

Exploring the Behaviour of Fine-Grain Management for Virtual Resource Provisioning

Fernando Rodríguez-Haro, Felix Freitag, Leandro Navarro, and Rene Brunner

*Department of Computer Architecture
Polytechnic University of Catalonia,*

**Workshop on Models, Algorithms and Methodologies
for Grid-enabled Computing Environments**
*7th International Conference on Parallel Processing and Applied Mathematics
PPAM 2007 Gdańsk, Poland, September 9-12, 2007*



Outline

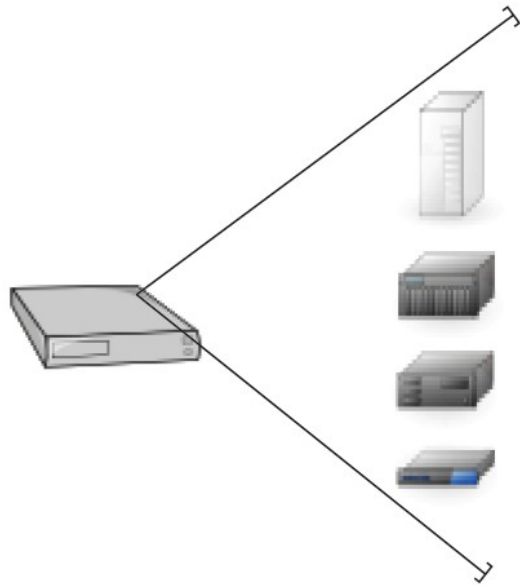
- Introduction
- Problem
- Prototype
- Results
- Conclusions



