

Virtualization, Cloud Computing, and Business Continuity ... Oh My!

Written by RON LaPEDIS, MBCP, MBCI, CISSP-ISSAP, ISSMP

Virtualization is all the rage this year, whether it is for load balancing and performance, high availability, or to lower development and operational costs. When IT employees hear the word, they think of server virtualization. As we shall see, there are many kinds of virtualization with many “flavors” within them.

When combined, the many kinds of virtualization can help with your business continuity program no matter what risks you face. I will show you how geographically dispersed virtualization (GDV) can not only move your data and processing power closer to your customers, but can also provide the foundation for continuous availability of your applications – ensuring no downtime in the event of an infrastructure failure.

First, I’m going to define and describe the various kinds and flavors of virtualization and show you how cloud computing with load balancing can increase everyday performance while lowering costs. I’ll follow that by showing you how these technologies also can be used to provide continuous availability of your applications.

Virtualization Defined

The dictionary says that virtualization is the conversion of something to a computer-generated simulation of reality. Think of the online, 3-D virtual world “Second Life,” for example. When you virtualize an operating system (OS), an application, or a PC desktop you are creating a simulation of the layer below which the software is running. You fool an operating system into thinking that it is running on the hardware for which it was designed. You fool an application into thinking that it is running on the OS for which it was designed.

When you virtualize an OS, you are unpairing the software from the hardware. Now you can run multiple instantiations of the same or different operating system on the hardware while each one thinks that it is the only resident. For example, the same hardware can run Windows and Linux. A virtual desktop is a special case of a virtualized OS in that a single instantiation is customized to support a specific user and can be delivered to wherever that user might be and on whatever device they are using to access the network.

OS Virtualization

There are two basic ways to virtualize an operating system – native or hosted. A native or bare-metal hypervisor is software that runs directly on a given hardware platform (as an operating system control program). A guest operating system thus runs at the second level above the hardware. A hosted hypervisor is software that runs within an existing operating system environment. The guest operating system thus runs at the third level above the hardware. In either case, the “container” which holds the guest OS is called a virtual machine (VM).

A hosted hypervisor frequently is used where the user prefers to use a specific OS but infrequently needs to use some software that doesn’t run on that OS. For example a Macintosh user might need to access Web sites that only support the use of Internet Explorer. Through a virtualization software solution, the user can run a copy of Windows as a guest on top of the host Mac OS. The guest OS on a hosted hypervisor will only get resources when they are given to it by the host OS. If the host OS fails for any reason, the guest OS fails with it.

If you are as old as I am, you know that mainframes have been virtualized for two decades through the use of LPARs. The LPAR is a subset of a computer’s hardware resources, virtualized as a separate computer and each housing a separate operating system. This technology is now coming to x86 based systems through the use of a native hypervisor. Since the multiple operating systems run on top of the hypervisor, they cannot affect each other as the hypervisor also acts as the referee to the hardware.

In many cases, companies limit their servers to only one application environment for simplicity and ease of maintenance, even if the hardware is severely underutilized. Rather than building the application environment directly on the server, it can be built on a VM, and multiple VMs can be run on a single server since each VM isolates the OS and applications from the OS and applications in the other VMs.

When new applications need to be deployed, additional VMs can be created instead of having to add additional servers. When it’s time for hardware maintenance, the VMs can be moved to another server, and when an application needs additional resources, it can be moved to a more powerful server.

In addition to virtualizing servers, you can virtualize desktops and laptops. The benefits are similar in that you can run multiple OSes, such as Mac OS, Linux, Windows XP, and Vista at the same time, or even mix and match 32- and 64-bit versions of the same OS.

Whether desktop or server, migrating to a 64-bit CPU can drive additional hardware consolidation. While many legacy applications won’t run on a 64-bit OS, you can run multiple virtualized copies of a 32-bit OS on a 64-bit CPU with the hypervisor “hiding” the extra 32 bits from your applications. Running multiple VMs on each server means you can support more users per box, and it’s extremely easy to scale up your environment or fail over to another datacenter.

