

ABOUT THE AUTHOR

Bernard Golden is CEO of Navica, an open source management consulting firm offering strategy and implementation services. Navica serves clients ranging from early stage startups to large IT shops, helping senior management take advantage of open source software while mitigating potential risk.

Mr. Golden is the author of *Succeeding with Open Source* (Addison-Wesley, 2005, published in four languages) and the forthcoming *Virtualization for Dummies* (Dummies Press, late 2007).

Mr. Golden is a recognized authority on open source software, particularly the use of open source in enterprise settings. He writes a regular blog on open source for CIO Magazine, and his work has been featured in technology publications like SearchCIO.com, Enterprise Open Source Journal, LinuxWorld, IT Manager's Journal, InfoWorld, and eWeek, as well as general business publications like Investor's Business Daily and Forbes. He is a frequent speaker on the topic of open source to many organizations and at conferences like the Red Hat Summit, LinuxWorld, the Catalyst Conference, and Data Center Decisions.

Learn more about Navica at www.navicasoft.com.

Mr. Golden can be reached at bgolden@navicasoft.com.

OPEN SOURCE AND VIRTUALIZATION: TWIN TRENDS TO DRIVE CUSTOMER VALUE

By Bernard Golden, CEO, Navica

A Red Hat sponsored whitepaper

EXECUTIVE SUMMARY

Virtualization is one of the hottest topics in IT today. InfoWorld named it as one of the top technology trends of 2007, while the Gartner Group identified virtualization as one of the four key technologies to drive technology further into business. Recognizing virtualization's potential benefits, many IT organizations are implementing or planning virtualization projects today.

Virtualization provides significant benefits, including reduced hardware costs, lower energy costs, improved infrastructure robustness, and more efficient IT operations.

While many organizations are moving forward with virtualization, others have delayed implementing virtualization due to concerns about the technology or the cost.

Following is an introduction to virtualization. Learn about the most common types of virtualization, as well as the most common applications of the technology. Discover the relationship between virtualization and open source, and gain insights into why open source provides significant benefits to virtualization.

While virtualization has been a major success delivered in a proprietary form, the marriage of virtualization and open source promises significant benefits well beyond today's virtualization achievements.

Virtualization presents a new relationship between hardware and software. In the past there has generally been a tight binding between hardware resources and the software packages that access those resources. For example, the hardware resources of a server—the processor, memory, storage, and network connectivity—have traditionally been available only to the operating system installed on that hardware. And operating systems have traditionally been written with an assumption that only one instance of an operating system will access those resources.

Today, however, virtualization has broken the one-to-one relationship between hardware and software. Virtualization encapsulates hardware and makes it possible for multiple operating systems to access a single piece of hardware, each being able to behave as though it were the sole software product able to access that hardware. Virtualization accomplishes this ability to multiplex operating systems by inserting a software layer between the hardware host and the guest operating systems.

Virtualization makes it possible to move from today's "one application, one server" data center architecture to an environment with five to ten virtual servers running on a single physical server. Since individual machines in data centers typically run at 10% to 15% utilization rates, virtualization is an ideal technology to run data centers more cost-effectively.

Several different kinds of server virtualization exist, each with its own strengths and most appropriate use cases. The three most common types are operating system virtualization (containers), hardware emulation, and paravirtualization.

OPERATING SYSTEM VIRTUALIZATION

Often called "containers," this form of virtualization is actually run within an existing operating system. A set of libraries is used to emulate an operating system, thereby allowing applications to run as though an entire operating system is dedicated to them; multiple operating environments can co-exist, each contained within its own "private" instance; thus, the term containers. Containers are very powerful, in that they can be used to allow many different applications to co-exist within a single true operating system, offering each application the illusion of a dedicated operating system and file structure.

Containers are often used in web hosting, where a single machine can be shared among numerous instances of virtual operating systems; each website is offered an environment that appears to be dedicated to it, complete with a root file system and segregated system applications like a web server and a database. One drawback to containers is that the operating environment offered to each container must be consistent with the underlying operating system, making it impossible to run different versions or even different patch levels within individual containers.

HARDWARE EMULATION

This is the most commonly used form of server virtualization, offered by the established leader in the market, VMware. In this approach to server virtualization, the virtualization software provides an emulated hardware environment, complete with virtual processor, memory, storage, and so on, even down to virtual device drivers. A primary advantage of hardware emulation is that actual instances of separate operating systems can run in virtual machines atop the virtualization software: not only can the separate instances of the operating systems differ in terms of version and patch levels, they can also be entirely different operating systems.