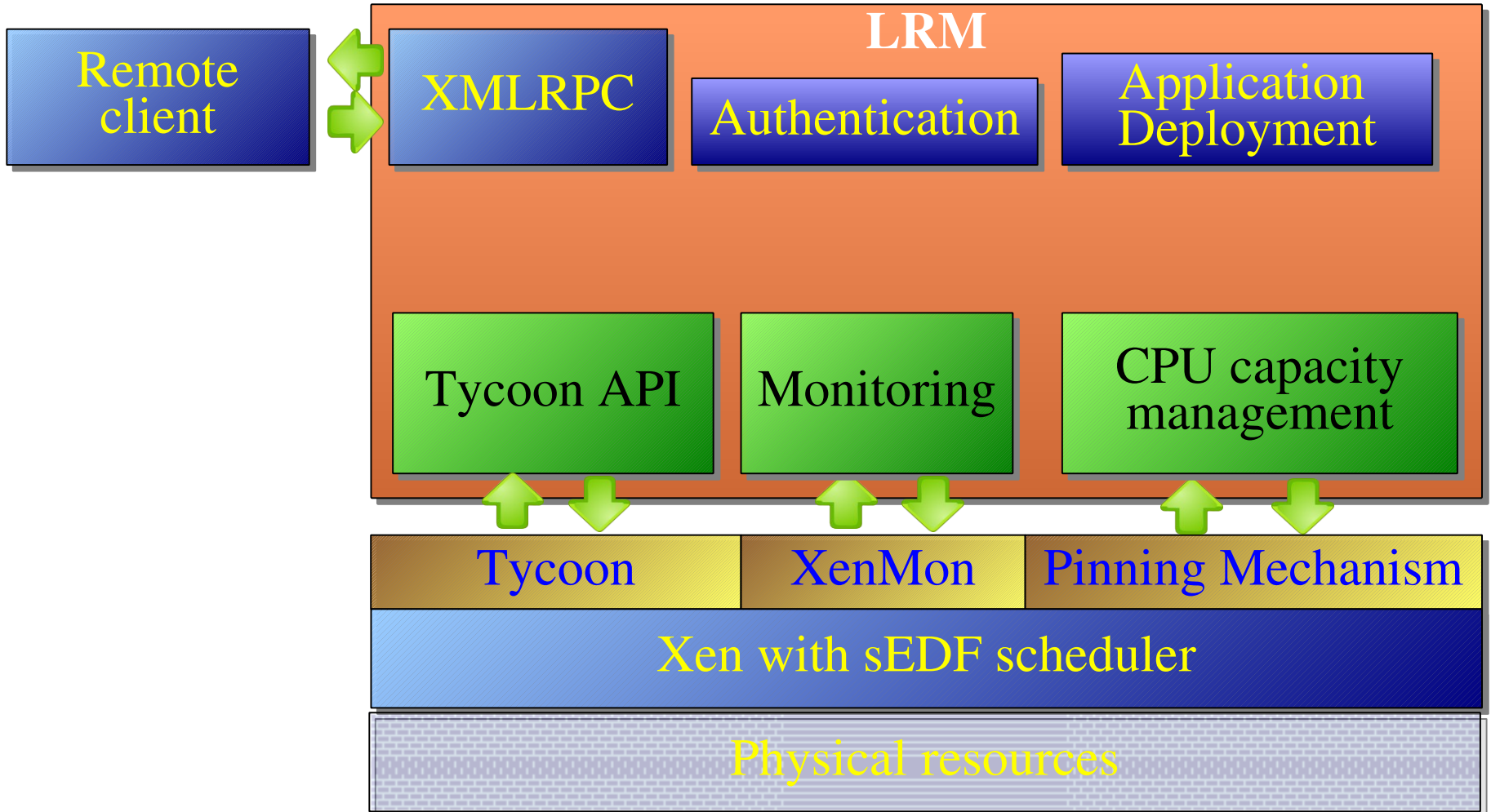
Context

- Recent advances in Virtualization
 - Boot multiple OS
 - SMP machines with Xen's sEDF round-robin based assignment of VMs and manual assignment via privileged commands
- Grid environments
 - Some processors could be over or underutilized and the VMs could receive more resources than specified
 - Manual assignment is not feasible in a dynamic environment and also requires being aware of each node's heterogeneity

