2

Xen Virtualization

Xen ▌Ian Pratt · Stephen Spector

Xen Introduction

Enterprises looking to increase server utilization, consolidate server farms, reduce complexity, and decrease total cost of ownership are embracing server virtualization. The Xen® hypervisor is the fastest and most secure infrastructure virtualization solution available today, supporting a wide range of guest operating systems including Windows®, Linux®, Solaris®, and various versions of the BSD operating systems.

With Xen virtualization, a thin software layer known as the Xen hypervisor is inserted between the server's hardware and the operating system. This provides an abstraction layer that allows each physical server to run one or more "virtual servers", effectively decoupling the operating system and its applications from the underlying physical server.

The Xen hypervisor is a unique open source technology, developed collaboratively by the Xen community and engineers at over 50 of the most innovative computing solution vendors, including AMD, Cisco, Citrix, Dell, Fujitsu, HP, IBM, Intel, Network Appliance, Novell, Oracle, Red Hat, Samsung, SGI, Sun, Unisys, and VALinux. Xen is licensed under the GNU General Public License(GPL2) and is available at no charge in both source and object format. Xen is, and always will be, open source, uniting the industry and the Xen ecosystem to speed the adoption of virtualization in the enterprise.

The Xen hypervisor delivers a secure environment for enterprise computing centers by selectively isolating critical components and delivering a lean core hypervisor with less than 150,000 lines of code. This translates to extremely low overhead and near-native performance for virtual sessions as well strong isolation to eliminate the impact of misbehaving virtual sessions. Moreover, this enables Xen to be configured for protection from device driver failure and protects both guests and the hypervisor from faulty or malicious drivers.

Xen History

Xen originated as a research project at the University of Cambridge, led by Ian Pratt, senior lecturer at Cambridge and founder of XenSource, Inc as well as Keir Fraser who architected and developed the initial hypervisor. XenSource took the Xen research project and continued the open source Xen project while also developing an enterprise virtualization solution on Xen. The first public release of Xen was made available in 2003.

In October 2007, XenSource was acquired by Citrix Systems and the Xen open source project was moved to http://www.xen.org. The Xen project is now managed by an Advisory Board(Xen AB), which currently has members from Citrix, IBM, Intel, Hewlett-Packard, Novell, Red Hat and Sun Microsystems. The Xen AB is chartered with oversight of the project's code management procedures and with development of a new trademark policy for the Xen mark.

The community website, www.xen.org, contains a blog (http://blog.xen.org), a wiki(http://wiki.xensource.com), a bug tracking system(http://bugzilla.xensource.com), and the latest source code and binary deliverables (www.xen.org/download). The community also holds regular meetings called Xen Summits which are held globally to ensure that all members have an opportunity to attend. Previous Xen Summit slides and videos are also available at www.xen.org/xensummit.

The mission of the Xen.org project focuses on six key areas:

- *Build the Industry Standard Open Source Hypervisor* – develop a core "engine" that is incorporated into multiple vendor's products
- *Maintain Xen's Industry Leading Performance* – be first to exploit new hardware acceleration features and help OS vendors paravirtualize their operating systems
- *Maintain Xen's Reputation for Stability and Quality* – emphasis on security for enterprise deployments
- *Support Multiple CPU Types; Big and Small Systems* – from servers to smart phones
- *Foster Innovation*
- *Drive Interoperability*

Xen Architecture

A Xen virtual environment consists of the following components:

- Xen Hypervisor
- Domains [Guests]
- Domain Management & Control Tools
- Virtualized Devices

Xen Hypervisor

The Xen hypervisor is the basic abstraction layer of software that sits directly on the hardware below any operating systems. It is responsible for CPU scheduling and memory partitioning of the various guests running on the hardware. The hypervisor not only abstracts the hardware but also controls the execution of guests as they share the common processing environment. The core hypervisor is the most privileged software running on the system.

Domains(Guests)

A Xen virtual environment contains a series of independent and isolated guests that access the processor and other computing functions via the Xen hypervisor. These guests are categorized into two types based on their need of an Intel vt-d or AMD-v processor. Fully-virtualized guests are not necessarily aware of the Xen hypervisor and cannot run as a guest unless the Intel or AMD virtualization hardware is present on the machine. Paravirtualized or enlightened guests(Microsoft terminology) are aware of the Xen hypervisor and do not require the Intel or AMD processors to run properly. Xen hypervisors are capable of supporting all guest types including fully-virtualized guests with additional enlightened capabilities which enhance their knowledge of the hypervisor for better performance.