Hardware emulation is quite flexible, enabling a single system to support virtual machines running a mix of Windows and Linux simultaneously. Common uses of hardware emulation-style virtualization include server consolidation, where multiple physical servers are converted to virtual machines and installed on a single hardware server; server farms, where a number of virtualized servers act as hosts for virtual machines, with the ability to move (or migrate) virtual machines from one host to another based on load and hardware availability; and server pooling, in which a number of virtualized servers are treated as one large “pool” of virtualization capability, with the virtualization software coordinating all of the actual physical servers coordinates placement and management of individual virtual machines.

Hardware emulation must provide a very high level of fidelity to the guest operating system, so that it is unaware that it is running in a virtualized environment. This requires intercepting and emulating (or translating) all the hardware related operations that the guest initiates. Hardware emulation is extremely powerful due to its flexibility of use; however, the process of intercepting hardware-related requests from the guests imposes a performance overhead, since an additional software layer is inserted between the it and the physical hardware. Also, incorporating new hardware (e.g., storage devices using new communication protocols) can sometimes be a problem with hardware emulation virtualization, because the virtualization software contains the device drivers. Until the virtualization software is released with the drivers to support the new hardware, customers are precluded from taking advantage of it.

PARAVIRTUALIZATION

Paravirtualization is a relatively new approach to virtualization. Instead of inserting a layer of software that emulates hardware and translates calls from the virtual machine operating systems to the underlying physical hardware, paravirtualization uses a much thinner piece of software (called a hypervisor) that multiplexes access by guest virtual machines to the underlying hardware; coordination to the physical resources is controlled by a special guest virtual machine referred to as a “privileged guest.” Most hardware resource calls travel from a guest virtual machine to the privileged guest, which performs the actual calls to the underlying hardware.

Consequently, the paravirtualization approach enables guest operating systems to take advantage of new devices very quickly. As long as the privileged guest has device drivers for the new hardware, virtual guest machines can access the hardware. Performance of paravirtualized systems is typically quite good, since there is little translation necessary.

The primary drawback to this virtualization approach has been its requirement that guest operating systems be configured prior to installation to be able to communicate with the hypervisor. This static configuration requirement has meant that source code for the guest operating systems must be available to enable modification of the hardware calls. While source code access has not been a problem for Linux-based guest operating systems, similar access has not been possible for Windows-based operating systems. This problem is mitigated by the virtualization capabilities of a new generation of processor chips from AMD and Intel. With these new hardware assisted virtualization capabilities, a paravirtualization hypervisor can support unmodified operating systems, making Windows support possible.

STORAGE VIRTUALIZATION

The abstraction of logical from physical resources is applicable beyond servers. While server virtualization is perhaps the best-known use of virtualization, the concept is applicable to data storage as well.

Most IT organizations are drowning in a torrent of data. With the growth of data, many organizations are outgrowing the data storage capacity of their servers. Simply put, applications are generating more data than can be contained in the typical dual-drive configuration of most servers. Beyond the raw capacity issue, many organizations find managing data strewn among hundreds or thousands of servers to be extremely time-consuming and manpower-intensive. Worse, it is difficult to know with certainty that critical data has been backed up and is safe from outages or natural disasters.

To respond to these data challenges, many IT organizations have turned to storage virtualization, moving data away from individual servers to centralized locations that provide greater capacity and better data management. The most common storage virtualization technologies are Network Attached Storage (NAS) and Storage Area Network (SAN).

