Chapter 1

THE GRID IN A NUTSHELL

Ian Foster[1] and Carl Kesselman[2]
[1] *Mathematics and Computer Science Division, Argonne National Laboratory*
[2] *Information Sciences Institute, University of Southern California*

Abstract The emergence and widespread adoption of Grid computing has been fueled by continued growth in both our understanding of application requirements and the sophistication of the technologies used to meet these requirements. We provide an introduction to Grid applications and technologies and discuss the important role that resource management will play in future developments.

1. INTRODUCTION

The term "the Grid" was coined in the mid-1990s to denote a (then) proposed distributed computing infrastructure for advanced science and engineering. Much progress has since been made on the construction of such an infrastructure and on its extension and application to commercial computing problems. And while the term "Grid" has also been on occasion applied to everything from advanced networking and computing clusters to artificial intelligence, there has also emerged a good understanding of the problems that Grid technologies address, as well as a first set of applications for which they are suited.

Grid concepts and technologies were initially developed to enable resource sharing within scientific collaborations, first within early Gigabit testbeds [CS92, Cat92, Mes99] and then on increasingly larger scales [BCF$^+$98, GWWL94, BJB$^+$00, SWDC97, JGN99, EH99]. At the root of these collaborations was the need for participations to share not only datasets but also software, computational resources, and even specialized instruments such as telescopes and microscopes. Consequently, a wide range of application types emerged that included distributed computing for computationally demanding data analysis (pooling of compute power and storage), the federation of diverse distributed datasets, collaborative visualization of large scientific datasets, and coupling of scientific instruments such as electron microscopes, high-energy x-ray sources, and experimental data acquisitions systems with remote users, computers, and archives (increasing functionality as well as availability) [BAJZ98, Joh99, HKL^{+}00].

We have argued previously that underlying these different usage modalities there exists a common set of requirements for *coordinated resource sharing and problem solving in dynamic, multi-institutional collaborations* [FKT01]. The term *virtual organization* is often applied to these collaborations because of their distributed and often ephemeral nature. This same requirement for resource sharing across cross organizational collaborations arises within commercial environments, including enterprise application integration, on-demand service provisioning, data center federation, and business-to-business partner collaboration over the Internet. Just as the World Wide Web began as a technology for scientific collaboration and was adopted for e-business, we see a similar trajectory for Grid technologies.

The success of the Grid to date owes much to the relatively early emergence of clean architectural principles, de facto standard software, aggressive early adopters with challenging application problems, and a vibrant international community of developers and users. This combination of factors led to a solid base of experience that has more recently driven the definition of the service-oriented Open Grid Services Architecture that today forms the basis for both open source and commercial Grid products. In the sections that follow, we expand on these various aspects of the Grid story and, in so doing, provide context for material to be presented in later chapters.

2. VIRTUAL ORGANIZATIONS

We have defined Grids as being concerned with enabling coordinated resource sharing and problem solving in dynamic, multi-institutional virtual organizations. The sharing that we are concerned with is not primarily file exchange, as supported by the Web or peer-to-peer systems, but rather direct access to computers, software, data, services, and other resources, as required by a range of collaborative problem-solving and resource-brokering strategies emerging in industry, science, and engineering. This sharing is, necessarily, highly controlled, with resource providers and consumers defining clearly and carefully just what is shared, who is allowed to share, and the conditions under which sharing occurs. A set of individuals and/or institutions defined by such sharing rules form what we call a *virtual organization* (VO) [FKT01].

VOs can vary greatly in terms of scope, size, duration, structure, distribution, and capabilities being shared, community being serviced, and sociology. Examples of VOs might include

- the application service providers, storage service providers, cycle providers, and consultants engaged by a car manufacturer to perform scenario evaluation during planning for a new factory;
- members of an industrial consortium bidding on a new aircraft;
- a crisis management team and the databases and simulation systems that they use to plan a response to an emergency situation; and
- members of a large international high-energy physics collaboration.

These examples only hint at the diversity of applications enabled by cross-organizational sharing. In spite of these differences, however, study of underlying technology requirements leads us to identify a broad set of common concerns and requirements. We see a need to establish and maintain highly flexible sharing relationships capable of expressing collaborative structures such as client-server and peer-to-peer, along with more complex relationships such as brokered or sharing via intermediaries. We also see requirements for complex and high levels of control over how shared resources are used, including fine-grained access control, delegation, and application of local and global policies. We need basic mechanisms for discovering, provisioning, and managing of varied resources, ranging from programs, files, and data to computers, sensors, and networks, so as to enable time critical or performance-critical collaborations, and for diverse usage modes, ranging from single user to multi-user and from performance sensitive to cost sensitive and hence embracing issues of quality of service, scheduling, co-allocation, and accounting.

3. GRID APPLICATIONS

Various Grid application scenarios have been explored within both science and industry. We present here a representative sample of thirteen such applications that collectively introduce the broad spectrum of usage scenarios that are driving Grid adoption and development. These applications include compute-intensive, data-intensive, sensor-intensive, knowledge-intensive, and collaboration-intensive scenarios and address problems ranging from multi-player video gaming, fault diagnosis in jet engines, and earthquake engineering to bioinformatics, biomedical imaging, and astrophysics. Further details on each application can be found in a set of case studies collected in a recent book [FK04].

