A review of the Intel Grid Programming Environment (GPE)

CERN openlab II quarterly review

31 January 2007

Håvard Bjerke

OGSA

- The “Open Grid Services Architecture”
- Has become the de facto standard for a Grid architecture
- Everything is a service - SOA
- All services are web services - “Web Services Resource Framework”
  - All services communicate via web services
  - WSRF exposes interfaces for managing state

OGSA

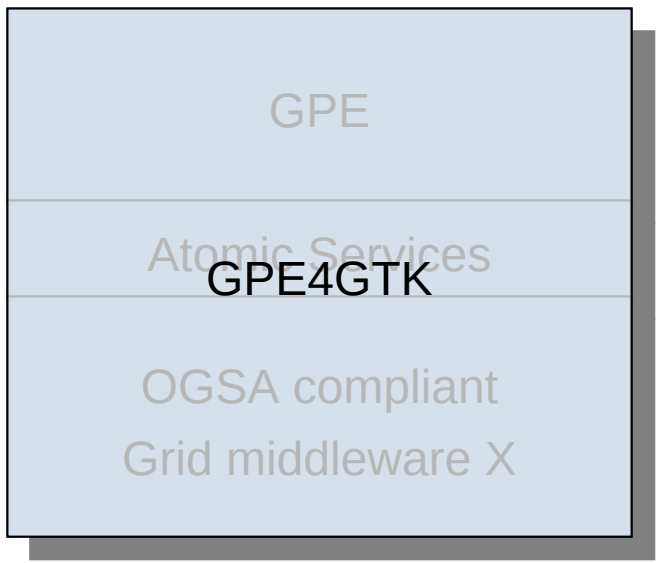
- Gives recommendations for which services should be in a Grid architecture
  - High-level services
    - Storage service
    - Data transfer service
    - Monitoring service
    - Workflow service
      - "Coordinates multiple application tasks"
  - Low-level services
    - *Factories*
    - *Registries*

What is Intel's GPE

- A complete and modern Grid middleware implementation
- An API over the OGSA
- Applications can be developed as GridBeans
  - A job package that can include GUI, job description, workflow
- The basic idea:
  - Take any OGSA compliant Grid middleware, put an *Atomic Services* interface on top of it, and you can use the GPE API to develop grid applications

Architecture

- GPE is independent from Grid middleware implementations
- An implementation, *GPE4GTK,* is provided, based on Globus Toolkit 4



- Implements
  - OGSA
  - Atomic services

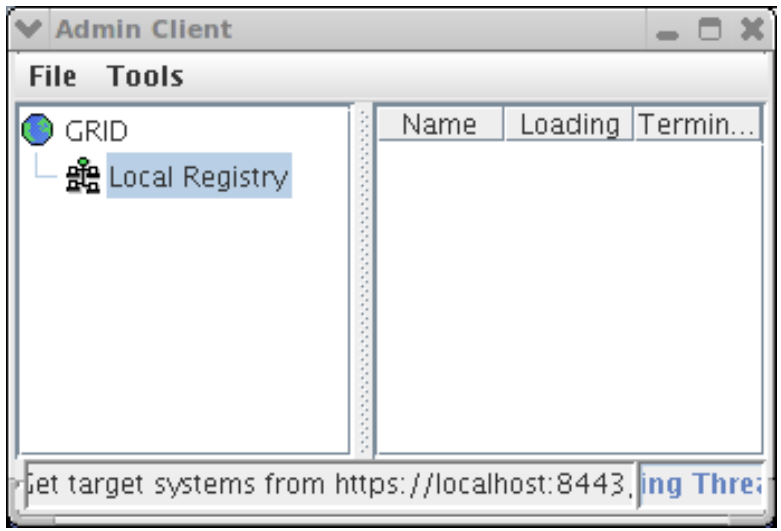
WSRF				
Target System Service	Target System Factory	Storage Management Service	File Transfer Service	Job Management Service
GT4 middleware				

