# Enterprise Security in the Brave New (Virtual) World

***Tal Garfinkel***

*VMware Advanced Development*

# The VMware Market Perspective

30 Largest Commercial Banks

5 Largest Securities Companies

5 Largest Chemical Companies

12 Largest Pharmaceutical Companies

10 Largest Aerospace/Defense Companies

5 Largest Entertainment Companies

100% of Fortune 100

91% of Fortune 1000

# Other perspectives as well

Microsoft
Novell.
Sun microsystems
symantec.
CITRIX
QUMRANET
Virtual Iron
Parallels

# Many Business/Deployment Models

In House

Managed Outsourcing

Cloud (Local/Remote)

# Adoption Drivers/Use Cases

**Test and Development** – Rapidly provision test configurations; libraries of pre-configured test machines, high volume cross platform testing

**Server Consolidation and Dynamic Resource Managment** – Increased utilization, decreased hardware cost, dynamic load balancing, dynamic power management

**Business Continuity** – Disaster recovery, consolidated backup, high availability

**Enterprise Desktop Management** – Remote display, managed mobile desktops

# Overview

What is Virtualization

From Virtual Machines To Virtual Infrastructure

Security Challenges and Opportunities

Emerging Technologies

# What is Virtualization?

**vir•tu•al (adj): existing in essence or effect, though not in actual fact**

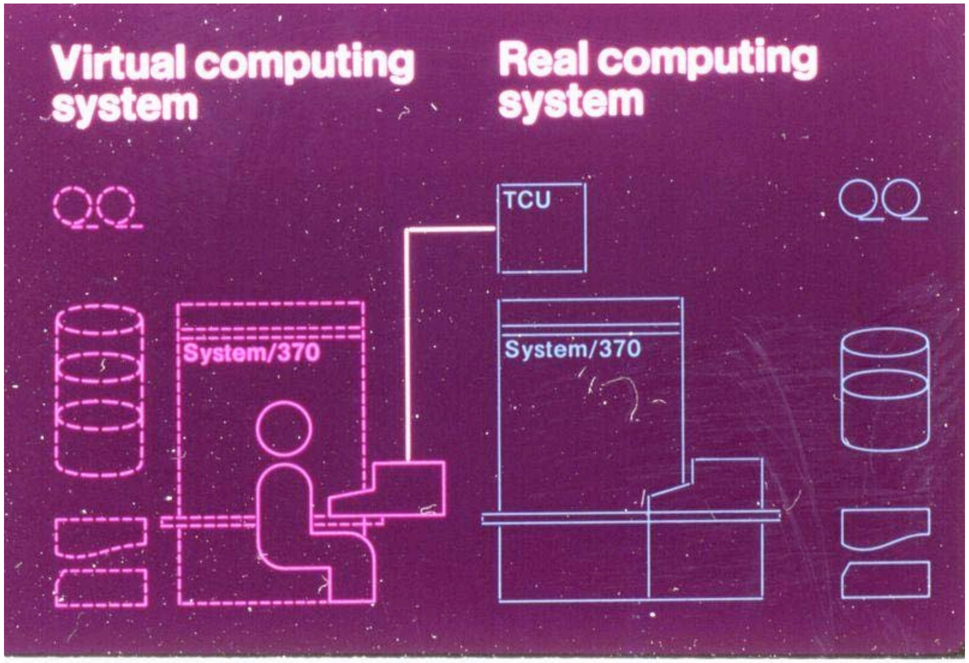
Virtual systems

- Abstract physical components using logical objects
- Dynamically bind logical objects to physical configurations

Examples

- Network – Virtual LAN (VLAN), Virtual Private Network (VPN)
- Storage – Storage Area Network (SAN), LUN
- Computer – Virtual Machine (VM), simulator

# VMMs Past



From IBM VM/370 product announcement, *ca*. 1972

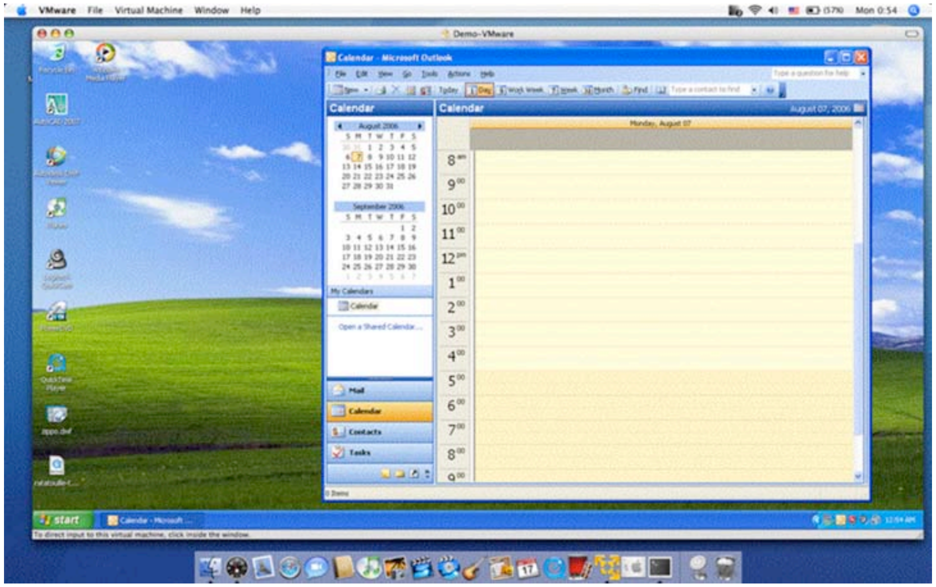
## An Old Idea

- Hardware-level VMs since ’60s
- CP/CMS, IBM S/360 and VM/370 mainframes
- Timeshare multiple single-user OS instances on expensive hardware

## Classical VMM

- Run VM directly on hardware
- Vendors had vertical control over proprietary hardware, operating systems, VMM

# VMMs Present


VMware File Virtual Machine Window Help
Demo-VMware
Calendar - Microsoft Outlook
Calendar
Mail
Contacts
Tasks
start

VMware Fusion for Mac OS X running WinXP, 2006

## Renewed Interest

- Academic research since ’90s
- VMs for commodity systems
- Broad range of IT applications

## VMM for x86

- Industry-standard hardware, from laptops to datacenter
- Commodity guest operating systems

## Not your father’s VMM

# Modern Divergence

## Old World

- Machine centric view
- VMs relatively static
- Focus on logical partitioning
- Mainframe

## New World

- Resource centric view
- VMs highly mobile
- Focus on IT automation
- Multitude of uses
  - Consolidation
  - IT automation engine
  - OS independent technology platform
  - Server to Desktop

# VMM Platform Types

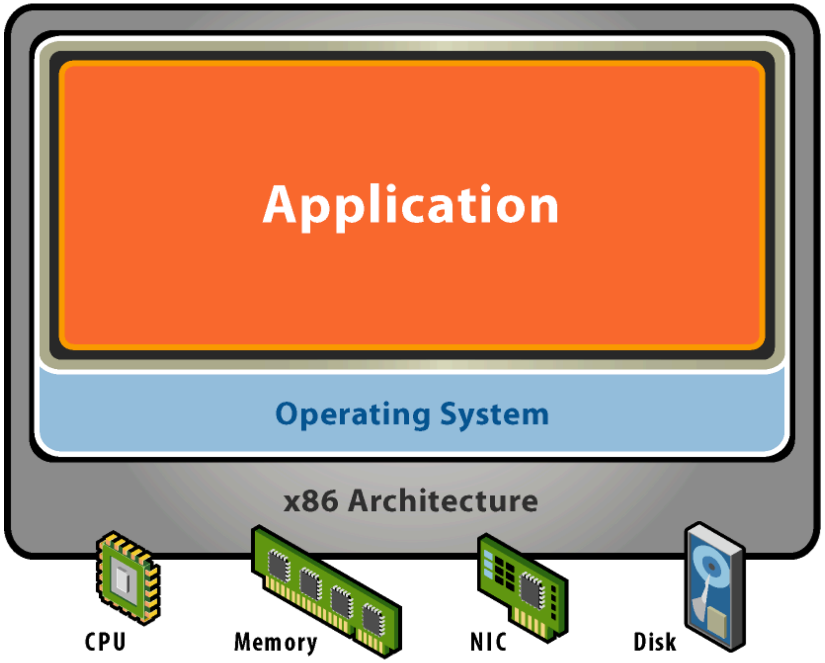
Hosted Architecture (what most people know)

- Install as application on existing x86 “host” OS, *e.g.* Windows, Linux, OS X
- Small context-switching driver
- Leverage host I/O stack and resource management
- Examples: VMware Workstation and Fusion, Parallels

Bare-Metal Architecture (what most companies use)

- “Hypervisor” installs directly on hardware
- Dominant architecture for enterprise servers
- Does its own resource management
- Examples: VMware ESX Server, Xen, Microsoft Hyper-V