The Xen virtual environment can run any type guest regardless of operating system and level or enlightenment.

Domain Management & Control Tools

A collection of management and control tools support the overall Xen virtual environment and run in guests on the Xen hypervisor. The tools support a variety of functions that are required to manage the Xen virtual environment including the start and stopping of domains. Typically, domain management controls are run in a single guest commonly referred to as domain 0; however, more advanced Xen configurations have moved many control functions into separate domains to ensure that control functions only have privileged access rights they need. This enhances overall security of the Xen virtual environment.

Virtualized Devices

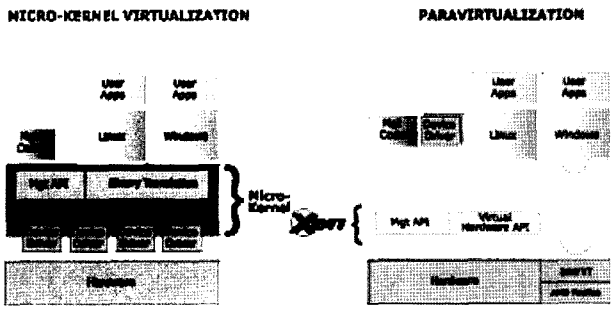
The Xen virtual environment handles network and disk access from guests to the physical hardware via management and control guests specifically designed to process these requests. The management and control guest contains the "back-end" driver which communicates directly to the hardware and is able to receive data from guests running on the hypervisor. The guests wishing to do a disk or network access will leverage its own local driver which sends data to the guest with the back-end driver instead of the hardware.

Xen Feature Highlights

Paravirtualization

Paravirtualization, developed by the founders and early members of the Xen hypervisor project, fundamentally altered the way virtualization technology was architected. With this technology the virtual servers and hypervisor co-operate to achieve very high perfor-

mance for I/O, CPU, and memory virtualization. The Xen hypervisor appears to the virtualized server as an idealized hardware abstraction layer that offers superb performance. In fact, the Xen hypervisor offers a smaller code base, greater security, and less overhead then alternative virtualization approaches.



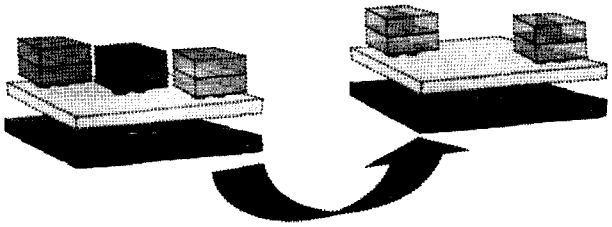
In addition, the Xen hypervisor takes advantage of hardware virtualization support from Intel and AMD processors to enable fully-virtualized guests to run on the hardware while still achieving very high performance. With alternative approaches, the hypervisor must binary patch running guests to prevent them from interacting with the hardware, resulting in high performance overhead, stability, and security risks. Moreover, this approach results in significant I/O performance impact.

It also leverages all of the native device drivers and therefore supports an extremely diverse set of drivers. Xen's paravirtualized drivers run outside the core hypervisor, where they implement policy for resource sharing between multiple guests, providing fine-grained partitioning of I/O between multiple virtual servers. Another benefit of this approach is that drivers run at a lower protection level than Xen, making the hypervisor impervious to driver failure.

Live Relocation

The ability to move an active virtual machine from one server to another without interruption of service is known as Live Relocation. This feature enables administrators to easily accommodate for system downtimes, power conservation, high-availability services, and hardware maintenance.

A shared storage solution such as NAS, SAN, or other block sharing service such as drdb is required to support live relocation as both machines must be able to access the guest's virtual disk. All local memory



is synchronized from one machine to the other for data consistency during the start of live relocation so any migration will start with a duplicate copy of memory on the back-up machine.

For a complete technical understanding of the architecture behind live relocation, the following paper is available from the 2004 NSDI proceedings: http://www.cl.cam.ac.uk/research/srg/netos/papers/2005-migration-nsdi-pre.pdf.

