V revolves around the lifecycle management of desktop virtualization and application publishing (through desktop virtualization). The images and policies are configured and tested from the host environment the majority of the time.Since the tools leverage the packaging of images and the testing and designation of test images (importing, tweaking, etc.) it makes more sense to align it with the client so a MED-V Administrator can:

1. Prepare the VPC Image
2. Import the Image and test it against a policy
3. Pack and upload/import an image

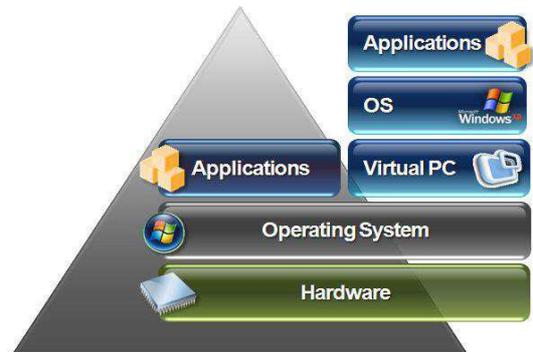All this is occurring while working on the same host computer (Fig: 4.2)

Fig. 4.2. *Architecture of Enterprise VMs Management Server on same host computer*

4.3. Enterprise VMs Desktop Client

The MED-V client requires authentication to ensure that only authorized users access the MED-V virtual images. This verification is performed against the management server, which queries Active Directory for user and group information. Therefore, the management server must be part of the domain to which the user is trying to authenticate.

MED-V leverages Active Directory security policies. When an account is disabled or locked in Active Directory (for instance, if the user typed a wrong password three times), the user is not allowed to start the MED-V workspace. In addition, if the password is about to expire, the user is offered to change the password before completing the MED-V authentication.

Once the authentication process is complete, the MED-V

client queries the MED-V management server for the most recent policies and settings. This action ensures that the endpoint is using the most updated MED-V workspaces and allows administrators to control and monitor active users, as described in following sections.

The domain credentials used for authenticating the MED-V client are also used to login to the Windows instance inside the virtual machine, so that the user is not required to type the domain credentials twice. The user may choose to save the credentials for future sessions of the same user, so that they are automatically used by MED-V client the next time the user attempts to start a MED-V workspace. Note that the user is required to authenticate by the MED-V client, even if the host and the guest operating system use the same credentials.

When the virtual machine is running in a MED-V session, it is locked after a predefined idle time or when the physical device enters hibernation or sleep mode. The authenticated user is required to type his password to unlock the virtual machine and continue working.

MED-V uses Microsoft Virtual PC to run a virtual machine locally on the endpoint. The MED-V client controls all aspects of virtual machine management, including retrieving or updating a virtual image, customizing the virtual machine for the specific user or device, initiating, suspending or terminating a virtual machine session, and monitoring the virtualization engine for malfunctions (watchdog). End users remain unaware that a virtual machine is running in the background.

The MED-V client can be configured to take a snapshot of the virtual machine (similar to a laptop sleep mode) at the end of each work session or when the user logs off the host workstation. This reduces the time required for re-initiating the virtual machine.

When a virtual machine is suspended, and a different, authorized user attempts to use it, the virtual machine shuts down first (pending user confirmation). The existing session is lost, similar to Windows behavior when an authorized user attempts to access a locked device (Fig: 4.3).

Fig. 4.3. Architecture of virtual machine when an authorized user attempts to use a locked device

5. Conclusions

In this paper, we proposed new approach for distribution, execution and management of the virtual machines and application softwares, data based on virtualization.

Virtualization allows VMs , Application software's and data into virtual layers. This decouples the hardware software and data. There are some vendor-specific implementations of software virtualization. Previous techniques have no rights to update themselves; the user may have restrictions on the use of application. Our proposed method, customized application, has a well-informed area to allow the user to write some data at a specific folder and to restore the initial state of customized application. We define customized application and show the architecture of it. Customized application is a single execution file that is made of application execution logic and application repository. Customized application generates the virtual boundary of application to separate and isolate the hardware resources at the execution time. Unlikely general Windows's application, customized application calls the system libraries through system library filter and guarantees the secured call proposed method was very helpful because it made applications runeven when they couldn't run because of older versions or other software that was denying the software to run easily.

Acknowledgments

References

[1]   http://en.wikipediaorg/wiki/Software_Virtualization _Solution

[2]   VMware ThinApp, http://www.thinstall.com/

[3]   Software Virtualization Solution,

[4]   http://www.altiris.com/Products/SoftwareVirtualizationSolutio n.aspx

[5]   AppStrearn, http://www.appstrearn.com

[6]   PortableApps, http://portableapps.com/

[7]   AutoIt V3, http://www.autoitscript.com/

[8]   http://en.wikipedia.org/wiki/Application_virtualization

[9]   http:/www.microsoft.com/ipd

[10] Won-Hyuk Choi, Won-Young Kim, Ji-Hoon Choi, Wan Choi, "A Design and Implementation of Application virtualization method using virtual supporting system and Copy-on-Write Scheme", Conference of Korean Contents Association, 2007.

[11] Won Hyuk Choi, Won-Young Kim, Wan Choi , " Design and Implementation of Portable software converting method using Software installation information extraction tool and AutoIt Script", Conference of Korean Society for Internet Information , 2008.

[12] WonHyuk Choi, Won-Young Kim, Sung-Jin Hur , "A Method of SW Installation and Execution Information Extraction using Virtual Execution Handler and Multiple Virtual Execution Environments on Dirty PC Environment," 1st International Conference on Internet (ICONI) 2009.

[13] Personalized SW service platform Youfree, http://www.youfree.or.kr

[14] Kyong-I Ku, Won-Hyuk Choi, Moonyoung Chung, Kiheon Kim, Won-Young Kim, Sung-Jin Hur , Method for Distribution, Execution and Management of the Customized Application based on Software Virtualization

[15] César S. Aguiar, Daniel I. Cruz, Roberta S. Ulson, Marcos A. Cavenaghi , The application of distributed virtual machines for enterprise computer management: a two-tier network file system for image provisioning and management