

REVIEW: SERVER VIRTUALIZATION

VI3 swims through our server consolidation test, showing some amazing capabilities and a few quirks

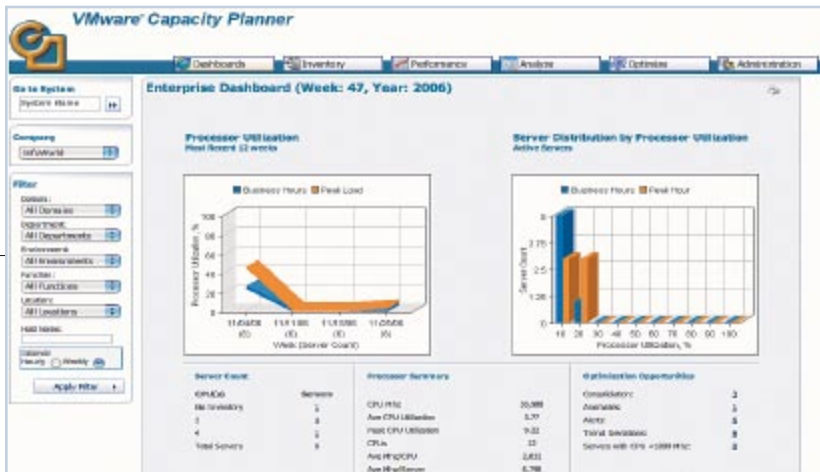
THE SELLING POINTS OF X86 SERVER VIRTUALIZATION are by now common knowledge. By moving systems off dedicated, underutilized servers, and using virtual machines to consolidate them on fewer boxes, you can reduce power, cooling, and space requirements, and you can save a bundle in hardware costs. After the bean counting, VMs can help ease provisioning, load balancing, and disaster recovery.

Less understood is the path to achieving these gains. Once you've caught the consolidation bug, what's really involved in making the move, both in terms of technical requirements and physical labor? And what kind of control do you have over the new environment? To find out, we brought the heavyweight champ of virtualization platforms, VMware Infrastructure 3 (VI3), in for a deep look, subjecting the software and a supporting team of VMware engineers to one of our real-world, Fergensmeir test scenarios.

In the end, VI3 and the VMware team passed our test with flying colors, successfully migrating a number of Windows and Linux systems (see sidebar) and impressing us

DEEP DIVE into VMware's VIRTUAL INFRASTRUCTURE

BY PAUL VENEZIA



VMware Capacity Planner is a combination of local application and hosted service that gauges performance requirements when migrating a physical datacenter into the virtual realm.

with a wealth of useful tools and automated management capabilities. We also discovered some curious limitations in VI3, however, that made our path to a virtual infrastructure a little less straightforward than it otherwise might have been.

There's much more to VI3 than we could fit into the space of this article. For our take on important and even invaluable VI3 capabilities such as VM snapshotting and cloning, VMware's consolidation planning tool, and how the virtualization wave is driving innovation above and below the hypervisor, please see the sidebars to this article online at infoworld.com/4726.

Taking the Plunge

Our test began on a bright October morning. The first order of business was to pick a free blade in our Dell PowerEdge 1955 blade server chassis, install Windows Server 2003, join that server to the domain, and install VMware VirtualCenter Server. This installation was straightforward, with all requirement packages present on the install CD. Although there weren't any Infrastructure 3 servers to manage yet, the groundwork was laid. Next, the first VI3 server was built on a second blade in the Dell cabinet.

Like its predecessor, VI3 is built on a Linux base, leveraging the stability and light footprint of a highly customized Red Hat operating system to provide

foundation elements, but relying on a VMware kernel and VMware I/O drivers and schedulers, to squeeze the most out of the hardware. The Linux folks will immediately notice that the installer is unabashedly built on Red Hat's Anaconda, and installation is generally as easy as booting the CD and clicking Next a few times, ensuring that the required I/O devices are discovered and configured. In the case of our Dell server, I/O was limited to one gigabit front-end NIC and one gigabit back-end NIC for iSCSI SAN interaction. Within a few minutes, the first VI3 server was booting, and the gathered geeks toasted the achievement with a brief swig of Red Bull.

