1

The Grid: past, present, future

Fran Berman,[1] Geoffrey Fox,[2] and Tony Hey[3,4]

[1]San Diego Supercomputer Center, and Department of Computer Science and Engineering, University of California, San Diego, California, [2]Indiana University, Bloomington, Indiana, [3]EPSRC, Swindon, United Kingdom, [4]University of Southampton, Southampton, United Kingdom

1.1 THE GRID

The Grid is the computing and data management infrastructure that will provide the electronic underpinning for a global society in business, government, research, science and entertainment [1–5]. Grids, illustrated in Figure 1.1, integrate networking, communication, computation and information to provide a virtual platform for computation and data management in the same way that the Internet integrates resources to form a virtual platform for information. The Grid is transforming science, business, health and society. In this book we consider the Grid in depth, describing its immense promise, potential and complexity from the perspective of the community of individuals working hard to make the Grid vision a reality.

> *Grid infrastructure will provide us with the ability to dynamically link together resources as an ensemble to support the execution of large-scale, resource-intensive, and distributed applications.*

Grid Computing – Making the Global Infrastructure a Reality. Edited by F. Berman, G. Fox and T. Hey
© 2002 John Wiley & Sons, Ltd ISBN: 0-470-85319-0

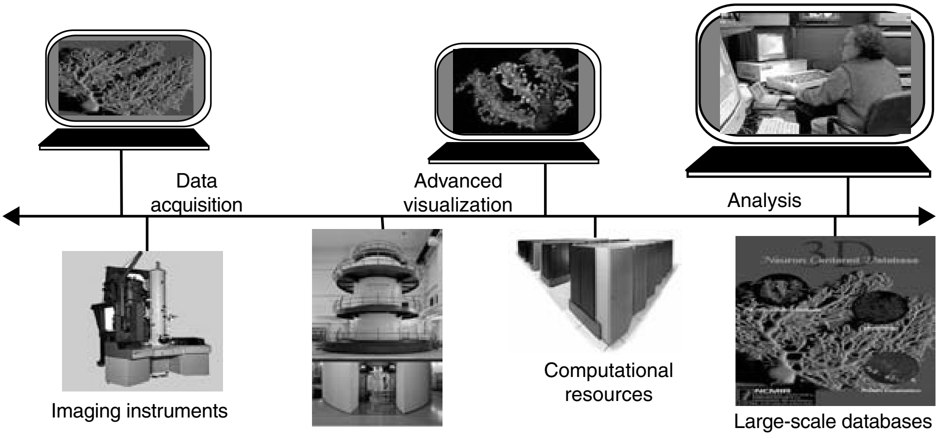

Figure 1.1 Grid resources linked together for neuroscientist Mark Ellisman's Telescience application (http://www.npaci.edu/Alpha/telescience.html).

Large-scale Grids are intrinsically distributed, heterogeneous and dynamic. They promise effectively infinite cycles and storage, as well as access to instruments, visualization devices and so on without regard to geographic location. Figure 1.2 shows a typical early successful application with information pipelined through distributed systems [6]. The reality is that to achieve this promise, complex systems of software and services must be developed, which allow access in a user-friendly way, which allow resources to be used together efficiently, and which enforce policies that allow communities of users to coordinate resources in a stable, performance-promoting fashion. Whether users access the Grid to use one resource (a single computer, data archive, etc.), or to use several resources in aggregate as a coordinated 'virtual computer', the Grid permits users to interface with the resources in a uniform way, providing a comprehensive and powerful platform for global computing and data management.

In the United Kingdom this vision of increasingly global collaborations for scientific research is encompassed by the term *e-Science* [7]. The UK e-Science Program is a major initiative developed to promote scientific and data-oriented Grid application development for both science and industry. The goals of the e-Science initiative are to assist in global efforts to develop a Grid e-Utility infrastructure for e-Science applications, which will support in silico experimentation with huge data collections, and assist the development of an integrated campus infrastructure for all scientific and engineering disciplines. e-Science merges a decade of simulation and compute-intensive application development with the immense focus on data required for the next level of advances in many scientific disciplines. The UK program includes a wide variety of projects including health and medicine, genomics and bioscience, particle physics and astronomy, environmental science, engineering design, chemistry and material science and social sciences. Most e-Science projects involve both academic and industry participation [7].

Box 1.1 Summary of Chapter 1

This chapter is designed to give a high-level motivation for the book. In Section 1.2, we highlight some historical and motivational building blocks of the Grid – described in more detail in Chapter 3. Section 1.3 describes the current community view of the Grid with its basic architecture. Section 1.4 contains four building blocks of the Grid. In particular, in Section 1.4.1 we review the evolution of the networking infrastructure including both the desktop and cross-continental links, which are expected to reach gigabit and terabit performance, respectively, over the next five years. Section 1.4.2 presents the corresponding computing backdrop with 1 to 40 teraflop performance today moving to petascale systems by the end of the decade. The U.S. National Science Foundation (NSF) TeraGrid project illustrates the state-of-the-art of current Grid technology. Section 1.4.3 summarizes many of the regional, national and international activities designing and deploying Grids. Standards, covered in Section 1.4.4 are a different but equally critical building block of the Grid. Section 1.5 covers the critical area of applications on the Grid covering life sciences, engineering and the physical sciences. We highlight new approaches to science including the importance of collaboration and the e-Science [7] concept driven partly by increased data. A short section on commercial applications includes the e-Enterprise/Utility [10] concept of computing power on demand. Applications are summarized in Section 1.5.7, which discusses the characteristic features of 'good Grid' applications like those illustrated in Figures 1.1 and 1.2. These show instruments linked to computing, data archiving and visualization facilities in a local Grid. Part D and Chapter 35 of the book describe these applications in more detail. Futures are covered in Section 1.6 with the intriguing concept of autonomic computing developed originally by IBM [10] covered in Section 1.6.1 and Chapter 13. Section 1.6.2 is a brief discussion of Grid programming covered in depth in Chapter 20 and Part C of the book. There are concluding remarks in Sections 1.6.3 to 1.6.5.

General references can be found in [1–3] and of course the chapters of this book [4] and its associated Web site [5]. The reader's guide to the book is given in the preceding preface. Further, Chapters 20 and 35 are guides to Parts C and D of the book while the later insert in this chapter (Box 1.2) has comments on Parts A and B of this book. Parts of this overview are based on presentations by Berman [11] and Hey, conferences [2, 12] and a collection of presentations from the Indiana University on networking [13–15].

In the next few years, the Grid will provide the fundamental infrastructure not only for e-Science but also for e-Business, e-Government, e-Science and e-Life. This emerging infrastructure will exploit the revolutions driven by Moore's law [8] for CPU's, disks and instruments as well as Gilder's law [9] for (optical) networks. In the remainder of this chapter, we provide an overview of this immensely important and exciting area and a backdrop for the more detailed chapters in the remainder of this book.

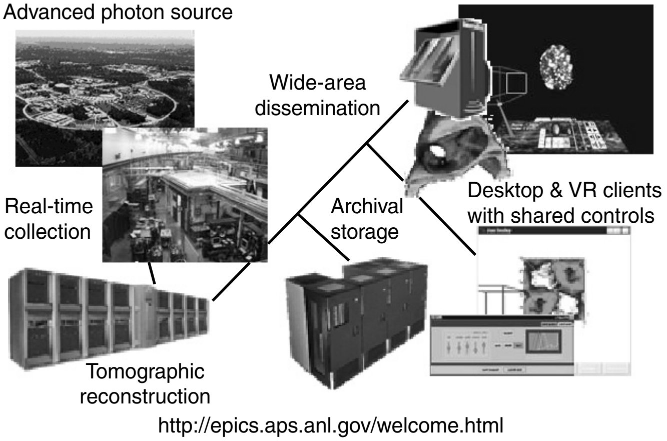
Advanced photon source
Wide-area dissemination
Real-time collection
Archival storage
Desktop & VR clients with shared controls
Tomographic reconstruction
http://epics.aps.anl.gov/welcome.html

Figure 1.2 Computational environment for analyzing real-time data taken at Argonne's advanced photon source was an early example of a data-intensive Grid application [6]. The picture shows data source at APS, network, computation, data archiving, and visualization. This figure was derived from work reported in "Real-Time Analysis, Visualization, and Steering of Microtomography Experiments at Photon Sources", Gregor von Laszewski, Mei-Hui Su, Joseph A. Insley, Ian Foster, John Bresnahan, Carl Kesselman, Marcus Thiebaux, Mark L. Rivers, Steve Wang, Brian Tieman, Ian McNulty, Ninth SIAM Conference on Parallel Processing for Scientific Computing, Apr. 1999.

1.2 BEGINNINGS OF THE GRID

It is instructive to start by understanding the influences that came together to ultimately influence the development of the Grid. Perhaps the best place to start is in the 1980s, a decade of intense research, development and deployment of hardware, software and applications for parallel computers. Parallel computing in the 1980s focused researchers' efforts on the development of algorithms, programs and architectures that supported simultaneity. As application developers began to develop large-scale codes that pushed against the resource limits of even the fastest parallel computers, some groups began looking at distribution beyond the boundaries of the machine as a way of achieving results for problems of larger and larger size.

During the 1980s and 1990s, software for parallel computers focused on providing powerful mechanisms for managing communication between processors, and development and execution environments for parallel machines. Parallel Virtual Machine (PVM), Message Passing Interface (MPI), High Performance Fortran (HPF), and OpenMP were developed to support communication for scalable applications [16]. Successful application paradigms were developed to leverage the immense potential of shared and distributed memory architectures. Initially it was thought that the Grid would be most useful in extending parallel computing paradigms from tightly coupled clusters to geographically distributed systems. However, in practice, the Grid has been utilized more as a platform for the integration of loosely coupled applications – some components of which might be running in parallel on a low-latency parallel machine – and for linking disparate resources (storage, computation, visualization, instruments). The fundamental Grid task of managing these heterogeneous components as we scale the size of distributed systems replaces that of the tight synchronization of the typically identical [in program but not data as in the SPMD (single program multiple data) model] parts of a domain-decomposed parallel application.

During the 1980s, researchers from multiple disciplines also began to come together to attack 'Grand Challenge' problems [17], that is, key problems in science and engineering for which large-scale computational infrastructure provided a fundamental tool to achieve new scientific discoveries. The Grand Challenge and multidisciplinary problem teams provided a model for collaboration that has had a tremendous impact on the way large-scale science is conducted to date. Today, interdisciplinary research has not only provided a model for collaboration but has also inspired whole disciplines (e.g. bioinformatics) that integrate formerly disparate areas of science.

The problems inherent in conducting multidisciplinary and often geographically dispersed collaborations provided researchers experience both with *coordination* and *distribution* – two fundamental concepts in Grid Computing. In the 1990s, the US Gigabit testbed program [18] included a focus on distributed metropolitan-area and wide-area applications. Each of the test beds – Aurora, Blanca, Casa, Nectar and Vistanet – was designed with dual goals: to investigate potential testbed network architectures and to explore their usefulness to end users. In this second goal, each testbed provided a venue for experimenting with distributed applications.

The first modern Grid is generally considered to be the information wide-area year (I-WAY), developed as an experimental demonstration project for SC95. In 1995, during the week-long Supercomputing conference, pioneering researchers came together to aggregate a national distributed testbed with over 17 sites networked together by the vBNS. Over 60 applications were developed for the conference and deployed on the I-WAY, as well as a rudimentary Grid software infrastructure (Chapter 4) to provide access, enforce security, coordinate resources and other activities. Developing infrastructure and applications for the I-WAY provided a seminal and powerful experience for the first generation of modern Grid researchers and projects. This was important as the development of Grid research requires a very different focus than distributed computing research. Whereas distributed computing research generally focuses on addressing the problems of geographical separation, Grid research focuses on addressing the problems of integration and management of software.

I-WAY opened the door for considerable activity in the development of Grid software. The Globus [3] (Chapters 6 and 8) and Legion [19–21] (Chapter 10) infrastructure projects explored approaches for providing basic system-level Grid infrastructure. The Condor project [22] (Chapter 11) experimented with high-throughput scheduling, while the AppLeS [23], APST (Chapter 33), Mars [24] and Prophet [25] projects experimented with high-performance scheduling. The Network Weather Service [26] project focused on resource monitoring and prediction, while the Storage Resource Broker (SRB) [27] (Chapter 16) focused on uniform access to heterogeneous data resources. The NetSolve [28] (Chapter 24) and Ninf [29] (Chapter 25) projects focused on remote computation via a client-server model. These, and many other projects, provided a foundation for today's Grid software and ideas.

In the late 1990s, Grid researchers came together in the Grid Forum, subsequently expanding to the Global Grid Forum (GGF) [2], where much of the early research is now evolving into the standards base for future Grids. Recently, the GGF has been instrumental in the development of the Open Grid Services Architecture (OGSA), which integrates Globus and Web services approaches (Chapters 7, 8, and 9). OGSA is being developed by both the United States and European initiatives aiming to define core services for a wide variety of areas including:

- *Systems Management and Automation*
- *Workload/Performance Management*
- *Security*
- *Availability/Service Management*
- *Logical Resource Management*
- *Clustering Services*
- *Connectivity Management*
- *Physical Resource Management.*

Today, the Grid has gone global, with many worldwide collaborations between the United States, European and Asia-Pacific researchers. Funding agencies, commercial vendors, academic researchers, and national centers and laboratories have come together to form a community of broad expertise with enormous commitment to building the Grid. Moreover, research in the related areas of networking, digital libraries, peer-to-peer computing, collaboratories and so on are providing additional ideas relevant to the Grid.

Although we tend to think of the Grid as a result of the influences of the last 20 years, some of the earliest roots of the Grid can be traced back to J.C.R. Licklider, many years before this. 'Lick' was one of the early computing and networking pioneers, who set the scene for the creation of the ARPANET, the precursor to today's Internet. Originally an experimental psychologist at MIT working on psychoacoustics, he was concerned with the amount of data he had to work with and the amount of time he required to organize and analyze his data. He developed a vision of networked computer systems that would be able to provide fast, automated support systems for human decision making [30]:

> *'If such a network as I envisage nebulously could be brought into operation, we could have at least four large computers, perhaps six or eight small computers, and a great assortment of disc files and magnetic tape units – not to mention remote consoles and teletype stations – all churning away'*

In the early 1960s, computers were expensive and people were cheap. Today, after thirty odd years of Moore's Law [8], the situation is reversed and individual laptops now have more power than Licklider could ever have imagined possible. Nonetheless, his insight that the deluge of scientific data would require the harnessing of computing resources distributed around the galaxy was correct. Thanks to the advances in networking and software technologies, we are now working to implement this vision.

In the next sections, we provide an overview of the present Grid Computing and its emerging vision for the future.

1.3 A COMMUNITY GRID MODEL

Over the last decade, the Grid community has begun to converge on a layered model that allows development of the complex system of services and software required to integrate Grid resources. This model, explored in detail in Part B of this book, provides a layered abstraction of the Grid. Figure 1.3 illustrates the *Community Grid Model* being developed in a loosely coordinated manner throughout academia and the commercial sector. We begin discussion by understanding each of the layers in the model.

The bottom horizontal layer of the Community Grid Model consists of the hardware *resources* that underlie the Grid. Such resources include computers, networks, data archives, instruments, visualization devices and so on. They are distributed, heterogeneous and have very different performance profiles (contrast performance as measured in FLOPS or memory bandwidth with performance as measured in bytes and data access time). Moreover, the resource pool represented by this layer is highly dynamic, both as a result of new resources being added to the mix and old resources being retired, and as a result of varying observable performance of the resources in the shared, multiuser environment of the Grid.

The next horizontal layer (*common infrastructure*) consists of the software services and systems which virtualize the Grid. Community efforts such as NSF's Middleware Initiative (NMI) [31], OGSA (Chapters 7 and 8), as well as emerging de facto standards such as Globus provide a commonly agreed upon layer in which the Grid's heterogeneous and dynamic resource pool can be accessed. The key concept at the common infrastructure layer is community agreement on software, which will represent the Grid as a unified virtual platform and provide the target for more focused software and applications.

