MANAGEMENT REVIEW

Grid Computing for Different Applications

Sanjeev Narayan Bal

Asst.Professor
Dept.of Comp.Science
TACT, Bhubaneswar

Abstract

Grid Computing delivers on the potential in the growth and abundance of network connected systems and bandwidth: computation, collaboration and communication over the Advanced Web. At the heart of Grid Computing is a computing infrastructure that provides dependable, consistent, pervasive and inexpensive access to computational capabilities. The use of grid computing in the context of e-learning shows what advantages a utilization of grid computing may have to offer and which applications could benefit from it. In capturing knowledge and heuristics about how to select application components and computing resources, and using that knowledge to generate automatically executable job workflows for the Grid. In one of the latest developments in Grid Computing Peer-to-Peer (P2P) networks have become fashionable for the Grid system architecture. The current implementations are either Data Grids or CPU sharing Grids.

Keywords

Grid Computing, network connected systems, application components

Introduction

Grid Computing delivers on the potential in the growth and abundance of network connected systems and bandwidth: computation, collaboration and communication over the Advanced Web. At the heart of Grid Computing is a computing infrastructure that provides dependable, consistent, pervasive and inexpensive access to computational capabilities. By pooling federated assets into a virtual system, a grid provides a single point of access to powerful distributed resources.

Researchers working to solve many of the most difficult scientific problems have long understood the potential of such shared distributed computing systems. Development teams which focused on technical products, like semiconductors, are using Grid Computing to achieve higher throughput. Likewise, the business community is beginning to recognize the importance of distributed systems in applications such as data mining and economic modeling.

With a grid, networked resources — desktops, servers, storage, databases, and even scientific instruments — can be combined to deploy massive computing power wherever and whenever it is needed most. Users can find resources quickly, use them efficiently, and scale them seamlessly.

The Grid Concept

The term ‘grid’ is variously used to describe a number of different, but related, ideas, including utility computing concepts, grid technologies, and grid standards. In this paper the term ‘Grid is used in the widest sense to describe the ability to pool and share Information Technology (IT) resources in a global environment in a manner which achieves seamless, secure, transparent, simple access to a vast collection of many different types of hardware and software resources, (including compute nodes, software codes, data repositories, storage devices, graphics and terminal devices and instrumentation and equipment), through non-dedicated wide area networks, to deliver customized resources to specific applications.

At the most general level Grid is independent of any specific standard or technology. Any practical grid is realized through specific distributed computing technologies and standards that can support the necessary interoperability. Today, there are no universally agreed grid standards, but there are freely available, open source and proprietary grid technologies that implement emerging standards recommendations. Separate web services standards are also emerging which have many grid-like capabilities. Indeed grids are already being built by integrating and enhancing web standards technology.

Practical Realizations

Practical grids are generally described in terms of layers (see Fig 1). The lowest layers (the 'platform') comprise the hardware resources, including computers, networks, databases, instruments, and interface devices. These devices, which will be geographically distributed, may present their data in very different formats, are likely to have different qualities of service (e.g. communication speeds, bandwidth) and are likely to utilize different operating systems and processor architectures. A key concept is that the hardware resources can change over time - some may be withdrawn, upgraded or replaced by newer models, others may change their performance to adapt to local conditions - for example restrictions in the available communications bandwidth.

The middle layers (sometimes referred to as 'middleware') provide a set of software functions that 'buffer' the user from administrative tasks associated with access to the disparate resources. These functions are made available as services and some provide a 'jacket' around the hardware interfaces, such that the different hardware platforms present a unified interface to different applications. Other functions manage the underlying fabric, such as identification and scheduling of resources in a secure and auditable way. The middle layer also provides the ability to make frequently used patterns of functions available as a composed higher-level service using workflow techniques.

The highest layers contain the user 'application services'. Pilot projects have already been carried out in user application areas, such as life sciences (e.g. computational biology, genomics), engineering (e.g. simulation and modeling, just in time maintenance) and healthcare (e.g. diagnosis, telematics). These services could include horizontal functions such as workflow (the linkage of multiple services into a single service), web portals, data visualization and the language/semantic concepts appropriate to different application sectors.