VM Control Center

When the first VI3 server was up and running, we installed the VirtualCenter client on a Windows XP workstation. Unlike the management tools of previous VMware platforms, such as GSX Server, the VI3 management tool base is Windows-only, built on a .Net platform and requiring the most recent Microsoft build. Luckily, the installer detects the current version and prompts the user to download and install the latest release from Microsoft. After this task was completed, it was the work of a few seconds to add the VI3 server to the management console.

VI3 server management in VirtualCenter is based on the familiar hierarchical view of many Microsoft-based

tools, and provides a reasonable amount of sorting and organizing options, including multiple views of the available host servers, virtual servers, and clusters and groups. In the case of our Fergenshmeir Ltd. test scenario, implementing a VMware cluster was the way to go, since the advanced features of VI3 such as HA (High Availability) and DRS (Distributed Resource Scheduler) require a clustered environment. Luckily, this is as easy as right-clicking on the datacenter name defined during installation and adding a new, empty cluster. After that's done, new VI3 hosts are simply added to the cluster — no other configuration is necessary.

In order to use services such as HA, DRS, and VMotion, every VI3 server needs shared storage in one form or another. Fibre Channel and iSCSI SANs are supported, as is standard NFS, but NFS comes with a performance hit. We brought the cluster together by carving a 600GB LUN from the available storage pool on the EqualLogic SAN, and masked to permit access from the dedicated iSCSI NIC on the VI3 server.

Fergenshmeir had implemented a dedicated network segment for iSCSI traffic, both to reduce bandwidth consumption of front-end segments and to enable jumbo frames. This iSCSI segment wasn't routed, however, which presented some problems here. When running with iSCSI, VI3 annoyingly requires that the primary interface of the physical server and any dedicated iSCSI NICs have access to the iSCSI target in order to handle auto discovery. The network admin added this route in the core switch, and after we configured the VI3 host with the proper iSCSI target address, that 600GB LUN was suddenly visible. Notwithstanding the extra step during setup, the iSCSI

VI3 will stop VMotions from occurring unless the processors are nearly identical.

support in VI3 is handled nicely, and it definitely performs well.

Migratory Patterns

With the first VI3 server built and ready for action, we could begin the first of our five P2V (physical-to-virtual) migrations. To handle these migrations, we used VMware's P2V Assistant and a beta release of VMware's new Converter product. In terms of architecture, these two tools couldn't be more different.

P2V Assistant is based on an old Knoppix Live CD, which is booted on the source server. When (and if) all storage and network devices are discovered and configured, you can use a text-based menuing system from the source server console to migrate the server to a VM on a specific destination server. Interestingly, given its Linux roots, P2V Assistant has an easier time with Win-

dows servers than Linux servers, and with newer server hardware than older gear. P2V Assistant had trouble detecting the RAID and NIC hardware in our Dell PowerEdge 2950 and 850 servers, but flawlessly inventoried an HP ProLiant DL360 G3. The migration from physical to virtual took only about 10 minutes, and correctly resized the destination disk, necessary because the source server had more than 60GB of unused space in the primary partition. As soon as the migration was finished, we booted the new VM and powered off the old server. Other than the downtime caused by booting the domain controller from the CD, we encountered no other problems

For the next migration, we put the new VMware Converter tool to the test. This tool offers both live and offline migration options. The live version

runs as a stand-alone application on a Windows server. For this test, the Microsoft Exchange Server 2003 system was selected for migration. Converter has a simple interface that allows admins to supply a Windows server name or IP address and log-in credentials, modify settings for the destination VM, and optionally resize partitions. After that, it's simply a matter of clicking a button to convert the server. During the course of the conversion, we added several users and Exchange mailboxes to the domain to see how complete a live conversion could be under normal operations. In production, you wouldn't want to migrate any servers running databases, e-mail, or other datacenter tasks in this way, but it was a good opportunity to test the thoroughness of the tool. Of the three users added to the Exchange server, the first two, which

Fergenschmeir Ltd. Goes Virtual

THE PROOF-OF-CONCEPT WAS HATCHED THE DAY THAT CRAIG Windham, CFO of our legendary and fictitious enterprise Fergenschmeir Inc., called John Traylor, CFO of Fergenschmeir Ltd. (a wholly-owned subsidiary) and not-so-politely asked about the most recent electrical bills. The message was clear: Your datacenter costs too much — do something about it.

