Maintaining HMI and SCADA Systems Through Computer Virtualization

Jon Reeser
Systems Engineer
PROFI-VISION Automation
1150 Glenlivet Drive
Allentown, PA 18106
jon.reeser@profi-vision.com

Thomas Jankowski
President
PROFI-VISION Automation
1150 Glenlivet Drive
Allentown, PA 18106
thomas.jankowski@profi-vison.com

Greg M. Kemper
Senior Member, IEEE
Director Project Engineering
Lehigh Hanson Inc.
7660 Imperial Way
Allentown, PA 18195
gkemper@htcnam.com

Abstract - This paper examines computer virtualization as a solution to issues encountered in maintaining HMI and SCADA systems based on commercial off the shelf computing hardware. Following a brief primer, the paper outlines issues arising from the rapid advancement of personal and server computing hardware, including automation software conflicts with new operating systems, device driver availability for old operating systems, and migrating backup images to new hardware. Opportunities for improvement are identified in backup and recovery, operator access to computing resources, and prototyping HMI and program changes. The concept of a virtual machine is introduced and common architectures are examined, followed by the description of an example virtualized control room. Finally, the paper examines the experiences of a multi-site cement producer in implementing virtualization in two of their plants. This examination explores the reasoning, advantages, disadvantages, and limitations encountered in moving existing systems to a virtualized platform.

Index Terms – Architecture, Host, Hypervisor, Network Attached Storage, SCADA, Server Hardware, Storage, Thin Clients, Virtual Desktop Infrastructure, Virtual Machine, Virtualization

I. NOMENCLATURE

ATA - Advanced Technology Attachment
CPU – Central Processing Unit
HMI – Human Machine Interface
MTBF – Mean Time Before Failure
OEM – Original Equipment Manufacturer
PC – Personal Computer
RAM – Random Access Memory
SAS - Serial Attached SCSI
SCADA – Supervisory Control and Data Acquisition
SCSI - Small Computer System Interface
VDI – Virtual Desktop Infrastructure
VM – Virtual Machines
X86 - Represents any 8086 compatible CPU

II. INTRODUCTION

Aging computer hardware presents a number of maintenance issues. First there is the availability of replacement parts. These are often in limited supply or only available in used or refurbished condition. Parts that are readily available, such as Ethernet cards, may not have device drivers for older operating systems.

With low new PC prices the idea of completely replacing an old PC is an attractive alternative to repair. Given that the mean time between failures (MTBF) for the average PC is 25,000 hours [1], just shy of three years, process critical PCs may be scheduled for replacement as part of standard maintenance. As control system software packages are tied to a small set of operating systems, migrating to a new server requires an older operating system to

978-1-4799-3294-8/14/$31.00 © 2014 IEEE

be installed on the new PC. This often results in the PC having devices without drivers and may result in suboptimal performance.

Installing old operating systems on new hardware is further hindered by operating system availability. Retail packages for older operating systems are in limited supply. The license agreement of current, released to retail, operating systems may or may not include downgrade rights. The operating system installed on an old PC being replaced may not be transferable to a different PC. This is common with discounted OEM installed operating systems bundled with PC's.

These issues may be addressed through computer virtualization.