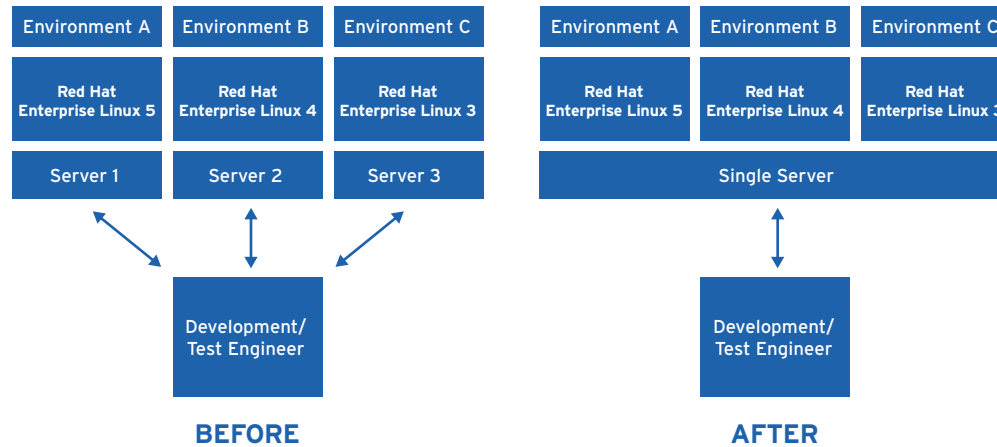
NAS virtualization provides a central location for data storage, typically a dedicated machine with a number of drive bays. While NAS virtualization provides better storage capacity than locally connected disks, true storage virtualization implies using a SAN. SANs not only enable large centralized storage capacity, they also enable flexible management of storage so that the amount of storage assigned to an application can be easily increased or decreased as necessary. SANs also provide redundancy to allow data to be managed independently of the underlying hardware, thereby improving data availability and allowing storage maintenance.

COMMON APPLICATIONS OF VIRTUALIZATION

The application of virtualization spans the entire IT lifecycle and infrastructure. Development organizations often use virtualization to make development and testing more efficient (see Figure 1). By being able to load multiple systems on a single machine, virtualization makes developing distributed applications far easier than the previous method of installing multiple pieces of hardware so that engineers can ensure their

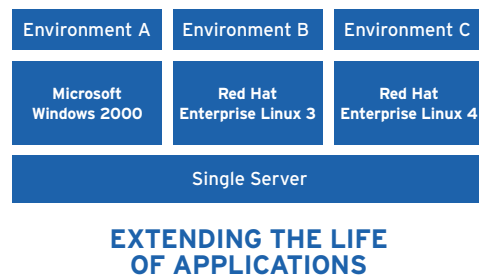
software works across multiple systems. Quality assurance organizations implement virtualization to make testing more efficient; the ability to repeatedly load a virtual machine image rather than perform a physical installation of software speeds up testing cycles significantly. Furthermore, being able to test applications on different operating systems and versions reduces the cost of performing quality assurance.

FIGURE 1
DEVELOPMENT AND TEST WITH VIRTUALIZATION



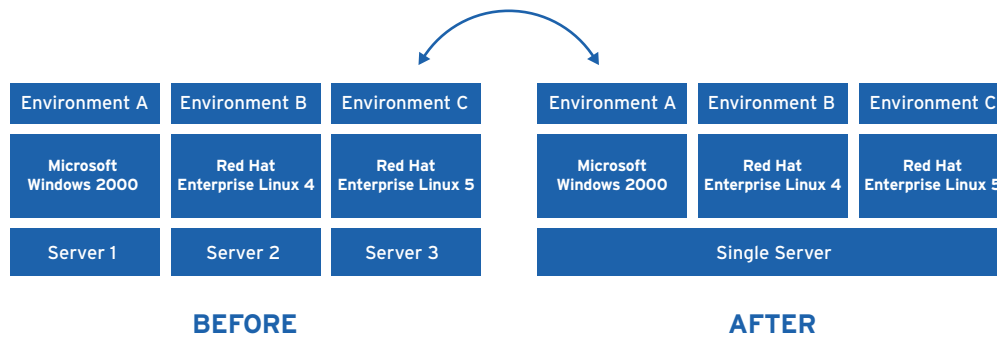
Many IT organizations also use virtualization to extend the life of applications. Many applications require an operating system that is not supported on new hardware. Rather than run the risk of running the application on older fragile hardware, IT organizations convert the old system into a virtual machine that is then run on more modern hardware (see Figure 2).

FIGURE 2
EXTENDING THE LIFE OF APPLICATIONS WITH VIRTUALIZATION



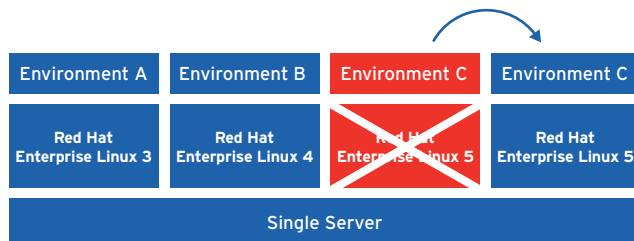
Virtualization is used to affect the way server hardware is used. A common use of virtualization is server consolidation (see Figure 3). In this application of virtualization, one or more physical servers have their software components—the operating system and applications that run on the physical server—moved to a single server. Server consolidation reduces the total numbers of physical servers running in a data center, providing benefits in terms of energy use and reduced capital requirements. While organizations often begin their virtualization journey with server consolidation, many find that the benefits of extending their use of server virtualization are significant, especially once they discover that the low cost and flexibility the technology provides encourages using more applications and systems.