The problem quickly ran downhill. IT Director Brad Richter quickly called a meeting of Fergenschmeir Ltd.'s stalwart and unflinchingly handsome IT staff, and laid out the plan: We need to virtualize.

With lightning speed, an RFP was written and sent to VMware. The task was simply to show us how Infrastructure 3 worked, why it worked, and how to migrate the infrastructure. VMware was on the case.

When VMware's engineers arrived at the Fergenschmeir datacenter, located in the Advanced Network Computing Lab at the University of Hawaii at Manoa's School of Ocean and Earth Sciences and Technology, we provided the gory details:

Fergenschmeir Ltd. ran a variety of systems, including Windows Server 2003 servers, a Windows Server 2003 domain,

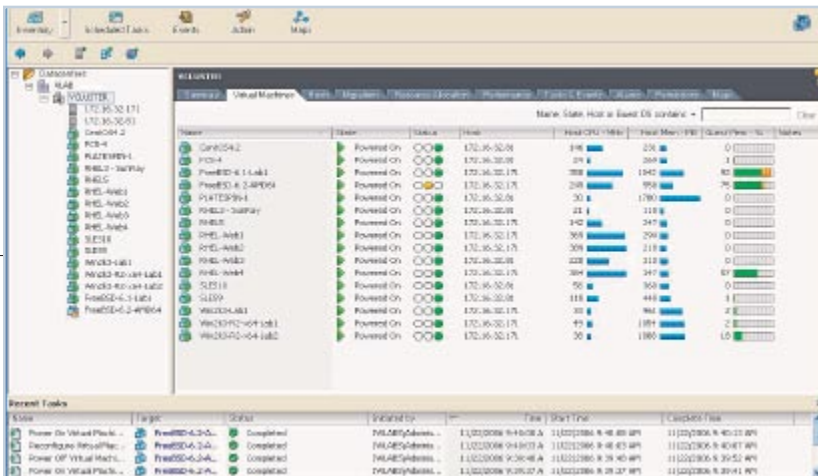
Microsoft Exchange, and Linux-based Web and MySQL servers. In addition to this, the hardware running in the datacenter was a mix of Dell, HP, and white-box servers, with an EqualLogic PS400E iSCSI SAN tying storage together. We gave VMware two days to build the virtual infrastructure from the ground up, based on Dell PowerEdge 1955 blade hardware, and to migrate the collection of production servers to VMs.

Two days proved more than adequate: All five physical servers were successfully migrated before lunch on the first day, giving us another two days to explore, and frequently marvel at, the platform's VM management capabilities.

At the end of the proof-of-concept, the network admins were eager to reclaim large numbers of switchports in the crowded datacenter, the server geeks were too busy playing with VMotion to comment rationally, and plans for new AC units were shelved. VMware had passed the test.

Brad Richter had a sly smile on his face, knowing that this was going to make everyone happy — the CFOs, the IT staff, and most importantly, himself.

— Paul Venezia



VMware's VirtualCenter provides the central admin console for an entire virtual datacenter. This view shows the vital statistics of 16 VMs running Windows, Linux, and FreeBSD operating systems.

were added during the first half of the migration, were present in the resulting VM. The last user, added when the migration was 85 percent complete, was present in Active Directory but missing a mailbox. In addition, the Exchange services failed to start when the Exchange Server VM was initially booted, but they did start manually, and the server appeared to suffer no ill effects from the migration. This form of P2V is certainly attractive, because it requires no real downtime, but should be used only in cases of largely quiescent servers, or for servers with static tasks, such as Web servers and file servers with external storage.

