Virtualization Infrastructure at Karlsruhe

HEPiX Fall 2007

Volker Buege[1),2)], Ariel Garcia[1)], Marcus Hardt[1)], Fabian Kulla[1)] ,Marcel Kunze[1)], Oliver Oberst[1),2)], Günter Quast[2)], Christophe Saout[2)], Lizeh Wang[1)]

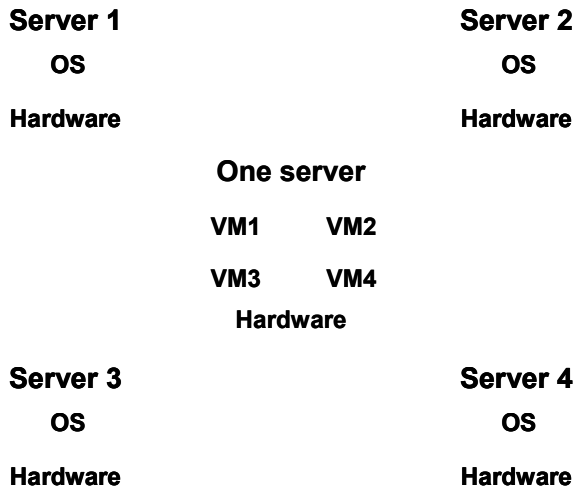
1) IWR – Forschungzentrum Karlsruhe (FZK)
2) IEKP – University of Karlsruhe

Summary

- Virtualization
- XEN / VMWare Esx
- Virtualization at IWR (FZK)
  - VMWare Esx
  - XEN
- Virtualization at IEKP (UNI)
  - Server Consolidation / HA
- Virtualization in Computing Development:
  - Dynamic cluster partitioning
  - Grid Workflow Systems on virtual machines (VMs)

Virtualization

- Possible Definition:
  - Possibility to share resources of one physical machine between different independent operating systems (OS) in Virtual Machines (VM)
- Requirements:
  - Support multiple OS like Linux and Windows on commodity hardware
  - Virtual machines have to be isolated
  - Acceptable performance overhead

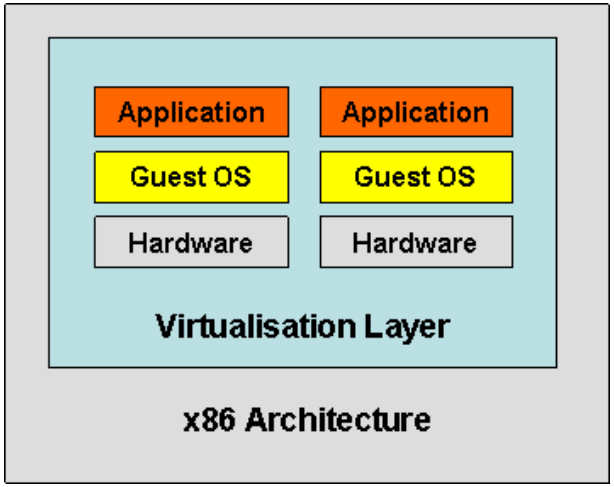


Why Virtualization

- Load balancing / Consolidation
  - Server load is often less than 20%
  - Economization of energy, climate and space
- Ease of Administration
  - Higher flexibility
    - **Templates** of VMs
      - Fast setup of new servers and test machines
    - Backups of VMs / **Snapshots**
    - Interception of short load peaks (CPU / Memory) through **Live Migration**
    - Support for older operation systems on new hardware (SLC 3.0.x)
    - High reliability through hardware redundance (Desaster Recovery)

