ACTIVE NETWORK APPROACH TO GRID MANAGEMENT & SERVICES

Alex Galis[1], Jean-Patrick Gelas[2], Laurent Lefèvre[2], Kun Yang[1]

[1] University College London, Department of Electronic and Electrical Engineering,
Torrington Place, London WC1E 7JE, UK, Tel +44-207-679 5738, Email: {agalis|kyang}@ee.ucl.ac.uk
[2] INRIA RESO / LIP - Ecole Normale Supérieure de Lyon - 46, allée d'Italie- 69364 LYON Cedex 07 – France
Tel: +33 (0)4 7272 8228 Email : Jean-Patrick.Gelas@ens-lyon.fr, Laurent.Lefevre@inria.fr

Extended abstract

ABSTRACT

Due to the large scale of Grid environment and its rapid expansion, both in Grid resources and Grid network environment, it is getting more imperative to provide a Grid management mechanism that can enable Grid Computing to adapt to various application requirements in a flexible and automated way. This paper proposes an Active Grid architecture and middleware for rapid and autonomic GRID service creation, deployment, activation and management.

1. INTRODUCTION

The widespread Internet is the living environment of Grid computing. As described in [Foster2], a basic premise of Open Grid Services Architecture (OGSA) is that everything is represented by a service, which is a ***network-enabled*** entity that provides capability to users. In addition there is a network paradigm shift dictated by the need of rapid and autonomic service creation, deployment, activation and management combined with context customization and customer personalization. Such motivation can be traced in different organizations, fora, and research activities as well as market forces. This paper aims to contribute to the state of the art of Grid research field by focusing on network, especially the enhancement of the network by using active networks technology to open up the functionalities of networks. And particularly, the policy-based Grid management largely maximizes the efficiency of this enhancement.

In a highly connected Internet world the needs for integrated services across distributed, heterogeneous, dynamic domains of different resources are getting more imperative. The providing of this integrated service can be technically very challenging due to two obvious facts: the large varieties of the underlying native platforms and the large varieties of the management systems for managing these platforms. Research on Grid computing was originated to cope with the first challenge, which has obviously resulted in a long list of Grid supporting platforms, among which Globus [Globus] attracts more attention and has been widely adopted as a Grid technology solution for scientific and technical computing [Foster2]. But Globus Toolkit is less concerned about the underlying network issues that are vital to a successful Grid environment. On the other hand, the complexity of Grid supporting environment, together with the various resources, also cast serious management issues. All these challenges must be addressed intelligently and effectively. This paper tends to initially practice the applicability of active networks technology to Grid supporting environment and the use of policy-based management method to Grid management.

While Active Network research has precisely tackled that problem domain in fixed and wireless network environments, the particular conditions of Grid networks have not sufficiently been taken into account. Tamanoir [Gelas00] aims to provide such a case study by focusing on the significant active network support for Grid computing from the network engineering's point of view.

Most challenging problems in Grid context come from heterogeneity of Grid resources and network elements, sheer largeness and inter-domain complexities of programming environment and service deployment [Foster1]. These problems can be solved in some extent using active network technology.

This paper proposes an Active Grid architecture [Gelas02] and middleware for rapid and autonomic GRID service creation, deployment, activation and management. This architecture relies on the support of active networks technology between distant sites.

The paper is structured as follows. Based on the discussion in this section, Section 2 analyses the requirements for Grid Management and Services that highlights the use of policy-based management method for overall Grid management and the use of active networks technology for speeding up and opening up the network layer. Then based on the analysis of OGSA, an active Grid architecture (together with its PBM middleware and active network middleware) is described in Section 3. Section 4 details the active networks support for policy-based Grid management and active networks support for Grid middleware services. And finally, Section 5 concludes the paper.

2. REQUIREMENTS FOR GRID SERVICES & GRID MANAGEMENT

2.1 Requirements for Grid Services

A distributed application running in a Grid environment requires various kinds of data streams: Grid control streams and Grid application streams. First of all, we can classify two kinds of basic Grid infrastructures:

2.1.1 Meta cluster computing

A set of parallel machines or clusters are linked together with IP networks (Internet) to provide a very large parallel computing resource. Grid environments like [Globus], [Polder] or [Netsolve] are well designed to handle meta-cluster computing session to execute long-distance parallel applications.

