Paper 218-31

SASÒ AND VMWAREÒ TO CREATE AN ENVIRONMENT FOR COMPUTER SYSTEMS VALIDATION IN A PHARMACEUTICAL COMPANY

Wayne Woo, Chiron Corporation, Emeryville, CA

ABSTRACT

21 CFR Part 11 states that computer systems used in the clinical trials process must adhere to certain criteria, one of which is validation. The computer systems validation (CSV) paradigm exists to provide guidance and structure to this process. Within the SAS® programming environment, there is a lot of debate on how to approach validation, both of the SAS programming environment itself, and the applications that get created. There often are contentious viewpoints. The CSV model calls for separate areas for development, testing, and production to ensure that there is an orderly, documented path to the development of SAS programs and enhancements to the SAS infrastructure itself. This paper examines how virtualization of systems can play a role in adhering to CSV principles for SAS systems. An example is given of how our company is using VMWARE® technology to host a development and testing environment.

INTRODUCTION

Computer systems validation (CSV) is a hot button issue within SAS programming departments. SAS programmers and statisticians are interested in turning out reports quickly to meet the information needs of internal consumers and the FDA. However, at the same time, everyone recognizes the need to perform adequate checks on the underlying programs and system to ensure that the output is correct. The amount and nature of this quality assurance (QA) and the associated documentation is what spurs legions of debate among biostatistical and QA personnel.

Traditionally, the SAS system itself, i.e. the programming environment (operating system, programmer tools, etc.), was not subject to much scrutiny. System administrators and programmers themselves would install SAS and run a few, if any, programs to see that everything was working. The programs developed within SAS might come under slightly more scrutiny through code review and independent programming. In the world of CSV and far-reaching QA departments, this has changed.

The system itself now gets examined closely. QA personnel have asked software vendors for documentation on how software was developed and validated. SAS has written white papers about this topic and even created installation qualification (IQ) and operational qualification (OQ) tools to support this activity. QA personnel have asked for the establishment of separate hardware to host SAS environments for development, testing, and production. For example, a service pack might be installed first on development, then moved to the testing environment for formal validation, and finally applied to production. Supposedly this ensures good documentation of what happened and helps maintain a stable and reliable system.

The downsides include personnel time to support this process, and also additional hardware and software costs to set up separate environments. Hardware virtualization can help reduce costs associated with extra hardware and personnel time required for setup and maintenance. This paper describes the possibilities that virtualization (using products from VMWARE) can bring to the SAS computing environment. We will not dwell on a lot of technical details, but instead focus on concepts.

DEFINITION OF VIRTUALIZATION

In our context here, virtualization is the “software-izing” of hardware. In other words, software has been written to encapsulate the characteristics of hardware components. When viewed by an operating system or application, the software imitation of hardware is indistinguishable from real physical hardware. A computer with virtual hardware is called a “virtual machine”.

There are two major virtualization software vendors in the marketplace today, VMWARE and Microsoft. VMWARE is the acknowledged leader in the field. The company sells an assortment of products.

VMWARE PRODUCTS

There are three core products with which virtual machines can be built and run. In addition, VMWARE makes additional software and tools that make for an easier transition to, management of, and provisioning of virtual machines.

The three core products are called "Workstation", "GSX Server", and "ESX Server". Complementing products include the "P2V Assistant", "ACE", and the "VMWARE Player". Each is briefly described below.

ESX Server is enterprise-level software that can be employed in corporate environments to create and host virtual machines to be used for testing, development, and production. It requires beefy hardware and is expensive. It is licensed based on the number of CPUs in the physical host server, whether it be 2,4,8, or 16. It actually is an operating system itself and does not require an underlying OS.

GSX Server is a scaled-down version of ESX Server and requires a host operating system onto which it is installed. The target adopters are departments within organizations that might wish to virtualize a small set of computers. GSX is less expensive and has slightly fewer physical hardware requirements. Because an underlying operating system is required, an extra layer exists between the virtual machine and the physical hardware.

Workstation is a desktop version of VMWARE for people desiring to run multiple isolated virtual machines on a single machine. Perhaps this might be to simulate a network of computers, or to have a separate test machine, or even to access the Internet without worrying that viruses will corrupt the host computer. It is very inexpensive, doesn't require much setup time, and is ideal for the developer or hobbyist who in the past may have bought multiple computers or had to enable dual booting. A host operating system is required.

The P2V Assistant allows the "snapshot" imaging of a physical machine to transform it into a virtual machine. P2V means "physical-to-virtual" and provides the pathway for the conversion of space-occupying physical machines into virtual machines consisting of a few files that encapsulate the entire hardware and software setup.

