Open Computing @ IBM

# Virtualization and Xen

**Jim Elliott**

**Advocate – Infrastructure Solutions**
**Manager – System z Operating Systems**

**IBM Canada Ltd.**

Innovation that matters

Open Computing @ IBM

# An Introduction to Virtualization

*What is it, and why is it important to you?*

Innovation that matters

# Today's business challenges

- **Control and reduce corporate operating costs**
- **Invest in innovation that delivers business value in shorter cycles**
- **Maximize the return on past investments including IT**
- **Improve management of corporate risks**

## How to do more with less

# Technology implications

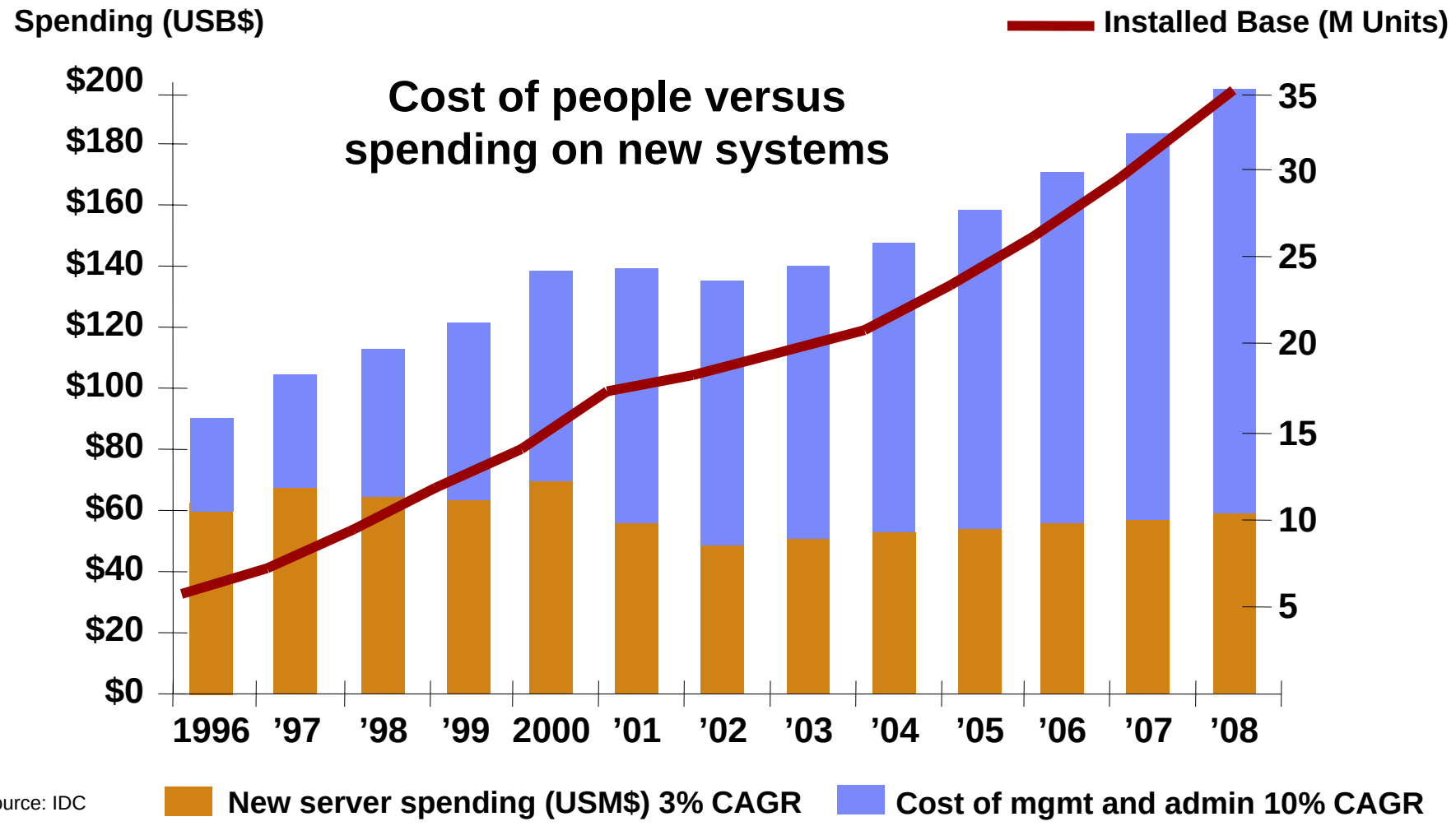
- **CEO agenda**
  - Control and reduce corporate costs
  - Innovation now
  - Maximize return on past investment
  - Manage corporate risks

- **CIO agenda**
  - Control and reduce IT costs
  - Leverage new technologies which improve time to market business value
  - Maximize return on past infrastructure investments
  - Improve IT resilience and security

# The simplification imperative

- ***"Our infrastructure grew organically over the years without a master plan – things got bolted on – and now we are stuck where we are."***
  - CIO from a Fortune 1000 company
- ***"Data centers have become so fragile that administrators are fearful to touch the existing infrastructure, since any changes may set off a series of events that can bring a company to its knees. Consequently, many enterprises are restricted in deploying innovative applications that could potentially create competitive advantage."***
  - The Yankee Group January 2005

# Complexity is driving costs




Source: IDC

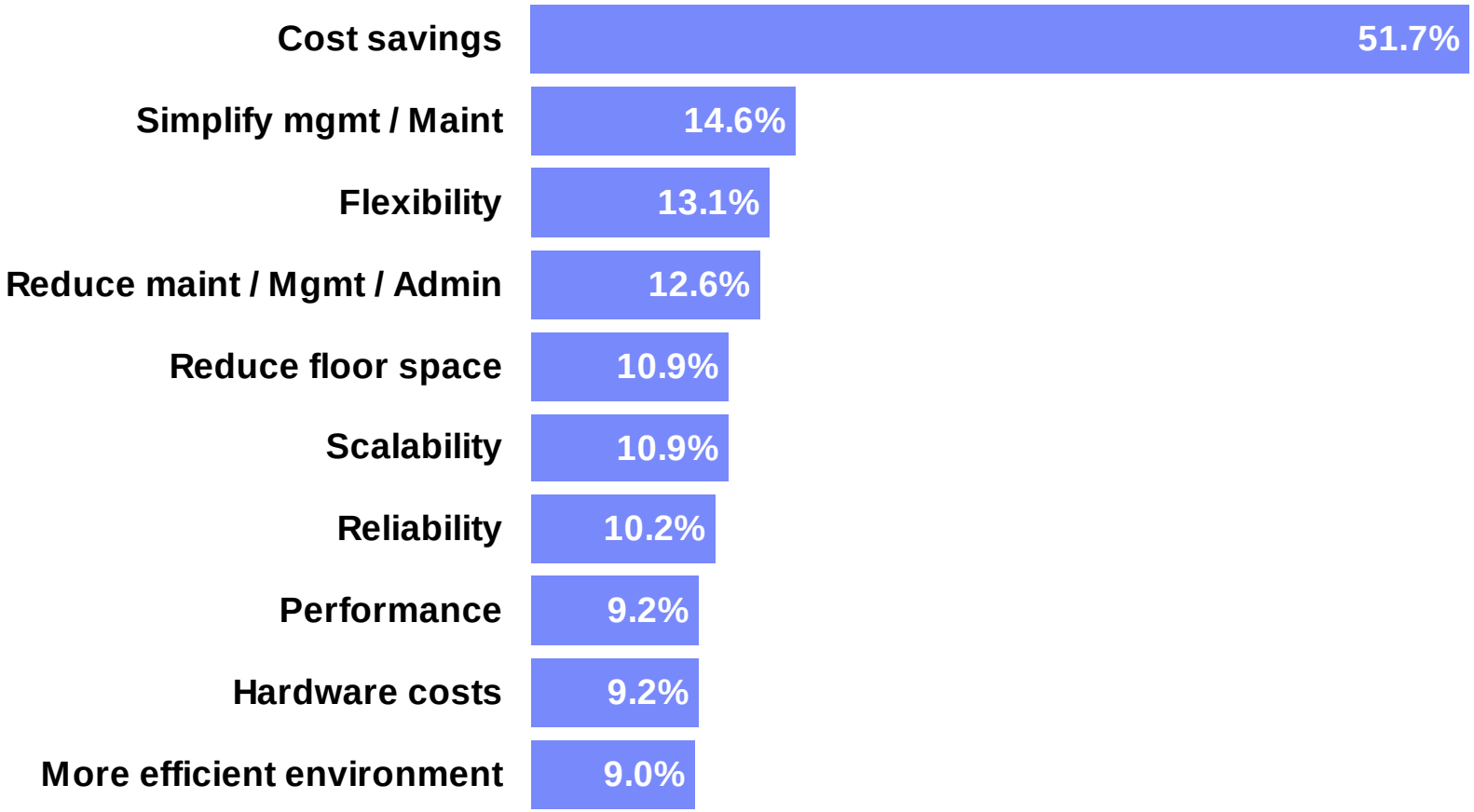
# Virtualization is a fundamental imperative

- ***"Virtualization is the process of presenting computing resources in ways that users and applications can easily get value out of them, rather than presenting them in a way dictated by their implementation, geographic location, or physical packaging. In other words, it provides a logical rather than physical view of data, computing power, storage capacity, and other resources."***
  - Jonathan Eunice, Illuminata Inc.

# Value of a virtualized infrastructure

- **Increase utilization**
  - Most practical choice to achieve full consolidation
  - Capability to pool resources to service a workload
  - Can improve availability and reliability (LPAR, SAN, Clustering)
- **Improve productivity**
  - Creates virtualized infrastructure for test and development
  - Improves rapid deployment and scaling of workloads
  - Use common tools across many systems, simplified resource management
- **Link infrastructure performance to business goals**
  - Use policy to adjust resources based on requirements of the business
  - Analyze application performance based on business policy
  - Improve business resilience

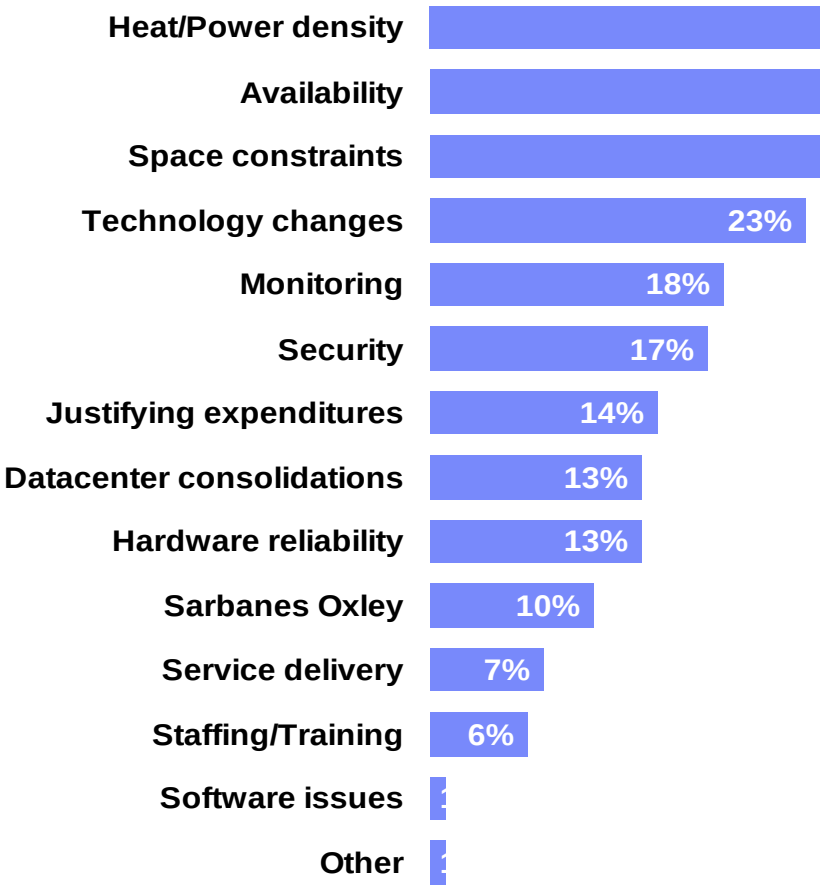
# Top virtualization motivators

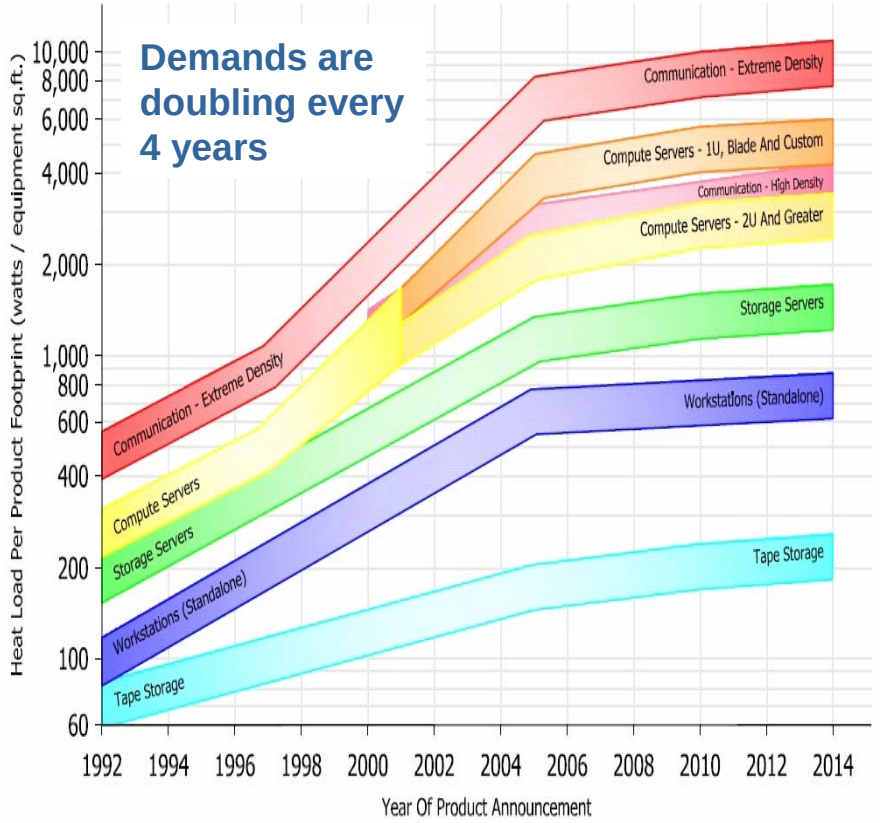



IDC Server Virtualization Multi-Client Study, 2005 “Question: What are the top 3 reasons that your organization virtualized servers?” Multiple responses allowed, therefore percentages may exceed 100% of the sample. Sample size = 420

# Power and cooling

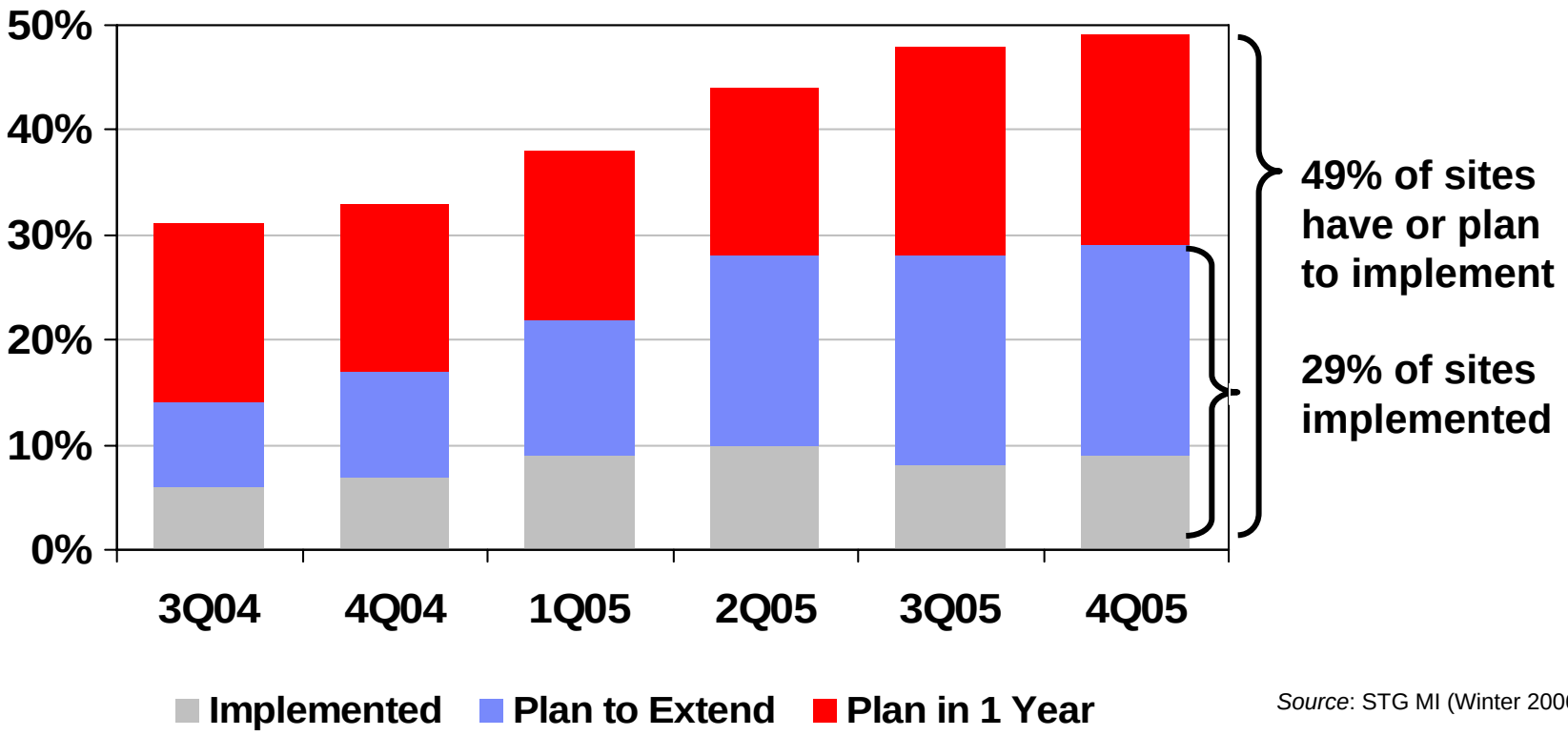
## *Installation heat and power levels are major concern*




Sources:
http://www.liebert.ws/liebertmadara/liebertmadara_files/Default.htm#nopreload=1
Datacom Equipment Power Trends and Cooling Applications, ASHRAE 2005
* http://www.ibm.com/press/us/en/pressrelease/7775.wss

# Virtualization momentum




*Source*: STG MI (Winter 2006)

# What is virtualization?

- **Logical representation of resources not constrained by physical limitations**
  - Create many virtual resources within single physical device
  - Reach beyond the box – see and manage many virtual resources as one
  - Dynamically change and adjust across the infrastructure

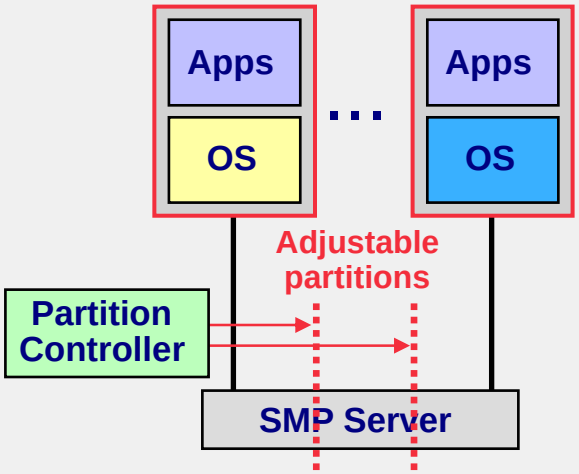
# What is partitioning?