We can classify various network needs for meta-clustering sessions as follows:

- Grid environment deployment: the Grid infrastructure must be easily deployed and managed: OS heterogeneity support, dynamic topology re-configuration, fault tolerance, etc.
- Grid application management: Two kind of collective communications are needed: multicast and gather. The source code or binaries of applications, parameters, data can be multicast to a set of machines for distant running. Homogeneous multicast groups can be supported to deploy jobs on dedicated target machines. After the running phase, results of distributed tasks must be collected by the environment in a gathering communication operation.
- Grid support: The Grid environment must collect control data: node synchronization, node workload information. The information exchanged is also needed to provide high-performance communications between nodes inside and outside the clusters.

2.1.2 Large scale or global computing

These environments usually provide support on thousand of connected machines (like [Condor-G], [Entropia] [Nimrod-G] or [XtremWeb]).

We can classify various network needs for Global-computing sessions:

- Grid environment deployment: Dynamic enrolment of unused machines must be taken into account by the environment to deploy tasks over the mega-computer architecture.
- Grid application deployment: The Grid infrastructure must provide a way to easily deploy and manage tasks on distant nodes. To avoid the restarting of distributed tasks when a machine crashes or become unusable, Grid environments propose check-pointing protocols, to dynamically re-deploy tasks on valid machines.
- Grid support: various streams are needed to provide information to Grid environment about workload information of all subscribed machines. Machine and network sensors are usually provided to optimize the task mapping and to provide load-balancing.

These two Grid infrastructure can support various usage : computational Grid, Data Grid…

2.2 Policy-based Grid Management

End-to-end Grid services can be very complex in the Grid computing environment, and this raises the increasing requirement for the management of Grid system as a whole. Most current researches with this goal are carried out from the Grid resources themselves' point of view, with examples as Condor-G system [CondorG] and Nimrod-G Grid resource broker [NimrodG]. The research towards flexible Grid Services from the network point of view has yet been significantly taken into consideration. But network, as the transporting media for Grid services, is critical to guarantee fully efficient Grid services. Obviously, the bad quality of service in the networks can significantly obstruct the efficient provisioning of Grid services. Due to the complexity of Grid system, and the trend of getting more complex in both hardware/software and service requirements, the management of the overall Grid system itself and the services it provides in a flexible way is getting more and more important. It is time-consuming and error-prone for Grid administrator or resource manager/broker to configure his system manually. And it is extremely hard for him to configure his local resource while considering other domains in the whole Grid system.

Policy-based management (PBM) is a good candidate for such complex management environment. In comparison with previous traditional network management approaches, such as [TMN] or [TINA-C], PBM offers a more flexible, customisable management solution that allows controlled elements to be configured or scheduled on the fly, for a specific requirement tailored for a customer [ponder][Yang1]. The aim of PBM is to apply integrated management system so that system management, network management, and service management can cooperate in Grid computing. PBM method has been widely used in the IP network management field, whereas the application of PBM to the Grid management field has yet attracted much attention. Yang, et al [Yang2] presented a policy-based Grid management architecture supervising the overall Grid management, but without considering the Grid services.

2.3 Active Network for Grid Management and Grid Services

Based on the requirement analysis given above, we can see that the requirements cast by Grid services can be satisfied when dynamic injection of new functionalities into current Grid architecture is enabled; whereas the flexibility promised by PBM doesn't come without the automation of policy transit, policy enforcement and code downloading. Active Networks (AN), as an enabling technology, have been proposed as an architectural solution for the fast and flexible deployment of new network services. The basic idea of active networks is to enable third parties (end users, operators, and service providers) to inject application-specific services (in the form of code) into the networks. Applications are thus able to utilise these services to obtain required support in terms of element and network management resources, thus becoming network-aware.

