Contribution ID: **76** Type: **Oral contribution**

Real time computing for financial applications

Thursday 2 March 2006 14:30 (30 minutes)

Computing grids are quite attractive for large scale financial applications: this is especially evident in the segment of dynamic financial services, where applications must complete complex tasks within strict deadlines. The traditional response has been to over-provision for making sure there is plenty of 'headroom' in resource availability, thereby maintaining large computational resources booked and unused with a great cost in terms of infrastructure. Moreover nowadays some of these complex tasks need an amount of computing power that is unfeasible to keep in house.
Computing grids can deliver the amounts of power needed in such a scenario, but there are still large limitations to overcome. In this brief report we address the solution we developed to provide real time computing power through the EGRID facility for a test case financial application.
The test case we consider is an application that estimates the sensitivities of a set of stocks
to specific risk factors: technical details about the procedure can be found elsewhere; we will present here only the computational details of the application to better define the problem we faced, and the solutions adopted for porting it to the grid.

We implemented different technical solutions for our application in a sort of trial and error fashion. We will present briefly all of the attempts.

All implemented solutions rely on a "job reservation mechanism": we allocate grid resources in advance to eliminate latency due to the job submission mechanism. In this way, as soon as we get enough resources allocated we can interact with them in real time.
The drawback is that being an advanced booking strategy, for "best effort"services this approach could be unfeasible. It is not the case for this experimental work though, but the limitation should be taken into account when approaching production runs.
The booking mechanism has been implemented in the following way. An early submission of a bunch of jobs is run for securing the availability of WN at a given time.
Each pooled node will executes a program that regularly checks a host (usually the UI, but not necessarily). The contacted host enrolls this WN for the user's program, as soon as the user executes that program. When the
execution terminates the results are available in real time without any delay introduced by WMS of the grid. The WNs remain booked, and so are ready to be enrolled again for other program executions; eventually they are freed by the user. This approach, where the WN asks to be enrolled in a computation thereby acting as a client, is needed because the WN cannot be reached directly from the UI.

Author: Dr COZZINI, Stefano (CNR-INFM Democritos and ICTP)

Co-author: DI MEO, Riccardo (Egrid project ICTP)

Presenter: Dr COZZINI, Stefano (CNR-INFM Democritos and ICTP)

Session Classification: 2c: Special type of jobs (MPI, SDJ, interactive jobs, ...) - Information systems

Track Classification: Special type of jobs (MPI, SDJ, interactive jobs, ...) and information systems