- **Partitioning is the division of a single server's resources* into multiple, independent, isolated systems capable of running their own operating system**
- **Three types of partitioning:**
  - ***Hardware*** – resources are allocated to partitions on a one-to-one basis with the underlying physical hardware (no sharing among partitions)
  - ***Logical*** – resources are managed by hardware firmware and allocated to partitions with a finer granularity than hardware partitioning (resource sharing among partitions)
  - ***Software*** – resources are managed by a software layer, aggregated into shared resource pools, and apportioned to users as virtual system resources, separating the presentation of the resources from the actual physical entities

* Resources include: processors, memory, I/O adapters and devices, networking interfaces, co-processors

# Server Virtualization Approaches

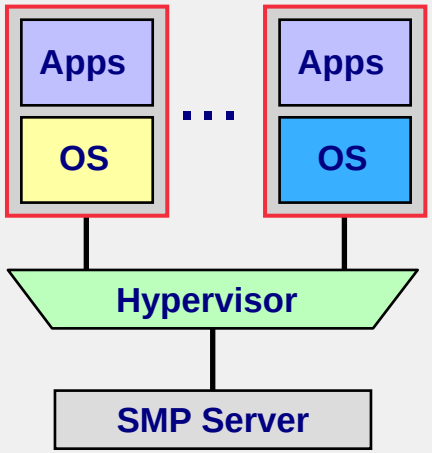
## Hardware Partitioning




**Physical partitioning**
- Sun Domains, HP nPartitions

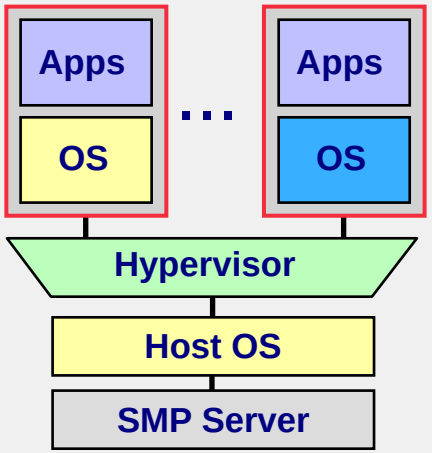
**Logical partitioning**
- System p LPAR, HP vPartitions

## Hypervisor: Type 1




**Hypervisor software/firmware runs directly on server**
- System z PR/SM and z/VM
- POWER Hypervisor
- HP Integrity VM
- VMware ESX Server

## Hypervisor: Type 2




**Hypervisor software runs on a host operating system**
- VMware GSX
- Xen Hypervisor
- Microsoft Virtual Server
- User Mode Linux

- Hardware partitioning subdivides a server into fractions, each of which can run an OS
- Hypervisors use a thin layer of code to achieve fine-grained, dynamic resource sharing
- Type 1 hypervisors with high efficiency and availability will become dominant for servers
- Type 2 hypervisors will be mainly for clients where host OS integration is desirable

# "Trapping and mapping" method
## *Hypervisor technologies*

- **Guest operating system runs in user mode**
- **Hypervisor runs in privileged mode**
- **Privileged instructions issued by guest operating system(s) are trapped by hypervisor**
- **IA-32 (Intel) complications:**
  - Some instructions behave differently in privileged and user modes
    - For example, "POPF" treatment of the interrupt enable flag
  - User mode instructions that access privileged resources/state cannot be trapped; instruction must be changed to something that can be trapped
- **Some guest kernel binary translation may be required**
- **Originally used by mainframes in 1960s and 1970s (VM/370)**
- **Used today by VMware, Microsoft VS, …**

# Hypervisor call method ("paravirtualization")

*Hypervisor technologies*

- **Guest operating system runs in privileged mode**
- **Hypervisor runs in super-privileged mode**
- **Guest operating system kernel (e.g., AIX, i5/OS, Linux) is modified to do hypervisor calls for I/O, memory management, yield rest of time slice, etc.**
- **Memory mapping architecture is used to isolate guests from each other and to protect the hypervisor**
- **Used by XEN and POWER5 today**

# Direct hardware support method

*Hypervisor technologies*

- **Guest operating system runs in privileged mode**
- **Guest operating system can be run unmodified, but can issue some hypervisor calls to improve performance or capability**
  - I/O (z/VM) or yield time slice (PR/SM and z/VM)
- **Extensive hardware assists for hypervisor (virtual processor dispatching, I/O pass-through, memory partitioning, etc.)**
- **Used by System z (PR/SM and z/VM) today**
  - Used by VMware and Xen with Intel VT and AMD Pacifica in the future

# Intel and AMD software solutions

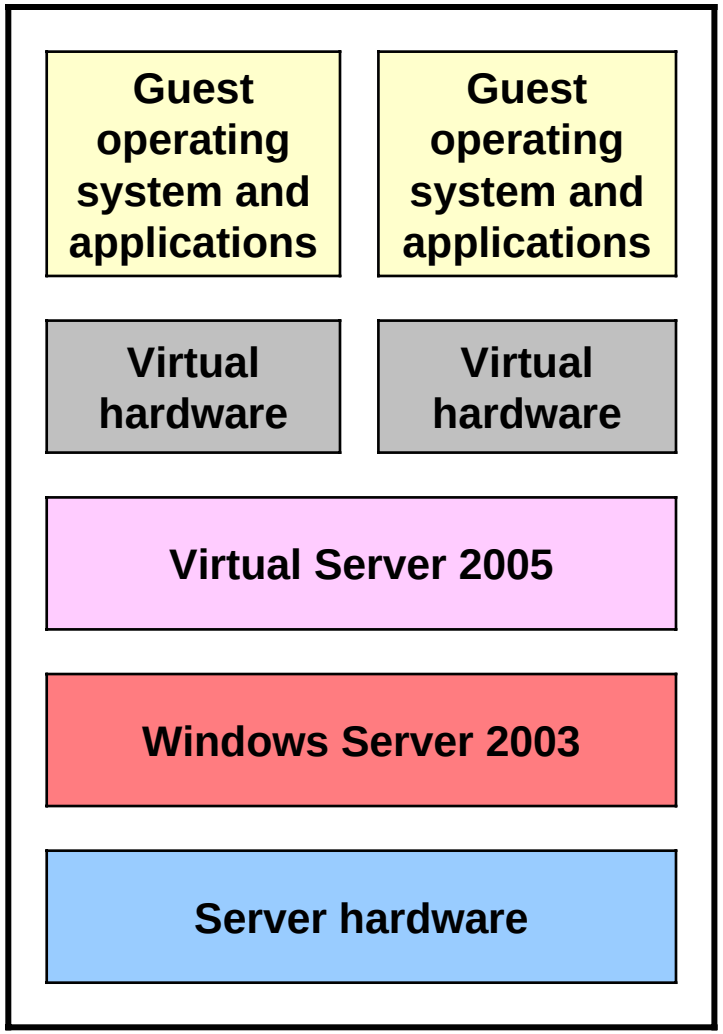
*Platform overview of virtualization technology*

- **Microsoft Virtual Server 2005**
- **Xen**
- **VMware ESX Server**
- **Others (e.g., SWsoft Virtuozzo, QEMU, Virtual Iron, …)**
- **Hardware architecture enhancements are arriving now**
  - Intel: Virtualization Technology (Silverdale/Vanderpool) – *"With enhancements to Intel's various platforms, Intel Virtualization Technology can improve the robustness and performance of today's software-only solutions."*
  - AMD: Pacifica – *"Pacifica is designed to provide foundation technologies to deliver IT resource utilization advantages through server consolidation, legacy migration and increased security*."

# Microsoft Virtual Server 2005

- **Allows multiple operating systems to run simultaneously on the same processor**
- **Each independent virtual machine functions as a self-contained computer**
- **Virtual machines are encapsulated in portable Virtual Hard Disks (VHDs)**
  - Up to 32 VHDs can connect to a single virtual machine
  - VHDs can expand as data is added and “differenced”
- **Virtual networking options are available**
- **Requires a hosting operating system (Windows Server 2003)**
  - No charge for MS VS, but you need a WS2003 license for every four guests
- **Well suited for hosting unsupported Windows NT environments**
- **Native Windows Server 2003 environment still recommended for many workload deployments**

# Microsoft Virtual Server 2005 – architecture




- Windows NT 4 Server, Windows 2000 Server, Windows Server 2003
- 1 CPU per virtual machine
- Up to 3.6 GB of memory per virtual machine
- Multithreaded virtual machine monitor (VMM)
- provides isolation
- Windows Server 2003 32-bit supported host
- Broad device compatibility
- Optimized for 2 to 8 way servers
- Scales up to 32 CPUs and up to 64 GB

# Xen 3.0

- **Open Source virtualization software solution based on Linux**
- **Uses paravirtualization to abstract CPU, memory, and I/O resources**
- **Guest operating systems are responsible for allocating and managing page tables**
- **Management and control software runs in Domain 0**
- **Currently does not support Windows (kernel modification required)**
- **Intel Virtualization Technology (IVT) will enable hosting of unmodified guest operating systems**
- **IBM is actively contributing to the Xen open source project**

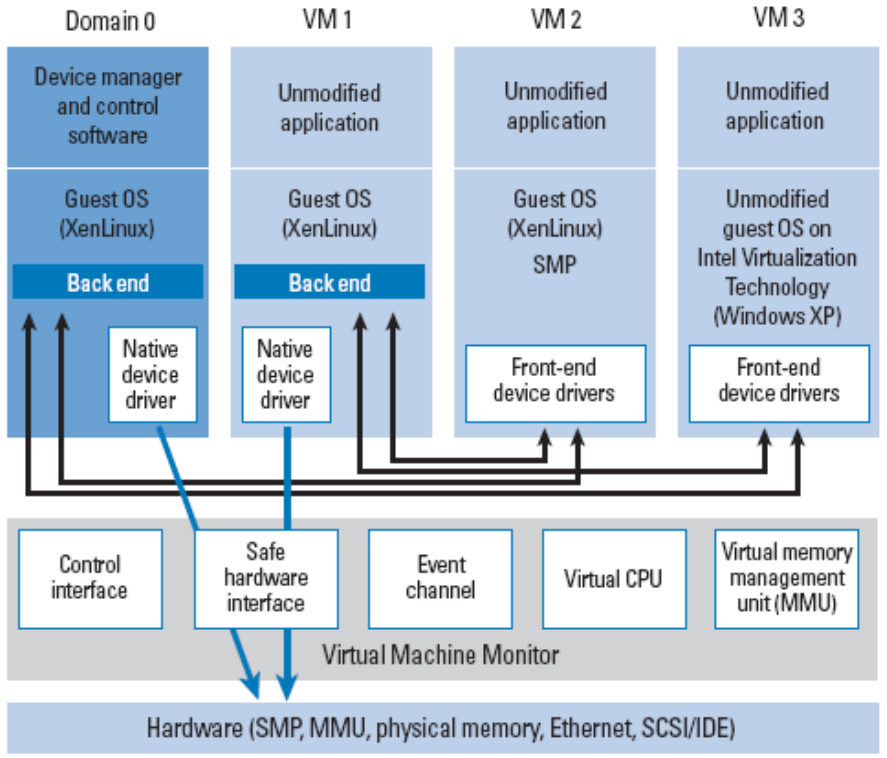
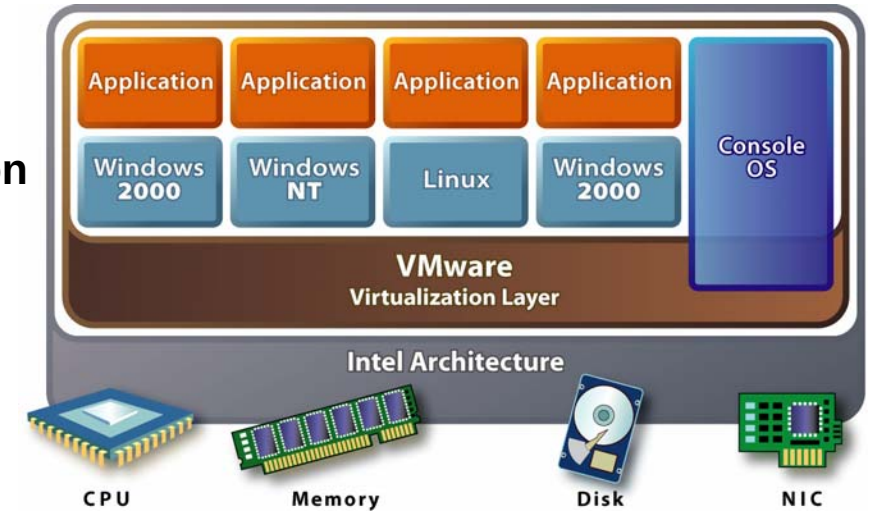



Figure 1. Xen 3.0 architecture: Hosting four VMs

# z86 partitioning with VMware Infrastructure 3

- **VMware Infrastructure 3 runs directly on the hardware**
  - No hosting operating system required (as is the case with VMware Server)
- **Supports blade servers as well as standalone Intel servers**
- **Creates multiple virtual machines on a single Intel system**
  - Supports a maximum of 80 virtual machines per VMware image (depending on system resources)
- **Manages resource allocations**
  - Strong fault and security isolation (uses CPU hardware protection)
  - Virtual networking support available (MAC or IP addressing)
  - Direct I/O passthrough
- **Shared data cluster-ready**
- **Scalable to large virtual machines and high (Intel) performance**
- **VMware File System allows multiple virtual disks to be stored on a single LUN or partition**
- **Virtual machines are encapsulated**
- **Add-on products available**
  - VMware Virtual SMP allows virtual machines to be configured with up to 4 CPUs
  - VMware VirtualCenter

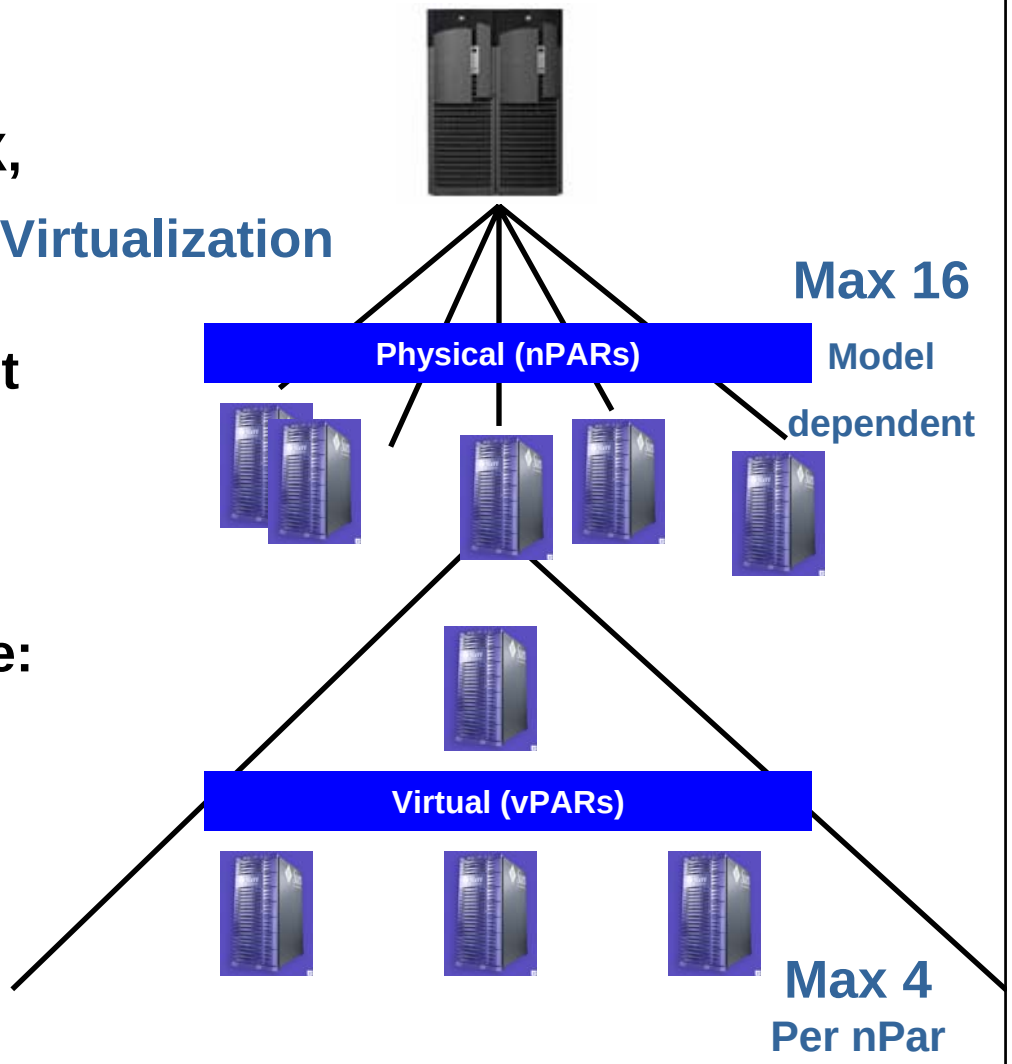
# Non-x86 software solutions

*Platform overview of virtualization technology*

- **HP**
  - Node Partitions (nPARs) – hardware partitioning solution
  - Virtual Partitions (vPARs) – software solution that only supports HP/UX
  - Virtual Server Environment (VSE) – for HP Integrity and HP 9000 servers
- **Sun**
  - Dynamic Domains – hardware partitioning solution
  - Solaris Containers – all containers (“zones”) share the same copy of Solaris
- **IBM POWER5**
  - LPAR-like solution with I/O and network sharing accomplished via hosted partitions
  - Supports AIX, Linux, and i5/OS
- **IBM System z**
  - LPAR and hypervisor solutions

