

FedGrid: An HLA Approach to Federating Grids

Son Vuong¹, Xiaojuan Cai¹, Juan Li¹, Sukanta Pramanik¹,
Duncan Suttles², and Reggie Chen²

¹ Computer Science Department, University of British Columbia
Vancouver, BC V6T 1Z4 Canada

{vuong, xjcai, juanli, pramanik}@cs.ubc.ca

² Magnetar Games Corp, Vancouver, BC Canada

{duncan, reggie}@magnetargames.com

Abstract. Research on Grids has received considerable attention in recent years. Whereas existing work on connecting the grids primarily focused on grid services, in our work we propose a unified approach, so-called *FederationGrid* (or *FedGrid* for short), which integrates both virtual organization (VO) and grid services, thus enabling formation of virtual communities on top of grids. FedGrid is based on the standard High Level Architecture (HLA) that enjoys the advantages of high-level information modeling, including re-usability of software components and real-time performance of the overall systems. In addition, FedGrid is a fractal grid, comprised of hierarchical HLA federations, e.g. *Realms* and *Lobby*, which supports well the concept of virtual communities and scalability.

Keywords: Grid, Federation, HLA, RTI, Grid Computing, Grid Services, Grid Access, Virtual Organization, Community Grid, Collaborative Computing.

1 Introduction

Grid Computing [1-2] has made a rapid progress in both concept and evaluation in recent years, which is evident in several large-scale applications in high-energy physics, earth science, and other areas [3-6]. It glues a great amount of distributed resources and powers together to allow for the provisioning of ultra extra time and resources to solve problems previously intractable and accelerate computing and process time. It also allows widely distributed organizations or branches of organizations to create a virtual organization to cooperate and share resources. The need for virtual organization support in the grids has been increasing as it allows and attracts more and more grid users to form the grids' virtual communities; hence enabling interactive and collaborative distributed applications to run in the grid environment. Our work focuses on federating grids via the High Level Architecture (HLA) [7], the US DoD Standard which has become an IEEE Standard defined for modeling and execution of interoperable distributed simulations. HLA has experienced tremendous growth over the past few years, expanding its uses in various distributed application areas such as multi-user games and distance learning.

The multidimensional advantages of grid computing have led to many new Grid projects in both scientific research and enterprise domains. However, since Grids are

owned and controlled by specific groups, and are built out of differing substrate technologies with perhaps incompatible communication systems, it is inevitable that Grids will be established as islands of resource collections. Therefore, although continued support of Open Grid Services Architecture (OGSA) and Open Grid Services Infrastructure (OGSI) is essential, interoperability standards are also needed to support the federation of distributed Grids developed by distributed team. The federation of grids is essential to deal with a range of important problems such as unified interface, interoperable/federated services, sharing information, communication across boundaries, and so on.

There has been existing work that addresses the problem of federating different types of grids, e.g. the Narada brokering [8], but such work mainly focuses on the core lower-level service, rather than the distributed object-based service level. Besides offering the message-based service at a higher-level than Narada, FedGrid via HLA can be implemented as a distributed P2P service rather than a client-server service, a fact which can enable rapid and widespread use of the technology on the Internet. By using HLA and the special virtual community supporting federations, so-called *Realms*, FedGrid can provide all benefits of Narada, and at the same time it can federate grids from both the perspectives of higher-level virtual organizations and grid service federations.

This paper is organized into five sections as follows. In Section 2 we briefly introduce HLA/RTI and existing work on how to federate grids and manage interactive grid applications. In Section 3, we present our proposal of the FedGrid framework. Finally, in Section 4 we describes some major work in progress and for future research on FedGrid and offer some concluding remarks.

2 Background and Related Works

The Grid concept embodies a vision of a global computing and communications infrastructure providing transparent, flexible and pervasive access to distributed resources.

2.1 Virtual Organization and OGSA

In effect, Grid enables resource sharing among dynamic collections of autonomous entities that can be individuals and institutions, so-called virtual organizations [9], which are governed by definite rules for authorization, authentication, resource access, etc. These distributed resources across heterogeneous virtual organizations, developed inevitably by distributed teams, need some inter-grid protocols to ensure interoperability. The OGSA (Open Grid Service Architecture) [10] addresses this issue by aligning grid technology with Web Service [11] technologies. OGSA defines a set of standard conventions and WSDL (Web Service Description Language) interfaces from which all Grid Services are implemented. This core set of consistent interfaces control the semantics of the Grid Service instance: its creation, naming, lifetime, communication and thereby facilitating the construction of higher-order services.