Standards

- Other standards
  - Job Submission Description Language (jobs)
    - Open Grid Forum's standard for specifying requirements for jobs
  - Business Process Execution Language (workflows)
    - Orchestrating interaction between webservices
    - Backed by IBM, SAP, Microsoft
  - Common Information Model (resources)
    - Model description of resources

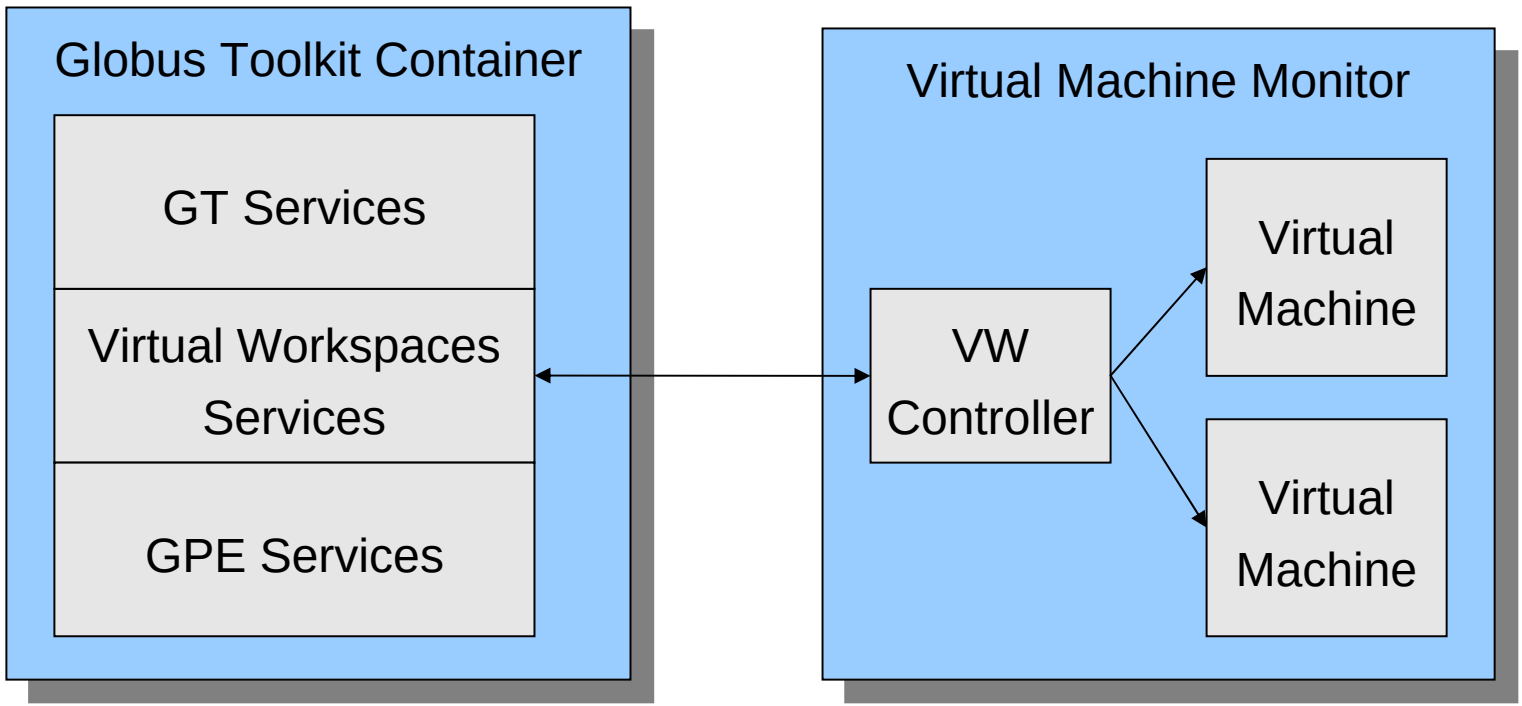
GPE Services

- Resources are encapsulated into *Target Systems*
  - Storage
  - Software
- *Target Systems*
  - *Created by Target System Factories*
  - *Registered in Target System Registries*