# HP virtualization / partitioning capabilities

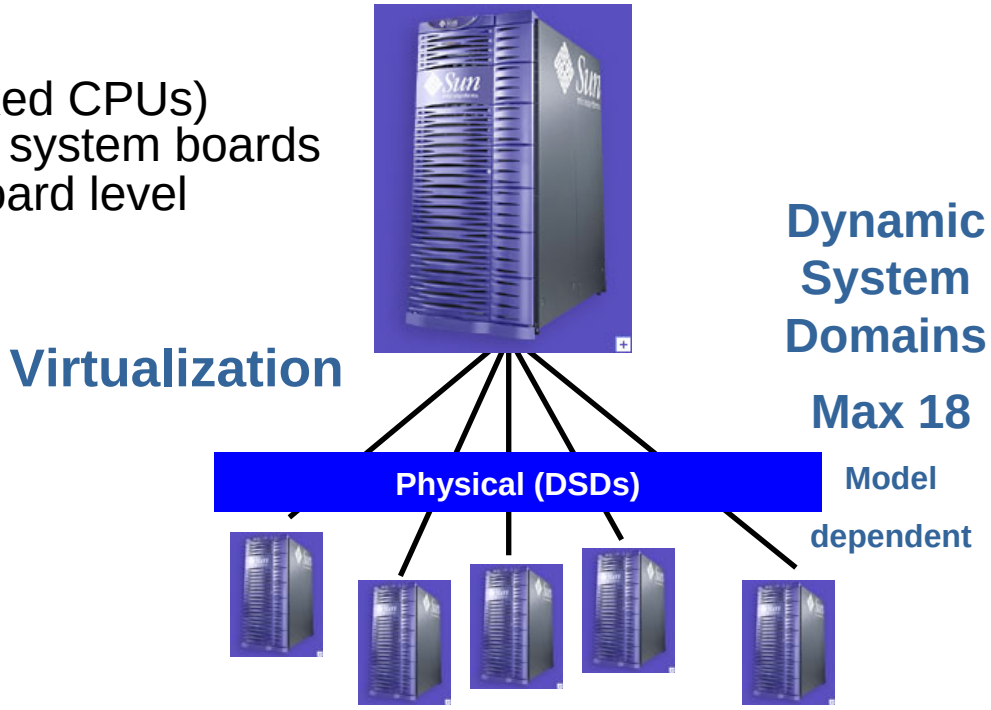
- **nPars: PA-RISC servers with HP-UX, IPF servers with HP-UX, Linux, Windows and in the future OpenVMS**
- **vPars: PA-RISC only, not currently supported on IPF (but imminent)**
- **Resource partitions: HP-UX only (similar to Solaris "containers")**
- **Partition management software:**
  - Parmgr: basic nPar management for PA-RISC and IPF servers
  - vParmgr: basic vPar management within nPars for PA-RISC servers




All commentary on this page is based upon IBM's view.

# Sun virtualization / partitioning capabilities

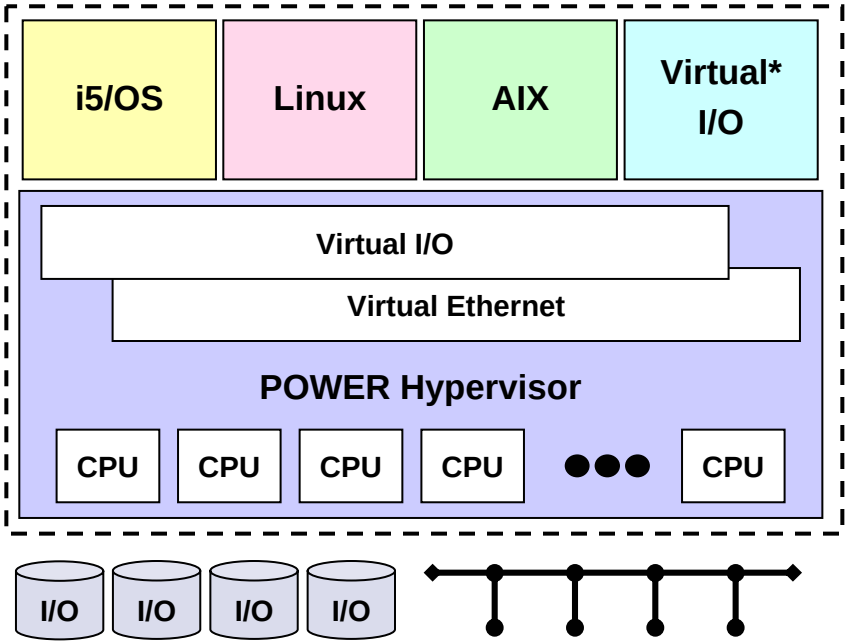
- **Solaris containers**
  - Solaris containers are NOT a virtualization capability.
  - Enables multiple applications to run on a single OS instance, claiming:
    - Application independence and isolation
    - Managed resources via Solaris resource manager
    - High server utilization
- **Sun domains**
  - Allows blended technology (mixed CPUs) and electrical isolation between system boards
  - Partitioning granularity at the board level
    - CPUs and memory move together
    - I/O moved in four slot increments
    - Sub processor allocation not possible
    - Partitioning is not available on V480, V490, V880, V890, V1280 or V2900




All commentary on this page is based upon IBM's view.

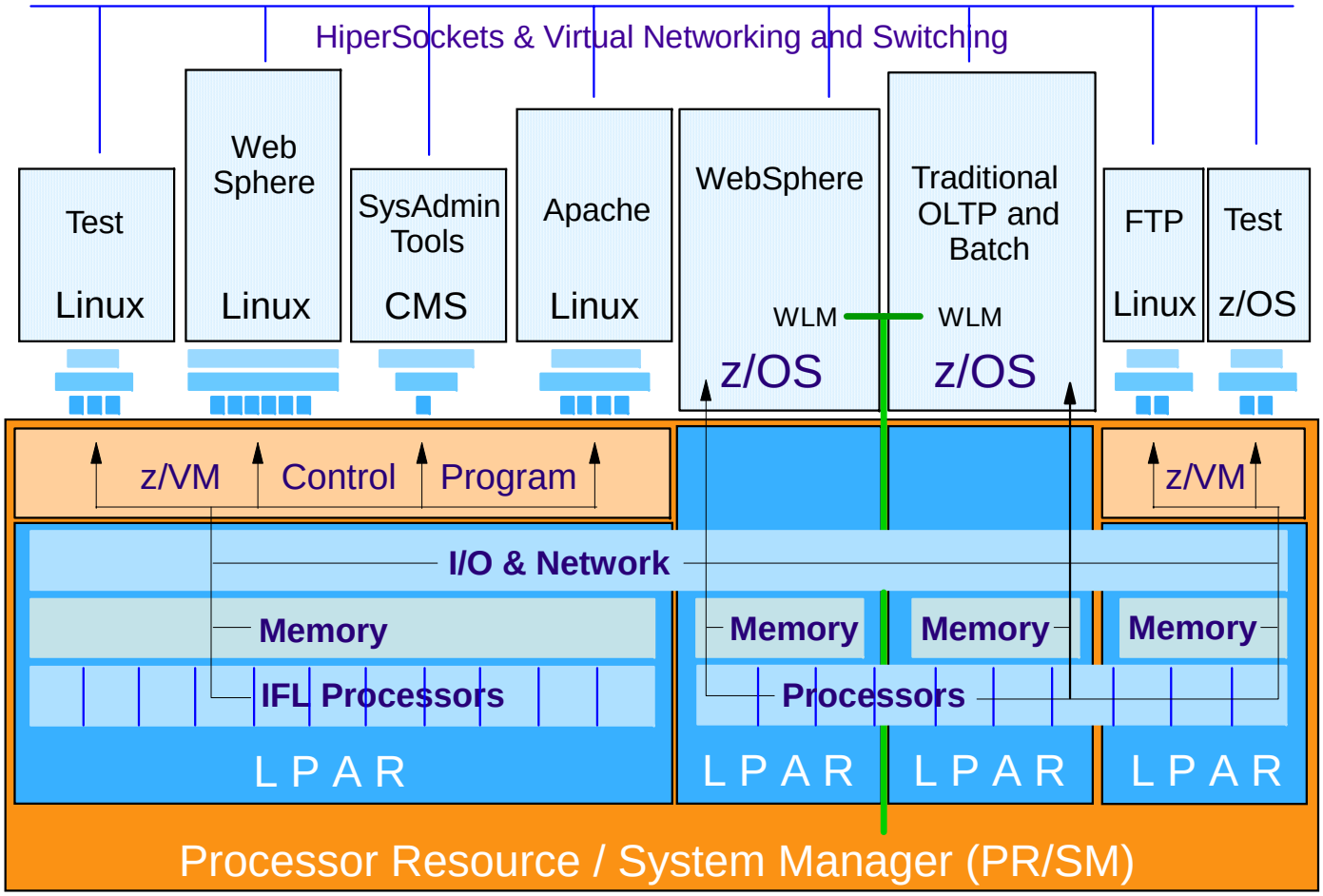
# IBM Advanced Power Virtualization (APV)

- **Dynamic logical partitioning**
  - Minimum 1/10 of a CPU
  - Dedicate or share processors
  - Dedicate or share I/O adapters
- **Virtual I/O***
  - Virtual SCSI
  - Shared Ethernet
  - Clients:  AIX 5.3, Linux
- **Virtual Ethernet**
  - In memory cross partition network
  - Clients:  AIX 5.3, Linux, i5/OS
- **Dynamic operations**
  - AIX, i5/OS, SLES 9
  - Can be automated
- **Partition security**
  - POWER4: EAL4+
  - Statement of direction for POWER5



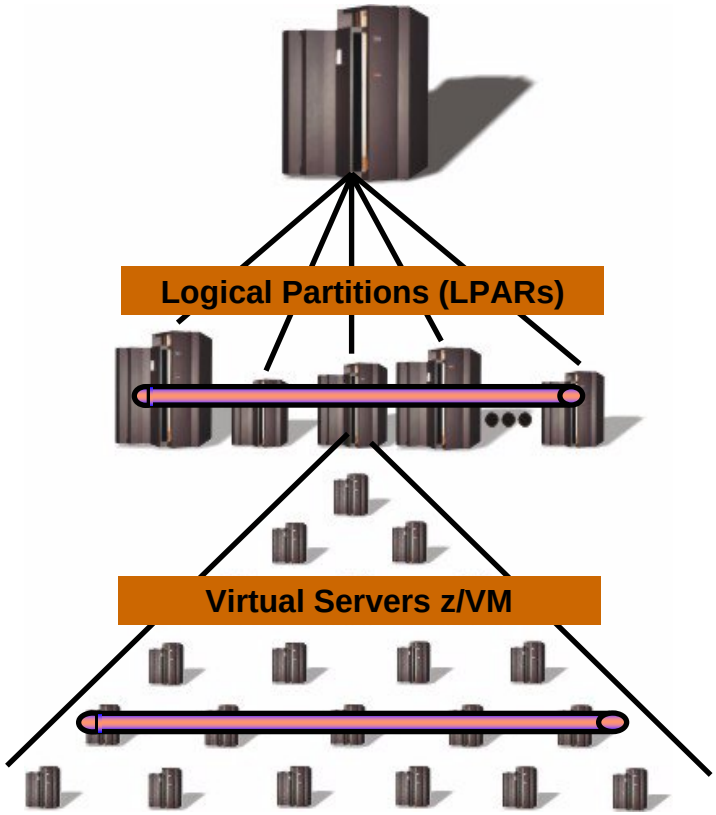

* Provided by Virtual I/O Server on System p and i5/OS hosting partition on System i

# System z virtualization / partitioning




- **System z provides two levels of partitioning**
  - PR/SM enables scalable virtual server hosting for LPAR environments
  - z/VM provides hypervisor function for highly scalable virtualization

# z/VM – Unlimited virtualization




**z/VM 5.2** – 64-bit support – real and virtual

- **Mature technology – z/VM introduced in 1967**
- **Software Hypervisor integrated in hardware**
  - Sharing of CPU, memory and I/O resources
  - Virtual network – virtual switches/routers
  - Virtual I/O (mini-disks, virtual cache, …)
  - Virtual appliances (SNA/NCP, etc.)
- **Easy management**
  - Rapid install of new servers – cloning or IBM Director task z/VM Center
  - Self-optimizing workload management
  - Excellent automation and system management facilities
  - Flexible solution for test and development systems

# Key points to remember about virtualization

- **Virtualization is not about higher performance**
  - Applications are not going to run faster on servers engaged in application consolidation. It is very important, though, that applications not run perceptibly slower. Increased efficiency and simplified management is the objective.
- **Higher utilization (virtualization) is directly proportional to available server throughput**
  - All virtualization technologies must work within resource constraints of server infrastructure. (All servers have a finite throughput capability.)
  - Throughput potential can best be identified with “changed data” benchmarks.
- **Virtualization will not overcome a weak server architecture**
  - Virtualization introduces mixed / random / unpredictable behavior. Applications have different instruction and data working sets, putting dynamic pressure on the server interconnect.
  - Customers cannot afford to consolidate applications on low reliability servers!
- **A history with virtualization matters**
  - Virtualization on IBM System z and POWER is an integration of hardware, firmware and software. All other architectures are physical and/or software implementations.
  - Security, isolation, fair-share scheduling are not buzzwords. Common criteria certification.

Open Computing @ IBM

# Xen and the Art of Virtualization

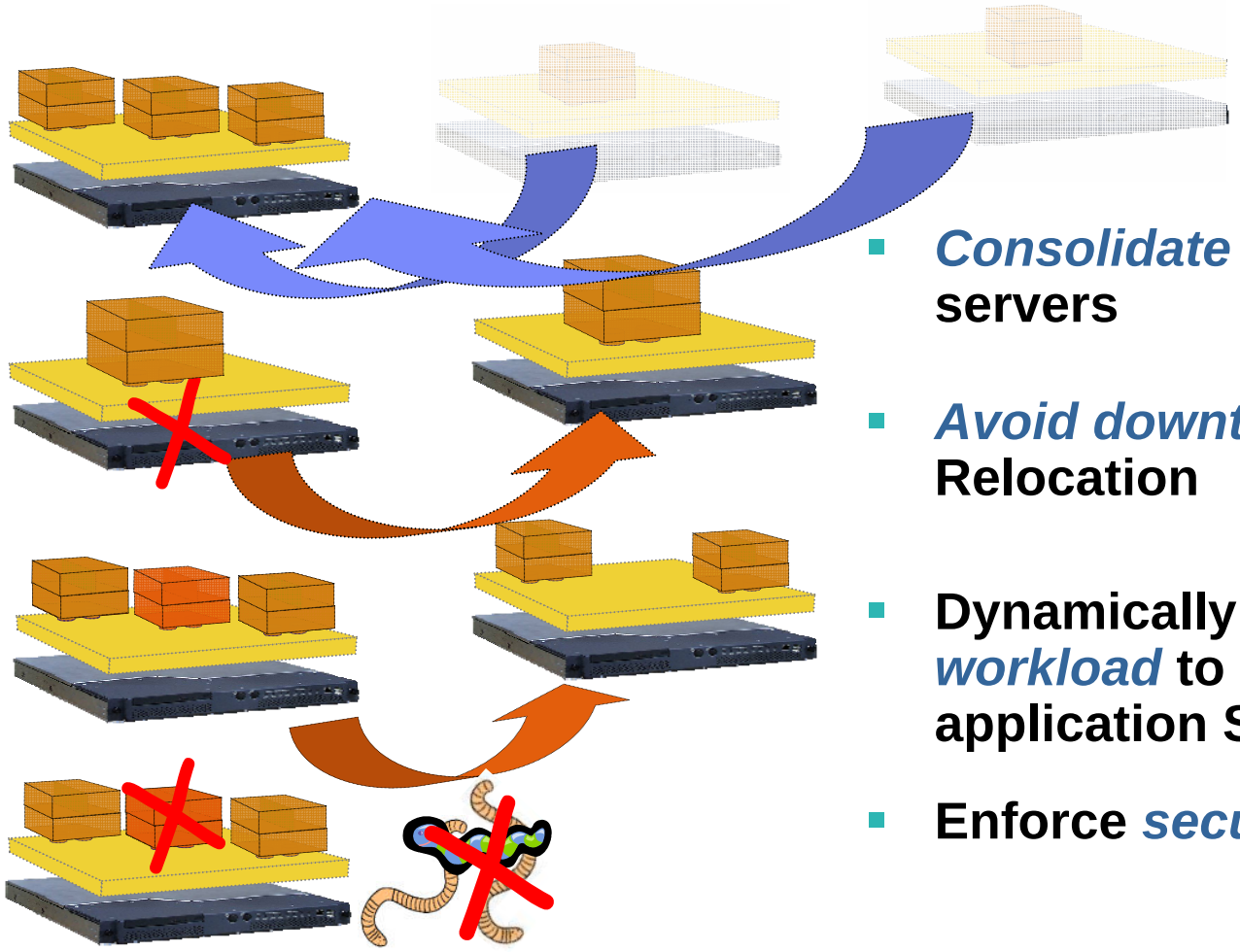
Innovation that matters

# Outline

- **Xen Architecture**
- **New Features in Xen 3.0**
- **VM Relocation**
- **Xen Roadmap**
- **Questions**

- ***Thanks to Ian Pratt of the Xen project and XenSource for permission to use his presentation as a base for this section***

# Virtualization in the Enterprise



- ***Consolidate*** **under-utilized servers**
- ***Avoid downtime*** **with VM Relocation**
- **Dynamically *re-balance workload* to guarantee application SLAs**
- **Enforce *security* policy**