2.2 HLA/RTI and Distributed Interactive Simulation

While OGSA is still evolving, there are already existing solutions for distributed simulation systems such as HLA [7] that can be used as underlying framework for grids. The High Level Architecture (HLA) is a U.S. DoD-wide initiative to provide an architecture for support of interoperability and reusability of all types of models for Distributed Interactive Simulations. HLA has been adopted by IEEE as an open standard, the IEEE 1516. HLA models individual simulation participant as a federate and all federates in a simulation forms a federation. Federates interact with the federation through the interface between federates called Run-Time Infrastructure (RTI). The Key components of HLA are:

- *HLA rules*: principles and conventions that must be followed to achieve proper interaction among federates in a federation.
- *Interface specification*: describes RTI services and interfaces, and identify “callback” functions each federate must provide to accept the information from RTI.
- *Object Model Template (OMT)*: describes allowed structure of a Federate Object Model (FOM). It provides a standardized framework for describing HLA object models. It defines the format and syntax (but not content) of HLA object models. OMT is a meta-model for all FOMs. Its main components are:

Some well-known implementations of HLA/RTI include the DoD DMSO, the Pitch’s portable RTI, and more recently Magnetar’s RTI implementation in .NET over Directplay, called *Chronos*, and the UBC’s RTI implementation in Java over JXTA, called *Galaxy*.

2.3 HLA Approach to Grid

There has been research attempts to use grid services in support of HLA functions. Zajac *et al.* [12, 13] adopts a three-level approach to migrate HLA concept into Grid. In the first step they address the issue of HLA Federate Discovery mechanism by publishing them as a Grid Service. The second step deals with Grid-based data communication protocols like Globus GridFTP [14] as an alternative to RTI communication. Their final step encapsulates the RTI library within the Grid Service. Similarly, the CrossGrid project [15] considers HLA from the perspective of the Grid hosting an HLA Federation. In contrast, our FedGrid approach makes use of HLA in support of scalable grid services and their integration in a unified information model. In our FedGrid approach, HLA is used to form an integral part of the global-scale Grid architecture and to contribute to enhanced Grid functionality, scalability and performance.

3 System Architecture

From both the information modeling point of view and the fact that it can be implemented as a P2P service, HLA is deemed to be the ideal way to federate the

virtual organizations to support interactive distributed applications. The scalable data distribution mechanism via HLA also provides benefits to the FedGrid approach. The OGSi Open Grid Services Infrastructure takes a step in the direction of HLA by supporting service data variables which hold service state and may be exposed through a client side proxy. FedGrid incorporates this OGSi concept while proposing a major extension through the mechanism of a federate proxy "*Realm Engine*" which exposes a federated object model to clients. The benefits of this extension are considerable, as listed below and explained in the next 2 subsections.

1. Uniform information modeling of grid services through the HLA OMT standard.
2. Ability to easily create real-time updates between Realm peers using the HLA runtime infrastructure (RTI).
3. Proven integration technology for federating heterogeneous grid services.
4. Provide a mechanism for composing basic object models BOM [16] into aggregate models.

3.1 Overall Framework

FedGrid uses High Level Architecture (HLA) modeling and simulation standards to implement a super-scalable grid supporting real-time collaborative applications. The FedGrid is organized as hierarchical HLA federations, which form "Realms" defined by Federated Object Models (FOM). *Realms* correspond to the grid concept of virtual organization and are federation communities. For example, both the *chess realm* and the massive multiplayer online games (*MMOG*) *realm* require the support of a game lobby and repositories of resources or game digital assets. The HLA Run Time Infrastructure (RTI) can provide very advanced P2P services used in the Federation Grid's three-tiered network architecture consisting of core, super-peer, and peer federations. This is a view of federation grid from the perspective of application network building. Figure 1 shows this network architecture.