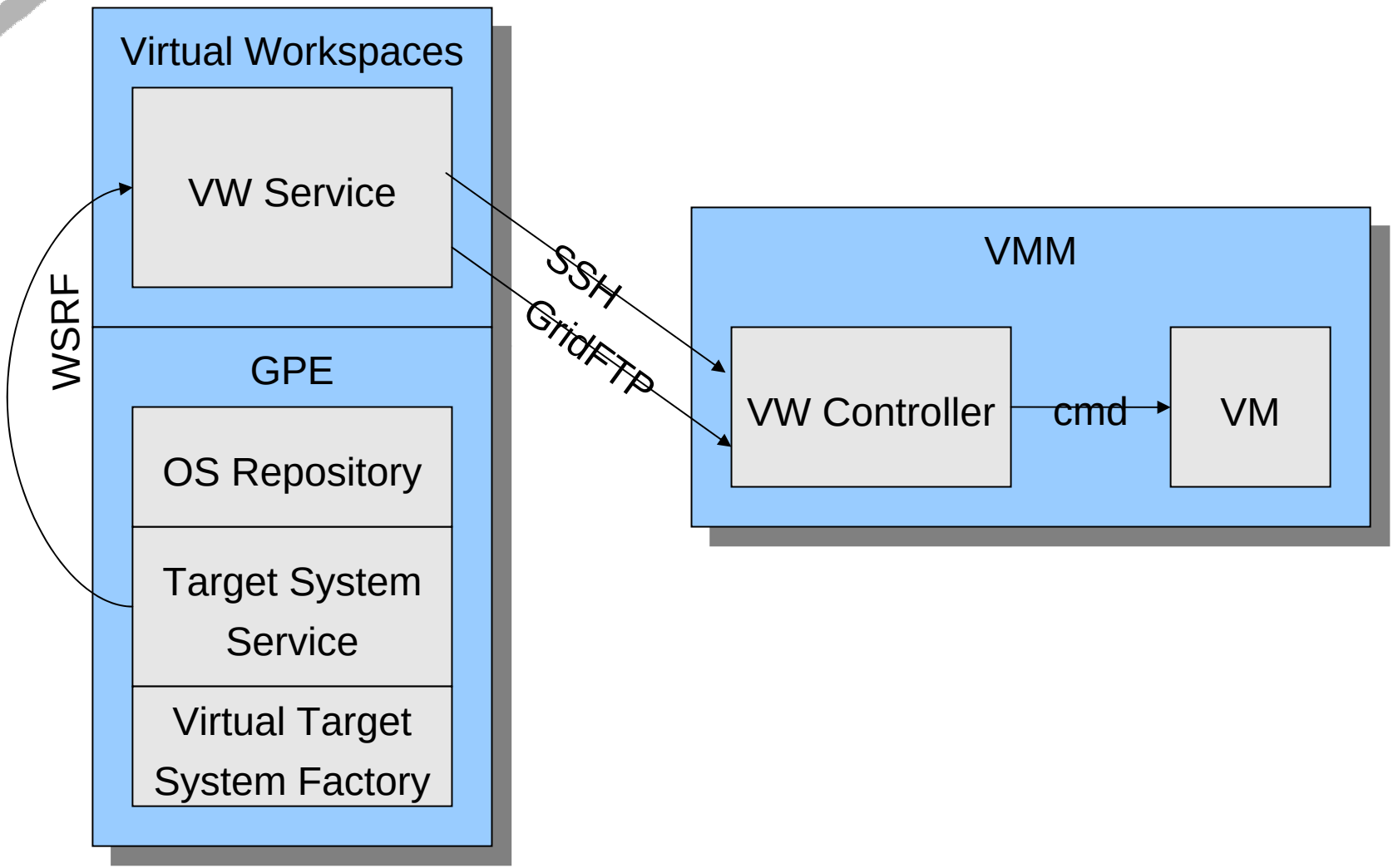


Virtual Target Systems

- The same as Target Systems, only they run in Xen Virtual Machines
  - Relies on *Virtual Workspaces* from Globus