The next horizontal layer (user and application-focused Grid *middleware, tools and services*) contains software packages built atop the common infrastructure. This software serves to enable applications to more productively use Grid resources by masking some of the complexity involved in system activities such as authentication, file transfer, and

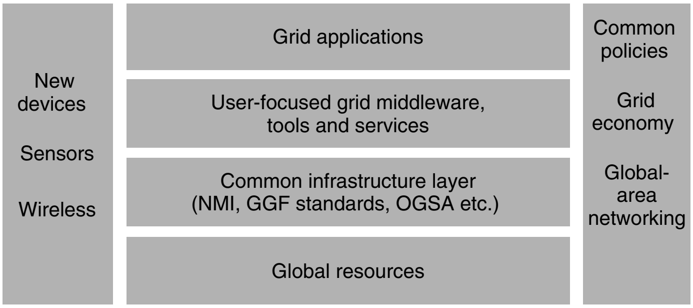

Figure 1.3 Layered architecture of the Community Grid Model.

so on. Portals, community codes, application scheduling software and so on reside in this layer and provide middleware that connects applications and users with the common Grid infrastructure.

The topmost horizontal layer (*Grid applications*) represents applications and users. The Grid will ultimately be only as successful as its user community and all of the other horizontal layers must ensure that the Grid presents a robust, stable, usable and useful computational and data management platform to the user. Note that in the broadest sense, even applications that use only a single resource on the Grid are Grid applications if they access the target resource through the uniform interfaces provided by the Grid infrastructure.

The vertical layers represent the next steps for the development of the Grid. The vertical layer on the left represents the *influence of new devices* – sensors, PDAs, and wireless. Over the next 10 years, these and other new devices will need to be integrated with the Grid and will exacerbate the challenges of managing heterogeneity and promoting performance. At the same time, the increasing globalization of the Grid will require serious consideration of *policies for sharing* and using resources, global-area networking and the development of *Grid economies* (the vertical layer on the right – see Chapter 32). As we link together national Grids to form a Global Grid, it will be increasingly important to develop Grid social and economic policies which ensure the stability of the system, promote the performance of the users and successfully integrate disparate political, technological and application cultures.

The Community Grid Model provides an abstraction of the large-scale and intense efforts of a community of Grid professionals, academics and industrial partners to build the Grid. In the next section, we consider the lowest horizontal layers (individual resources and common infrastructure) of the Community Grid Model.

1.4 BUILDING BLOCKS OF THE GRID

1.4.1 Networks

The heart of any Grid is its network – networks link together geographically distributed resources and allow them to be used collectively to support execution of a single application. If the networks provide ‘big pipes’, successful applications can use distributed resources in a more integrated and data-intensive fashion; if the networks provide ‘small pipes’, successful applications are likely to exhibit minimal communication and data transfer between program components and/or be able to tolerate high latency.

At present, Grids build on ubiquitous high-performance networks [13, 14] typified by the Internet2 Abilene network [15] in the United States shown in Figures 1.4 and 1.5. In 2002, such national networks exhibit roughly 10 Gb s^{-1} backbone performance. Analogous efforts can be seen in the UK SuperJanet [40] backbone of Figure 1.6 and the intra-Europe GEANT network [41] of Figure 1.7. More globally, Grid efforts can leverage international networks that have been deployed (illustrated in Figure 1.8) including CA*net3 from Canarie in Canada [42] and the Asian network APAN [43], (shown in detail in Figure 1.9). Such national network backbone performance is typically complemented by

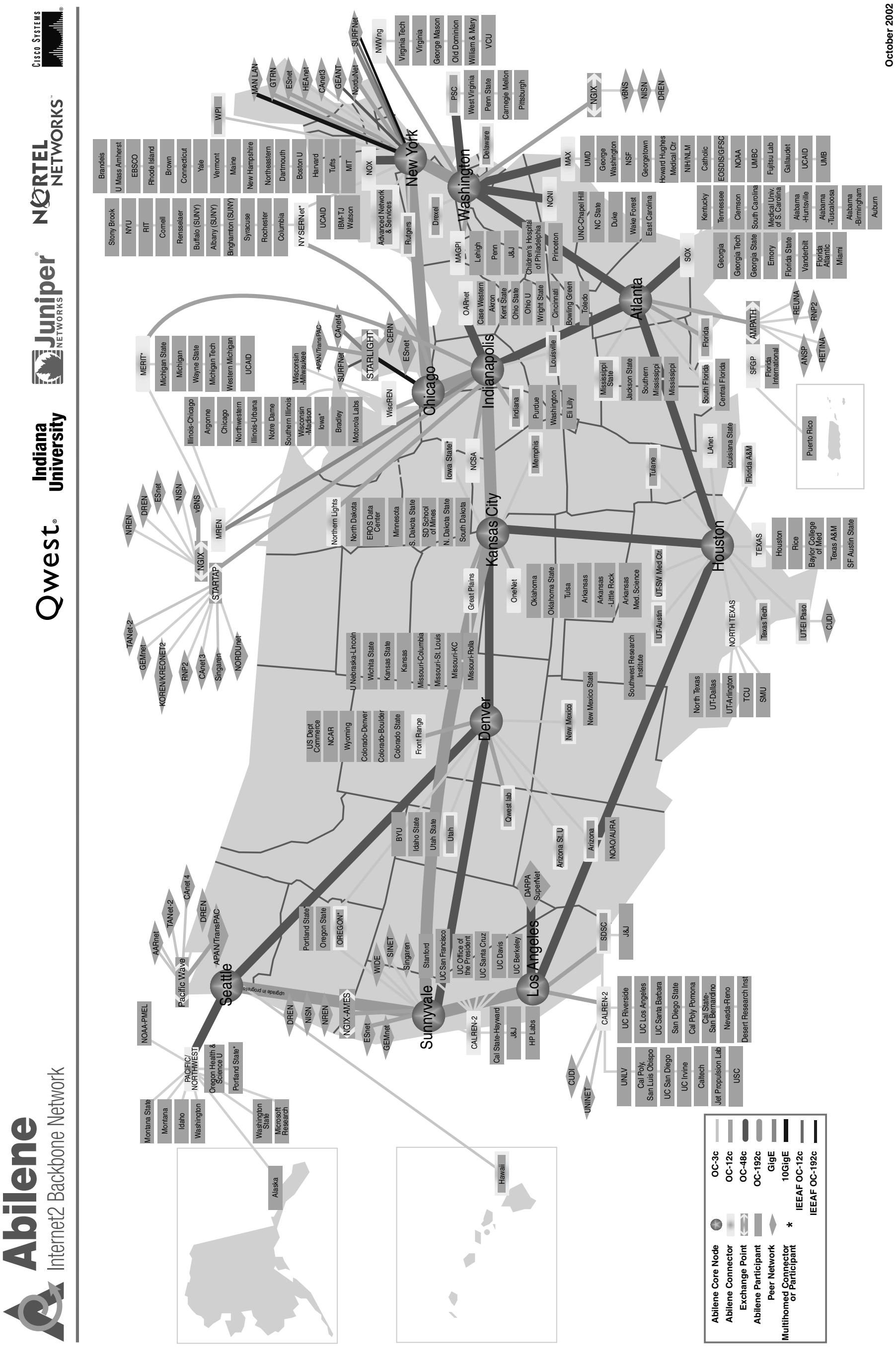

Figure 1.4 Sites on the Abilene Research Network.

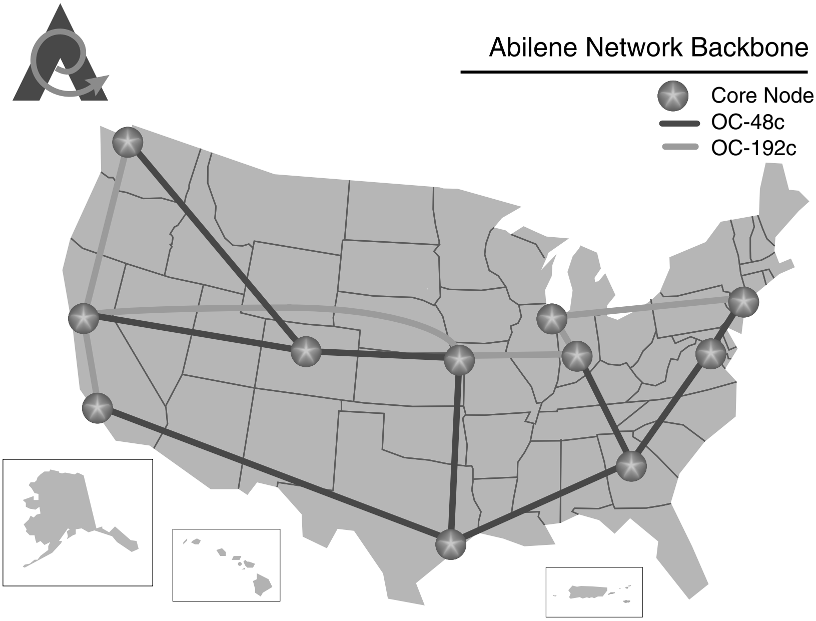

Figure 1.5 Backbone of Abilene Internet2 Network in USA.

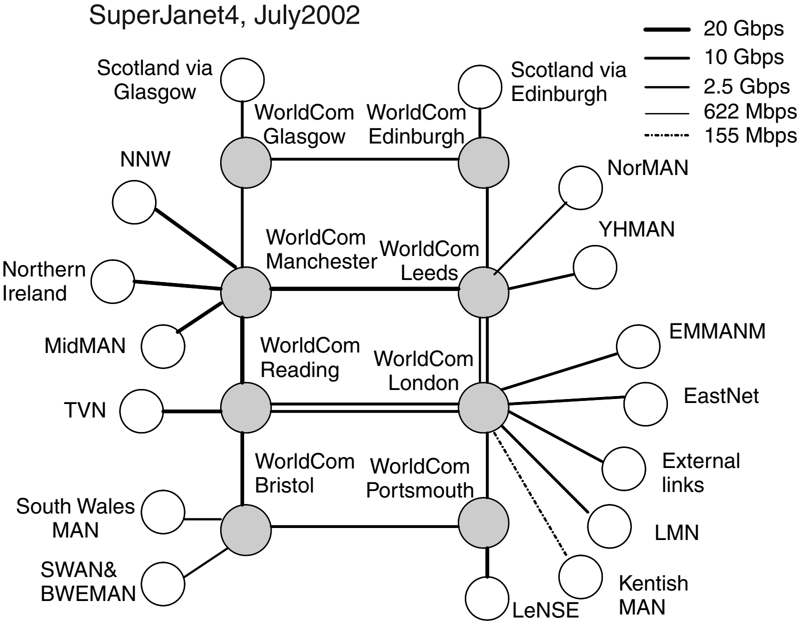

Figure 1.6 United Kingdom National Backbone Research and Education Network.

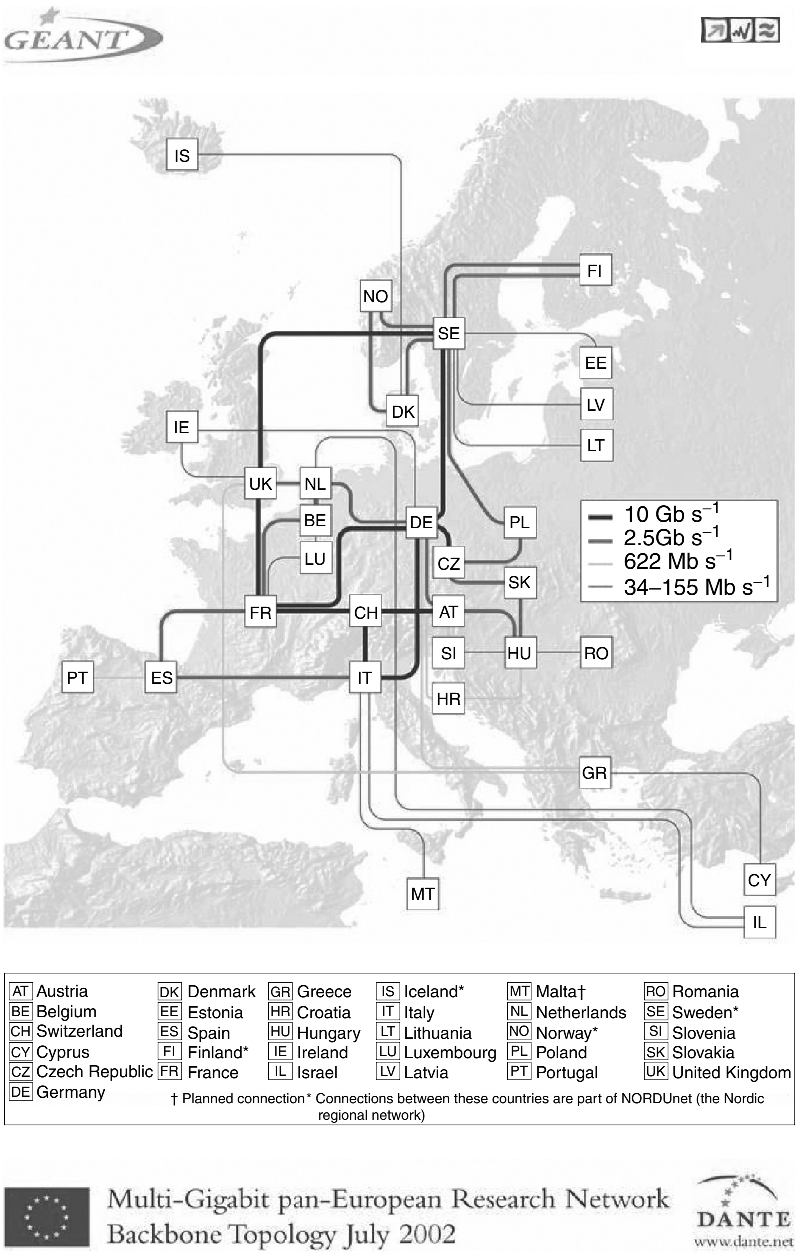

Figure 1.7 European Backbone Research Network GEANT showing countries and backbone speeds.

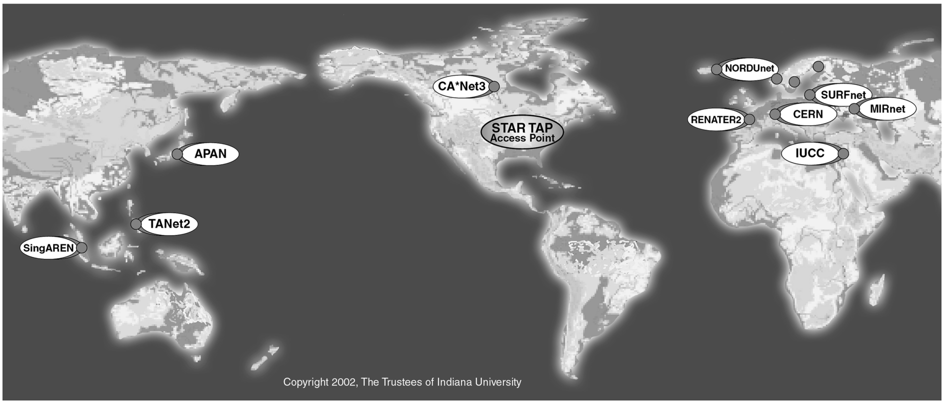

Figure 1.8 International Networks.

