A democratic Grid: collaboration, sharing and computing for everyone

Ruby Krishnaswamy[1], Leandro Navarro[2], Vladimir Vlassov[3]
[1]*France Telecom R&D, 38-40 rue du General Leclerc, 92794 Issy-Les-Moulineaux, France*
Email: ruby.krishnaswamy@orange-ftgroup.com
[2]*Technical University of Catalonia, Jordi Girona 1-3 D6, Barcelona, 08034, Spain*
Email: leandro@ac.upc.edu
[3]*Royal Institute of Technology KTH/ICT/ECS, Forum 120, SE-164 40 Kista, Sweden*
Email: vladv@kth.se

Abstract: This paper presents an integrated vision, architecture, middleware and applications of a public and large scale Grid for everyone: supporting collaboration for groups of people who can interact and share work and share or trade computing resources among them. This public Grid is decentralized and self-adapting to the dynamics of the online world with networks, computers and people who come and go, fail and recover and applications with varying loads and resource needs. Initial evaluation based on the first release of the middleware components and applications shows how our Grid can operate in a dynamic and decentralized environment by a combination of pooling and market mechanisms that adapt supply and demand for resources, self-managing services and applications that react to environmental changes and generic data-sharing services for concurrent write-sharing. The potential for societal impact is enormous as it can open Grid computing and collaboration to everyone on the Internet.

1. Introduction

The Grid4All project [1] promotes the concept of a democratic grid by opening up participation in a global computing infrastructure accessible for everyone, to modest groups of end-users such as schools, families, NGOs or small businesses, by lowering costs, facilitating usage and supporting collaboration and data sharing. This is a grand challenge for the grid comparable in societal impact to the democratisation of networking by the Internet.

A grid is a distributed system that virtualises and aggregates resources exposed by separate institutions. A key concept is the Virtual Organisations (VO), a virtualised collection of users or institutions that pools their resources into a single virtual administrative domain, for some common purpose. Grid4All shares the requirements of the Open Grid Services Architecture (OGSA) [2]. However, previous grid systems, such as OGSA, are designed for “big-iron” use cases, for instance commercial data centres, large-scale scientific computing, etc. They remain complex and heavy-weight. Our target use cases, involve schools, learning institutions, families, and small businesses using low-end and less reliable PC and networks. In terms of usage, our scenarios are mostly oriented to supporting sharing information, collaboration, whereas remote execution of large computations is less central than in existing grids. In the area of data services the Grid4All middleware provides support for concurrent access and modification of documents and application data even with some disconnected participants (remote mode).

In this model everyone can be an active participant: anyone has the right to create a group (a VO), anyone can be invited to join a VO being free to contribute resources or participate in it, any VO can buy or sell resources or services in this community, any networked computer able to run the Grid4All middleware can be offered and contributed as a resource, no one will be limited by the capacity of his personal computer or the place where it lives as computing and storage resources can be obtained from the network and applications may allocate as many resources as it requires to run, anyone can develop or use any compliant application on it as the interfaces are publicly available. Affordable, fast, flexible and easy access to computing can help create more egalitarian societies.

This paper presents the vision, challenges, design and initial results and lessons learned in the development of a distributed computing middleware and applications that extends the model of Grid or Cloud computing in three main directions: (a) decentralization, as resources can be contributed by anyone and be of any type of computer, not just dedicated data-center servers; (b) self-management, as services and applications can work despite the dynamics of changes such as disconnections and failures, and (c) collaboration, as data services are provided to facilitate the development of collaborative write-sharing applications that allow the concurrent modification and reconciliation of changes by multiple participants even in disconnected mode. This middleware, installed in any number of computers will form an open, self-managing, and decentralized global computing infrastructure where collaborative work and learning can take place by using the applications developed within the project and other applications, new or ported to use the Grid4All middleware.

The rest of the paper is organized as follows. Section 2 presents the objective and specific challenges being addressed. The next section describes the methodology and the issues we focus on. After that, section 4 describes the technology used to meet these challenges, followed by the developments and results achieved and ongoing. Section 6 presents the expected business benefits and the paper ends with some overall conclusions.