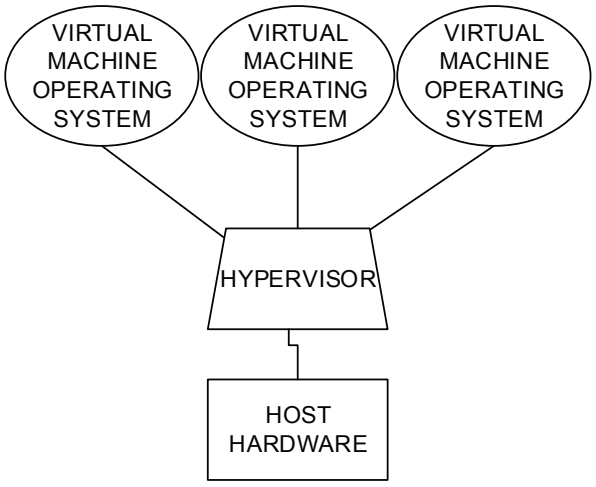
III. TECHNOLOGY DISCUSSION

A. The Virtual Machine

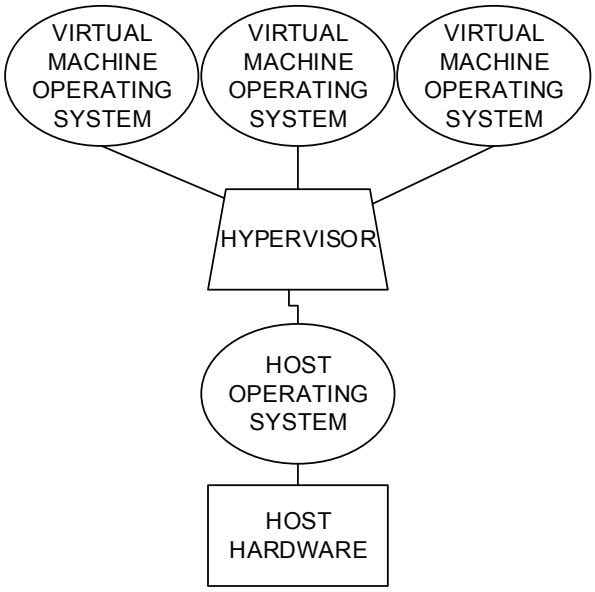
Virtualization refers to a variety of technologies. This paper focuses on platform virtualization, also referred to as hardware virtualization and operating system virtualization. Platform virtualization is the execution of one or more virtual machines on a shared physical host computer. Each virtual machine represents a complete, isolated, computer with its own operating system and software. The layer responsible for providing the virtual computer abstraction is the virtual machine monitor [2] also referred to as a Hypervisor.

Hypervisors are divided into two categories; Type 1 and Type 2. In Type 1, a single Hypervisor executes directly on the hardware and all operating systems run on virtual machines managed by the manager. In Type 2, a traditional operating system runs directly on the hardware and executes the Hypervisor. In this case, the Hypervisor executes as a regular program on the host operating system. A Type 2 monitor is subject to the host operating system's scheduling and resource management. Additionally, a crash due to a fatal error in the host operating system causes the Hypervisor and its virtual machines to be reset. In both types, a virtual machine crash due to a fatal error in a virtual machine is isolated to the virtual machine in question.

Software vendors typically view virtual machines as independent machines for licensing purposes. If a Hypervisor is running ten HMI client virtual machines, each of the ten virtual machines has its own operating system license and HMI software license.