# Starting Point: A Physical Machine



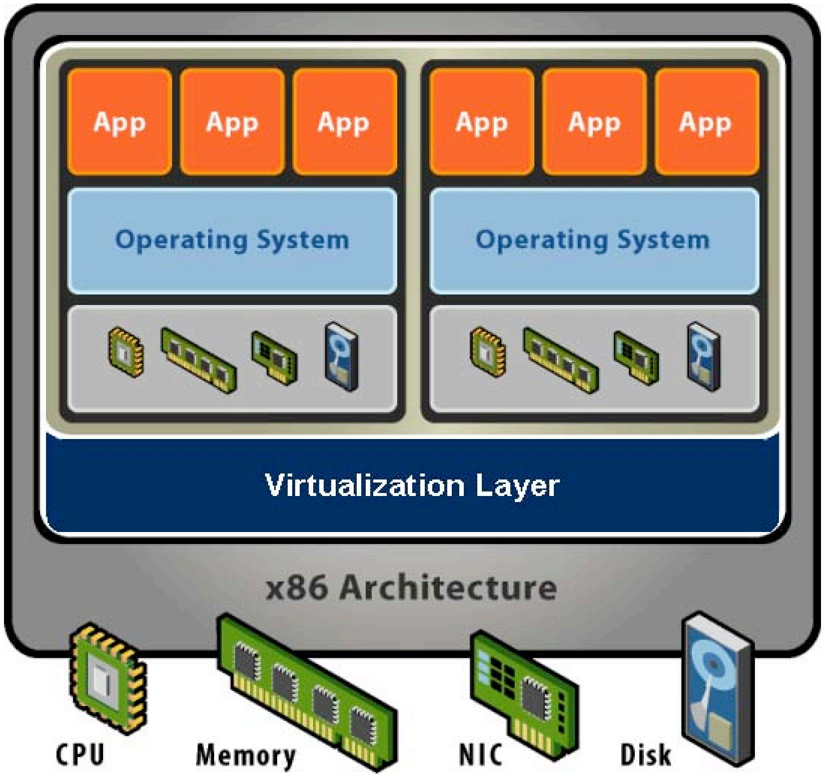
## Physical Hardware

- Processors, memory, chipset, I/O bus and devices, etc.
- Physical resources often underutilized

## Properties

- OS Tightly coupled to hardware
- Single active OS image
- OS controls hardware
- OS Abstractions focus on sharing

# What is a Virtual Machine?



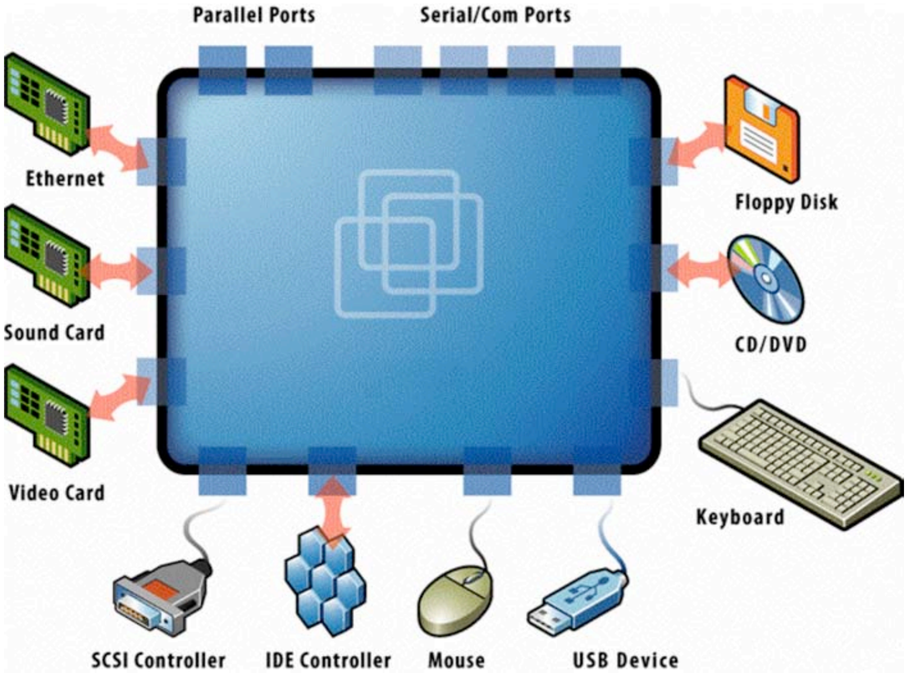
Hardware-Level Process Abstraction: CPU, memory, chipset, I/O devices, etc.

- Virtual NIC instead of sockets
- Virtual disk instead of file system
- Hardware state becomes software state

Virtualization Software

- Hardware and software decoupled
- Abstractions focus on isolation

# VM Compatibility



Hardware-Independent interface

- Emulate common set of devices (lowest common denominator), SCSI, E1000
- Standard paravirt interface (common case)

Create Once, Run Anywhere

- No configuration issues
- Migrate VMs between hosts (sort of)

VM becomes common image format

- no worries about upgrades, hardware diversity, driver hell, etc.

# VM Isolation

Application
Operating System
Application
Operating System
Application
Operating System

Multiplexing

- Run multiple VMs on single physical host
- Processor hardware isolates VMs, *e.g.* MMU + protection rings

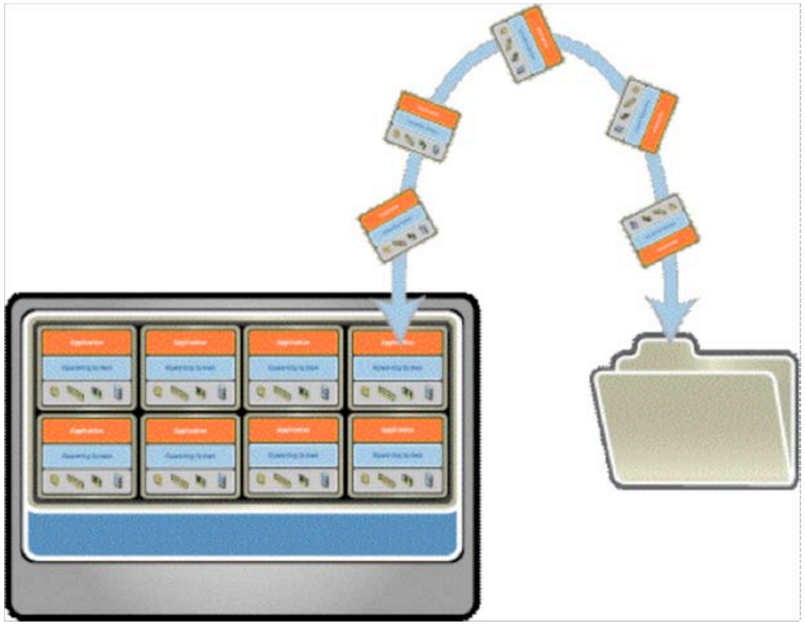
Security and Fault Isolation

- Software bugs, crashes, malicious code

Performance Isolation

- Partition system resources
- Example: VMware controls for reservation, limit, shares

# VM Encapsulation



Entire VM is a File

- OS, applications, data
- Memory and device state

Snapshots and Clones

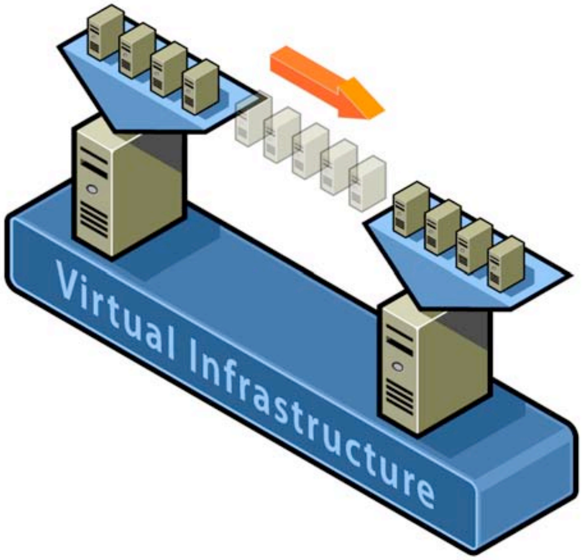
- Capture VM state on the fly and restore to point-in-time
- Rapid system provisioning, backup, remote mirroring

Easy Content Distribution

- Pre-configured apps, demos
- Virtual appliances

OS Independent state abstraction

# VM Hot Migration(aka VMotion)



“Hot” migrate VM across hosts

- Transparent to guest OS, apps
- Minimal downtime (sub-second)

Details

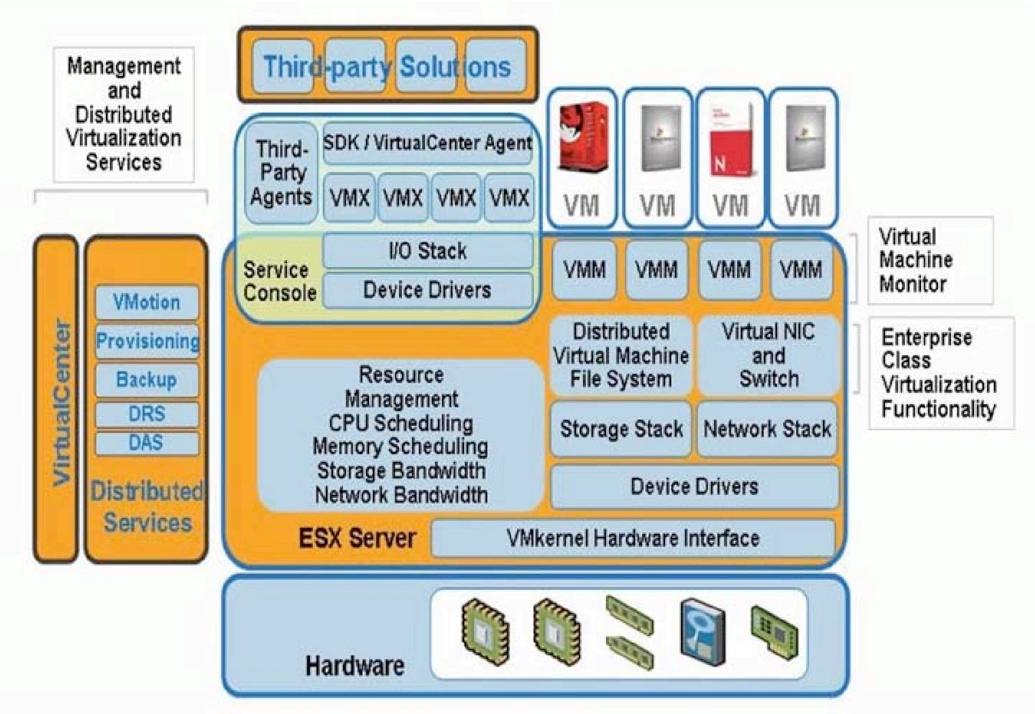
- Pre-copy iteration sends modified pages
- Repeatedly pre-copy “diff” until converge

Zero down time hardware maintenance

Compare to traditional process migration...