Active network transforms the store-and-forward network into store-*compute*-and-forward [Tennenhouse] [Galis]. The innovation here is that packets are no longer passive but rather active in the sense that they carry executable code together with their data payload. This code is dispatched and executed at designated (active) nodes performing operations to change the current state of the node. Active network is distinguished from any other networking environment by the fact that it can be programmed. In this paper, this programmability is guided by the management policies.

3. ACTIVE GRID ARCHITECTURE AND ITS MANAGEMENT

3.1 Overall Architecture

Even though *Open Grid Services Architecture (OGSA)* is still in its draft and continues to be revised, it has attracted a lot of attention and is regarded as a promising means for providing pervasive services across the Internet. Building on concepts and technologies from the Grid and Web services communities, OGSA architecture defines a uniform exposed service semantics (the *Grid service*) and standard mechanisms for creating, naming, and discovering transient Grid service instances; it also provides location transparency and multiple protocol bindings for service instances and supports integration with underlying native platform facilities [Foster2]. Due to its many benefits, OGSA is adopted in this paper as a guideline for Grid services. It is further integrated with both policy-based management method and active network technology thus resulting an active Grid architecture as depicted in **Figure 1**. This architecture aims to provide mechanisms automatically adapting Grid network elements to different Grid services and the management of the Grid system itself.

In this paper, the method to add programmability to Grid management is to extend the widely used Grid supporting tool, Globus, which is also the powerful supporting tool for OGSA [Foster2]. Both policy-based Grid management middleware and active network middleware can be used by Grid supporting environment to facilitate the corresponding functionality so as to achieve better usage and management of different Grid resources such as massive storage resources, computing resources and special scientific instruments.

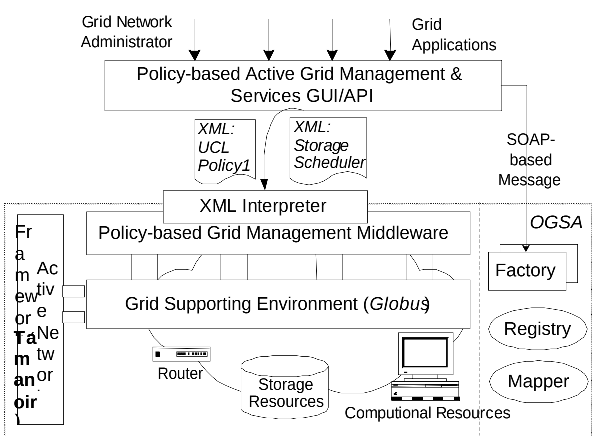

Figure 1: Overall Active Grid Architecture

In OGSA, the user interface to an OGSA-enabled system will typically be structured as a registry, one or more factories, and a handle mapper service. Each factory is recorded in the registry, to enable clients to discover available factories. When a factory receives a client request to create a Grid service instance, the factory invokes hosting-environment-specific capabilities to create the new instance, assigns it a handle, registers the instance with the registry, and makes the handle available to the handle mapper service. The implementations of these various services map directly into local operations [Foster2].

Active node, the core of the active network, is based on the *Tamanoir* Execution Environment. The Tamanoir [Gelas00] suite is an active network framework that primarily addresses the network management challenges. Based on standards, Tamanoir can easily be deployed in Grid networks. Detailed information about Tamanoir can be found in section 4.

3.2 Policy-based Management Middleware

Policy-based Grid management middleware is part of the Active Grid Management Environment and is used to control and manage the Grid environment by defining new policies, e.g., to apply a new DiffServ shaper, or modifying existing policies, e.g., to add a new massive storage accessing role.

In order to deploy PBM technology, a standardization process should be followed to ensure the interoperability between equipment from different vendors and, furthermore, PBM systems themselves from different developers. The framework and policy information model defined by Internet Engineering Task Force (IETF) Policy Framework Group [IETF-policy] gains wider popularity and is adopted as the baseline for the PBM system used in this paper.