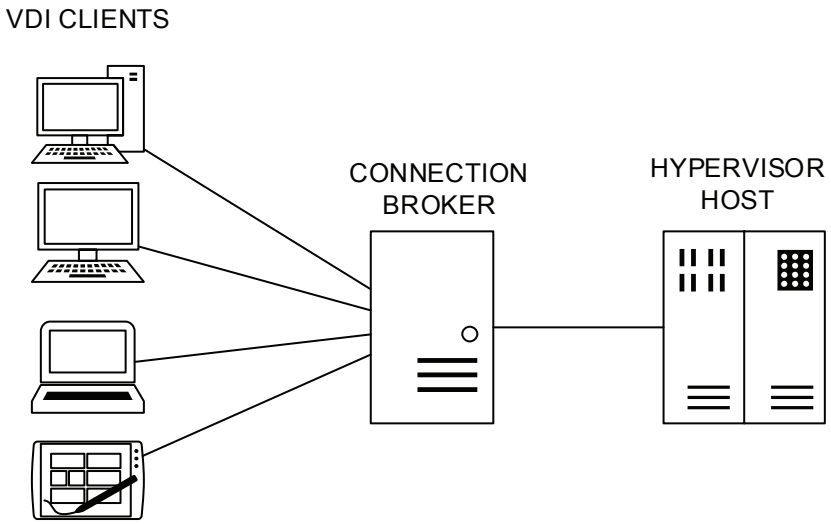
Type 1 Hypervisor



Type 2 Hypervisor

B. Virtual Desktop Infrastructure

Virtual Desktop Infrastructure implementations enable end user interaction with virtual machines. A typical implementation contains multiple end user virtual machines running on Type 1 Hypervisors. End users may remotely view the virtual machines via the VDI implementation's remote access protocol. Since a user's programs are executed on the virtual machine, the end user hardware's role is limited to display and user input. This decoupling allows for a wide variety of end user hardware, such as standard PCs, Thin Clients, and mobile devices. The VDI end user client connects to a connection broker, which handles authentication and manages the connection between the VDI client and the virtual machine.



C. Virtualization Advantages

The abstraction between physical hardware and the devices of a virtual machine extends the life of older operating systems and software. The virtual machine's operating system is unaware of the physical hardware. For example, the virtual machine may have a parallel ATA hard drive, while the host has a new SAS drive and controller that is unsupported by the virtual machine's operating system.

Server and PC resources are typically underutilized. A Type 1 Hypervisor is able to efficiently execute multiple virtual machines. This allows multiple servers or PCs to be consolidated, reducing the number of required physical servers.

Hypervisors typically store virtual machine's hard drives as a set of files, allowing for backup and restore to be a simple file copy process. This portability also allows a virtual machine to be easily moved from one physical host machine to another and allows for the creating of point in time snapshots.

Many Hypervisors allow for snapshots, cloning, or migration to another host machine or live virtual machines. Live migration minimizes downtime when transferring a virtual machine from one host to another.

VDI allows for Thin Clients to be used in place of PCs for operator stations. Thin Clients improve system reliability through a higher mean time between failures and use significantly less energy. Operating costs and downtime are further reduced as the time to replace a faulty Thin Client is limited to the time required to swap devices, configure its network access, and configure the connection to the VDI broker. All software runs in the virtual machine, which remains intact on the server, eliminating the need to rebuild or restore a PC's software environment.

Provisioning a new virtual machine does not require additional hardware.

978-1-4799-3294-8/14/$31.00 © 2014 IEEE

D. Virtualization Disadvantages

X86 virtual machine implementations focus on the standard PC components. This makes servers with specialized (non-Ethernet) hardware for field bus communication poor candidates for virtualization. Some virtual machine implementations allow for hardware pass-through. This results in a virtual machine that is partially physical, the Hypervisor no longer completely manages the virtual machine. This may neutralize some advantages, such as portability and live backup.

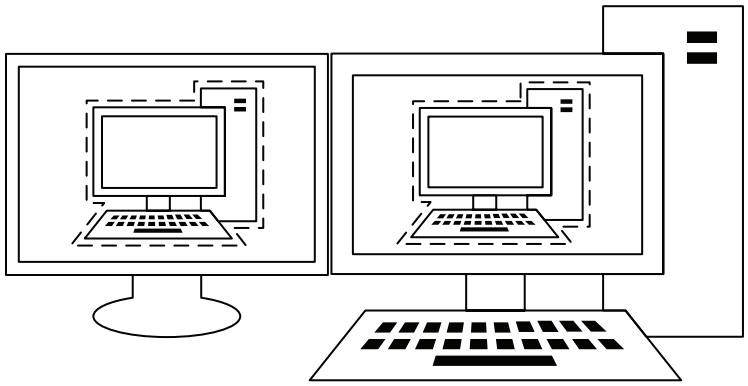
E. Virtualization in HMI and SCADA Systems

Virtualization opportunities exist for both server and client components of SCADA systems. Clients represent a relatively light processing load and may be consolidated onto a Hypervisor host at higher ratios than servers. While servers are more demanding on system resources, servers generally exhibit a constant and predictable workload. This makes it possible to confidently specify host hardware requirements for running critical servers as virtual machines. Only Type 1 Hypervisors are recommended for server virtual machines. The host operating system running a Type 2 Hypervisor limits the Hypervisor's ability to balance the system's resources and introduces a point of failure that can halt all virtual machines.

