

**Presented at the
ISPE – ISA Automation Forum - 2010**

DCS Development Systems in a Virtualized Environment

Author Name	Scott Thompson
Title	Systems Consultant
Company	R.E. Mason Co.
Address	PO Box 34645
City/State/Country/Postal Code	Charlotte, NC USA 28206
Telephone	704-375-1567
Fax Number	704-377-5246
E-mail	scott.thompson@remasonco.com

KEY WORDS

Virtualization, Virtualized, DCS, Development.

ABSTRACT

The power and capability of today's servers are well beyond the demand of most DCS systems. The currently shipping rack mount server has more than enough computing power for a development system, not just a configuration server. Through the use of virtualization on development systems a significant reduction of hardware costs can be realized without sacrificing performance. The additional ability to snap-shot the system before making software patches allows quick recovery if a problem develops. Implementing ESXi on three physical hosts has allowed us to run 15 configuration server instances, two batch servers, one Iconics web server, one Emerson Syncade server, one Emerson Control Desktop (NT), one Mynah MiMic Server as well as some VMWare system monitoring and central control application servers. There is also the additional benefit of being able to have a new instance of a server up and running in under an hour which reduces machine setup overhead.

Virtualization of a DCS Development System

Requirements for DCS development systems have increasingly required higher flexibility, faster turnaround, increased robustness and reduced cost. Add the additional requirements for reduced energy usage and replicating production system computers one-for-one becomes impossible.

For our discussion let us use a development system consisting of a configuration server, operator (remote terminal) server, two application servers and capacity for 12 virtual controllers. Using the same computers as a production system this would require seven servers.

As to how this development system meets our requirements for higher flexibility, faster turnaround of testing, and increased robustness. We find that we are not all that flexible. There are seven dedicated computers assigned to specific functions just as with the production system. Turnaround time (amount of time required to “reset” from a test configuration) may vary significantly depending on how much work needs to be done to return to the pre-test configuration. Possibly as much as four hours per machine if an image is to be restored onto each server. Robustness is similar; if we are using the development system to test system patches or new software, the amount time it will take to recover from an inappropriately applied patch or failed install of an application will likely be the time it takes to restore an image of the computer. As to reducing cost, if these servers were to be the currently shipping Dell R710 servers the hardware cost would be ~\$50,400. And as to reducing energy usage, the servers consume ~4.788 kW of electricity and require the dispersal of ~16,000 BTU/hr of heat (~1.36 tons of cooling).

The power and capability of today's servers are well beyond the demand of most DCS systems. The currently shipping rack mount server has more than enough computing power for a development system, not just a single configuration server.

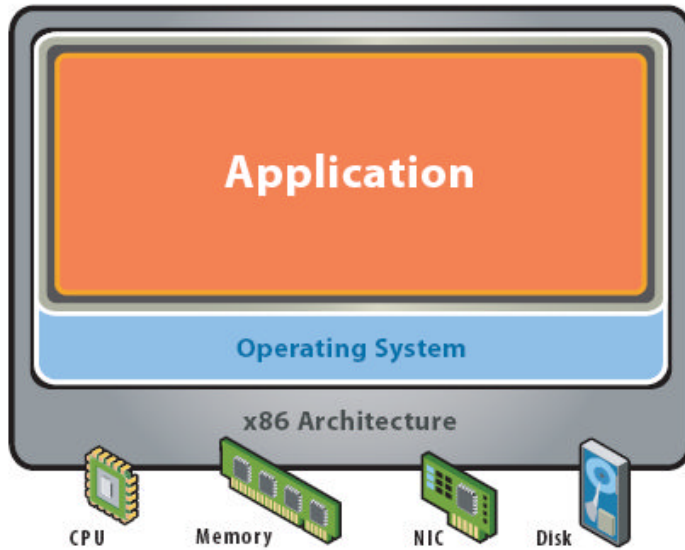
Figure 1 shows the CPU usage of a configuration server with project work active. The server is a Dell PowerEdge 2900. Average CPU usage for the hour was less than 11% with a maximum of 35%. Therefore the server was completely idle well over 75% of the time.



Figure 1

What is virtualization?

Simply stated, virtualization is the process of sharing physical resources between multiple “virtual” computers. **Figure 2** shows a representation of a typical computer without virtualization. Hardware is dedicated to a single operating system running the application(s). Since not all applications can be installed on a single computer, multiple sets of hardware would be required.