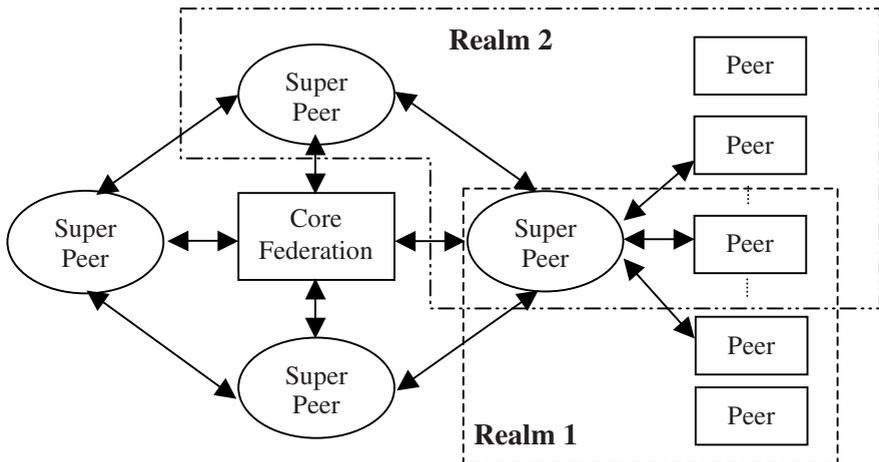


Fig. 1. FedGrid Network Architecture

The *core federation* provides authorization and discovery services. The *super-peers* are *Realm* service provider and they manage each federation community, while the peers are the entry point to the entire network. The super-peers, typically powerful servers, can be viewed to constitute an overlay network. *Realms* are virtual communities that constitutes of a set of peers and super-peers which abide by some definite FOM. They can also overlap as is shown in Figure 1.

Federation Grid further leverages HLA by introducing a single process federation (SPF) of *Agents* architecture for software *Engines* and through federation of engines into *Systems*. As is illustrated in Figure2, a System Federation is essentially a federation of Systems each of which is again a federation of Engines communicating using an internal RTI. The Engines are designed as an SPF of Agents and the Agents themselves talk via a blackboard. The detailed structure layering is described in the next section.

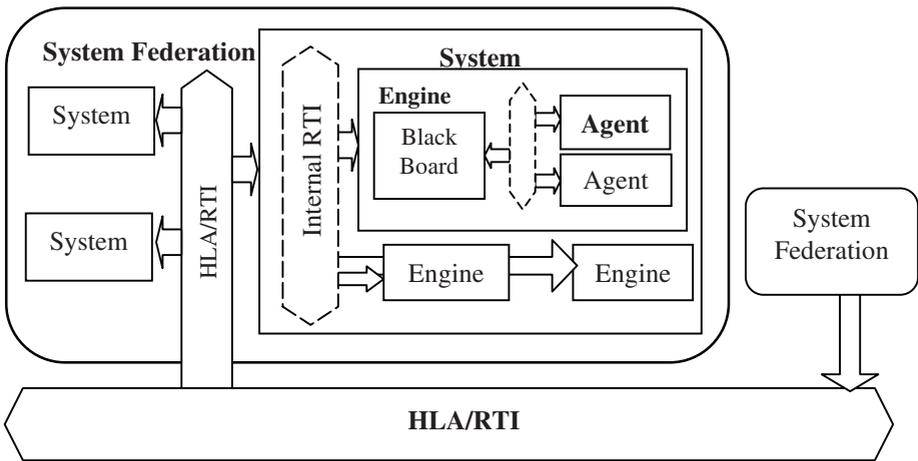


Fig. 2. Hierarchy of Federations

3.2 Federation Grid Layers

In the context of FedGrid organization, “*Realms*” are defined as collections of Virtual Organizations, which share a common information model. The basic idea of the realm model is illustrated in Figure 3. From the layer structuring point of view, *Engines*, which are single process federations (SPF) of *Agents* (software module), constitute the lowest rung in a HLA hierarchical federation model, which can be federated up to form *Systems*. Systems can in turn be federated to form a System Federation or so-called *Federation Community*. A FedGrid is formed by a collection of Federation Communities. A FedGrid can also be viewed as global virtual grid, analogous a fractal grid, that comprises an integration of grids within an application domain. These hierarchical federations are described as follows:

3.2.1 Engine as an Internal Federation

The HLA use is extended to include a SPF model for complex software “Engines” composed of plugin “Agent” federates. SPF acts like an internal bus for those agents. The idea is shown in Figure 3. The SPF agent paradigm is a promising avenue in defining re-usable AI component technology thereby reducing the cost of developing new engines. The re-use strategy also creates the basis for a true Meta product line technology. For this internal federation, there is an internal FOM for all the agents to follow. For example, in the realm engine, there are different messenger adapter like IM, Jabber, and different directory service adapter to LDAP server, X500 directory etc., and all these agents will use the “realm” Federate Object Models (FOM). Realms correspond to the grid concept of virtual organization and are federation communities.