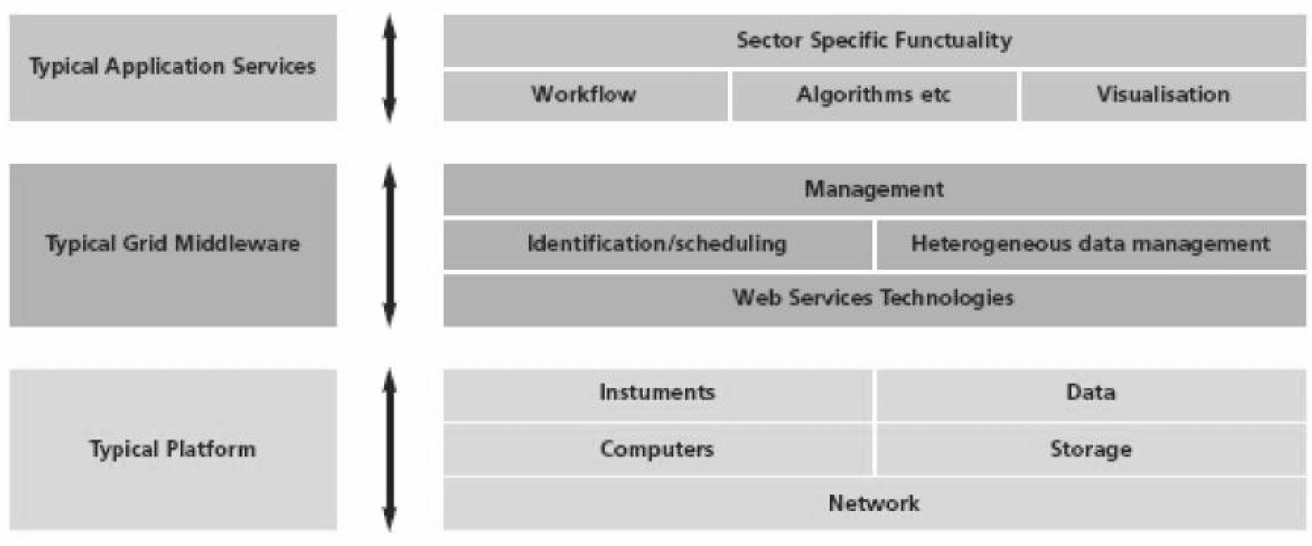


Figure 1: Simplified Grid Architecture

Grid Developments and Deployment

A key issue facing the industry is the timing and mode of deployment of Grid technology to ensure that it is sufficiently mature to deliver the expected business benefits. There is emerging evidence that the technology can achieve significant operational benefits (e.g. in telemedicine), improvements in performance (e.g. in climate modeling and genomics) and a significant reduction in costs. Nevertheless, current grid technologies are not yet viewed as sufficiently mature for industry scale use, and remain largely unproven in terms of security, reliability, scalability, and performance.

Short term

For the short term (within the next two years), Grid is most likely to be introduced into large organizations as internal 'Enterprise grids', i.e. built behind firewalls and used within a limited trust domain, perhaps with controlled links to external grids. A good analogy would be the adoption into business of the Internet, where the first step was often the roll out of a secure internal company 'Intranet', with a gradual extension of capabilities (and hence opportunity for misuse) towards fully ubiquitous Internet access. Centralized management is expected to be the only way to guarantee qualities of service.

Typically users of this early technology will be expecting to achieve IT cost reduction, increased efficiency, some innovation and flexibility in business processes. At the same time the distinction between web services and grid services is expected to disappear, with the capabilities of one merging into the other and the interoperability between the two standards being taken for granted.