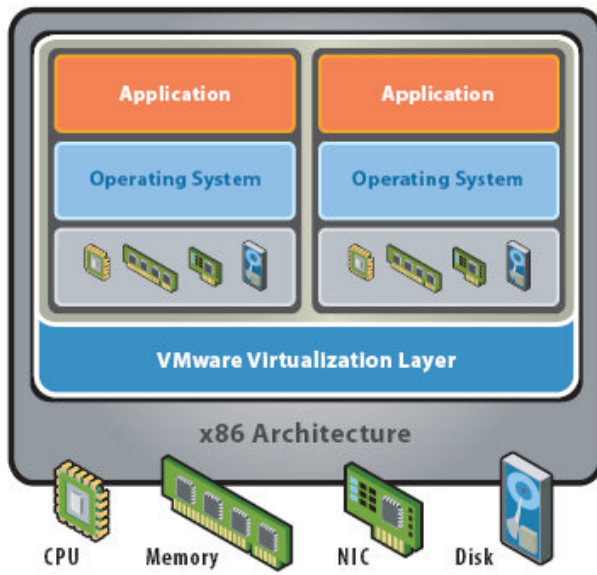


Before Virtualization:

- Single OS image per machine
- Software and hardware tightly coupled
- Running multiple applications on same machine often creates conflict
- Underutilized resources
- Inflexible and costly infrastructure

Figure 2

Figure 3 shows a virtualized configuration where the physical hardware is shared between more than one virtual computer. Incompatible applications can then be installed on independent virtual computers.

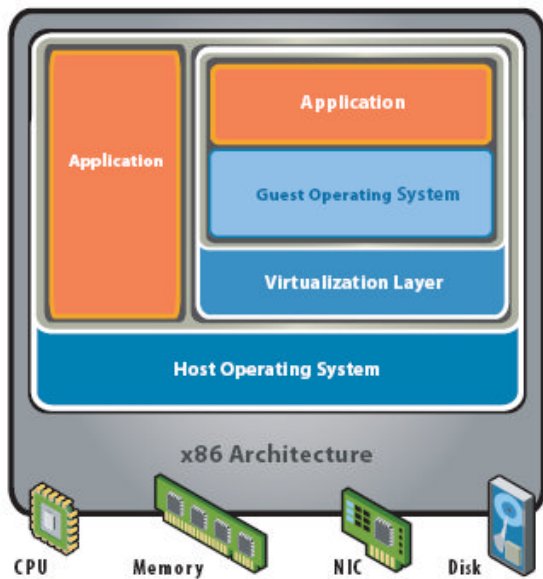


After Virtualization:

- Hardware-independence of operating system and applications
- Virtual machines can be provisioned to any system
- Can manage OS and application as a single unit by encapsulating them into virtual machines

Figure 3

There are two basic varieties of virtualization, Hosted and Bare-Metal. A Hosted virtualization solution relies on a virtualization application running on the host inside the host operating system (**Figure 4**). This solution is limited by the capabilities of the host operating system (Windows Server Standard only supports 4GB RAM) and the host operating system and other applications take their piece of the available memory and CPU. Examples of this solution would be VMware Server and Microsoft Virtual PC.

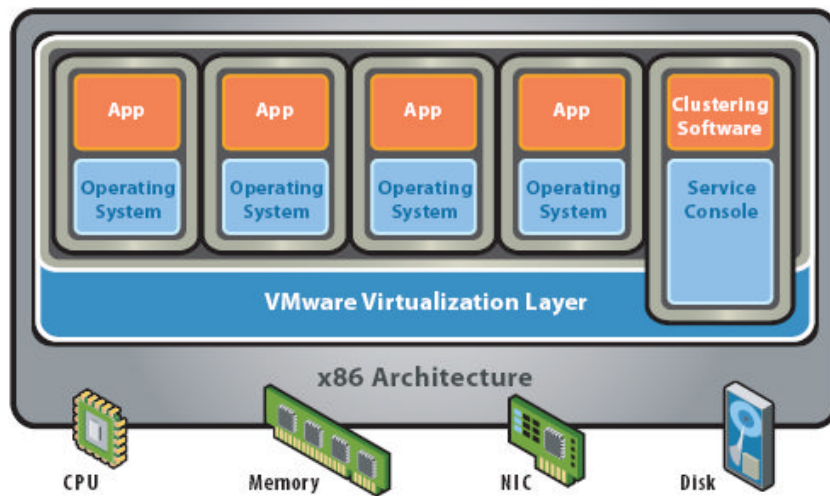


Hosted Architecture

- Installs and runs as an application
- Relies on host OS for device support and physical resource management