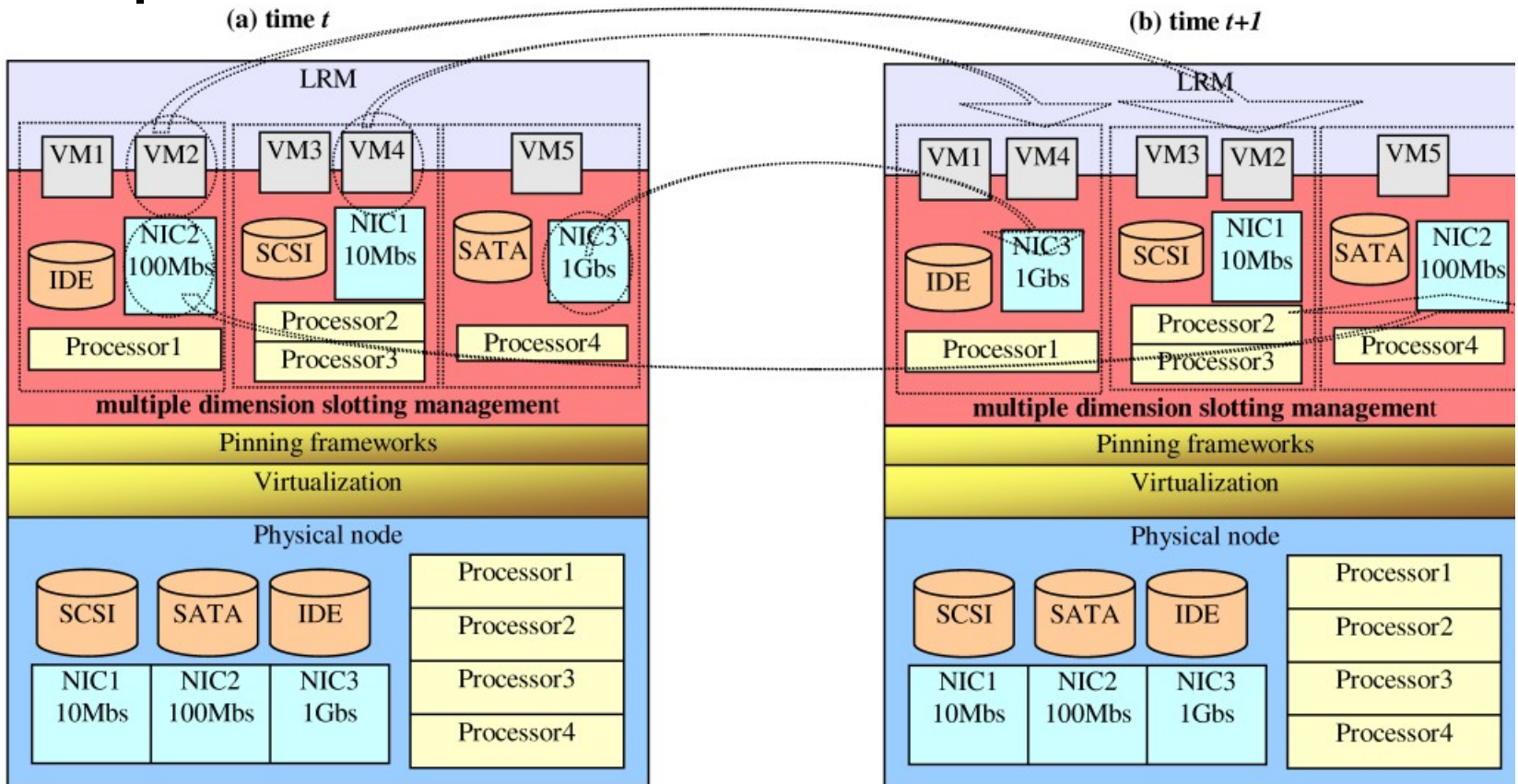
● ● ● | Problem



● ● ● | Prototype



MuDiS



Execution of a job (steps)

- First, the user define a configuration file with each job application requirements

```
[JOB.1]
slaCPU = 1000           #expressed in MHz
slaCPUfloor=0.10       #slaCPU tolerable percent degradation
slaBUDGETceiling=2.5   #spendable budget per GHz per unit of time
job = cputime.py       #application, source or binary
output = cputime.csv   #output file
necessary=True         #means schedule obligation
[JOB.2]
slaCPU = 1500
...
```

- Second, the user query the action to parse and execute the job plan definitions

```
python txtLRM.py --lrm=147.83.30.203 --jobplan=job6fabrics
```



Execution of a job (steps)

○ Third

- If the user requirements can be fulfilled by the resource provider then some processes are executed such as VMs creation and booting, application deployment via ssh credentials, launching and monitoring.
- Otherwise, the user receives information about the jobs's requirements that can not be fulfilled.

○ Finally, the user can retrieve the output files, if the job plan applications have not yet finished it is properly informed.



Experiments

- Hardware of the resource provider node
 - Pentium D 3.00GHz (two processors), 1Gb of RAM, one hard disk (160GB, 7200 rpm, SATA), and one network interface card (Broadcom NetXtreme Gigabit).
- Software
 - The operating system is Fedora Core 4 and for virtualization we use Xen 3.0.2. The Tycoon client and auctioneer are 0.4.1 version.