# The Virtual Machine Monitor

(Operating system that runs operating systems)



## Lots of unique problems

Fair accounting (similar to realtime)

Co-scheduling (similar to MPP or cluster)

- VM level
- core level

Efficient Memory management

- Multi-level overcommit
- Content based sharing

## IO

- Many VMs
- Incoming traffic/Cross Traffic
- Multi-pathing (fate sharing)

## Commodity VMM is a myth

- Same interface
- Different performance, reliability, fairness

# Overview

What is Virtualization

From Virtual Machines To Virtual Infrastructure

Security Challenges and Opportunities

Emerging Technologies

# Virtual Infrastructure (Distributed system with VMs)

Data center scale resource management

- Virtual CPU, storage, networks, etc.
- Independent of OS and physical infrastructure

Engine for IT workflow  automation (SAP for IT)

- VMM’s + storage deal with data
- Management infrastructure deals with metadata

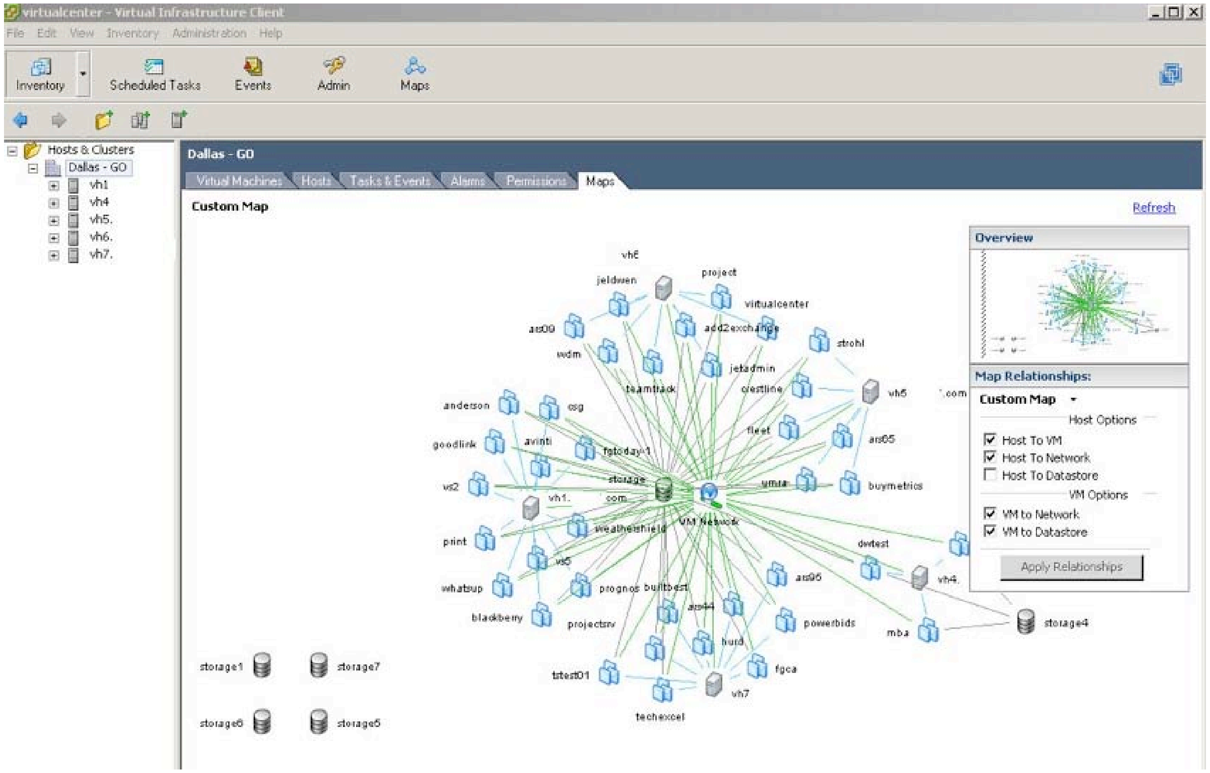
One way to manage everything

- Image management, backup, remote display, etc.
- Desktops, servers, laptops
- Can drop VMs on the metal when sensible

# The Decoupled Datacenter (Physical configuration as Metadata)

Physical (traditional)

Virtual



Map Relationships:

Custom Map

Host Options

☑ Host To VM

☑ Host To Network

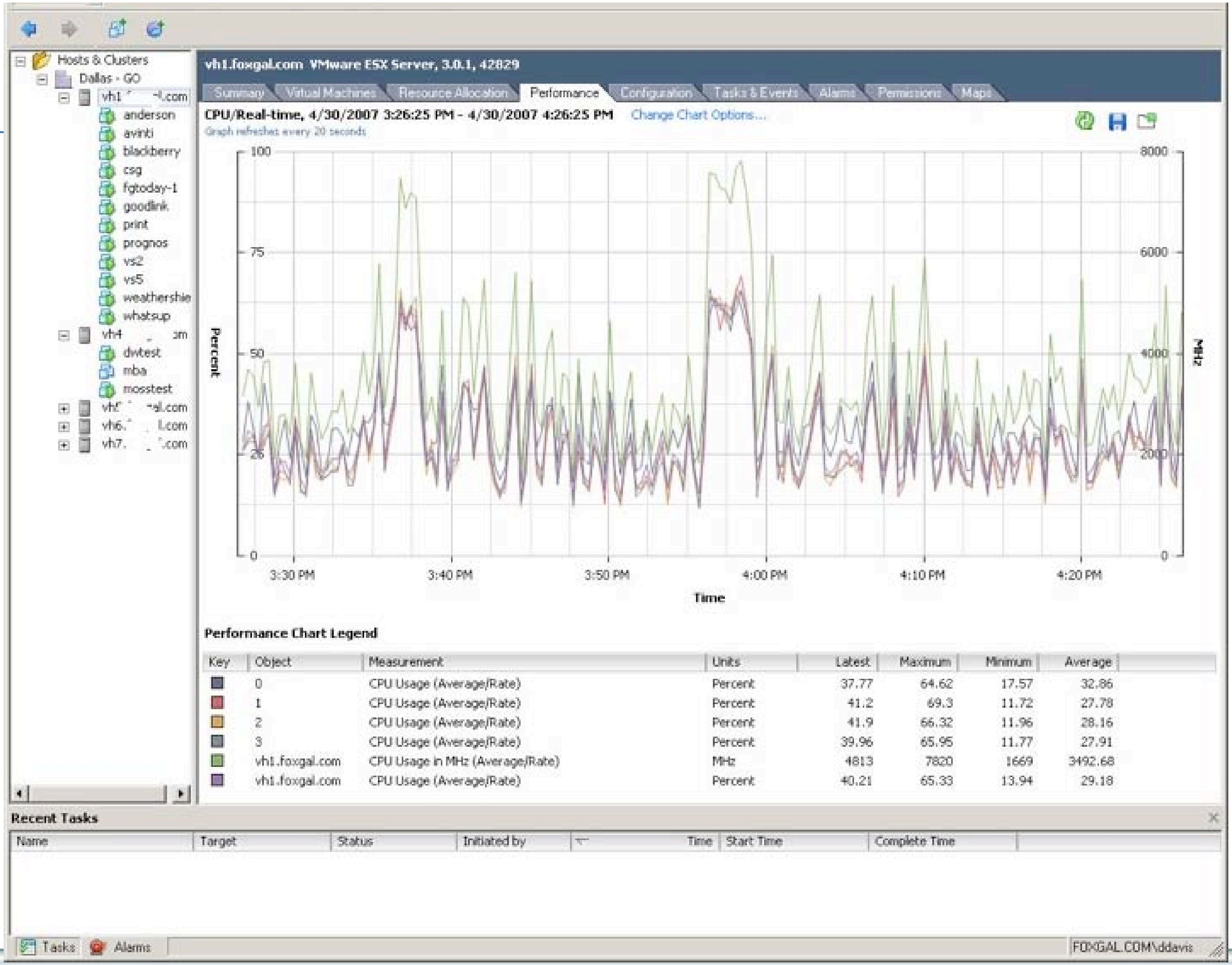
☐ Host To Datastore

VM Options

☑ VM to Network

☑ VM to Datastore

Apply Relationships



vh1.foxgal.com VMware ESX Server, 3.0.1, 42829

CPU/Real-time, 4/30/2007 3:26:25 PM - 4/30/2007 4:26:25 PM

Graph refreshes every 20 seconds

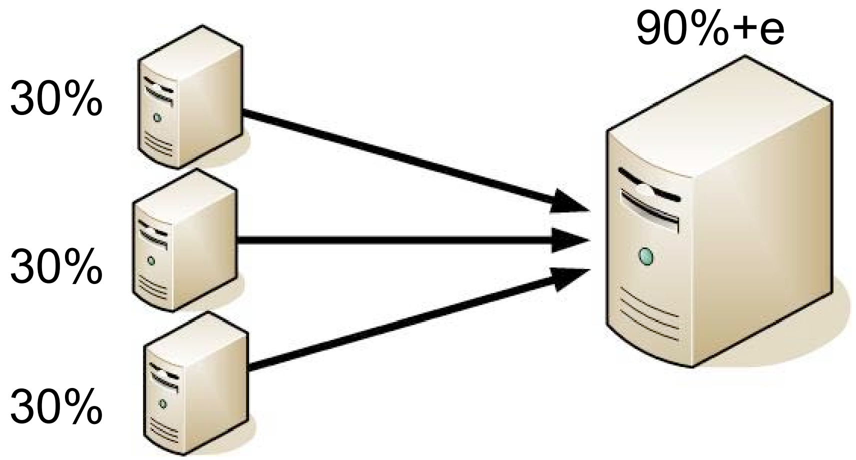
Performance Chart Legend

| Key | Object | Measurement | Units | Latest | Maximum | Minimum | Average |
|---|---|---|---|---|---|---|---|
| ■ | 0 | CPU Usage (Average/Rate) | Percent | 37.77 | 64.62 | 17.57 | 32.86 |
| 🟥 | 1 | CPU Usage (Average/Rate) | Percent | 41.2 | 69.3 | 11.72 | 27.78 |
| 🟧 | 2 | CPU Usage (Average/Rate) | Percent | 41.9 | 66.32 | 11.96 | 28.16 |
| ■ | 3 | CPU Usage (Average/Rate) | Percent | 39.96 | 65.95 | 11.77 | 27.91 |
| 🟩 | vh1.foxgal.com | CPU Usage in MHz (Average/Rate) | MHz | 4813 | 7820 | 1669 | 3492.68 |
| 🟪 | vh1.foxgal.com | CPU Usage (Average/Rate) | Percent | 40.21 | 65.33 | 13.94 | 29.18 |

Recent Tasks

| Name | Target | Status | Initiated by | Time | Start Time | Complete Time |
|---|---|---|---|---|---|---|

# Server Consolidation



The basic idea

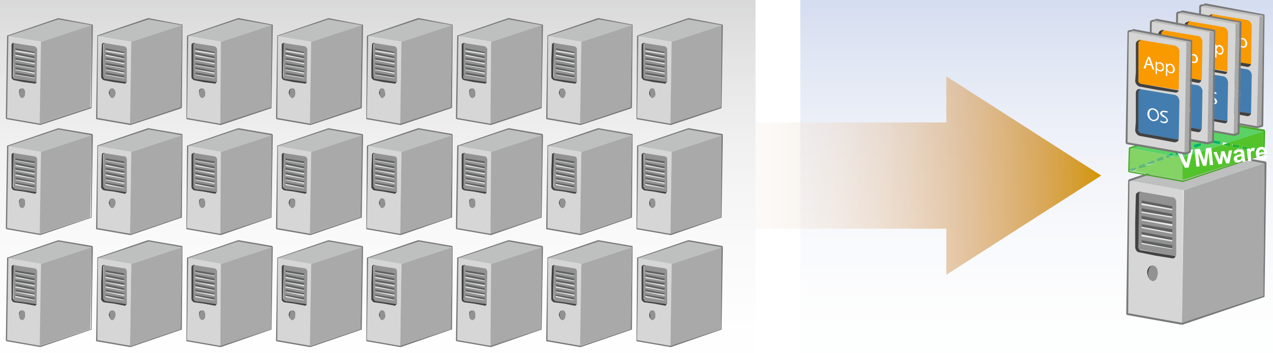
- e scales with #VMs

Consolidation Ratio's imply cost savings

- 2x more VMs
- 2x fewer machines
- 2x + c less power

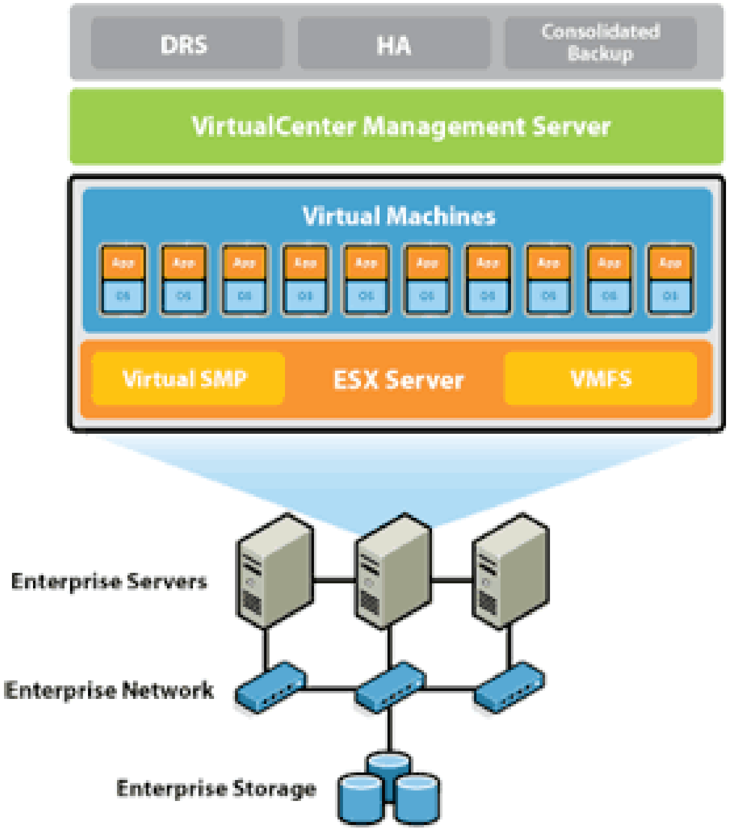
# Consolidation In the Real World

| | BEFORE | AFTER |
|---|---|---|
| Servers | **1,000** | **80** |
| Storage | **Direct attach** | **Tiered SAN and NAS** |
| Network | **3000 cables/ports** | **400 cables/ports** |
| Facilities | **200 racks** | **10 racks** |
| | **400 power whips** | **20 power whips** |



*Note Change in Storage Configuration...*

# Brainpower Consolidation



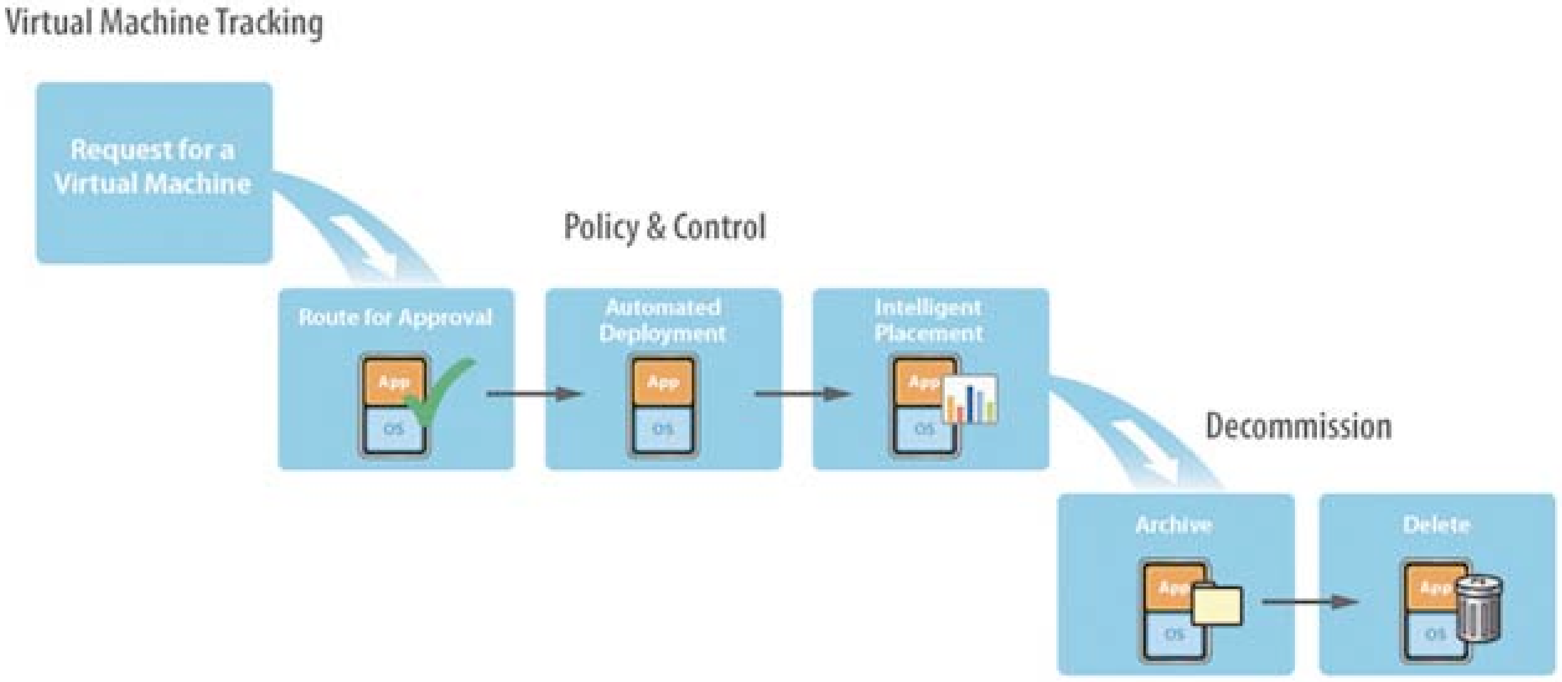
Who manages your virtual infrastructure?