VMWare ESX

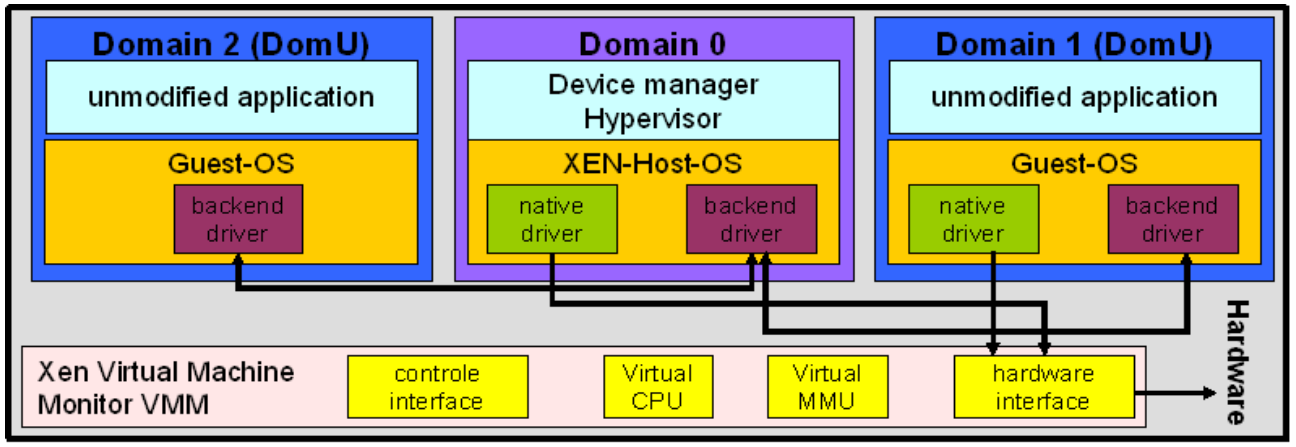
- Full Virtualization
- Virtualization layer is directly installed on the hardware host
- Optimized for certified hardware
- Provides advanced administration tools
- Near native performance while emulating hardware components
- Some Features:
  - Memory ballooning
  - Over-commitment of RAM
  - Live migration of VMs