Application Virtualization

A virtualized application is not installed in the traditional sense, although it still may be executed as if it is. The application is fooled at runtime into believing that it is directly interfacing with the original operating system and all the resources managed by it, when in reality it is not. Application virtualization can improve portability, manageability, and compatibility of an application by unpairing it from the underlying operating system on which it is executed.

There are multiple ways of virtualizing applications. With server side application virtualization applications run in the data center and are displayed on the user’s PC through a browser or specialized client. The application does not need to be compatible with the operating system running on the PC.

Figure 1 – OS Virtualization



Figure 2 – Server Side Application Virtualization

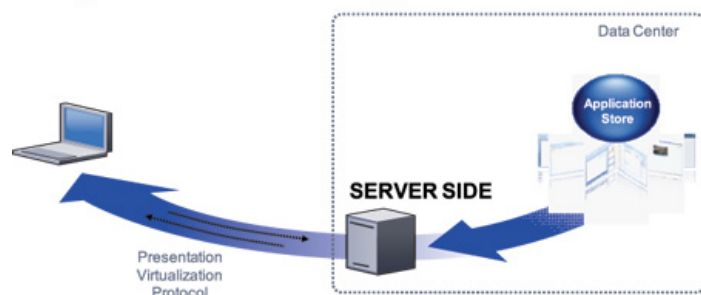
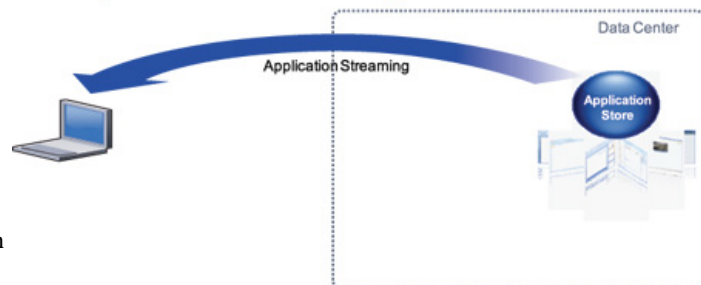


Figure 3 – Client Side Application Virtualization



With streamed or client side virtualization the application resides in the data center but is delivered to the user's computer to be run locally. Because it is running locally, the resources that normally would be installed into the OS, such as dynamic linked libraries (DLL), code frameworks, control panels, and registry entries are installed into an application container and the entire container is streamed.

The container can be sent to the PC every time that it is needed, or it can be stored on the user's PC for a specific period of time before it expires and needs to be streamed again. The latter method allows for use of the application even when not connected to the network, for example, while on an airplane.

As with the first method, application updates are easy since there is only one copy of each application, and it resides in the data center. This means that only one copy gets updated, rather than needing to push updates out to hundreds or thousands of PCs on your corporate network.

Another way to virtualize an application is similar to the previous approach in that the application is still packaged into its own container, but it permanently resides on the user's PC instead of being streamed. When the application needs to be updated, a new container is downloaded to the PC.

An immediate benefit to virtualizing an application in any of the ways shown above is the elimination of dependency hell (sometimes called "DLL hell" on Windows systems), which happens when incompatible applications are installed on the same OS. A common and troublesome problem occurs when a newly installed program overwrites a working system file with an incompatible version and breaks the existing applications.

Desktop Virtualization

Desktop virtualization or virtual desktop infrastructure (VDI) provides a personalized PC desktop experience to the end user while allowing the IT department to centrally run and manage the desktops. Desktop virtualization is an extension of the thin client model and provides a "desktop as a service."

The user does not know and does not care where their desktop is running. They access it through a "window," which may be a specialized client or Web browser. In fact, depending on the security policy they may be able to access their desktop from anywhere using any device, even one that is not compatible with the desktop OS being served.

Since virtualized desktops are centralized, it is easy to keep them patched, prevent users from installing software, making configuration changes they shouldn't, and load balancing the users or upgrading their OS as needed without needing to upgrade the user's endpoint hardware.

When you virtualize a desktop and add virtualized applications on top of it, the user is provided with a brand new PC experience every time that they connect to their desktop. The well-known problem of PCs slowing down as they are used becomes a thing of the past.

Cloud Computing

Utility or cloud computing started out as a general concept that incorporated software as a service where the common theme was reliance on the Internet for satisfying the computing needs of the users. For example, Google Apps provides common business applications online that are accessed from a Web browser, while the software and data are stored on the servers.