Federation Grid engines are implemented using MAGNETAR (Metaprogrammable AGent NETwork ARchitecture) [17]. The structure of a MAGNETAR engine is a single process federation, which uses an internal HLA infrastructure engine (that acts as a Blackboard) to provide communication between “agent” plugin components. The first MAGNETAR engine is “Chronos”, a full HLA RTI implementation. Further development of MAGNETAR will include generic logic components, which can script behavior by driving the engines agents.

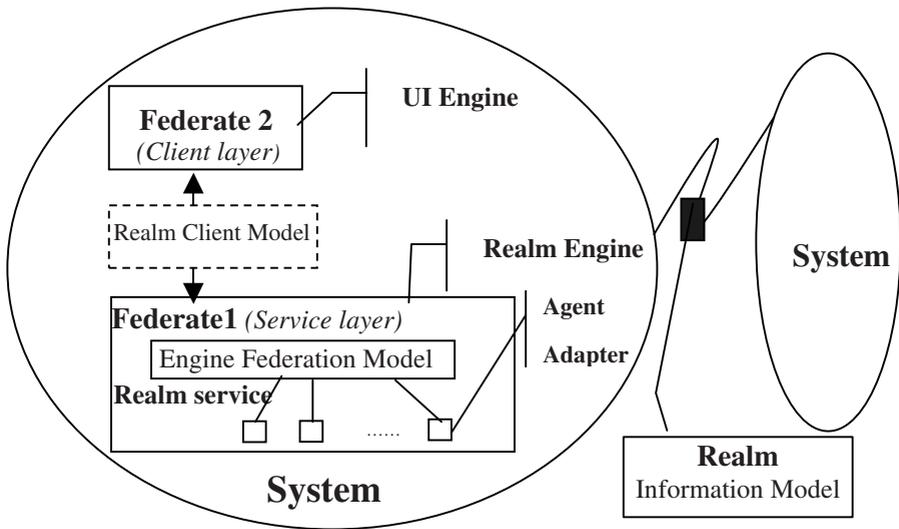


Fig. 3. The Realm Federation Model

3.2.2 System as a Federation of Engines

A System is a federation of engines. For example, in a game system, we need different types of engines, such as realm engine, game engine, realm user interface engine, network engine and so on, to work together to achieve the overall game functions.

Figure 3 above shows a system federation to illustrate the main idea of FedGrid. The UI engine and the realm engine follow the FOM of engine/service model, so that the realm could provide a unified interface to the client, no matter how various the services that the realm engine supports might be. The existing grid service (even from different types of grids) can be implemented as a grid service adapter (shown as an

agent/adaptor in Figure 3) in the realm engine so as to provide a uniform interface to the user.

3.2.3 Federation Communities

To form a virtual organization or even a global virtual grid, systems need to interact with each other. These systems will form a high-level federation, called a *Federation Community*. An example is a *Realm federation* for a realtime interactive application environment in which we need to provide realtime directory information for purposes such as effective game match making. In this case, the systems will exchange realtime directory information through the HLA Realm federation.

4 Work in Progress and Concluding Remarks

We present a unified approach to federate grids for supporting global-scale interactive applications over virtual organizations. In FedGrid, HLA forms an integral part of the Grid architecture and contributes to significantly improve the Grid functionality, reusability, scalability and performance. We have developed a framework and partially experimented with the FedGrid approach by implementing the underlying components. FedGrid can find its applications in a variety of global-scale integrated grid services such as **Community Grid**, **Content Grid** and **Computing Grid**. We have developed two working versions of RTI/HLA engines and some working version of the game lobby and repository. We are in the process of integrating them into a distributed chess prototype, including a *ChessGrid* system, to demonstrate the concept of FedGrid. Concurrently, we are working on a Web-based Flash FedGrid prototype, which is motivated by the increasing popularity of *Flash Games* and the fact that distributed web simulations has become increasingly important, falling in line with the XMSF initiative [18]. FedGrid is intended provide a complete XMSF-based solution. The popularity of “Flash Games” can be seen by noting that the Google search for “Flash Games” returns with 769000 hits as compared to 20,100 hits for “Wireless Games”, a supposedly very hot area, i.e. less than 3% of the hits for “Flash Games”.