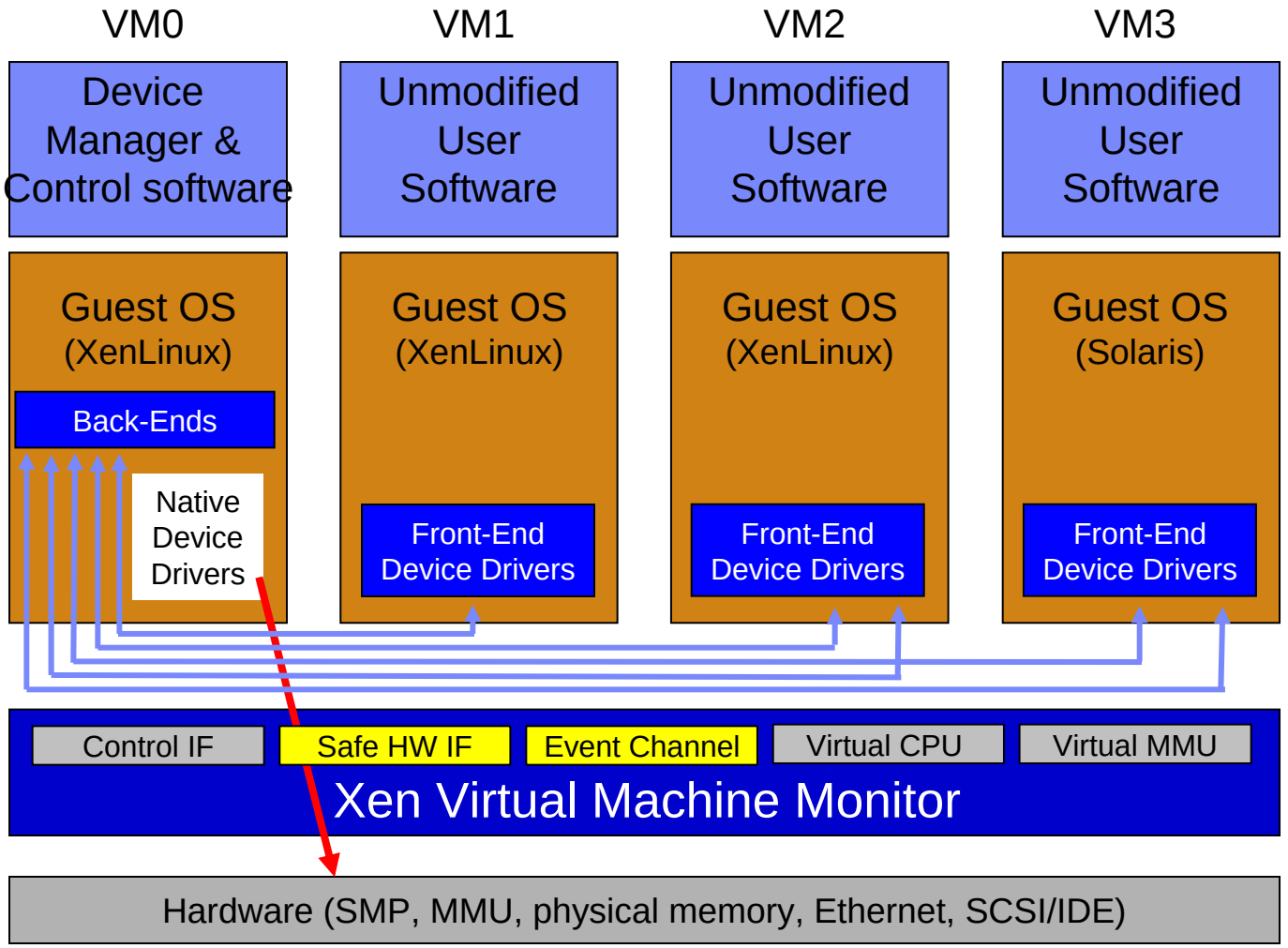
# Xen 2.0 2005-11-05

- **Secure isolation between VMs**
- **Resource control and QoS**
- **Only guest kernel needs to be ported**
  - User-level apps and libraries run unmodified
  - Linux 2.4/2.6, NetBSD, FreeBSD, Plan9, Solaris x86
- **Execution performance close to native**
- **Broad x86 hardware support**
- **Live relocation of VMs between Xen nodes**

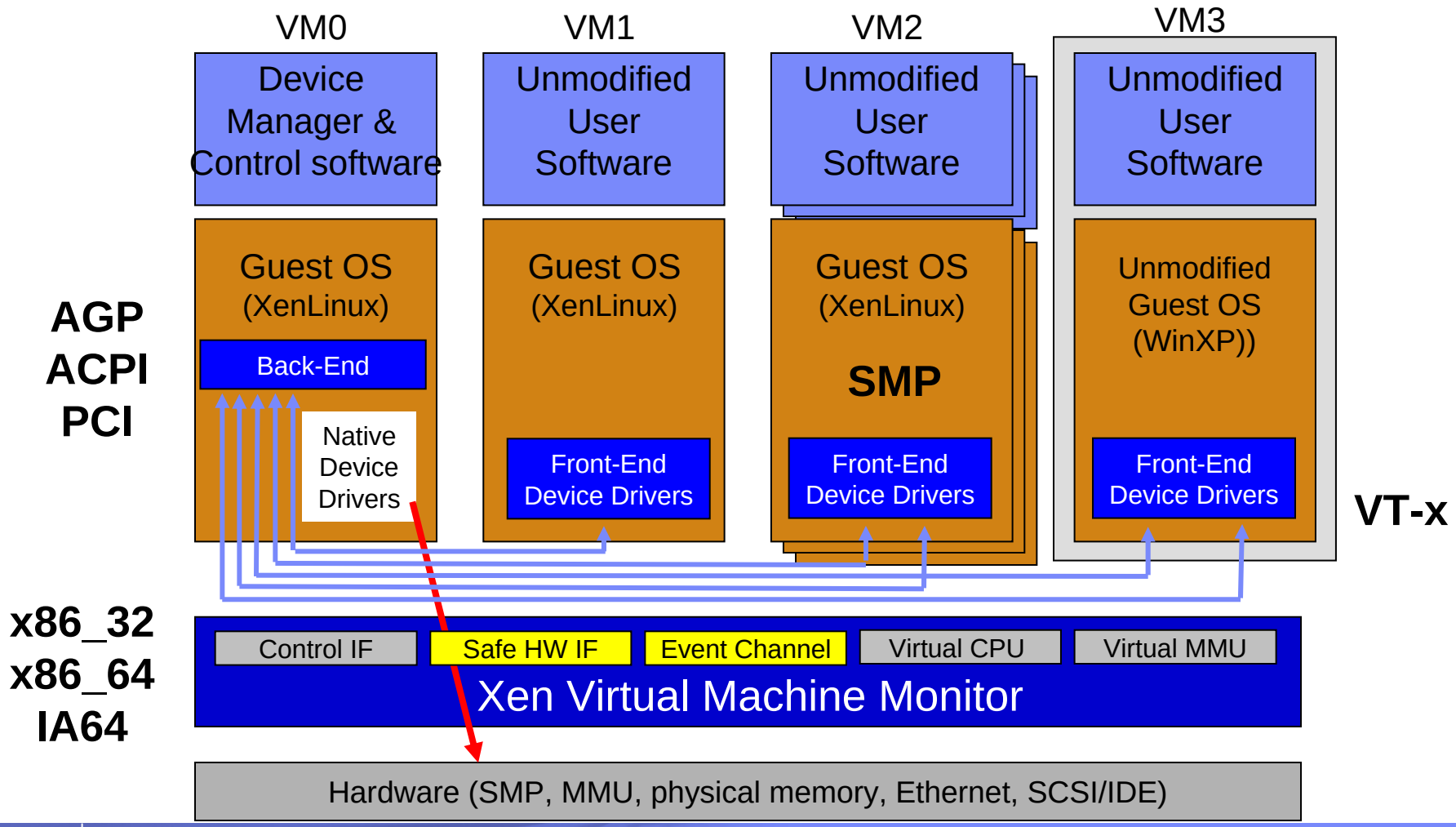
# Para-Virtualization in Xen

- **Xen extensions to x86 arch**
  - Like x86, but Xen invoked for privileged ops
  - Avoids binary rewriting
  - Minimize number of privilege transitions into Xen
  - Modifications relatively simple and self-contained
- **Modify kernel to understand virtualised environment**
  - Wall-clock time vs. virtual processor time
    - Desire both types of alarm timer
  - Expose real resource availability
    - Enables operating system to optimise its own behaviour

# Xen 2.0 Architecture



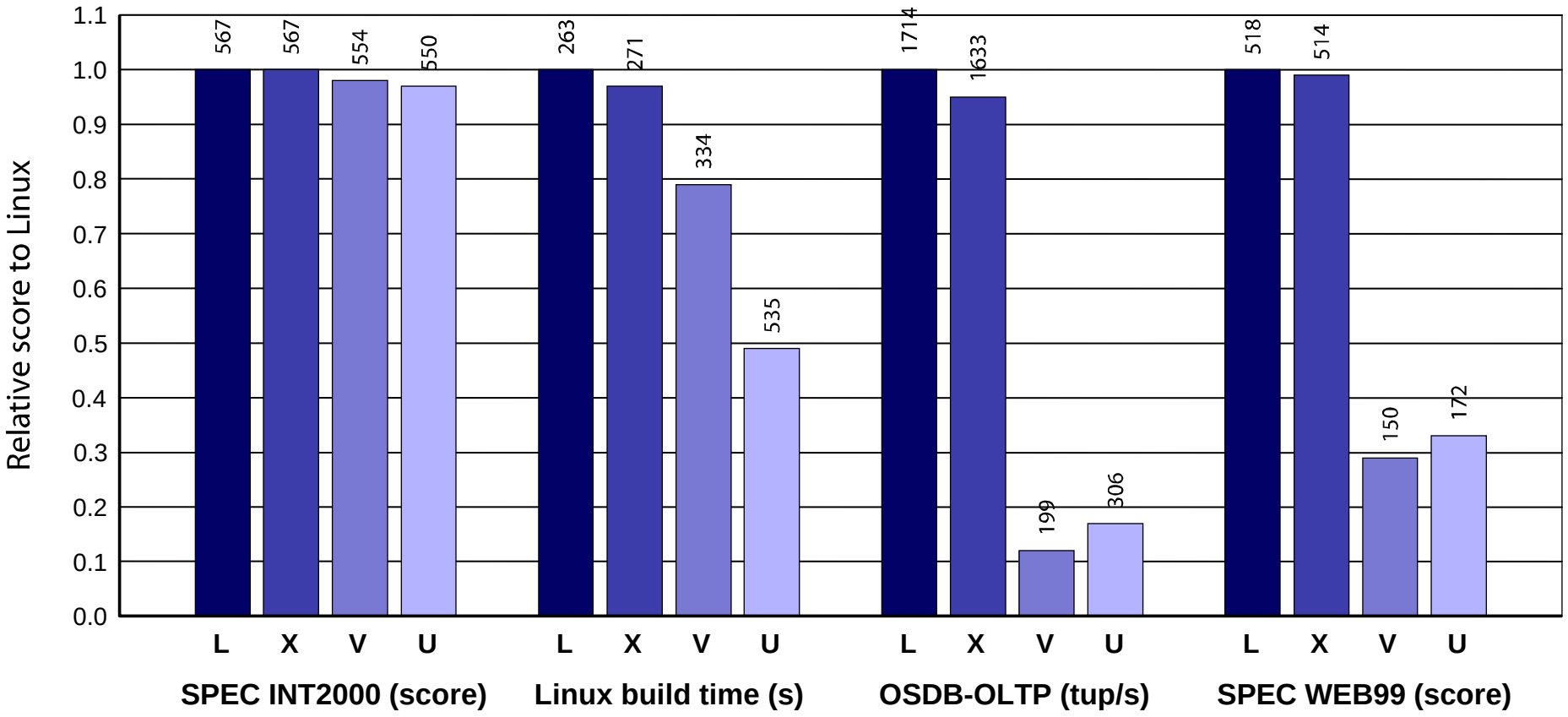
# Xen 3.0 Architecture



# I/O Architecture

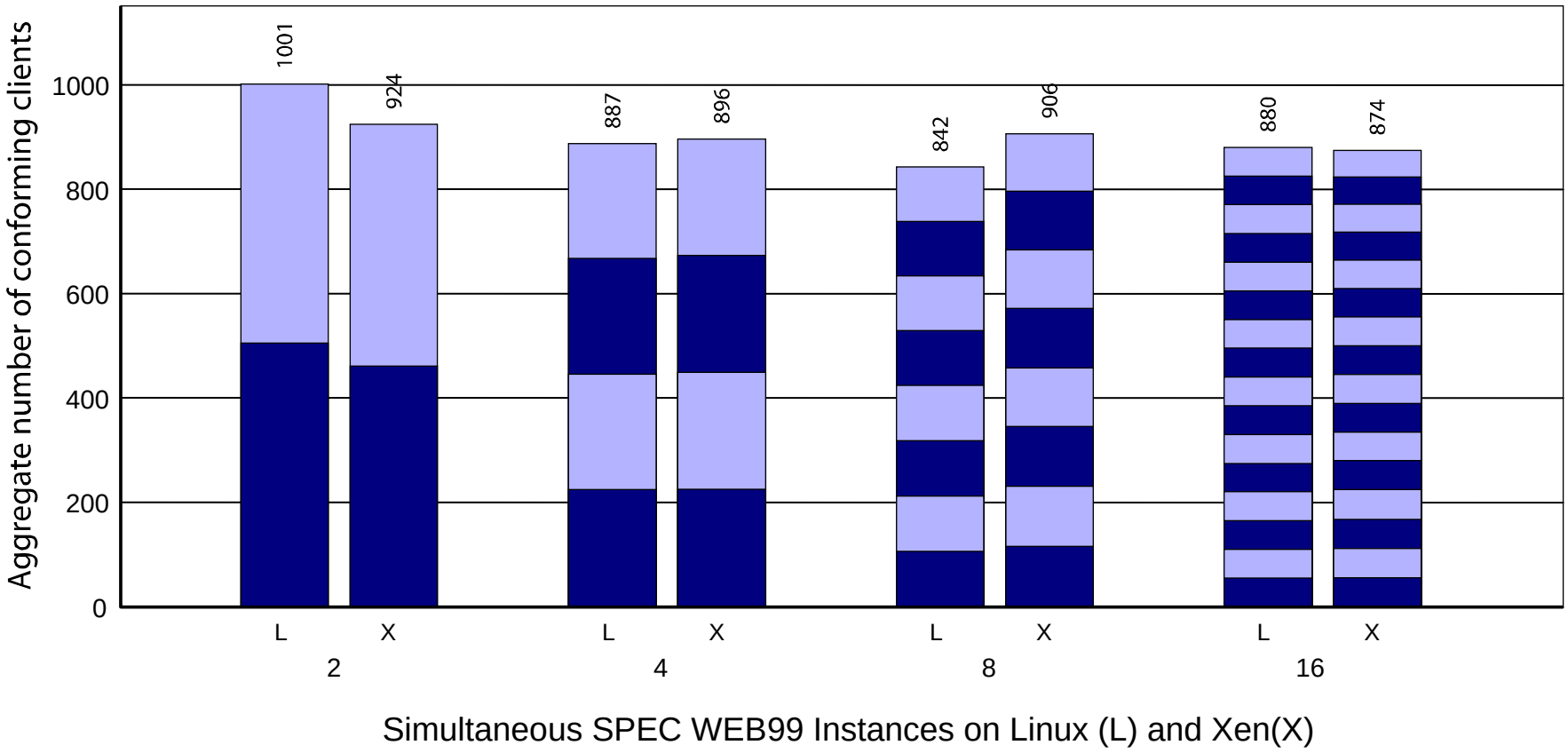
- **Xen IO-Spaces delegate guest operating systems protected access to specified hardware devices**
  - Virtual PCI configuration space
  - Virtual interrupts
  - (Need IOMMU for full DMA protection)
- **Devices are virtualized and exported to other VMs via Device Channels**
  - Safe asynchronous shared memory transport
  - 'Backend' drivers export to 'frontend' drivers
  - Net: use normal bridging, routing, iptables
  - Block: export any blk dev e.g. sda4,loop0,vg3
- **(Infiniband / "Smart NICs" for direct guest IO)**