- IT guy?, network guy?, storage guy? Applications guys?

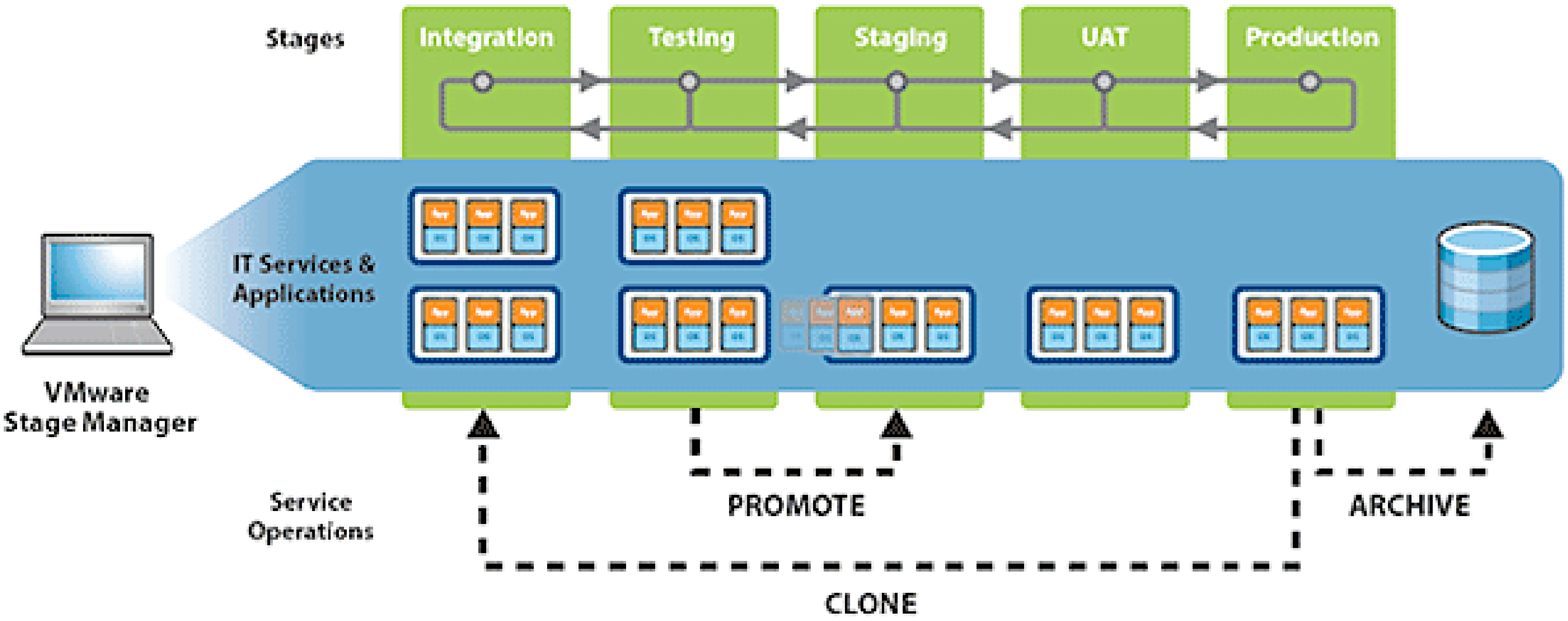
Power in unified/simplified abstractions

- Virtual switches, VMFS
- OS independence => orthogonal management plane

# IT Process Automation (Lifecycle Managment)



# IT Process Automation(Staging)



# Overview

Virtualization Overview

From Virtual Machines To Virtual Infrastructure

Security Challenges and Opportunities

Emerging Technologies

# Traditional IT

## Static

- Homogenous, static, manual ad-hoc

## Slow Scaling

- # machines limited by capital equipment budget
- growth limited by physical/process constraints

# Diversity

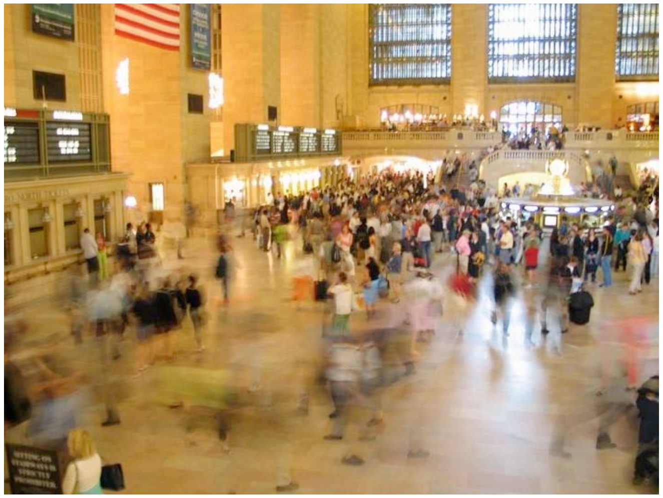
Support wide range of uses

- Ease upgrade cycle (multiple OS versions concurrently)
- Different OS versions for testing
- Task specific VMs (build environment/demo), VMs as a script (infrequent use/specialized)
- Application specific OSes
- OS independant primitivites (HA, backup)

Kills traditional management infrastructure

- N versions of everything
- Lack of admin control
- Infrequent use provides less incentive to break stuff with maintenance

# Transience



## Normal Environment

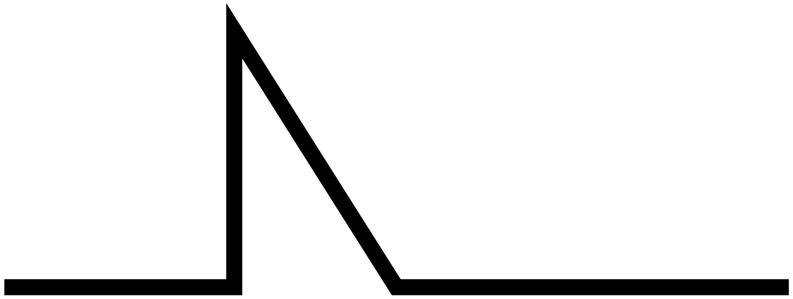
- Machines generally on
- Relatively low churn
  - exceptions: laptops,dual boot
- Unused machines cost money

## Virtual Environment

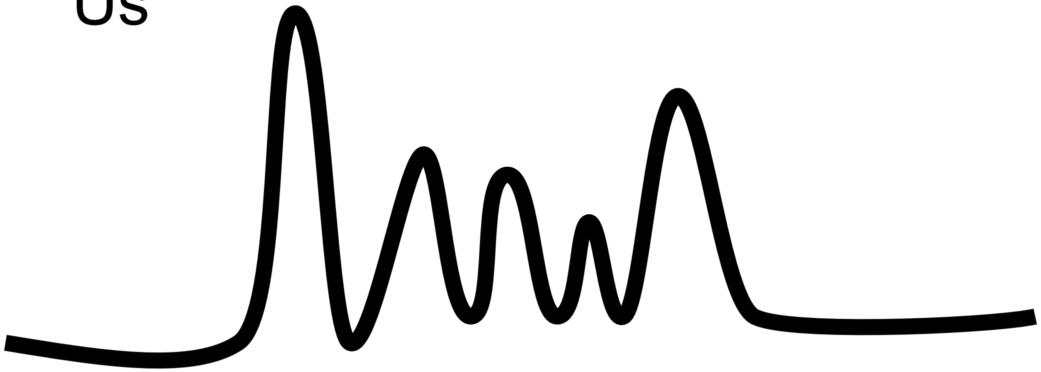
- Snapshots, suspend/resume
- Rapid/low cost VM creation
- Potentially large transient population
- Unused machines free

# Engineering Network Infection Profile(Circa 2003)

Traditional Environment



Us



# Impact of Transience

Loss of visibility

- Patch updates
- Vulnerability scans
- Infection symptoms

Startup Lag (unhappy AV)

- Patch updates
- Virus Scans

Time Dilation (unhappy people)

- Key Aging
- Password Expiration

# Coping With Transience

Reconsider how and when to...

- Patch, Scan, Update

Lots of points in design space...

- Loopback file system (offline/out of band)
- Sandbox (online/in-band)
- Scheduling
  - Periodic
  - On demand (e.g. NAC)

# Mobility

## Traditional Environments

- Static hosts the common case (Desktop/Data Center)
- Mobility = Laptop (or someone switches offices)
- Most machines live in a place (port 5 on building B switch), with a person

## Ownership & Accountability Implicit

- Who can get new machines, Who approves it, who owns it
- Who do I blame, where do I pull the plug

## Virtual Environments

- Copy VMs with scp, CIFS, NFS
- Put them on a USB drive
- Hot migrate them.

# Mobility in the network

Todays networks not built w/ mobility in mind

- Static firewall rules/ACLs, etc. don’t like mobility
- Stateful security elements (DPI,NIPS) don’t like mobility

Exploit VLANs for mobility?

- Topologies get stupid quickly

Alternatives:

- Migrate per-connection state
- Migrate virtual enforcement elements
- Making routing smarter

# Mobility Ownership and Accountability

Lack of traditional identity

- Office #, port #, mac address

No intrinsic notion of ownership

- Box owner != virtual machine owner
- No ownership history
- When one VM goes bad, penalize whole box?