The Flash FedGrid prototype includes a Flash-based game lobby with community service called *Flash Tryst* which implements the complete FedGrid Realm model for massive multiplayer game applications. Such prototypes can be adapted for interactive e-learning and other collaborative distributed applications. The Realm engine will interface with one or more P2P instant messaging services and provide a federated object model to the applications. Flash Tryst consists of a Flash client and a .Net Realm engine with *Chronos*, a P2P RTI engine developed by Magnetar, as the communication infrastructure between them. Another ongoing project is to build a Federated Lobby prototype to prove the concept of FedGrid. Yet another interesting project is to consider the issue of QoS and performance in HLA [19] and therefore in FedGrid.

The FedGrid approach is deemed unique and powerful as it promises to provide a unified information model for federating existing grids in an efficient, scalable and seamless manner. The FedGrid work presented in this paper serves to highlight some preliminary ideas to encourage further open concerted development in this direction.

While FedGrid offers a promising unified global scale approach to gluing (federating) existing grids together within its own information model, it can also be viewed as complementary to the OGSA approach as FedGrid aims at federating grids at the higher (information model) level whereas OGSA focuses on integrating the grids at one level below (below the HLA level). In fact, OGSA grids or other existing grids can be integrated into the FedGrid environment by simply developing the appropriate agent adaptors for the respective grids. Further research and experiment will be necessary to fully demonstrate and validate the FedGrid approach.

Acknowledgments. The work in this paper is supported in part by the joint grants from the Canadian Institute for Telecommunication Research (CITR) Network of Centres of Excellence, the Advanced Systems Institute of B.C., Telus Mobility, Proface INDE, and Magnetar Games. The authors also wish to thank Aleks Mandic, Mandi Paugh, and Ryan Levere from Magnetar Games for the fruitful discussions on HLA and RTI implementation.

References

1. Foster, I. And Kesselman, C. (eds.), *The Grid: Blueprint for a New Computing Infrastructure*, Morgan Kaufman, 1999.
2. Foster, I., *What is the Grid? A three checkpoints list*, GridToday Daily News and Information for the Global Community, July 22, 2002.
3. The Grid for UK Particle Physics, <http://www.gridpp.ac.uk>
4. The GriPhyN Project, <http://www.griphyn.org/>
5. NEES Earthquake Engineering Grid, <http://www.neesgrid.org/>
6. SCEC Earthquake Science Grid <http://www.scec.org>
7. IEEE 1516 Standard for Modeling and Simulation (M&S) High Level Architecture (HLA) - Federate Interface Specification 2000, ISBN 0-7381-2622-5, <http://shop.ieee.org/store/product.asp?prodno=SS94883>
8. Narada Brokering, <http://www.naradabrokering.org/>
9. Foster, I., Kesselman, C. and Tuecke, S., *The Anatomy of the Grid: Enabling Scalable Virtual Organizations*, International J. Supercomputer Applications, 15(3), 2001.
10. Foster, I., Kesselman, C., Nick, J.M. and Tuecke, S., *The Physiology of the Grid: An Open Grid Services Architecture for Distributed Systems Integration*, Open Grid Service Infrastructure WG, Global Grid Forum, June 22, 2002.
11. Web Services, <http://www.w3.org/2002/ws/>
12. Zajac, K., Tirrado-Ramos, A., Zhao, Z., Sloot, P. and Bubak, M., *Grid Services for HLA based Distribution Systems Framework*
13. Tirrado-Ramos, A., Zajac, K., Zhao, Z., Sloot, P. Albada, D.V. and Bubak, M., *Experimental Grid Access for Dynamic Discovery and Data Transfer in Distributed Interactive Simulation Systems*
14. Allcock, W., Bester, J., Bresnahan, J., Chervenak, A., Liming, L., Meder, S. and Tuecke, S., *GridFTP Protocol Specification*, GGF GridFTP Working Group Document, September 2002.
15. CrossGrid Project, <http://www.eu-crossgrid.org/>
16. Base Object Models, <http://www.boms.info>
17. GameXML, <http://www.gamexml.org>
18. Web Enabled Modeling and Simulation Network, <http://www.websim.net/>
19. Hui Zhao, "HLA Real-Time Extension", Concurrency and Computation: Practice and Experience Journal. (Selected by the IEEE DS-RT 2001 Workshop). ISSN: 1532-0626.