NA4: Tips for SEE Developers

Fotis Georgatos
GRNET

GRID-APP: Application's Development
December 19th, 2006
NTUA, Athens, Greece

Overview

- Developers – Prerequisites
- Tips: How consistent is the IS?
- Tips: Applications South Eastern Europe can run
- Tips: Why do checkpointing?
- Tips: How to do checkpointing

Developers - Prerequisites

- You need to understand the system, in order to be able to use it
- But, you don't need to know *all* the details: The only real requirement is a robust knowledge of a skill-set which is enough to kick-start with.
- Read the gLite User Guide's (v3.0?) index every now and then. You should at least be able to tell: "I can find the answer in the User Guide".
- Be ready to invest significant amount of time... ...and most likely you will get back any second of it!

Tips: Applications SEE can run

- **Simulations (CPU intensive, SEE has >1000 of CPUs)**
- **Bulk data processing (Storage, SEE has >100TBs+CPU)**
- **Parallel jobs (SEE: quite a few MPI supporting clusters)**
  - MPI fully supported in many (huge) clusters within SEE (40-200)
  - Interconnects are typically Gigabit, but some sites have Myrinet!
  - Constellation-like calculations should be possible in the future
  - HellasGrid is designed with Gbit provision e2e. Take advantage!
- **We'd like to see: workflows of ~medigrid applications**
  - Weather models, Fire behaviour, Flooding, Landslides etc
  - Anything with direct or indirect impact on level-of-living for SEE
- **We'd like to see: Scavenged resources & Applications**
  - We can do that, LiveWN proved that it is technically possible
  - Needs lots of experimentation on resource & job matchmaking
  - Needs lots of evolution in the Information System (Glue Schema)

Tips: How consistent is the IS?

- **Do clusters have WNs with...**
  - Same OS? Same CPU? Same clock-rate? Same RAM? ...
- **There is some heterogeneity in the resources provided by a given cluster (eg. slightly different clock-rates).**
- **It is imposed by the combination of hardware diversity and the capabilities of the IS, ie. GLUE schema v1.x!**
- **GLUE schema v2 will allow the description of "subclusters", so this issue will be overcome; going to be released in early 2007 (or not?).**
- **For the time being, users have to put their minimum job requirements in a .jdl, and if they see an error, to report immediately to Operations (SA1)'s help-desk tools.**

Tips: Why do checkpointing?

- **Check-pointing is the technique of storing the state of your job, typically in a set of files, so that you can later "reanimate" it. It is very critical for “long jobs”.**
- **If jobs take more than 24 hrs, it is imperative for users to consider check-pointing for a number of reasons:**
  - User: software engineering reasons, ie. debugging a checkpointable job is simpler to debug on longer-term issues, since it is possible to restart it on eg. the 6th day of execution
  - User: overcome the limit of grid queues (infinite jobs!) and be able to pause and reanimate the job arbitrarily
  - User: by limiting the execution time of your jobs, you can now tap much more resources (queues), which will decrease the total time
  - Systems administration: cluster downtimes can happen for either scheduled reasons (m/w upgrade, hardware installation or reconfiguration etc) or various unscheduled reasons (power/network/airconditioning outages, security issues)

Tips: How to do checkpointing

- **There are currently two well-understood techniques to implement checkpointing (c/p):**
  - Update some central service eg. some database like AMGA, about the current state of the job (easy, if state==integer)
  - Save the state of the job in a set of files, make a tarball (.tar.gz) & register it on a nearby SE with a predefined LFN!
- **Middleware is supposed to be able to assist, but in gLite v3.0 this is currently unsupported function (?)**
- **In effect, workflow and job management tools (ganga?) do state-maintenance at the large-scale of a gridified application, can this be integrated with checkpointing?**
- **Maybe future tools that are able to do workflow management, will be able to implement this internally!**
- **A known trade-off: high-throughput vs low latency c/p**

No users, no systems' value!

Contact info

Thank You

gef@grnet.gr