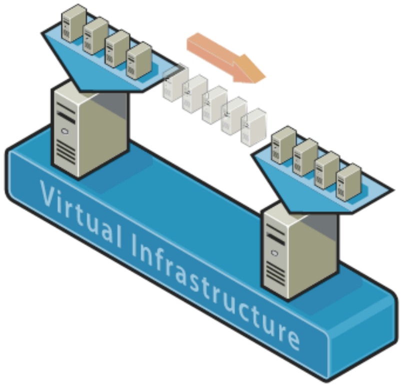
# Mobility and  the TCB

## Expanded/Fluid TCB

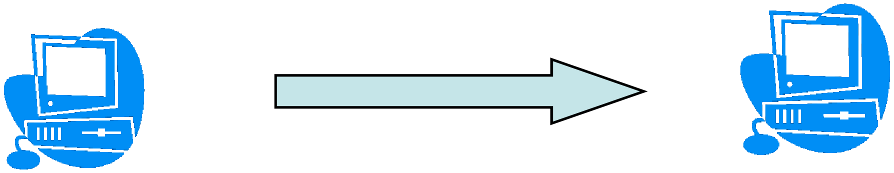
- Where has your VM been?

## Data Lifetime

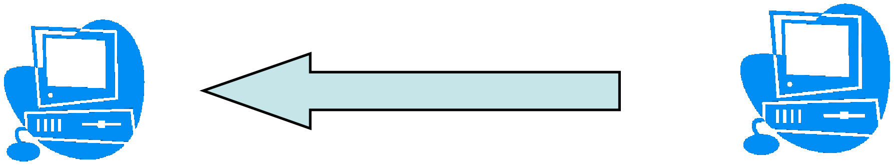
- Where did it get left?



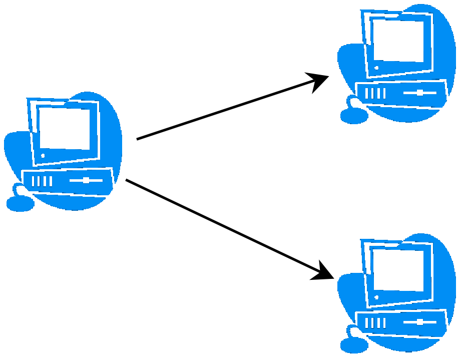
# Virtual Time



Traditional time: sequential and montonic



Virtual Time: Not always so monotonic



Virtual Time: Not always so sequential

# Virtual Time(Real Problems)

Many practical things don't like being rolled back

- AV Signature files
- Patches
- Firewall and Other network configuration state
- Access Controls
- Passwords/User account information

Potential Solutions

- VM config management
- Move state or entire function out of VM
  - Virtualization aware filesystems: Monotonic files
  - Extra-VM state store
  - Carry out certain tasks in seperate/dedicated VM

    *e.g. patch management*

# Virtual Time (math problems)

Many protocols assume a given transaction has not been seen before.

Unfortunately, we can’t rollback an attackers memory

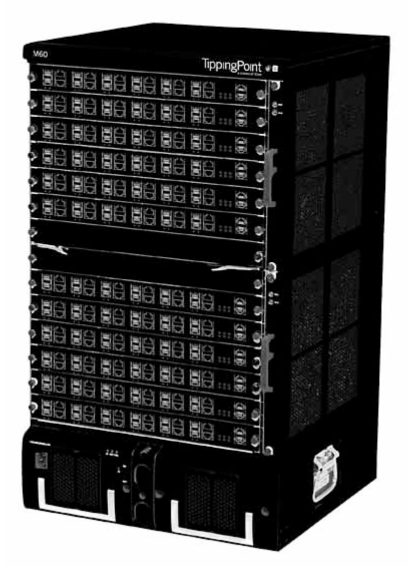
Trivial Example:  One-time passwords (e.g. S/Key), attacker can easily replay old passwords

# Virtual Time(more math problems)

Anything that relies on a “fresh” random number R breaks.

- Break a stream cipher (if R is the session key)
- Allow TCP hijacking (if R is the initial sequence number)
- Leak the secret signing key in DSS (if R is used to generate signatures)
- And more...

# Virtualizing Network Security (challenges)

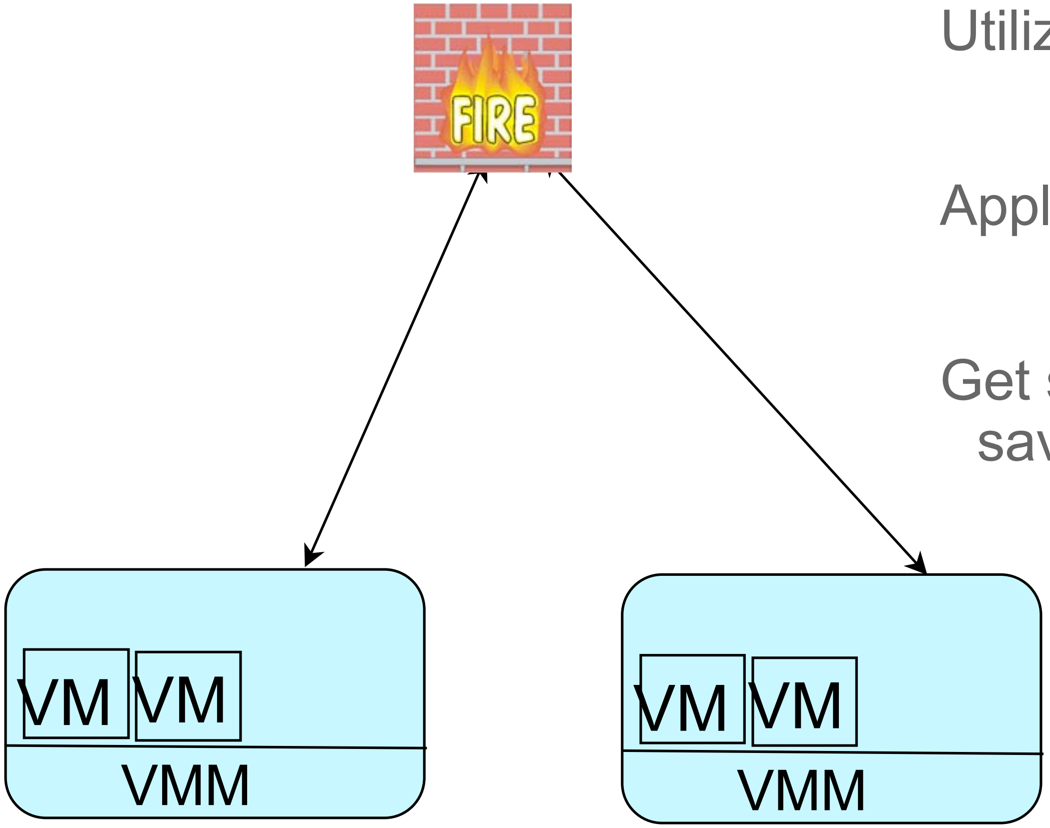


Dedicated hardware goes fast

- Does analysis scale out or up?

How do we deal with inter-VM traffic?

# Virtualizing Network Security (Opportunities)



Utilize commodity hardware

Apply more ubiquitously

Get scaling, fail over, power saving, etc. for free.