Medium Term

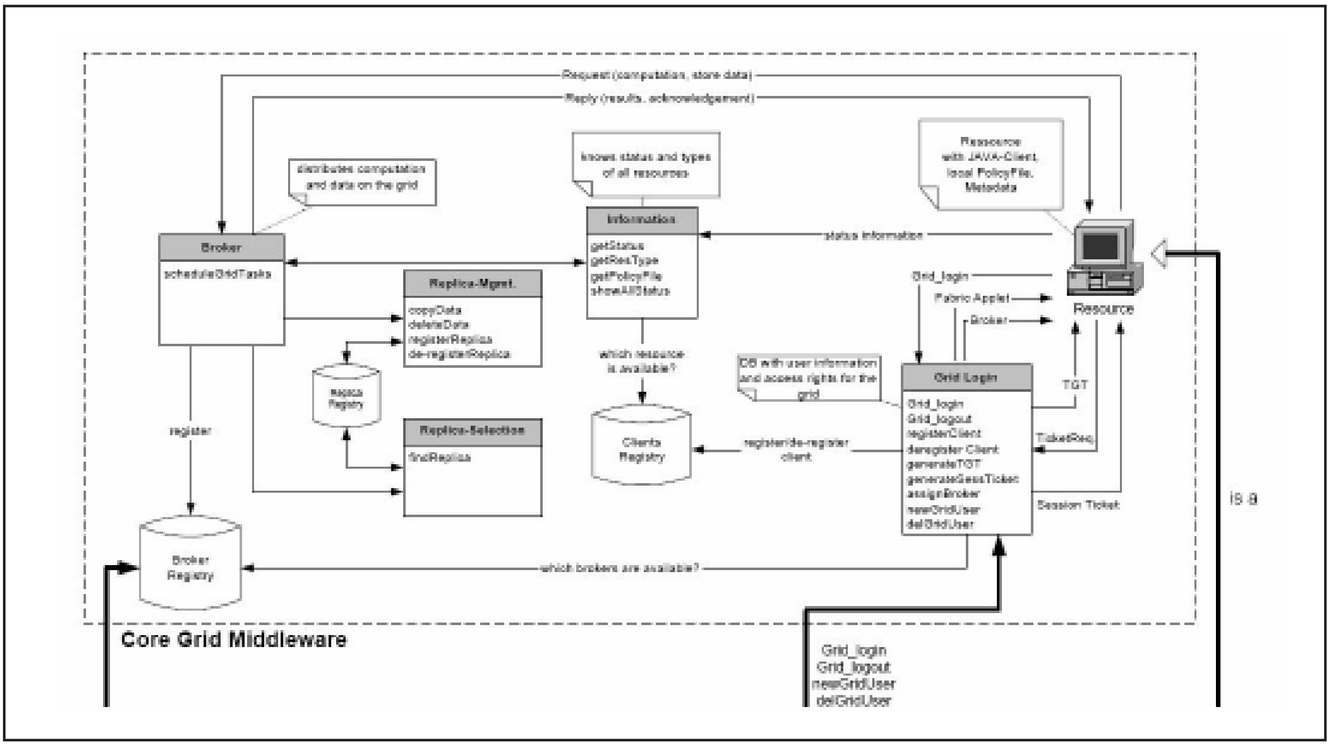
In the mid term (say a five year timeframe) expect to see wider adoption – largely for resource visualization and mass access. The technology will be particularly appropriate for applications that utilize broadband and mobile/air interfaces, such as on-line gaming, 'visualization-on-demand' and applied industrial research. The emphasis will move from use within a single organization to use across organizational domains and within Virtual Organizations, requiring issues such as ownership, management and accounting to be handled within trusted partnerships. There will be a shift in value from provision of computer power to provision of information and knowledge. At the same time open standards based tooling for building service oriented applications are likely to emerge and Grid technology will start to be incorporated into off-the-shelf products. This will lead to standard consumer access to virtualized compute and data resources, enabling a whole new range of consumer services to be delivered.

Long term

In the longer term, Grid is likely to become a prerequisite for business success -central to business processes, new types of service, and a central component of product development and customer solutions. A key business change will be the establishment of trusted service providers, probably acting on a global scale and disrupting the current supply chains and regulatory environments.