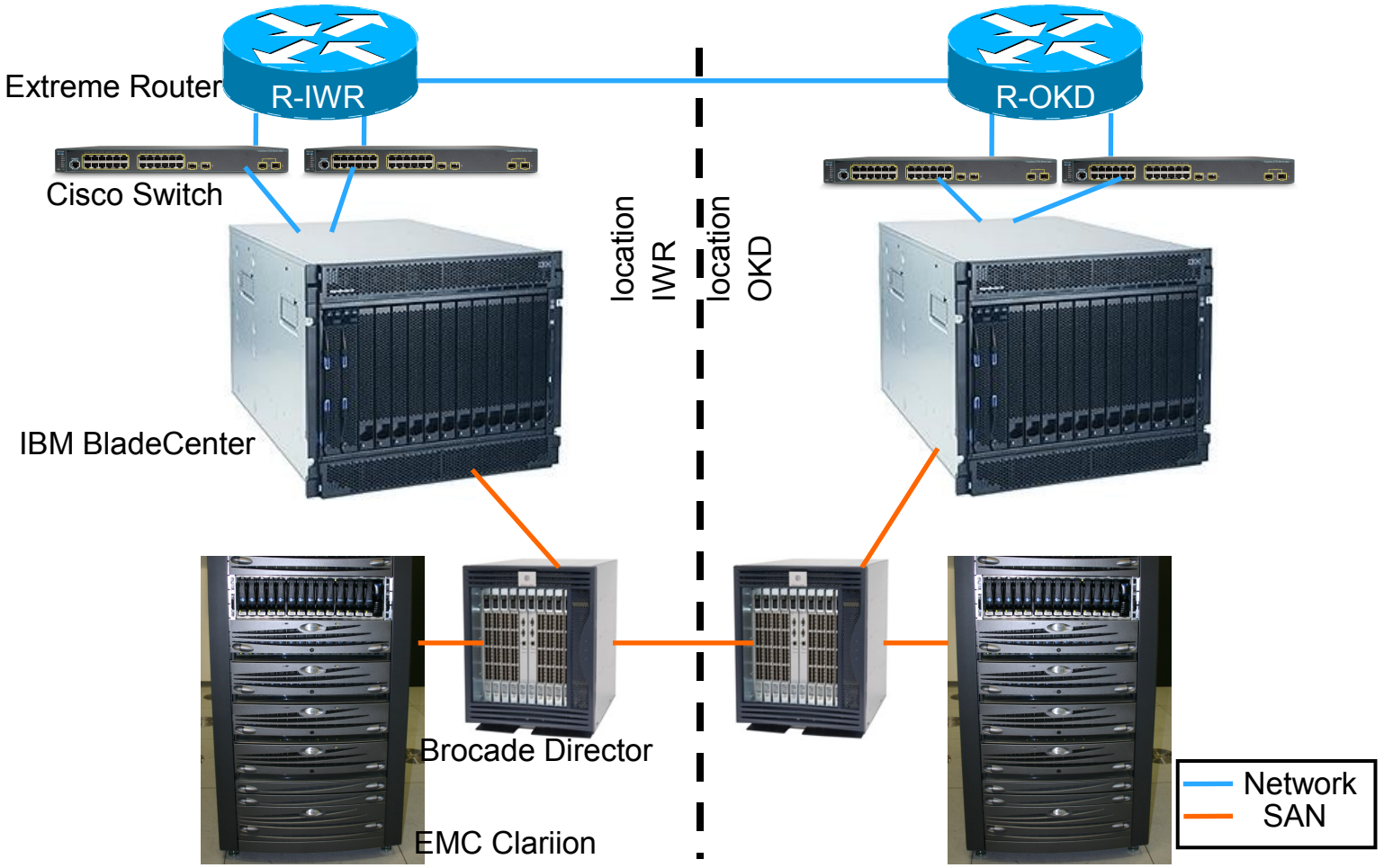
Schematic overview of VMware ESX-Server

XEN (Open Source)

- Paravirtualization (or full virtualization – CPU support needed)
  - Hardware is not fully emulated ⟹ Small performance loss
- Layout:
  - Hypervisor (xend) runs on the privileged host system (dom0)
  - VMs (domUs) work cooperatively
- Host and Guest Kernels have to be adopted in Kernel < 2.6.23. But most of common Linux distributions provide XEN packages (XEN-kernel / XEN tools)
- Some Features:
  - Memory ballooning
  - Live-migration



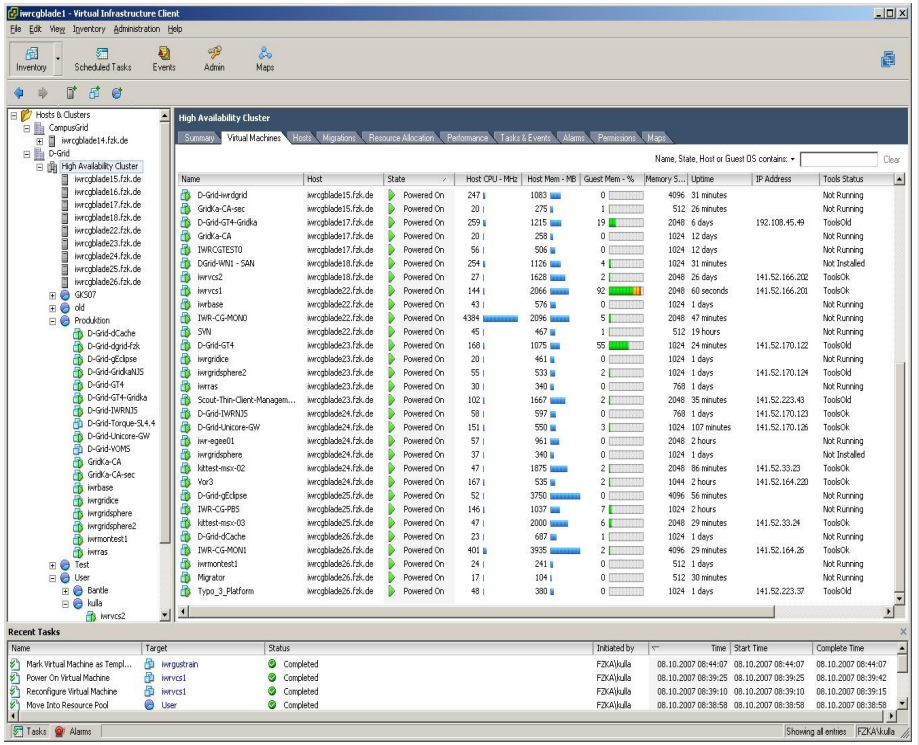
Virtualization at IWR (FZK) – The Hardware