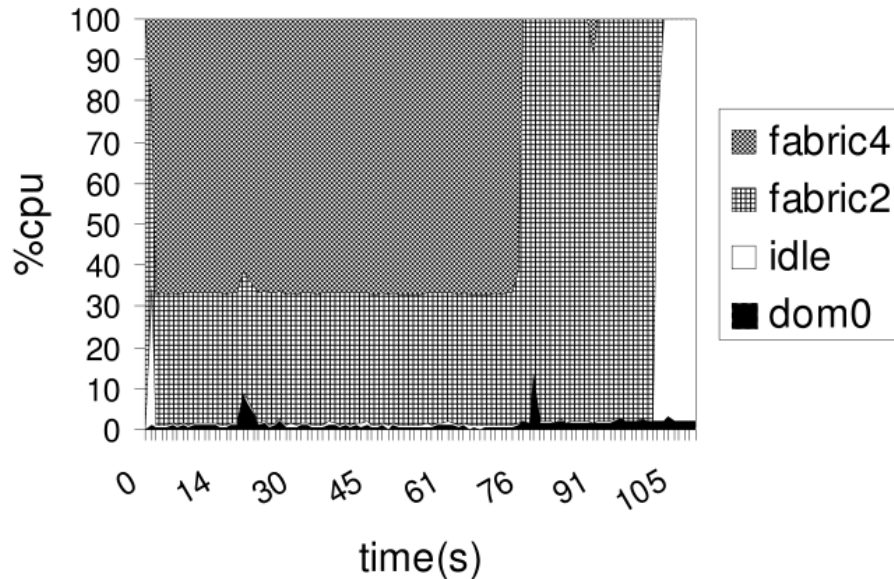


Scenarios

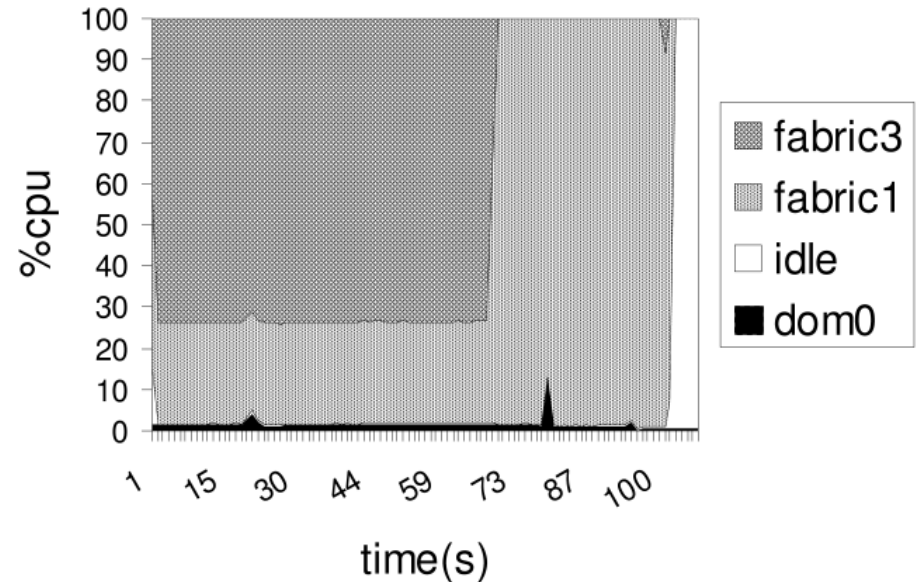
- Experiments with four and six competing virtual machines
- Application that behaves as a CPU intensive job. Executes 500 transactions
- CPU requirements (slaCPU) which must be expressed in Hertz
 - 300MHz, 600MHz, 900MHz, and 1200MHz for fabric1, fabric2, fabric3, and fabric4, respectively
 - 300MHz, 600MHz, 900MHz, 1200MHz, 900MHz, and 2100MHz corresponding to fabric1, fabric2, fabric3, fabric4, fabric5, and fabric6, respectively

Baseline experiment

processor 0

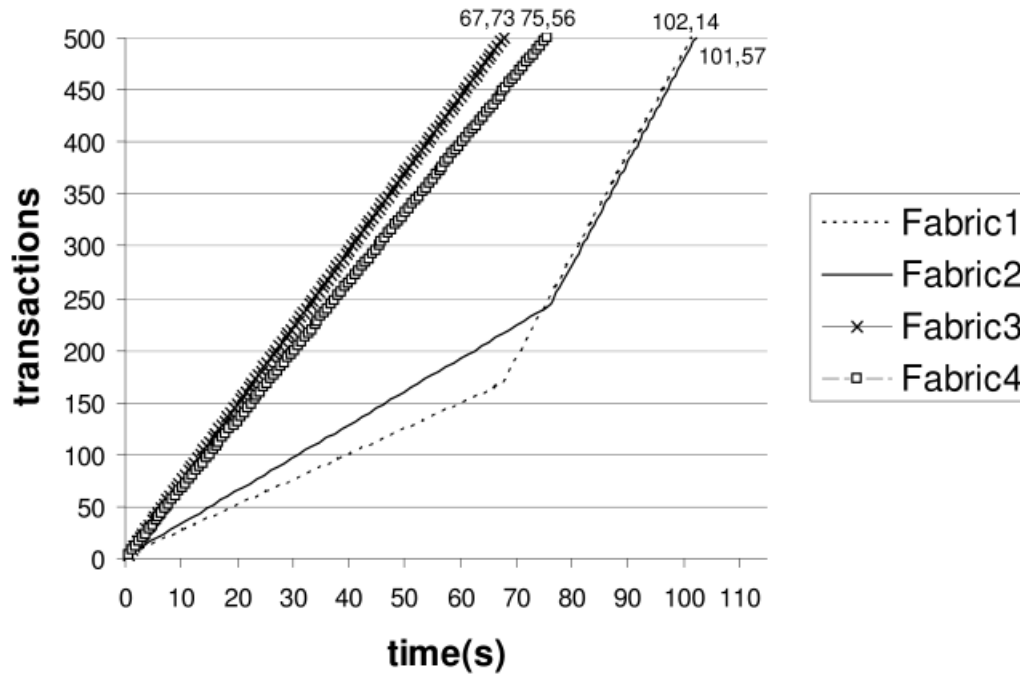


processor 1



○ four virtual workspaces, assigned by the original coarse-grain approach, executing the same CPU intensive job on a physical machine with two processors. The assignment of virtual machines is done in a round robin fashion by Xen VMM during creation. The four VMs are distributed on the two processors.