This makes sense if you look at how you use electricity. In developed countries each building doesn't generate its own electricity but rather is supplied by the electric grid. In essence, the grid is a cloud of generators, transformers, and transmission lines. When you plug in something, it draws upon this grid for power, and more power is generated somewhere else to make sure that everyone has enough. Why shouldn't computer power be supplied in a similar manner? If you need CPU cycles and someone else doesn't, why should both of you be wasting power generating your own? Wouldn't it be great to plug into a computing grid and get all of the computing cycles and storage that you need when you need it and only pay for what you are using?

But many companies either don't want to put, or are prohibited by regulation from putting confidential information in the cloud. For them we can offer the latest definition of cloud computing which is a superset of the virtualization concepts mentioned above. Not only is the software unpaired from the hardware, it is so unpaired that it can run on any compatible hardware in the company's network. That is, you can now turn your entire computer room into its own cloud.

The OS and desktop virtual environments are built in a manner that allows them to run on any compatible hardware that the company owns. Now they can be moved from computer to computer as the load dictates and as CPU cycles become available. On a major holiday, a bank can run more of the virtual environments that support their ATMs by taking down environments for nonessential functions, such as virtual desktops for workers on holiday. Storage area networks are invaluable when creating a cloud-computing environment as they allow you to virtualize your storage as well, moving an environment's data along with the environment itself.

The main downside to internal cloud computing is managing the environment. While you no longer need to know where all of the pieces are, you still need to ensure that all of them have the proper resources, are meeting their users' response time requirements, and most importantly are being backed up in case disaster strikes.

If your environment is a cloud, how do you connect to the application that you need? This is where the load balancers come in. The purpose of a load balancer is to direct traffic to an application no matter where that application lies in the cloud. This means that the load balancer needs to know where the application is at any moment. Whether steady-state or in a peak situation, there could be dozens of instances of the application, and it is the load balancer's job to figure out which instance is least busy and route the request to it. But the load balancers cannot get this information on their own and this is another reason that you need to be running cloud management software.

Lowering Costs Through Technology Virtualization

According to a Forrester-validated TCO/ROI tool, desktop virtualization can lower your cost of supporting desktops by up to 40 percent by breaking the image, secure, deploy, maintain, backup, and retire cycle required by traditional PC usage. It also lowers your support desk costs because applications just work – without your users needing to babysit them.

Server virtualization can lower everyday costs by squeezing out all of the performance that your existing systems have to offer. Just as you only pay for the electricity that you actually use, your company can lower the cost of IT by only paying for the computing cycles that are actually needed.

Figure 4 – Application Isolation

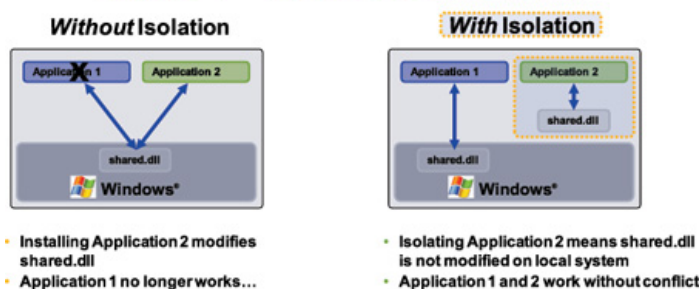
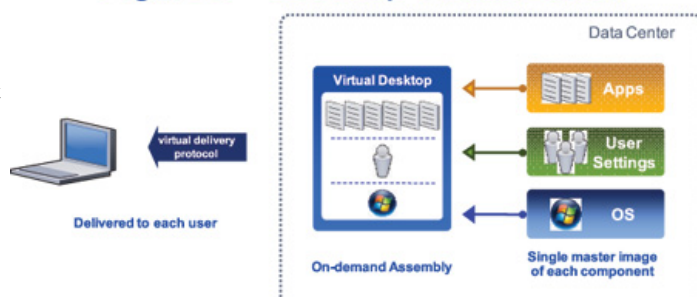


Figure 5 – Desktop Virtualization



Before you can start this process, you need a solid understanding of the relative importance of your business processes so that you know which of them can be dialed back or eliminated when you hit your peak capacity. This is not unlike adjusting your air conditioner or turning off your big screen television or electric dryer when the local electric utility tells their customers that there will be a blackout if they don't lower the load.