by Fabian Kulla

Virtualization at IWR (FZK) – VMWare ESX

- Two ESX Environments:
  - Production:
    - 10 hosts (Blades) used
    - 30 VMs running D-Grid servers
    - 50 VMs others
  - Test:
    - 4 hosts used
    - 40 VMs
- ESX @ Gridka-School 07
  - ~50 VM for the workshops
    - gLite Introduction Course (UIs)
    - Unicore
    - ...

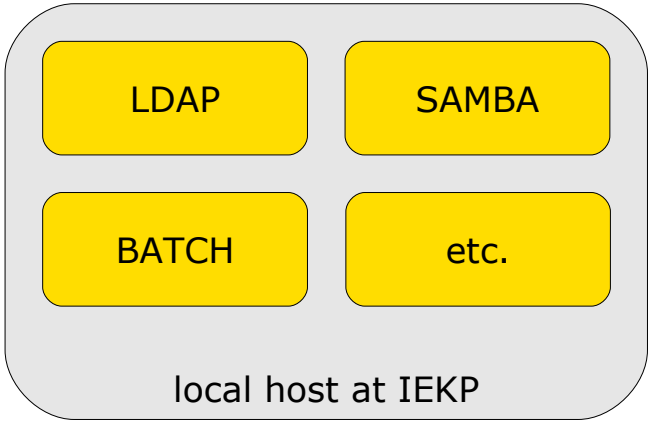


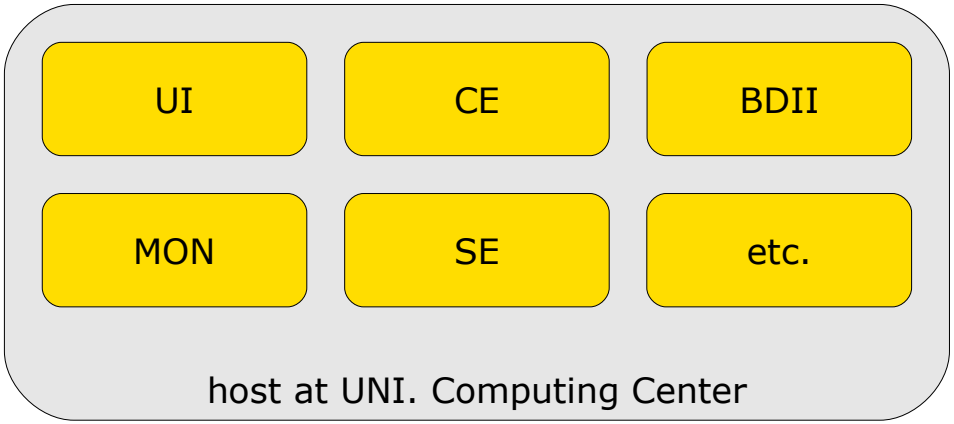
Virtualization at IWR (FZK) – XEN

- Running on the Blade Center and on older Gridka Hardware
  - ~30 Hosts: Xen 3.0.1-3, Debian stable
- Server infrastructure for different Grid-Sites:
  - Used in former Gridka-Schools
  - 16 VMs :D-Grid site infrasturcture production and testing
  - 14 VMs : gLite test machines
  - 21 VMs: int.eu.grid site infrastructure
  - 4 VMs : EGEE training nodes
- The int.eu.grid and D-Grid sites worker nodes are running on the Gridka Cluster
  - `/opt` is mounted via nfs containing the software required by the D-Grid and int.eu.grid virtual organizations (VO)