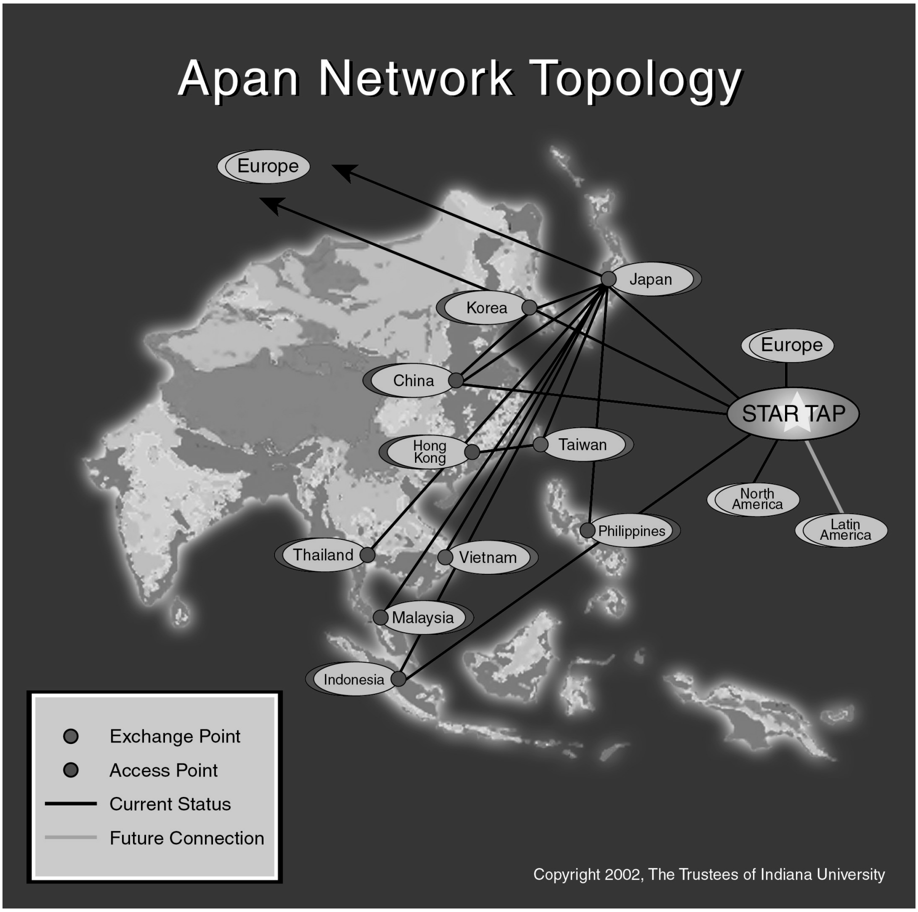

Figure 1.9 APAN Asian Network.

a 1 Gb s^{-1} institution-to-backbone link and by a 10 to 100 Mb s^{-1} desktop-to-institutional network link.

Although there are exceptions, one can capture a typical leading Grid research environment as a 10 : 1 : 0.1 Gbs^{-1} ratio representing *national: organization: desktop* links.

Today, new national networks are beginning to change this ratio. The GTRN or Global Terabit Research Network initiative shown in Figures 1.10 and 1.11 link national networks in Asia, the Americas and Europe with a performance similar to that of their backbones [44]. By 2006, GTRN aims at a 1000 : 1000 : 100 : 10 : 1 gigabit performance ratio representing *international backbone: national: organization: optical desktop: Copper desktop* links. This implies a performance increase of over a factor of 2 per year in network performance, and clearly surpasses expected CPU performance and memory size increases of Moore's law [8] (with a prediction of a factor of two in chip density improvement every 18 months). This continued difference between network and CPU performance growth will continue to enhance the capability of distributed systems and lessen the gap between Grids and geographically centralized approaches. We should note that although network bandwidth will improve, we do not expect latencies to improve significantly. Further, as seen in the telecommunications industry in 2000–2002, in many ways network performance is increasing 'faster than demand' even though organizational issues lead to problems. A critical area of future work is network quality of service and here progress is less clear. Networking performance can be taken into account at the application level as in AppLes and APST ([23] and Chapter 33), or by using the Network Weather Service [26] and NaradaBrokering (Chapter 22).

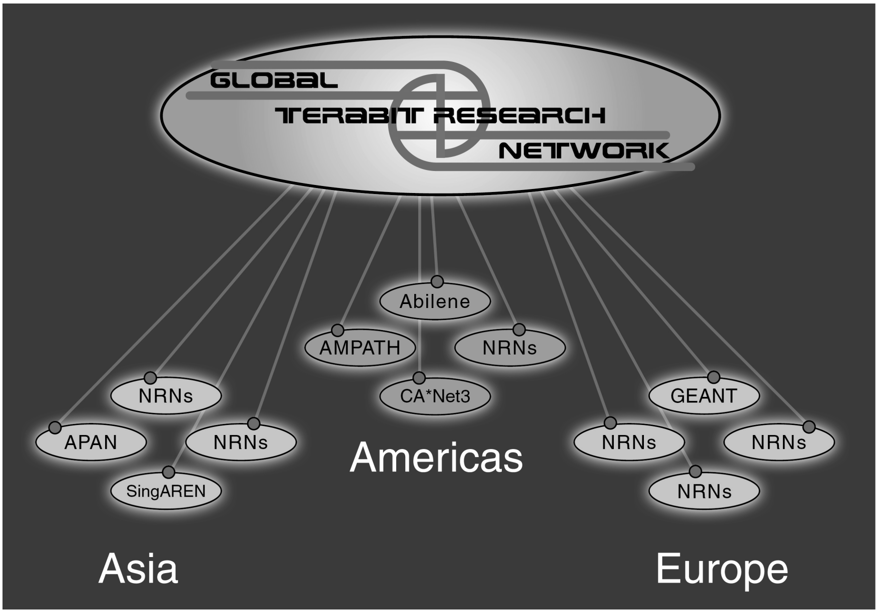

Figure 1.10 Logical GTRN Global Terabit Research Network.

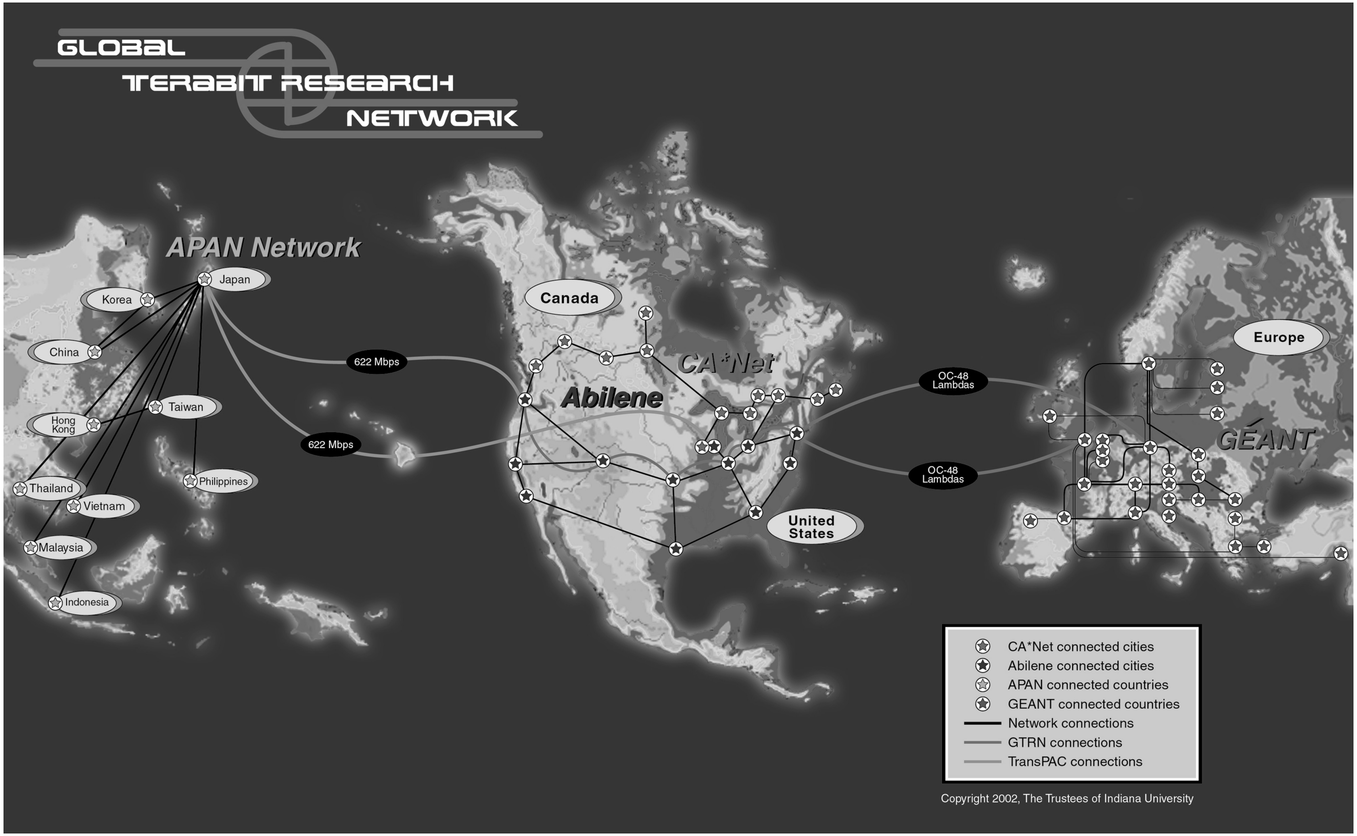

Figure 1.11 Physical GTRN Global Terabit Research Network.

High-capacity networking increases the capability of the Grid to support both parallel and distributed applications. In the future, wired networks will be further enhanced by continued improvement in wireless connectivity [45], which will drive integration of smaller and smaller devices into the Grid. The desktop connectivity described above will include the pervasive PDA (Personal Digital Assistant included in universal access discussion of Chapter 18) that will further promote the Grid as a platform for e-Science, e-Commerce and e-Education (Chapter 43).

1.4.2 Computational 'Nodes' on the Grid

Networks connect resources on the Grid, the most prevalent of which are computers with their associated data storage. Although the computational resources can be of any level of power and capability, some of the most interesting Grids for scientists involve nodes that are themselves high-performance parallel machines or clusters. Such high-performance Grid 'nodes' provide major resources for simulation, analysis, data mining and other compute-intensive activities. The performance of the most high-performance nodes on the Grid is tracked by the Top500 site [46] (Figure 1.12). Extrapolations of this information indicate that we can expect a peak single machine performance of 1 petaflops/sec (10^{15} operations per second) by around 2010.

Contrast this prediction of power to the present situation for high-performance computing. In March 2002, Japan's announcement of the NEC Earth Simulator machine shown in Figure 1.13 [47], which reaches 40 teraflops s^{-1} with a good sustained to peak performance rating, garnered worldwide interest. The NEC machine has 640 eight-processor nodes and offers 10 terabytes of memory and 700 terabytes of disk space. It has already been used for large-scale climate modeling. The race continues with Fujitsu announcing

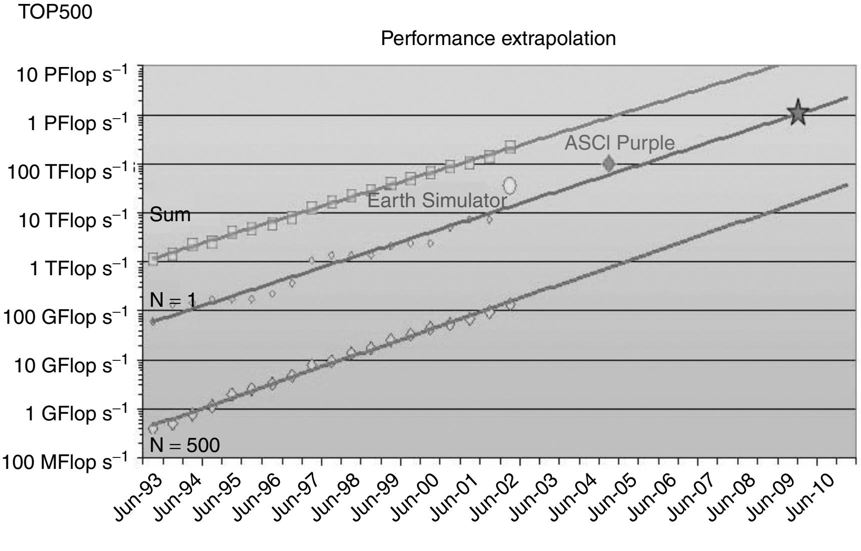

Figure 1.12 Top 500 performance extrapolated from 1993 to 2010.

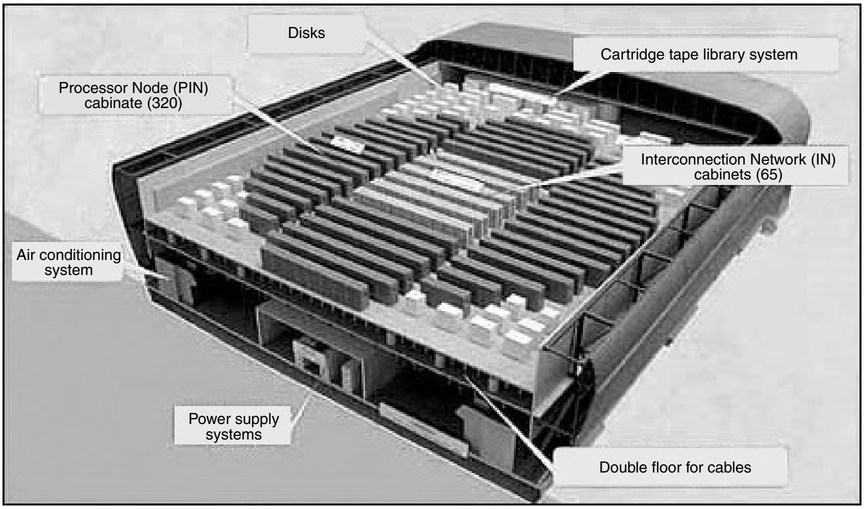
Disks
Cartridge tape library system
Processor Node (PIN) cabinate (320)
Interconnection Network (IN) cabinets (65)
Air conditioning system
Power supply systems
Double floor for cables

Figure 1.13 Japanese Earth Simulator 40 Teraflop Supercomputer.

in August 2002, the HPC2500 with up to 16 384 processors and 85 teraflops s^{-1} peak performance [48]. Until these heroic Japanese machines, DOE's ASCI program [49], shown in Figure 1.14, had led the pack with the ASCI White machine at Livermore National Laboratory peaking at 12 teraflops s^{-1}. Future ASCI machines will challenge for the Top 500 leadership position!

Such nodes will become part of future Grids. Similarly, large data archives will become of increasing importance. Since it is unlikely that it will be many years, if ever, before it becomes straightforward to move petabytes of data around global networks, data centers will install local high-performance computing systems for data mining and analysis. Complex software environments will be needed to smoothly integrate resources from PDAs (perhaps a source of sensor data) to terascale/petascale resources. This is an immense challenge, and one that is being met by intense activity in the development of Grid software infrastructure today.

