Requirements Analysis for Dependability-Aware Grid-Computing

Geert Deconinck, Johan Driesen, Adrian Duşa, Rodica Tirtea, Koen Vanthournout

K.U.Leuven – ESAT – ELECTA, Kasteelpark Arenberg 10, B-3001 Leuven, Belgium
Geert.Deconinck@esat.kuleuven.ac.be
www.esat.kuleuven.ac.be/electa

Summary

Based on middleware, computational grids aim to seamlessly integrate heterogeneous computing and storage systems in order to provide a single execution platform for computation-intensive applications. Furthermore, such resources and the underlying networks can be dynamically added to or removed from the operational set in the grid, due to failures, restoration or administrator's/user's actions. The operational state of each of these resources has its impact on the dependability and performance. In order to meet the availability and integrity requirements of the end-user applications, algorithms and a set of middleware modules is required to deal with these dynamic aspects. Otherwise, faults propagate in the grid infrastructure and impair correct application execution [1, 2].

As stated in the recent European Expert Group report on Next Generation Grids [3], reliability and hence resilience are of key importance for grid infrastructures, especially in view of highly interactive applications (immersive e-learning, virtual reality systems, etc.). These topics however are identified as "weakly addressed" in the current grid oriented research efforts, resulting in a strong recommendation to gear research in the direction of resilient grid infrastructures. For instance, middleware and tools for computational grids currently mainly rely mostly on *static* parameters for application management (scheduling, load balancing, batch management etc.). This has been sufficient for applications with statically defined parameters on cluster-type of systems, but current trends go towards interconnected clusters and towards light-weight grids, where the infrastructure has no static configuration. In these cases algorithms and grid middleware modules are necessary that exploit *dynamic* information on the infrastructure in order to increase the efficiency, performance and dependability of the grid and its applications.

In order to extend grid middleware with these features, a requirements analysis is being performed from two different viewpoints.

- Firstly, the infrastructure perspective focuses on the impact of node / link / storage (i.e. component) availability. It needs to answer questions such as which fault and failure models are adequate; what is the impact of transient and intermittent faults on the availability of the upper layers; how do physical, malicious and operational faults contribute to the errors in the grid components; etc. New fault models are required because classical fault assumptions -such as single, independent crash failures- are inadequate in heterogeneous grids [4].

- Secondly, the application perspective focuses on the end-to-end availability. This needs to aggregate dependability information for all involved computational, communication and storage elements used by an application. This latter viewpoint is also becoming more important because many applications are moving towards service-oriented grid computing (e.g. Open Grid Service Architecture) [5].

To this extend, we are currently setting up several measurement campaigns to identify different characteristics of the local grid (*computing cluster* with 16 Opteron CPU's + *storage server* of 1 terabyte). This grid is connected to BEgrid [6], which provides (early 2005) over 200 compute nodes and about 5 terabyte of storage capacity. It uses the LCG 2.2.0 as middleware, which relies partly on the Globus toolkit. The measurement campaigns consist of several performance benchmark applications and small monitoring applications that are submitted to the grid under several scenarios (several target platforms, queues, distribution aspects, time periods, load periods, etc.).

A first analysis yields to several observations:

- The current grid middleware has several tools for monitoring available, but very few information is automatically integrated in the operational middleware for system management; much is left to the operator for deciding actions upon. A wealth of information concerning the operational status of grid components is available, but little information is automatically stored concerning faults, errors and failures, fault propagation, detection mechanisms, etc.
- There is little support for information security. Users belong to so-called Virtual Organizations and have to be authenticated when an application is started; a 'proxy' of this authentication follows the application software through its lifecycle (submission to job schedulers, transferring data, local access rights, transferring results, etc.). There is little support for confidentiality, non-repudiation, integrity, etc. (e.g. no application data encryption, root privileges for some application tasks).
- The current grid middleware provides a homogeneous view on top of a heterogeneous hardware platform, but leads to long response times (in the order of minutes) before tasks are started; this endangers availability for short-running applications.

References

1. Y.S. Dai, M. Xie, K.L. Poh, "Reliability analysis of grid computing systems," Proc. Pacific Rim Int. Symp. on Dependable Computing, Dec. 2002, 97-104
2. D. Thain, M. Livny, "Error scope on a computational grid: theory and practice," Proc. 11th IEEE Int. Symp. on High Performance Distributed Computing, Jul. 2002, 199-208
3. Anon., "Next Generation Grids 2 - Requirements and Options for European Grids Research 2005-2010 and Beyond", ftp://ftp.cordis.lu/pub/ist/docs/ngg2_eg_final.pdf, Jul. 2004, 32p.
4. R. Medeiros, W. Cirne, F. Brasileiro, J. Sauve, "Faults in grids: why are they so bad and what can be done about it?", Proc. 4th Int. Workshop on Grid Computing, Nov. 2003, 18-24
5. W. Bausch, C. Pautasso, G. Alonso, "Programming for dependability in a service-based grid", Proc. 3rd IEEE Int. Symp. on Cluster Computing and the Grid, May 2003, 164-171
6. Anon., "Belnet grid initiative," http://grid.belnet.be, *visited Jan. 2005*