Virtualization

Xen

What is Xen?

- Xen is a is a virtual machine monitor (VMM) that allows multiple guest operating systems to run on the same computer hardware

- Allows for the increase of server utilization and consolidation.
  - **More processes can be run on less hardware**

Design Principles

- Support for unmodified application binaries
  - **Allows user-level applications to be run without being modified**
- Support for multiple full multi-application OS
  - **Each guest can contain a complex sever configuration**
- Use paravirtualization
  - **Needed to obtain performance and resource isolation on architectures that do not facilitate virtualization (ie x86)**
- Hide effects of virtualization from guests
  - **Each guest OS does not know about other guests**

Paravirtualization

- Interface presented to a guest OS is not identical to underlying hardware
  - **Allows VMM to correctly handle instructions in architectures that do not support full virtualization (such as x86)**
- In some cases it is beneficial for the host to see real and virtual resources
  - **Providing real and virtual time can allow a guest OS to handle time sensitive tasks more efficiently**
- Drawback: guest OS must be modified

Structure of a Xen system

- The Xen *hypervisor* provides an abstraction layer that sits between system hardware and one or more guest operating systems
- Each guest OS is executed within its own virtual machine, called a *domain*
  - **Domain0: has special management privileges and is used to create the other domains**
  - **DomainU: contains one guest OS**

Xen Hierarchy

- Levels of a Xen system

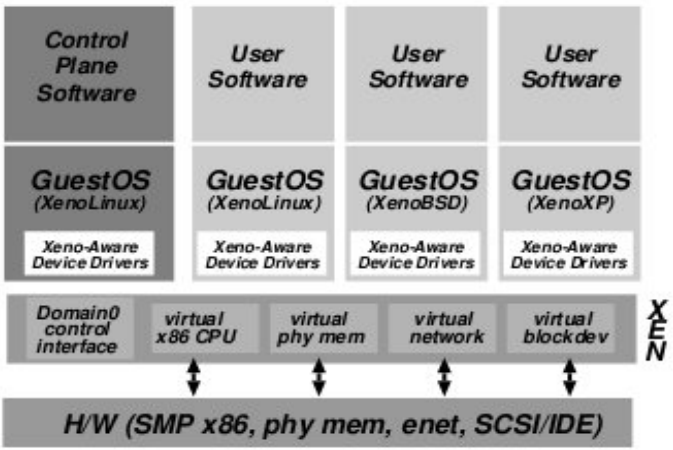


Figure 1: The structure of a machine running the Xen hypervisor, hosting a number of different guest operating systems, including *Domain0* running control software in a XenoLinux environment.

The Virtual Machine Interface: Memory

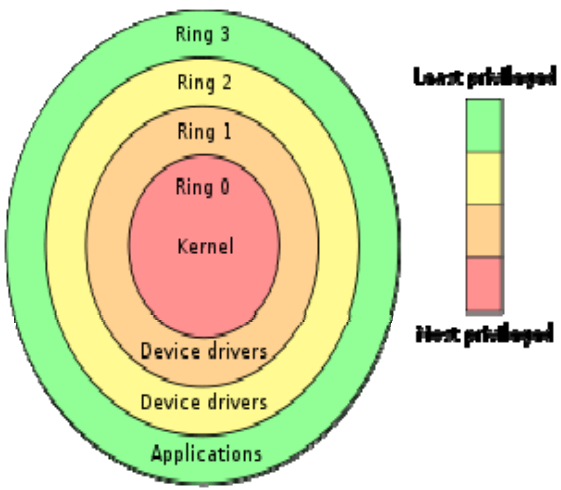
- New page tables are allocated from a guest OS memory reservation and registers it with Xen
- All subsequent writes to the page table are validated by Xen
  - **This ensures that a guest OS only writes to tables it owns, and is isolated from other guests**

The Virtual Machine Interface: CPU

- An operating system is typically the most privileged entity of a system
- With Xen, the hypervisor sits between a guest OS and the CPU
- The hypervisor is the most privileged entity in a Xen system
- Xen uses protection rings to allow the hypervisor to be more privileged than a guest OS

The Virtual Machine Interface: CPU

- x86 privilege rings



- Typical:
  - **OS runs in ring 0**
  - **Applications run in ring 3**
  - **Rings 1 and 2 unused**
- Xen:
  - **Hypervisor runs in ring 0**
  - **Guest OS runs in ring 1**

The Virtual Machine Interface: I/0

- Xen provides simple device abstractions
  - **This helps realize goal of protection and isolation**
- Data sent to and from each domain through the hypervisor
- I/O descriptor rings are used for asynchronous data transfer

I/O Descriptor Rings

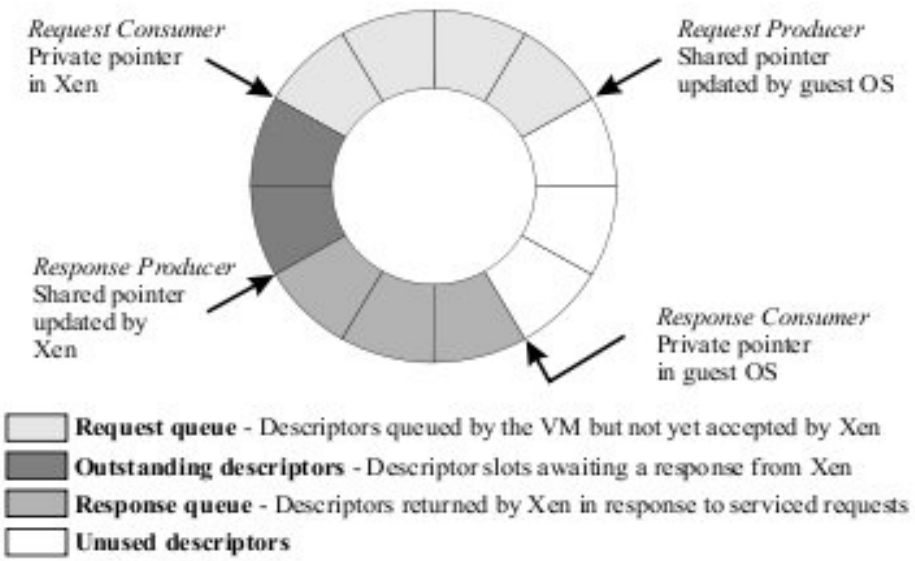


Figure 2: The structure of asynchronous I/O rings, which are used for data transfer between Xen and guest OSes.