1.4.3 Pulling it all together

The last decade has seen a growing number of large-scale Grid infrastructure deployment projects including NASA's Information Power Grid (IPG) [50], DoE's Science Grid [51] (Chapter 5), NSF's TeraGrid [52], and the UK e-Science Grid [7]. NSF has many Grid activities as part of Partnerships in Advanced Computational Infrastructure (PACI) and is developing a new Cyberinfrastructure Initiative [53]. Similar large-scale Grid projects are being developed in Asia [54] and all over Europe – for example, in the Netherlands [55], France [56], Italy [57], Ireland [58], Poland [59] and Scandinavia [60]. The DataTAG project [61] is focusing on providing a transatlantic lambda connection for HEP (High Energy Physics) Grids and we have already described the GTRN [14] effort. Some projects

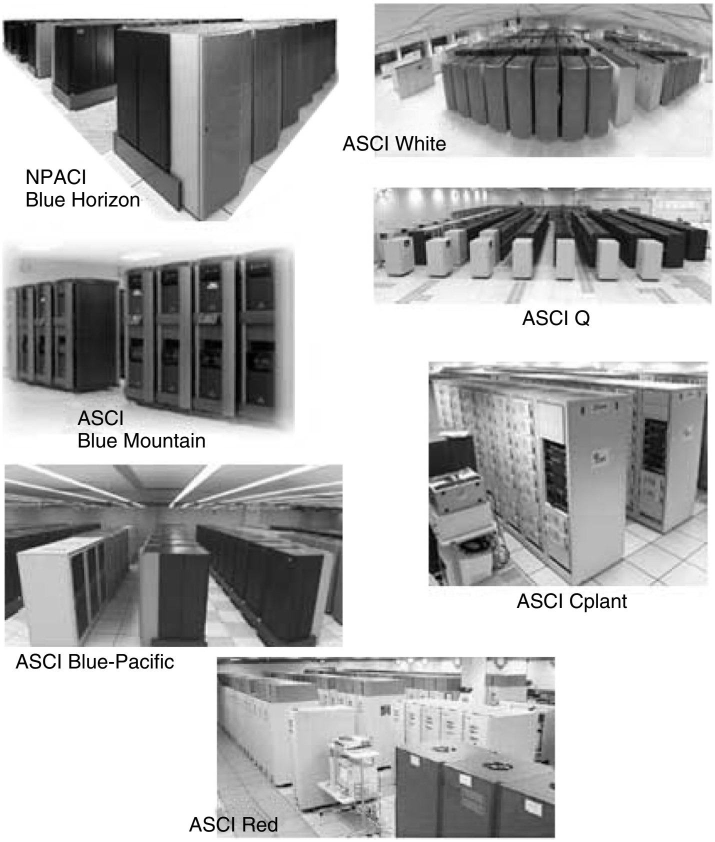

Figure 1.14 Constellation of ASCI Supercomputers.

are developing high-end, high-performance Grids with fast networks and powerful Grid nodes that will provide a foundation of experience for the Grids of the future. The European UNICORE system ([62] and Chapter 29) is being developed as a Grid computing environment to allow seamless access to several large German supercomputers. In the United States, the ASCI program and TeraGrid project are using Globus to develop Grids linking multi-teraflop computers together [63]. There are many support projects associated with all these activities including national and regional centers in the UK e-Science effort [64, 65], the European GRIDS activity [66] and the iVDGL (International Virtual Data Grid Laboratory) [67]. This latter project has identified a Grid Operation Center in analogy with the well-understood network operation center [68].

Much of the critical Grid software is built as part of infrastructure activities and there are important activities focused on software: the Grid Application Development System (GrADS) [69] is a large-scale effort focused on Grid program development and execution environment. Further, NSF's Middleware Initiative (NMI) is focusing on the development and documentation of ready-for-primetime Grid middleware. Europe has started several major software activities [62, 70–75]. Application Grid projects described in more detail in Section 1.5 include magnetic fusion [76], particle physics [68, 77, 78] (Chapter 39), astronomy [77, 79–81] (Chapter 38), earthquake engineering [82] and modeling [83], climate [84], bioinformatics [85, 86] (Chapters 40 and 41) and, more generally, industrial applications [87]. We finally note two useful Web resources [88, 89] that list, respectively, acronyms and major projects in the Grid area.

One of the most significant and coherent Grid efforts in Europe is the UK e-Science Program [7] discussed in Section 1.1. This is built around a coherent set of application Grids linked to a UK national Grid. The new 7 Teraflop (peak) HPC(X) machine from IBM will be located at Daresbury Laboratory and be linked to the UK e-Science Grid [90, 91] shown in Figure 1.15. In addition to the HPC(X) machine, the UK Grid will provide connections to the HPC Computer Services for Academic Research (CSAR) service in Manchester and high-performance clusters only accessible to UK university researchers via Grid digital certificates provided by the UK Grid Certification Authority. This is located at Rutherford Laboratory along with the UK Grid Support Centre and the Engineering Task Force. The UK e-Science Grid is intended to provide a model for a genuine production Grid that can be used by both academics for their research and industry for evaluation. The accompanying set of application projects are developing Grids that will connect and overlap with national Grid testing interoperability and security issues for different virtual communities of scientists. A striking feature of the UK e-Science

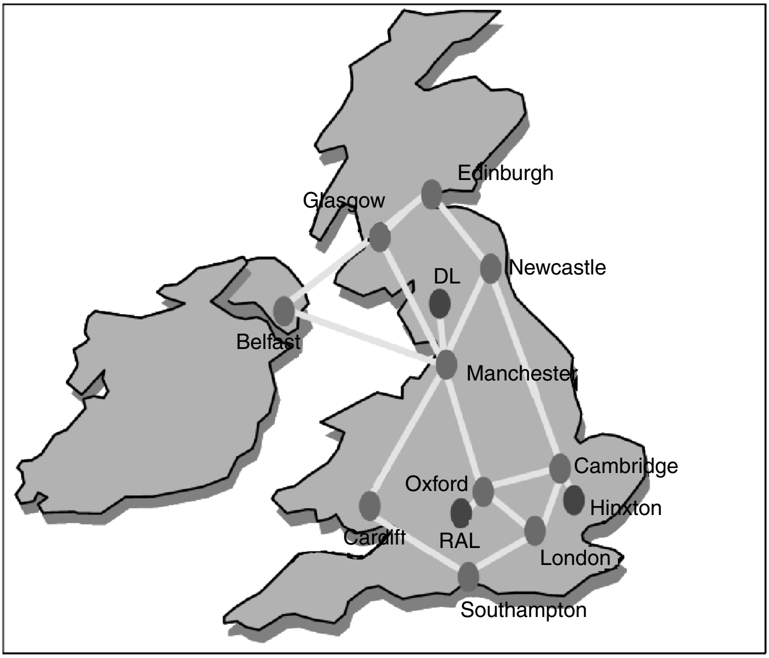

Figure 1.15 UK e-Science Grid.

initiative is the large-scale involvement of industry: over 50 companies are involved in the program, contributing over $30 M to supplement the $180 M funding provided by the UK Government.

The portfolio of the UK e-Science application projects is supported by the Core Program. This provides support for the application projects in the form of the Grid Support Centre and a supported set of Grid middleware. The initial starting point for the UK Grid was the software used by NASA for their IPG – Globus, Condor and SRB as described in Chapter 5. Each of the nodes in the UK e-Science Grid has $1.5 M budget for collaborative industrial Grid middleware projects. The requirements of the e-Science application projects in terms of computing resources, data resources, networking and remote use of facilities determine the services that will be required from the Grid middleware. The UK projects place more emphasis on data access and data federation (Chapters 14, 15 and 17) than traditional HPC applications, so the major focus of the UK Grid middleware efforts are concentrated in this area. Three of the UK e-Science centres – Edinburgh, Manchester and Newcastle – are working with the Globus team and with IBM US, IBM Hursley Laboratory in the United Kingdom, and Oracle UK in an exciting project on data access and integration (DAI). The project aims to deliver new data services within the Globus Open Grid Services framework.

Perhaps the most striking current example of a high-performance Grid is the new NSF TeraGrid shown in Figure 1.16, which links major subsystems at four different sites and will scale to the Pittsburgh Supercomputer Center and further sites in the next few years. The TeraGrid [52] is a high-performance Grid, which will connect the San Diego Supercomputer Center (SDSC), California Institute of Technology (Caltech), Argonne National Laboratory and the National Center for Supercomputing Applications (NCSA).

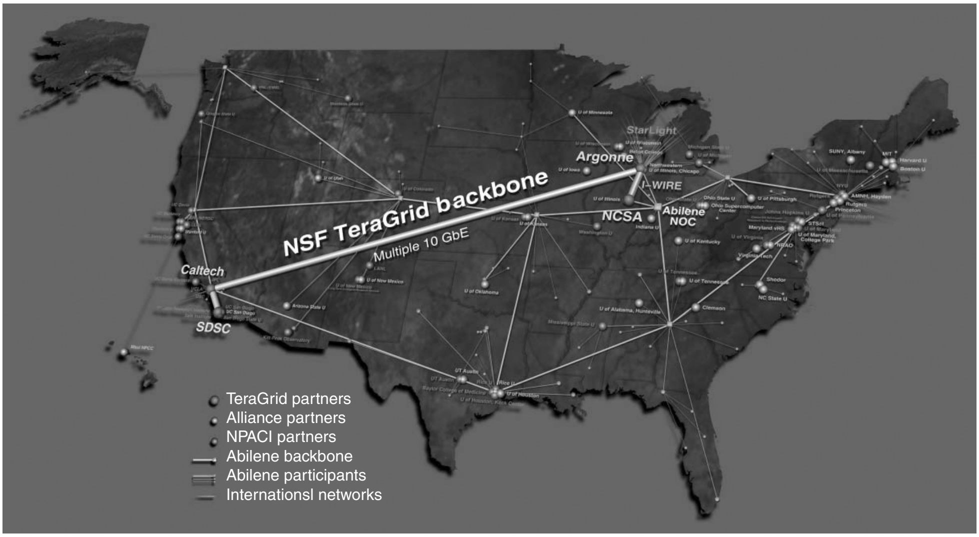

Figure 1.16 USA TeraGrid NSF HPCC system.

Once built, the TeraGrid will link the four in a Grid that will comprise in aggregate over 0.6 petabyte of on-line disk, over 13 teraflops compute performance, and will be linked together by a 40 Gb s^{-1} network.

Each of the four TeraGrid sites specializes in different areas including visualization (Argonne), compute-intensive codes (NCSA), data-oriented computing (SDSC) and scientific collections (Caltech). An overview of the hardware configuration is shown in Figure 1.17. Each of the sites will deploy a cluster that provides users and application developers with an opportunity to experiment with distributed wide-area cluster computing as well as Grid computing. The Extensible Terascale Facility (ETF) adds the Pittsburgh Supercomputer Center to the original four TeraGrid sites. Beyond TeraGrid/ETF, it is the intention of NSF to scale to include additional sites and heterogeneous architectures as the foundation of a comprehensive ‘cyberinfrastructure’ for US Grid efforts [53]. With this as a goal, TeraGrid/ETF software and hardware is being designed to scale from the very beginning.

TeraGrid was designed to push the envelop on data capability, compute capability and network capability simultaneously, providing a platform for the community to experiment with data-intensive applications and more integrated compute-intensive applications. Key choices for the TeraGrid software environment include the identification of Linux as the operating system for each of the TeraGrid nodes, and the deployment of basic, core and advanced Globus and data services.

The goal is for the high-end Grid and cluster environment deployed on TeraGrid to resemble the low-end Grid and cluster environment used by scientists in their own laboratory settings. This will enable a more direct path between the development of test and prototype codes and the deployment of large-scale runs on high-end platforms.

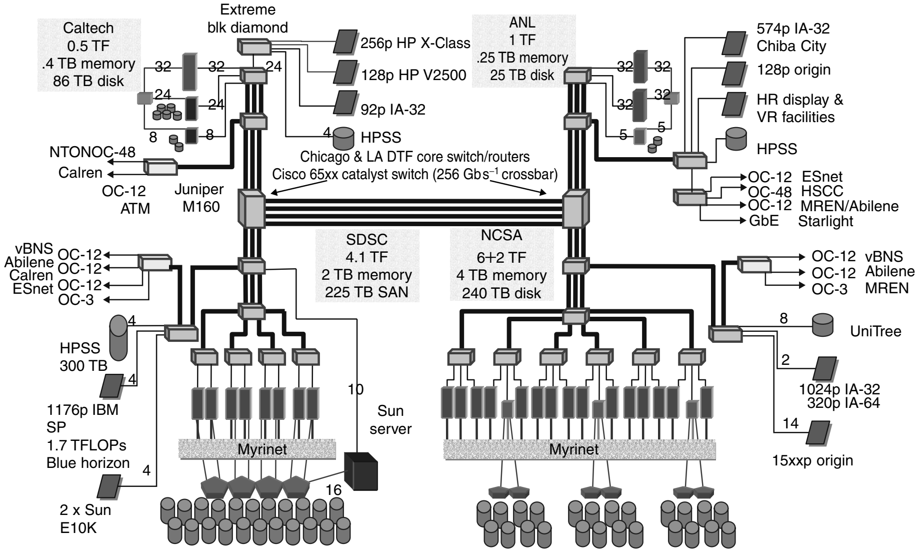

Figure 1.17 TeraGrid system components.

TeraGrid is being developed as a production Grid (analogous to the role that production supercomputers have played over the last two decades as the target of large-scale codes developed on laboratory workstations) and will involve considerable software and human infrastructure to provide access and support for users including portals, schedulers, operations, training, a distributed help-desk, and so on.

1.4.4 Common infrastructure: standards

For the foreseeable future, technology will continue to provide greater and greater potential capability and capacity and will need to be integrated within Grid technologies. To manage this ever-changing technological landscape, Grids utilize a common infrastructure to provide a virtual representation to software developers and users, while allowing the incorporation of new technologies. The development of key standards that allow the complexity of the Grid to be managed by software developers and users without heroic efforts is critical to the success of the Grid.

Both the Internet and the IETF [92], and the Web and the W3C consortium [93] have defined key standards such as TCP/IP, HTTP, SOAP, XML and now WSDL – the Web services definition language that underlines OGSA. Such standards have been critical for progress in these communities. The GGF [2] is now building key Grid-specific standards such as OGSA, the emerging *de facto* standard for Grid infrastructure. In addition, NMI [31] and the UK's Grid Core Program [7] are seeking to extend, standardize and make more robust key pieces of software for the Grid arsenal such as Globus [3] (Chapter 6), Condor [22] (Chapter 11), OGSA-DAI (Chapters 7 and 15) and the Network Weather Service [26]. In the last two decades, the development [16] of PVM [94] and MPI [95], which pre-dated the modern Grid vision, introduced parallel and distributed computing concepts to an entire community and provided the seeds for the community collaboration, which characterizes the Grid community today.

There are other important standards on which the Grid is being built. The last subsection stressed the key role of Linux as the standard for node operating systems [96]. Further within the commercial Web community, OASIS [97] is standardizing Web Services for Remote Portals (WSRP) – the portlet interface standard to define user-facing ports on Web services (Chapter 18). These standards support both commercial and noncommercial software and there is a growing trend in both arenas for open-source software. The Apache project [98] supplies key infrastructure such as servers [99] and tools to support such areas as WSDL-Java interfaces [100] and portals [101]. One expects these days that all software is either open source or provides open interfaces to proprietary implementations. Of course, the broad availability of modern languages like Java with good run-time and development environments has also greatly expedited the development of Grid and other software infrastructure.

Today, Grid projects seek to use common infrastructure and standards to promote interoperability and reusability, and to base their systems on a growing body of robust community software. Open source and standardization efforts are changing both the way software is written and the way systems are designed. This approach will be critical for the Grid as it evolves.

1.5 GRID APPLICATIONS AND APPLICATION MIDDLEWARE

The Grid will serve as an enabling technology for a broad set of applications in science, business, entertainment, health and other areas. However, the community faces a 'chicken and egg' problem common to the development of new technologies: applications are needed to drive the research and development of the new technologies, but applications are difficult to develop in the absence of stable and mature technologies. In the Grid community, Grid infrastructure efforts, application development efforts and middleware efforts have progressed together, often through the collaborations of multidisciplinary teams. In this section, we discuss some of the successful Grid application and application middleware efforts to date. As we continue to develop the software infrastructure that better realizes the potential of the Grid, and as common Grid infrastructure continues to evolve to provide a stable platform, the application and user community for the Grid will continue to expand.

1.5.1 Life science applications

One of the fastest-growing application areas in Grid Computing is the Life Sciences. Computational biology, bioinformatics, genomics, computational neuroscience and other areas are embracing Grid technology as a way to access, collect and mine data [e.g. the Protein Data Bank [102], the myGrid Project [103], the Biomedical Information Research Network (BIRN) [85]], accomplish large-scale simulation and analysis (e.g. MCell [104]), and to connect to remote instruments (e.g. in Telemicroscopy [105] and Chapter 33 [106]). The Biomedical Informatics Research Network links instruments and federated databases, illustrated in Figure 1.18. BIRN is a pioneering project that utilizes infrastructure to support cross-correlation studies of imaging and other data critical for neuroscience and biomedical advances.

