

Artificial Neural Networks and the Grid

Erich Schikuta and Thomas Weishäupl

Department of Computer Science and Business Informatics
University of Vienna
Rathausstraße 19/9, A-1010 Vienna, Austria
{erich.schikuta,thomas.weishaeupl}@univie.ac.at

Abstract. We introduce a novel system for the usage of neural network resources on a world-wide basis. Our approach employs the upcoming infrastructure of the Grid as a transparent environment to allow users the exchange of information (neural network objects, neural network paradigms) and exploit the available computing resources for neural network specific tasks leading to a Grid based, world-wide distributed, neural network simulation system, which we call *N2Grid*. Our system uses only standard protocols and services in a service oriented architecture, aiming for a wide dissemination of this Grid application.

1 Introduction

A Grid based computational infrastructure couples a wide variety of geographically distributed resources and presents them as a unified integrated resource which can be shared transparently by communities (virtual organizations).

The Grid started out as a means for sharing resources and was mainly focusing high performance computing. By the integration of Web Services as inherent part of the Grid infrastructure the focus evolved to the sharing of knowledge to enable collaborations between different virtual organizations or subjects.

The focus of the presented paper is the development of N2Grid, a neural network environment based on the Grid. It implements a highly sophisticated connectionist problem solution environment within a Knowledge Grid [1].

2 N2Grid Architecture

The N2Grid system is a neural network simulator using the Grid infrastructure as deploying and running environment. It is an evolution of the existing NeuroWeb [2] system. The idea of this system was to see all components of an artificial neural network as data objects in a database. Now we extend this approach and see them as objects of the arising world wide Grid infrastructure.

Figure 1 shows the overall application model of the N2Grid system in a final implementation phase. We assume a sophisticated Grid infrastructure, including independent resource brokers and replica manager. A N2Grid client does not care for the execution hosts or data sources. It can control an artificial neural network

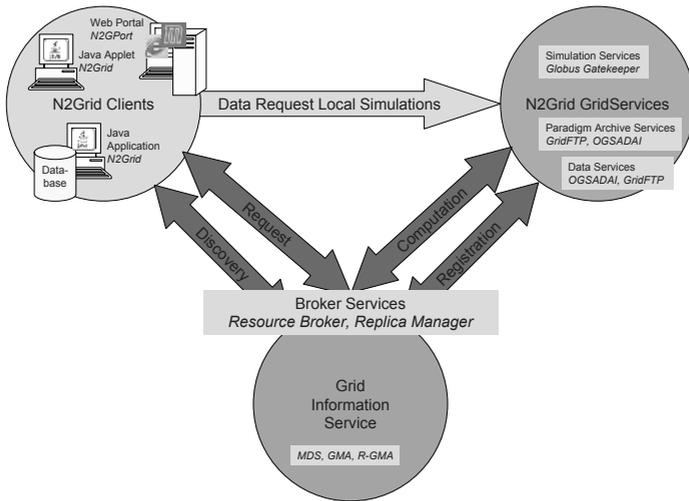


Fig. 1. N2Grid Application Model

simulation locally or remotely in the Grid infrastructure in a transparent way. Parts of the remote execution are the authentication to the Grid resources, the processing of the job descriptions and the usage of Grid data sources.

The components of the N2Grid are derived from the client-server model of NeuroWeb aiming for a novel, service oriented, tripartite Grid application. *N2Grid Services* realize Simulation Services, Paradigm Archive Services and Data Services. The end user can choose between three different *N2Grid clients*. These are a Java application with the possibility of direct data base connections, a Java Applet running in a web-browser, and a Web portal respectively. N2Grid services and clients are interconnected by standard Grid broker services, using standard Grid information services.

2.1 N2Grid Services

N2Grid Services are Grid Services hosted by the Grid infrastructure. They execute neural network simulation tasks (as creation, training and simulation), which are submitted to the Grid and do not consume local client resources. By the submission the work load is minimized on a local machine. Nevertheless the N2Grid system provides also the possibility for local artificial neural network simulations. We can run the following tasks remotely in the Grid:

1. Training of neural networks
2. Evaluation of neural networks
3. Processing of data by a neural network
4. Archiving of paradigms
5. Archiving of network objects (nodes, structure, weights, etc.)
6. Using data sources and archives

The *Simulation Service* accomplishes the training, evaluation and propagation function (task 1, 2, and 3) of the neural network simulator. A selected paradigm and network instantiation is executed on a Globus Gatekeeper (Version 2). Migrations are planned for Globus 3. The necessary data are provided by other N2Grid services.