# System Performance

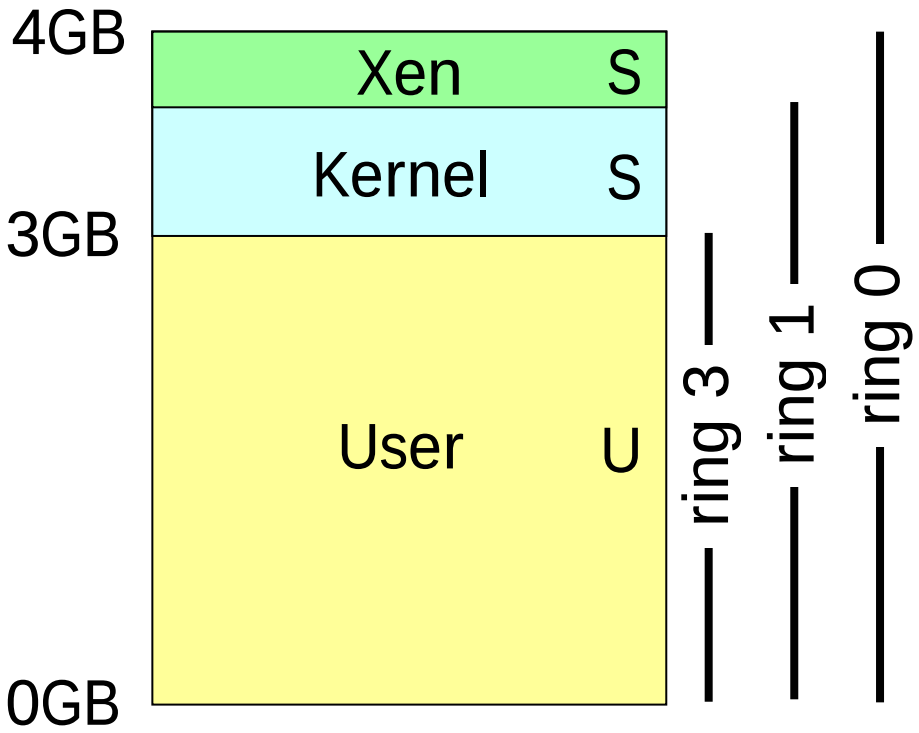



**Benchmark suite running on Linux (L), Xen (X), VMware Workstation (V), and UML (U)**

# Scalability

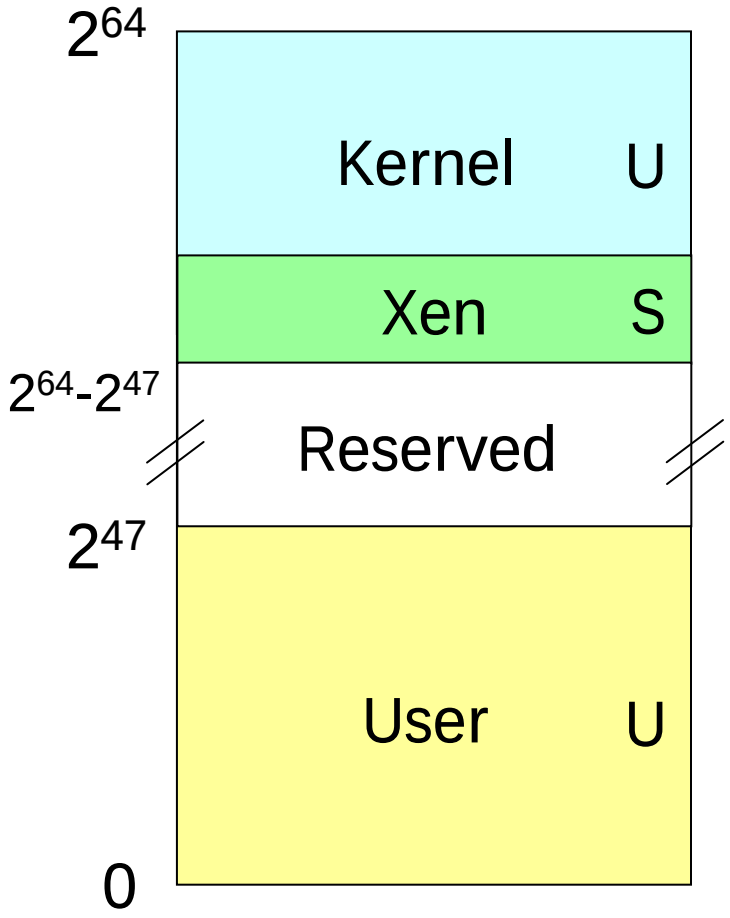


# x86_32




- **Xen reserves top of VA space**
- **Segmentation protects Xen from kernel**
- **System call speed unchanged**
- **Xen 3 now supports PAE for >4GB mem**

# x86_64

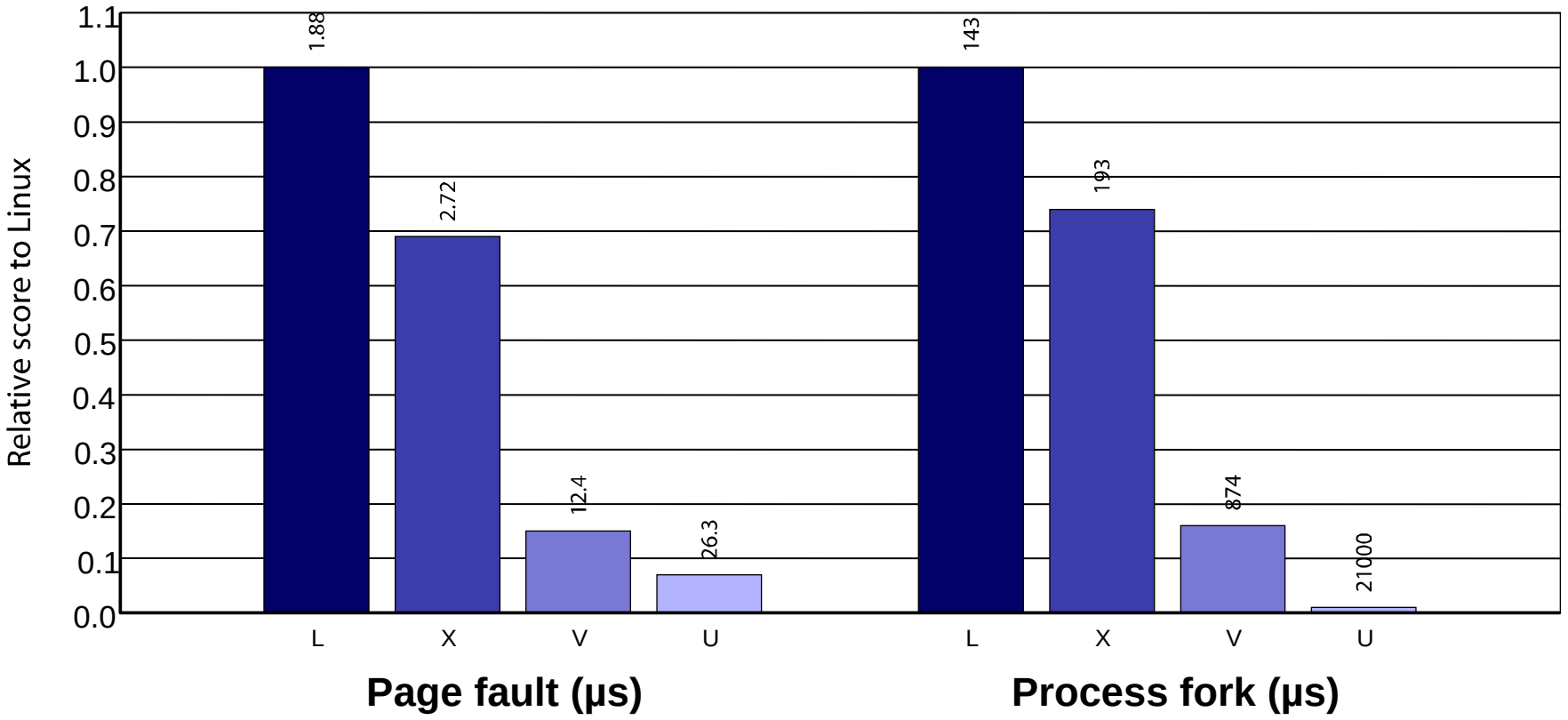



- **Large VA space makes life a lot easier, but:**
- **No segment limit support**
  - ➔Need to use page-level protection to protect hypervisor

# x86 CPU virtualization

- **Xen runs in ring 0 (most privileged)**
- **Ring 1/2 for guest OS, 3 for user-space**
  - GPF if guest attempts to use privileged instr
- **Xen lives in top 64MB of linear addr space**
  - Segmentation used to protect Xen as switching page tables too slow on standard x86
- **Hypercalls jump to Xen in ring 0**
- **Guest OS may install ‘fast trap’ handler**
  - Direct user-space to guest OS system calls
- **MMU virtualisation: shadow vs. direct-mode**

# MMU Micro-Benchmarks




**Imbench results on Linux (L), Xen (X), VMware Workstation (V), and UML (U)**

# SMP Guest Kernels

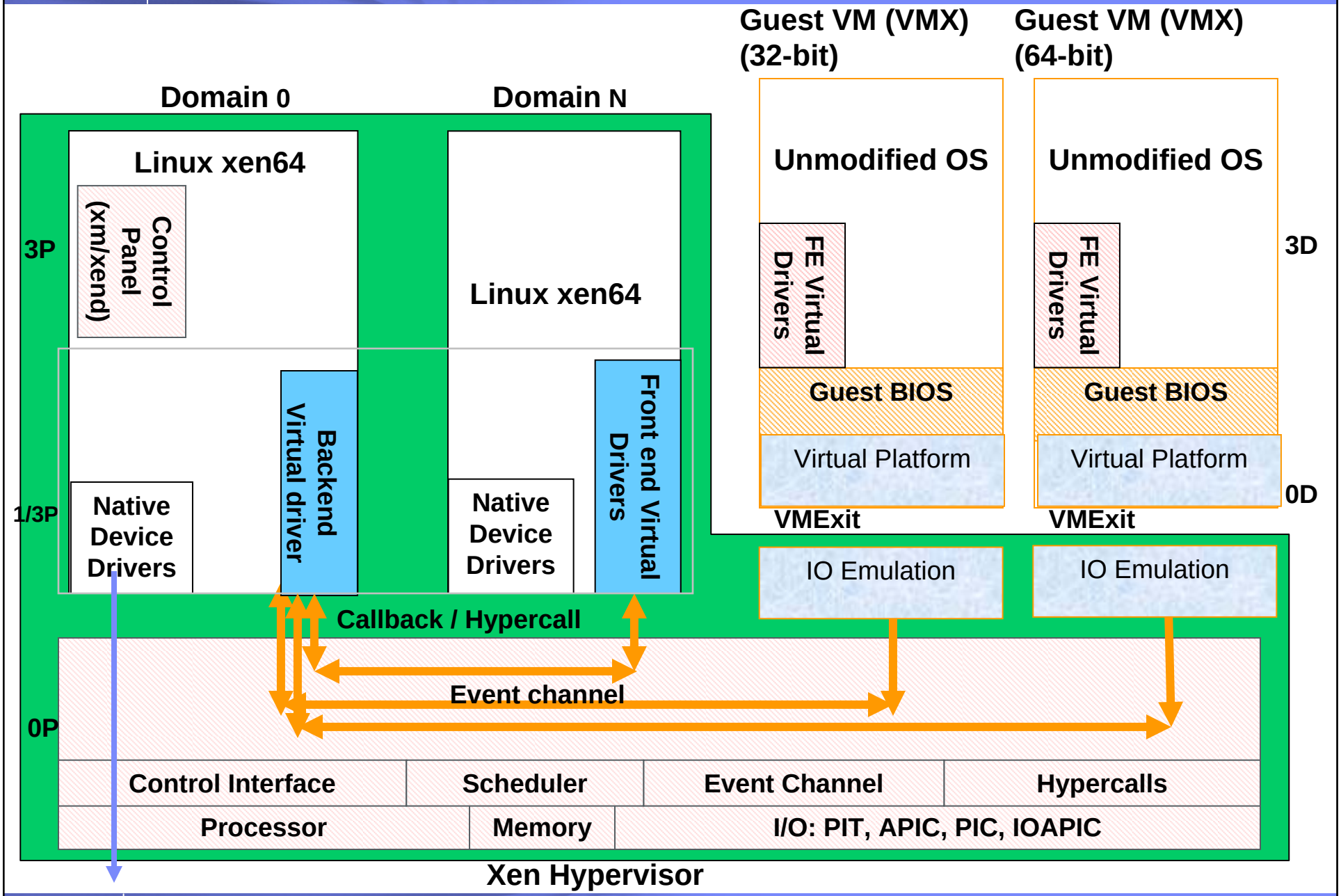
- **Xen extended to support multiple VCPUs**
  - Virtual IPI's sent via Xen event channels
  - Currently up to 32 VCPUs supported
- **Simple hotplug/unplug of VCPUs**
  - From within VM or via control tools
  - Optimize one active VCPU case by binary patching spinlocks
- **Note: Many applications exhibit poor SMP scalability – often better off running multiple instances each in their own operating system**

# SMP Guest Kernels

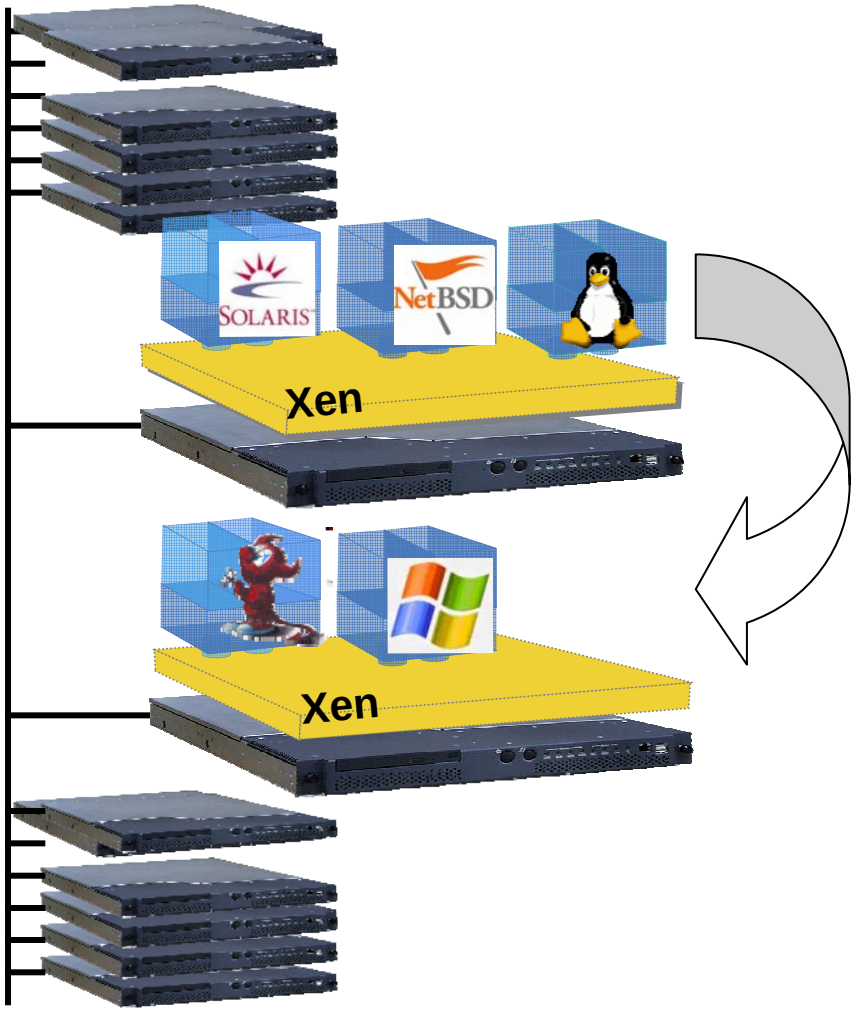
- **Takes great care to get good SMP performance while remaining secure**
  - Requires extra TLB synchronization IPIs
- **SMP scheduling is a tricky problem**
  - Wish to run all VCPUs at the same time
  - But, strict gang scheduling is not work conserving
  - Opportunity for a hybrid approach
- **Paravirtualized approach enables several important benefits**
  - Avoids many virtual IPIs
  - Allows ‘bad preemption’ avoidance
  - Auto hot plug/unplug of CPUs

# VT-x / Pacifica : hvm

- **Enable guest operating systems to be run without modification**
  - e.g. legacy Linux, Windows XP/2003
- **CPU provides vmexits for certain privileged instrs**
- **Shadow page tables used to virtualize MMU**
- **Xen provides simple platform emulation**
  - BIOS, apic, iopaic, rtc, Net (pcnet32), IDE emulation
- **Install paravirtualized drivers after booting for high-performance IO**
- **Possibility for CPU and memory paravirtualization**
  - Non-invasive hypervisor hints from OS

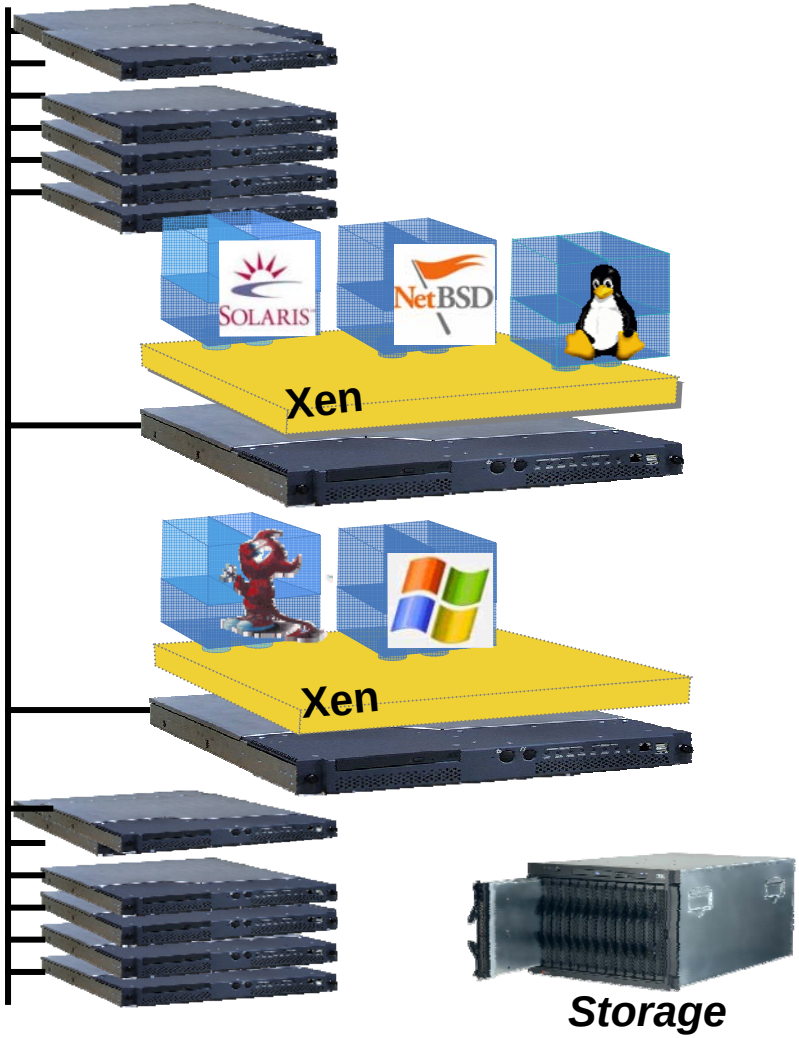

Guest VM (VMX) (32-bit)
Guest VM (VMX) (64-bit)
Domain 0
Domain N
Linux xen64
Control Panel (xm/xend)
Linux xen64
Unmodified OS
Unmodified OS
FE Virtual Drivers
FE Virtual Drivers
3P
3D
Guest BIOS
Guest BIOS
Virtual Platform
Virtual Platform
0D
1/3P
Native Device Drivers
Backend Virtual driver
Native Device Drivers
Front end Virtual Drivers
VMExit
VMExit
IO Emulation
IO Emulation
Callback / Hypercall
Event channel
0P
Control Interface
Scheduler
Event Channel
Hypercalls
Processor
Memory
I/O: PIT, APIC, PIC, IOAPIC
Xen Hypervisor