FIGURE 3
DEVELOPMENT AND TEST WITH VIRTUALIZATION



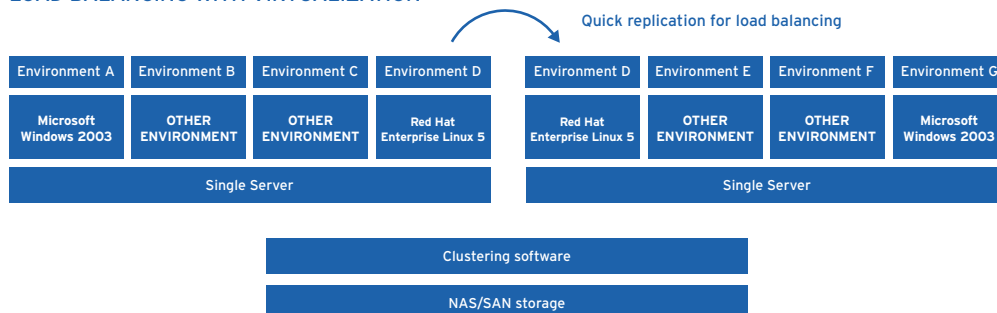
As companies discover the benefits of moving beyond simple server consolidation, they usually begin to explore another application of virtualization called failover (see Figure 4). This refers to the ability of virtualization software to monitor the status of guest machines and, should one crash, quickly restart another instance of the guest. Failover uses virtualization to create a more robust IT infrastructure and can eliminate costly server downtime. Failover can occur within a single server or across multiple servers.

FIGURE 4
FAILOVER WITH VIRTUALIZATION



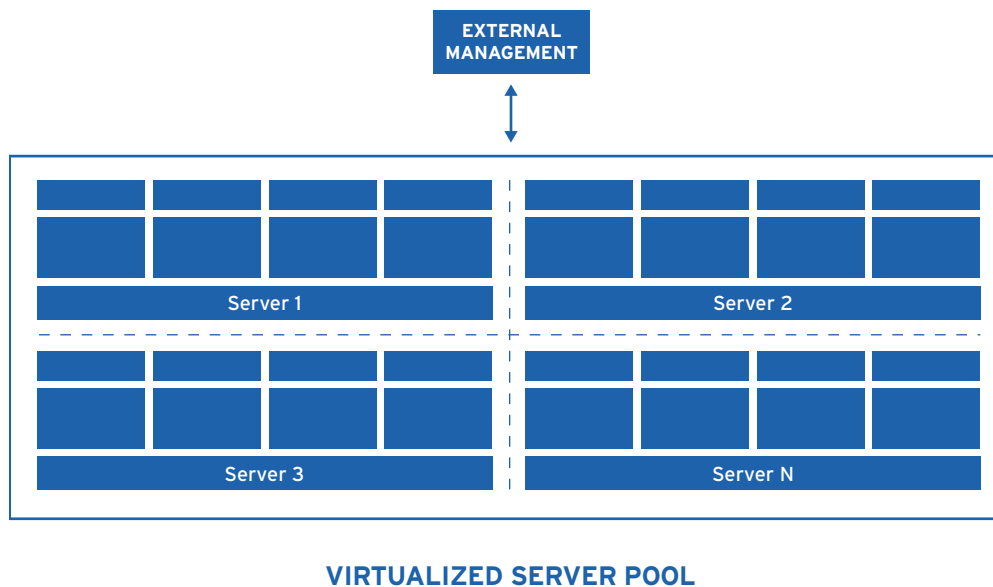
A step beyond failover is the use of multiple virtualized servers to host guest virtual machines. As seen in Figure 5, with the use of two or more virtualized servers in a load-balancing configuration, guest machines can be migrated from a heavily loaded machine to less loaded machines. Furthermore, multiple instances of virtual machines can be run on two or more virtualized servers, enabling better responsiveness to peak demands as well as higher availability of infrastructure. Moving to load-balancing implies using storage virtualization, since data must be available to virtual machines no matter where in the infrastructure they are located; keeping data local to a single physical server would prevent access to it by virtual machines located on other servers.