VDI and Type 2 Hypervisors are both viable options for HMI clients. VDI is the optimal solution as client PCs are replaced with Thin Clients and the virtual machines are consolidated onto a Hypervisor host. Compared to PCs, Thin Clients have a reduced upfront cost, reduced energy requirements, and a longer service life. With the HMI clients moved from a standard desktop PC to a virtual machine on a Hypervisor host, reliability improvements are gained from the clients running on the Hypervisor host's server grade hardware.

Using current desktop PC hardware, Type 2 Hypervisors are adequate to run HMI client virtual machines. While this solution lacks many benefits of a VDI solution, it is easily implemented and can be a stepping stone to a full VDI solution. The typical use case is a failed HMI client; a new PC is procured, but installing the HMI's operating system and software is problematic due to device driver availability. The virtualization solution is to install a Type 2 Hypervisor, create a virtual machine, and install the HMI's operating system and software on it.

Type 2 Hypervisor and desktop PC RAM sizes usually limit the number of virtual machines to one or two. For HMI clients that are single screen by design, it may make sense to consolidate two HMI clients into virtual machines running on the same Type 2 Hypervisor; HMI client A displayed on the left screen and HMI client B displayed on the right.



Type 2 Hypervisor running independent virtual machines, one for each screen.

While it is possible for a single Hypervisor to run every virtual machine of a SCADA system, the server virtual machines should be spread across separate Hypervisors running on separate hardware in order to maintain the system's server redundancy. Likewise, to prevent a single point of failure from leaving operators blind, VDI virtual machines should be spread among host computers.

The ability to create new virtual machines without additional hardware and the ability to make runnable clones of existing virtual machines is a valuable tool for testing software updates and HMI changes in isolation from the production system.

F. Plant Virtualization Example

Figure 1 depicts a basic plant virtualization setup. Redundant servers are divided among physical hosts. Non-redundant servers and HMI clients are distributed among hosts using load balancing. A network storage device, accessible by all hosts, has been added for backup and restore purposes. A standard PC is used for Hypervisor management and doubles as a VDI client. Operator stations consist of Thin Clients which display VDI HMI stations.

Additional HMI Client virtual machines may be configured and not be permanently assigned to an operator station Thin Client. These may be used for floating VDI connections, accessible from office PCs or remotely through a firewall. This is similar to the floating login concept of HMI web clients and may be used in a similar manner, with the added benefit of providing access to a full HMI client.