# VM Relocation : Motivation




- **VM relocation enables:**
  - High-availability
    - Machine maintenance
  - Load balancing
    - Statistical multiplexing gain

# Assumptions

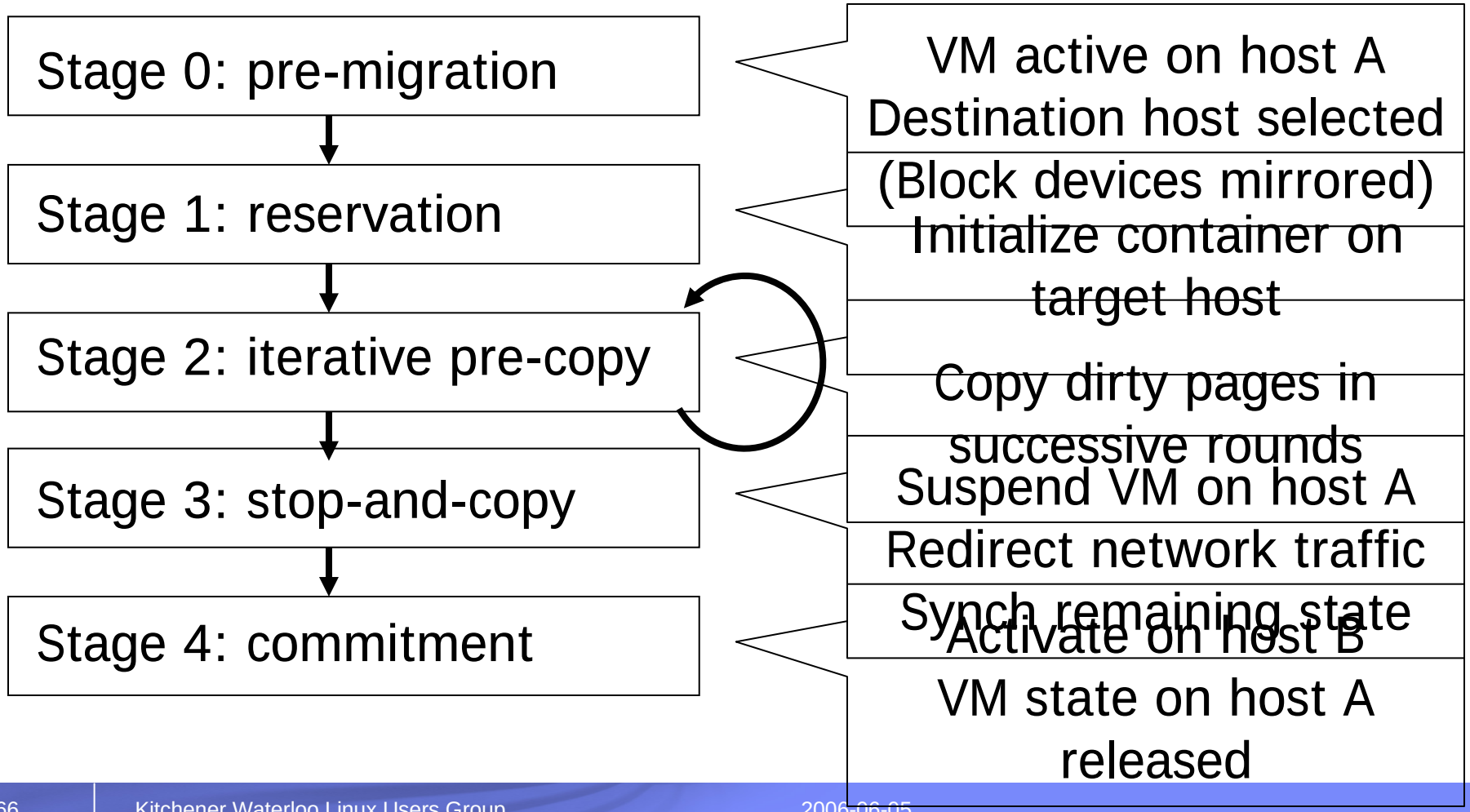



- **Networked storage**
  - NAS: NFS, CIFS
  - SAN: Fibre Channel
  - iSCSI, network block dev
  - drdb network RAID
- **Good connectivity**
  - common L2 network
  - L3 re-routeing

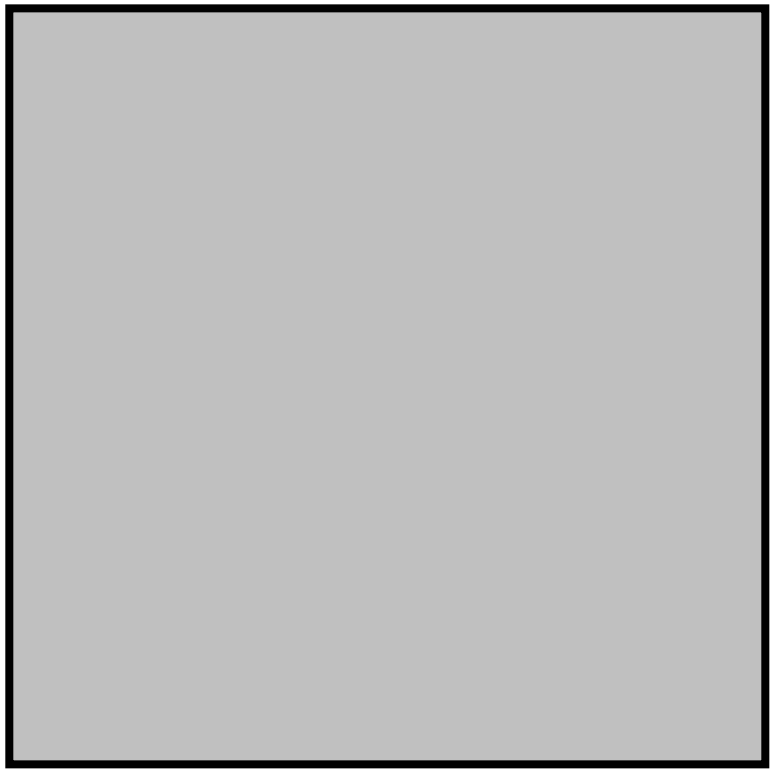
# Challenges

- **VMs have lots of state in memory**
- **Some VMs have soft real-time requirements**
  - e.g. web servers, databases, game servers
  - May be members of a cluster quorum
  - ➔ **Minimize down-time**
- **Performing relocation requires resources**
  - ➔ **Bound and control resources used**