E-Learning Grids

There are many conceivable applications for e-learning grids. Medicine students could use photo-realistic visualizations of a complex model of the human body to prepare for practical exercises. Such visualizations, computed in real-time, could improve the understanding of the three-dimensional locations of bones, muscles, or organs. Students should be able to rotate and zoom into the model and get additional information by clicking on each element of the model. With more advanced functionality such as virtual surgery, students could be provided with the possibility to grab, deform, and cut model elements (e.g.organs) with the click of a mouse. In biology courses the ability of grids to integrate heterogeneous resources could be used to integrate an electron microscope into the grid. We mention that the technical feasibility of this approach has already been demonstrated in the TeleScience project. However, this project could be widely extended to integrate the controls and output of the electron microscope into a learning environment so that students can be assigned tasks or read subject-related texts while operating the microscope. Similarly, in engineering courses complex simulations, e.g.in a wind channel, can be made accessible for each student by using grids.



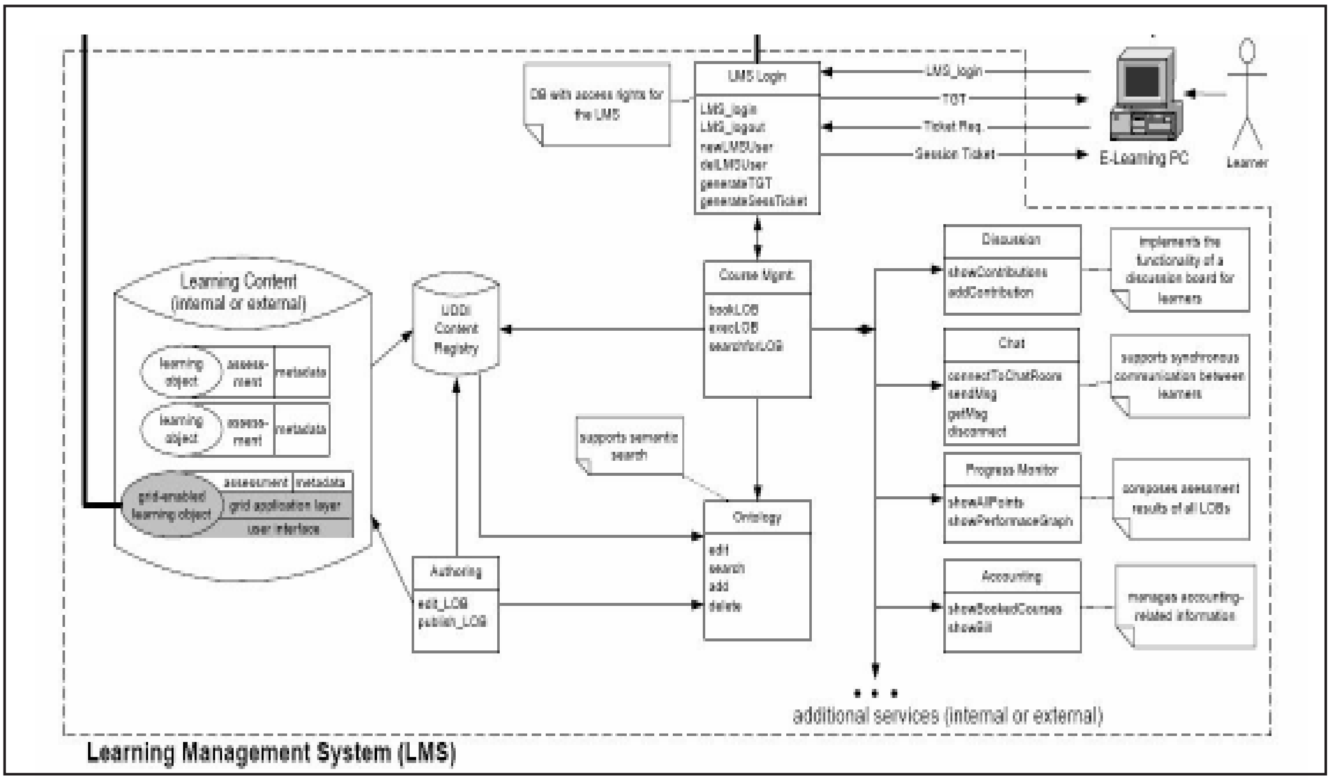


Figure2. Architecture of an E-Learning Grid.