The MCell collaboration between computational biologists and computer scientists to deploy large-scale Monte Carlo simulations using Grid technologies is a good example of a successful Grid-enabled life science application. Over the last decade, biologists have developed a community code called *MCell*, which is a general simulator for cellular microphysiology (the study of the physiological phenomena occurring at the microscopic level in living cells). MCell uses Monte Carlo diffusion and chemical reaction algorithms in 3D to simulate complex biochemical interactions of molecules inside and outside cells. MCell is one of the many scientific tools developed to assist in the quest to understand the form and function of cells, with specific focus on the nervous system. A local-area distributed MCell code is installed in laboratories around the world and is currently being used for several practical applications (e.g. the study of calcium dynamics in hepatocytes of the liver).

Grid technologies have enabled the deployment of large-scale MCell runs on a wide variety of target resources including clusters and supercomputers [107]. Computer scientists have worked with MCell biologists to develop the APST (AppLeS Parameter Sweep Template, described in Chapter 33) Grid middleware to efficiently deploy and schedule

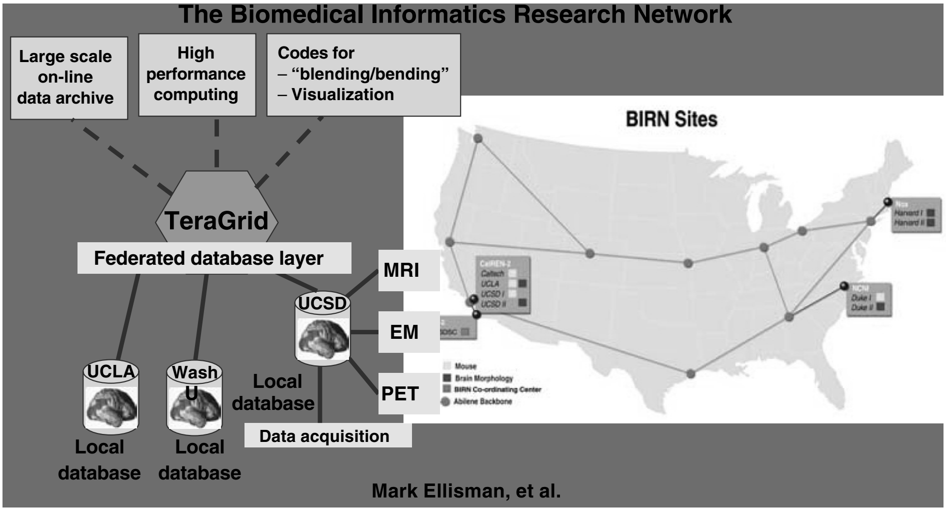

Figure 1.18 Biomedical Informatics Research Network – one of the most exciting new application models for the Grid.

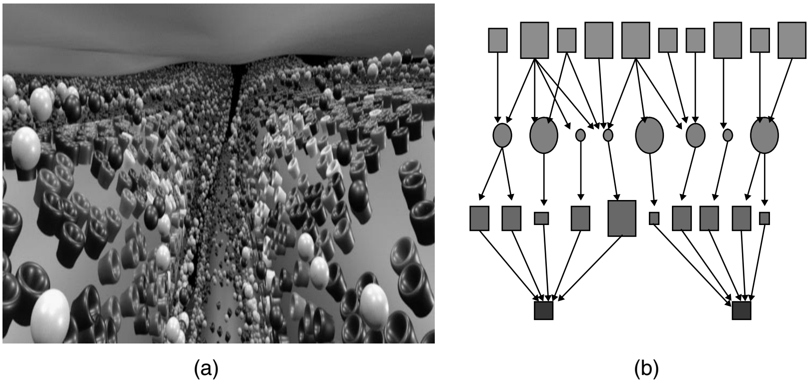

Figure 1.19 MCELL depiction of simulation of traversal of ligands in a cell (a) and program structure (b). On the right, we show linkage of shared input files, Monte Carlo " experiments" and shared output files.

large-scale runs in dynamic, distributed environments. APST has also been used by other distributed parameter sweep applications, forming part of the application-focused middleware layer of the Grid. Figure 1.19 shows MCell as seen by both disciplinary scientists and computer scientists. Figure 1.19a shows the traversal of ligands throughout the cell as simulated by MCell. Figure1.19b shows the program structure of MCell: the code comprises independent tasks that share common input files and output to common output files. Shared I/O requires data and output to be staged in order for the Grid to efficiently support application execution and this resulted in new computer science as well as computational science advances.

In the United Kingdom, the myGrid project [103] is a large consortium comprising the universities of Manchester, Southampton, Nottingham, Newcastle and Sheffield together with the European Bioinformatics Institute at Hinxton near Cambridge. In addition, GSK, AstraZeneca, IBM and SUN are industrial collaborators in the project. The goal of myGrid is to design, develop and demonstrate higher-level functionalities over the Grid to support scientists making use of complex distributed resources. The project is developing an e-Scientist's workbench providing support for the process of experimental investigation, evidence accumulation and result assimilation. A novel feature of the workbench will be provision for personalization facilities relating to resource selection, data management and process enactment. The myGrid design and development activity will be driven by applications in bioinformatics – one for the analysis of functional genomic data and another for supporting the annotation of a pattern database. The project intends to deliver Grid middleware services for automatic data annotation, workflow support and data access and integration. To support this last goal, the myGrid project will be a key application test case for the middleware being produced by the UK Core Programme project on OGSA – DAI [108].

1.5.2 Engineering-oriented applications

The Grid has provided an important platform for making resource-intensive engineering applications more cost-effective. One of the most comprehensive approaches to deploying production Grid infrastructure and developing large-scale engineering-oriented Grid applications is the NASA IPG [50] in the United States (Chapter 5). The NASA IPG vision provides a blueprint for revolutionizing the way in which NASA executes large-scale science and engineering problems via the development of

1. persistent Grid infrastructure supporting 'highly capable' computing and data management services that, on demand, will locate and co-schedule the multicenter resources needed to address large-scale and/or widely distributed problems,
2. ancillary services needed to support the workflow management frameworks that coordinate the processes of distributed science and engineering problems.

Figures 1.20 and 1.21 illustrate two applications of interest to NASA; in the first, we depict key aspects – airframe, wing, stabilizer, engine, landing gear and human factors – of the design of a complete aircraft. Each part could be the responsibility of a distinct, possibly geographically distributed, engineering team whose work is integrated together by a Grid realizing the concept of concurrent engineering. Figure 1.21 depicts possible Grid controlling satellites and the data streaming from them. Shown are a set of Web (OGSA) services for satellite control, data acquisition, analysis, visualization and linkage (assimilation) with simulations as well as two of the Web services broken up into multiple constituent services. Key standards for such a Grid are addressed by the new Space Link Extension international standard [109] in which part of the challenge is to merge a preGrid architecture with the still evolving Grid approach.

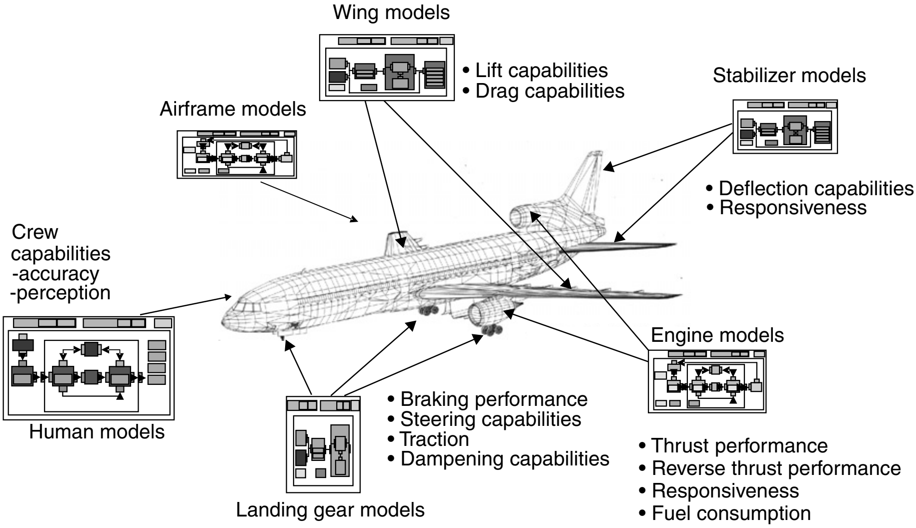

Figure 1.20 A Grid for aerospace engineering showing linkage of geographically separated subsystems needed by an aircraft.

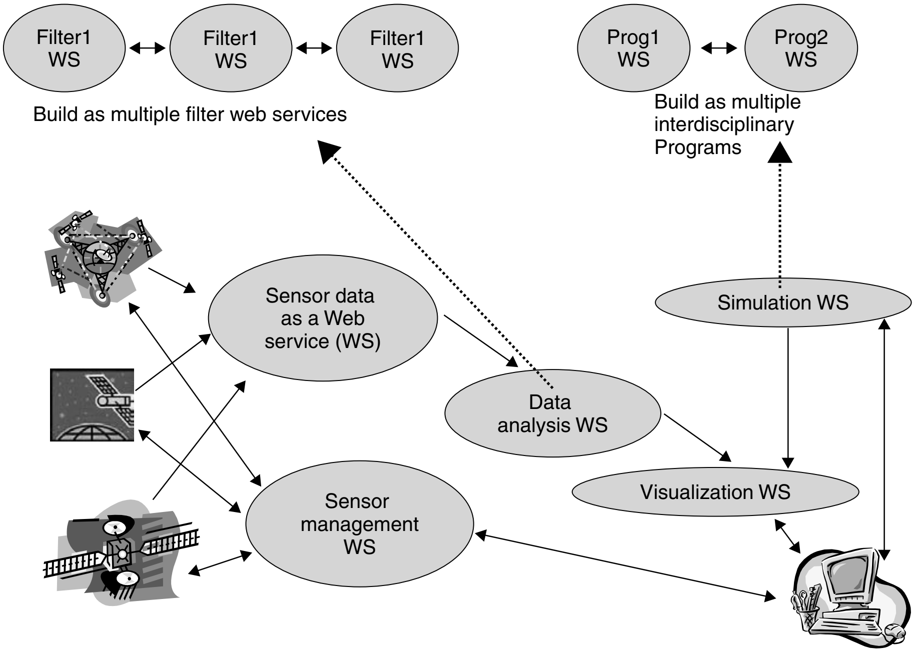

Figure 1.21 A possible Grid for satellite operation showing both spacecraft operation and data analysis. The system is built from Web services (WS) and we show how data analysis and simulation services are composed from smaller WS's.

In Europe, there are also interesting Grid Engineering applications being investigated. For example, the UK Grid Enabled Optimization and Design Search for Engineering (GEODISE) project [110] is looking at providing an engineering design knowledge repository for design in the aerospace area. Rolls Royce and BAESystems are industrial collaborators. Figure 1.22 shows this GEODISE engineering design Grid that will address, in particular the 'repeat engagement' challenge in which one wishes to build a semantic Grid (Chapter 17) to capture the knowledge of experienced designers. This of course is a research challenge and its success would open up many similar applications.

1.5.3 Data-oriented applications

As described in Chapter 36, data is emerging as the 'killer application' of the Grid. Over the next decade, data will come from everywhere – scientific instruments, experiments, sensors and sensornets, as well as a plethora of new devices. The Grid will be used to collect, store and analyze data and information, as well as to synthesize knowledge from data. Data-oriented applications described in Chapters 38 to 42 represent one of the most important application classes on the Grid and will be key to critical progress for both science and society. The importance of data for the Grid is also illustrated in several chapters: Chapters 7, 14 to 17 emphasize it in Part B of the book.

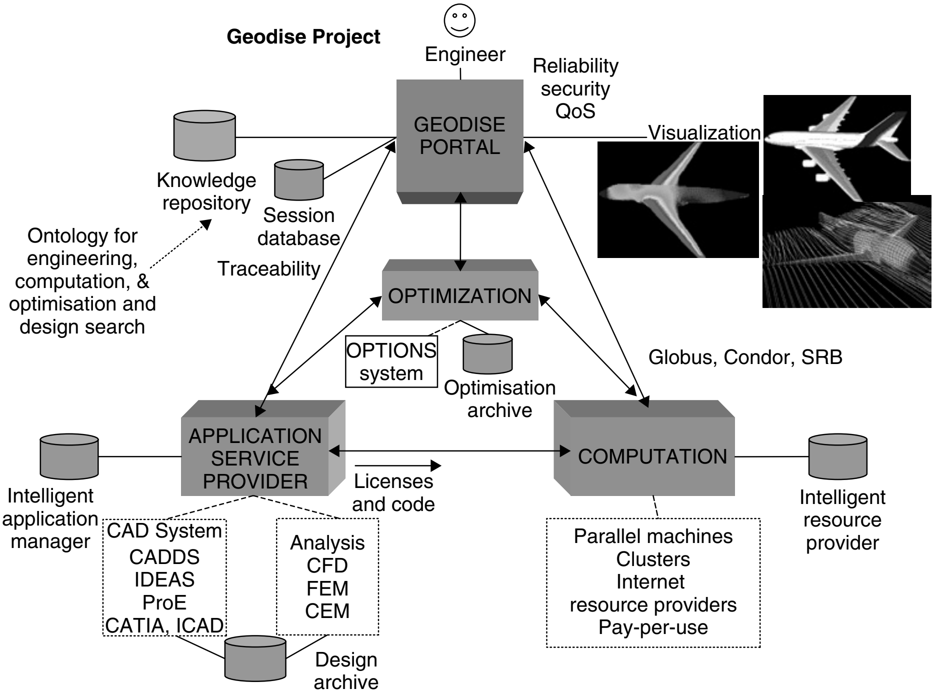

Figure 1.22 GEODISE aircraft engineering design Grid.

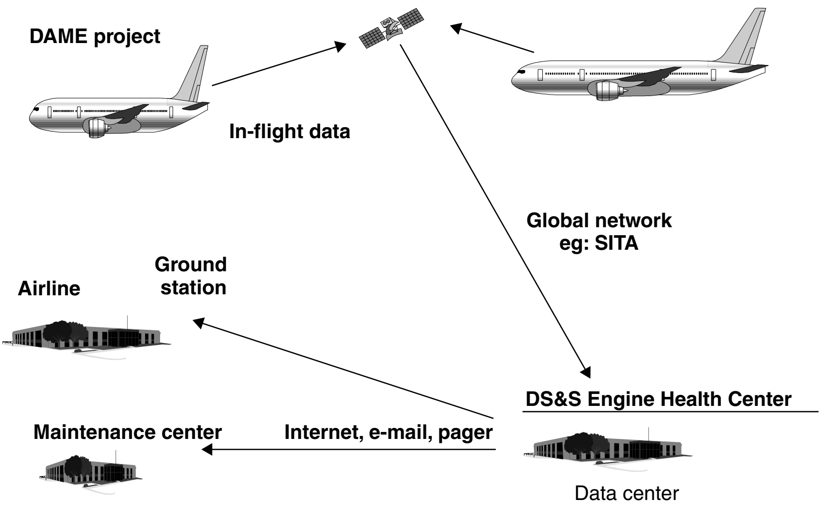

Figure 1.23 DAME Grid to manage data from aircraft engine sensors.

An example of a data-oriented application is Distributed Aircraft Maintenance Environment (DAME) [111], illustrated in Figure 1.23. DAME is an industrial application being developed in the United Kingdom in which Grid technology is used to handle the gigabytes of in-flight data gathered by operational aircraft engines and to integrate maintenance, manufacturer and analysis centers. The project aims to build a Grid-based distributed diagnostics system for aircraft engines and is motivated by the needs of Rolls Royce and its information system partner Data Systems and Solutions. The project will address performance issues such as large-scale data management with real-time demands. The main deliverables from the project will be a generic distributed diagnostics Grid application, an aero gas turbine application demonstrator for the maintenance of aircraft engines and techniques for distributed data mining and diagnostics. Distributed diagnostics is a generic problem that is fundamental in many fields such as medical, transport and manufacturing. DAME is an application currently being developed within the UK e-Science program.