Distributed Aircraft Engine Diagnostics. The U.K. Distributed Aircraft Maintenance Environment (DAME) project is applying Grid technologies to the challenging and broadly important problem of computer-based fault diagnosis, an inherently distributed problem in many situations because of the range of data sources and stakeholders involved [AJF+04]. In particular, DAME is working to diagnose faults in Rolls Royce aircraft engines, based on sensor data recorded at the rate of one gigabyte per engine per transatlantic flight.

NEESgrid Earthquake Engineering Collaboratory. The U.S. Network for Earthquake Engineering Simulation (NEES) is an ambitious project to enable remote access to, and collaborative use of, the specialized equipment used to study the behavior of structures, such as bridge columns when subjected to the forces of an earthquake. NEESgrid uses Grid technology to provide remote access to experimental access (i.e., teleobservation and telecontrol); to couple physical experiments with numerical simulation; and to archive, discover, and analyze simulation, experimental, and observational data [KPF04, PKF+01].

World Wide Telescope. Advances digital astronomy enable the systematic survey of the heavens and the collection of vast amounts of data from telescopes over the world. New scientific discoveries can be made not only by analyzing data from an individual instruments but also by comparing and correlating data from different sky surveys [SG04, SG01]. The emerging "World Wide Telescope," or virtual observatory, uses Grid technology to federate data from hundreds of individual instruments, allowing a new generation of astronomers to perform analysis of unprecedented scope and scale. While the most immediate challenges relate to data formats and data management, the need to manage the computation resources consumed by such data analysis tasks is a looming issue.

Biomedical Informatics Research Network. (BIRN). The goal of this U.S. project is to federate biomedical imaging data for the purpose of research and, ultimately, improved clinical case [EP04]. To this end, BIRN is deploying compute-storage clusters at research and clinical sites around the United States and deploying Grid middleware to enable the integration of image data from multiple locations.

In silico Experiments in Bioinformatics. The U.K. myGrid project is applying Grid technologies to the semantically rich problems of dynamic resource discovery, workflow specification, and distributed query processing, as well as provenance management, change notification, and personalization [GPS04].

story, a group of U.S. physicists and computer scientists completed a challenging data generation and analysis task for a high energy physics experiment, harnessing computing and storage resources at six sites to generate 1.5 million simulated events during a two-month run [GCC+04].

Virtual Screening on Desktop Computers. In this drug discovery application, an intra-Grid composed of desktop PCs was used for virtual screening of drug candidates [Chi04, CFG02, CCEB03]. An existing molecular docking application was integrated into a commercial Grid environment to achieve a significant increase in processing power over what drug companies would typically have dedicated to compound screening.

Enterprise Resource Management. GlobeXplorer, a provider of online satellite imagery, is an example of an enterprise that uses advanced resource management techniques to improve the efficiency and flexibility of intra-Enterprise resource usage [Gen04].

Infrastructure for Multiplayer Games. Butterfly.net, a service provider for the multiplayer videogaming industry, is using Grid technologies to deliver scalable hosting services to game developers [LW04, LWW03]. A potentially huge number of game participants and a need for interactive response lead to challenging performance requirements

Service Virtualization. As we discuss in greater detail below, virtualization is playing an increasingly important role in enterprise IT infrastructures [XHLL04]. In a recent success story, the deployment of a resource and service virtualization solution at a global investment bank resulted in significant performance improvements.

Access Grid Collaboration System. High-end collaboration and conferencing environments represent an application domain for Grid technologies that is rapidly growing in importance [CDO$^+$00, Ste04]. The delivery of rich group collaboration capabilities places heavy demands on Grid technologies.

Collaborative Astrophysics. An enthusiastic community of computational astrophysicists has been working for some years to apply Grid technologies to a range of problems in high-end collaborative science [ABH$^+$99, ADF$^+$01, AS04].

4. GRID TECHNOLOGY

The development of Grids has been spurred and enabled by the staged development of increasingly sophisticated and broadly used technologies. As illustrated in Figure 1.1, early experiments with "metacomputing" [CS92, Cat92, EH99, GWWL94, Mes99] worked primarily with custom tools or specialized middleware [GRBK98, FGN$^+$96, FGT96, Sun90] that emphasized message-oriented communication between computing nodes.

The transition from metacomputing to Grid computing occurred in the mid-1990s with the introduction of middleware designed to function as wide-area infrastructure to support diverse online processing and data-intensive applications. Systems such as the Storage Resource Broker [BMRW98], Globus Toolkit® [FK97], Condor [FTF$^+$02, LLM88], and Legion [GW97, GFKH99] were developed primarily for scientific applications and demonstrated at various levels of scale on a range of applications. Other developers attempted to leverage the middleware structure being developed for the World Wide Web by using HTTP servers or Web browsers as Grid computing platforms [BKKW96, BSST96, GBE+98]. These systems did not gain significant use, however, partly because the middleware requirements for distributed information systems such as the Web are different from those for Grid applications.

Globus Toolkit. By 1998, the open source Globus Toolkit (GT2) [FK97] had emerged as a de facto standard software infrastructure for Grid computing. GT2 defined and implemented protocols, APIs, and services used in hundreds of Grid deployments worldwide. By providing solutions to common problems such as authentication, resource discovery, resource access, and data movement, GT2 accelerated the construction of real Grid applications. And by defining and implementing "standard" protocols and services, GT pioneered the creation of interoperable Grid systems and enabled significant progress on Grid programming tools. This standardization played a significant role in spurring the subsequent explosion of interest, tools, applications, and deployments, as did early success stories such as a record-setting 100,000-entity distributed interactive simulation scenario in 1998 [BDG+98, BCF+98] and the solution in June 2000 of nug30 [ABGL02], a previously open problem in optimization theory.