We next outline an architecture for e-learning grids. To demonstrate the technical feasibility, the architecture will be

kept as simple as possible. It contains a Learning Management System (LMS) as well as grid middleware, which are both based on Web services and grid services, respectively (cf. Figure 2). In this figure, grid and Web services are depicted as rectangles containing a name as well as the most important operations. Note that grid services (with grey

name fields) can easily be distinguished from Web services. The LMS interacts transparently with the grid middleware so that a learner is not aware of the grid. Furthermore, the architecture is designed in such a way that a learner only needs a Java-enabled Internet browser to use both the LMS and the grid. The architecture shown in Figure 2.

Knowledge based approach Grids

Here the approach is twofold. First, we use declarative representations of knowledge involved in each choice of the workflow generation process. This includes Knowledge about how application components work, characteristics and availability of files, capabilities of the resources available, access control policies, etc. Second, this knowledge is uniformly available to the system at any point during workflow generation. This allows the system to make decisions and assignments in a flexible manner that

- takes into account previous and future choices , searching for a low-cost workflow configuration that satisfies the requirements from the user.
- is feasible in the given execution environment, and
- can adapt to changes in the overall system state.

Figure 3, illustrates our approach. Users provide high level specifications of desired results, as well as constraints on the components and resources to be used. These requests and preferences are represented in the knowledge base. The Grid environment contains middleware to find components that can generate desired results, the input data that they require, to find replicas of component files in specific locations, to match component requirements with resources available, etc. The knowledge currently used by Grid middleware (resource descriptions, metadata catalogs to describe file contents, user access rights and use policies, etc) would also be incorporated in the knowledge base. The system would generate workflows that have executable portions and partially specified portions, and iteratively add details to the workflow based on the execution of the initial portions of it and the current state of the execution environment. Much knowledge concerning descriptions of components, resources and the system's state is available from a variety of Grid middleware, as we describe in the next section. However, one must be an experienced Grid user to run jobs on the Grid, which means that much additional knowledge needs to be represented about what terms mean and how they related to one another. For example, an application component may be available in a file where it has been compiled for MPI. MPI is a Message Passing Interface, which means that the source includes calls to MPI libraries that will need to be available in the host computer where the code is to be run. Even a simple piece of Java code implies requirements in the execution host, namely that the host can run JVM (Java Virtual Machine).Our contribution is to organize this knowledge and reason about it within a uniform framework.

The peer to peer Market Grid(Model)

The peer to peer overlay network of "Grid" is defined by a connected graph. The nodes of the graph are the computing resources connected to the Grid and are either

- Buyer. This nodes tries to acquire a resource and bids a maximum price it is willing to pay.
- Seller. A node that sells exclusive access to its resource for a minimum price.

Either nodes type is un-satisfied when they are looking for or are offering service. Otherwise they are in a satisfied state. In the model a buyer node sends messages to its nearest neighbours informing it that it wants to purchase a resource unit at a price p_b . It also forwards messages from its neighbours until a preset time to live (TTL) has expired. Seller nodes look at incoming messages and decide whether their own price p_s can satisfy an incoming message. They also forward messages they can not satisfy to their neighbours. If a match can be made both nodes go into the satisfied state and the message is not forwarded any further by the Seller node. Later incoming possible matches are ignored. Satisfied nodes continue to pass on notes.

In the simulation updates of the nodes happen asynchronously. At each time step a node is picked randomly and the following actions take place. Parallel updates can induce system behaviour which is presumably artificial. It is also unrealistic to assume that all nodes have exactly the time globally.

