Computing in High-Energy-Physics: How Virtualization meets the Grid

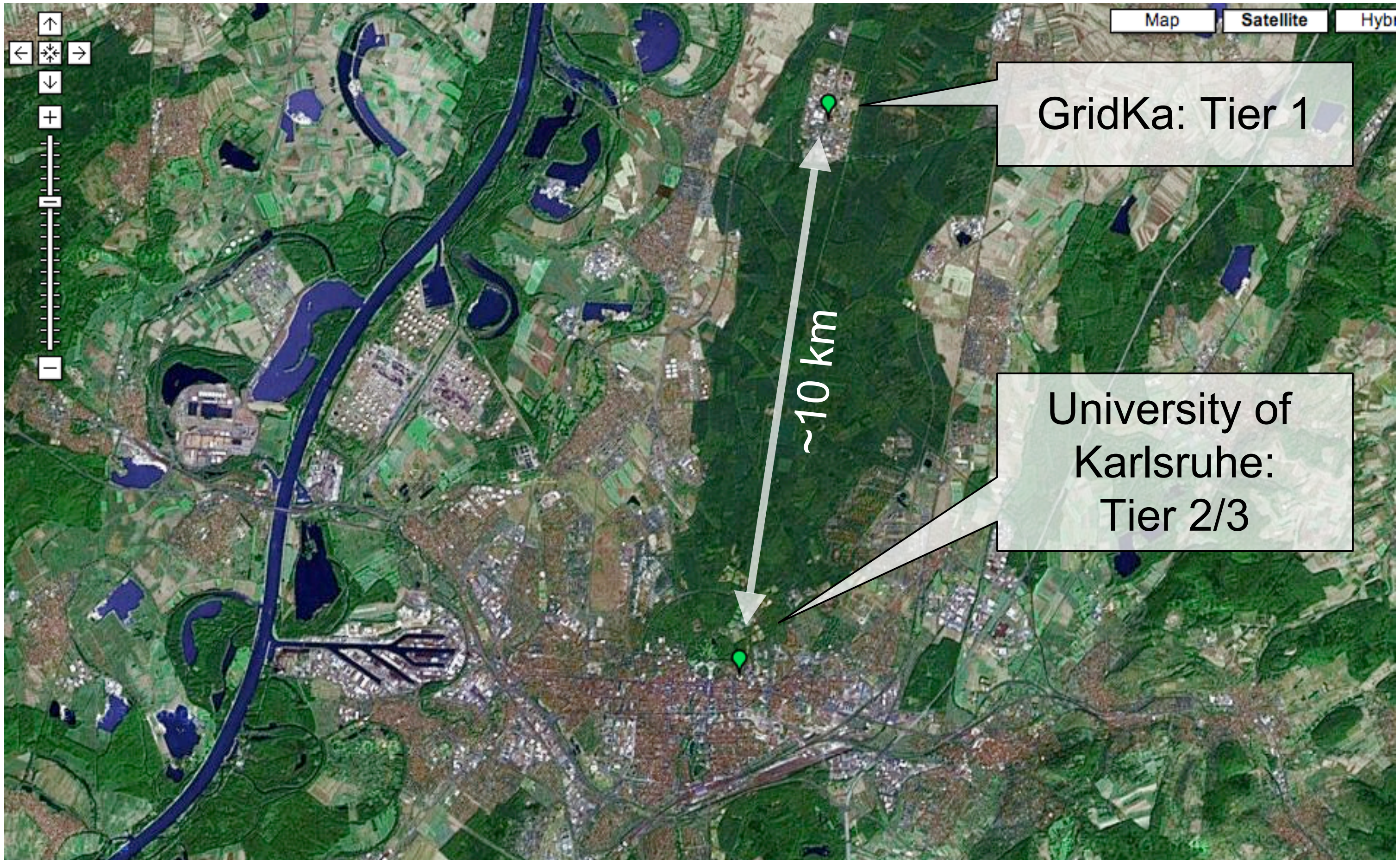
Yves Kemp

Institut für Experimentelle Kernphysik
Universität Karlsruhe

Outline:

- Problems encountered at Karlsruhe Tier 2/3:
  - Recycling of old hardware for Grid services
  - Different user groups with diverging OS requirements
- Virtualization techniques: Overview
- Usage of virtualization on the Grid:
  - Server consolidation
  - Horizontal partitioning of clusters
- Conclusion and outlook

2x Karlsruhe-Grid:



Tier 2/3 site at the University of Karlsruhe

- 30 Computing nodes
- 20 TB on file servers
- 100 Mbit/Gbit network

- 3 local user groups
  - CDF (20 users)
  - CMS (16 users)
  - AMS (6 users)
- Grid users through middleware:
  - Mainly CMS
  - Some CDF users (GlideCAF)

Problem: Site-Wide-Services

- LCG middleware
  - Computing Element
  - Storage Element
  - Monitoring Box
  - (User Interface)

  → Provide access to Cluster
- Two sites:
  - Production and testing
  - Six different computers minimum
- CDF GlideCAF
  - SAM-Station
  - GlideCAF
- Two more computers
- Total of 8 machines
- No heavy load on them
- "Recycling" of old machines:

→ Difficult to maintain

Problem: Different user groups

- CMS: Software requires SLC 3.0.X
- CDF: SL Fermi 3.0.X recommended
- AMS: Can easily recompile their software on different platforms
- gLite middleware: SLC 3.0.X recommended

- Now: Compromise possible: SLC 3.0.6 32bit
  - AMS could benefit from 64bit
- Future: Diverging needs:
  - e.g.: CMS SLC4, CDF SLC3
  - e.g.: CMS needs both SLC3 and SLC4
  - e.g.: Some need 32bit, other 64bit.
  - Sharing with other groups using modern distributions

Virtualization:

One possible answer

Virtualization: Products

Many virtualization products exist:

Win4Lin
Virtual Computing From Desktop to Data Center

Xen™
The Xen™ virtual machine monitor

Linux VServer

QEMU
open source processor emulator

vmware®
AN EMC COMPANY

parallels

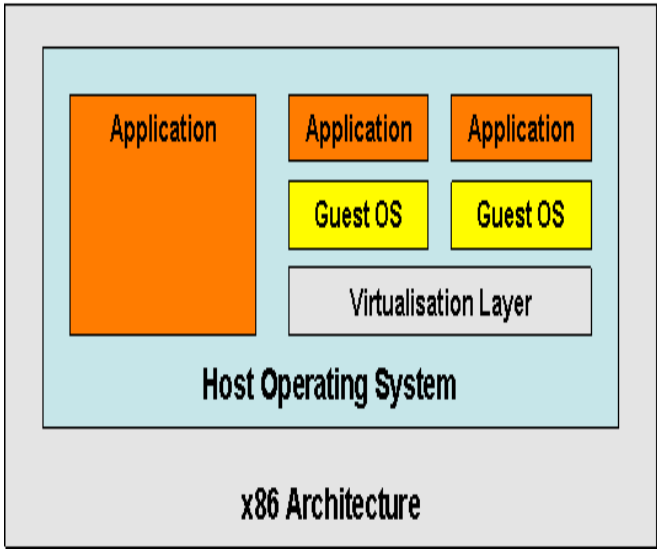
OpenVZ

Microsoft
Microsoft Virtual Server

and many more …