2. Objectives

The objective of this work is to provide a way for everyone to participate in a global computing infrastructure to collaborate and work together. This infrastructure should be open to contributions (any networked PC, shared or dedicated, at home or in a data center) and usage by everyone, and provide every time to (self-managing) applications the required resources to run, store and share information (mainly collaborative work and learning applications). This grid for everyone should work on today's PC and networks and be accessible, easy to use, and with minimal administration. There are some specific challenges:

- Dynamics: the work environment is changing and unpredictable. Participants and computers come and go. Load varies over time, and computing resources can fail. Nonetheless, the system should remain manageable to ensure some level of dependability (e.g. an educational project should not miss its deadline because of a glitch in the system). Otherwise, people will be reluctant to use it.
- The responsiveness of the system should not degrade despite partial failures, network problems, and disconnected-mode work.
- The system must appear simple to its users. The system should adapt automatically and autonomously to problems without requiring any support staff.
- Collaboration among distributed partners requires computer support. Activities among people in different locations may be not viable face-to-face and given the potential number and diversity of participants, there is need to use applications to support sharing, coordination and collaboration at the same time or different time.
- Participants may move to multiple locations: in the school, at home, in a library, on the move, or at work in other cities or countries elsewhere in the world.

- A collaboration incorporates rules and policies. For instance the members may need to abide to pre-existing rules of the group, or to make contributions, either monetary or in kind (e.g. computational resources), and system may need to support or enforce this.
- Participants must be able to work remotely in disconnected mode. It is not reasonable to expect that all participants are connected all the time.

The objective, considering the challenges, is achieved by the development of a middleware that, installed in every computer, forms decentralized overlay networks, implements self-management mechanisms to stabilize the system in a changing environment, and provide a collection of services to compute, store and share information that are used by collaborative applications to support groups of people to work together at a global scale.

3. Methodology

The methodology is based on a prototype implementation of each architectural component and a limited integration to show the feasibility of the vision by the demostration of end-user applications using the Grid4All API, the evaluation of the expected behavior of components under diverse conditions by simulation and prototypes. The work has followed several steps since its beginning in mid 2006: the conceptualization of the idea of a democratic grid, the design and development of a middleware to support that model, the development of applications that demonstrate how the middleware can be used to facilitate new ways of collaboration and social participation, and the evaluation, albeit limited and planned mostly for the first half of 2009, of some key aspects of the system in terms of the challenges presented previously.

Our design is oriented towards usability, content sharing, collaboration, and flexibility. Accordingly, our work extends on the current state of the art of grid systems, focusing on a small number of issues:

- Usability for lightweight, flexible and dynamic VOs: Many VOs will be short-lived (minutes, hours or days), but some will last very long. Most will be small (a few tens of users), but some may be very large. Users will join and leave, and resources will be added or become unavailable, in an unpredictable fashion. The manual effort to set up or terminate a VO should be extremely low.
- Novel and collaborative applications: Existing grid systems, designed for large-scale scientific computation, focus on massive remote execution services, and data management is relatively primitive. Grid4All provides novel support for remote execution, targeting home and school users. However its greater focus is to support novel collaborative applications, such as distributed authoring tools, e-learning environments, or decision-making.
- Application management in dynamic VOs: Not only must the VO be kept in a robust and coherent state in the face of the greater dynamism in democratic Grids but also the service/applications running within the VO. We need to provide the basic tools and monitoring/control infrastructure to enable application programmers to develop self-managing applications for Grid4All.
- Availability, scalability and decentralisation: The size of a VO in an educational institution might range from ten users to a few thousand. However, a popular VO may grow unpredictably. Even if individual VOs are small, there may be many, and they may have a large geographical distribution.

Therefore, a democratic grid system should be able to scale seamlessly over a wide range of sizes and of geographical and administrative extents. The volatility of resources makes achieving availability challenging. Accordingly, the architecture avoids any single point of failure, or centralised performance bottleneck.

The result is a Grid middleware and a set of collaborative applications that enable everyone to work together by pooling computers and work together by sharing data, files and running tasks.

4. Technology Description

To meet these challenges, the Grid4All architecture is based on a peer-to-peer overlay infrastructure, enables autonomous management, and provides federative and collaborative data services:

- Overlay infrastructure: We use peer-to-peer (P2P) networking techniques, e.g., self-organizing overlays and Distributed Hash Tables (DHT), which tolerate high rates of unplanned resource arrival and departure (churn). They scale well thus supporting VO evolution. The overlay infrastructure holds the VO together in dynamic environments assuring that individual resources, members, and components can be discovered, monitored and controlled.
- Autonomous management. To ensure usability in a dynamic environment, Grid4All aggressively promotes autonomous management control loops at all levels. Self-management is considered along four axes (Self-*): self configuration, self tuning (or self optimisation), self healing and self protection. Within a single component-based application or service this is enabled by a distributed component management service built on top of the overlay infrastructure.
- Federative and collaborative data services. To support flexible and dynamic VOs requires a novel approach to information sharing. The Grid4All data storage architecture provides three different levels of flexibility. (i) VO users can pool disk space over the network: either offered to each other from their own disks, or acquired on an open market. (ii) A VO has a associated virtual file system, a flexible federation of files names and file contents, exposed by VO users to one another. (iii) A semantic middleware layer allows users to update shared data in a flexible manner. Users may share content in either on-line or off-line mode; the semantic store resolves conflicting updates according to data semantics.