Xen Security

As the adoption of open source Xen continues within the Enterprise, security becomes a critical feature for customers. Having a secure hypervisor is critical to ensure that the multiple virtual machines running on a single hypervisor are not corrupted by other virtual machines, external threats, or the hypervisor itself. Xen protects itself from these attacks by leveraging new hardware features in the latest processors in chipsets such as IOMMU. Xen further strengthens security by moving management features into separate guests to restrict privileges. Keeping the Xen hypervisor code base small and efficient also helps prevents attacks as less software means less opportunities for intrusion as well as giving developers a smaller solution set for code inspection and review.

Xen also uses hardware security capabilities as Trusted Platform Modules(TPM) to build a layer of attestation and trust up from the hardware through the software. TPM provides a set of encrypted keys that allow the hardware and software to ensure that nothing has been changed in the existing environment thereby guaranteeing the user a secure environment. More on TPM can be found at http://en.wikipedia.org/wiki/Trusted_Platform_Module and https://www.trustedcomputinggroup.org/home.

Xen Performance

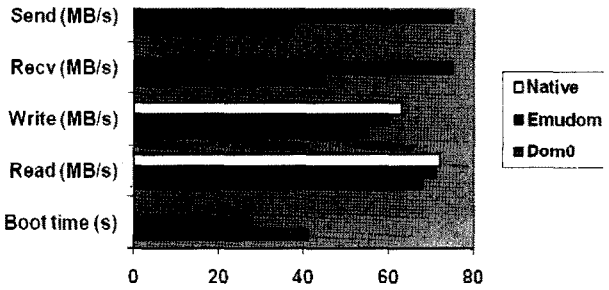
The latest release of the Xen engine, Xen 3.3 is

the most scalable, secure, and high performance hyper-visor available in the marketplace.

Emulation Domain Feature

This feature is the first in a series of changes coming to the Xen hypervisor solution as more service features are moved from Domain 0 into a series of separate guests and control functions. This change allows for greater scalability and security as individual service guests are able to run faster without blocking for other services within the guest as well as reducing the amount of services that run with elevated privileges.

The following diagram shows performance of the new emulation domain feature in Xen 3.3 compared to the previous version of Xen and native performance without Xen. In this new feature, the emulation software was moved from Domain 0 to a small guest that runs independently.



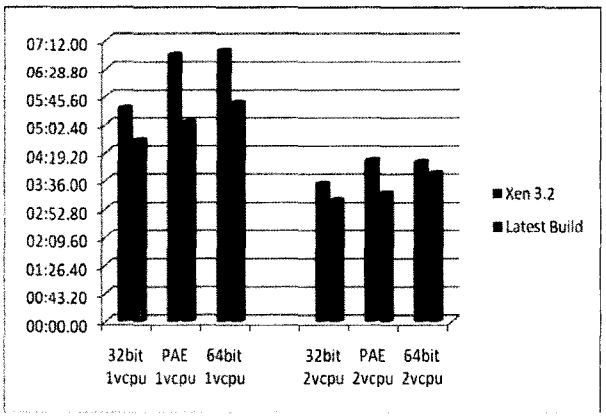
The data clearly shows that boot time is significantly reduced with this new feature and the send, recv, write, and read performance data shows improvement as well. Finally, the new feature is also approaching native benchmarks for writing and reading.

Shadow3 – Shadow Page Tables

Virtual environments require sophisticated memory management as guests see a memory address space that appears as a contiguous memory on physical hardware; however, in reality the memory is likely distributed throughout the local memory structure. The hypervisor must handle the memory translation of the virtual memory address space to the physical hardware using shadow page tables. These tables maintain a memory translation of the virtual address space for a given guest to the physical memory address space. Shadow page tables must be in synchronization with the guests' memory tables to ensure data consistency.

Shadow paging overhead is one of the largest source of cpu virtualization overhead for fully-virtualized guests. Because these guest operating systems don't know the physical frame numbers of the pages assigned to them, they use guest frame numbers instead. This requires the hypervisor to translate each guest frame numbers into machine frames in the shadow pagetables before they can be used by the guest. A series of changes to the shadow pagetable algorithms implemented in Xen 3.3 have greatly reduced the overhead from these issues. More information on the changes available on this blog post.

This table shows the Window 2003 DDK build time for a 32/64 bit single and double virtual CPU system with Xen 3.2 and the new Xen 3.3 release. The new feature delivered 15% to 26% improvement in time for a single virtual CPU and 7% to 20% improvement for a double virtual CPU.