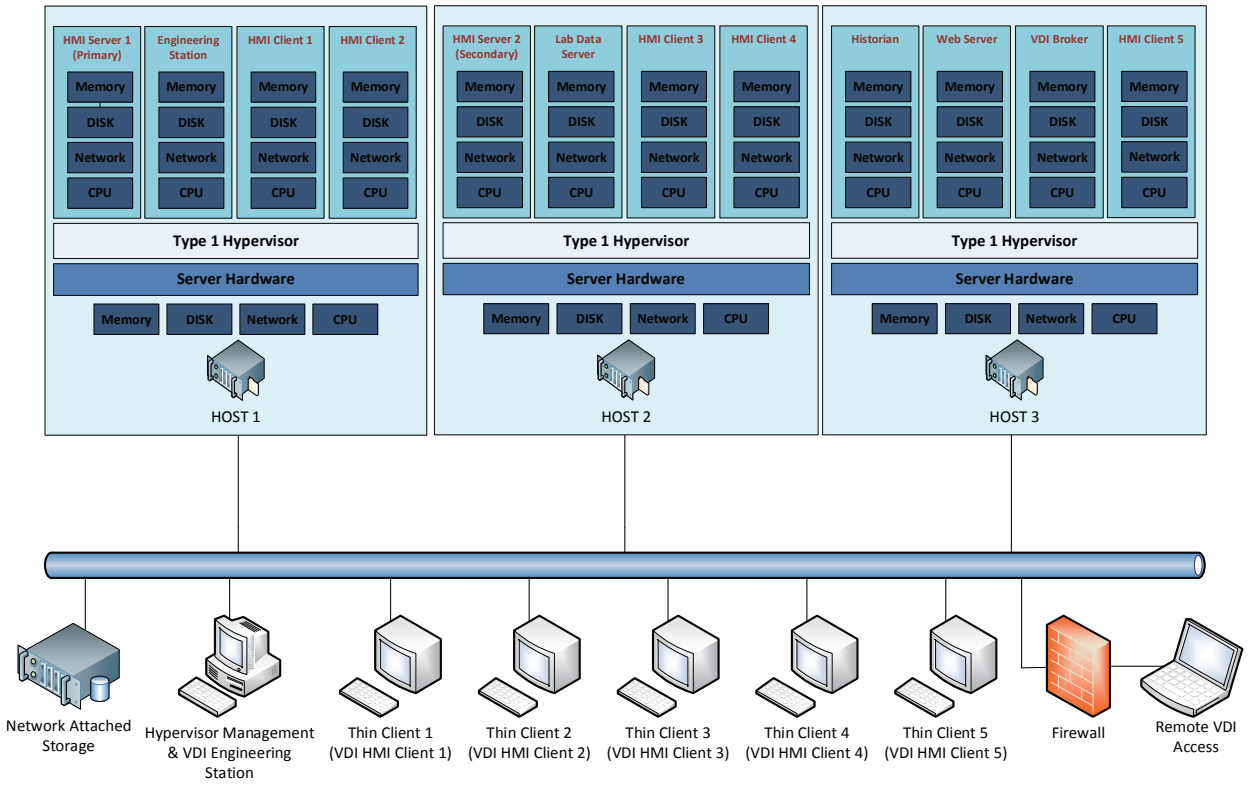


Figure 1. Basic Plant Virtualization Example.

IV. PRACTICAL APPLICATION

A. Introduction

In 2013 a multi-site cement producer engaged in the process of implementing control system virtualization in two of its plants. The two plants, A and B, run control systems from different vendors, providing a broad perspective on the virtualization process.

In both cases virtualization was applied to existing control systems. Plant A's aging control system hardware was due for replacement. With a lack of PC hardware compatible with the control system's operating system, the plant investigated upgrading the control system to the current version. The cost of upgrading the control system software alone far exceeded all other costs and available budgets. The virtualization option was explored as a means of extending the life of the existing control software.

B. Plant A Case Study

Upon analysis, Plant A, as illustrated in Figure 2, was an ideal candidate for virtualization. The control system was already Ethernet based, avoiding the main technical obstacle to virtualization; proprietary server IO cards. The control system is based on x86 computers. Recently, x86 virtualization has made large strides [3]. Multi-core CPUs are common and servers with many cores and large amounts of RAM are cheap compared to a few generations ago. This allows a type 1 hypervisor on a modern server to readily run many virtual machines, especially if the virtual machine is using an older operating system that does not need large amounts of RAM. The result is a reduction in the number of physical servers needed, lowering costs. With the added benefit of ease of complete virtual machine backup and restore, virtualization became the clear choice.

As with all newly introduced technology, the downside to virtualization is training the plant control system specialist. Fortunately, the personnel requiring training on the hypervisor is limited to those in charge of the server room. To operators and office personnel accessing the control system over Ethernet, there is no operational difference between the old system and the virtualized system. Plant A's upgrade was focused on the server room computers. Operator stations consisted of fairly recent PCs. Due to this, the operator HMI stations were not replaced with a VDI solution. Instead the PCs received a RAM upgrade and were repurposed to run the HMI as a Type 2 hypervisor. Each operator PC runs one virtual machine, which contains the HMI. By running the HMI in a virtual machine, future maintenance cost is reduced in the event of a PC failure. A replacement operator PC can be installed with minimal effort, network settings configured, hypervisor installed, and a backup of the VM copied to it. The VM image already contains the HMI environment.