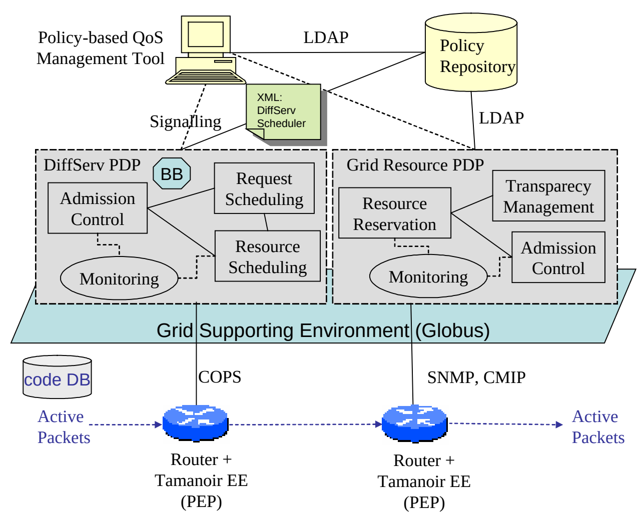

Figure 2: PBM Middleware Architecture

As illustrated in **Figure 2** from top down, the PBM system for Grid management mainly includes four components: policy management tool, policy repository, Policy Decision Point (PDP) and Policy Enforcement Point (PEP). Policy management tool serves as a policy creation environment for the administrator to define/edit/view policies in a high-level declarative language.

After validation, new or updated policies are translated into a kind of object oriented representation or so-called information objects and stored in the policy repository. The policy repository is used for the storage of policies in the form of LDAP (Lightweight Directory Access Protocol) directory. Once the new or updated policy is stored, signalling information is sent to the corresponding PDP, which then retrieves the policy and enforces it on PEP. There is a need of a transport protocol for communication between PDP and PEP so that PDP can send policy rules or configuration information to PEP, or read configuration and state information from the device. A wide range of protocols can be used here, such as SNMP, CMIP or COPS (Common Open Policy Service), among which COPS is becoming the standard.

A draft version of object oriented information model has been designed to represent the Grid policies. This information model is based on the IETF PCIM (Policy Core Information Model) [PCIM] but with addition of Grid resources management information and deduction of some rarely used classes to make the whole information model easy to implement. Due to the space limitation, this model will be introduced in this future paper. Furthermore, policies are represented by XML during its transit due to XML's built-in syntax check and its portability across the heterogeneous platforms.

All PDPs, such as DiffServ PDP and Grid Resource PDP, are integrated with Grid supporting tool, Globus [Globus]. There is also a PDP manager to coordinate the cooperation among different PDPs so as to support some complex Grid services which requirement the cooperation of more than one PDP. More detailed information about this PBM middleware was described in [Yang1]

Various services for Grid application can be introduced by defining new policies, e.g., to apply a new massive storage scheduler, or modifying existing policies. Then the Java classes for fulfilling these policies, which abide by class hierarchy and naming rules of policy information model developed within this system, can be instantiated by storage PDP according to these policies. And this Java bytecode is encapsulated into Tamanoir active packets and is delivered to the corresponding active nodes to fulfil the management tasks.

4. ACTIVE NETWORKS SUPPORT

4.1 Active Networks Supporting Environment: Tamanoir

The Tamanoir [Gelas00] suite is a complete software environment dedicated to deploy active routers and services inside the network. Tamanoir Active Nodes (TAN) provide persistent active routers, which are able to handle different applications and various data stream (audio, video,..) at the same time (multi-thread approach). The both main transport protocol TCP and UDP are supported by TAN. We currently rely on the [ANEP] (Active Network Encapsulated Protocol) format to send data over the active network (Figure 3).

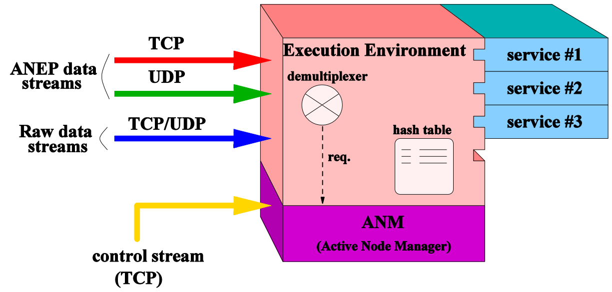

Figure 3: Tamanoir Active Node (TAN)

The Execution Environment relies on a demultiplexer receiving active packets and redirecting these packets towards the adapted service in function of a hash key contained in packets header. New services are plugged in the TAN dynamically. The Active Node Manager (ANM) is dedicated to deployment of active services and to update routing tables.