Baseline experiment

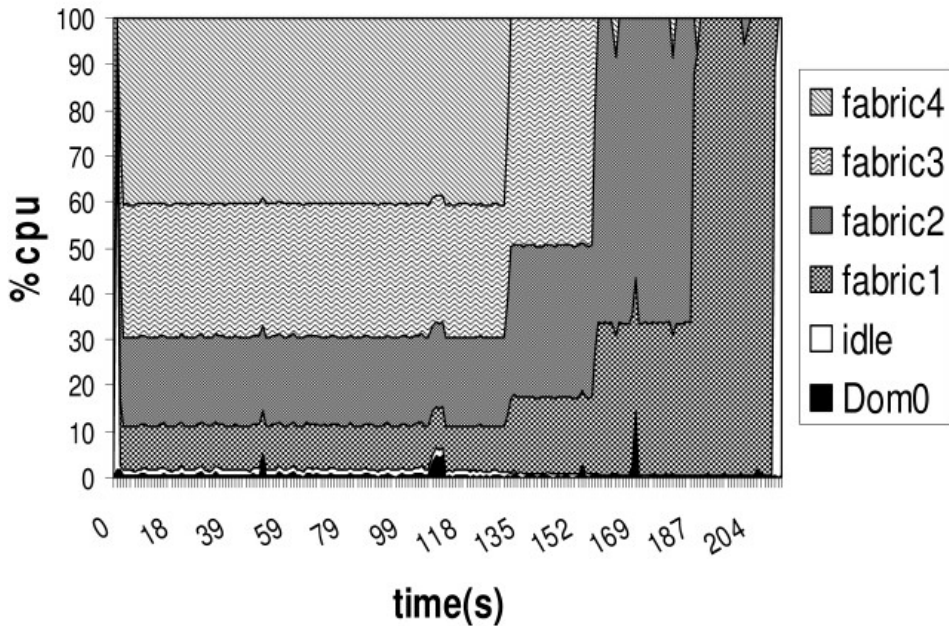


○ Transactions completed during job execution (original coarse-grain management). The time consumed per transaction is measured by the benchmark application.