The ACE product allows IT personnel to provide to workers an "(a)ssured (c)omputing (e)nvironment" that runs in an isolated fashion, providing for secured computing and other benefits. The virtual machine can be locked down from modifications and limited to accessing pre-specified resources. We'll talk more about how this product has potential for creating CSV compliant SAS PCs.

VMWARE Player is a new product that is free. By installing it, users are able to run any virtual machine created somewhere else. This includes machines created in the products above. It also includes the community of virtual machines now offered by a variety of sources that want to make it easier to test and evaluate their software without the upfront costs to set up a machine and without the risk of adversely affecting the "everyday" machine.

THE COMPUTER SYSTEMS VALIDATION (CSV) PARADIGM IN A NUTSHELL

CSV is defined on the web as "Confirmation by examination and provision of objective evidence that computer system specifications conform to user needs and intended uses, and that all requirements can be consistently fulfilled."

We will not go into this topic too deeply, but rather describe the implementations of this model in practice that enables the collection of evidence alluded to by the definition above. Achieving a validated state for a computer system usually includes the setup of a development, testing, and production environment so that software can be tested and implemented in a controlled fashion. This approach is intended to reduce the risk of failure and adverse effects on dependent software and systems. It also calls for adequate documentation and provides a sequence of steps to recover in the event of a disaster. CSV is a paradigm spelled out in government-issued guidances to regulated industries such as the pharmaceutical, health care, and biotechnology areas where there is potentially a huge impact on human well-being from mistakes and problems in computing and analyzing data.

TRADITIONAL SAS SYSTEMS SETUP

The SAS programming environment consists of two areas. On one hand, there is the system itself, including SAS software, supporting software, the operating system, and the hardware hosting these software. And on the other hand, there are the programs and applications written with SAS. An analogy would be a C++ environment, where you have the compiler, editor, programming language, etc. and then the software that gets written with it.

In this paper, we focus on the system part of the SAS environment. In particular, we highlight server computing where users log into a centralized computer and access SAS there.

Setting up three server environments for CSV usually involves buying three pieces of physical hardware, one each to host development, testing, and production. Identical software would be installed on each machine using the same documented process. Changes made to the systems would start in the development environment and then propagate upwards to testing and finally production.

The downside to this approach is the steep costs to purchase hardware, set it up, maintain it, find space for it, and even power consumption. Furthermore, the testing and development servers are not utilized much except for testing new system components, so there is the issue of wasted compute cycles and limited return on investment.

As a final word on system setup, an approach that smaller companies sometimes take is whereby users install SAS onto their own PCs. Usually companies doing this have not had a controlled process for installing patches and performing upgrades. Users tend to customize their individual PCs differently, installing additional applications, downloading different tools from the Internet, patching SAS at different times, etc. Later, we describe where virtual machines might fit into this scenario.

THE ROLE OF VMWARE

There are several possibilities whereby VMWARE products might address the downsides of traditional CSV system setup and even help a company without such a setup to achieve one.

ESX and GSX offer server consolidation benefits right away. With one physical server, multiple virtual computers can be created. For example, one server might house all three environments – production, testing, and development. Alternatively, there could be one server acting as production, while a VMWARE server houses the testing and development environments. In each scenario where there is at least one less physical machine, money for purchase is saved, power consumption is lower, space requirements are reduced, and setup time of each subsequent virtual machine is minimized. In fact, the P2V Assistant can create an exact replica of the production machine and this can be used to deploy the other two environments as virtual machines.

The ability to quickly create a clone of a virtual machine makes it easy. With a virtual machine approach, conceivably servers can be cloned and farms of SAS servers made available as a load-balancing solution.

The ease of deployment can even be extended to the desktop using SAS PC licenses. VMWARE allows the creation of single template PC with SAS on it. Cloning it can create a whole “bank” of standardized machines residing on a central server, which can then be assigned one each to each user. As patches are tested, these machines can be re-created and cloned and re-distributed to the users.

If it is desired to lock down workstations or have people working at home or on the road, VMWARE ACE is a product that is appropriate when installing SAS onto uncontrolled computers. By creating a secure, locked desktop and deploying to users to run on top of their existing systems, the SAS environment can be made the same for everyone. The virtual machines configured by ACE can be allowed to access the resources that the host can. At the same time, the SAS desktop is locked down. The user can still modify his underlying machine without worry that the SAS environment is contaminated.

One additional application of virtual machines is for the occasional need to evaluate new software that SAS or other vendors release. A recent example is the SAS Business Intelligence Suite Test Drive. With virtual machines, a test machine could be provided within minutes. Users can even install VMWARE PLAYER on their desktop and run a virtual machine provided by the IT department. All the while, the virtual machine is isolated from the everyday machine, making it unnecessary to uninstall the evaluation software at the end of the trial period. This definitely avoids the clutter of expired software on a PC.