FIGURE 5
LOAD BALANCING WITH VIRTUALIZATION



Finally, organizations may choose to move beyond managing separate virtualized servers and apply virtualization in a way that treats all the virtualized servers as a single resource. This application of virtualization, called server pooling, uses virtualization software to move beyond locating virtual machines on specific servers; instead, the virtualization software manages the overall collection of virtual machines and migrates them as necessary to keep individual server loads even (see Figure 6). New virtual machines are installed by the virtualization software on lightly loaded servers to maintain consistent loads throughout the server pool. Naturally, it's critical to implement storage virtualization in this virtualization scenario, since there is no specific server assigned as the host for a given virtual machine.

FIGURE 6
SERVER POOLING WITH VIRTUALIZATION



Many IT organizations begin with modest server consolidation projects. Once server consolidation is successfully implemented and the benefits of virtualization become clear, additional applications of the technology in areas like failover and load-balancing are implemented. Server pooling is a relatively new virtualization application scenario, but its potential is recognized and many organizations have put it onto their virtualization roadmap.

BUSINESS BENEFITS OF VIRTUALIZATION

When considering investing in a new technology, organizations typically want to understand the financial and operational benefits available. While virtualization is certainly highly regarded from an innovation perspective, what are the areas that most organizations find the greatest benefit from applying this new technology?

LOWER CAPITAL INVESTMENT IN HARDWARE

Perhaps the most obvious place virtualization provides financial benefit is in the area of hardware investment. Virtualization's ability to consolidate servers implies that every virtual machine means a server purchase can be avoided.

With many organizations supporting five to ten virtual machines per physical server, it's clear that the capital savings that can be achieved through the technology can be very impressive.

Beyond the savings available through reduced hardware purchasing, virtualization also helps many organizations avoid a common problem: data center saturation. With the increasing move to digitized business processes, many companies have filled their data centers to capacity. By using virtualization to reduce overall machine count, companies have been able to avoid the need for data center expansion, which can run into tens of millions of dollars.

REDUCED ENERGY COSTS

Every machine in a data center, whether running at a 10% utilization rate or at the 70% typical post-virtualization utilization rate, draws power and requires cooling. With the recent rise in electricity costs as well as the increased number of machines in data centers, energy costs have become a major issue for many companies.

Since virtualization provides the ability to reduce the overall number of machines in data centers, it is an ideal solution to address the problem of energy costs. Many organizations find that they reduce their power costs by 20% or more after moving to virtualization. A secondary benefit is that data centers temperatures often drop, due to lower load levels imposed on overtaxed air conditioning systems.

MORE ROBUST AND RESPONSIVE IT INFRASTRUCTURE

By applying some of the more advanced virtualization uses like failover and load balancing, virtualization helps IT organizations create a more robust infrastructure. Instead of a system crash taking an important application down for hours, virtualization enables a second virtual machine to be instantiated in seconds, dramatically reduced system outage time.

Furthermore, the ability to rapidly provision new virtual machine instances allows companies to more quickly respond to business condition changes. For example, if a company experiences heavy load due to seasonal demand or increased publicity, it can rapidly instantiate additional virtual machines instances to ensure the increased user demand receives acceptable response times.

MORE EFFICIENT OPERATIONS

Virtualization can also make IT operations more efficient. Managing fewer pieces of hardware obviously requires less system administration for installation and maintenance. Beyond this obvious benefit, virtualization can also help IT organizations more efficiently manage their ongoing hardware maintenance programs.

Instead of needing to schedule important hardware maintenance for off-hours or weekends, a server can have its virtual machines transferred to other servers and then can be maintained during regular working hours. This pays benefits not only in terms of reducing expensive overtime, but also enables organizations to have their most experienced talent do the work without requiring extra hours, thereby raising morale.

OPEN SOURCE AND VIRTUALIZATION

Virtualization offers enormous benefits to IT organizations. However, several factors have limited its adoption.

First, it has traditionally been the province of proprietary vendors. Some of these vendors are hardware manufacturers who provide virtualization software that enables better use of their hardware. Unfortunately, this approach allows the software to be used only on hardware from that manufacturer. For organizations that use hardware from multiple vendors, this is clearly undesirable, since it presents the need to run multiple virtualization products, which increases infrastructure complexity and reduces operational efficiency, thereby raising costs.

A more attractive proprietary approach comes from independent software vendors (ISVs); their software can be run on servers from many different manufacturers. This enables organizations to reduce the variety of virtualization software needed and to use a consistent virtualization infrastructure throughout their data center. Using ISV software from a vendor like VMware is clearly more attractive than attempting to manage multiple products from several hardware manufacturers.

