Influence of Virtualization on Process of Grid Application Deployment

CCM case study

Distributed Systems Research Group
Department of Computer Science AGH-UST
Cracow, Poland
Krzysztof Zieliński,
Background

Virtualization is about creating illusions

• there are many CPUs on my one-CPU desktop,
• there is more memory in my system than physical RAM,
• a device is available here while in reality it is 100 km away,
• there is more than one machine in my box.
Background

• Virtualization is the process of presenting computing resources in a way which gives benefit over the original configuration.

• Virtualization describes the separation of a resource or request for a service from the underlying physical delivery of that service.
Key Features of Virtualization

• Consolidation
  – more resources perceived as one e.g. many hard disk drives creating one storage space

• Partitioning
  – huge resources dedicated for particular users e.g. one server machine divided on many virtual servers
  – isolation, security, etc.

• Containment
  – different resources perceived in the same way e.g. video cards from different vendors available with the same programming interface
Techniques of Virtualization

• **Machine Virtualization:**
  – Hardware-level virtualization
  – Hosted virtualization
  – Hypervisor virtualization

• **CPU**
  – HyperThreading, TSS, FSS

• **Storage virtualization:**
  – NAS, SAN

• **Memory**
  – Virtual Memory

• **Language**
  – Virtual machines – Java VM, .NET CLR
The VM-spectrum also leads to the notion that performance management of modern VMs is a function of the time and distance scales on which their respective polling mechanisms operate.
Influence of Virtualization on Process of Grid Application Deployment

VM-Spectrum

- **Macro Scale**: Low frequencies
- **Meso Scale**: Visible frequencies
- **Micro Scale**: High frequencies

Scale:
- **Visible**: Long wavelengths
- **UV**: Medium wavelengths
- **X-rays**: Short wavelengths

 Technologies:
- **P2P**, **GRIDs**, **PVM**, **Sysplex**, **VMWare**, **Xen**, **FSS**, **TSS**, **LPAR**, **z/VM**, **Microcode**, **HTT**

Clusters:
- **Cluster**, **VServer**

Processes:
- **Task**, **Process**

Threads:
- **Thread**

Cracow Grid Workshop 2006
Influence of Virtualization on Process of Grid Application Deployment

Resource Virtualization levels

Granularity

- Grid
- Cluster
- Computer

Virtual Server
- Project
- Task
- Process
- Thread

Hardware Resources

Virtual Resource

Virtualization as powerful concept in Computer Resource Management

Cracow Grid Workshop 2006
# VM-Spectrum

## Table 1: VM-spectrum Scales

<table>
<thead>
<tr>
<th>Spectral Region</th>
<th>Distance Scale (m)</th>
<th>Polling Period</th>
<th>Polling Frequency</th>
</tr>
</thead>
<tbody>
<tr>
<td>Macro</td>
<td>$10^2$ to $10^6$</td>
<td>min to day</td>
<td>mHz to $\mu$Hz</td>
</tr>
<tr>
<td>Meso</td>
<td>$10^0$ to $10^2$</td>
<td>ms to min</td>
<td>kHz to mHz</td>
</tr>
<tr>
<td>Micro</td>
<td>$10^{-6}$ to $10^{-3}$</td>
<td>ns to $\mu$s</td>
<td>GHz to MHz</td>
</tr>
</tbody>
</table>
Micro-VM

Figure 2: Simple block diagram comparing a 2-way SMP (left) with an HTT-capable Intel processor (right). The two blocks labeled AS (Architectural State) are registers which present themselves to the OS as two VPUs.
**Figure 3:** Simple polling model of a generic hyper-threaded processor with one execution unit servicing four thread registers or single-entry buffers (e.g. UltraSPARC T1). The AS registers in Fig. 2 correspond to two thread buffers (e.g. Intel Xeon).
Micro-VM conclusions

Micro-VMs, in the form of VPUs, should not be regarded on the same footing as physical CPUs. They are more properly regarded as sophisticated polling systems, polled at rates in the GHz to kHz range, with the number of VPUs corresponding to the number of single-entry thread buffers. Internal state management in these micro-VMs introduces an intrinsic and often variable overhead.
Meso-VM

![Diagram of Meso-VM](image)

**Figure 7:** Organization of Xen 3.0 hypervisor supporting Linux, Linux SMP, and Windows XP meso-VMs.
Influence of Virtualization on Process of Grid Application Deployment