Figure 4

A Bare-Metal or Hypervisor architecture (**Figure 5**) replaces the host operating system with a thin kernel minimizing the resources used to manage hardware and the virtual computers. Examples of this solution would be VMware ESX(i), Microsoft Hyper-V or Citrix XenServer. The thin kernel supports the full capabilities of the hardware without the restrictions of a base OS.



Bare-Metal (Hypervisor) Architecture

- Lean virtualization-centric kernel
- Service Console for agents and helper applications

Figure 5

Why Virtualize?

As we saw in **Figure 1**, computers spend most of their CPU time waiting for slow humans to decide what to do. Even when typing rapidly the computer is still spending the majority of its processor time doing nothing (~68% doing nothing as I type this).

To put it into perspective, **Figure 6** and **Figure 7** show an ESXi hosted virtual configuration server with five users doing simultaneous code and graphics development. For the day shown in **Figure 6**, average CPU usage is 18.9% with a maximum of 65.6%. For the hour (shown in **Figure 7**) the average was 30.22% with a maximum of 73.73%. This is a percentage of the resources made available to the virtual machine which is not the full resources of the host computer. The actual resources assigned to this virtual computer were approximately half of the CPU of a Dell PowerEdge 2850 server.

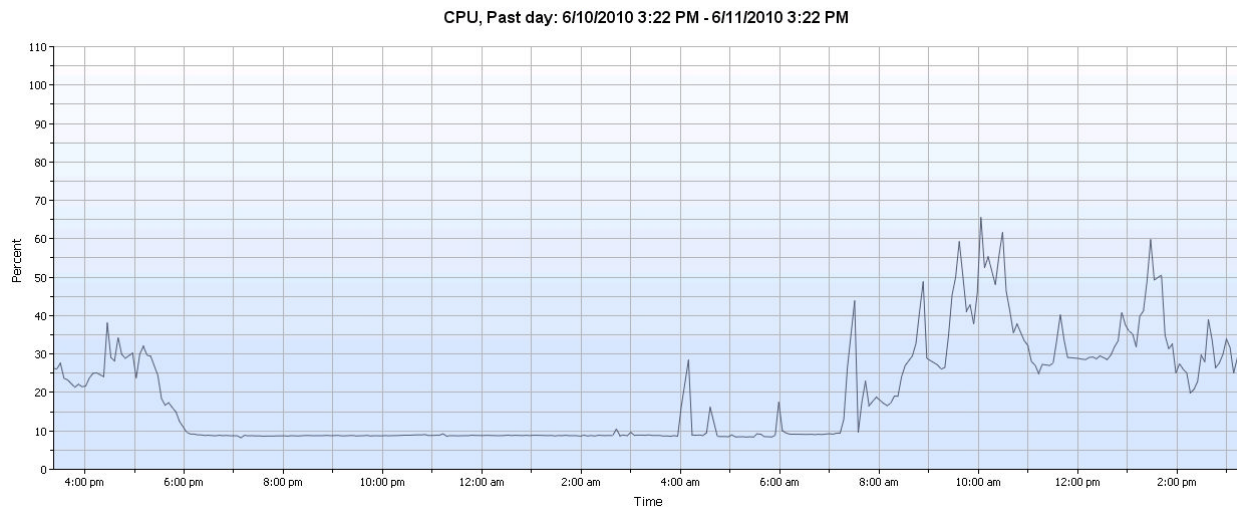


Figure 6

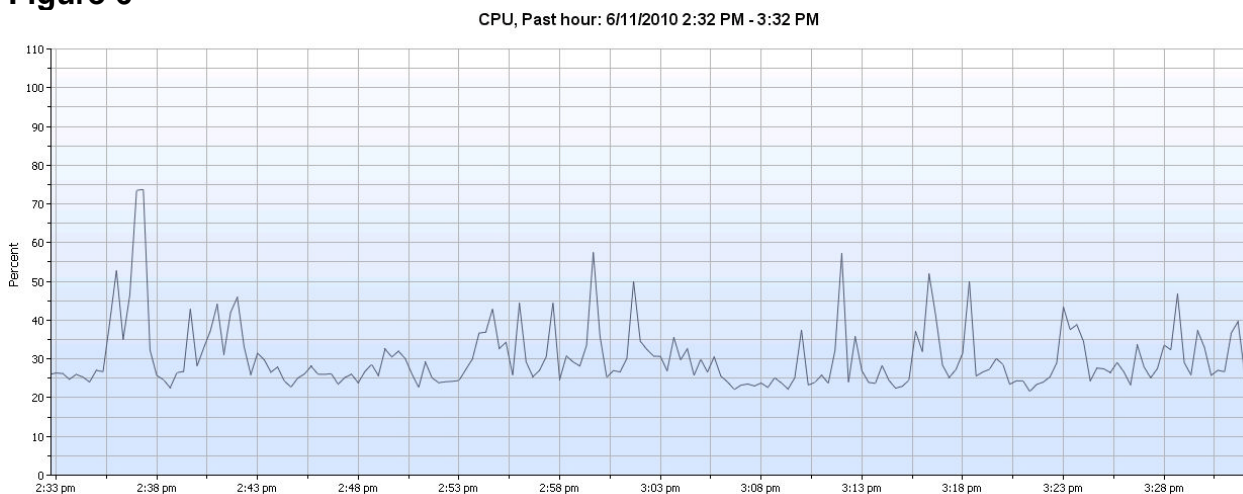


Figure 7

The standard CPU available for the R710 server (dual CPU configuration) is a quad-core processor with an effective 19.2GHz of CPU power available. The PowerEdge 2850 that was used for the graphs in **Figure 6** and **Figure 7** had less than 6GHz of CPU power available.

Let's return again to our requirements. With a virtualized solution our flexibility is greatly enhanced. The ability to "clone" or even create a new virtual computer in minutes is vital in being able to respond quickly to new requirements. Our turnaround time is greatly reduced. Before beginning a test a "snapshot" can be taken of the virtual computers (even in a running state) allowing a quick return to the snapshot state after a test scenario is completed. Snapshots can also be scheduled to ensure that one is always available if a user forgets to perform the snapshot manually. The snapshot capability also enhances our robustness when testing patches and new software. If incompatibilities become apparent the system can be restored to the last snapshot before the changes were made.