We choose to use a portable language for the active networks in order to facilitate users to define/ write their own services. Thus, the Tamanoir execution environment running in user space is entirely written in Java, which provides a great flexibility and is shipped with standard library (recent JVM releases (>= 1.3.x) give excellent performance for the mainstream hardware architecture, mainly due to the improvements of Just-In-Time (JIT) compilation).

Experiments of Tamanoir on Gbits networks have demonstrated the ability to support large number of active services deployed on TAN and to provide raw performances sufficient for high performance networks.

4.2 Active Networks Support for Dynamic service deployment

Each service is written in Java and inherited from a generic class called *Service*, itself inherited from the Java *Thread* class. Thus, each service is executed in an independent thread. For a given service, with TCP active streams, a thread service is dedicated for each stream while with UDP only one dedicated thread processes all streams. A given service can be applied on TCP or UDP active streams without change.

The injection of new functionalities, called services, is independent from the data stream: services are deployed on demand when streams reach an active node, which does not hold the required service. Two services deployment are available: by using a service repository, where TANs send all requests for downloading required services, by deploying service from TAN to TAN (TAN query the active node that sends the stream for the service). In order to avoid single point of failure service repository can be mirrored and replicated. When the service is available on a node, it is ready to process the stream.

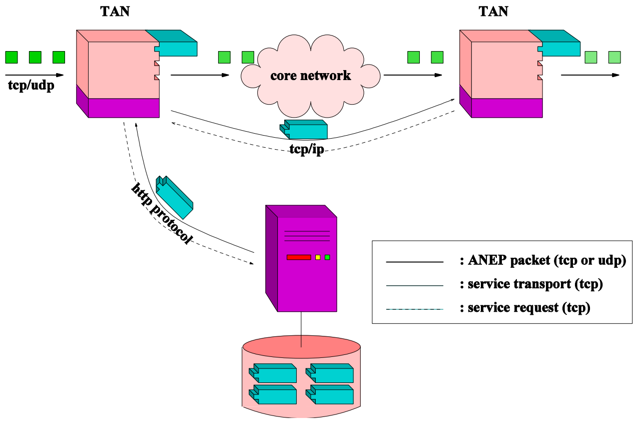

Figure 4: Examples of services deployment

To support most of Grid applications, the Active Grid architecture must deal with the two main Grid configurations:

4.2.1 Meta cluster computing

In this highly coupled configuration, an active node is mapped on network head of each cluster or parallel machine. This node manages all data streams coming or leaving a cluster. All active nodes are linked with other AN mapped at backbone periphery. An Active node delivers data streams to each node of a cluster and can aggregate output streams to others clusters of the Grid.

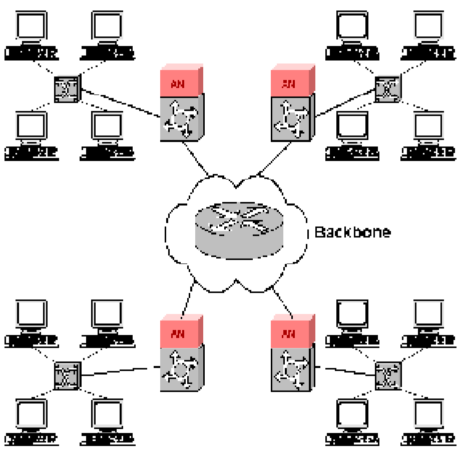

Figure 3: Meta cluster computing Active Grid Architecture

4.2.2 Global computing

In this loosely coupled configuration, an active node can be associated with each Grid node or can manage a set of aggregated Grid nodes. Hierarchies of active nodes can be deployed at each network heterogeneity point. Each AN manages all operations and data streams coming to Grid Nodes: subscribing operations of voluntary machines, results gathering, nodes synchronization and check-pointing. For both configurations, active nodes will manage the Grid environment by deploying dedicated services adapted to Grid requirements: management of nodes mobility, dynamic topology re-configuration and fault tolerance.