The GT2 protocol suite leveraged heavily existing Internet standards for security, resource discovery, and security. In addition, some elements of the GT2 protocol suite were codified in formal technical specifications and reviewed within standards bodies: notably, the GridFTP data transfer protocol (for which multiple implementations exist) [ABB+02] and elements of the Grid Security Infrastructure [FKTT98, TEF+02]. In general, however, GT2 "standards" were not formal, well documented, or subject to public review. They were not in themselves a sufficient basis for the creation of a mature Grid ecosystem. Similar comments apply to other important Grid technologies that emerged during this period, such as the Condor high throughput computing system.

This period also saw significant development of more user-oriented tools, most building on the Globus Toolkit. For example, MPICH-G [KTF03], a Grid-aware version of the public-domain Message Passing Interface (MPI) [GLS94], provided a standards-based message-passing environment. Tools such as NetSolve and Ninf [TNS+02] sought to deliver Grid-enabled software to a nonexpert user community, while portal toolkits [AC02, Nov02, PLL+03, RAD+02, TMB+01] allowed Grid-enabled applications to be delivered to end users via a standard Web browser. Workflow systems such as DAGman [DAG] and Chimera [FVWZ02], and scripting languages such as pyGlobus [Jac02], focus on coordination of components written in traditional programming languages.

Open Grid Services Architecture. As interest in Grids continued to grow, and in particular as industrial interest emerged, the importance of true stan-

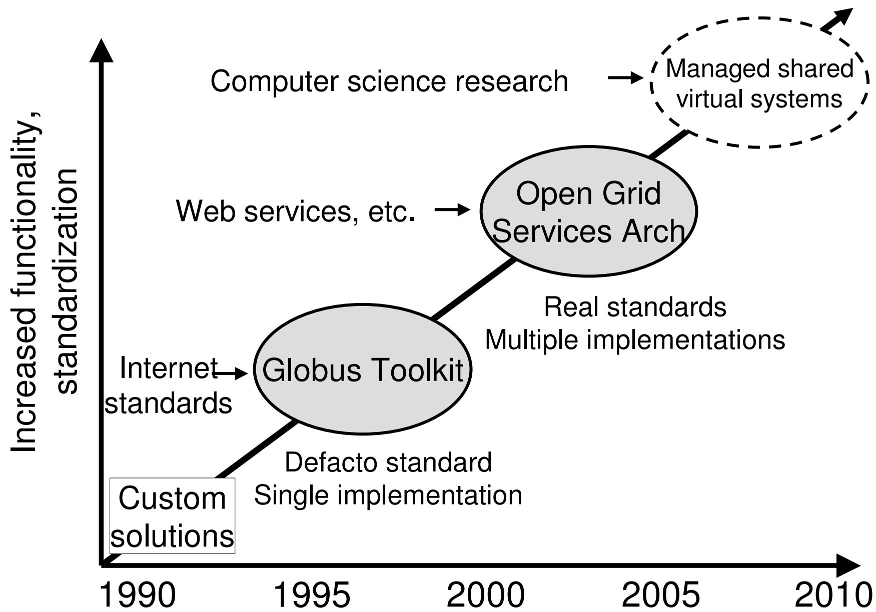

Figure 1.1. The evolution of Grid technology.

dards increased. The Global Grid Forum, established in 1998 as an international community and standards organization, appeared to be the natural place for such standards to be developed, and indeed multiple standardization activities are now under way. In particular, 2002 saw the emergence of the Open Grid Services Architecture (OGSA) [FKNT02], a true community standard with multiple implementations–including the OGSA-based Globus Toolkit 3.0 [WSF+03], released in 2003. Building on and significantly extending GT2 concepts and technologies, OGSA firmly aligns Grid computing with broad industry initiatives in service-oriented architecture and Web services.

In addition to defining a core set of standard interfaces and behaviors that address many of the technical challenges introduced above, OGSA provides a framework within which can be defined a wide range of interoperable, portable services. OGSA thus provides a foundation on which can be constructed a rich Grid technology ecosystem comprising multiple technology providers. Thus, we see, for example, major efforts under way within the U.K. eScience programme to develop data access and integration services [ACK+04, PAD+02].

Concurrent with these developments we see a growing recognition that large-scale development and deployment of Grid technologies is critical to future success in a wide range of disciplines in science, engineering, and the humanities [ADF+03], and increasingly large investments within industry in related areas. While research and commercial uses can have different concerns, they also have much in common, and there are promising signs that the required technologies can be developed in a strong academic-industrial partnership. The current open source code base and emerging open standards provide a solid foundation for the new open infrastructure that will result from this work.

Managed, Shared Virtual Systems. The definition of the initial OGSA technical specifications is an important step forward, but much more remains to be done before the full Grid vision is realized. Building on OGSA's service-oriented infrastructure, we will see an expanding set of interoperable services and systems that address scaling to both larger numbers of entities and smaller device footprints, increasing degrees of virtualization, richer forms of sharing, and increased qualities of service via a variety of forms of active management. This work will draw increasingly heavily on the results of advanced computer science research in such areas as peer-to-peer [CMPT04, FI03], knowledge-based [BLHL01, GDRSF04], and autonomic [Hor01] systems.

5. SERVICE ORIENTATION, INTEGRATION, AND VIRTUALIZATION

Three related concepts are key to an understanding of the Grid and its contemporary technologies and applications: service orientation, integration, and virtualization.