Virtualization – VMware GSX

Full Virtualization, e.g. VMware GSX

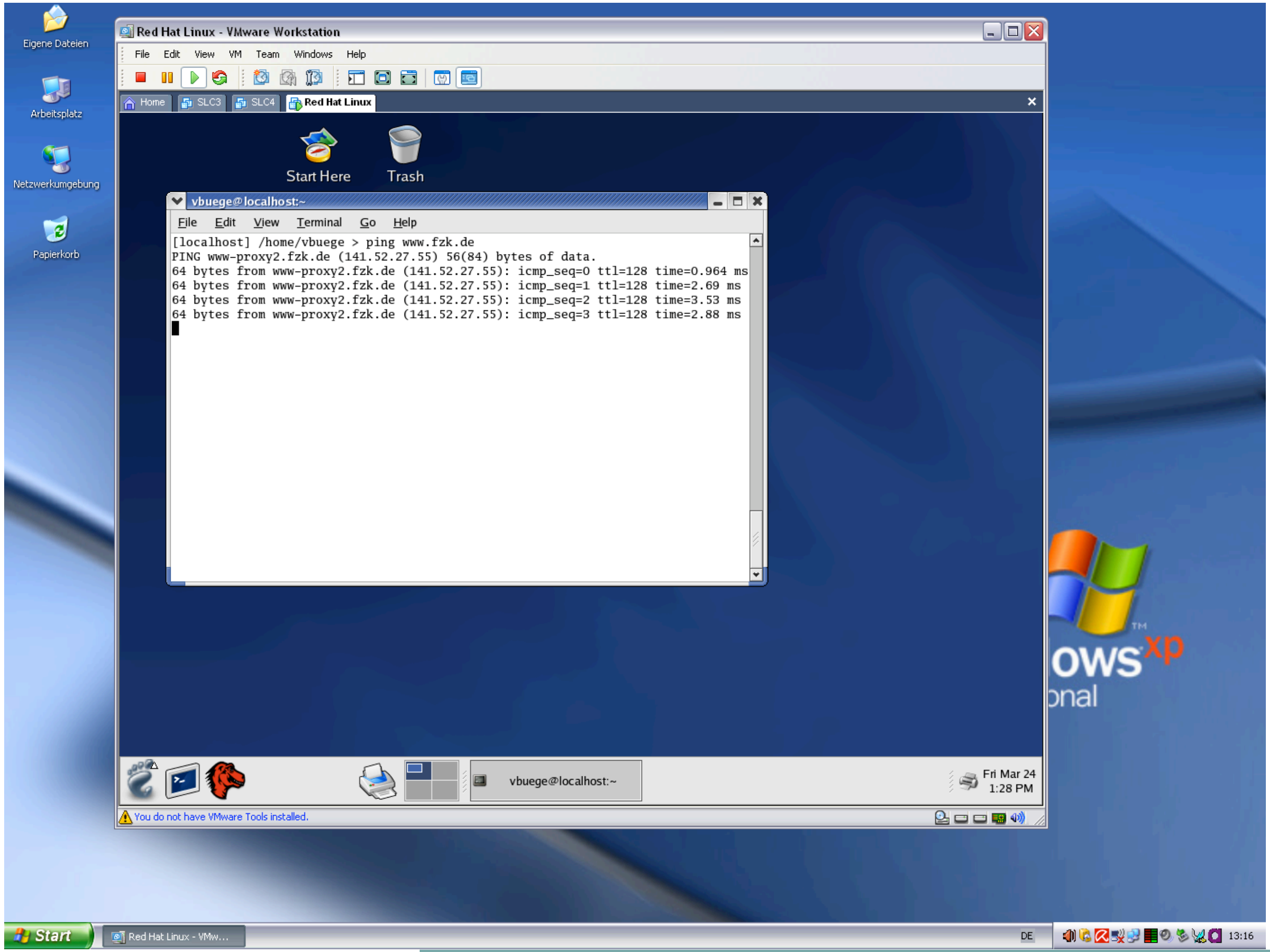


Schematic overview of
VMware GSX-Server

- The host OS emulates all hardware components except for the CPU for the VM
  → VM becomes independent from host configuration and can be used on different host systems
- VM is stored and run in files
- VMs contain native OS and are completely isolated …

… but such hardware emulations cost performance

Virtualization – VMware GSX



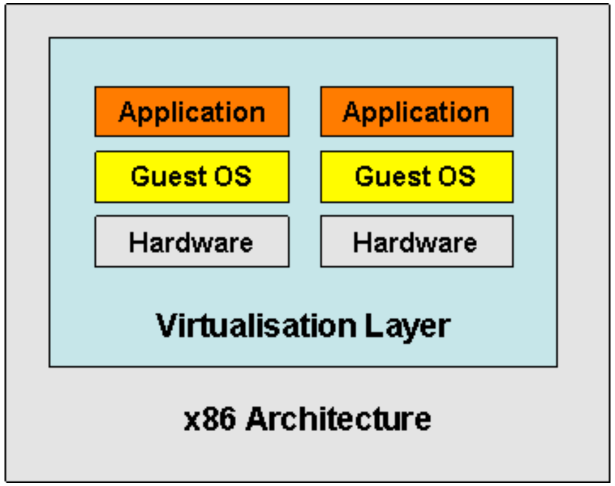
WindowsXP host OS with a *ScientificLinux Cern 3* VM

Used as User Interface for people with Windows Laptop

Virtualization – VMware ESX

Full Virtualization, e.g. VMware ESX

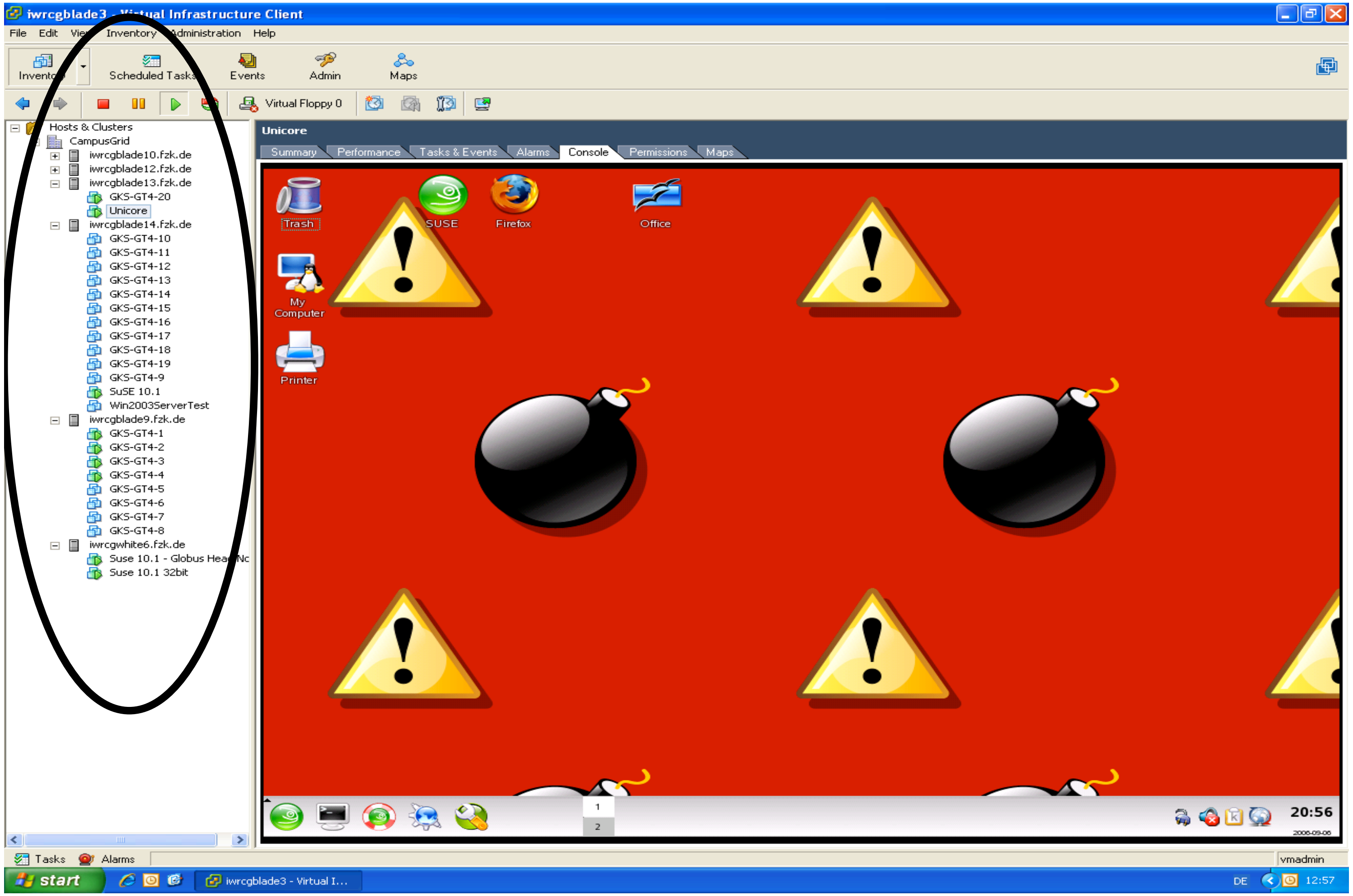
- Virtualization Layer is directly installed on the server hardware
- It is optimized for some certified hardware components
- Provides advanced administration tools



