Grid4All
democratize the grid
NEWSLetter
issue n° 4, March 2009

Events

The SICS Open House
Stockholm, Sweden, 24 April 2009

The SICS Open House is a opportunity for both industry and the general public to take part of the research done at SICS and interact with the researchers who present their works through talks and demonstrations.

We illustrate our distributed component management system, DCMS, for self-managing application by GMovie. GMovie is a CPU-intensive video trans-coding application that shortens execution time by splitting and distributing subtasks to multiple compute resources. With DCMS support GMovie performs self-management by healing when resources fail and configuring by adding/removing resources on demand.

Workshop Grids Meet Autonomic Computing
Barcelona, Spain, 15 June 2009
Associated with the 6th International Conference on Autonomic Computing (ICAC'09)

Managing democratic grids: architecture and lessons learnt by Leandro Navarro (UPC)

The talk will present the architecture, results and lessons learned in the EU project Grid4All (2006-2009) on the construction of a democratic grid infrastructure, where self-* properties are an essential element.

These grids have challenging and novel requirements: usability, content sharing and collaboration, security, scalability and availability via decentralization, brokerage of resources, and autonomic management of the dynamics of organizations and applications. These are considered essential factors for next generation Internet computing and generic enablers for cloud computing.

The talk will present the architecture of the Grid4All infrastructure, the self-* mechanisms incorporated to enable these grids built on conventional computers and networks contributed by the participants, and the lessons learnt during the construction and evaluation.

30 Months work

After 30 Months of work, the Grid4All consortium presents some conclusions obtained from the work done.

Overlay Infrastructure and Programming Models

At the infrastructure and platform level, this team focusses on core VO services, the bottommost layers in the Grid4All architecture. The focus is on **tools to develop services/applications** that can be deployed, run, and manage themselves in dynamic environments characterized by **high rate of churn**.

What's been achieved?

1. Development of the DCMS (Distributed Component Management System). DCMS is a distributed runtime execution system for self-managing applications & services in dynamic VOs. It provides basic support for **deploying distributed applications**, in cooperation with the VO resource service. It supports sensing changes in the state of components and environment, and allows individual components to be found and appropriately manipulated. It deploys management components and sets up the appropriate sensor and actuation support. The current prototype is largely completed; though, replication of management components (for robustness) is still rudimentary. The DCMS is a VO-wide infrastructure based on structured **overlays to provide good scalability and robustness** in the face of churn.

2. Development of a number of tools and abstractions that allow management logic to be specified at a high-level. There is currently an ADL compiler for **describing initial configurations** (which are compiled to DCMS instructions). Other tools, FractOz, Lactoz, demonstrate languages for creating self-configuring components/systems, though they are not yet coupled to the DCMS.

3. Considerable progress toward has been made, i.e. have enabled, the final goal of being able to **offer a full-fledged high-level programming system** for self-managing applications/services in dynamic environments.

As main innovation, we'd like to point the novel use of overlays in the DCMS, where an adapted structured overlay provides a **network-transparent sensing/actuation** infrastructure. We have also adapted known distributed algorithms to dynamic groups. DCMS provides uniform handling of executables and runtime structures as well as parameterized architecture descriptions which support dynamic reconfiguration and reuse of management logic.

Based on this work, the consortium considers that there are many aspects of **scalability** to deal with. For example, the size of the VO itself, the number of applications/services that are supported within it, the number of components and number of individual machines running a single application/service or the number of components in a component group. Without clear end-user-application requirements it is very hard to prioritize which scalability that needs the best support.

A middleware (such as DCMS) clearly reduces burden from an application developer because it enables and supports self-management by leveraging self-organizing properties of structured P2P overlays and by providing useful overlay services such as deployment and name-based communication. However it comes at a cost of self-management overhead, in particular, the cost of monitoring; though this cost is necessary for the democratic grid that operates on a dynamic environment and requires self-management. This opens new opportunities for research on efficient monitoring and information gathering/aggregating infrastructures to reduce this overhead. Another research focus is on high-level programming abstractions and a language support that should facilitate development of self-managing applications.

Virtual Organization and Resource Management

At a next level, this package provides services that are required to provision Virtual Organizations. Organizations that may grow or shrink according to activity conducted. Applications may require more resources or may also need to access needed services.