Task 4 is implemented as N2Grid *Paradigm Archive Service*, where the users can find specific neural network paradigms (similar to a replica manager). More information about the available paradigm implementations provides the paradigm archive service by a directory implemented as OGSADAI database (XML), residing on the same site.

Neural network paradigms are implemented as Java classes using the Java Commodity Grid (CoG) Kit [3]. They are transferred by the GridFTP protocol. In special cases the execution of the class will only be allowed on a specific N2Grid Simulation Service (e.g. one specific remote site) to protect securely intellectual property of the paradigm implementation.

Task 5 and 6 are unified by the N2Grid *Data Services*. OGSADAI provides the access to a database storing all training-, evaluation-, propagation-data and network objects (nodes, structure, weights, etc.). To provide more flexibility, references (GridFTP URLs) to flat files can be registered in the database which can be accessed directly by the neural network simulation system.

2.2 N2Grid Clients

We propose three different clients as shown in Figure 1.

First, there exists a N2Grid *Java Application* client for an advanced user, who can run also a local database storing his own data. The user can extend the functionality of his client by including own paradigm Java classes without influencing the rest of the N2Grid system.

Second, we provide a N2Grid *Java Applet* executing within a standard Web browser having a similar user interface as the Java application but with limited functionality. Because of the sandbox principle local database and file accesses are not allowed.

Third, for the purpose of thin clients a simple web browser can be used as a front end of the N2Grid system by accessing a web-portal called *N2GPport*. It provides control over running simulation jobs on the N2Grid services and presents their results.

3 Use Cases – Scenarios

For the N2Grid system we propose several use cases, depending on the state of the dynamic and changing Grid infrastructure. Table 1 shows the categorization of the scenarios according to the Grid layers. The prototype is under development, three scenarios are already fully implemented, which are the scenarios *Data Pull (GET)*, *Data Push (PUT)*, and *Stand-alone Local Execution*. The other scenarios are implemented partly until now.

Table 1. Grid Layers Mapping

Layer	N2Grid Architecture	Use Case
Knowledge Grid 2-dimensional	N2Grid Paradim Serv. N2Grid Java clients N2GPort web portal client	Search Paradigm Search Net Object Create Neuroal System
Information Grid 1-dimensional	N2Grid Simulation Service Resource Broker Replica Manager	Directed Remote Execution Data-driven Remote Execution Computation-dirven Remote Ex. Paradigm-driven Remote Exec. Stand-alone Local Execution
Data Grid 0-dimensional	N2Grid Data Service N2Grid Paradigm Archive Service	Data Pull (GET) Data Push (PUT) Paradigm Pull (GET) Net object Pull (GET) Net object Push (PUT)

4 Conclusion and Future Research

We presented the N2Grid project, as a next step in the program evolutions for neural network simulation. It is a framework for the usage of neural network resources on a world-wide basis by the upcoming Grid infrastructure. Our system uses only standard protocols and services to allow a wide dissemination and acceptance. To reach the full capability of our system and to develop more sophisticated systems further research has to be done in the following two areas,

- The description of the paradigm has to be enhanced, to establish easier sharing between paradigm providers and customers. These semantic description is a key concept specified only rudimentary until now by a directory implementation. We will define a semantic paradigm description language by a pattern and/or scheme approach using XML.
- The actual N2Grid client controls single simulation runs. To allow the building of large connectionist systems consisting of several neural network instantiations (possibly of different paradigms) an extension of the N2Grid system is on the way, which allows to control a flow of simulations by a specific neural network workflow language.

References

1. Cannataro, M., Talia, D.: The Knowledge Grid. *Communications of the ACM* **46** (2003) 89–93
2. Schikuta, E.: NeuroWeb: an Internet-based neural network simulator. In: 14th IEEE International Conference on Tools with Artificial Intelligence, Washington D.C., IEEE (2002) 407–412
3. von Laszewski, G., Foster, I., Gawor, J., Lane, P.: A Java Commodity Grid Kit. *Concurrency and Computation: Practice and Experience* **13** (2001) 643–662