- For a buyer node, the node checks whether its messages have timed out. If that is the case the node sends out new messages with a slightly increased price. The node also forwards all messages that have arrived from neighbouring nodes.
- A seller nodes checks all its incoming messages and if a match can be made it contacts the node the message originated from. If that node is still in Buyer mode both nodes go into the satisfied state. Otherwise the Sellers carries on as usual. After this all messages are forwarded to its neighbours.
- A satisfied node passes all its incoming messages.
- In the dynamic model a satisfied returns to its previous mode. It also changes the state of it peer node. Buyer nodes lower their price to a fraction of what they have just paid, $p'_b = (1 - \Delta p)p_b$. Seller nodes increase their price in the same way p'_s = (1 + “p)p_s.

Initially the nodes set into Buyer or Seller mode and the prices are drawn uniformally from two overlapping intervals.

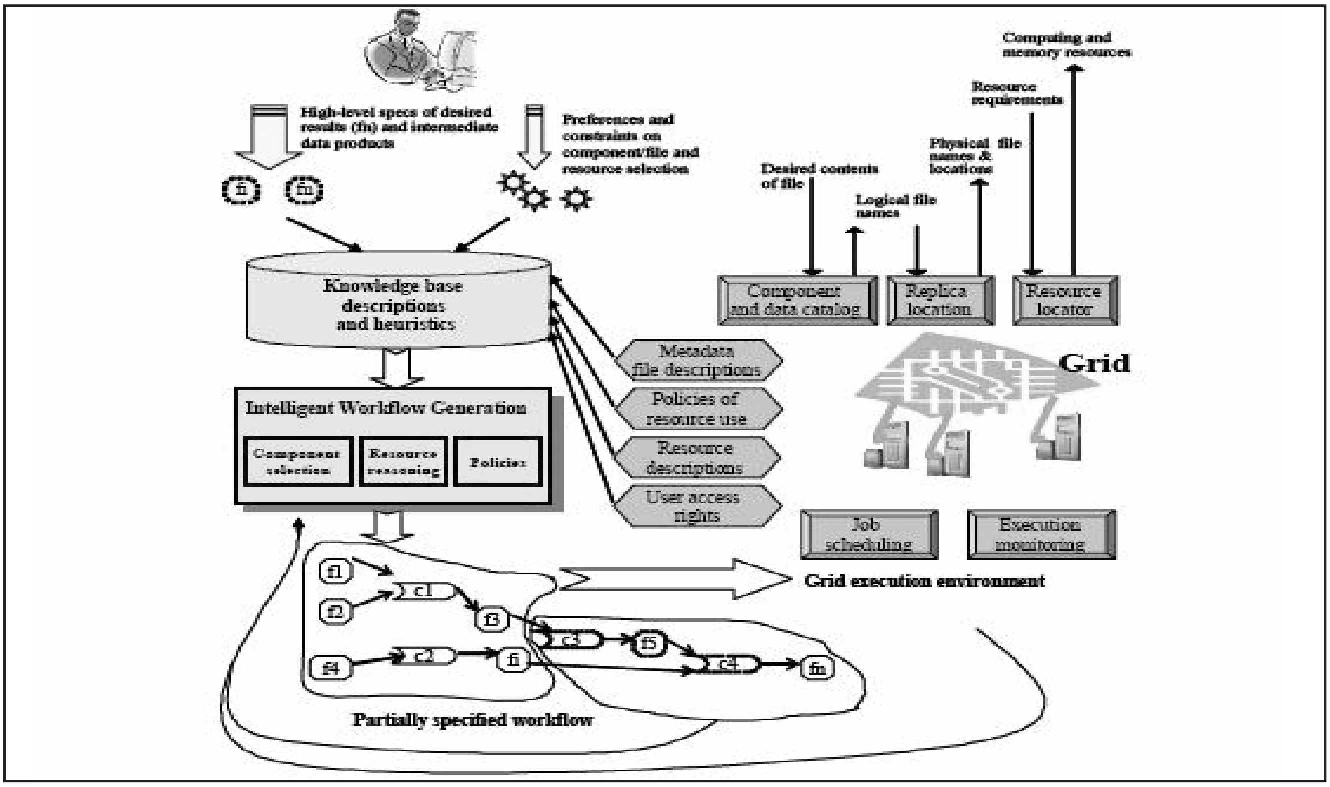


Figure 3: Application development in the Grid Environment

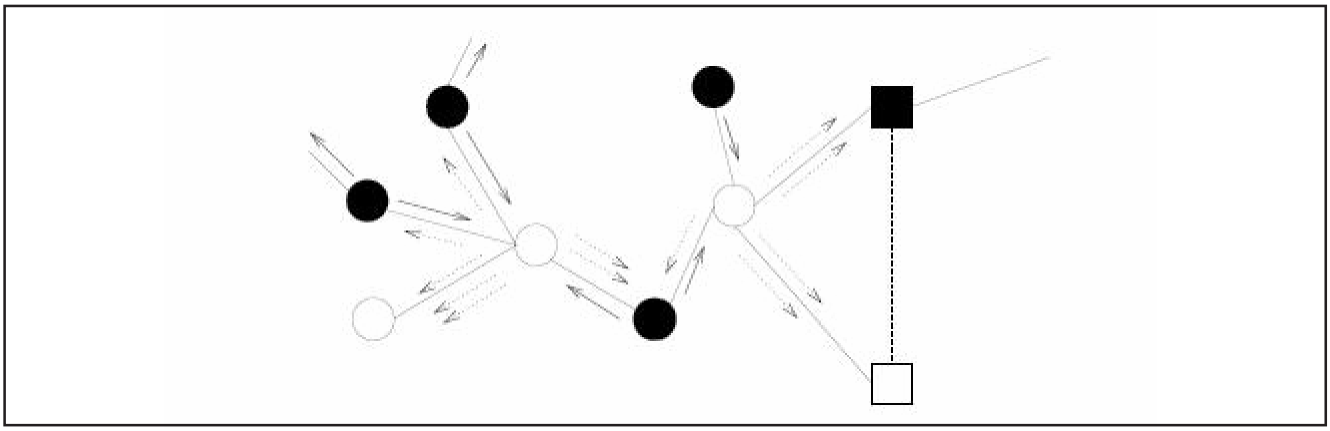


Figure 1: A partial sketch of a p2p system. Solid nodes are sellers and the others buyers. Circles indicate nodes looking for matches. Squares are satisfied nodes. The solid lines are peer relationships , the dashed line is satisfaction relationship. The solid arrows are first generation messages and the dotted ones second generations.