A REAL-WORLD PERSPECTIVE

At Chiron, our validated SAS Windows environment consists of a physical production server and an ESX server housing the testing and development machines, and a handful of licenses for unvalidated PCs. We decided on a physical server

for the production environment because of the need to have four CPUs, and because the ESX product at the time did not support assigning more than two CPUs on each virtual machine. However, the newest versions of ESX do now support four CPUs. Thus, it would have been possible to put all three environments onto one physical machine, thereby saving hardware costs. Nevertheless, we saved the cost of a third machine, the time needed to set up that machine, and reduced space and power requirements.

Our process includes the creation of a P2V snapshot of the production server and placing it on the ESX machine as a development environment for trying out new software, upgrades, patches, tweaks, etc. Since the virtual machine is really a set of files, they are copied and stored as an image. We can turn it on to get an instant testing server for use in formal validation. To avoid machine name and IP address clashes, we can turn off the development machine. When validation testing completes, the production machine is updated and a new snapshot of the updated environment is made and it then becomes the new development environment.

The ability to stage a virtual machine on any VMWARE server also provides a convenient method for disaster recovery by reducing the time for provisioning of a workable machine while the real production machine gets repaired or replaced. In our plans for disaster recovery, the virtual image of the production machine can be fired up on an ESX server and provides the same computing infrastructure for temporary usage by programmers and statisticians. This is made possible because the data store housing all the SAS data, programs, and applications is on disks external to the machine. The data drives can be connected to the virtual machine with a few modifications.

We have adopted this approach for the SAS infrastructure itself. For the programs and applications developed within SAS, we implement the “dev-test-prod” environments via directory structures on the production server. This negates the need to move files between servers and ensures the use of a common environment to run SAS programs. This has the benefit of isolating programming issues to the program itself rather than worry about whether the underlying software contributes to the problem. This topic is beyond the scope of this paper, but there are many debates in this area of how to validate SAS programs.

SOFTWARE LICENSING

Adopting virtualization technologies undoubtedly raises questions on software licensing. It is easy to forget this legal aspect when it is so easy to create, duplicate, and provision a virtual machine. Because virtualization has become popular, software licensing terms are being discussed and revised at many software vendors to accommodate this new technology rather than hinder it.

Like many vendors, SAS licensing for servers is based on the physical hardware characteristics of the host server and operating system. It is not based on the virtual machine characteristics. However, this raises some issues since the virtual machine is clearly an independent machine but with a fraction of the computing power of the underlying physical machine. For example, a company may wish to implement a SAS virtual machine on a large hardware box. The company would need to license SAS for the large machine when SAS is really only using a small piece of it. As I understand, SAS is revisiting the licensing structure in this type of setup since the current policies might have a negative impact on virtual machine adoption.

However, the direction is positive. As an example, Microsoft recently announced that buying one copy of Windows Server entitles that copy to be installed onto a number of virtual machines that are hosted on the physical machine for which the software was bought. There is a summary of these licensing terms at the web address: http://download.microsoft.com/download/6/8/9/68964284-864d-4a6d-aed9-f2c1f8f23e14/virtualization_brief.doc

Regardless of licensing terms, it is important to have the proper number of software licenses. The rule of thumb is to treat each virtual machine as if it were a physical machine. The legal maze of licensing is becoming increasingly complicated with technologies such as virtualization and multi-core CPUs. The message here is that there is some legal homework to do when setting up a virtual infrastructure.

CONCLUSION

The availability of virtualization software greatly reduces the cost of setting up separate development, QA, and production machines required to have a validated infrastructure. Even if a company decides to install SAS on individual desktops, the use of VMWARE can facilitate the provisioning of standardized, secure, and validated virtual machines that run

isolated from a user's unvalidated environment. Virtual machines introduce some new issues into the corporate computing landscape, but the long-term benefits justify the time spent tackling each issue and planning for the future.

REFERENCES AND RESOURCES

http://www.vmware.com
http://www.microsoft.com/windowsserversystem/virtualserver/default.mspx
VMWARE Technology Network (http://www.vmware.com/vmtn)
VMworld 2006 conference (http://www.vmware.com/vmworld)
Dictionary of terms: www.data-core.com/glossary-of-terms.htm

ACKNOWLEDGMENTS

I wish to thank my managers at Chiron for the opportunity to write this paper.

BIOGRAPHY AND CONTACT

Wayne Woo is a senior project manager performing a mix of project leading, clinical trials programming and a variety of other technical activities. He is a SAS Certified Professional V6 and has presented at WUSS and SUGI. When there's free time, he experiments with lots of new and old technologies to find better ways to accomplish a task.

Wayne Woo
Chiron Corporation
4560 Horton St.
M/S U-201
Emeryville, CA 94608
wayne_woo@chiron.com