We also tried Converter's offline mode. As with P2V Assistant, this requires booting the server from a CD into a limited OS. Unlike P2V Assistant, however, Converter's CD is built on Windows PE, a curious decision. Converter required far more time to boot than the Linux-based P2V Assistant, and it didn't offer any better hardware detection.

The next migration was the Fergenshmeir file server, a white-box server built from spare parts. Because the CD-based methods weren't likely to recognize the mishmash of hardware, this server was migrated live with VMware Converter. On top of the live migration, we put the server under heavy load at the time, pushing nearly 100MB per second to simulated users. There was even a third wrinkle to this plan: the bulk of

the files being served were residing on an iSCSI LUN.

This scenario presented two obvious options for conversion: Migrate the data on the iSCSI LUN into a VMware disk image, or simply map the LUN straight through to the resulting VM. Both options would retain the storage on the iSCSI SAN, but the second would permit the LUN to be bound by other servers. It turned out that VI3 also offers a middle route here, which is to wrap the existing iSCSI LUN in a virtual disk layer. To configure this, the LUN must be visible from all VI3 hosts, and

the new disk built for the VM selected with Virtual Disk Mode, which adds the virtualization layer.

We chose this third option, and it worked well. In fact, the server's performance held up well during the conversion, and the following reboot/powerdown required to complete the migration caused only a few seconds of downtime. The new VM came up normally, and because the iSCSI LUN set aside for file storage on the original physical server was already mapped to the VM, it immediately began serving files as if nothing had happened. The performance of the file server did suffer a dip in the new environment, as you would expect, averaging 85MB per second through a gigabit uplink to the network that was shared with other VMs. But, all considered, not too shabby.

Having successfully migrated all the Windows servers within an hour, we turned to the two Linux servers, a Dell PowerEdge 2950 running MySQL and a Dell PowerEdge 850 running a Web application linking to that database. Try as we might, none of the VMware tools would properly detect the hardware in these two servers, and the live transfer option was out, because Converter doesn't support Linux.

Officially, VMware doesn't support P2V migrations of Linux servers at all, which is a significant black eye, not only considering that VI3 is built on a Linux base, but also in light of VMware's history of extensive Linux support. Fortunately, it's far easier to migrate Linux servers manually than Windows servers, so we built two new VMs running identical configurations of 64-bit Red Hat Enterprise Linux 4, then copied over the databases and Web applications, all of which was the work of about 90 minutes. VMware

VMware Infrastructure 3

VMware, vmware.com

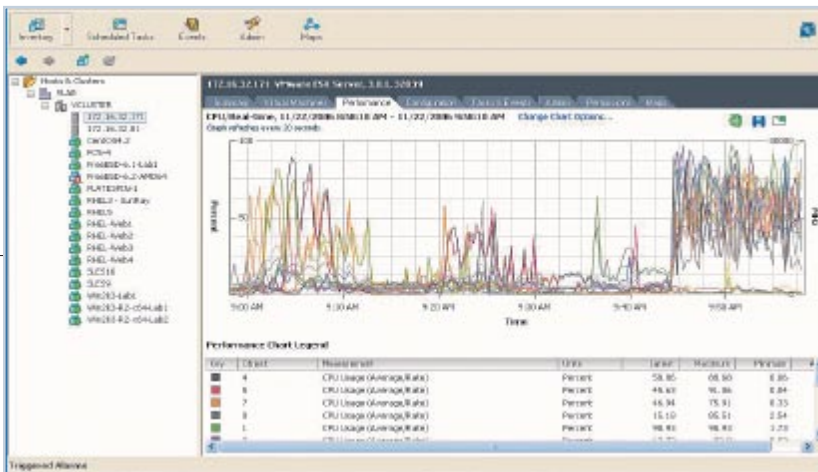
EXCELLENT 8.7

Ease-of-use (25%)	9
Manageability (25%)	8
Performance (15%)	9
Setup (15%)	9
Configuration (10%)	8
Value (10%)	9

COST: Pricing per two sockets: Starter, \$1,000; Standard, \$3,750 (includes Virtual SMP); Enterprise, \$5,750 (includes Virtual SMP, VMotion, DRS, and HA)

PLATFORMS: ESX runs on Intel- and AMD-based server hardware and supports Windows, Red Hat Linux, Suse Linux, Sun Solaris, and Novell NetWare OSes.