A service is an entity that provides some capability to its clients by exchanging messages. A service is defined by identifying sequences of specific message exchanges that cause the service to perform some *operation*. By thus defining these operations only in terms of message exchange, we achieve great flexibility in how services are implemented and where they are located. A service-oriented architecture is one in which all entities are services, and thus any operation that is visible to the architecture is the result of message exchange.

By encapsulating service operations behind a common message-oriented service interface, service orientation isolates users from details of service implantation and location. For example, a storage service might present the user with an interface that defines, among other things, a store file operation. A user should be able to invoke that operation on a particular instance of that storage service without regard to how that instance implements the storage service interface. Behind the scenes, different implementations may store the file on the user's local computer, in a distributed file system, on a remote archival storage system, or in free space within a department desktop pool-or even to choose from among such alternatives depending on context, load, amount paid, or other factors. Regardless of implementation approach, the user is aware only that the requested operation is executed-albeit with varying cost and other qualities of service, factors that may be subject to negotiation between the client and service. In other contexts, a *distribution framework* can be used to disseminate work across service instances, with the number of instances of different services deployed varying according to demand [AFF+01, GKTA02, XHLL04].

While a service implementation may directly perform a requested operation, services may be *virtual*, providing an interface to underlying, distributed services, which in turn may be virtualized as well. Service virtualization also introduces the challenge, and opportunity, of *service integration*. Once applications are encapsulated as services, application developers can treat different services as building blocks that can be assembled and reassembled to adapt to changing business needs. Different services can have different performance characteristics; and, in a virtualized environment, even different instances of the *same* service can have different characteristics. Thus new distributed system integration techniques are needed to achieve end-to-end guarantees for various qualities of service.

6. THE FUNDAMENTAL ROLE OF RESOURCE MANAGEMENT

Fundamental to both service virtualization and integration is the ability to discover, allocate, negotiate, monitor, and manage the use of network-accessible capabilities in order to achieve various end-to-end or global qualities of service. Within a service-oriented architecture, these capabilities may include both traditional *resources* (computational services offered by a computer, network bandwidth, or space on a storage system) and virtualized *services* (e.g., database, data transfer, simulation), which may differ in the function they provide to users but are consistent in the manner in which they deliver that function across the network. Nevertheless, for historical reasons, and without loss of generality, the term *resource management* is commonly used to describe all aspects of the process of locating various types of capability, arranging for their use, utilizing them, and monitoring their state.

In traditional computing systems, resource management is a well-studied problem. Resource managers such as batch schedulers, workflow engines, and operating systems exist for many computing environments. These resource management systems are designed and operate under the assumption that they have complete control of a resource and thus can implement the mechanisms and policies needed for effective use of that resource in isolation. Unfortunately, this assumption does not apply to the Grid. We must develop methods for managing Grid resources across separately administered domains, with the resource heterogeneity, lose of absolute control, and inevitable differences in policy that result. The underlying Grid resource set is typically heterogeneous even within the same class and type of resource. For example, no two compute clusters have the same software configuration, local resource management software, administrative requirements, and so forth. For this reason, much of the early work in Grid resource management focused on overcoming these basic issues of heterogeneity, for example through the definition of standard resource management protocols [CFK+98, CFK+02] and standard mechanisms for expressing resource and task requirements [RLS98].

More important than such issues of plumbing, however, is the fact that different organizations operate their resources under different policies; the goals of the resource user and the resource provider may be inconsistent, or even in conflict. The situation is further complicated by the fact that Grid applications often require the concurrent allocation of multiple resources, necessitating a structure in which resource use can be coordinated across administrative domains [CFK99, FFR+02]. Much current research in Grid resource management is focused on understanding and managing these diverse policies from the perspective of both the resource provider and the consumer [BWF+96, AC02, KNP01, CKKG99] with the goal of synthesizing end-to-end resource management in spite of the fact that the resources are independently owned and administered.

The emergence of service-oriented architecture, the increased interest in supporting a broad range of commercial applications, and the natural evolution of functionality are collectively driving significant advances in resource management capabilities. While today's Grid environment is primarily oriented toward best-effort service, we expect the situation to become substantially different in the next several years, with end-to-end resource provisioning and virtualized service behavior that is indistinguishable from nonvirtualized services becoming the rule rather than the exception.

We possess a good understanding of the basic mechanisms required for a provisioned Grid. Significant challenges remain, however, in understanding how these mechanisms can be effectively combined to create seamless virtualized views of underlying resources and services. Some of these challenges lie strictly within the domain of resource management, for example, robust distributed algorithms for negotiating simultaneous service level agreements across a set of resources. Other issues, such as expression of resource policy for purposes of discovery and enhanced security models that support flexible delegation of resource management to intermediate brokers are closely tied to advances in other aspects of Grid infrastructure. Hence, the key to progress in the coming years is to create an extensible and open infrastructure that can incorporate these advances as they become available.

7. SUMMARY

We have provided both a historical and a technological introduction to Grid computing. As we have discussed, the dynamic and cross-organizational nature of the Grid is at the root of both the opportunities and challenges that are inherent in Grid infrastructure and applications. These same issues also have a profound effect on resource management. While many of the resource management techniques developed over the past 40 years have applicability to the Grid, these techniques must be reassessed in the context of an environment in which both absolute knowledge of system state and absolute control over resource policy and use are not possible.

The development of reasonably mature Grid technologies has allowed many academic and industrial application groups to achieve significant successes. Nevertheless, much further development is required before we can achieve the ultimate goal of virtualized services that can be integrated in flexible ways to deliver strong application-level performance guarantees. Advances in resource management will be key to many of these developments.