The main pre-commissioning task was performing a clean install of the control system on the virtual machines. This was performed offsite and the virtual machines were loaded on the hypervisor during commissioning. Select servers containing extensive archive data did not have a virtual machine equivalent. Instead, the physical machine was converted to a virtual machine during commissioning with tools provided by the hypervisor manufacturer. During this conversion, the archive server's data storage was increased. The physical machine to virtual machine conversion tool also proved valuable for machines that couldn't be rebuilt cleanly due to software installers that could not be located. Following physical to virtual machine conversion, each operating system had to be reactivated due to the change.

Upon completion, Plant A's control system realized a gain in hardware performance, storage capabilities, IO throughput, and backup capabilities. The use of a Type 2 hypervisor on HMI clients results in the virtualization platform being briefly visible to operators during a computer restart. This is a minor inconvenience that could be eliminated by moving to a VDI solution as the operator PCs reach end of life. Plant A was accomplished for a quarter of the total costs of a full upgrade.

978-1-4799-3294-8/14/$31.00 © 2014 IEEE

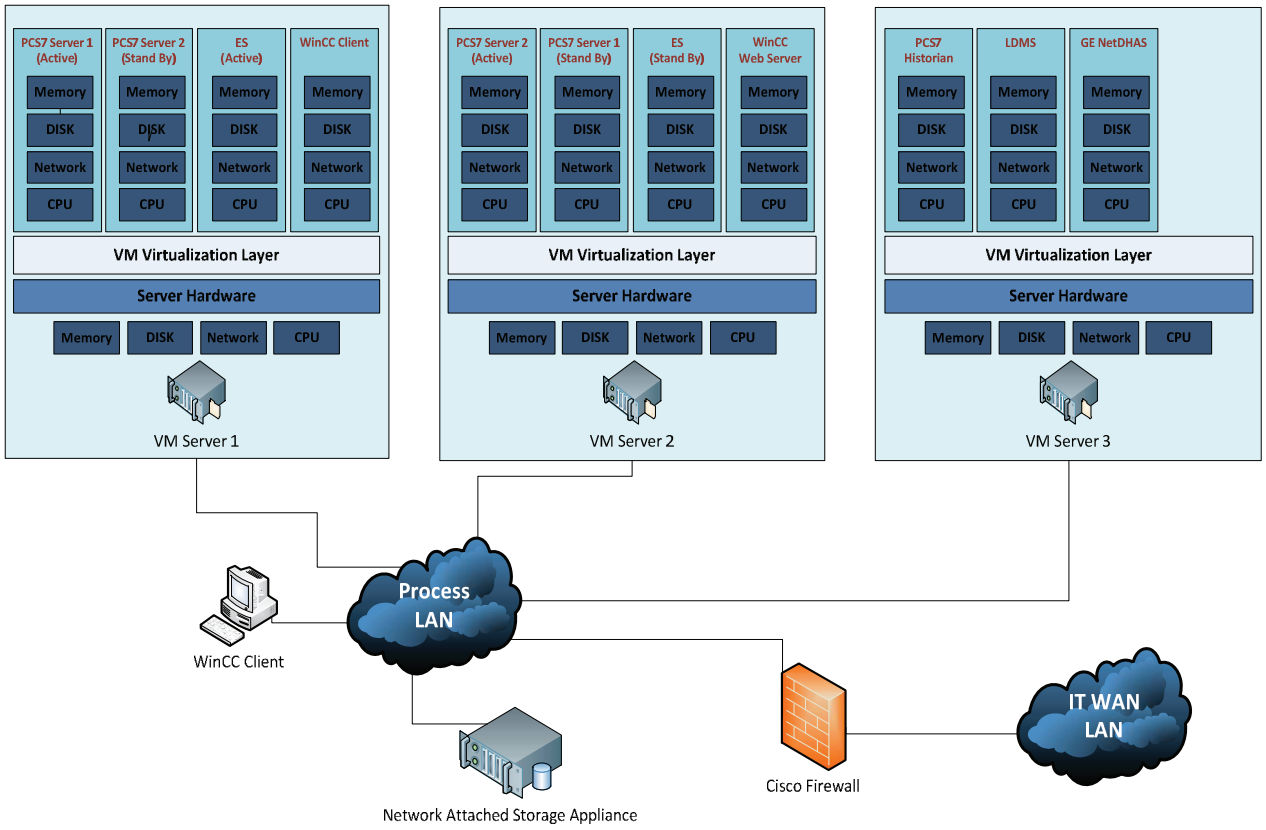


Figure 2. Plant A Virtualization Layout

C. Plant B Case Study

Plant B, as illustrated in Figure 3 was in a similar situation to Plant A, the control system servers were at end of life and starting to experience component failure. Unlike Plant A, Plant B's servers communicated with the fieldbus via proprietary communication boards. The plant had out grown the fieldbus; many plant expansions resulted in the bus being pushed to its