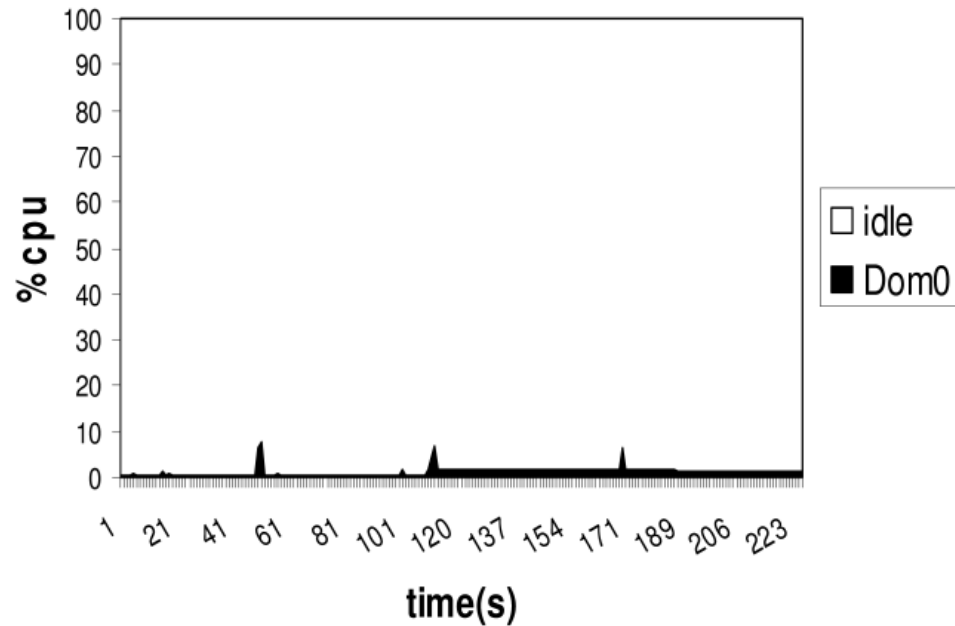


LRM enabled

processor 0

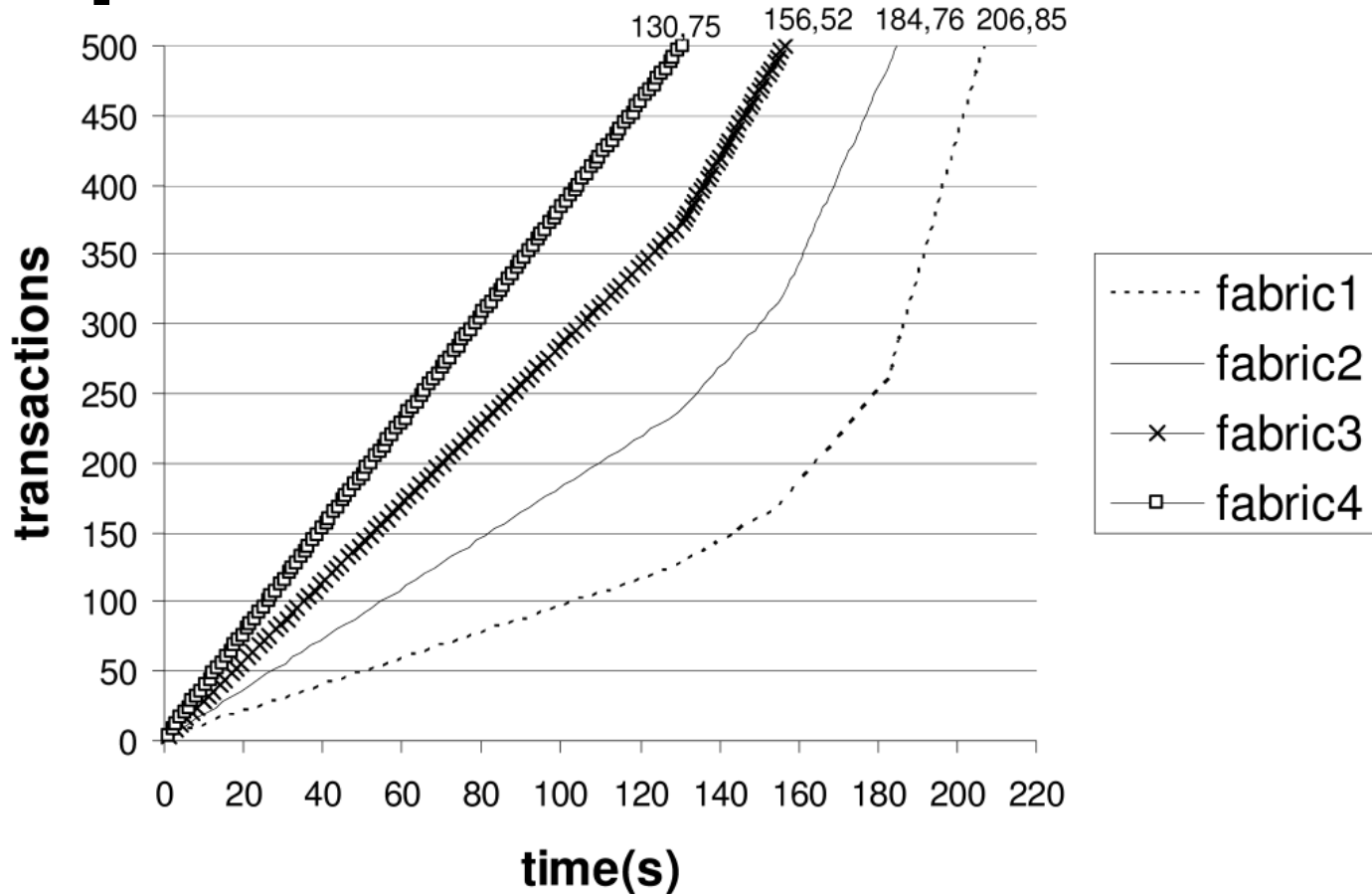


processor 1



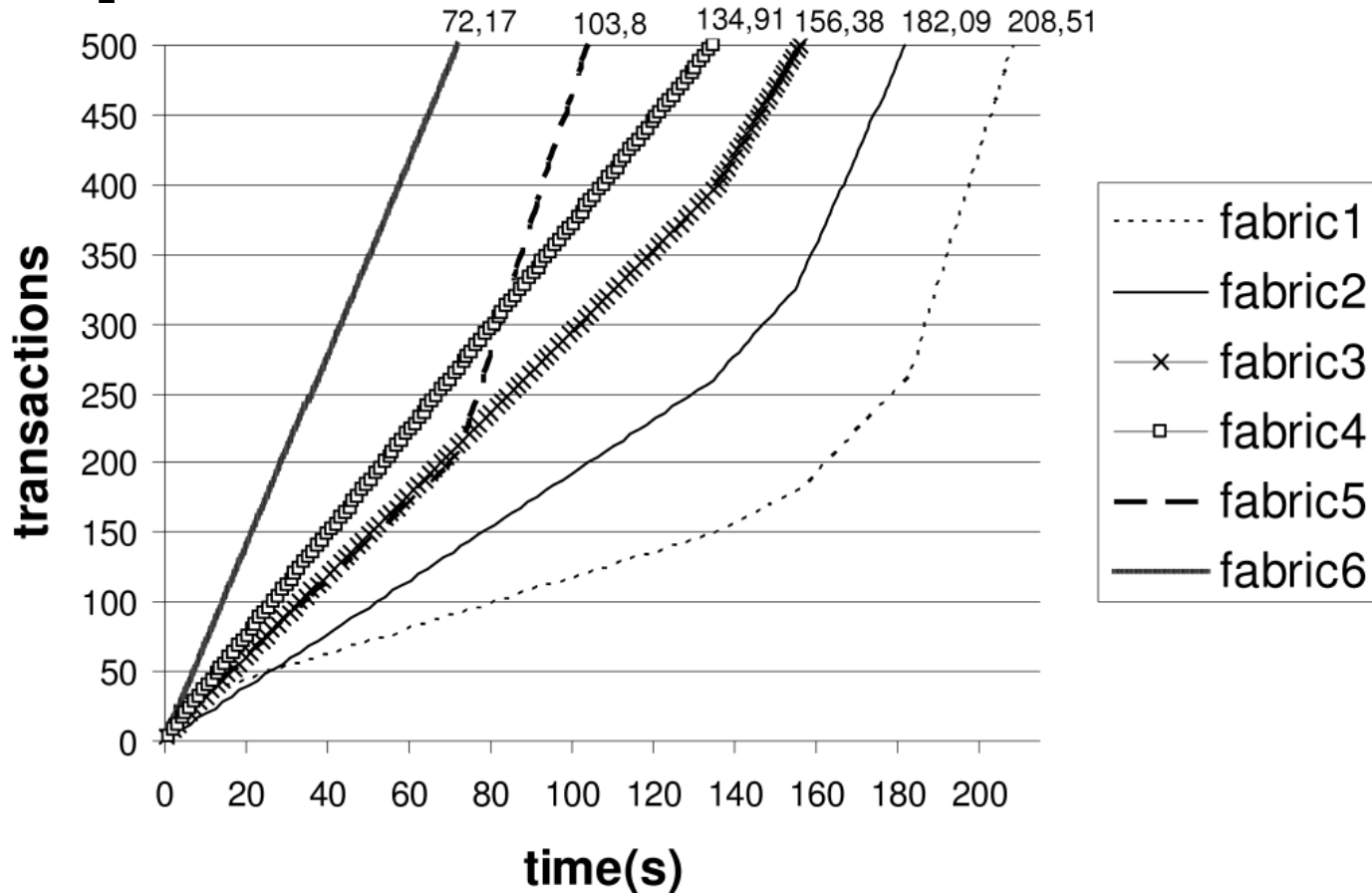
○ Four VMs assigned by fine-grain resource management through pinning mechanisms and sEDF scheduler. The four VMs are assigned to processor 0.

Experiment 1b



- Transactions completed during job execution (fine-grain approach). The sequence in the creation of VMs does not affect the expected performance; hence the execution time is according to requested share.

Experiment 2



Allocation of six VMs with LRM according to requested Hertz. The sequence in the creation of VMs does not affect the expected performance; hence the execution time is according to requested share.



Conclusions

- LRM. Fine-grain assignment of the virtualized resources of a SMP machine
- Local physical resources were properly assigned
 - Performance in *tps* was accurate and fulfills to agreed job plan requirements.
- Certainty about the expected completion times of the executed jobs.
- Better fulfilling certain subjective constraints given in the job plan requirements, like completions orders.