Virtualization at IEKP (UNI) – Server Consolidation

- Two main server infrasturctures:
  - local services (ldap, cups, samba, local batch system, ...)
  - gLite grid services of the UNI-KARLSRUHE Tier 3 site
    - moved to Computing Center of the University test cluster from local IEKP cluster

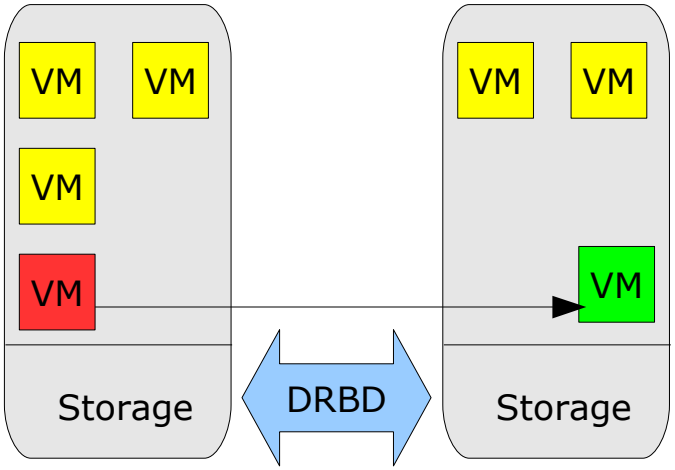




- Virtualization Hardware:
  - Two hosts (local IEKP):
    - AMD Athlon 64 X2 4200+
    - 6 GB RAM
    - 400 GB Raid10 disk space for VMs
  - Virtualization Portal at Uni. KA computing center:
    - 2x Dual-Core AMD Opteron
    - 8GB RAM
    - 400GB Disk Space

Virtualization at IEKP (UNI) – High Availability

- Combination of spare machines and SAN is an overkill if only a few critical services are hosted (example: IEKP)
- Solution should be without too much hardware overhead
- Possibility: Use two powerful host machines with same architecture in combination with a *Distributed Replicated Block Device* (DRBD) to mirror disk space between the machines (Raid 1 over Ethernet) for the VM images



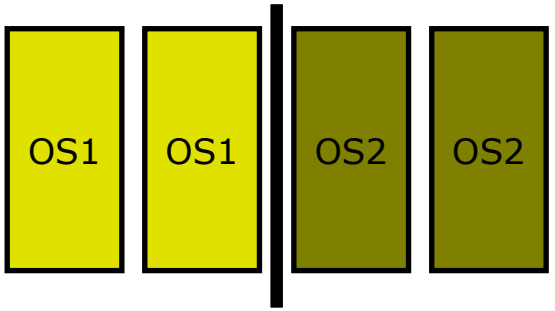
- In case of hardware problems or high load the machines can easily be migrated
- Not yet implemented:
  - Heartbeat: in case of complete hardware breakdown the machines will be restarted on the other host

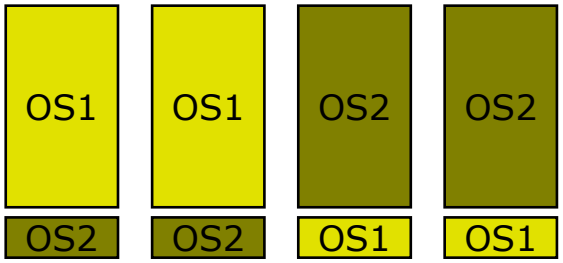
Dynamic Cluster Partitioning Using Virtualization