Semantic information services (SIS) provides access to and manipulate information about **markets trading resources and services** in the Grid environment.

Democratization of Grids implies bridging between different and multiple perspectives of agents participating in the Grid environment. Semantic descriptions and matching between different perspectives at the semantic level tends to be mandatory within a democratic environment. The Semantic Information Service **enables this objective**.

The SIS has shown two important steps towards democratization; stakeholders for SIS are end-users and application developers who can use a web-based GUI to interact with the system. Software agents can interact with SIS via the SIS-API. SIS provides **automatic tools to annotate services**, thus relieving developers of obligations to master semantic technologies. This, together with an important collection of algorithms to align different ontologies can **facilitate the seamless migration** of current Grid and service entities to become semantic Grid entities.

Nevertheless this comes with a cost; the initial **prognosis about scalability** is indeed true. An important lesson learnt is that future research should search a balance between semantic approaches and distributed data management. Work remains to further develop decentralized and distributed services for the manipulation of semantic entities. Future and long term deployment of semantic information services will indeed become a reality.

SIS and Grid4All market place tools are symbiotic. Agents (software/human) discover trading instances for resources and services by querying the semantic information service (SIS). This **provides a ranked list of markets or services** that satisfy user requirements and preferences. Further work will investigate the live update of advertisements' status as the market dynamically evolves over time."

Grid4All approach to decentralized market places relies on efficient access to market feedback that aggregates per commodity prices decided at multiple and simultaneous but independent resource auctions. A number of interesting problems are opened; how to guide tradeoff between accuracy of such information with respect to cost of obtaining such information, how to monitor activity and what intelligent tools may be deployed by market place operators to facilitate agents that require or offer resources.

Data Storage

The goal of the Data Storage is to **develop new technologies** for data sharing, enabling new styles of cooperative work within virtual organisations:

- VOFS, an easy-to-use approach to sharing files by federating users' resources.
- The Telex replication middleware that enables users to edit shared documents and reconcile divergent updates.
- Two systems for concurrent editing, P2P-Xwiki and Treedoc.

These technologies have been **successfully** designed, developed, integrated, demonstrated and evaluated in WP3. Key to the success was clear, **simple and common architectural vision**, resulting from frequent meetings and a lot of joint engineering work.

Interesting research sprung up as an unplanned side-effect of the collaboration: the development of concurrent editing algorithms, work on security in distributed collaborative systems, and research on partial replication.

Distributed, replicated, stateful systems are extremely complex; we learned that, instead of reacting to problems, it is **better to design the system to be immune** to them. For instance, designing concurrent operations to commute avoids concurrency control and reconciliation issues. Another example is the P2P paradigm that automatically converges despite failures and churn.

Scalability remains an open issue. Scalability in number of users is less critical than initially thought, as a collaborative workgroup will rarely be very large. We were unable to find the "killer app" for complex reconciliation scenarios.

The platforms and technologies developed in WP3 will continue to support future research in new areas such as cloud computing and social networks, with a focus on P2P technologies, large-scale data sharing, and privacy.

Applications

This group has been developing or adapting pre-existent applications for a decentralized environment as Grid4All.

Developing applications for the grid4All environment has proven to be more **challenging** than for an idealized (centralized, reliable and stable) model. Partly this difficulty has come from the lack of equivalent tools to facilitate the work under this environment (something already expected). Discounting this, it continues to be harder as application developers have to deal with issues that previously were invisible (transparent) to them. This extra price was also expected, but slightly underestimated. Partly is obviously due to the lack of maturity of the API.

However this extra effort has shown valid when **the benefits of enabling applications to work on a wider range of environmental conditions** are desired (more dynamic and therefore more realistic: e.g. dealing with environmental changes using DCMS, VOMS, GRIMP, indirectly using the Niche overlay), taking advantage of some characteristics of the infrastructure without having to deal with all their drawbacks (Virtual Organizations and VOMS as a way to isolate from boundaries, the Scheduling service and GRIMP to profit from additional resources).

The grid4All infrastructure also **enables applications to share information** in ways that other applications try to avoid (e.g. concurrent modification of shared objects with Telex) leaving the complex details to the infrastructure, and provides a common environment where data (file naming and sharing with VOMS) and complex information (e.g. semantic information about resources with SIS) can be **easily shared across multiple applications** with common security checks.

More information on overall results can be found at www.grid4all.eu.