References—

Adelsberger. H.H.. B. Collis. J.M. Pawlowski. eds. Handbook on Information Technologies for Education and Training. Springer-Verlag, Berlin. 2002.

Andrews. T.. F. Curbera. H. Dholakia. et al. "Specification: Business Process Execution Language for Web Ser-

vices Version 1.1" IBM developer Works. 05 May 2003 http://www.ibm. com/developer works/library/ws-bpel/

Arnold. D.C.. S.S. Vadhiyar. J. Dongarra. "On the Convergence of Computational and Data Grids."" Parallel Processing Letters 11, 2001, pp. 187-202.

Barrit. C. "CISCO Systems Reusable Learning Object Strategy - Designing Information and Learning Objects Through Concept. Fact Procedure. Process, and Principle Templates " Version 4.0. WJiite Paper, CISCO Systems. Inc.. November 2001.

Bellwood. T.. L. Clement. D. Ehnebuske. et al. "UDDI Version 3.0." UDDI Spec Technical Committee Spec-

ification, 19 July 2002 http://uddi.org/pubs/uddi-v3.00 published-20020719.htm

Boag. S.. Chamberlin. D.. Fernandez. M.F, Florescu. D., Robie. J.. J. Simeon, eds. "XQuery 1.0: An XML

Query Language," W3C Working Draft 02 May 2003. http://ww.w3.org/TR/'J2003/\VD-xquery-20030502/

Box. D.. D. Ehnebuske, G. Kakivaya. et al. "Simple Object Access Protocol (SOAP) 1.1." WSC Note 08 May 2000 hnp://ww.w3.org/TR/'2000/NOTE-SOAP-2000050S

Berman. F.. Fox, G., Hey, T., eds. Grid Computing: Making the Global Infrastructure a Reality, John Wiley and Sons. Inc., New York. 2003.