- Motivation:
  - Shared Cluster between several groups with different needs (OS, architecture)
    - Example: New shared cluster at the University of Karlsruhe computing center (in the end 2007)
      - ~ 200 worker nodes:
        - » CPU: 2x Intel Xeon quad core
        - » RAM: 32 GB
        - » Network: Infiniband
      - ~200 TB Storage:
        - » File system: Lustre
      - OS: Red Hat Enterprise 5
      - Shared between 7 different university institutes
      - IEKP relies on Scientific Linux 4 to run CMS experiment software (CMSSW) and to share the cluster in WLCG as the new UNI-KARLSRUHE Tier 3

Dynamic Cluster Partitioning Using Virtualization

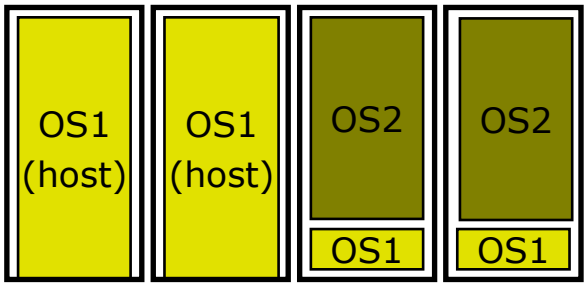
- Static partitioned cluster:
  - No load balancing between the partitions
  - changing the partitions is time consuming
- Dynamic partitioned cluster:
  - First approach (tested on IEKP local production cluster:
    - Using XEN to host the virtualized worker nodes
    - All needed VMs are running simultaneously. Minimum memory is assigned to the not needed VM
    - Managed by additional software daemon controlling batch system and VMs
    - Tests were run for several weeks on local IEKP cluster





Dynamic Cluster Partitioning Using Virtualization

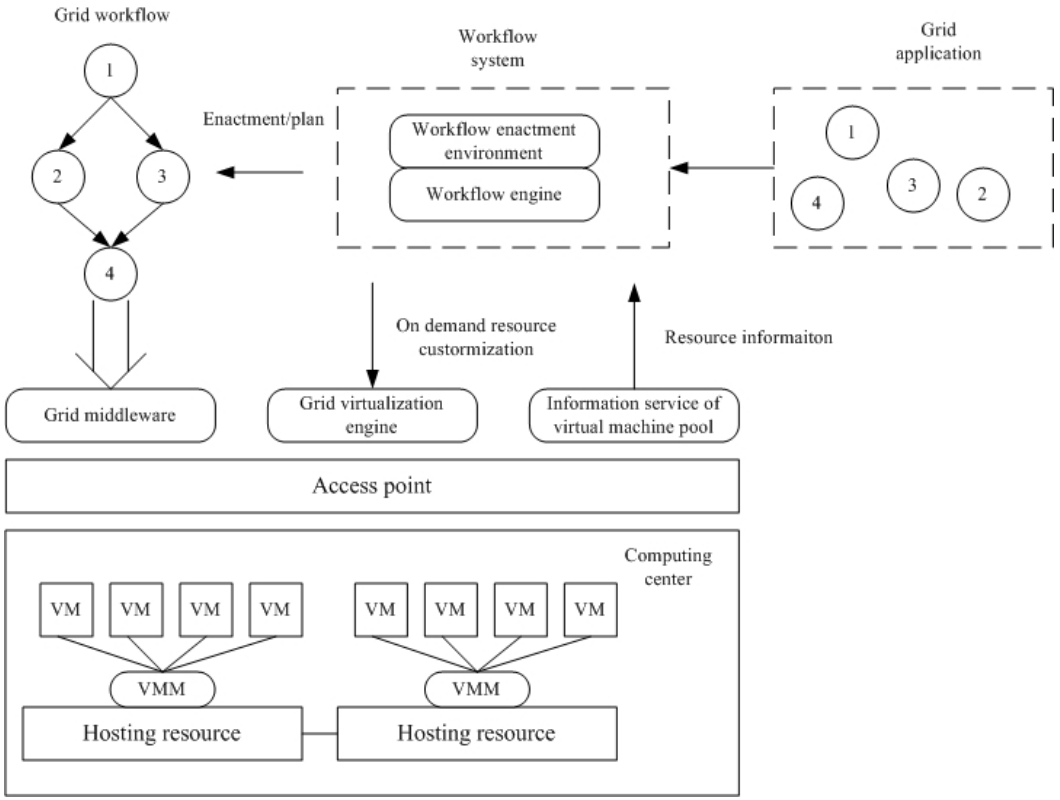
- New Approach:
  - Pre-configured VM Images
  - “wrap jobs” start the VM on the host worker node and pass the original job to the booted VM
  - Finishing jobs stop the VM after job output is passed out
  - Job cancels simply kills the VM instantly
- Main Advantages:
  - “Bad” grid jobs which may leave bad processes in memory are intrinsically stopped and modified VMs are removed after job
  - No software is needed everything is done by the batch system
  - VM Images could be deployed by the VO with tested software installation!!