Schematic overview of VMware ESX-Server

- Allows emulation of hardware components for the VMs at near-native performance
- Provides features like memory ballooning, over-commitment of RAM, live migration …
- Supports up to 128 powered-on Virtual Machines
- Relatively expensive

Virtualization – VMware ESX



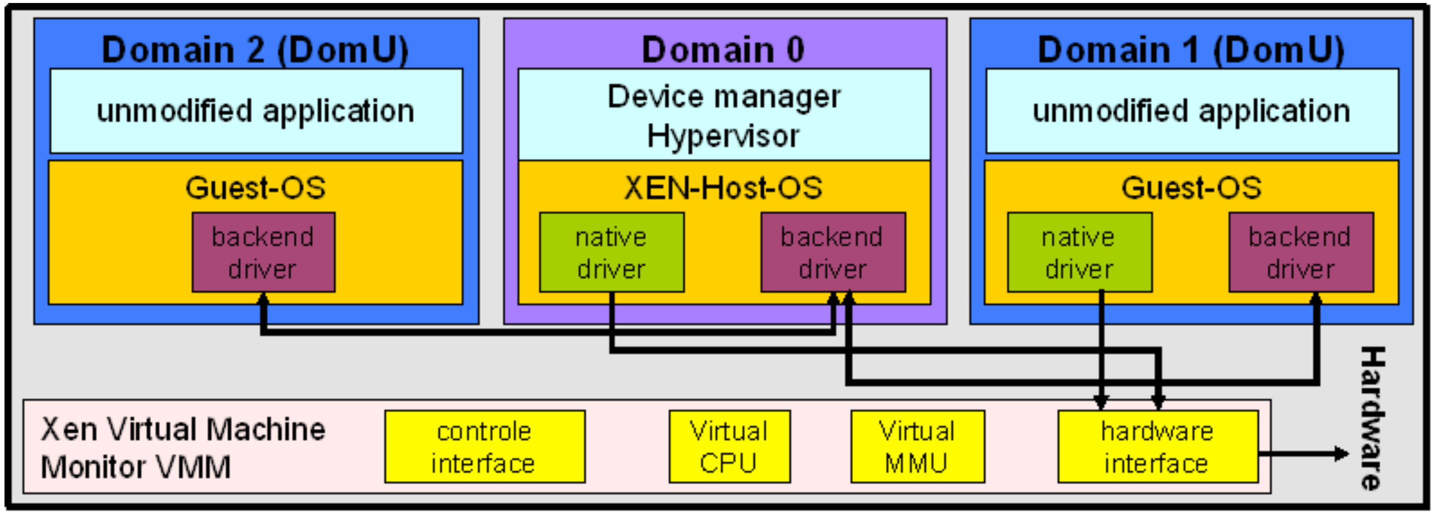
ESX Server on a Blade Center

hosting virtual machines for a course at the GridKa school 2006

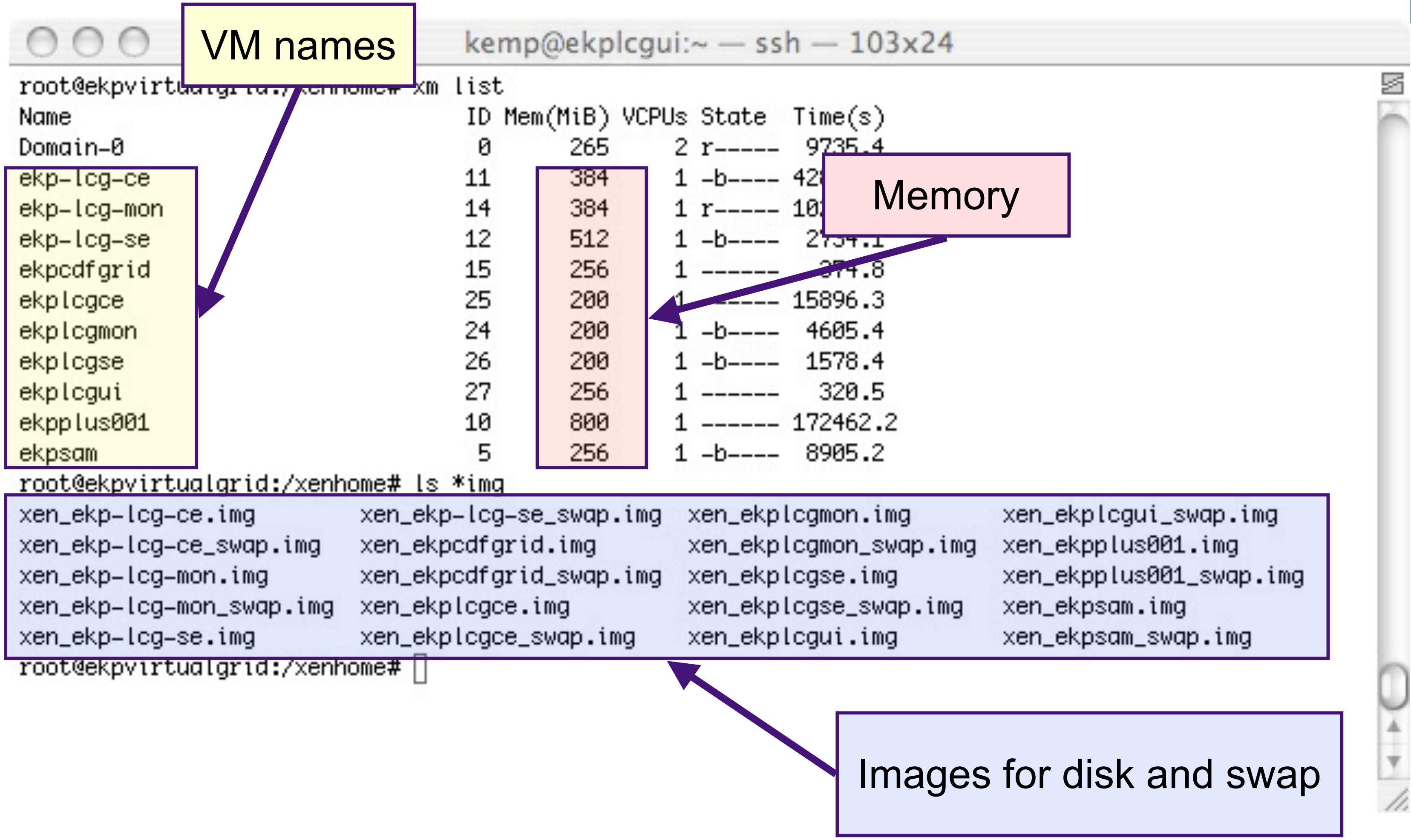
Virtualization: XEN

Para Virtualization, e.g. XEN

- Different hardware components are not fully emulated by the host OS. It only organises the usages → Small loss of performance
- Layout of a Xen based system: Privileged host system (Dom0) and unprivileged guest systems (DomUs)
- DomUs are working cooperatively!
- Guest-OS has to be adapted to XEN (Kernel-Patch), but not the applications – this changes with processors supporting virtualization