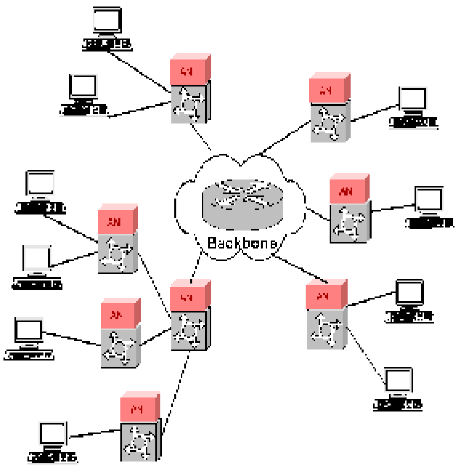

Figure 6: Large scale / Global Computing Active Grid Architecture

Using active Grid architecture can improve the communications needs of Grid applications:

- Application and data deployment: to efficiently deploy applications and to send large set of data parameters to distant jobs, active reliable multicast protocols can reduce the transport of applications (source code, binaries, bytecode..) by minimizing the number of messages in the network.. ARM will also be useful in complement to Grid Tools (like GridFTP) to massively deploy parameters and application data to distant architectures. Active node will deploy dedicated multicast protocols and services and guarantee the reliability of deployment by using storage capabilities of active nodes.

- Grid support: the Active architecture can provide information to Grid framework about network state and task mapping. Active nodes must be open and easily coupled with all Grid environment requirements. Active nodes will implement permanent Grid support services to generate control streams between the active network layer and the Grid environment.

- Wide-area parallel processing: with the emergence of grid parallel applications, tasks will need to communicate by sending computing data streams with QoS requests. The A-Grid architecture must also guarantee an efficient data transport to minimize the software latency of communications. Active nodes deploy dynamic services to handle data streams: QoS, data compression, ``on the fly'' data aggregation..

- Coupled (Meta) Application: the Active architecture must provide heterogeneity of services applied on data streams (data conversion services..). End to end QoS dynamic services will be deployed on active nodes to guarantee an efficient data transport (in terms of delay and bandwidth).

Most of services needed by Grid environments: high performance transport, dynamic topology adapting, QoS, on-the-fly data compression, data encryption, data multicast, data conversion, errors management must be easily and efficiently deployed on demand on an Active Grid architecture.

5. CONCLUSIONS AND FUTURE WORK

This paper contributes to the definition and development of the active Grid architecture and technologies needed for rapid and autonomic GRID service creation, deployment, activation and management.

It analyses the requirements for Grid Management and Services that highlights the use of policy-based management method for overall Grid management. It highlights the use of active networks technology in Grids for opening up the network layer by the use of the TAMANOIR active network node systems. An OGSA compatible active Grid architecture, together with its Policy Based Management middleware and active network middleware is proposed and currently under development. Final paper will present first experimental results obtained with Tamanoir active environment as support for Grid infrastructure and deployment of Policy Based Middleware.

6. ACKNOWLEDGEMENTS

This paper partly describes work in progress in the context of the EU IST project CONTEXT [CONTEXT] The IST programme is partially funded by the Commission of the European Union. This research is also supported by French RNTL Etoile project.

7. REFERENCES

[ANEP] The Active Network Encapsulation Protocol (ANEP). http://www.cis.upenn.edu/~switchware/ANEP/

[Condor-G] Frey J., Tannenbaum T., Foster I., Livny M., and Tuecke S. "Condor-G: A Computation Management Agent for Multi-Institutional Grids", Proceedings of the Tenth IEEE Symposium on High Performance Distributed Computing (HPDC10), San Francisco, USA, August 2001.