Acknowledgments

This work was supported in part by the Mathematical, Information, and Computational Sciences Division subprogram of the Office of Advanced Scientific Computing Research, U.S. Department of Energy, Office of Science, under Contract W-31-109-Eng-38, and by the NSF Middleware Initiative.

References

[ABB$^+$02] W. Allcock, J. Bester, J. Bresnahan, A. Chervenak, L. Liming, S. Meder, and S. Tuecke. GridFTP protocol specification. Technical report, Global Grid Forum GridFTP Working Group, September 2002.

[ABGL02] K. Anstreicher, N. Brixius, J.-P. Goux, and J. T. Linderoth. Solving large quadratic assignment problems on computational Grids. *Mathematical Programming*, 91(3):563–588, 2002.

[ABH$^+$99] G. Allen, W. Benger, C. Hege, J. Masso, A. Merzky, T. Radke, E. Seidel, and J. Shalf. Solving Einstein's equations on supercomputers. *IEEE Computer Applications*, 32(12):52–58, 1999.

[AC02] G. Aloisio and M. Cafaro. Web-based access to the Grid using the Grid Resource Broker Portal. *Concurrency and Computation: Practice and Experience, Special Issue on Grid Computing Environments*, 14:1145–1160, 2002.

[ACK$^+$04] Malcolm Atkinson, Ann Chervenak, Peter Kunszt, Inderpal Narang, Norman Paton, Dave Pearson, Arie Shoshani, and Paul Watson. Data access, integration, and management. In Ian Foster and Carl Kesselman, editors, *The Grid: Blueprint for a New Computing Infrastructure (Second Edition)*. Morgan Kaufmann, 2004.

[ADF$^+$01] G. Allen, T. Dramlitsch, I. Foster, N. Karonis, M. Ripeanu, E. Seidel, and B. Toonen. Supporting efficient execution in heterogeneous distributed computing environments with Cactus and Globus. In *Proceedings of SuperComputing (SC'01)*, 2001.

[ADF$^+$03] Daniel E. Atkins, Kelvin K. Droegemeier, Stuart I. Feldman, Hector Garcia-Molina, Michael L. Klein, David G. Messerschmitt, Paul Messina, Jeremiah P. Ostriker, and Margaret H. Wright. Revolutionizing science and engineering through cyberinfrastructure: Report of the National Science Foundation blue ribbon advisory panel on cyberinfrastructure. Technical report, NSF, 2003. Available from `http://www.communitytechnology.org/nsf_ci_report/report.pdf`.

[AFF+01] Karen Appleby, Sameh Fakhouri, Liana Fong, German Goldszmidt, Michael Kalantar, Srirama Krishnakumar, Donald Pazel, John Pershing, and Benny Rochwerger. Oceano: SLA based management of a computing utility. In *Proceedings of the Seventh IFIP/IEEE International Symposium on Integrated Network Management*, 2001.

[AJF+04] Jim Austin, Tom Jackson, Martyn Fletcher, Mark Jessop, Peter Cowley, and Peter Lobner. Predictive maintenance: Distributed aircraft engine diagnostics. In Ian Foster and Carl Kesselman, editors, *The Grid: Blueprint for a New Computing Infrastructure (Second Edition)*. Morgan Kaufmann, 2004.

[AS04] Gabrielle Allen and Ed Seidel. Collaborative science: Astrophysics requirements and experiences. In Ian Foster and Carl Kesselman, editors, *The Grid: Blueprint for a New Computing Infastructure (Second Edition)*. Morgan Kaufmann, 2004.

[BAJZ98] J. Bonkalski, R. Anderson, S. Jones, and N. Zaluzec. Bringing telepresence microscopy and science collaboratories into the class room. *TeleConference Magazine*, 17(9), 1998.

[BCF+98] Sharon Brunett, Karl Czajkowski, Steven Fitzgerald, Ian Foster, Andrew Johnson, Carl Kesselman, Jason Leigh, and Steven Tuecke. Application experiences with the Globus toolkit. In *Proceedings of the Seventh IEEE International Symposium on High-Performance Distributed Computing (HPDC-7)*, pages 81–89, 1998.

[BDG+98] S. Brunett, D. Davis, T. Gottschalk, P. Messina, and C. Kesselman. Implementing distributed synthetic forces simulations in metacomputing environments. In *Proceedings of the Heterogeneous Computing Workshop*, pages 29–42, 1998.

[BJB+00] Judy Beiriger, Wilbur Johnson, Hugh Bivens, Steven Humphreys, and Ronald Rhea. Constructing the ASCI Grid. In *Proceedings of the Ninth IEEE International Symposium on High-Performance Distributed Computing (HPDC-9)*, 2000.

[BKKW96] A. Baratloo, M. Karaul, Z. Kedem, and P. Wyckoff. Charlotte: Metacomputing on the Web. In *Proceedings of the Ninth International Conference on Parallel and Distributed Computing Systems*, 1996.

[BLHL01] Tim Berners-Lee, James Hendler, and Ora Lassila. The semantic Web. *Scientific American*, 284(5):34–43, 2001.

[BMRW98] C. Baru, R. Moore, A. Rajasekar, and M. Wan. The SDSC Storage Resource Broker. In *Proceedings of the Eighth Annual IBM Centers for Advanced Studies Conference (CASCON '98)*, 1998.

[BSST96] T. Brecht, H. Sandhu, M. Shan, and J. Talbot. ParaWeb: Towards world-wide supercomputing. In *Proceedings of the Seventh ACM SIGOPS European Workshop on System Support for Worldwide Applications*, 1996.