System monitoring tools also come into play. While you can guess that your air conditioner, big screen television, or electric dryer pull much more power than your electric clocks and room lights, it is difficult to guess which processing environments use the most cycles without some kind of monitor. You need to build an inventory of your total CPU cycles and the CPU cycles required by each processing environment across a wide range of loads.

Think of your computing capacity as a bunch of boxes and your processing requirements like children's building blocks. Both can be different sizes and you can mix and match so that you fill each box with the maximum number of blocks. But unlike real building blocks, your blocks can get bigger and smaller depending on the time of day, day of month, month of year, current events, and so on.

Historical capacity models allow you to increase the number of environments required for a specific business processes or move them around before the load hits. If an unexpected peak occurs, you can shed nonessential loads and add additional critical environments.

A good example is a large company that has an end of month processing run that takes up a lot of resources. The environment is only loaded one day a month and other environments that are not needed during this time can be unloaded. When the process is completed, its environment is unloaded and any unloaded environments are reloaded.

You may need to spend some money up front on monitoring and analysis tools as well as training your staff on how to use them, but now you can use your computing capacity to the fullest, saving the cost of additional servers, the power and floor space they require, and the air conditioning that keeps them cool.

Continuous Availability of Applications

For the most part, server virtualization tends to remain within the same data center. As discussed above, you can host multiple VMs on the same server or move VMs from one server to another for load balancing. The ease with which VMs can move from server to server can be leveraged for disaster recovery since your primary and backup data centers can run disparate hardware with the virtualization layer hiding that difference from the software stack.

The network configuration and possibly SAN addresses within the VMs need to be mapped to the hardware appropriate to the data center that is hosting them, but this usually is done through a configuration manager. The software stack within the VM does not need to change.

The criticality of your business processes will define the criticality of the applications that support them, which in turn will define the order in which each VM needs to be brought to a running state. VMs that host non-critical applications do not need to be brought up in the backup data center, and the server location of the VMs that are brought up can be mixed and matched based on the resources available. More VMs which host lower priority applications can be put on the same server because you already have decided that you can tolerate the lowered performance.

By combining all of the pieces described in this article you can achieve close to 100 percent business process uptime no matter what happens. Whether it is a transit strike, white out conditions from a winter storm, a major disaster, or an avian flu pandemic, your business processes will survive. The key to this is geographically dispersed virtualization. Not only do you modularize and virtualize your processing environment, but you geographically distribute it as well. If one site goes down for any reason you jettison lower priority environments and bring up critical environments in a different data center.

In addition to putting your data on a SAN (storage area network), you need to ensure that critical information on the SAN is replicated to one or more backup sites so that it is available when the backup environments are brought up.

By splitting your processing load across multiple data centers, the failure of one doesn't make the service unavailable, it just slows it down until additional environments are brought up in a surviving data center. Additionally by analyzing where your external traffic is coming from, you can locate data centers closer to your customers thereby lowering latency and increasing application responsiveness.

Through the use of clustered computing, load-balancing appliances, data replication, and remote access technologies, your recovery time objective can be brought to zero and your recovery point objective minimized.

Conclusion

By virtualizing your entire environment from applications to operating systems to desktops for your workers you have not only made your organization more resilient, but you can substantially lower your real estate, electrical, and support costs and shrink your carbon footprint. You can deliver better response time to your employees and customers while increasing your ability to survive local or regional outages.

This is not going to be an overnight change – in fact it could take several years. But a journey of a thousand miles starts with one step, and as a business continuity professional, you can lead the way.

Ron LaPedis is a trusted advisor at Sea Cliff Partners which brings together business continuity and security disciplines. He has taught and consulted in these fields around the world for more than 20 years and has published many articles. Ron has two virtualization patents pending and is a licensed amateur (ham) radio operator, instructor, and volunteer examiner. He can be reached at rlapedis@seacliffpartners.com

"Appeared in DRJ's Spring 2009 Issue"