1.5.4 Physical science applications

Physical science applications are another fast-growing class of Grid applications. Much has been written about the highly innovative and pioneering particle physics–dominated projects – the GriPhyN [77], Particle Physics Data Grid [78], and iVDGL [67] projects in the United States and the EU DataGrid [70], the UK GridPP [112] and the INFN (Italian National Institute for Research in Nuclear and Subnuclear Physics) Grid projects [57]. Figure 1.24 depicts the complex analysis of accelerator events being targeted to the Grid in these projects, which are described in more detail in Chapter 39. The pipelined structure of the solution allows the code to leverage the considerable potential of the Grid: In this case, the CERN linear accelerator will provide a deluge of data (perhaps 10 Gb s^{-1} of the

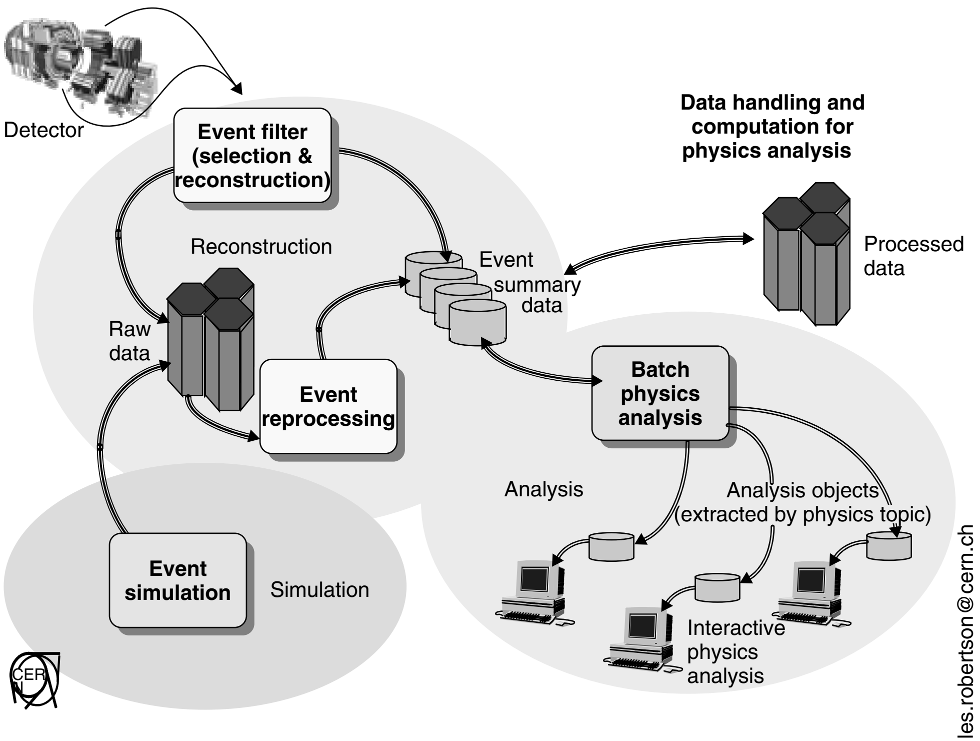
Detector
Event filter (selection & reconstruction)
Data handling and computation for physics analysis
Reconstruction
Event summary data
Processed data
Raw data
Event reprocessing
Batch physics analysis
Analysis
Analysis objects (extracted by physics topic)
Event simulation
Simulation
Interactive physics analysis
CERN
les.robertson@cern.ch

Figure 1.24 Architecture of particle physics analysis Grid (Chapter 39).

100 Gb s^{-1} GTRN network) while each physics event can be processed independently, resulting in trillion-way parallelism.

The astronomy community has also targeted the Grid as a means of successfully collecting, sharing and mining critical data about the universe. For example, the National Virtual Observatory Project in the United States [79] is using the Grid to federate sky surveys from several different telescopes, as discussed in Chapter 38. Using Grid technology, the sky surveys sort, index, store, filter, analyze and mine the data for important information about the night sky. High-performance Grids, such as TeraGrid, will enable NVO researchers to shorten the process of collecting, storing and analyzing sky survey data from 60 days to 5 days and will enable researchers to federate multiple sky surveys. Figures 1.25 and 1.26 show the NVO [113] and the potential impact of TeraGrid capabilities on this process.

In Europe, the EU AVO project and the UK AstroGrid project complement the US NVO effort in astronomy. The three projects are working together to agree to a common set of standards for the integration of astronomical data. Together, the NVO, AVO and AstroGrid efforts will provide scientists and educators with an unprecedented amount of accessible information about the heavens. Note that whereas in many other communities, data ownership is often an issue, astronomy data will be placed in the public domain in

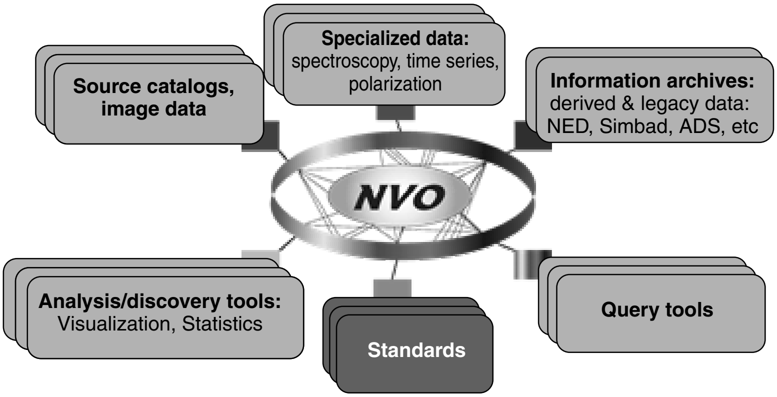
Specialized data:
spectroscopy, time series, polarization
Source catalogs, image data
Information archives:
derived & legacy data: NED, Simbad, ADS, etc
NVO
Analysis/discovery tools:
Visualization, Statistics
Standards
Query tools

Figure 1.25 Architecture for the national virtual observatory.

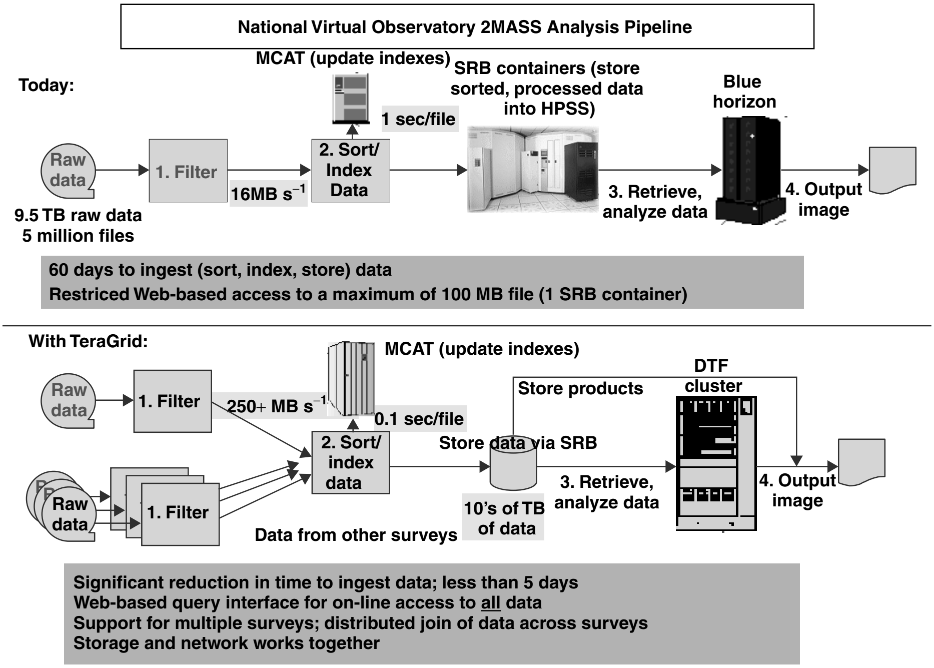
National Virtual Observatory 2MASS Analysis Pipeline
Today:
MCAT (update indexes)
SRB containers (store sorted, processed data into HPSS)
Blue horizon
1 sec/file
Raw data
1. Filter
16MB s^{-1}
2. Sort/ Index Data
3. Retrieve, analyze data
4. Output image
9.5 TB raw data
5 million files
60 days to ingest (sort, index, store) data
Restriced Web-based access to a maximum of 100 MB file (1 SRB container)
With TeraGrid:
MCAT (update indexes)
DTF cluster
Raw data
1. Filter
250+ MB s^{-1}
0.1 sec/file
Store products
Store data via SRB
2. Sort/ index data
3. Retrieve, analyze data
4. Output image
Raw data
1. Filter
10's of TB of data
Data from other surveys
Significant reduction in time to ingest data; less than 5 days
Web-based query interface for on-line access to all data
Support for multiple surveys; distributed join of data across surveys
Storage and network works together

Figure 1.26 e-Science for the 2MASS astronomical data analyzed by the TeraGrid.

these Virtual Observatories after a set period of time in which the astronomers who took the original data have exclusive use of it. This points the way to a true 'democratization' of science and the emergence of a new mode of 'collection-based' research to be set alongside the traditional experimental, theoretical and computational modes.

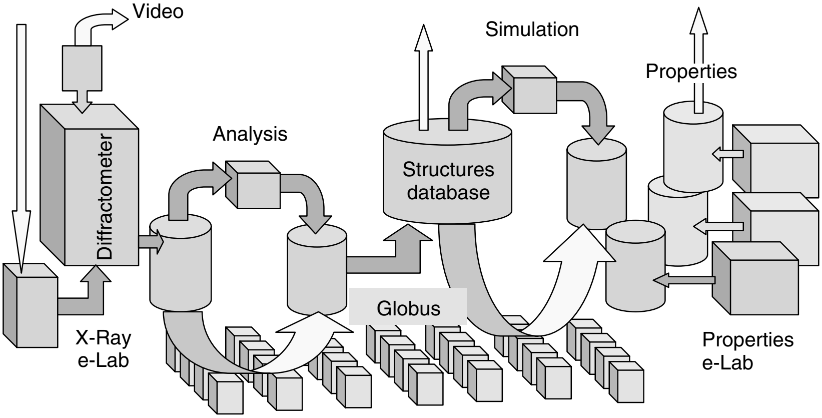

Figure 1.27 A combinatorial chemistry Grid (Chapter 42).

Finally, Figure 1.27 captures the combinatorial chemistry application of Chapter 42; experiments in this field create their deluge by parallel execution. Here we see 'experiment-on demand' with a smart laboratory (e-Lab) running miniGrid software and performing needed experiments in real time to fill in knowledge holes.

1.5.5 Trends in research: e-Science in a collaboratory

The interesting e-Science concept illustrates changes that information technology is bringing to the methodology of scientific research [114]. e-Science is a relatively new term that has become particularly popular after the launch of the major United Kingdom initiative described in Section 1.4.3. e-Science captures the new approach to science involving distributed global collaborations enabled by the Internet and using very large data collections, terascale computing resources and high-performance visualizations. e-Science is about global collaboration in key areas of science, and the next generation of infrastructure, namely the Grid, that will enable it. Figure 1.28 summarizes the e-Scientific method. Simplistically, we can characterize the last decade as focusing on simulation and its integration with science and engineering – this is computational science. e-Science builds on this adding data from all sources with the needed information technology to analyze and assimilate the data into the simulations.

Over the last half century, scientific practice has evolved to reflect the growing power of communication and the importance of collective wisdom in scientific discovery. Originally scientists collaborated by sailing ships and carrier pigeons. Now aircraft, phone, e-mail and the Web have greatly enhanced communication and hence the quality and real-time nature of scientific collaboration. The collaboration can be both 'real' or enabled electronically – as evidenced by Bill Wulf [115, 116] early influential work on the scientific collaboratory.

e-Science and hence the Grid is the infrastructure that enables collaborative science. The Grid can provide the basic building blocks to support real-time distance interaction, which has been exploited in distance education as described in Chapter 43. Particularly

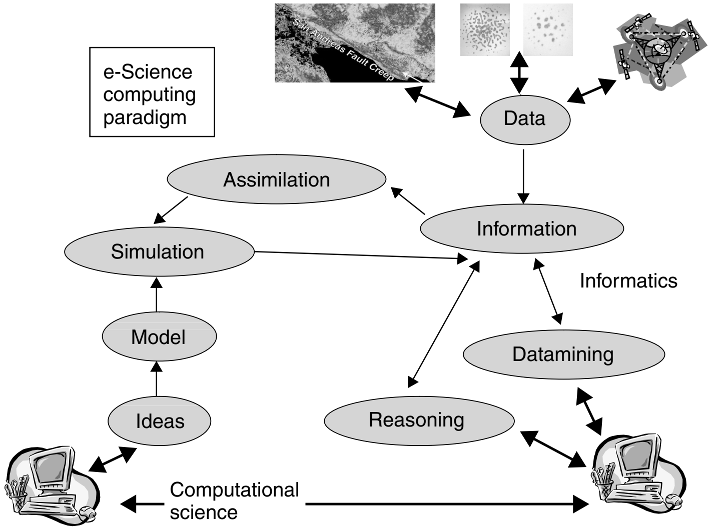

Figure 1.28 Computational science and information technology merge in e-Science.

important is the infrastructure to support shared resources – this includes many key services including security, scheduling and management, registration and search services (Chapter 19) and the message-based interfaces of Web services (Chapter 18) to allow powerful sharing (collaboration) mechanisms. All of the basic Grid services and infrastructure provide a critical venue for collaboration and will be highly important to the community.

1.5.6 Commercial Applications

In the commercial world, Grid, Web and distributed computing, and information concepts are being used in an innovative way in a wide variety of areas including inventory control, enterprise computing, games and so on. The Butterfly Grid [117] and the Everquest multiplayer gaming environment [118] are current examples of gaming systems using Grid-like environments. The success of SETI@home [36], a highly distributed data-mining application with the goal of identifying patterns of extraterrestrial intelligence from the massive amounts of data received by the Arecibo radio telescope, has inspired both innovative research and a cadre of companies to develop P2P technologies. Chapter 12 describes the Entropia system, one of the intellectual leaders in this area of P2P or Megacomputing. Another interesting application of this type, climateprediction.com [119], is being developed by the UK e-Science program. This will implement the ensemble (multiple initial conditions and dynamical assumptions) method for climate prediction on a megacomputer.

Enterprise computing areas where the Grid approach can be applied include [10]

- end-to-end automation,
- end-to-end security,

- virtual server hosting,
- disaster recovery,
- heterogeneous workload management,
- end-to-end systems management,
- scalable clustering,
- accessing the infrastructure,
- ‘utility’ computing,
- accessing new capability more quickly,
- better performance,
- reducing up-front investment,
- gaining expertise not available internally, and
- Web-based access (portal) for control (programming) of enterprise function.

Chapter 43 describes issues that arise in incorporating Web services into enterprise computing. In addition to these enterprise applications, the concept of ‘e-Utility’ has emerged to summarize ‘X-on-demand’: computing-on-demand, storage-on-demand, networking-on-demand, information-on-demand and so on. This generalizes the familiar concept of Application Service Providers (ASPs). Some clear examples today of computing-on-demand come from systems like Condor and Entropia (Chapters 11 and 12). The use of Grid technologies to support e-Utility can be merged with those of autonomic computing (Chapter 13 and Section 1.6.1) in a new generation of commercial systems. Other interesting commercial Grid activities include the Sun Grid Engine [32] and Platform Computing [34] implementing resource management and scheduling opportunities similar to those addressed by Condor in Chapter 11.

The growing partnership between the commercial sector and the academic community in the design and development of Grid technologies is likely to bear fruit in two important ways: as a vehicle for a new generation of scientific advances and as a vehicle for a new generation of successful commercial products.