However, using an ISV's virtualization offering presents other shortcomings. Relying on a single proprietary software vendor, while reducing complexity, creates dependency on that vendor. This dependency can also be characterized as vendor lock-in, symbolizing the fact that the user organization is trapped into a restrictive relationship, dependent upon the vendor's product decisions and time frames. From an end user perspective, this situation is extremely undesirable, since it means that end user plans are subject to the feature road map and release schedule of the vendor. If a user organization needs a particular feature to meet business requirements, it may be unable to obtain that feature in the necessary time frame. An associated shortcoming of the single vendor decision is early retirement; in this scenario, a vendor ends support for a software product, forcing the end user to upgrade or modify its infrastructure on a time frame not of its own choice.

Single ISV providers of virtualization present another kind of challenge in terms of evolution of the product and, crucially, inclusion of new technology innovations in the product.

There are many new hardware innovations being developed throughout the technology industry that hold great promise for increasing the value of virtualization. For example, new networking technologies, offering higher performance and better virtualization support are being made available from manufacturers like Intel and Siemens.

For virtualization products that come from proprietary ISVs, integration with these new technology products can proceed only at the pace of the ISV. This means that important technology innovations may not be available in the time frame a user organization desires.

Finally, of course, proprietary software is typically associated with high prices. Many users feel that virtualization products carry a burdensome price tag, diminishing the potential attractiveness of the technology. The high price of proprietary virtualization software means that many organizations are precluded from using it or are applying it in useful, but less critical applications. For example, the high price of virtualization software keeps some organizations from applying virtualization in areas like development and testing or in pilot implementations of business applications.

For these and other reasons, open source offers significant benefits when compared to the proprietary approach to virtualization.

OPEN SOURCE VIRTUALIZATION: RAPID EVOLUTION OF THE TECHNOLOGY

Proprietary software means that a single vendor controls the code base of the product. Only the vendor's employees can make changes to the product's code. Moreover, people or companies with innovative ideas, unless they work for the vendor, are precluded from direct contribution to the product and forced to work through the proprietary vendor. Naturally, this arm's length approach retards the evolution of the product.

By contrast, open source products welcome participation from a much larger pool of contributors. New ideas and technical approaches can be incorporated into the virtualization product much more rapidly than a proprietary counterpart.

Furthermore, support for complementary products such as new networking technologies or processor components can move forward very quickly. The developers of the technologies (e.g., networking hardware companies) can directly participate in developing support for their products in the virtualization software. This means that innovations can be tested and incorporated by the experts in the technology, thereby accelerating final support of the technology in the product. Naturally, this has the effect of rapidly improving the functionality of the open source product; the ability of open source to improve more quickly than proprietary software has been seen in a number of software markets, and this dynamic is present in the virtualization market as well.

CONTRIBUTION BY MEMBERS OF THE VIRTUALIZATION ECOSYSTEM

One of the key challenges for proprietary vendors is the limited perspective they bring to the development of their software. This is inevitable, since they have less experience using their software than real-world end users. No single company can know as much about the variety of applications of their software as the overall ecosystem of system users.

The role of complementary technology providers was touched on in the previous section. Certainly their ability to directly test and integrate their products offers the opportunity for their domain knowledge to be included in the open source product. This means that the project benefits from the detailed knowledge of these members of the product ecosystem.

Other members of the ecosystem can directly contribute to the product as well. If we look at Linux, it's clear that much of the code contribution to the project comes from individuals employed by vendors providing products complementary to the operating system. By having their employees working on the Linux code base, these vendors are able to ensure drivers for their products work well with Linux. Furthermore, by participating in the product development community, these vendors are also able to ensure that developers working in other portions of the kernel are aware of hardware requirements and can keep these requirements in mind while designing and coding their work. This kind of consistent, engineer-to-engineer communication pays dividends well beyond the kind of business development relationship typically engaged in by proprietary companies. By collaborating on open source work, members of a product's ecosystem can improve the product's support of new hardware and software technologies.

Contributions by domain experts can extend well beyond vendors that manufacture products for the domain. For many open source projects, sophisticated end users also contribute source code improvements to the project, thereby ensuring that innovative product functionality is integrated into the product. This can extend to product features, which, were the product proprietary, would not be deemed important enough for the vendor to implement. In open source, there is a way for these features to be implemented and offered to other product users.