5. Developments and Results

The main Grid4All results consist of software frameworks, middleware and applications. The elements of the overall architecture of Grid4All VOs are presented on Figure 1. By analogy with Foster et al.'s decomposition of Grid systems [3], the Grid4All VO architecture can be decomposed as follows, from bottom to top:

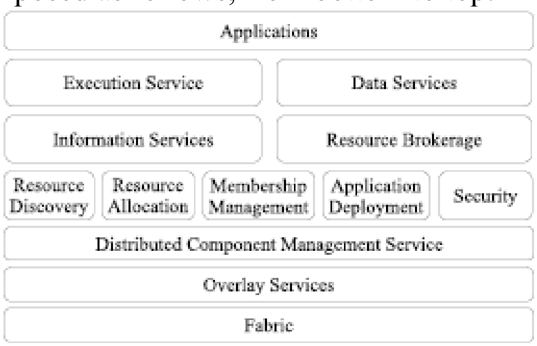

Figure 1. Overall Grid4All architecture.

The **Fabric** includes computers, storage, files, application binaries and sources, and other basic resources provided to the VO by its members. In what follows, the term 'resource' without further qualification refers to a fabric resource.

The Core VO Support Architecture comprises the following basic services:

- **Overlay** Services deal with connectivity of VO resources and elements. Participating nodes (i.e., computers) form a logical network called an overlay. We use structured overlay networks because of their inherent self-management properties and scalability. These services provides basic naming, communication, and data storage services that are used by upper layer services.
- The **Distributed Component Management** Service (DCMS) supports distributed VO-aware self-* services. DCMS allows the application delevoper to program self-* functionality of component-based services. In what follows, we refer to the latter as self-* code. DCMS is overlay-based, and it delegates resource discovery and allocation to VO management services introduced next.
- The **VO Management** Services that include Resource Discovery and Allocation, Membership Management, Application Deployment and Security services. VO Management Services maintain information about VO users, and based on that information decide on sharing of VO resources between VO users. It also provides the basic mechanism to start new services in a VO.

The **Information** Services provides match-making between semantic service descriptions and client requirements.

Resources model computational and storage elements needed to execute applications. The available resources on a VO have to be managed internally but they can be also brokered with other VO at the market place. The **Resource Brokering** Service facilitates interaction between VO [7, 9]. This service enables applications and VOs to find external resources as they need them. The brokering facility is a virtual resource market place where providers and consumers can meet and trade grid resources. This adds another dimension to the adaptability of VO to a changing demand.

Our support for collaborative and federative VOs focuses on data services for sharing storage and content, i.e., a **VO-oriented file system** (VOFS) and a semantic consistency middleware **Telex** facilitating the development of applications that handle shared data that can be concurrently modified with write-sharing and remote (disconnected) access, and an **Execution Service** for deploying computation tasks based on a bag-of-tasks model.

Supporting flexible and dynamic VOs requires a novel approach to information sharing. The Grid4All data storage architecture provides three different levels of flexibility:

(i) VO users can pool disk space over the network: either offered to each other from their own disks, or acquired on an open market.

(ii) A VO has a associated virtual file system, a flexible federation of files names and file contents, exposed by VO users to one another.

(iii) A semantic middleware layer allows users to update shared data in a flexible manner. Users may share content in either on-line or off-line mode; the semantic store resolves conflicting updates according to data semantics.

Applications use the lower layers, e.g., to obtain computation, storage and content resources or to achieve self-* behaviour.

The project aims at building proof-of-concept prototypes that will contribute to build a Democratic Grid (a middleware, a component framework, services) and a number of demonstrator self-managing applications. At the current stage, most components, API and alfa quality end-user oriented applications have been released (see in [1]) by June 2008. Integrated parts of the Grid4All environment are expected to be released by January 2009. After further evaluation, the project will deliver the final prototype systems by June 2009.