- Performance:
  - measured a performance loss of about 3-5% with experiment software (CMSSW)
  - VM boot time: about 45s at the test cluster (old hardware)
  - the possiblity to participate whithin the shared cluster makes that acceptable

Grid Workflow Systems on Virtual Machines

- Grid Workflow?
  - Used to model Grid applications
  - Execution environment is a computational Grid
  - Participants across multiple administrative domains
  - heterogeneous resource types also in kinds of Virtualization (Vmware ESX + Server, XEN)



Lizhe Wang et. al

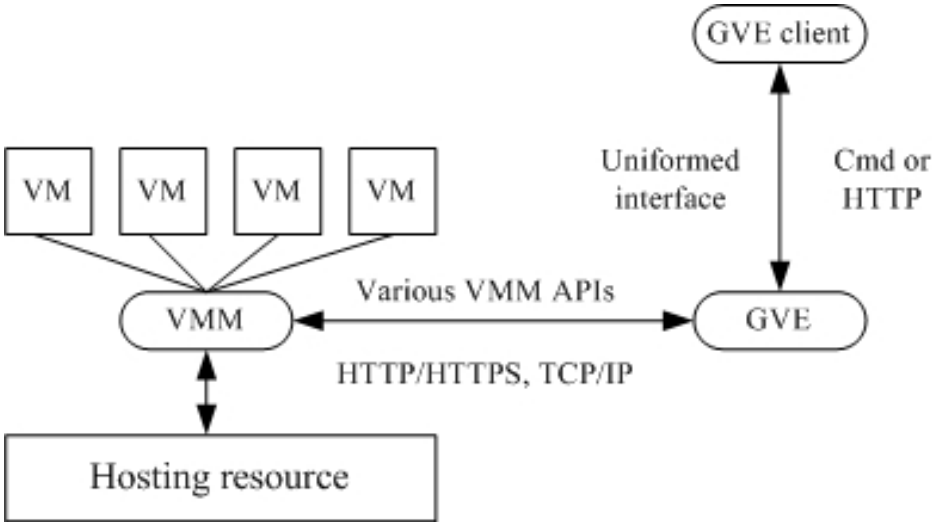
Lizhe.Wang@iwr.fzk.de

Grid Workflow Systems on Virtual Machines

- Requirements:
  - Grid Virtualization Engine GVE
    - Interface for deployment of the VMs at the specific Grid site on the different Virtualization Infrastructures – our contribution
  - Monitor/analyze/plan virtual machines with Grid Middleware
    - Information service of VM pool (our contirbution)
    - Interface to workflow planner
  - Execute Grid applications on virtual machines
    - Workflow engine: VDS (existing work from Globus alliance)
    - Globus Toolkit + Condor

GVE – Grid Virtualization Engine

- Definition:
  - Abstract layer on varous VMMs
  - Remotely operation on VMs via APIs provided by VMMs
- Implementation:
  - VMM APIs
  - HTTP/HTTPS, TCP/IP
- VMM:
  - XEN
  - VMware Server
  - VMware ESX



Questions?

Oliver.Oberst@iwr.fzk.de