# Overview

Virtualization Overview

From Virtual Machines To Virtual Infrastructure

Security Challenges and Opportunities

Emerging Technologies

# Virtual Machine Introspection

Traditional Host IDS

- OK visibility (previously great...)
- Poor Isolation

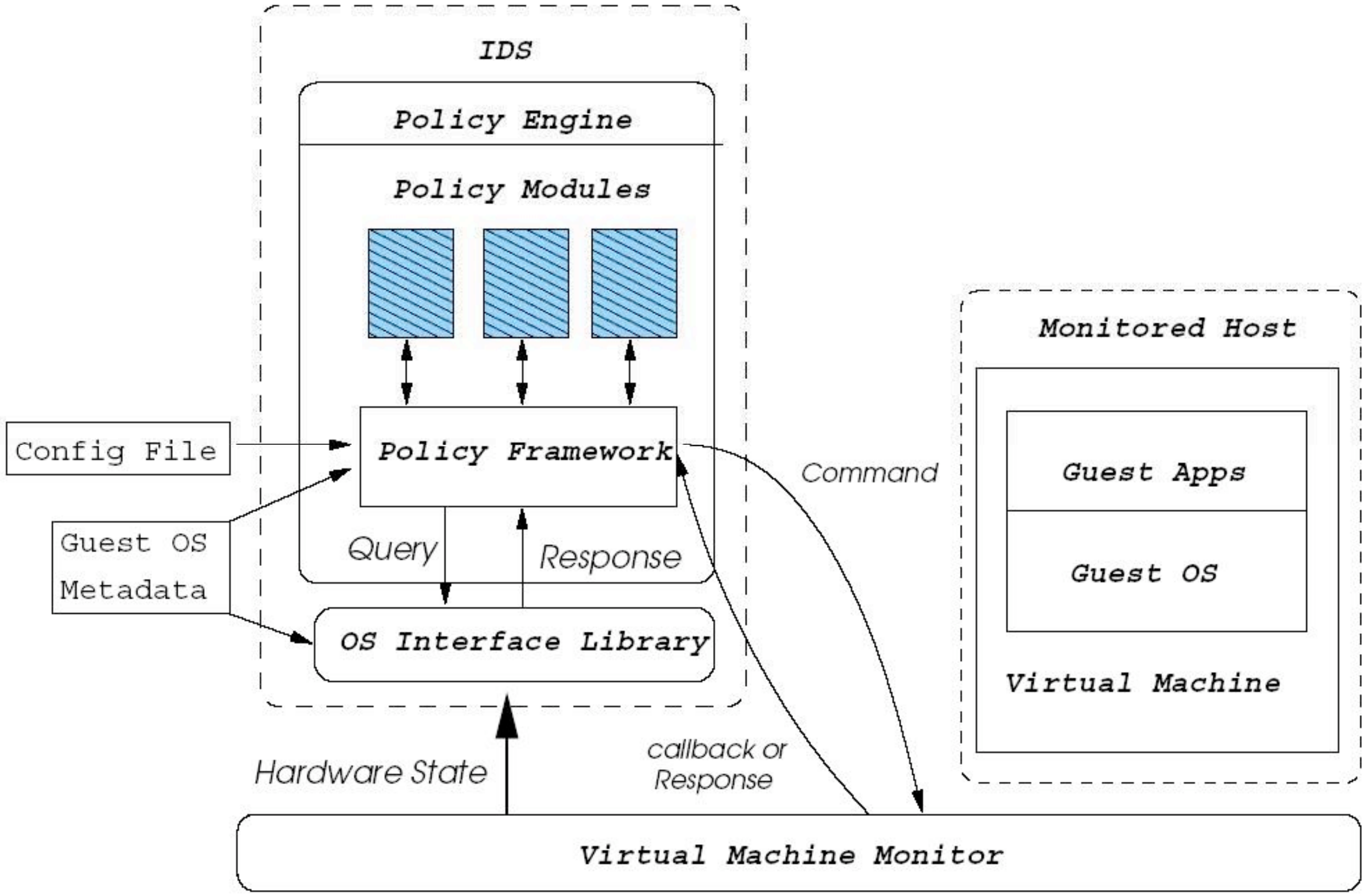
Traditional NIDS/NIPS

- Great Isolation
- Poor visibility

Virtual Machine Introspection (lift IDS out of Host)

- Very good isolation
- excellent visibility

# VMI Example (Livewire)



# Virtual Machine Introspection Benefits

Simplifies approaches like crossview detection

Eases opportunities for cooperation (see CloudAV)

Eliminates need for modifying OS (see patch guard)

Combine host and network knowledge:

- end-2-end crypto not your friend in the enterprise

# Not a Panacea

If something is compromised, soundness is out!

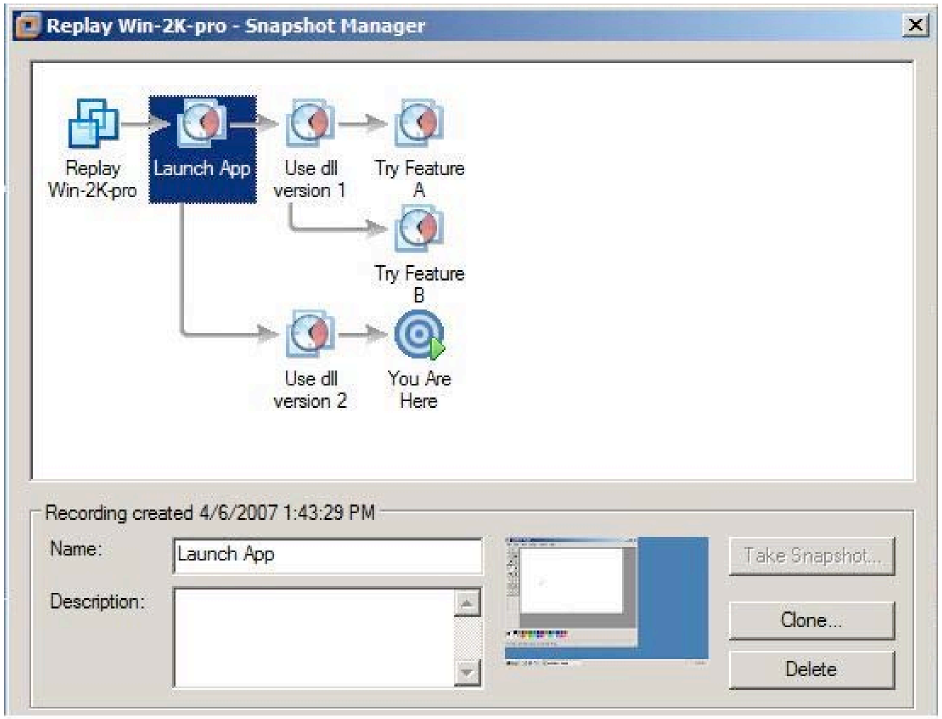
Defender has the upper hand, but its still just HID/HIP

- all the same tools
- many rehashes of old themes

General intuition about arms race

- Pin hardware invariants, work your way up.
- IDTR -> system call table -> system call text -> system call data structures ...

# Virtual Machine Record/Replay



Recording created 4/6/2007 1:43:29 PM

Name: Launch App

Description:

Basic Mechanism

- Capture Snapshot
- Replay all non-deterministic input
  - net, keyboard, timing

Capture all of execution

- about 56Kbps + 5-15% cpu

# What does replay give you

Complete dynamic execution state

- every memory location/register value/disk block and every step

Ability to decouple analysis from execution

- analyze now in parallel, later offline

# Decoupling Analysis and Execution (offline)

Completeness

- Full history for auditing, logging, forensics, analysis

Analysis can be arbitrarily expensive offline

Analysis can be done when needed/when possible.

# Decoupling Analysis and Execution (online)

Dawdle behind the running VM

- almost realtime IDS

Provide synchronization points (e.g. sync on output)

- Only need to synchronize for containment
- VM Rollback for remediation

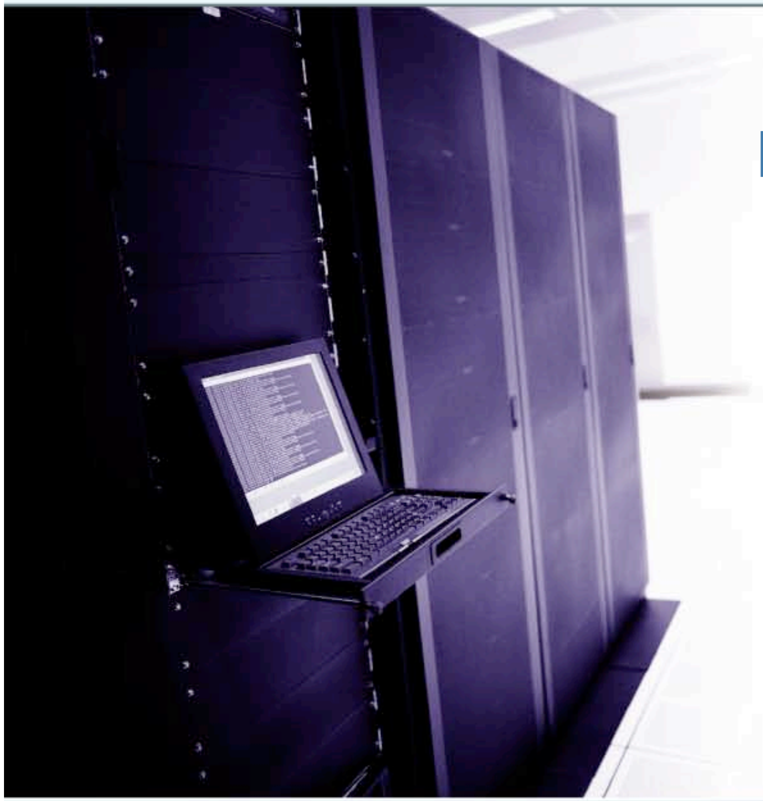
# Summary

Virtualization is becoming ubiquitous

Virtualization substantially changing the way we design systems, existing security architectures must adapt

Virtualization provides many cool new mechanisms and degree’s of freedom -- lots of space to innovate

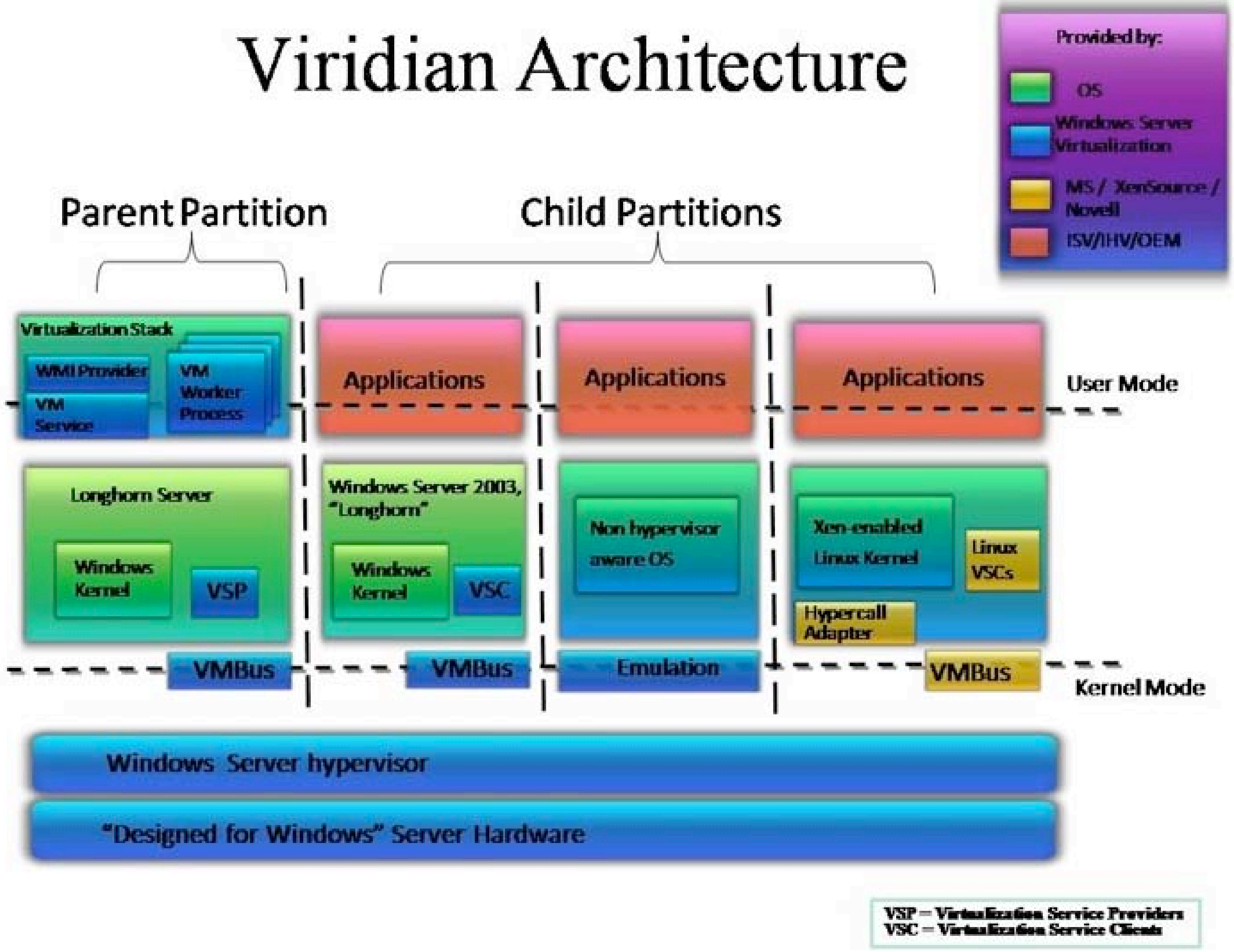
# Questions?