Performance

- In summary, Xen performs well
  - **Multipile domains can be hosted without any noticeable loss of performance by end user**
- *Xen and the Art of Virtualization* identifies a scalability goal of 100 domains on modern (c. 2003) server-class hardware
  - **Tests demonstrate that 128 domains can be run with only 7.5% loss of throughput relative to stand-alone Linux**

Performance

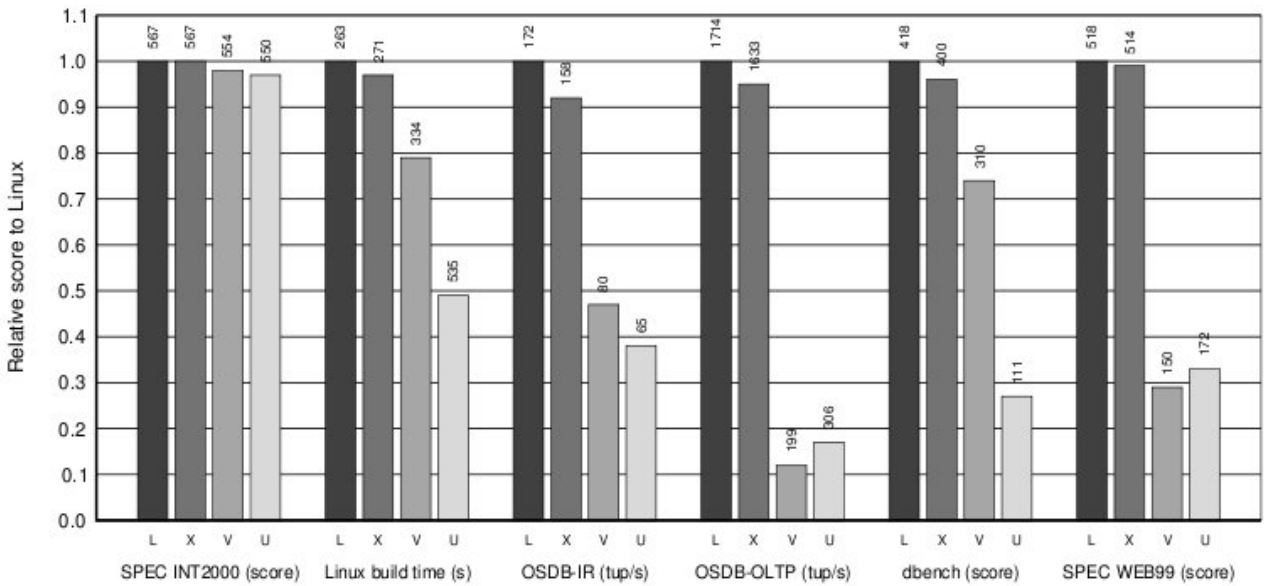


Figure 3: Relative performance of native Linux (L), XenoLinux (X), VMware workstation 3.2 (V) and User-Mode Linux (U).

Performance

Running multiple web servers:

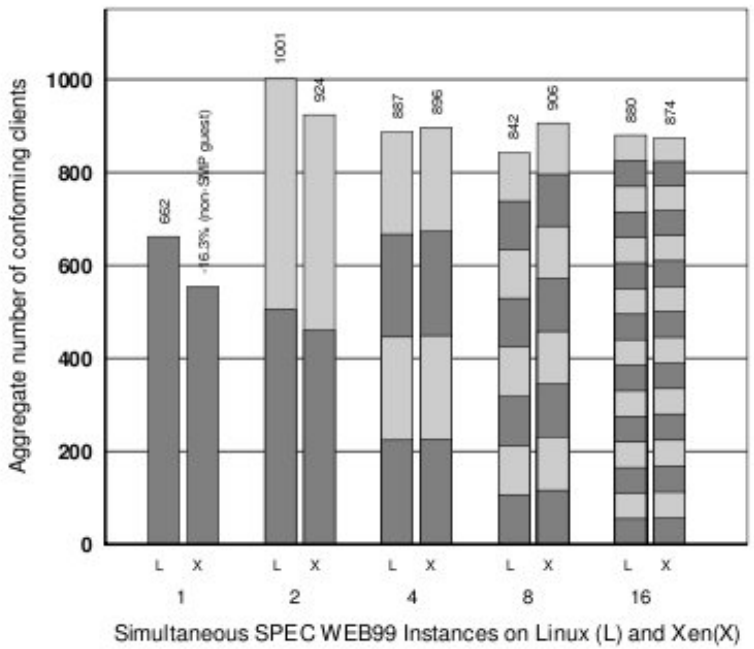


Figure 4: SPEC WEB99 for 1, 2, 4, 8 and 16 concurrent Apache servers: higher values are better.

Demonstration

- Xen is an open source product that is easy to install and configure on Linux
  - **Requires a modified Linux kernel**
- Xen provides a live CD which can be used to try the hypervisor without installing any software

Questions?