Job execution in virtualized runtime environments in grid

Lev Shamardin, Andrey Demichev, Ilya Gorbunov, Alexander Kryukov

Scobeltsyn Institute of Nuclear Physics, Moscow State University (SINP MSU)

Why virtualization

Current gLite infrastructure is limited to one OS: **Linux**, and even worse, one particular version/flavour.

Benifits of virtualization:

- Better isolation of jobs.
- Enforced resource sharing policies.
- Different OSes.

Considerations

Wide adaptaion is possible if it is easy to install and requires minimal intrusion to gLite:

- minimal modification of CE, WN middleware;
- minimal impact on regular functions of the CE/WN;
- no modifications to the informational system or to core serivices (e.g. WMS)
- No modifications to the UI.

On the CE

Jobs submitted via RB/WMS are actually wrapper scripts.

Prepare the environment,
Fetch sandboxes,
...
Start the real job.
...
Shutdown procedures

Job, as arrived to CE from WMS

Our software patches the WMS wrapper script

Prepare the environment,
Fetch sandboxes,
...
Call wrapper with the real job as an argument
...
Shutdown procedures

Modified job, submitted to the WN

All actual work is done on the WN

Wrapper script launches the VM and emulates a "regular" job behaviour for the batch system. This simplifies integration with informational system (publishing node/cpu busy status, etc.)

In the informational system

Just advertize some new SoftwareRunTimeEnvironments

Example: a jdl for a MS Windows executable

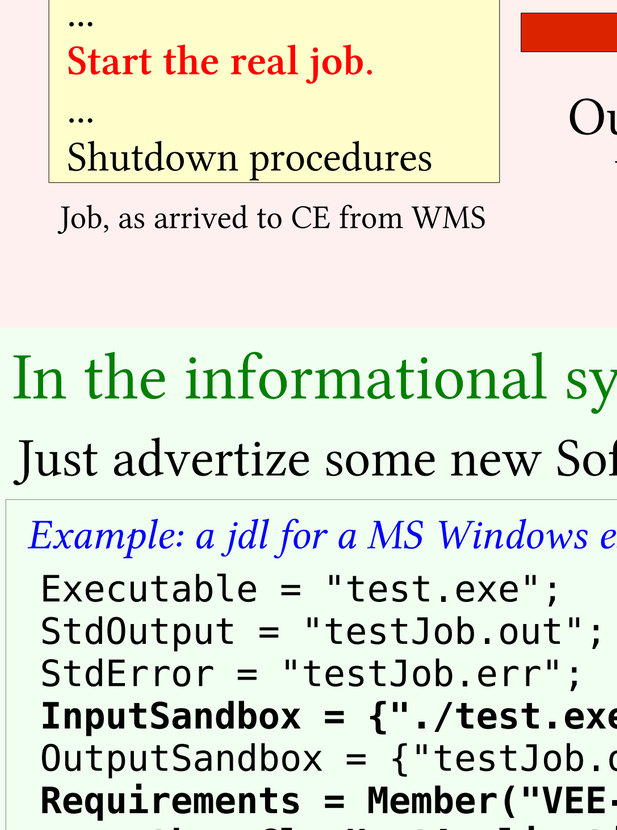
```
Executable = "test.exe";
StdOutput = "testJob.out";
StdError = "testJob.err";
InputSandbox = {"./test.exe", "job.jdl"};
OutputSandbox = {"testJob.out","testJob.err"};
Requirements = Member("VEE-WindowsXP",
    other.GlueHostApplicationSoftwareRunTimeEnvironment);
```

On the WN

VM Job Wrapper does:

- Check job requirements from JDL file *and local policies*
- Setup the VM image
- Mount VM image and copy inputs
- Start the VM
- Sleep and listen for events, wait for VM to terminate
- Mount VM image again and copy outputs
- It behaves as a "regular" job for the batch system

Prepared guest OS VM image



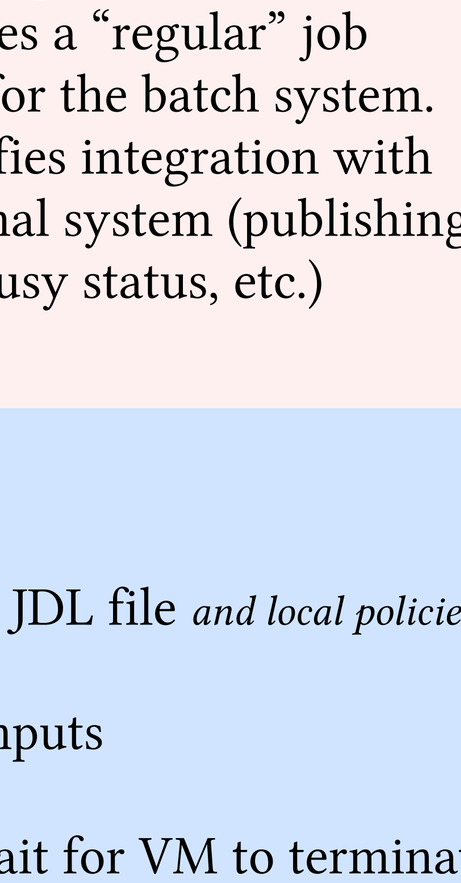
Guest OS image requires some preparation. It must:

- Automatically start the job from the preset location.
- Feed stdin to the job and capture stdout/stderr.
- Automatically shutdown when the job is terminated.

We have prepared images for MS Windows, Fedora Linux and standard gLite WN

User can provide its own machine disk image instead of the job

WN



Modified WN can be configured to wrap all jobs into gLite WN images

gLite integration

On the CE

- Packaged jobwrapper with a jobwrapper and helper scripts (rpm, obsoletes lcg-sam-jobwrapper).
- Informational system entries in /opt/edg/var/info

On the WN

- Storage area for the VM images, populated with blank prepared OS images.
- Xen
- Sudo configuration

Current limitations

- Only job sandboxes are transfered.
- No standard grid-aware mechanism to get/put sandboxes to a custom VM image.
- Jdl file must be a part of the sandbox

This work was partially sponsored by by The Federal Agency for Science and Innovation (contract No. 02.514.11.4072) and RFBR (grant No. 07-07-12023)