1.5.7 Application Summary

Applications are key to the Grid, and the examples given above show that at this stage we have some clear successes and a general picture as to what works today. A major purpose of the broader Grid deployment activities described in the Section 1.4.3 is to encourage further application development. Ultimately, one would hope that the Grid will be the operating system of the Internet and will be viewed in this fashion. Today we must strive to improve the Grid software to make it possible that more than the ‘marine corps’ of application developers can use the Grid. We can identify three broad classes of applications that today are ‘natural for Grids’ [120].

- *Minimal communication applications*: These include the so-called ‘embarrassingly parallel’ applications in which one divides a problem into many essentially independent pieces. The successes of Entropia (Chapter 12), SETI@Home and other megacomputing projects are largely from this category.

- *Staged/linked applications (do Part A then do Part B)*: These include remote instrument applications in which one gets input from the instrument at Site A, compute/analyze data at Site B and visualizes at Site C. We can coordinate resources including computers, data archives, visualization and multiple remote instruments.
- *Access to resources (get something from/do something at Site A)*: This includes portals, access mechanisms and environments described in Part C of the book.

Chapters 35 to 42 describe many of the early successes as do several of the chapters in Part C that describe Grid environments used to develop problem-solving environments and portals. One influential project was the numerical relativity simulations of colliding black holes where Grids have provided the largest simulations. The Cactus Grid software was developed for this (Chapter 23) and an early prototype is described in Chapter 37. An interesting example is Synthetic Forces (SF) Express [121], which can be considered as an example of Grid technology applied to military simulations. This large-scale distributed interactive battle simulation decomposed terrain (Saudi Arabia, Kuwait, Iraq) contiguously among supercomputers and performed a simulation of 100 000 vehicles in early 1998 with vehicle (tanks, trucks and planes) location and state updated several times a second. Note that the military simulation community has developed its own sophisticated distributed object technology High-Level Architecture (HLA) [122] and the next step should involve integrating this with the more pervasive Grid architecture.

Next-generation Grid applications will include the following:

- *Adaptive applications (run where you can find resources satisfying criteria X),*
- *Real-time and on-demand applications (do something right now),*
- *Coordinated applications (dynamic programming, branch and bound)* and
- *Poly-applications (choice of resources for different components).*

Note that we still cannot ‘throw any application at the Grid’ and have resource management software determine where and how it will run.

There are many more Grid applications that are being developed or are possible. Major areas of current emphasis are health and medicine (brain atlas, medical imaging [123] as in Chapter 41, telemedicine, molecular informatics), engineering, particle physics (Chapter 39), astronomy (Chapter 38), chemistry and materials (Chapter 42 and [124]), environmental science (with megacomputing in [119]), biosciences and genomics (see Chapter 40 and [125, 126]), education (Chapter 43) and finally digital libraries (see Chapter 36).

Grid applications will affect everybody – scientists, consumers, educators and the general public. They will require a software environment that will support unprecedented diversity, globalization, integration, scale and use. This is both the challenge and the promise of the Grid.

1.6 FUTURES – GRIDS ON THE HORIZON

In many ways, the research, development and deployment of large-scale Grids are just beginning. Both the major application drivers and Grid technology itself will greatly change over the next decade. The future will expand existing technologies and integrate new technologies. In the future, more resources will be linked by more and better networks. At the end of the decade, sensors, PDAs, health monitors and other devices will be linked to the Grid. Petabyte data resources and petaflop computational resources will join low-level sensors and sensornets to constitute Grids of unprecedented heterogeneity and performance variation. Over the next decade, Grid software will become more sophisticated, supporting unprecedented diversity, scale, globalization and adaptation. Applications will use Grids in sophisticated ways, adapting to dynamic configurations of resources and performance variations to achieve goals of Autonomic computing.

Accomplishing these technical and disciplinary achievements will require an immense research, development and deployment effort from the community. Technical requirements will need to be supported by the human drivers for Grid research, development and education. Resources must be made available to design, build and maintain Grids that are of high capacity (rich in resources), of high capability (rich in options), persistent (promoting stable infrastructure and a knowledgeable workforce), evolutionary (able to adapt to new technologies and uses), usable (accessible, robust and easy-to-use), scalable (growth must be a part of the design), and able to support/promote new applications.

Today, many groups are looking beyond the challenges of developing today's Grids to the research and development challenges of the future. In this section, we describe some key areas that will provide the building blocks for the Grids of tomorrow.

1.6.1 Adaptative and autonomic computing

The Grid infrastructure and paradigm is often compared with the Electric Power Grid [127]. On the surface, the analogy holds up – the Grid provides a way to seamlessly virtualize resources so that they can provide access to effectively infinite computing cycles and data storage for the user who 'plugs in' to the Grid. The infrastructure managing which machines, networks, storage and other resources are used is largely hidden from the user in the same way as individuals generally do not know which power company, transformer, generator and so on are being used when they plug their electric appliance into a socket.

The analogy falls short when it comes to performance. Power is either there or not there. To the first order, the location of the plug should not make electrical devices plugged into it run better. However, on the Grid, the choice of the machine, the network and other component impacts greatly the performance of the program. This variation in performance can be leveraged by systems that allow programs to adapt to the dynamic performance that can be delivered by Grid resources. Adaptive computing is an important area of Grid middleware that will require considerable research over the next decade.

Early work in the academic community (e.g. the AppLeS project on adaptive scheduling [23], the GrADS project on adaptive program development and execution environments [69], Adaptive Middleware projects [128], the SRB [27], Condor [22] and others) have provided fundamental building blocks, but there is an immense amount of work that remains to be done. Current efforts in the commercial sector by IBM on 'Autonomic Computing', as discussed in Chapter 13, provide an exciting current focus likely to have a strong impact on the Grid. Through 'Project Eliza', IBM is exploring the concepts of software which is self-optimizing, self-configuring, self-healing and self-protecting to ensure that software systems are flexible and can adapt to change [129]. Moore's law [8] has of course a profound impact on computing. It describes the technology improvement that governs increasing CPU performance, memory size and disk storage. Further, it also underlies the improvement in sensor technology that drives the data deluge underlying much of e-Science. However, technology progress may provide increased capability at the cost of increased complexity. There are orders of magnitude more servers, sensors and clients to worry about. Such issues are explored in depth in Chapter 13 and we expect this to be an important aspect of Grid developments in the future. Both the nodes of the Grid and their organization must be made robust – internally fault-tolerant, as well as resilient to changes and errors in their environment. Ultimately, the Grid will need self-optimizing, self-configuring, self-healing and self-protecting components with a flexible architecture that can adapt to change.

1.6.2 Grid programming environments

A curious observation about computational environments is that as the environment becomes more complex, fewer robust and usable tools seem to be available for managing the complexity and achieving program performance. The 1980s saw more maturity in the development of parallel architecture models than effective parallel software, and currently, efforts to develop viable programming environments for the Grid are limited to just a few forward-looking groups.

In order for the Grid to be fully usable and useful, this state of affairs will need to change. It will be critical for developers and users to be able to debug programs on the Grid, monitor the performance levels of their programs on Grid resources and ensure that the appropriate libraries and environments are available on deployed resources. Part C of the book, which discusses about Grid computing environments, and is summarized in Chapter 20, describes this area. To achieve the full vision of the Grid, we will need compilers that can interact with resource discovery and resource selection systems to best target their programs and run-time environments that allow the migration of programs during execution to take advantage of more optimal resources. Robust, useful and usable programming environments will require coordinated research in many areas as well as test beds to test program development and run-time ideas. The GrADS project [69] provides a first example of an integrated approach to the design, development and prototyping of a Grid programming environment.

A key part of the user experience in computational environments is the way in which the user interacts with the system. There has been considerable progress in the important area of portals but the increasing complexity of Grid resources and the sophisticated manner in which applications will use the Grid will mandate new ways to access the Grid. 'Programming the Grid' really consists of two activities: preparation of the individual application nuggets associated with a single resource and integrating the nuggets to form a complete Grid program. An application nugget can be many things – the Structured Query Language (SQL) interface to a database, a parallel image processing algorithm and a finite element solver. Integrating nuggets to form a complete system may involve the dynamic integration of all the Grid and Portal system services.

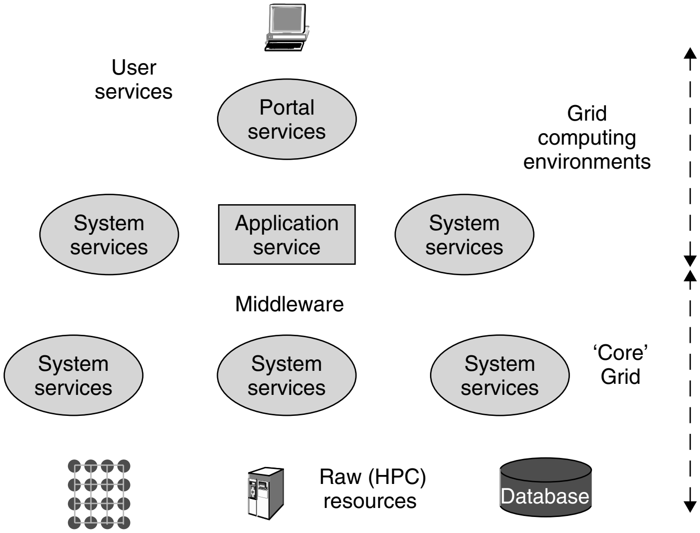

Figure 1.29 Grids, portals, and Grid computing environments.

An important area of research will target the development of appropriate models for interaction between users and applications and the Grid. Figure 1.29 illustrates the interrelation of Grid components involved in developing portals and Grid Computing Environments (GCEs): The horizontal direction corresponds to application and/or resource functionality (parallel simulation, sensor data gather, optimization, database etc.). The vertical direction corresponds to system functionality from scheduling to composition to portal rendering. Note that system state is determined by its environment, by user request and by the running application in some dynamic fashion. Currently, there is no 'consensus complete model' from user to resource and correspondingly no clear distinction between GCE and 'core' Grid capabilities (shown at the top and bottom of Figure 1.29, respectively). The matrix of capabilities sketched in Figure 1.29 and elaborated in Chapter 20 is very rich and we can expect different approaches to have value for different applications.

1.6.3 New Technologies

At the beginning of the twenty-first century, we are witnessing an immense explosion in telecommunications. The ubiquitous cell phones and PDAs of today are just the beginning of a deeper paradigm shift predicated upon the increasing availability of comprehensive information about the world around us.

Over the next decade, it will become increasingly important to application developers to integrate new devices and new information sources with the Grid. Sensors and sensornets embedded in bridges, roads, clothing and so on will provide an immense source of data. Real-time analysis of information will play an even greater role in health, safety, economic stability and other societal challenges. The integration of new devices will provide software and application challenges for the Grid community but will create a whole new level of potential for scientific advances.

1.6.4 Grid Policy and Grid Economies

Large-scale entities, such as the Science Collaboratories of Section 1.5.5, require organization in order to accomplish their goals. Complex systems from the Internet to the human cardiovascular system are organized hierarchically to manage/coordinate the interaction of entities via organizational structures that ensure system stability. The Grid will similarly require policies, organizational structure and an economy in order to maintain stability and promote individual and group performance. An important activity over the next decade will be the research, development and testing required to identify useful Grid policies, economies and ‘social structures’, which ensure the stability and efficiency of the Grid.

The Grid provides an interesting venue for policy. Grid resources may lie in different administrative domains and are governed by different local and national policies; however, the process of building and using the Grid is predicated on shared resources, agreement and coordination. Global collaboration heightens the need for community and culturally sensitive trust, policy, negotiation and payment services.

Most important, the Grid provides an exercise in cooperation: resource usage and administration must bridge technological, political and social boundaries, and Grid policies will need to provide an incentive to the individual (users and applications) to contribute to the success (stability) of the group.

1.6.5 A final note

The Grid vision is absolutely critical to future advances of science and society, but vision alone will not build the Grid. The promise and potential of the Grid must drive agendas for research, development and deployment over the next decade. In this book, we have asked a community of researchers, Grid developers, commercial partners and professionals to describe the current state of Grid middleware and their vision for the efforts that must drive future agendas. Building the Grid is one of the most challenging and exciting efforts in the science and technology community today, and more so because it must be done cooperatively and as a community effort. We hope that this book provides you, the reader, an insider’s view of the challenges and issues involved in building the Grid and a sense of excitement about its potential and promise.

Box 1.2 Summary of Parts A and B of book (Chapters 1 to 19)

The initial chapter gives an overview of the whole book. Chapter 20 summarizes Part C and Chapter 35 summarizes Part D of the book. Here we summarize Parts A and B. Part A of this book, Chapters 1 to 5, provides an overview and motivation for Grids. Further, Chapter 37 is an illuminating discussion on Metacomputing from 1992–a key early concept on which much of the Grid has been built. Chapter 2 is a short overview of the Grid reprinted from Physics Today. Chapter 3 gives a detailed recent history of the Grid, while Chapter 4 describes the software environment of the seminal I-WAY experiment at SC95. As discussed in the main text of Section 1.2, this conference project challenged participants – including, for instance, the Legion activity of Chapter 10–to demonstrate Gridlike applications on an OC-3 backbone. Globus [3] grew out of the software needed to support these 60 applications at 17 sites; the human intensive scheduling and security used by I-WAY showed the way to today's powerful approaches. Many of these applications employed visualization including Computers and Advanced Visualization Environments (CAVE) virtual reality stations as demonstrated in the early Metacomputing work of Chapter 37. Chapter 5 brings us to 2002 and describes the experience building Globus-based Grids for NASA and DoE.

Turning to Part B of the book, Chapters 6 to 9 provide an overview of the community Grid approach in which the components of the Globus toolkit are being reimplemented as generalized OGSA Web services. Chapter 11 also fits into this thrust as Condor can operate as a stand-alone Grid but can also be thought of as providing workload and scheduling services for a general (Globus) Grid. Chapter 10 describes the Legion and Avaki approach, which pioneered the object model for Grids and provides an end-to-end solution that is compatible with the architecture of Figure 1.3. We will need to see how Globus, Condor and Avaki look after reformulation as Web services and if interoperability is possible and useful. All these systems should support the autonomic principles defined and described in Chapter 13.

This book illustrates Industry interest with Chapters 8, 10, 12, and 13 highlighting Avaki, Entropia and IBM; other key participation from Sun Microsystems (from the Sun Grid engine scheduling system [32] to the JXTA peer-to-peer network) [33]) and Platform Computing [34] are discussed in Chapters 3 and 18.

Chapter 12 on Entropia illustrates *megacomputing* or the harnessing of unused time on Internet clients [35]. Entropia can be thought of as a specialized Grid supplying needed management and fault tolerance for a megacomputing Grid with disparate unmanaged Internet or more structured enterprise nodes providing computing-on demand. Although early efforts of this type were part of I-WAY (see Fafner in Chapter 3), these ideas were developed most intensively in projects like SETI@home [36], which uses millions of Internet clients to analyze data looking for extraterrestrial life and for the newer project examining the folding of proteins [37]. These are building distributed computing solutions for applications, which can be divided into a huge number of essentially independent computations, and a central server system doles out separate work chunks to each participating client. In the parallel computing community, these problems are called 'pleasingly or embarrassingly parallel'. Other projects of this type include United Devices [38] and Parabon computation [39]. As explained in Chapter 12, other applications for this type of system include financial modeling, bioinformatics, Web performance and the scheduling of different jobs to use idle time on a network of workstations. Here the work links to Condor of Chapter 11, which focuses on more managed environments.

Chapters 14 to 17 address critical but different features of the data Grid supporting both the deluge from sensors and the more structured database and XML metadata resources. This is imperative if the Grid is to be used for what appear as the most promising applications. These chapters cover both the lower-level integration of data into the Grid fabric and the critical idea of a Semantic Grid – can knowledge be created in an emergent fashion by the linking of metadata enriched but not intrinsically intelligent Grid components.