Christensen. E., F. Curbera, G. Meredith. S. Weerawarana. "Web Services Description Language (WSDL) 1.1," WSC Note 15 March 2001 http:// www.w3.org/TR/2001/NOTE wsdl-20010315

Casati. F.. U. Dayal. eds. "Special Issue on Web Services." IEEE Bulletin of the Technical Committee on Data Engineering, 25 (4) 2002.

Chervenak. A.. I. Foster. C. Kesselman. C. Salisbury, S.Tuecke. "The Data Grid: Towards an Architecture for the Distributed Management and Analysis of Large Scientific Datasets " J. Network and Computer Applications 23. 2001.

De Roure, D.. M.A. Baker. K.R. Jennings, N.R. Shadbolt. The Evolution of the Grid." in [8], 2003. pp. 65-100.

Aukolekar. A.. Burstein. M.. Hobbs. J.. Lassila. O.. Martin. D.Mcllraith. S., Narayanan, S., Paolucci. M.. Payne, T., Sycara. K..Zeng, H.. DAML-S: Semantic Markup for Web Services, Proceedings of the International Semantic Web Working Symposium (ISWWS). 2001.

Amiis. J., Zhao, Y.. Voeckler, J.. Wilde. M.. Kent. S. and Foster. I., Applying Chimera Virtual Data Concepts to Cluster Finding in the Sloan Sky Survey, in Supercomputing. 2002.

Baltimore. MD.Bemers-Lee. T.. James Hendler and Ora Lassila. "The Semantic Web" Scientific American. May 2001.

Blythe. J.. Deelman. E., Gil, Y., Kesselman. C. Agarwal, A.,Mehta. G.. Vahi. K.. The Role of Plamiing in Grid

Computing, in Proc. IntL Confi on AI Planning and Scheduling. (ICAPS) 2003

Boutilier. C. Dean, T., Hanks. S.. Decision-Theoretic Planning: Structural Assumptions and Computational Leverage. Joiwnal of Artificial Intelligence Research. 11. 1-94. 1999 CIM. 2002.

Singh. G.. A Metadata Catalog Service for Data Intensive Applications. 2002, GriPhyN technical report 2002-11.

Czajkowski. K.. Fitzgerald. S.. Foster. I. and Kesselman. C. Grid Information Services for Distributed Resource Sharins. in 10th IEEE International Symposium on High Performance Distributed Computing. 2001: IEEE Press.

Deelman. E.. J. Blythe. Y. Gil. C. Kesselman. G. Mehta. K. Vahi.K. Blackburn, A. Lazzarini. A. Arbree. R. Cavanaugh and S.Koranda. "Mapping Abstract Complex Workflows onto Grid Environments". Journal of Grid Computing, vol. 1. 2003

Deelman. E., et al.. From Metadata to Execution on the Grid: The Pegasus Pulsar Search. 2003. GriPhyN 2003-15.

Foster. I. and C. Kesselman. eds. The Grid: Blueprint for a New Computing Infrastructure. 1999. Morgan Kaufmann.

Foster. I.. C. Kesselman. and S. Tuecke. The Anatomy of the Grid: Enabling Scalable Virtual Organizations. International Journal of High Performance Computing Applications, 2001.15(3): p. 200-222.

U. Harder, P. Harrison, M. Paczuski, and T. Shah, "A dynamical model of a GRID market," tech. rep., Department of Computing, Imperial College London, October 2004.

J. Darlington, C. Richardson, and J. Cohen, "Global Open Grid.".

T. Eymann, B. Padovan, and D. Schoder, ''The catallaxy as a new paradigm for the design of information systems," in Proceedings of the 16th IFIP World Computer Congress, Conference on Intelligent Information Processing, 2000.

O. Ardaiz, P. Artigas, T. Eymann, F. Freitag, L. Navarro, and M. Reinicke, "The catallaxy approach for decentralized economic-based allocation in grid resource and service markets," Applied Intelligence, vol. 25, no. 2, pp. 131-145, 2006.