[CONTEXT] CONTEXT Project WWW Server: http://context.upc.es

[FAIN] FAIN Project WWW Server: – http://www.ist-fain.org

[Foster1] Foster, I., Kesselman, C. and Tuecke, S. The Anatomy of the Grid: Enabling Scalable Virtual Organizations. International Journal of High Performance Computing Applications, 15(3): 200-222. 2001. www.globus.org/research/papers/anatomy.pdf

[Entropia] www.entropia.com

[Foster2] Ian Foster, Carl Kesselman, Jeffrey M. Nick and Steven Tuecke1. The Physiology of the Grid: An Open Grid Services Architecture for Distributed Systems Integration. Open Grid Service Infrastructure Working Group (OGSI-WG) of Global Grid Forum. Available at: http://www.ggf.org/ogsi-wg/drafts/ogsa_draft2.9_2002-06-22.pdf

[Galis] Galis A., Plattner, B., Smith, J.M., Denazis S., Moeller E., Guo, H., Klein C., Serrat J., Laarhuis J., Karetsos, G.T., Todd, C. "A Flexible IP Active Networks Architecture". Proceedings of International Workshop on Active Networks, Tokyo, October 2000;

[Gelas00] Gelas, JP., Lefevre, L., "TAMANOIR: A High Performance Active Network Framework", Active Middleware Services, Kluwer Academic Publishers ISBN 0-7923-7973-X, August 2000; in Springer-Verlag, ISBN 3-540-41179-8.

[Gelas02] Gelas, JP., Lefèvre, L., "Towards the design of an active grid". In Lecture Notes in Computer Science, editor, *Computational Science - ICCS 2002*, volume 2230, pages 578-587, Amsterdam, The Netherlands, April 2002. ISBN 3-540-43593-X.

[Globus] Foster I. and Kesselman C. "Globus: A Toolkit-Based Grid Architecture". Foster, I. and Kesselman, C. (eds.), The Grid: Blueprint for a New Computing Infrastructure. Morgan Kaufmann, 1999. pages 259-278.

[IETF] IETF Policy workgroup web page: http://www.ietf.org/html.charters/policy-charter.html

[Netsolve] Netsolve project: http://icl.cs.utk.edu/netsolve/

[Nimrod/G] Buyya R. "Nimrod/G: An Architecture for a Resource Management and Scheduling System in a Global Computational Grid". Proc. 4th Int'l Conf. on High Performance Computing in Asia-Pacific Region (HPC Asia 2000), IEEE CS Press, Los Alamitos, USA, 2000.

[Polder] Polder Computing Environment http://www.science.uva.nl/research/scs/SCS4.html

[PCIM] IETF PCIM draft on web: http://www.ietf.org/internet-drafts/draft-ietf-policy-pcim-ext-08.txt, 2002

[FORCES] IETF FORCES (Forwarding and Control Elements Separation) working group www.ietf.org/FORCES

[Ponder] N. Damianou, N. Dulay, E. Lupu, M Sloman, "The Ponder Specification Language" Workshop on Policies for Distributed Systems and Networks (Policy2001), HP Labs Bristol, 29-31 Jan 2001 - http://www.doc.ic.ac.uk/~mss/Papers/Ponder-Policy01V5.pdf

[Tennenhouse] D. Tennenhouse, D. Wetherall – "Towards an active network architecture" Computer Communications Review, 26, 2 (1996), pp 5-18

[TMN]: www.itu.int/TMN/

[TMForum]: www.tmn.org/

[TINA]: www.tinac.com/

[Yang1] Kun Yang, Alex Galis, Telma Mota and Stylianos Gouveris. Automated Management of IP Networks through Policy and Mobile agents. - Proceedings of Fourth International Workshop on Mobile Agents for Telecommunication Applications -MATA2002: pp 249-258. LNCS-2521, Springer, ISBN: 3-540-00021-6. Barcelona, Spain, October 2002.

[Yang2] Kun Yang, Alex Galis, Chris Todd. Policy-based Active Grid Management Architecture. Proceedings of 10th IEEE International Conference on Networks (ICON02): 243-248, IEEE Press. ISBN: 0-7803-7533-5. Singapore, August 2002.