INCLUSION IN THE OPERATING SYSTEM

In the proprietary world, because companies seek to protect their markets, product integrations are often implemented by inferior technical solutions; while not desirable from a product use perspective, these integrations are necessary to ensure that each vendor's product can continue to be sold separately. Put another way, in the proprietary world, technical usefulness often takes a back seat to marketplace realities. The less desirable technical outcomes manifest through complex and unstable integration mechanisms, as well as through poor levels of product performance.

By contrast, open source technologies can be contributed and directly integrated into the operating system kernel. This means that integration is high performance and elegant.

The inclusion of features in the operating system also pays benefits to end users in terms of reduced costs. Because features are integrated directly into the operating system, it is no longer necessary to purchase and install additional software products. Instead, the functionality is easily available in the base operating system, often made available by nothing more complex than filling in an option during operating system install.

Furthermore, with the inclusion of key functionality into the operating system itself, ongoing upgrades and maintenance is easier as well. Instead of the end user needing to keep track of multiple products' release schedules and installing and testing a software stack each time one of the products is updated, the software distributor takes care of ensuring that the products are kept up to date and released in a consistent, tested package.

LACK OF VENDOR LOCK-IN

User organizations have learned from the past. Ceding control of significant parts of their computing infrastructure imposes significant costs and risks upon them. While vendors cite the benefits of standardizing on a single vendor's product stack - benefits like integration and "one throat to choke," users have learned through experience that these benefits are inevitably accompanied by significant drawbacks: poor responsiveness, ongoing mandatory upgrades, and inability to influence product direction. Many organizations have concluded that the latter drawbacks outweigh the benefits of single vendor dependence; in fact, the industry term for this relationship is "vendor lock-in," which illustrates the restrictions users experience when dependent upon a single vendor.

Open source software, by its very nature, reduces vendor lock-in. The source code for every open source product is available for any user to examine, modify, and redistribute. Consequently, this means that should a user be dissatisfied with an open source vendor, the user can terminate their relationship with the vendor and look to other avenues for development and support for the product. These other avenues can include other vendors, other members of the product community, or even the end-user.

While open source vendor/user relationships are not perfect, the kind of protracted user dissatisfaction, marked by resentment and bitterness from users, is unheard of in the open source world. The fact that end users have options beyond the vendor tends to discipline vendors and ensure a positive working relationship.

With regards to virtualization, the lack of vendor lock-in means that, should a customer or user decide that they wish to move to a different vendor, it's easy to migrate. Since all participants in open source share a common code base, vendors are precluded from holding users hostage. Should a user of Linux-based virtualization choose to move away from their current supplier, they can do so, secure that already-existing virtual machines and configurations will work properly with other Linux distributions.

COMMUNITY

Community is one of the most impressive aspects of open source. Consisting of developers, users, and other members of the product ecosystem like hardware manufacturers and system integrators, the community provides a rich resource for learning, support, and collaboration.

The concept of community extends well beyond people contributing code to a project. Because of the transparency typical of open source, there is open and frequent conversation among all community members.

Community provides a way for a broad range of users to contribute their knowledge and experience to the product, unlike the proprietary world where only some customers-typically those with the largest wallets-are able to make their views and requirements known. The meritocracy of community means that all users can share their perspective.

Moreover, community members can contribute to the product, even if they do not submit code. Bug reports, tutorials, and technical support are all valuable contributions from community members. One has only to look at the mailing list or forums of a typical open source product to recognize the way that community members assist one another, often

enabling product users to solve problems much more quickly than in a commercial support arrangement.

In the area of open source virtualization, the involvement of community members has enabled awareness and product uses to spread much more quickly than if the product were commercially based.

COST

Last, but certainly not least, is the cost advantage open source presents versus the proprietary competition. Because of the licensing conditions proprietary software is distributed under, proprietary products require a costly license fee prior to use, with mandatory maintenance fees a prerequisite for continued use and support.

By contrast, the expansive licensing conditions of open source imply that software is freely available for download at no cost. Users have full access to the product with no requirement for a financial transaction. And, while commercial support and services are available if desired, there are no financial requirements for continued use of the product.

Consequently, open source software can be far more cost-effective than proprietary software. The cost advantage of open source is very important with respect to virtualization software. Because previous virtualization products have been an add-on software product, carrying their own licensing fee, users have been forced to find additional budget in order to implement it. Furthermore, because commercial virtualization products are so expensive, IT organizations have been forced to move to dense virtualization infrastructures in which they load the largest possible number of virtual machines per physical server in order to reduce the overall cost of virtualization. Clearly this high density poses risk, since maximizing server load means that any individual failure can affect many systems.