[BWF+96] F. Berman, R. Wolski, S. Figueira, J. Schopf, and G. Shao. Application-level scheduling on distributed heterogeneous networks. In *Proceedings of SuperComputing (SC'96)*, 1996.

[Cat92] Charlie Catlett. In search of gigabit applications. *IEEE Communications Magazine*, 30(4):42–51, April 1992.

[CCEB03] Andrew Chien, Bradley Calder, Stephen Elbert, and Karan Bhatia. Entropia: Architecture and performance of an Enterprise desktop Grid system. *Journal of Parallel and Distributed Computing*, *to appear,* 2003.

[CDO+00] L. Childers, T. Disz, R. Olson, M. E. Papka, R. Stevens, and T. Udeshi. Access Grid: Immersive group-to-group collaborative visualization. In *Proceedings of the Fourth International Immersive Projection Technology Workshop*, 2000.

[CFG02] A. Chien, I. Foster, and D. Goddette. Grid technologies empowering drug discovery. *Drug Discovery Today*, 7(20):176–180, 2002.

[CFK+98] K. Czajkowski, I. Foster, N. Karonis, C. Kesselman, S. Martin, W. Smith, and S. Tuecke. A resource management architecture for metacomputing systems. In D. Feitelson and L. Rudolph, editors, *Job Scheduling Strategies for Parallel Processing (Proceedings of the Fourth International JSSPP Workshop; LNCS #1459)*, pages 62–82. Springer-Verlag, 1998.

[CFK99] K. Czajkowski, I. Foster, and C. Kesselman. Co-allocation services for computational Grids. In *Proceedings of the Eighth*

IEEE International Symposium on High Performance Distributed Computing (HPDC-8), August 1999.

[CFK+02] K. Czajkowski, I. Foster, C. Kesselman., V. Sander, and S. Tuecke. SNAP: A protocol for negotiating service level agreements and coordinating resource management in distributed systems. In D. Feitelson and L. Rudolph, editors, *Job Scheduling Strategies for Parallel Processing (Proceedings of the Eighth International JSSPP Workshop; LNCS #2537)*, pages 153–183. Springer-Verlag, 2002.

[Chi04] A. Chien. Massively distributed computing: Virtual screening on desktop computers. In Ian Foster and Carl Kesselman, editors, *The Grid: Blueprint for a New Computing Infrastructure (Second Edition)*. Morgan Kaufmann, 2004.

[CKKG99] S. J. Chapin, D. Katramatos, J. F. Karpovich, and A. S. Grimshaw. Resource management in Legion. *Future Generation Computing Systems*, 15:583–594, October 1999.

[CMPT04] Jon Crowcroft, Tim Moreton, Ian Pratt, and Andrew Twigg. Peer-to-peer technologies. In Ian Foster and Carl Kesselman, editors, *The Grid: Blueprint for a New Computing Infrastructure (Second Edition)*. Morgan Kaufmann, 2004.

[CS92] C. Catlett and L. Smarr. Metacomputing. *Communications of the ACM*, 35(6):44–52, 1992.

[DAG] The Condor Directed-Acyclic-Graph Manager (DAGMan). `http://www.cs.wisc.edu/condor/dagman/`.

[EH99] Thomas Eickermann and Ferdinand Hommes. Metacomputing in a gigabit testbed west. In *Proceedings of the Workshop on Wide Area Networks and High Performance Computing*, Lecture Notes in Control and Information Sciences, pages 119–129. Springer-Verlag, 1999.

[EP04] M. Ellisman and S. Peltier. Medical data federation: The biomedical informatics research network. In Ian Foster and Carl Kesselman, editors, *The Grid: Blueprint for a New Computing Infrastructure (Second Edition)*. Morgan Kaufmann, 2004.

[FFR+02] I. Foster, M. Fidler, A. Roy, V. Sander, and L. Winkler. End-to-end quality of service for high-end applications. *Computer Communications, Special Issue on Network Support for Grid Computing*, 2002.

[FGN+96] I. Foster, J. Geisler, W. Nickless, W. Smith, and S. Tuecke. Software infrastructure for the I-WAY high-performance distributed computing experiment. In *Proceedings of the Fifth IEEE International Symposium on High-Performance Distributed Computing (HPDC-5)*, pages 562–571, 1996.

[FGT96] I. Foster, J. Geisler, and S. Tuecke. MPI on the I-WAY: A wide-area, multimethod implementation of the Message Passing Interface. In *Proceedings of the 1996 MPI Developers Conference*, pages 10–17, 1996.

[FI03] I. Foster and A. Iamnitchi. On death, taxes, and the convergence of peer-to-peer and Grid computing. In *Proceedings of the Second International Workshop on Peer-to-Peer Systems (IPTPS)*, 2003.

[FK97] I. Foster and C. Kesselman. Globus: A metacomputing infrastructure toolkit. *International Journal of Supercomputer Applications*, 11(2):115–129, 1997.

[FK04] Ian Foster and Carl Kesselman, editors. *The Grid: Blueprint for a New Computing Infrastructure (Second Edition)*. Morgan Kaufmann, 2004.

[FKNT02] I Foster, C. Kesselman, J. M. Nick, and S. Tuecke. Grid services for distributed systems integration. *IEEE Computer*, 35(6):37–46, 2002.

[FKT01] I. Foster, C. Kesselman, and S. Tuecke. The Anatomy of the Grid: Enabling Scalable Virtual Organizations. *International Journal of High Performance Computing Applications*, 15(3):200–222, 2001. Also available from `http://www.globus.org/research/papers/anatomy.pdf`.