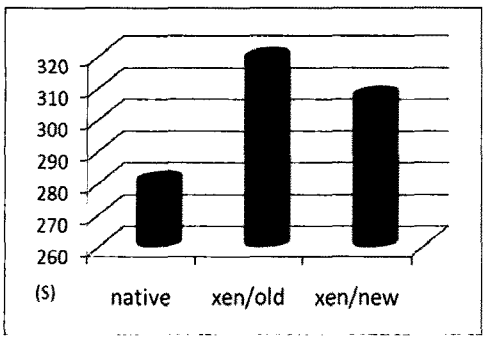


Xen also supports hardware solutions including AMD-V nested page tables and Intel-Vt extended page tables.

Domain Lock Removal for Guests

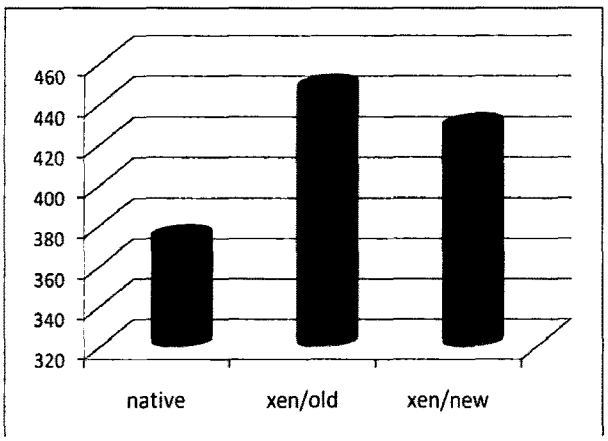
Another important feature in Xen 3.3 is the removal of domain lock for page table updates on enlightened guests. In previous versions of Xen, only one virtual CPU(VCPU) at a time per VM could update its page-tables. This leads to scalability issues for individual guests as they leverage more VCPUs. In Xen 3.3, the bottleneck of updating 1 VCPU at a time has been removed and is showing excellent performance improvement.

Parallel kernel build on an 8 VCPU PV Linux guest(32b and 64b)
32 bit, Intel server



Overhead reduced from 14% to 10%

64 bit AMD server



Overhead reduced from 20% to 15%

Xen Future

Xen Client Initiative

Xen is currently being developed to work on a variety of client devices such as laptops, cell phones, or other portable machines. The Xen Client Initiative is a new project underway in the community with support from vendors such as Intel, AMD, Samsung, Lenovo, Phoenix, IBM, Neocleus, and Fujitsu. The project has four main areas of focus:

- Client Hypervisor Development – updating the existing Xen hypervisor to support client specific features such as graphics cards, battery status, wireless devices, etc.
- Virtual Machine Services – development of cross-vm services that all client virtual machines can leverage such as sharing network interfaces, hard drive access, encrypted storage, etc.
- Hypervisor Domain – developing rules and procedures for virtual machines to operate on a given client device
- Client Hypervisor Service Framework – development of SDKs and APIs for industry standardization on running virtual machines on client devices

Samsung's recent announcement and release of the Xen hypervisor ported to the ARM processor(ARM-9 266MHz/Freescale i.MX21) at the Xen Summit in Boston, MA is an example of delivering the Xen hypervisor for the next generation of 3G/4G mobile phones. Details of this project are available at http://wiki.xensource.com/xenwiki/XenARM.

The Xen Client Initiative is ongoing and all information is available at http://wiki.xensource.com/xenwiki/ in the Xen Client Initiative Section.

Ian Pratt is the chief architect of the Xen project, and chairman of xen.org. He has played a key role in both the architecture of Xen and formation of industry partnerships that led to the emergence of Xen as the leading open source virtualization technology. Ian was a member of faculty at the University of Cambridge Computer Laboratory, where he led the Systems Research Group for over 7 years. He was a founder of XenSource, and is now VP for Advanced Products at Citrix.

E-mail : ian.pratt@xen.org

Stephen Spector brings more than 15 years' experience in software engineering, product marketing, and developer and alliance marketing programs to the Xen.org community as the current community program manager. He has spent more than 10 years at Citrix, founding the Citrix Developer Network and supporting the release of the first Windows CE 1.0 client, as well as working on various marketing and alliance programs.

E-mail : stephen.spector@xen.org