Peer-to-peer (P2P) networks are an example of an approach to Gridlike systems which although crude today appears to offer both the scaling and autonomic self-sufficiency needed for the next generation Grid systems. Chapter 18 explains how to integrate P2P and Grid architectures, while Chapter 19 uses discovery of Web services to illustrate how P2P technology can provide the federation of disparate dynamic data resources.

REFERENCES

1. Foster, I. and Kesselman, C. (eds) (1999) *The Grid: Blueprint for a New Computing Infrastructure*. San Francisco, CA: Morgan Kaufmann.
2. The Global Grid Forum Web Site, http://www.gridforum.org.
3. The Globus Project Web Site, http://www.globus.org.
4. Berman, F., Fox, G. and Hey, T. (2003) *Grid Computing: Making the Global Infrastructure a Reality*. Chichester: John Wiley & Sons.
5. Web Site associated with book, *Grid Computing: Making the Global Infrastructure a Reality*, http://www.grid2002.org.
6. von Laszewski, G., Su, M.-H., Foster, I. and Kesselman, C. (2002) Chapter 8, in Dongarra, J., Foster, I., Fox, G., Gropp, W., Kennedy, K., Torczon, L., and White, A. (eds) *The Sourcebook of Parallel Computing*, ISBN 1-55860-871-0, San Francisco: Morgan Kaufmann Publishers.
7. Taylor, J. M. and e-Science, http://www.e-science.clrc.ac.uk and http://www.escience-grid.org.uk/.
8. Moore's Law as Explained by Intel, http://www.intel.com/research/silicon/mooreslaw.htm.
9. Gilder, G. (ed.) (2002) Gilder's law on network performance. *Telecosm: The World After Bandwidth Abundance*. ISBN: 0743205472, Touchstone Books.
10. Wladawsky-Berger, I. (Kennedy Consulting Summit) (2001) November 29, 2001, http://www-1.ibm.com/servers/events/pdf/transcript.pdf and http://www-1.ibm.com/servers/events/gridcomputing.pdf.
11. Berman, F. Presentations 2001 and 2002, http://share.sdsc.edu/dscgi/ds.py/View/Collection-551.
12. US/UK Grid Workshop, San Francisco, August 4, 5, 2001, http://www.isi.edu/us-uk.gridworkshop/presentations.html.
13. McRobbie, M. and Wallace, S. (2002) Spring 2002, Arlington, VA, Internet2 Member Meeting, May 6–8, 2002, http://www.internet2.edu/activities/html/spring_02.html and http://www.indiana.edu/~ovpit/presentations/.
14. Global Terabit Research Network, http://www.gtrn.net.
15. Abilene Network and Control Center, http://www.abilene.iu.edu/.
16. Dongarra, J., Foster, I., Fox, G., Gropp, W., Kennedy, K., Torczon, L. and White, A. (eds) (2002) *The Sourcebook of Parallel Computing*, ISBN 1-55860-871-0, San Francisco: Morgan Kaufmann Publishers.
17. NSF Grand Challenge as part of 1997 HPCC Implementation Plan http://www.itrd.gov/pubs/imp97/; their 2002 view at

http://www.itrd.gov/pubs/blue02/national-grand.html, National Grand Challenge Applications, National Coordination Office for Information Technology Research and Development.

18. CNRI, Corporation for National Research Initiatives, Gigabit Testbed Initiative Final Report, December, 1996, http://www1.cnri.reston.va.us/gigafr/.
19. Grimshaw, A. S. and Wulf, W. A. (1997) The legion vision of a worldwide virtual computer. *Communications of the ACM*, **40**(1), 39–45.
20. Grimshaw, A. S., Ferrari, A. J., Lindahl, G. and Holcomb, K. (1998) Metasystems. *Communications of the ACM*, **41**(11), 46–55.
21. Legion Worldwide Virtual Computer Home Page, http://legion.virginia.edu/index.html.
22. The Condor Project, http://www.cs.wisc.edu/condor/.
23. Berman, F., Wolski, R., Figueira, S., Schopf, J. and Shao, G. (1996) Application level scheduling on distributed heterogeneous networks. Proceedings of Supercomputing '96, 1996.
24. Gehrinf, J. and Reinfeld, A. (1996) Mars – a framework for minimizing the job execution time in a metacomputing environment. Proceedings of Future general Computer Systems, 1996.
25. Weissman, J. B. (1999) Prophet: automated scheduling of SPMD programs in workstation networks. *Concurrency: Practice and Experience*, **11**(6), 301–321.
26. The Network Weather Service, http://nws.cs.ucsb.edu/.
27. SDSC Storage Resource Broker, http://www.npaci.edu/DICE/SRB/.
28. Netsolve RPC Based Networked Computing, http://icl.cs.utk.edu/netsolve/.
29. Ninf Global Computing Infrastructure, http://ninf.apgrid.org/.
30. Internet Pioneer *J.C.R. Licklider,* http://www.ibiblio.org/pioneers/licklider.
31. NSF Middleware Initiative, http://www.nsf-middleware.org/.
32. Sun Grid Engine, http://wwws.sun.com/software/gridware/.
33. JXTA Peer-to-Peer Technology, http://www.jxta.org.
34. Platform Grid Computing, http://www.platform.com/grid/index.asp.
35. Foster, I. (2000) Internet computing and the emerging grid. *Nature*, **7**, http://www.nature.com/nature/webmatters/grid/grid.html.
36. SETI@Home Internet Computing, http://setiathome.ssl.berkeley.edu/.
37. Folding@home Internet Protein Structure, http://www.stanford.edu/group/pandegroup/Cosm/.
38. United Devices Internet Computing, http://www.ud.com/home.htm.
39. Parabon Java Computing, http://www.parabon.com.
40. superJANET4 United Kingdom Network, http://www.superjanet4.net/.
41. GEANT European Network by DANTE, http://www.dante.net/geant/.
42. CANARIE Advanced Canadian Networks, http://www.canarie.ca/advnet/advnet.html.
43. Asia-Pacific Advanced Network Consortium (APAN), http://www.apan.net/.
44. McRobbie, M. A. (Indiana University & Internet2), Wallace, S. (Indiana University & Internet2), van Houweling, D. (Internet2), Boyles, H. (Internet2), Liello, F. (European NREN Consortium), and Davies, D. (DANTE), A Global Terabit Research Network, http://www.gtrn.net/global.pdf.
45. California Institute for Telecommunications and Information Technology, http://www.calit2.net/.
46. Top 500 Supercomputers, http://www.top500.org.
47. Earth Simulator in Japan, http://www.es.jamstec.go.jp/esc/eng/index.html.
48. Fujitsu PRIMEPOWER HPC2500, http://pr.fujitsu.com/en/news/2002/08/22.html.
49. ASCI DoE Advanced Simulation and Computing Program, http://www.lanl.gov/asci/, http://www.llnl.gov/asci/, and http://www.sandia.gov/ASCI/.
50. NASA Information Power Grid, http://www.ipg.nasa.gov/.
51. DoE Department of Energy Science Grid, http://www.doesciencegrid.org.
52. TeraGrid Project, http://www.teragrid.org/.
53. NSF Advisory Committee on Cyberinfrastructure, http://www.cise.nsf.gov/b_ribbon/.
54. Asian Grid Center, http://www.apgrid.org/.
55. DutchGrid: Distributed Computing in the Netherlands, http://www.dutchgrid.nl/.
56. INRIA French Grid, http://www-sop.inria.fr/aci/grid/public/acigrid.html.

57. INFN and CNR Grids in Italy, http://www.ercim.org/publication/Ercim_News/enw45/codenotti.html.
58. CosmoGrid – National Computational Grid for Ireland, http://www.grid-ireland.org/.
59. PIONIER: Polish Optical Internet – Advanced Applications, Services and Technologies for Information Society, http://www.kbn.gov.pl/en/pionier/.
60. NorduGrid Scandinavian Grid, http://www.nordugrid.org/.
61. Transatlantic Grid, http://datatag.web.cern.ch/datatag/.
62. GRIP Unicore (Chapter 29 and http://www.unicore.org/links.htm), Globus Interoperability Project, http://www.grid-interoperability.org/.
63. Rheinheimer, R., Humphries, S. L., Bivens, H. P. and Beiriger, J. I. (2002) The ASCI computational grid: initial deployment. *Concurrency and Computation: Practice and Experience* **14**, Grid Computing Environments Special Issue 13–14.
64. National United Kingdom e-Science Center, http://umbriel.dcs.gla.ac.uk/NeSC/.
65. UK Grid Support Center, http://www.grid-support.ac.uk/.
66. GRIDS Grid Research Integration Deployment and Support Center, http://www.grids-center.org/.
67. International Virtual Data Grid Laboratory, http://www.ivdgl.org/.
68. iVDGL Grid Operations Center, http://igoc.iu.edu/.
69. GrADS Grid Application Development Software Project, http://nhse2.cs.rice.edu/grads/.
70. European DataGrid at CERN Accelerator Center, http://eu-datagrid.web.cern.ch/eu-datagrid/.
71. EUROGRID Grid Infrastructure, http://www.eurogrid.org/.
72. European Grid Application Toolkit and Testbed, http://www.gridlab.org/.
73. European Cross Grid Infrastructure Project, http://www.crossgrid.org/.
74. DAMIEN Metacomputing Project on Distributed Applications and Middleware for Industrial use of European Networks, http://www.hlrs.de/organization/pds/projects/damien/.
75. Grid Resource Broker Project, http://sara.unile.it/grb/grb.html.
76. DoE National Magnetic Fusion Collaboratory, http://www.fusiongrid.org/.
77. Grid Physics (Particle Physics, Astronomy, Experimental Gravitational waves) Network GriPhyn, http://www.griphyn.org/.
78. Particle Physics Data Grid, http://www.ppdg.net/.
79. National Virtual (Astronomical) Observatory, http://www.us-vo.org/.
80. European Astrophysical Virtual Observatory, http://www.eso.org/avo/.
81. European Grid of Solar Observations EGSO, http://www.mssl.ucl.ac.uk/grid/egso/egso_top.html.
82. NEES Grid, National Virtual Collaboratory for Earthquake Engineering Research, http://www.neesgrid.org/.
83. Solid Earth Research Virtual Observatory, http://www.servogrid.org.
84. DoE Earth Systems (Climate) Grid, http://www.earthsystemgrid.org/.
85. Biomedical Informatics Research Network BIRN Grid, http://www.nbirn.net/.
86. North Carolina Bioinformatics Grid, http://www.ncbiogrid.org/.
87. European Grid Resources for Industrial Applications, http://www.gria.org/.
88. Grid Acronym Soup Resource, http://www.gridpp.ac.uk/docs/GAS.html.
89. List of Grid Projects, http://www.escience-grid.org.uk/docs/briefing/nigridp.htm.
90. UK e-Science Network, http://www.research-councils.ac.uk/escience/documents/gridteam.pdf.
91. HPC(x) Press Release July 15 2002, http://www.research-councils.ac.uk/press/20020715supercomp.shtml.
92. The Internet Engineering Task Force IETF, http://www.ietf.org/.
93. The World Wide Web Consortium, http://www.w3c.org.
94. Parallel Virtual Machine, http://www.csm.ornl.gov/pvm/pvm_home.html.
95. Parallel Computing Message Passing Interface, http://www.mpi-forum.org/.
96. Linux Online, http://www.linux.org/.
97. OASIS Standards Organization, http://www.oasis-open.org/.
98. Apache Software Foundation, http://www.apache.org/.
99. Apache HTTP Server Project, http://httpd.apache.org/.

100. Apache Axis SOAP and WSDL Support, http://xml.apache.org/axis/index.html.
101. Apache Jakarta Jetspeed Portal, http://jakarta.apache.org/jetspeed/site/index.html.
102. Protein Data Bank Worldwide Repository for the Processing and Distribution of 3-D Biological Macromolecular Structure Data, http://www.rcsb.org/pdb/.
103. MyGrid – Directly Supporting the e-Scientist, http://www.mygrid.info/.
104. Mcell: General Monte Carlo Simulator of Cellular Microphysiology, http://www.mcell.cnl.salk.edu/.
105. National Center for Microscopy and Imaging Research Web-Based Telemicroscopy, http://ncmir.ucsd.edu/CMDA/jsb99.html.
106. Telescience for Advanced Tomography Applications Portal, http://gridport.npaci.edu/Telescience/.
107. Casanova, H., Bartol, T., Stiles, J. and Berman, F. (2001) Distributing MCell simulations on the grid. *International Journal of High Performance Computing Applications*, **14**(3), 243–257.
108. Open Grid Services Architecture Database Access and Integration (OGSA-DAI) UK e-Science Project, http://umbriel.dcs.gla.ac.uk/NeSC/general/projects/OGSA_DAI/.
109. Space Link Extension Standard, http://www.ccsds.org/rpa121/sm_review/.
110. GEODISE Grid for Engineering Design Search and Optimization Involving Fluid Dynamics, http://www.geodise.org/.
111. Distributed Aircraft Maintenance Environment DAME, http://www.iri.leeds.ac.uk/Projects/IAProjects/karim1.htm.
112. The Grid for UK Particle Physics, http://www.gridpp.ac.uk/.
113. Djorgovski Caltech, S. G. New Astronomy With a Virtual Observatory, and other presentations, http://www.astro.caltech.edu/~george/vo.
114. Fox, G. (2002) e-Science meets computational science and information technology. *Computing in Science and Engineering*, **4**(4), 84–85, http://www.computer.org/cise/cs2002/c4toc.htm.
115. W. Wulf. (1989).The National Collaboratory – A White Paper in Towards a National Collaboratory, Unpublished report of a NSF workshop, Rockefeller University, New York, March 17–18, 1989.
116. Kouzes, R. T., Myers, J. D. and Wulf, W. A. (1996) Collaboratories: doing science on the Internet, IEEE Computer August 1996, IEEE Fifth Workshops on Enabling Technology: Infrastructure for Collaborative Enterprises (WET ICE '96), Stanford, CA, USA, June 19–21, 1996, http://www.emsl.pnl.gov:2080/docs/collab/presentations/papers/IEEECollaboratories.html.
117. Butterfly Grid for Multiplayer Games, http://www.butterfly.net/.
118. Everquest Multiplayer Gaming Environment, http://www.everquest.com.
119. Ensemble Climate Prediction, http://www.climateprediction.com.
120. Fox, G. (2002) Chapter 4, in Dongarra, J., Foster, I., Fox, G., Gropp, W., Kennedy, K., Torczon, L. and White, A. (eds) *The Sourcebook of Parallel Computing*, ISBN 1-55860-871-0, San Francisco: Morgan Kaufmann Publishers.
121. Synthetic Forces Express, http://www.cacr.caltech.edu/SFExpress/.
122. Defense Modeling and Simulation Office High Level Architecture, https://www.dmso.mil/public/transition/hla/.
123. MIAS Medical Image Grid, http://www.gridoutreach.org.uk/docs/pilots/mias.htm.
124. Condensed Matter Reality Grid, http://www.realitygrid.org/.
125. ProteomeGRID for Structure-Based Annotation of the Proteins in the Major Genomes (Proteomes), http://umbriel.dcs.gla.ac.uk/Nesc/action/projects/project_action.cfm?title =34.
126. BiosimGRID for Biomolecular Simulations, http://umbriel.dcs.gla.ac.uk/Nesc/action/projects/project_action.cfm?title =35.
127. Chetty, M. and Buyya, R. (2002) Weaving computational grids: how analogous are they with electrical grids?. *Computing in Science and Engineering*, **4**, 61–71.
128. Darema, F. (2000) Chair of NSF Sponsored Workshop on Dynamic Data Driven Application Systems, March 8–10, 2000, http://www.cise.nsf.gov/eia/dddas/dddas-workshop-report.htm.
129. Horn, P. IBM, Autonomic Computing: IBM's Perspective on the State of Information Technology, http://www.research.ibm.com/autonomic/manifesto/autonomic_computing.pdf