device limit. One solution to the fieldbus limitation was upgrading the controllers to use Ethernet based communication. As upgrading to Ethernet also facilitates virtualization, the fieldbus upgrade was approved and implemented together with virtualization in a phased approach.

During Phase 1, the controllers were upgraded and the software was modified to exchange data over Ethernet. With data exchange verified over Ethernet, Phase 2, virtualization, was undertaken. Plant B's existing servers were converted to virtual machines using the hypervisor manufacturer's conversion utility. Additional virtual machines were created so that non-server programs could be relocated from the control system virtual machines. This resulted in cleaner control system virtual machines and a reduced chance of program conflicts.

As with Plant A, Plant B utilized a Type 2 hypervisor for HMI clients. Upon completion, Plant B's control system also realized a gain in hardware performance, storage capabilities, IO throughput, and backup capabilities. Plant B was accomplished 45% less than the total cost to perform the full upgrade.

Plant A and B run entirely different control systems. However, since the virtualization platform is essentially a means of running Ethernet enabled virtual machines, differences in the virtualization platform between the plants is minimal. Both systems strongly resemble the example in Figure 1. This allows for standardized virtualization architecture to be used across plants, even if the plants have differing control systems. With a standardized

architecture, the primary change between plants is a scaling of server count or server specifications to serve the required number of virtual machines. The difference in the control systems are determined by the virtual machines being run.