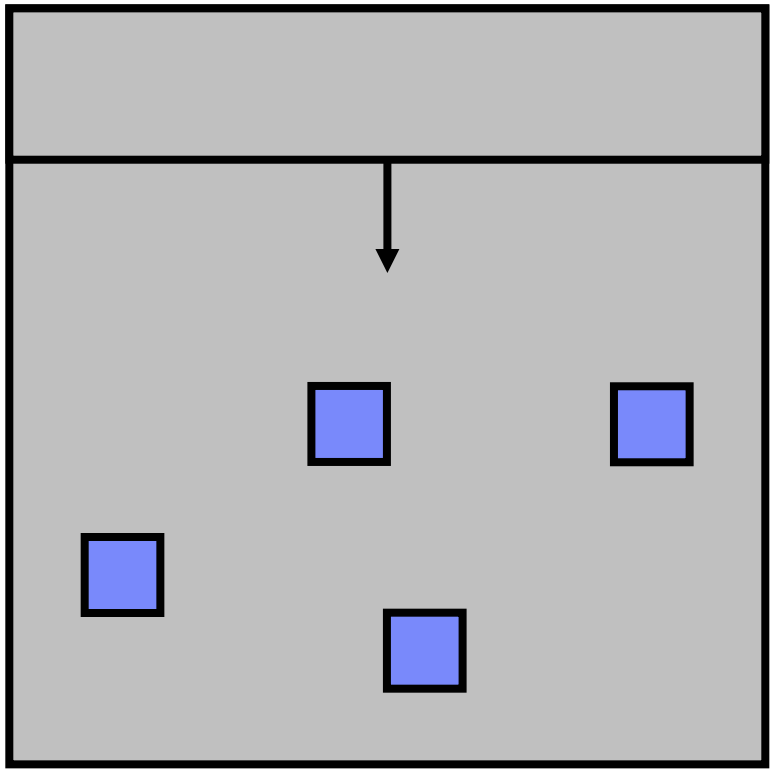
# Relocation Strategy

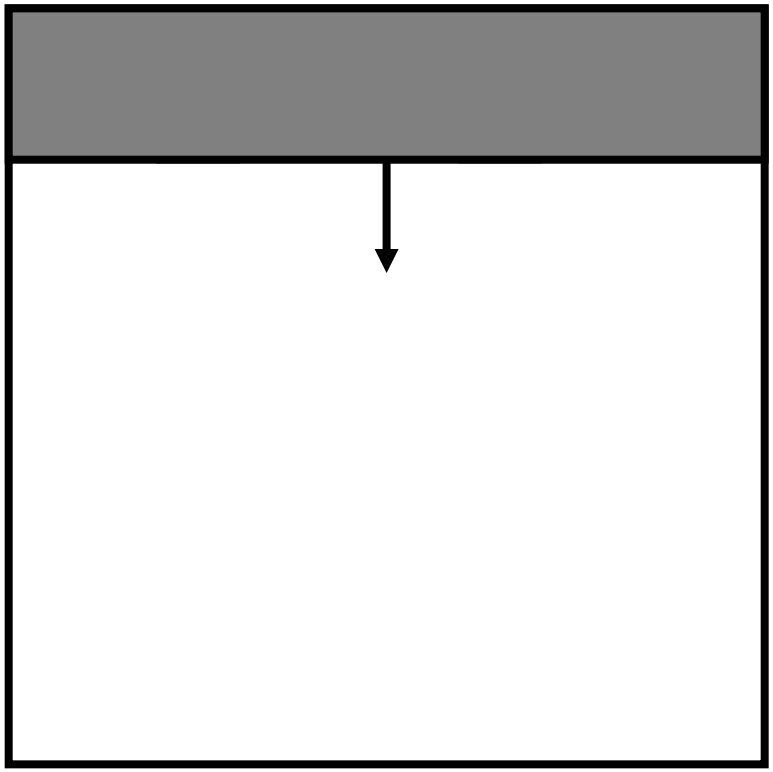


# Pre-Copy Migration: Round 1

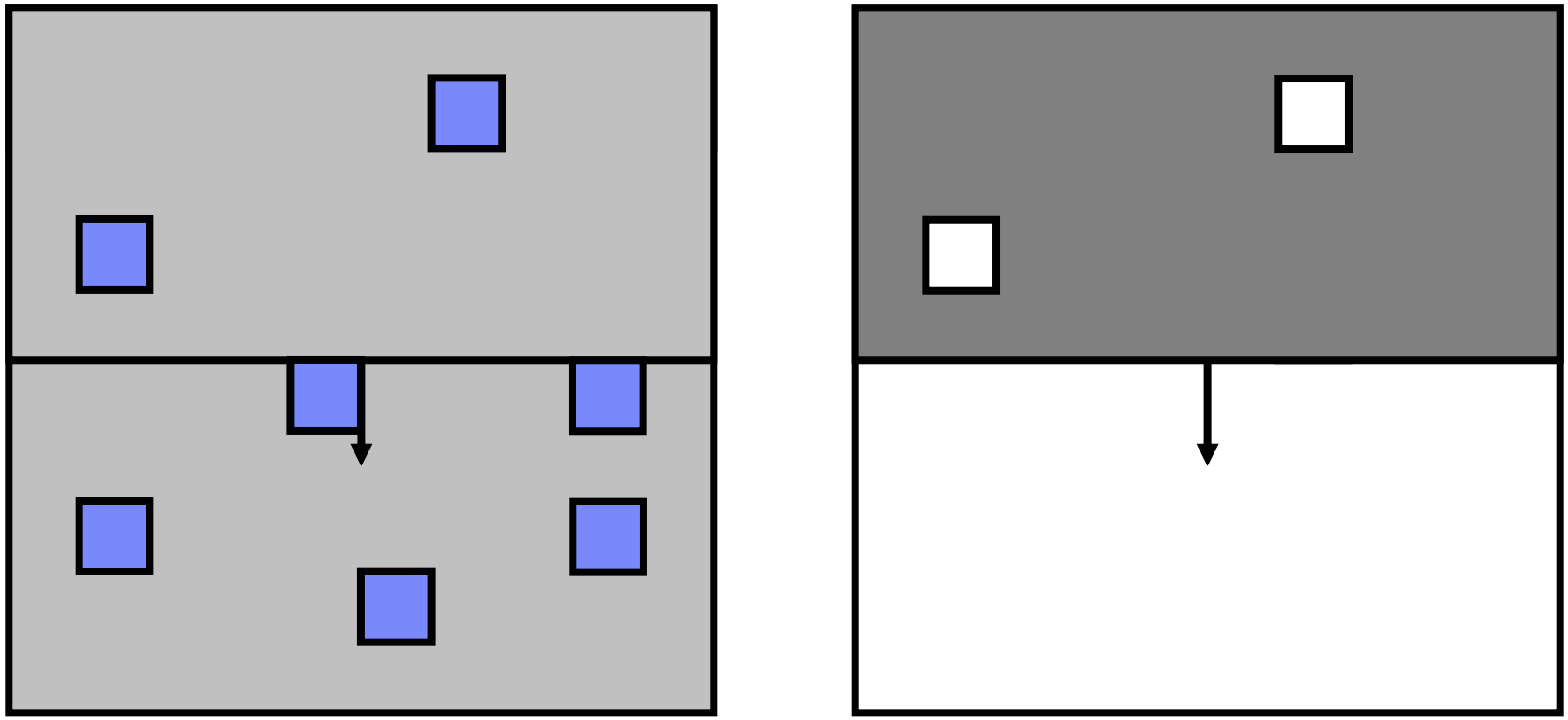


# Pre-Copy Migration: Round 1

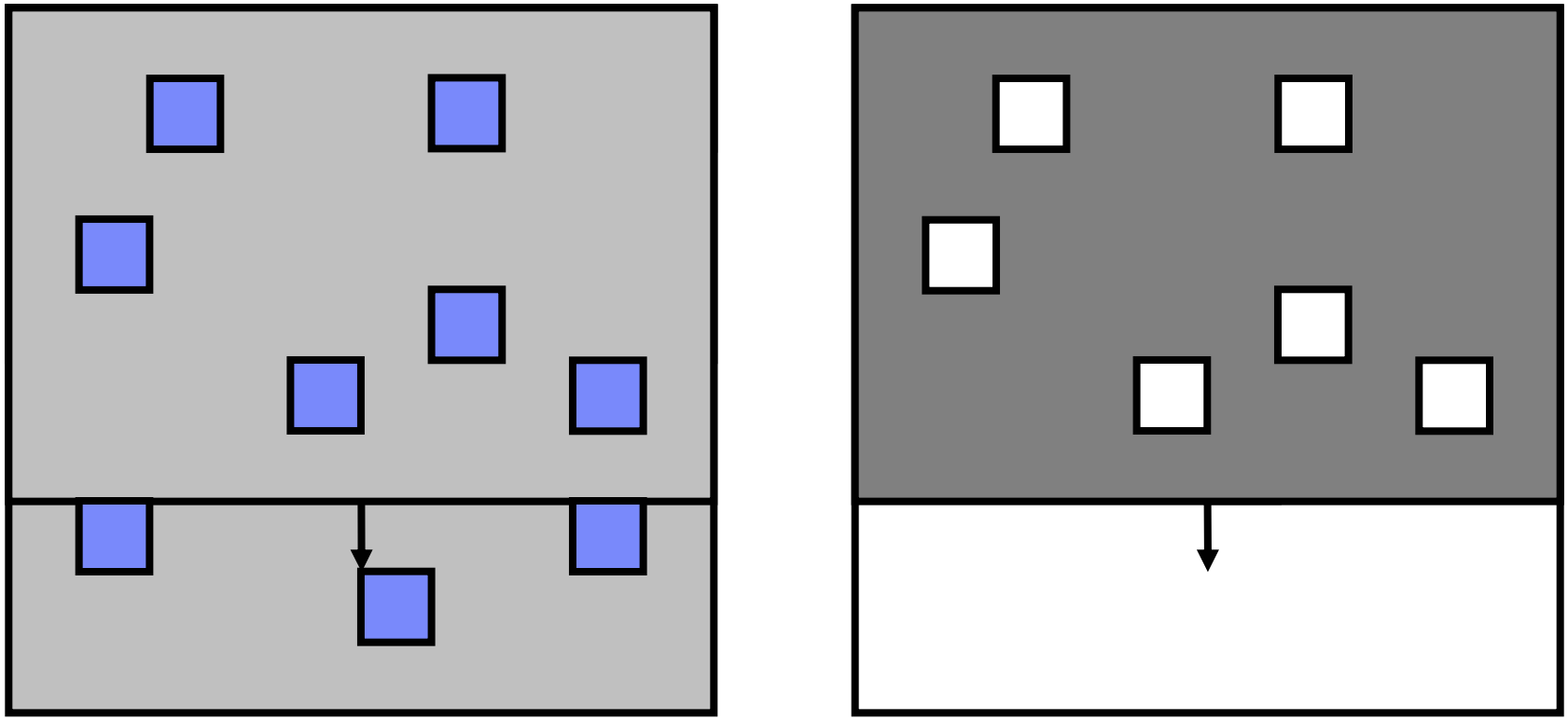




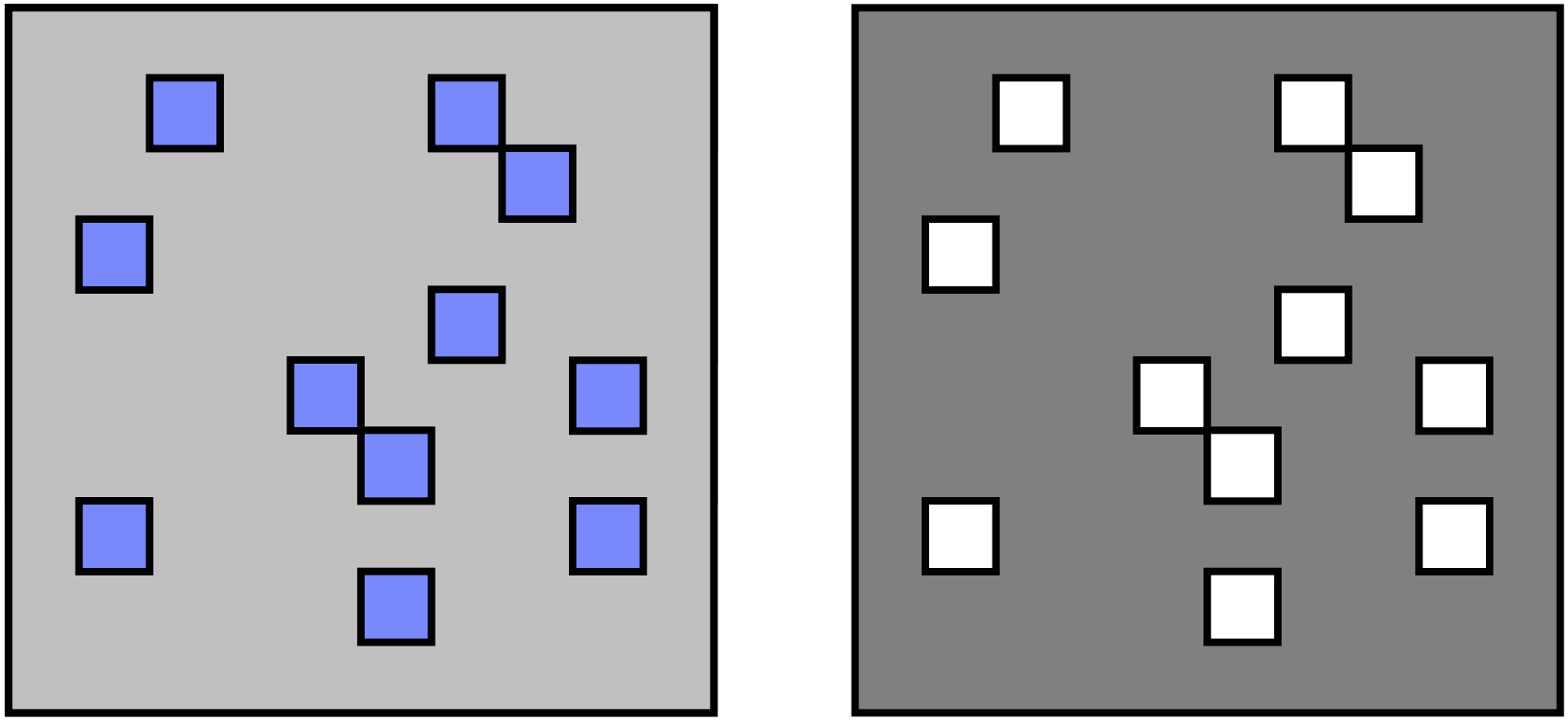
# Pre-Copy Migration: Round 1



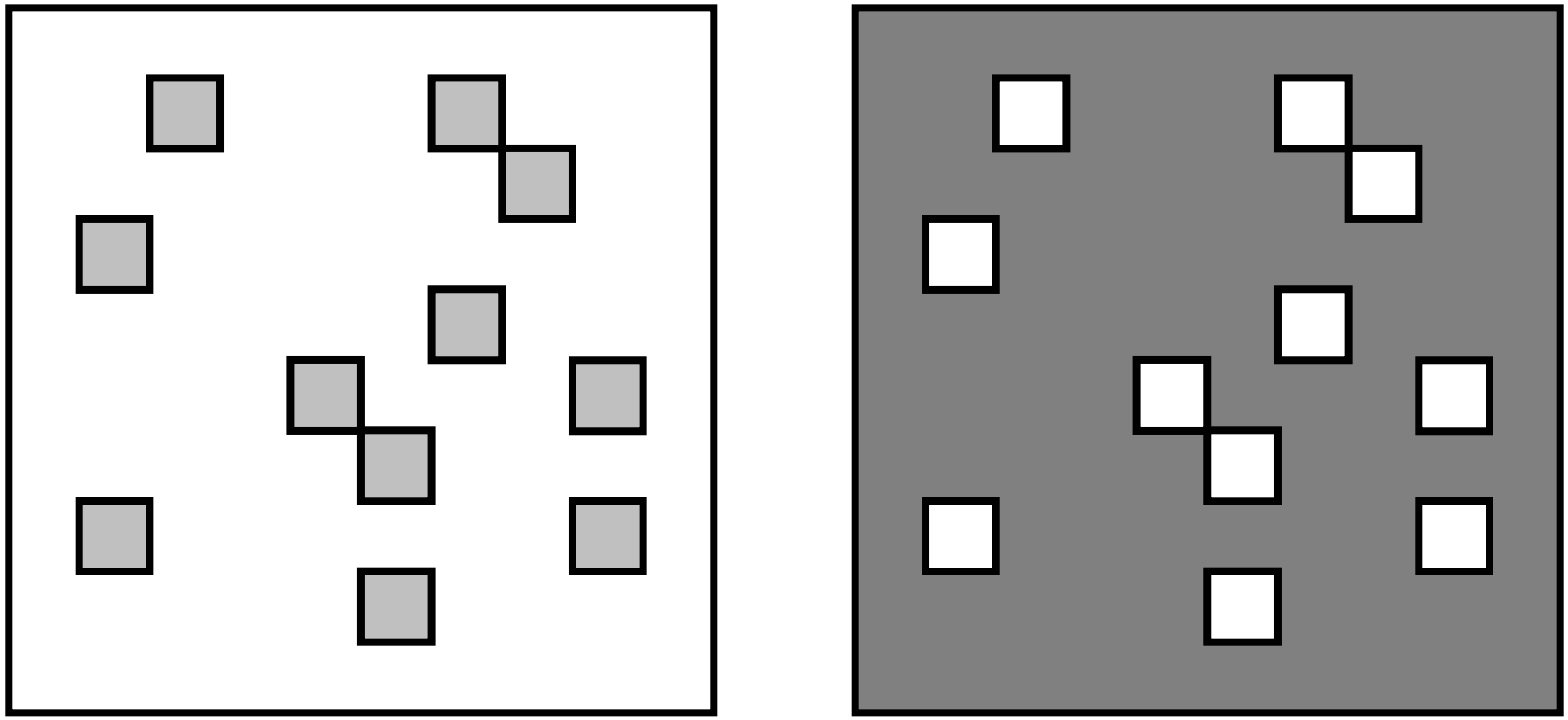
# Pre-Copy Migration: Round 1



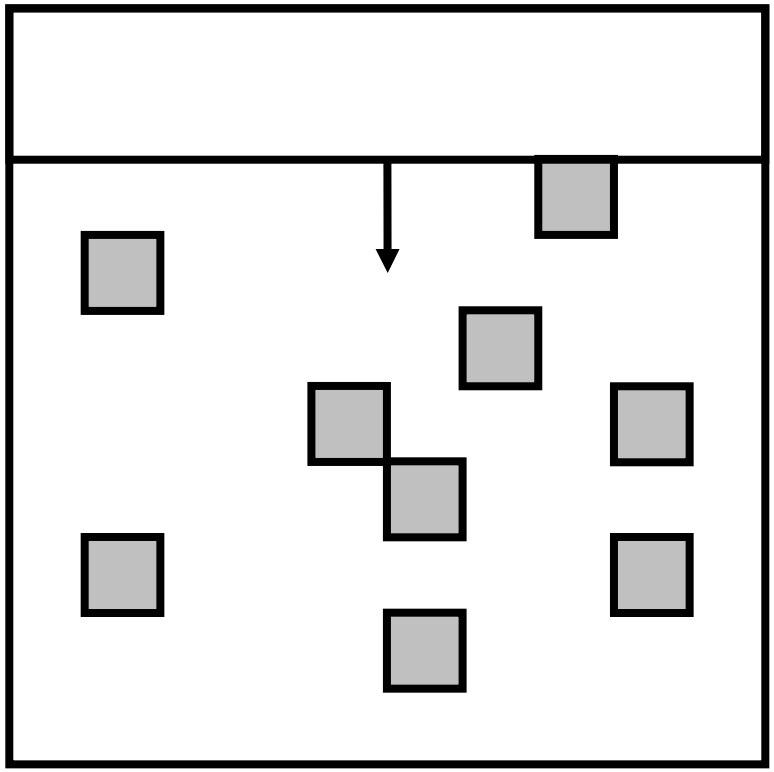
# Pre-Copy Migration: Round 1

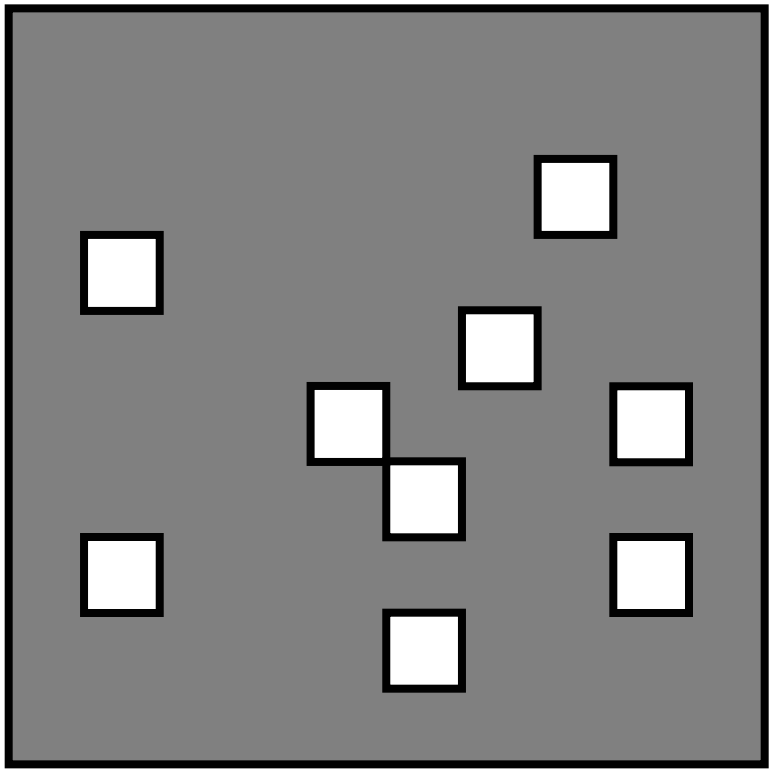


# Pre-Copy Migration: Round 2

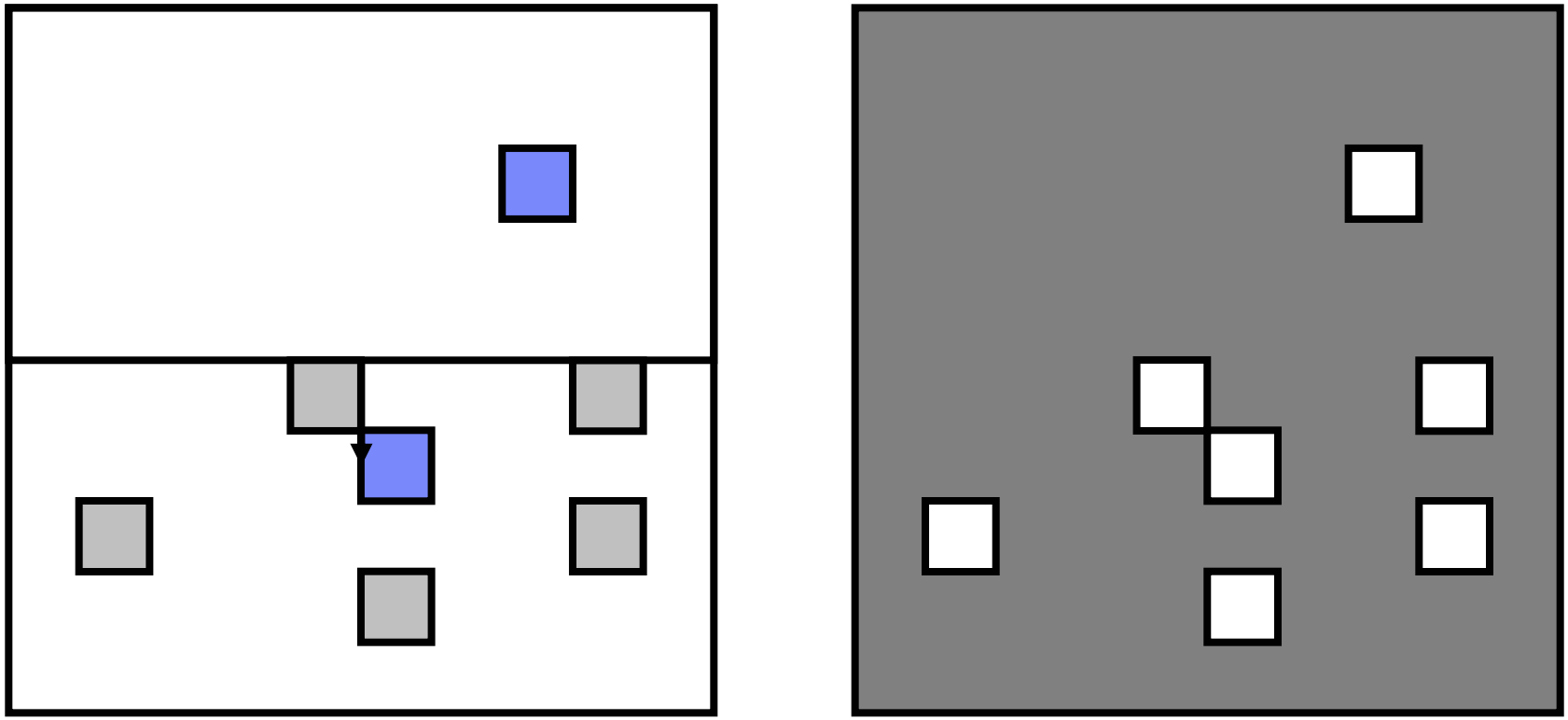


# Pre-Copy Migration: Round 2

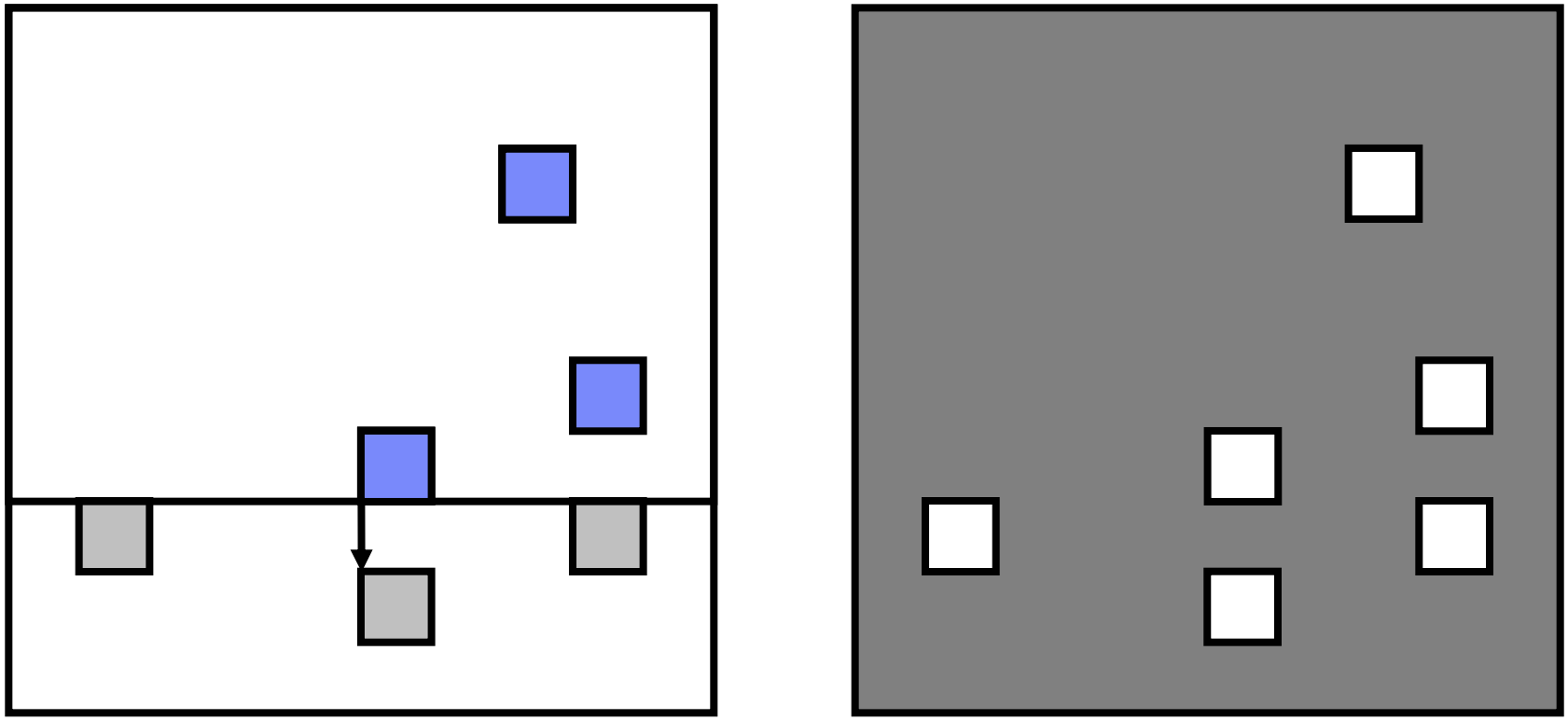




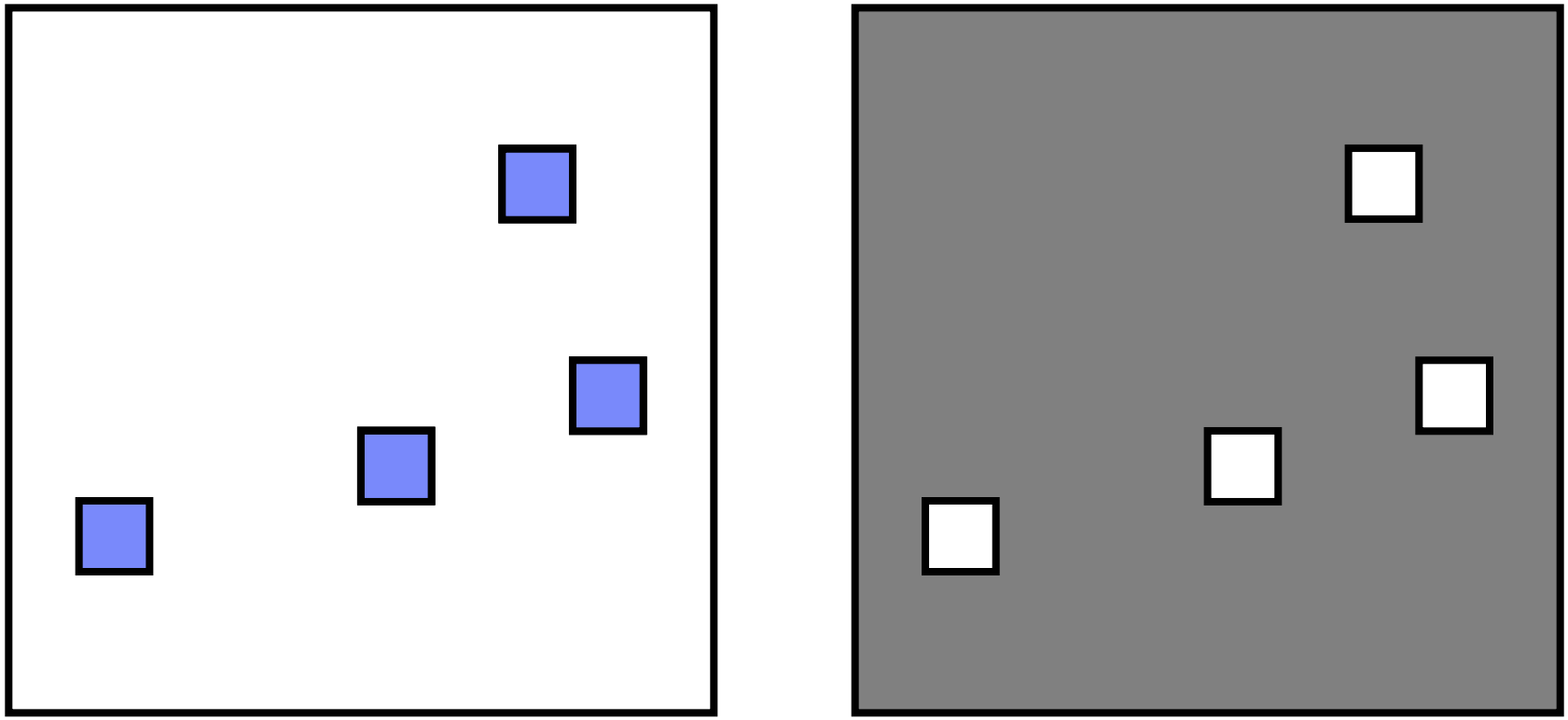
# Pre-Copy Migration: Round 2



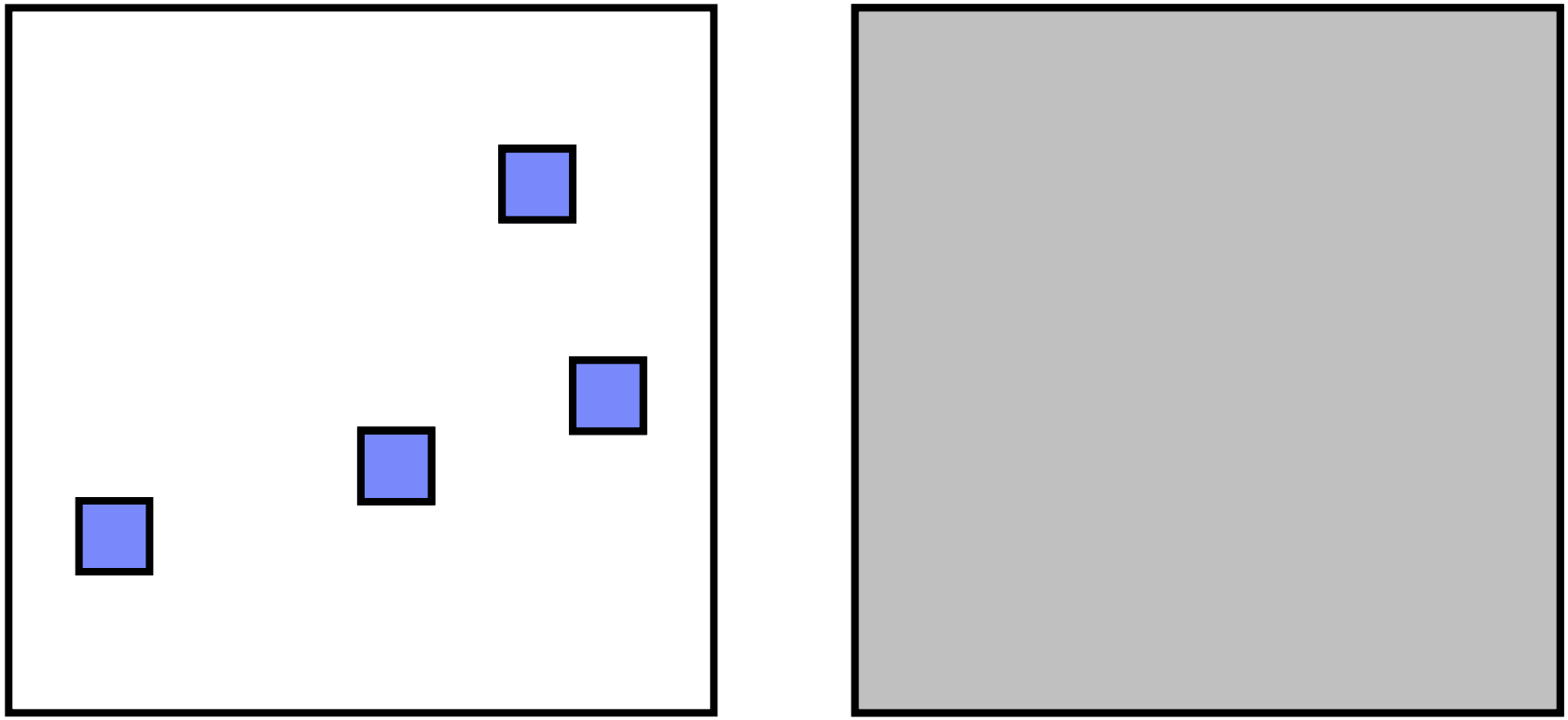
# Pre-Copy Migration: Round 2



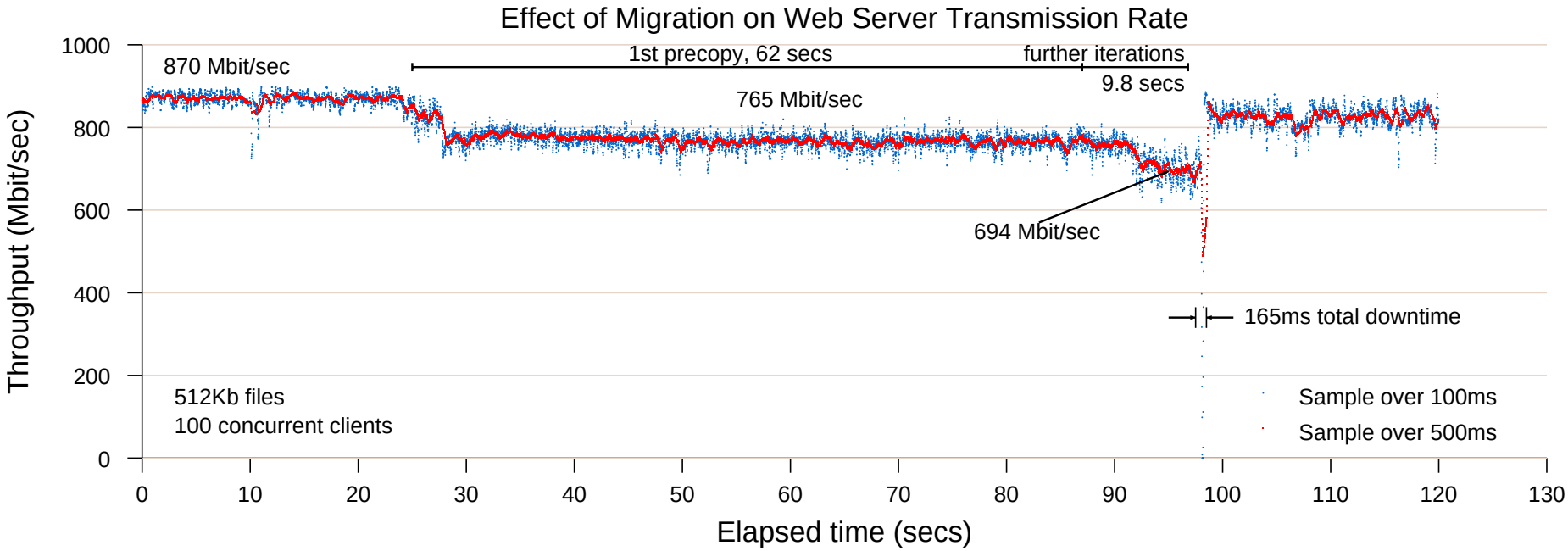
# Pre-Copy Migration: Round 2



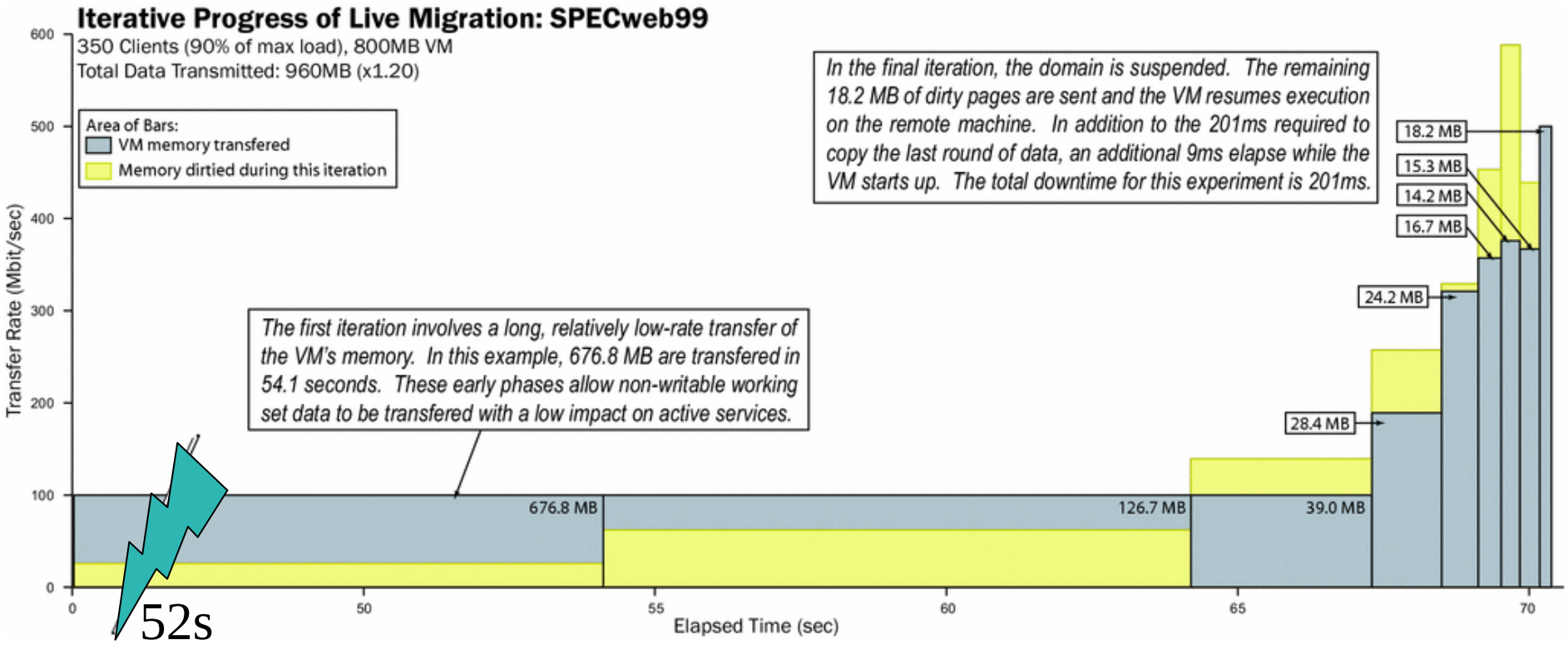
# Pre-Copy Migration: Final



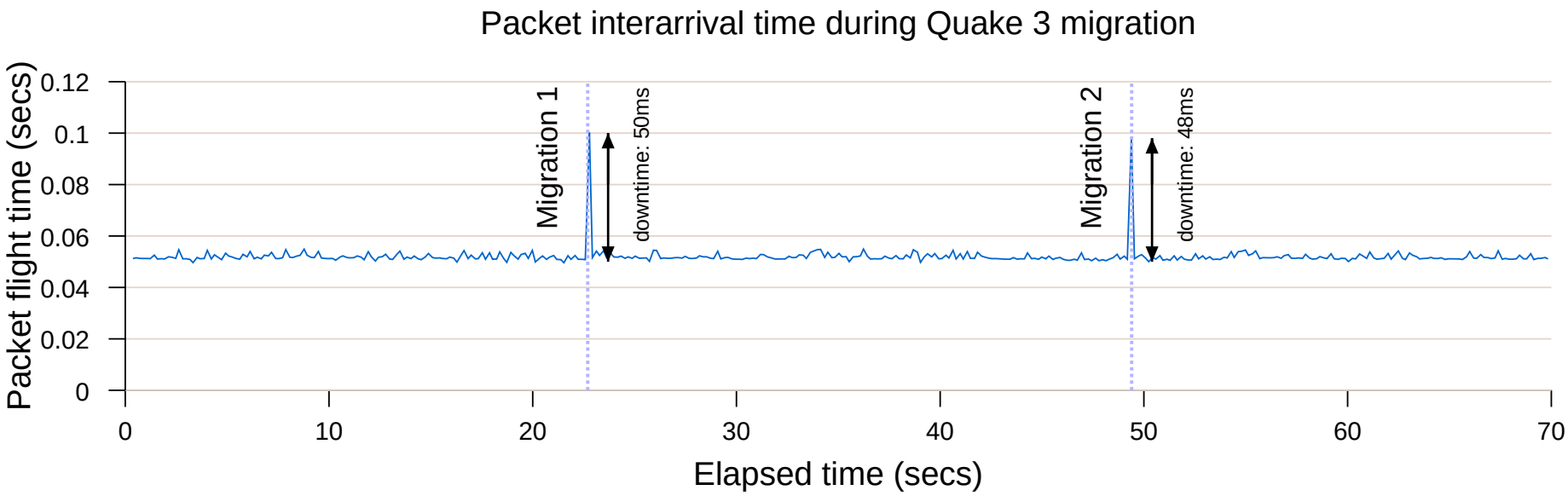
# Web Server Relocation



# Iterative Progress: SPECWeb



# Quake 3 Server relocation



# Current Status

| | x86_32 | x86_32p | x86_64 | IA64 | Power |
|---|---|---|---|---|---|
| Privileged Domains | | | | | |
| Guest Domains | | | | | |
| SMP Guests | | | | | |
| Save/Restore/Migrate | | | | | |
| >4GB memory | | | | | |
| VT | | | | | |
| Driver Domains | | | | | |

# 3.1 Roadmap

- **Improved full-virtualization support**
  - Pacifica / VT-x abstraction
  - Enhanced IO emulation
- **Enhanced control tools**
- **Performance tuning and optimization**
  - Less reliance on manual configuration
- **NUMA optimizations**
- **Virtual bitmap framebuffer and OpenGL**
- **Infiniband / “Smart NIC” support**

# IO Virtualization

- **IO virtualization in software incurs overhead**
  - Latency vs. overhead tradeoff
    - More of an issue for network than storage
  - Can burn 10-30% more CPU
- **Solution is well understood**
  - Direct hardware access from VMs
    - Multiplexing and protection implemented in h/w
  - Smart NICs / HCAs
    - Infiniband, Level-5, Aaorhi etc
    - Will become commodity before too long

# Research Roadmap

- **Whole-system debugging**
  - Lightweight checkpointing and replay
  - Cluster/distributed system debugging
- **Software implemented hardware fault tolerance**