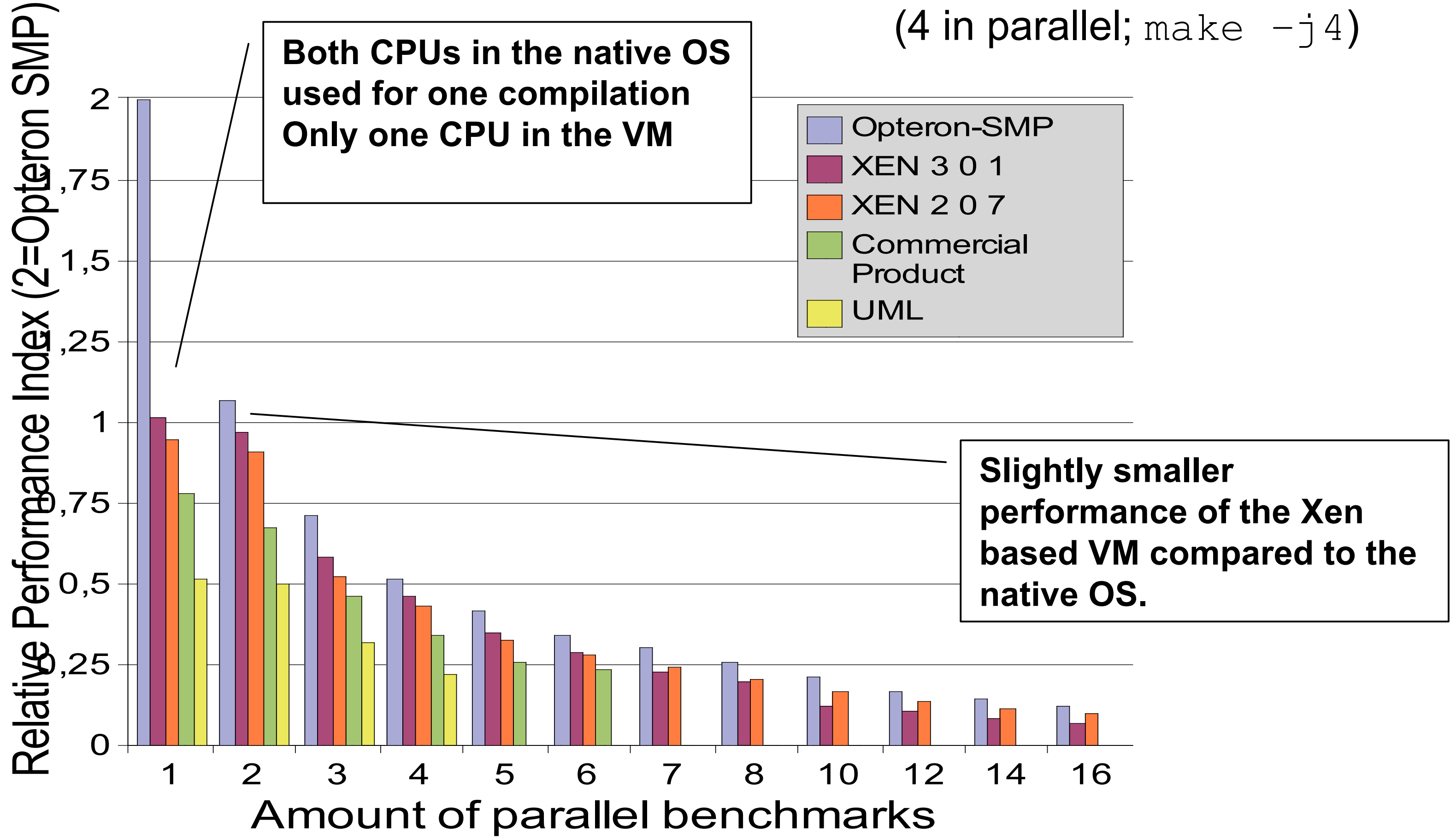
XEN in action:



Performance comparison

Standard application benchmark: Linux kernel compilation

(4 in parallel; `make -j4`)

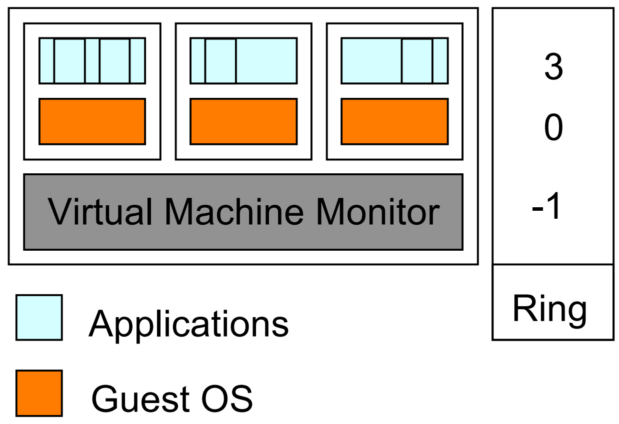


Virtualization – Hardware

New processor generation has extension for virtualization,
e.g. Vanderpool (Intel) and Pacifica (AMD)

- **Per definition, x86 platforms do not support virtualization**
- **OS is executed in Ring 0, Applications in Ring 3 – What about VMM?**

- **The new processor generation provides a Ring -1 for the VMM**
- **Guest OS is executed in Ring 0 and moderated by privileged Virtual Machine Monitor**
- **Application remains in Ring 3**
- **Overhead for translation reduced**

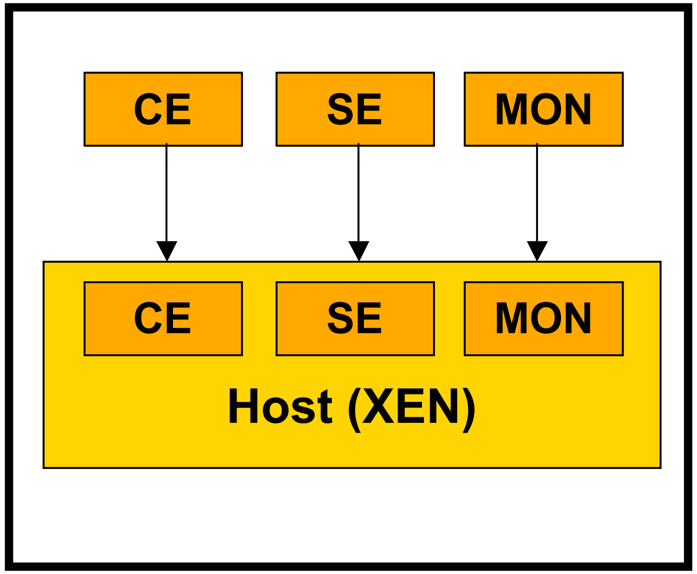


First application: Server Consolidation

The Grid site-wide services:

- for reasons of stability: recommended to run each service in an isolated OS instance.
- varying load on the different machines
  - → no full usage of resources
  - → “recycling” of older machines leads to a heterogeneous hardware structure

→ high administrative effort for installation and maintenance of the system



Virtualization of these machines leads to one single machine to be maintained and to homogenous OS installations

Realization at the EKP

- host system with Virtual Machine Monitor (VMM) Xen (3.0.2)
  - AMD Dual Opteron with 4 GB RAM, 600 GB RAID 10
  - OS: Debian stable (with 2.6 kernel)
- Guest systems:
  - gLite production environment: CE, SE and MON on SLC 3.0.8
  - gLite test environment: CE, SE and MON on SLC 3.0.8
  - CDF Grid: Two machines on SL fermi 3.0.5
- All environments fully integrated into the batch and storage system

→Three separate Grid infrastructures and eight VMs running on one physical host

Contribution to eScience 06 conference:

V. Büge, Y. Kemp, M. Kunze, G. Quast

Application of Virtualisation Techniques at a University Grid Centre

Advantages of Server Consolidation

Advantages through virtualization:

- a reduction of hardware overhead : Only one single high-performance machine needed for the complete LCG installation including a test WN

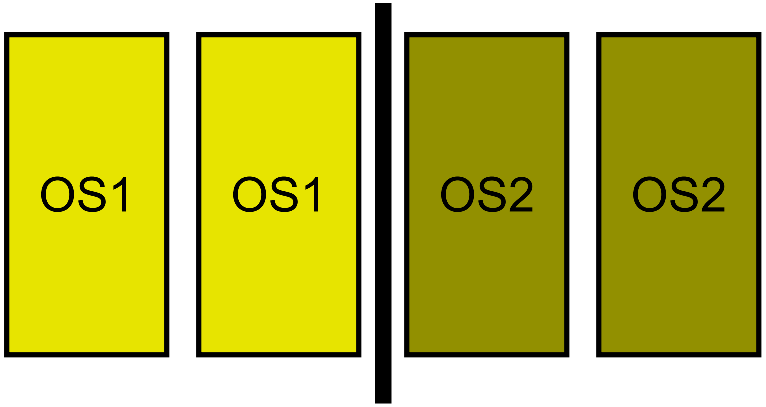
  →cheaper and easier to maintain
- easy and fast setup of basic OS by copying VMs image files
- possibility of migrating VMs to other machines and backup
- cloning of VMs before upgrades of LCG to enable tests
  - → less service interrupts and a more effective administration
- balanced load and efficient use of the server machine
  - → interception of CPU peaks

Second application of virtualization

- Encountered problems at a computing center:
  - Worker nodes need dedicated OS as middleware is installed on them
  - Typical institute: Different groups need different OS
  - One group might even need different OS because of different software versions
  - Computing cluster: Shared between local users and grid users: Want to enhance security and hide local information to grid users
  - New hardware but old OS needed

→ Partition your cluster! But how?

Static (vertical) partitioning

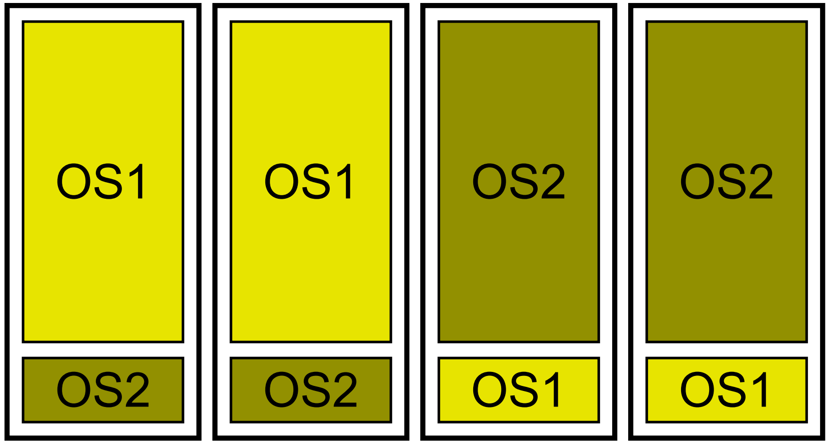


Example:

- 4 nodes, 2 groups
  - 2 nodes with OS1
  - 2 nodes with OS2
- Sharing common storage, network and control infrastructure

- Changes in the resource allocation difficult
- Old OS on new hardware problem persists
- No real resource sharing possible

Dynamic (horizontal) partitioning



- All nodes have two OS running all the time
- The OS needed gets all CPU and RAM resources

- Sharing all resources

- Dynamic and fast changes in resource allocation
- Only host OS must fit the hardware
- Security and privacy through encapsulation

Using Virtualization

Performance considerations

- No noticeable performance loss due to virtualization:
  - Around 3-4% loss for CMS software
- Even performance gain is possible:
  - AMS group could benefit from 64 bit, but 32 bit common agreement
  - Galprop runs 22% faster in a virtual 64-bit machine than on 32-bit native system!