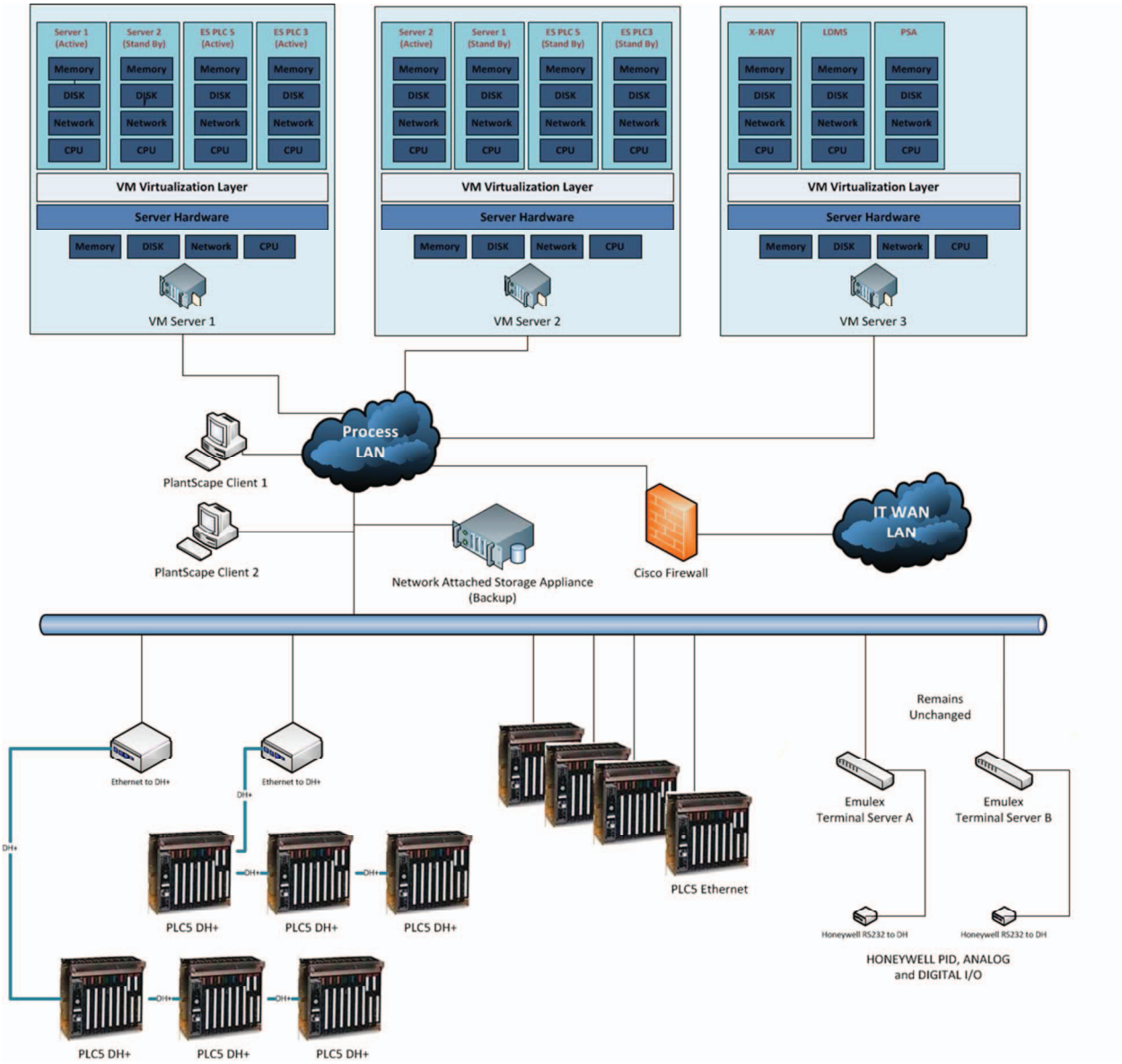


Figure 3. Plant B Virtualization Layout

V. CONCLUSIONS

The maintenance of HMI and SCADA systems based on aging hardware and software platforms can be performed using various methods. One such method, virtualization, can be an effective solution to bridge the gap between expensive full platform replacement and doing nothing at all. Advancement of technology in personal and server computing hardware coupled with the software conflicts with new operating systems and device driver availability for older systems create opportunities for virtualized systems. These opportunities include backup and recovery, operator access to computing resources and prototyping HMI and program changes.

The concept of a virtualized machine was first introduced by analyzing a variety of technologies. Focusing on platform virtualization, infrastructure implementations were discussed pointing out advantages and disadvantages. Further insight into how the virtualization technology can be used was illustrated in two case studies. As discussed in these case studies, several benefits were achieved. These benefits include; reduced physical servers, on-line backup solution, adherence to corporate standards, reduction in administration time, and reduction in hardware / software costs. The result was the development of a system that was easily maintained but more critically postponed costs of full upgrade projects. In addition, Hypervisors were installed resulting in an environment where systems can be rapidly tested prior to deployment and monitored thereafter. Although there are some disadvantages, such as non-Ethernet hardware, the advantages described above clearly make a strong case for virtualization to be considered in any upgrade scenario.

REFERENCES

[1] http://www.wyse.com/sites/default/files/documents/whitepapers/Wyse_Environmental_Benefits_WhitePaper.pdf

[2] Gerald J. Popek and Robert P. Goldberg. 1974. Formal requirements for virtualizable third generation architectures. Commun. ACM 17, 7 (July 1974), 412-421. DOI=10.1145/361011.361073 http://doi.acm.org/10.1145/361011.361073

[3] Principato, M. 2010. Virtualization technology and Process Control System upgrades. Cement Industry Technical Conference, 2010 IEEE-IAS/PCA 52nd, 1 – 12. DOI= 10.1109/CITCON.2010.5469770

978-1-4799-3294-8/14/$31.00 © 2014 IEEE