  - Exploit deterministic replay
- **Multi-level secure systems with Xen**
- **VM forking**
  - Lightweight service replication, isolation

# IBM and Xen

- **IBMers on the Xen development team**
  - Tony Breeds – Canberra
  - Sean Dague – Poughkeepsie
  - Todd Deshane – Poughkeepsie
  - James Dykman – Poughkeepsie
  - Jerone Young – Austin
- **IBM contributions to the Xen project**
  - Research Hypervisor – rHype has been developed to validate virtualization features in new hardware architectures (such as x86, Cell BE and POWER) and to study fundamental research issues in virtualization
  - Secure Hypervisor – sHype is a hypervisor security architecture in various stages of implementation in several hypervisors

# Conclusions

- Xen is a complete and robust hypervisor
- Outstanding performance and scalability
- Excellent resource control and protection
- Vibrant development community
- Strong vendor support

- Try the demo CD to find out more!
  (or Fedora 4/5, OpenSUSE 10.x)

- http://xensource.com/xen/

# Contact info

**Jim Elliott**
Advocate – Infrastructure Solutions and
Manager – System z Operating Systems
IBM Canada Ltd.
jim_elliott@ca.ibm.com
905-316-5813

**Linux at IBM ➔ ibm.com**/linux

**System z ➔ ibm.com**/systems/z

**My web site ➔ ibm.com**/vm/devpages/jelliott

**My blog ➔** linux.ca/drupal/blog/58

# Notices



**Notes:**

This publication was produced in Canada. IBM may not offer the products, services or features discussed in this document in other countries, and the information may be subject to change without notice. Consult your local IBM business contact for information on the product or services available in your area.
All statements regarding IBM's future direction and intent are subject to change or withdrawal without notice, and represent goals and objectives only.
Information about non-IBM products is obtained from the manufacturers of those products or their published announcements. IBM has not tested those products and cannot confirm the performance, compatibility, or any other claims related to non-IBM products. Questions on the capabilities of non-IBM products should be addressed to the suppliers of those products.
Prices subject to change without notice. Contact your IBM representative or Business Partner for the most current pricing in your geography.