Despite the manifest business benefits of virtualization described earlier, the high cost of commercial virtualization has put IT organizations in an awkward position. They need to spend significant sums on software before beginning to realize the financial benefits of virtualization technology. The high cost of commercial virtualization certainly has slowed the pace of adoption of virtualization. But with the advent of open source virtualization, the adoption of virtualization technology is expected to skyrocket.

OPEN SOURCE AND VIRTUALIZATION: A SUMMARY

Virtualization is a powerful technology transforming today's data centers. Organizations are quickly moving to implement virtualization throughout their infrastructures to realize the many benefits of virtualization.

VIRTUALIZATION OFFERS SIGNIFICANT BUSINESS BENEFITS

There are many business benefits of virtualization.

Better hardware utilization. By implementing virtualization on data center servers, IT organizations can raise the utilization of their servers from 10% or 15% to as high as 70% to 75%, achieving much better efficiency while still allowing performance headroom for demand spikes. Higher utilization rates also make more efficient use of corporate capital.

Reduced energy costs. Running servers at low utilization rates not only wastes capital, it wastes energy spending as well. Even though a machine may be running at only 15% of its capacity, the machine may still use almost the same amount of energy as if it were running at 90% utilization. By using virtualization, fewer pieces of hardware are required, and thereby energy use is reduced.

Reduced crowding in data centers. Many organizations have maxed out their data center capacity. They cannot fit any more machines into the rack space they have available. This is a critical issue, since the cost of increasing space is typically not incremental; increasing data center space easily can run into tens of millions of dollars. Virtualization reduces the number of servers needed and obviates the need to increase data center floor space.

Better operational efficiency. All hardware needs to be maintained and repaired. Sprawling data centers packed full of machines impose large operational costs as systems are taken out of service for repair and upgrade. Using less hardware means less system administration work and enables better operational efficiency.

APPLICATION PROFILES OF VIRTUALIZATION

There are many applications of virtualization. Many organizations begin by the most direct application of virtualization: server consolidation. Server consolidation is straightforward; the operating systems and applications running on a number of separate pieces of hardware are migrated into a number of virtual machines running on a single piece of hardware that has virtualization software running on it. Depending on how many physical servers are migrated, server consolidation can achieve hardware reduction ratios of five to one or even ten to one.

However, virtualization offers significant benefits beyond mere physical consolidation of servers. Virtualization software can be configured to automatically restart (failover) virtual machines upon a system crash, thereby increasing the robustness of IT infrastructures and improving IT responsiveness.

A step beyond failover is using virtualization's capability of migrating machines between one server and another to implement load balancing, which refers to the use of multiple systems to better ensure system performance and robustness. By running multiple systems, virtualization enables high traffic loads to be shared across application instances running on different machines, thereby achieving better use of hardware capability and also removing performance bottlenecks.

It's a short step from load balancing to server pooling. While load balancing refers to the ability to configure two or more machines to share application load, server pooling refers to completely abstracting the relationships between application environments (that is, applications and the underlying operating systems) to the point where the virtualization software itself controls what specific machine an individual application environment will be installed upon. Server pooling uses virtualization to create a large pool of hardware resources, all capable of accepting new virtual machines as they are inserted into the pool.

OPEN SOURCE AND VIRTUALIZATION ARE NATURAL COMPLEMENTS

The many advantages of open source software improve the already impressive benefits of virtualization. Since the development process of open source enables a far larger pool of developers to contribute to the technology, open source is quickly improving the capability of virtualization.

Because many hardware manufacturers participate in open source development, open source virtualization can rapidly benefit from hardware achievements. For example, the recent work in network virtualization can quickly be incorporated into open source virtualization, since the manufacturers of the network virtualization have individuals working to integrate the technology into the main code base of the Linux operating system.

Furthermore, by using open source virtualization, end users avoid the danger of vendor lock-in posed by use of proprietary virtualization technologies. Using open source virtualization also enables end users to benefit from the community surrounding it. Finally, open source virtualization is far more cost-competitive than its proprietary counterparts, since the technology is contained within the operating system rather than being a separate software product.

The obvious business benefits of virtualization can be realized in a variety of application profiles: server consolidation, failover, load balancing, and even server pooling. Open source allows end users to achieve the full promise of virtualization in a form most congenial to user control and technology cost-effectiveness.