**Meso-VM**

\[ N = \{N_r, N_g, N_b\} \]

**Run-queue**

\[ \{S_r, S_g, S_b\} \]

**Physical CPU**

**Users/groups**

**Figure 8:** Time-share scheduler model.
Meso-VM Polling

- Fair-share scheduler polling model of a meso-VM like Fig. 7. The hardware platform has the same logical association to the guest VMs as the physical CPU does to the VPs in Fig. 3.

\[ S_{VM}^{g} = \frac{S_{g}}{E_{g}} \]

FSS polls in every 4000ms

Process-level polling is essentially the same as standard TSS, while VM-share polling controls process-level capacity consumption.

Cracow Grid Workshop 2006
Meso-VM Conclusions

Meso-VMs are also implemented as polling systems operating at rates in the kHz to mHz range. Proportional shares are used to create software VPUbs in which the service rate is scaled by share-based entitlements. Proper share allocation can be critical for capacity management.
Macro-VM

The goal of macro-VMs is to enable scalable virtual organizations to provide a set of well-defined services. Key to performance is the network topology and its associated bandwidth.

Polling protocols are employed by macro-VMs in at least two ways: maintaining connectivity between peers, and security on the network.
Deployment Process

1. Installation tool
2. Configuration tool
3. Planning tool
4. Preparation tool
5. Lunching tool

Repository Manager
Requirements
Software package
Target
Resources
Deployment plan

The same, selected, virtualization layer

DSRG
Influence of Virtualization on Process of Grid Application Deployment

Cracow Grid Workshop 2006
Influence of Virtualization on Process of Grid Application Deployment

Matching an Application to Resources

Virtualization decisions

Cracow Grid Workshop 2006
D&C Specification

- Deployment and Configuration of Component-based Distributed Applications Specification
- Defined by OMG
  - first adopted version – April 2006
- Allows describing both resource properties and application requirements
- Platform Independent Model
  - PSM → CCM
D&G – Target Domain
What is a NODE?

- Most obviously
  - a computer system

- But may be something smaller:
  - an OS on virtual machine
  - a Solaris project
  - a process – in case of CCM a ComponentServer
  - a thread

- And why not something larger:
  - a cluster
Node as a Computer System

Computer System/OS in VM
- resources:
  - CPU, memory, storage, NIC,
- interconnect:
  - computer network
- bridge:
  - switch, router
Node as a Process

CCM Component Server
- resources:
  - heap, stack, threads, priority, CPU usage, start time
  - [JVM] GC algorithm, vendor
- interconnect:
  - IPC shared memory, pipes, signals
  - socket (unix, tcp, etc.)
- bridge:
  - proxy, gateway

SatisfierProperty
- name: "debugMode"
  kind: "Attribute"
  dynamic: false
  value: false

SatisfierProperty
- name: "niceLevel"
  kind: "Attribute"
  dynamic: false
  value: "32"

SatisfierProperty
- name: "heapSize"
  kind: "Capacity"
  dynamic: true
  value: 20970520

SatisfierProperty
- name: "stackSize"
  kind: "Capacity"
  dynamic: true
  value: 1048576
Domain Description Techniques

- Common Information Model (CIM)
- Directory Enabled Networking (DEN)
- DEN ng
- WS-Management
D&C – App. Requirements

Component Implementation Description

Monolithic Implementation Description

Named Implementation Artifact

Implementation Requirement

Requirement

Component Assembly Description

xor

1..*

primaryArtifact

deployRequirement

1

referencedArtifact

* deployment

* deployRequirement
Influence of Virtualization on Process of Grid Application Deployment

Deployment Process

1. Installation tool
2. Configuration tool
3. Planning tool
4. Preparation tool
5. Lunching tool

Requirements
Software package
Resources
Target

Deployment plan

The same description language

DSRG Cracow Grid Workshop 2006
Definition of D&C Domain
D&C – App. Requirements
Our goals

• Provide/compile different resource description models for different virtualization levels

• Supplement the D&C model in order to simplify complex and time consuming planning phase
  – deployment will be performed starting on the highest virtualization layer
Influence of Virtualization on Process of Grid Application Deployment

Matching an Application to Resources

Target Domain

Application
Thank you!

Any questions?