A collection of end-user oriented applications are being developed that use in different ways the API provided by the modules developed within the project:

- Collaborative File Sharing (CFS) allows users to collaborate, interact and share information. The collaboration can be at the same time or different time (asynchronous) and it presents folders, files, forums organized in a hierarchical structure that has been developed as an extension of the Firefox browser and uses several Grid4All API, particularly the data services (Telex to handle concurrent modifications and resolving conflicts, VBS+DFS to actually store and access the files in a distributed manner, and the security infrastructure to check the identity and permissions of participants).

- Collaborative Network Simulator Environment (CNSE), aims at supporting collaborative learning scenarios within the context of networking education. More specifically, CNSE allows students to carry out simulations of network scenarios described in simulation scripts. Such simulations generate trace files that generate animations of the behaviour of the different elements of the scenario that can be collaboratively visualized by students. Interestingly, CNSE can be employed to carry out parameter-sweep simulations (i.e. simulations of the same network scenario varying the value of one or more parameters). CNSE is a Java application with a collection of application-level services developed in Globus that uses Grid4All services for the distributed storage of simulation data, execution services for launching the simulation tasks, the Semantic Information service to locate services, and DCMS for self-management of some services.
- Emeeting is an on-line synchronous collaborative tool that allows sharing not only voice and video but also text, and other forms of sharing such as a group annotation and drawing tool and a polling tool. It is developed using Flash and Java for the client part, and a number of services that use the Grid4All API to store files, handle conflicts and check participants, among other potential uses.
- gMovie is a service accessible via a web interface, allowing distributed video transcoding on top of VO resources. Transcoding means changing the encoding and/or encapsulation format of the video. However we will also support some scale transformation. It uses the market and scheduling services to obtain resources and plan tasks, and the deployment service to install and start each worker process.
- Shared Calendar (SC) is an application that allows multiple participants to schedule meetings handling conflicts among concurrent and overlapping proposals for meetings or tasks. Each user has his own and independent calendar with private events, and he can organize meeting with other collaborators. SC is a Java application that uses Telex to resolve conflicts and VBS+DFS to store the files.

These applications demonstrate how support for different types of end-user tasks can be provided to end-users taking profit from the services of the Grid4All infrastructure (supporting groups [CNSE, CFS], group awareness [CNSE], sharing [CFS, CNSE, eMeeting], synchronous (same time) meetings [eMeeting], coordination [SC, CFS], and specific tasks [CNSE, gMovie]) relevant in the educational scenario selected in the project and applicable to the wider range of scenarios of interest for the project. They are developed using diverse software technologies and development environments (e.g. an extension of the Mozilla browser framework for CFS, Globus for CNSE, Flash for eMeeting). They make use of different aspects of the Grid4All API, covering directly or indirectly most of the functions.

Initial experiments with several parts of the system show the feasibility and the correct behavior of the whole set (e.g. the development, integration and testing of the Niche overlay network, the DCMS self-managing framework and the YASS storage service together [8]; the integration of CFS and Telex; the intregration and initial evaluation of the resource brokering elements to form the Grid4All Marketplace or GRIMP [9]). However, as [11] explains, the complexity of integrating so many components is a great challenge that may not be fully performed and evaluated within the scope of the project.

6. Business Benefits

Grid4All provides an opportunity for citizens to democratize computing by opening up participation by: lowering costs, facilitating usage and supporting collaboration and data sharing.

Potential users of Grid4All software are application developers, technology and service providers. Nevertheless end users may use the produced middleware, services and applications to setup virtual organizations and to use the tools for collaboration.

The DCMS framework simplifies programming of robust self-managing applications that could execute on unpredictable environments. This is one step towards Global computing where idle Internet resources can be reliably used by society and small business.

The market place facilitates the global trading of computing and storage resources and even services beyond the pooling that occurs within VO. This opens a global marketplace where anyone from an individual to a large utility computing provider can participate in this decentralized market either by selling or contributing their resources, or buying any additional amount of resources to perform certain computing tasks.

The Telex middleware simplifies programming of distributed collaborative applications. Telex helps developers by clarifying building blocks, providing consistency control and persistence management facilities. Using Telex a wide category of collaborative applications may be used in volatile environments.

Together, the API and the reference implementation being developed constitutes the basis for a sort of global operating system for a future decentralized distributed computing environment where computers can be easily traded and put together or pooled as resources, people can work together and share information, and developers can write collaborative applications without being aware of the constant reorganization and adjustment of the underlying system required to adapt to environmental changes, with the flexibility to obtain the required resources on demand from the global pool regulated by the marketplace.