[FKTT98] I. Foster, C. Kesselman, G. Tsudik, and S. Tuecke. A security architecture for computational Grids. In *Proceedings of the Fifth ACM Conference on Computer and Communications Security Conference*, 1998.

[FTF+02] J. Frey, T. Tannenbaum, I. Foster, M. Livny, and S. Tuecke. Condor-G: A computation management agent for multi-institutional Grids. *Cluster Computing*, 5(3):237–246, 2002.

[FVWZ02] Ian Foster, Jens Vockler, Michael Wilde, and Yong Zhao. Chimera: A virtual data system for representing, querying, and automating data derivation. In *Proceedings of the Fourteenth*

International Conference on Scientific and Statistical Database Management (SSDBM'02), 2002.

[GBE+98] Amin (Grid2001)Vahdat, Eshwar Belani, Paul Eastham, Chad Yoshikawa, Thomas Anderson, David Culler, and Michael Dahlin. WebOS: Operating system services for wide area applications. In *Proceedings of the Seventh IEEE International Symposium on High-Performance Distributed Computing (HPDC-7)*, 1998.

[GCC+04] Greg Graham, Richard Cavanaugh, Peter Couvares, Alan DeSmet, and Miron Livny. Distributed data analysis: Federated computing for high energy physics. In Ian Foster and Carl Kesselman, editors, *The Grid: Blueprint for a New Computing Infrastructure (Second Edition)*. Morgan Kaufmann, 2004.

[GDRSF04] Carole A. Goble, David De Roure, Nigel R. Shadbolt, and Alvaro Fernandes. Enhancing services and applications with knowledge and semantics. In Ian Foster and Carl Kesselman, editors, *The Grid: Blueprint for a New Computing Infrastructure (Second Edition)*. Morgan Kaufmann, 2004.

[Gen04] Wolfgang Gentzsch. Enterprise resource management: Applications in research and industry. In Ian Foster and Carl Kesselman, editors, *The Grid: Blueprint for a New Computing Infrastructure (Second Edition)*. Morgan Kaufmann, 2004.

[GFKH99] A. S. Grimshaw, A. J. Ferrari, F. Knabe, and M. A. Humphrey. Wide-area computing: Resource sharing on a large scale. *IEEE Computer*, 32(5), May 1999.

[GKTA02] Sven Graupner, Vadim Kotov, Holger Trinks, and Artur Andrzejak. Control architecture for service Grids in a federation of utility data centers. Technical Report HPL-2002-235, HP Labs, 2002.

[GLS94] W. Gropp, E. Lusk, and A. Skjellum. *Using MPI: Portable Parallel Programming with the Message Passing Interface*. MIT Press, 1994.

[GPS04] Carole Goble, Steve Pettifer, and Robert Stevens. Knowledge integration: In silico experiments in bioinformatics. In Ian Foster and Carl Kesselman, editors, *The Grid: Blueprint for a New Computing Infrastructure (Second Edition)*. Morgan Kaufmann, 2004.

[GRBK98] E. Gabriel, M. Resch, T. Beisel, and R. Keller. Distributed computing in a heterogenous computing environment. In *Proceedings of EuroPVM/MPI'98*, 1998.

[GW97] A. S. Grimshaw and W. A. Wulf. The Legion vision of a worldwide virtual computer. *Communications of the ACM*, 40(1):39–45, January 1997.

[GWWL94] A. Grimshaw, J. Weissman, E. West, and E. Lyot. Metasystems: An approach combining parallel processing and heterogeneous distributed computing systems. *Journal of Parallel and Distributed Computing*, 21(3):257–270, 1994.

[HKL+00] M. Hadida, Y. Kadobayashi, S. Lamont, H.W. Braun, B. Fink, T. Hutton, A. Kamrath, H. Mori, and M.H. Ellisman. Advanced networking for telemicroscopy. In *Proceedings of the Tenth Annual Internet Society Conference*, 2000.

[Hor01] P. Horn. The IBM vision for autonomic computing. Technical report, IBM, 2001. Available from `www.research.ibm.com/autonomic/manifesto`.

[Jac02] K. Jackson. pyGlobus: a Python interface to the Globus toolkit. *Concurrency and Computation: Practice and Experience*, 14(13-15):1075–1084, 2002.

[JGN99] William E. Johnston, Dennis Gannon, and Bill Nitzberg. Grids as production computing environments: The engineering aspects of NASA's Information Power Grid. In *Proceedings of the Eighth IEEE International Symposium on High-Performance Distributed Computing (HPDC-8)*, 1999.

[Joh99] William Johnston. Realtime widely distributed instrumentation systems. In Ian Foster and Carl Kesselman, editors, *The Grid: Blueprint for a New Computing Infrastructure*, chapter 4, pages 75–103. Morgan Kaufmann, 1999.

[KNP01] K. Kurowski, J. Nabrzyski, and J. Pukacki. User preference driven multiobjective resource management in Grid environments. In *Proceedings of the First IEEE International Symposium on Cluster Computing and the Grid (CCGrid'01)*, May 2001.

[KPF04] C. Kesselman, T. Prudhomme, and I. Foster. Distributed telepresence: The NEESgrid earthquake engineering collaboratory. In Ian Foster and Carl Kesselman, editors, *The Grid: Blueprint for a*

New Computing Infrastructure (Second Edition). Morgan Kaufmann, 2004.