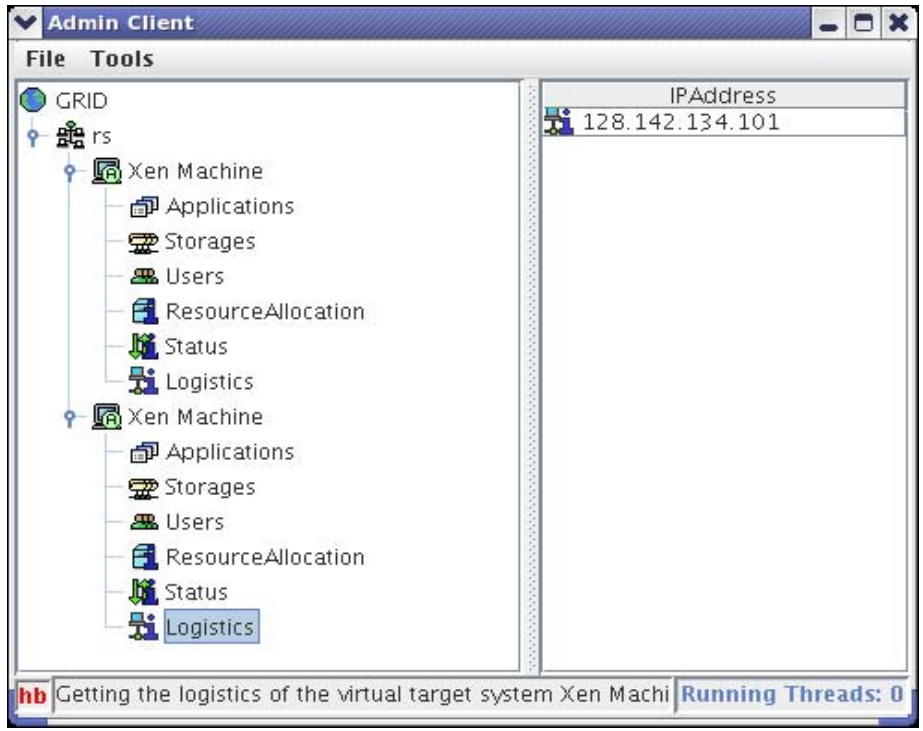


Virtual Target Systems

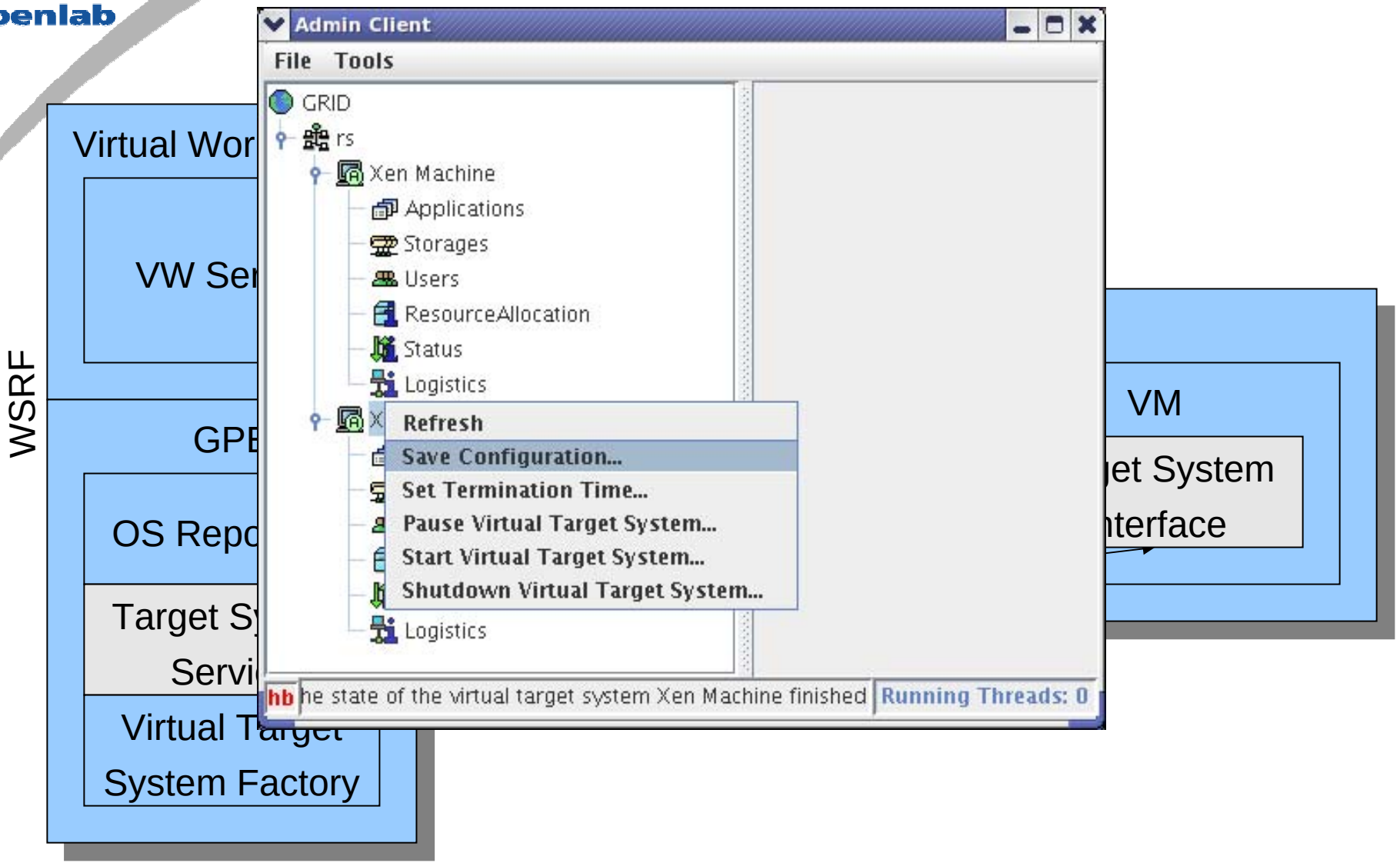


Virtual Target Systems

- Can use Virtual Workspaces' built in scheduler or manually
- OS Repository can be independent



Target System Interface



Comparison

- gLite
  - Based on GT2
  - Does not implement OGSA
  - Does not (yet) support virtualization
    - Several projects looking at using virtualized gLite services
- GPE
  - Incorporates standards for describing job requirements, equipment and workflows
  - The only Grid middleware to offer full virtualization support

Conclusions

- gLite
  - is already established as a de facto framework inside EGEE/LCG
  - resources are too static – there is a need to bring up resources more dynamically
- Architecturally GPE offers something significant to compete
  - ... but the community needs to be convinced
- There are several EGEE interoperability projects (e.g. Unicore)

Three scenarios for adaptation

- gLite interoperability with GPE
  - Interest from LCG depends on the availability of resources in GPE grids
- GPE could provide an API on top of gLite
  - Add atomic services layer on top of gLite
  - Problem: gLite is not OGSA compliant
- Use GPE's Virtual Target Systems to make gLite resources more dynamic
  - Problem: gLite is not WSRF compliant

OS Repository Service