The resulting environment opens a great opportunity for developers to use the collective computing power in the Internet and develop applications that use the Grid4All API to empower people in more efficient, cost-effective, dependable and sophisticated ways than the current browser-based web model based on a patchy and "thin" presentation layer at the end-user PC with a complex and usually centralized structure on the server side, with the HTTP protocol in between.

End-users, businesses and organisations can thus benefit from these new opportunities where dependable and effective applications can use the Internet to support collaboration and sharing in a similar but "more democratic" way than the dreams of Grid or Cloud computing.

7. Related work

The Grid4All vision and middleware is unique in adressing the needs of distributed computing for small organizations and educational users working together with a collection of middleware pieces that cover several key aspects in an integrated way. LaColla [5] or JXTA [6] follow a similar decentralized approach, however they do not consider self-management and specific support for collaborative work. In the Grid area, there are projects such as the GridBus [4] and SORMA [10] where marked mechanisms are introduced in Grid computing middleware as a way to adapt to changes in demand and supply of computational resources, but decentralization or self-management are out of scope which is a key aspect for the scenarios considered here. Our vision is close to the idea of Grid or Cloud computing but more prescriptive in several aspects such as decentralization, self-management or data services.

8. Conclusions

The Grid4All project presents an integrated vision, architecture, middleware and applications of a public and large scale Grid for everyone, supporting collaboration among groups (VO) of people who can interact and share work and computing resources among them, and trade resources with other groups. The applications and preliminary evaluation results with the first release of most components of the middleware shows promising results in how this decentralized Grid self-adapts to the dynamics of the online world with networks, computers and people who come and go, fail and recover and applications which varying loads and resource needs. However, further development and evaluation is required and planned until the end of the project in May 2009 to demonstrate that the architecture and implementation is feasible and works as expected for the target scenarios.

The Grid for everyone is an opportunity to go beyond the current Internet network to a seamless collaborative environment, as simple for everyone to use as plugging in a computer to both the power grid and the computing grid, and as useful to enable rich forms of collaboration, by creating groups, pooling and sharing computers, facilitating data sharing and concurrent modification of any type of data or documents, and being able to easily perform complex tasks using the resources of the Grid. Therefore the Grid4All project paves the way to the Grid to become the next ICT-based societal environment after the Internet and the Web.

Acknowledgement

This work will not be possible without the contributions from everyone participating on the Grid4All project. The list is too long to reproduce here, but they know who they are.

This work has been partially supported by the EU FP6-2005-IST-5 034567, and the Spanish Ministry of Education with grant P2PGrid TIN2007-68050-C03-01.

References

[1] Grid4All web site: http://www.grid4all.eu/

[2] Ian Foster et al. The Open Grid Services Architecture, Version 1.5, GFD-I.080, http://forge.gridforum.org/projects/ogsa-wg

[3] Ian Foster, Carl Kesselman, and Steven Tuecke. The anatomy of the grid: Enabling scalable virtual organizations. Int. J. High Perform. Comput. Appl., 15(3):200–222, 2001.

[4] GridBus project, http://www.gridbus.org

[5] Joan Manuel Marquès, Xavier Vilajosana, Thanasis Daradoumis, Leandro Navarro: LaCOLLA: Middleware for Self-Sufficient Online Collaboration. IEEE Internet Computing 11(2):56-64 (2007).

[6] Li Gong, JXTA: a network programming environment, Internet Computing, IEEE, Volume 5, Issue 3, May/Jun 2001, pp. 88 – 95.

[7] Nejla Amara-Hachmi, Xavier Vilajosana, Ruby Krishnaswamy, Leandro Navarro-Moldes, Joan Manuel Marquès: Towards an Open Grid Marketplace Framework for Resources Trade. OTM Conferences 2007:1322-1330.

[8] Noel de Palma, Konstantin Popov, Vladimir Vlassov, Joel Hoglund, Ahmad Al-Shishtawy and Nikos Parlavantzas, A Self-Management Framework for overlay-based applications, IEEE WET-ICE COPS Workshop, Rome, 2008.

[9] Ruby Krishnaswamy, Leandro Navarro, René Brunner, Xavier León, Xavier Vilajosana, Grid4All: Open Market Places for Democratic Grids, Gecon2008, The 5th International Workshop on Grid Economics and Business Models, Las Palmas, August 25, 2008.

[10] SORMA project (Self-Organizing ICT Resource Management), http://www.sorma-project.eu/

[11] Steve Muir, The Seven Deadly Sins of Distributed Systems, USENIX First Workshop on Real, Large Distributed Systems, http://www.usenix.org/events/worlds04/tech/muir.html