BOTTOM LINE: VI3 reinforces VMware's standing as the leading virtualization platform vendor. Plenty of new features including iSCSI support and Distributed Resource Scheduler, as well as improved performance, will grease the wheels of server virtualization in enterprise datacenters. VI3 does have rough edges, including the absence of 10-Gig support, but overall it's a hit.



Performance monitoring within VirtualCenter is strong, offering multiple views of host and VM performance. Here we see the effects of a medium workload on an eight-processor, Opteron-based host.

does plan to officially support Linux P2V migrations in the near future.

VMs in Motion

After these base-layer tasks were out of the way, it was time to show off. We built two more VI3 servers on two more blades and added them to the cluster. These new servers were identical to the first server, right down to the licensing configuration. Because VI3 builds are based on Anaconda, most can be completely automated, and the automation functions are nearly identical to Red Hat's Kickstart server provisioning tools.

As soon as the two new servers were brought online, we configured the cluster for DRS, VMware's resource management framework. DRS manages server load by dynamically distributing VMs across multiple servers to take advantage of all the resources available in the cluster. Although enabling DRS is as simple as checking a box, there's really more to it than that. Every server in the cluster must be configured identically, and all network interfaces and virtual switches must share the same names. Further, every VM must be built on shared storage — in this case, the iSCSI LUN — and VMotion must be enabled on every server.

VMotion is the magic behind VMware's live server migrations, enabling a VM to be moved from one

VI3 host to another without missing a beat. It works by migrating the control over the VM to another VI3 server in the cluster. This migration is achieved by remapping the storage pointer to the new host, moving the memory footprint of the running VM to the new host, sending out RARP (Reverse ARP) packets to inform switches that the MAC addresses assigned to the VM have moved, then actually switching to the new host. The transfer happens within a minute, generally, and it happens seamlessly; the VM doesn't know the difference.

The DRS and VMotion combo is the key to a healthy and scalable VMware installation. There are some caveats, however. VI3 is very sensitive to host CPU differences, and will stop VMotions from occurring unless the processors are nearly identical. This is to prevent running applications that use certain CPU extensions from crashing and possibly corrupting data when they are migrated to a CPU without those extensions. Thus, building a cluster with dual-core and single-core Opteron CPUs in the VI3 hosts is guaranteed to be problematic, and even a cluster with different revisions of Intel EM64T CPUs might not pass muster. Migrating offline VMs between disparate host processor types works because the VM will properly determine the CPU type at the next boot.

To put DRS through its paces, we installed a PHP/MySQL application on our two new VMs, one VM a dedicated Web server, the other a dedicated MySQL server. The application was built to randomly distribute load between the two servers when hit with a large number of Web requests. The front end would serve static pages in response to the majority of the requests, and serve dynamic pages with heavy database calls to a small number of requests, causing the load to shift randomly between the two servers.

With both of these VMs running on a single VI3 host, the load generator was fired up and pointed at the Web server. Within a minute or so, the load on both servers grew, and DRS noticed. As soon as DRS determined that the MySQL server had the highest resource requirements, it automatically moved the VM (by triggering VMotion) to another VI3 host, and the performance of both VMs improved. When we later tasked DRS with high loads on a larger number of VMs, it again moved several VMs around seamlessly to distribute the load evenly among all available VI3 hosts. In fact, DRS responded to the heaviest load across all Windows and Linux VMs by migrating several VMs in the space of two minutes, and the Web and file servers on the target VMs didn't miss a beat. Slick.

DRS can be configured in two ways: automatic and manual. Manual DRS skips the automatic VMotion step, instead informing admins that changes should be made, and providing information on the steps that should be taken, but stopping short of triggering the move.

Failure? What Failure?