→ A overall performance gain can be possible
(at least no drastic performance losses)

Connection to the Batch Queue

Users do not login to the nodes: Using Batch Queuing Server!

Users are not to control the resources: Batch Queuing Server?

- The different VM running on one host are not independent:

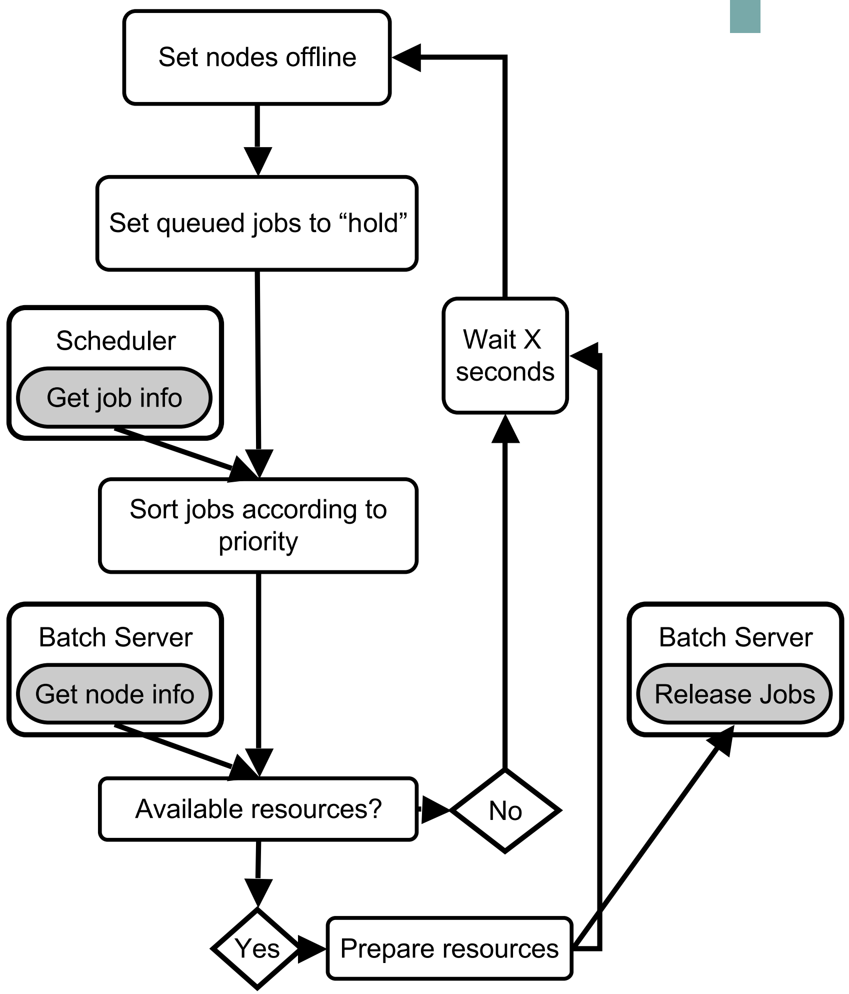
  They share the same resources
- The batch queue server must know about this sharing
  - Either natively
  - Or with the help of a separate program

Requirements of such a program:

- Independence of batch system server and scheduler:
  - No modifications
  - Flexibility
- Respect current policies:
  - Node occupancy
  - Prioritization

Prototype implementation

- Maui/Torque (used at EKP): Concept of grouping of resources not known.
- Daemon implemented in Perl language
  - Running on a test-system: 2 Dual Opteron machines simulate cluster with 19 nodes of two categories
  - Working stable
  - To be deployed on the production cluster
- Native implementation preferable…



Contribution to XHPC / ISPA 06
Virtualizing a Batch Queuing System at a University Grid Center
V. Büge, Y. Kemp, M. Kunze, O. Oberst, G. Quast

Conclusions & Outlook

- Variety of virtualization products exists, following different approaches
- User Interface as (Linux) Virtual Machine on Windows Host
- Server consolidation
  - Eases maintenance
  - Better usage of resources
  - Working stable at the EKP: three Grid sites in one box!
- Virtualized Worker Nodes:
  - Improved security through OS encapsulation
  - Optimal OS for every user, dynamic resource allocation
  - Good performance behavior
  - Integration into Maui/Torque: daemon running on a test system, to be installed on the production system at EKP in the next weeks
  - Optimal resource sharing among different groups in a institute
  - Enables resource sharing with other groups of a university
- Benefits from new x86 CPU with Virtualization support