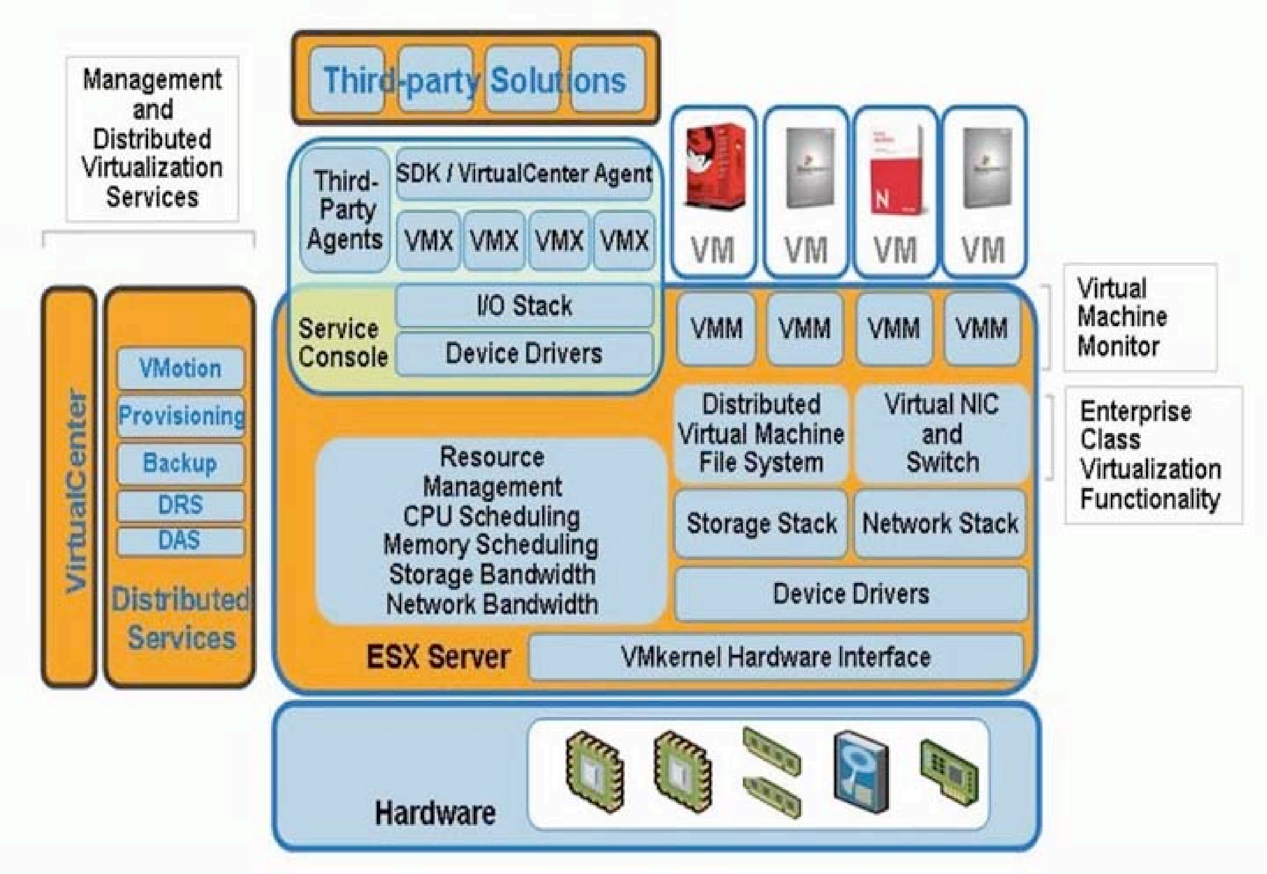
# Extra slides

# Architectures

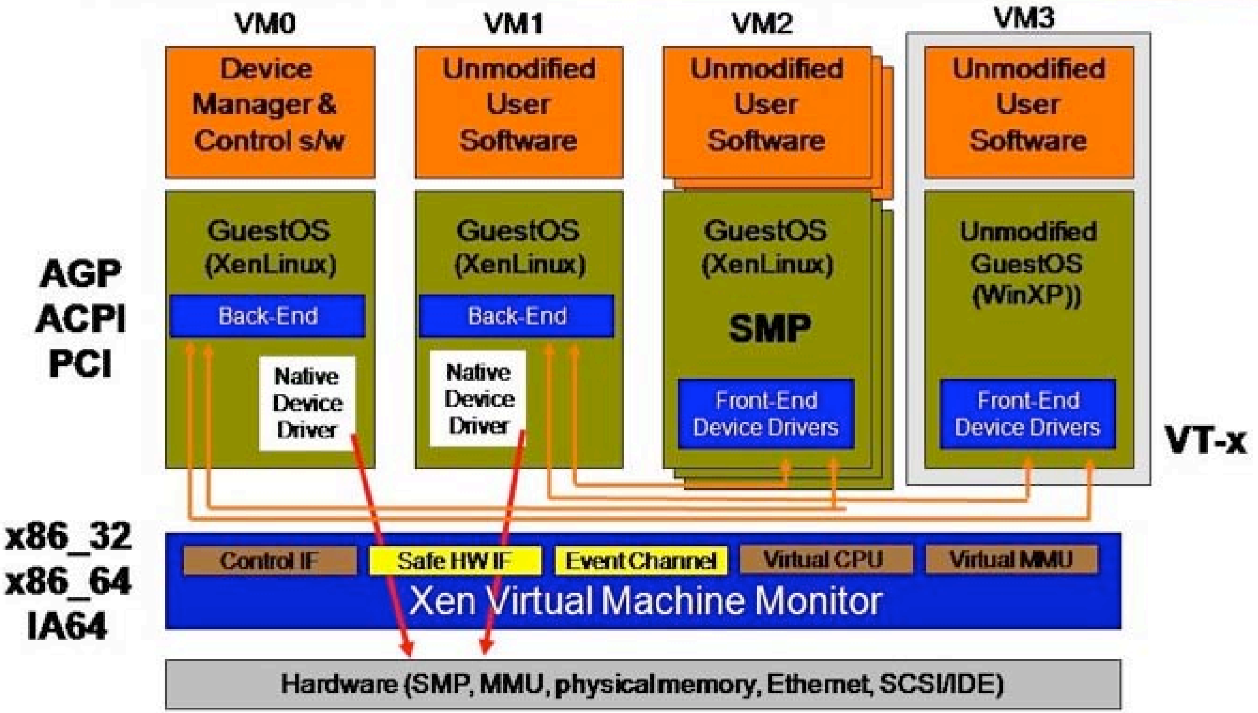
# Viridian Architecture



VSP = Virtualization Service Providers
VSC = Virtualization Service Clients


Management and Distributed Virtualization Services
Third-party Solutions
Third-Party Agents
SDK / VirtualCenter Agent
VMX
VMX
VMX
VMX
VM
VM
VM
VM
Service Console
I/O Stack
Device Drivers
VMM
VMM
VMM
VMM
Virtual Machine Monitor
VirtualCenter
VMotion
Provisioning
Backup
DRS
DAS
Distributed Services
Distributed Virtual Machine File System
Virtual NIC and Switch
Enterprise Class Virtualization Functionality
Resource Management
CPU Scheduling
Memory Scheduling
Storage Bandwidth
Network Bandwidth
Storage Stack
Network Stack
Device Drivers
ESX Server
VMkernel Hardware Interface
Hardware

# Xen 3.0 Architecture



# Assurance

# What matters for assurance

## Attack Surface

- Code size is overrated
- VMMs lend themselves to narrow/stable attack surface (so far)

## Unhealthy fixations on assurance

- 1 remote exploit in XP since SP2
- In real world, airgaps aren’t generally airgaps
- likely that sky isn’t falling
- Amdahl's law, not just for performance
- Real world problems mostly stem from misconfiguration
- Security just one parameter in the design space