At one point during our test, after the full set of servers had been migrated to

VMs and the blades were positively humming, a VMware engineer nonchalantly walked over to the rack and pulled a VI3 blade out of the chassis. VirtualCenter took a few seconds to register that the rug had been pulled out from under the host, and quickly made some changes. All the VMs that had been running on the “failed” blade suddenly appeared under other VI3 hosts and began booting. Within a minute or two, those VMs were up and available. Obviously, problems such as file system corruption that are encountered with any unexpected server shutdown could result, but the downtime was limited to only a few minutes. And although the existing VI3 hosts now had a much heavier load to handle, when the blade was reseated and booted, DRS obligingly spread out the load again, this time with no reboots required thanks to VMotion.

This is VMware’s High Availability in action. Licensed separately, HA can be deployed only in a cluster of VI3 hosts, subject to the same shared storage rules as VMotion. Further, HA is heavily dependent on DNS, which can prove to be an Achilles’ heel. If the VI3 hosts cannot contact each other by DNS name, then they cannot engage in HA actions. If the DNS servers are VMs that were running on failed hosts, for instance, you’re out of luck. VMware has a few recommendations for avoiding this problem, including running DNS servers on physical servers outside the VMware realm, which is rather ridiculous, considering DNS servers are prime candidates for virtualization (their workload is generally low, but the need for availability is quite high). Another option is manually configured host files on the VI3 servers themselves. Hopefully a more elegant solution to this Catch 22 will be forthcoming from VMware.

State of the Virtualization Art

All told, VMware Day at Fergenshmeir was a raging success. Ultimately, VI3 gave us everything we needed to move forward with Fergenshmeir’s virtualization strategy. Some structural issues and missing pieces required some planning to work around, however.

First, no drivers are available for 10-gigabit network interfaces, which can be quite limiting, especially when deploying a large virtualization infrastructure. Network admins would much rather leverage a single 10-gig port on redundant switches per blade chassis or eight-way VI3 server than wrangle a dozen or more network cables and ports per chassis. Implementing 10-gig on a blade chassis or high-capacity server would make it far easier to handle failover and sup-

port high-bandwidth applications, all while simplifying cabling and management.

Also on the networking front, VMware itself seems to be slightly in the dark as to how true load-balancing and fail-over NIC configurations should be handled. VI3 offers simple transmit load-balancing within the network configuration of each host but provides no clear-cut way to enable fully redundant NIC teaming. In fact, VMware engineers at our test site seemed to be at odds about this.

Another oddity is that VirtualCenter doesn’t handle management of VMware Server, which means that an environment running both VI3 and VMware Server requires multiple points of administration, which is rather obtuse. VMware expects to add this support in the future.

VMware is not alone in the x86 virtualization business. The many vendors offering enterprise virtualization platforms include Virtual Iron, XenSource, and Microsoft, which is developing a VMware ESX-like hypervisor and VM management framework that will supersede Virtual Server. Another is SWsoft, whose low-overhead Virtuozzo excels at host-based virtualization and management. (See our review of Virtuozzo 3.0 at infoworld.com/4105, and our beta preview of Virtual Iron 3.1 on page 42.)

VMware certainly has the jump on the competition, as well as the lion’s share of the market at the moment, and the array of features and performance available in VI3 shows why. VMware Infrastructure 3 is clearly the best hardware-emulation platform available today, but the market is changing quickly, the competition is heating up, and VMware will have to keep hustling to maintain its lead. ☞

VI3 Highs and Lows

VMware’s server virtualization platform has many smooth moves and a few rough edges.

The good

- ▶ Distributed Resource Scheduler automatically balances server loads by redistributing VMs across physical hosts.
- ▶ High availability reduces downtime by relocating and restarting VMs in response to hardware failures.
- ▶ Native iSCSI support is strong and fast.
- ▶ New network-based licensing system is simple and dependable.

The bad

- ▶ 10 Gigabit Ethernet is not supported.
- ▶ High availability requires functional DNS, which may cause problems during a failure.
- ▶ Redundant networking is complex and tricky to configure; even VMware has shown some confusion on the proper methods.
- ▶ A few cosmetic bugs lend a slightly rushed feel to VirtualCenter.