The cost for our example system has also been reduced significantly. There is still the hardware cost but this time for a single server, Operating System license cost for six servers

(assuming one license was purchased with the hardware) and additional RAM (16 GB for a total of 20 GB). The hardware and licensing cost for our system has now become ~\$13,000. Energy costs have also been reduced. Our single server is now consuming ~700 W of power and only 2334 BTU/hr of heat need be dissipated (0.19 tons of cooling).

Table 1 below gives a side-by-side comparison between the virtualized and non-virtualized solutions.

	Without Virtualization	With Virtualization	Savings
Cost	\$50,400	\$13,000	\$37,400 one-time
Power	4,788 W	700 W	\$2,100 per year (at \$0.06 per kWh)
Consumption			\$697 per year (assumes very efficient A/C system, 0.33 K- factor)
Heat	16,000 BTU/hr	2,334 BTU/hr	
Dissipation	(1.36 tons)	(.19 tons)	

Table 1

In some cases, presenting savings will not always present a convincing argument. Dollar expenditures for equipment may come from a tightly restricted pool of money. However, the green initiatives typically are budgeted differently than routine equipment expenditures. Presenting replacing an existing development system with a virtualized development system as a reduction of carbon footprint may get approval where a simple dollar savings may not be approved.

Not all costs associated with the system have been detailed in this comparison, e.g., the licensing of the DCS software, remote terminal licensing from Microsoft. As these costs would be the same for both solutions they are irrelevant for this comparison. For the requirements of the system presented here there is no need to use the licensed VMware ESX software so the cost for the VMware ESXi software is \$0.

Stability

REM Services has been utilizing virtualization for our development systems for nearly two years. The stability of the virtualized computers has been consistent with conventional physical servers used concurrently over that period.

Conclusion

Virtualization of a development system can reduce the initial cost of a development system as well as reduce the reoccurring costs of the system. There is also the possibility of re-using existing hardware to further reduce the initial cost as we have done at REM Services. The additional improvement in flexibility and reduction of testing reset time further enhance the savings of virtualizing.