[KTF03] N. Karonis, B. Toonen, and I. Foster. MPICH-G2: A Grid-enabled implementation of the message passing interface. *Journal of Parallel and Distributed Computing*, *to appear,* 2003.

[LLM88] M. J. Litzkow, M. Livny, and M. W. Mutka. Condor - a hunter of idle workstations. In *Proceedings of the 8th International Conference on Distributed Computing Systems*, pages 104–111, 1988.

[LW04] David Levine and Mark Wirt. Interactivity with scalability: Infrastructure for multiplayer games. In Ian Foster and Carl Kesselman, editors, *The Grid: Blueprint for a New Computing Infrastructure (Second Edition)*. Morgan Kaufmann, 2004.

[LWW03] David Levine, Mark Wirt, and Barry Whitebook. *Practical Grid Computing for Massively Multiplayer Games*. Charles River Media, 2003.

[Mes99] Paul Messina. Distributed supercomputing applications. In *The Grid: Blueprint for a New Computing Infrastructure*, pages 55–73. Morgan Kaufmann, 1999.

[Nov02] J. Novotny. The Grid Portal Development Kit. *Concurrency and Computation: Practice and Experience*, 14(13-15):1145–1160, 2002.

[PAD+02] Norman W Paton, Malcolm P Atkinson, Vijay Dialani, Dave Pearson, Tony Storey, and Paul Watson. Database access and integration services on the Grid. Technical report, U.K. National eScience Center, 2002. Available from `http://umbriel.dcs.gla.ac.uk/Nesc/general/technical_papers/dbtf.pdf`.

[PKF+01] T. Prudhomme, C. Kesselman, T. Finholt, I. Foster, D. Parsons, D. Abrams, J.-P. Bardet, R. Pennington, J. Towns, R. Butler, J. Futrelle, N. Zaluzec, and J. Hardin. NEESgrid: A distributed virtual laboratory for advanced earthquake experimentation and simulation: Scoping study. Technical report, NEESgrid, Technical Report, 2001. Available from `http://www.neesgrid.org/documents/NEESgrid_TR.2001-01.pdf`.

[PLL+03] S. T. Peltier, A. W. Lin, D. Lee, S. Mock, S. Lamont, T. Molina, M. Wong, L. Dai, M. E. Martone, and M. H. Ellisman. The telescience portal for tomography applications. *Journal of Parallel and Distributed Computing*, 2003.

[RAD+02] M. Russell, G. Allen, G. Daues, I. Foster, E. Seidel, J. Novotny, J. Shalf, and G. von Laszewski. The astrophysics simulation collaboratory: A science portal enabling community software development. *Cluster Computing*, 5(3):297–304, 2002.

[RLS98] R. Raman, M. Livny, and M. Solomon. Matchmaking: Distributed resource management for high throughput computing. In *Proceedings of the Seventh IEEE International Symposium on High-Performance Distributed Computing (HPDC-7)*, 1998.

[SG01] Alexander Szalay and Jim Gray. The world-wide telescope. *Science*, 293:2037–2040, 2001.

[SG04] A. Szalay and J. Gray. Scientific data federation: The world wide telescope. In Ian Foster and Carl Kesselman, editors, *The Grid: Blueprint for a New Computing Infrastructure (Second Edition)*. Morgan Kaufmann, 2004.

[Ste04] Rick Stevens. Group-oriented collaboration: The Access Grid collaboration system. In Ian Foster and Carl Kesselman, editors, *The Grid: Blueprint for a New Computing Infrastructure (Second Edition)*. Morgan Kaufmann, 2004.

[Sun90] V. Sunderam. PVM: A framework for parallel distributed computing. *Concurrency: Practice & Experience*, 2(4):315–339, 1990.

[SWDC97] R. Stevens, P. Woodward, T. DeFanti, and C. Catlett. From the I-WAY to the national technology Grid. *Communications of the ACM*, 40(11):50–61, 1997.

[TEF+02] S. Tuecke, D. Engert, I. Foster, M. Thompson, L. Pearlman, and C. Kesselman. Internet X.509 public key infrastructure proxy certificate profile. Technical Report draft-ietf-pkix-proxy-01.txt, Internet Engineering Task Force, 2002. Available from `http://www.ietf.org/internet-drafts/draft-ietf-pkix-proxy-06.txt`.

[TMB+01] M. Thomas, S. Mock, J. Boisseau, M. Dahan, K. Mueller, and S. Sutton. The GridPort toolkit architecture for building Grid portals. In *Proceedings of the Tenth IEEE International Symposium on High-Performance Distributed Computing (HPDC-10)*, 2001.

[TNS+02] Y. Tanaka, H. Nakada, S. Sekiguchi, T. Suzumura, and S. Matsuoka. Ninf-G: A reference implementation of RPC based programming middleware for Grid computing. *Journal of Grid Computing*, 2002.

[WSF+03] V. Welch, F. Siebenlist, I. Foster, J. Bresnahan, K. Cajkowski, J. Gawor, C. Kesselman, S. Meder, L. Pearlman, and S. Tuecke. GSI: Security for Grid services. In *Proceedings of the Twelfth IEEE International Symposium on High-Performance Distributed Computing (HPDC-12)*, 2003.

[XHLL04] Ming Xu, Zhenhua Hu, Weihong Long, and Wayne Liu. Service virtualization: Infrastructure and applications. In Ian Foster and Carl Kesselman, editors, *The Grid: Blueprint for a New Computing Infrastructure (Second Edition)*. Morgan Kaufmann, 2004.