GRID COMPUTING

TERM PAPER REPORT

ON

"GRID COMPUTING"

Submitted By: - ANSHUL KUMAR SAXENA
64011
MCA-II YEAR

INSTITUTE OF MANAGEMENT AND ENTREPRENEURSHIP DEVELOPMENT,

BHARATI VIDYAPEETH UNIVERSITY

Acknowledgement

It gives me immense pleasure to put forward this practical venture. But surely, it would not have been possible without proper guidance and encouragement. So I would like to thank all those people without whose support this paper would not have been a success.

I express my deep gratitude to all concerned persons, whose help and guidance have contributed a lot in accomplishing this paper.

I am very thankful to honorable Director Dr. M.S.Prasad, the esteemed director of our college for providing all facilities for making the project successful.

I owe a special thanks to Prof. S.A.Kadam (IT Faculty, MCA Department, IMED Bharati Vidyapeeth University, Pune) for his able guidance and valuable suggestions for choosing and shaping this paper. His contribution has helped a great deal in the timely completion of this term paper.

I would like to express my sincere thanks to all the faculty members of MCA Department, IMED Bharati Vidyapeeth University, Pune for the interest and commitment shown in the project, which made it possible for us to complete the term paper successfully.

Date: Anshul Kumar Saxena

Bharati Vidyapeeth University

Institute of Management and Entrepreneurship Development, Pune

(Accredited by NAAC with "A" grade)

Certificate

We certify that Mr. ***Anshul Kumar Saxena Roll No. 64011*** *respectively is a bonafide student studying for* ***MCA*** *degree programme of the university in this institute for the year 2006-2009.*

As a part of the Course Curriculum he has completed a term paper titled ***"GRID COMPUTING"*** *during the period from* ***Jan 2007*** *to* ***May 2007.***

The term paper report is prepared by the student under the guidance of Prof. S. A. Kadam.

(Dr. M.S.Prasad)	(Prof. A.D.More)	(Prof. S.A.Kadam)
DIRECTOR	HEAD OF DEPTT.	PROJECT GUIDE
IMED BVP PUNE	IMED BVP PUNE	IMED BVP PUNE

ABSTRACT

Grid computing is currently an area of intense activity involving a large community of researchers, developers, and users. Hundreds of experimental and production grids are in operation throughout the world. The development of grid computing has followed a path similar to the development of the World Wide Web, which started as a technology for scientific collaboration but was later adopted for use by a multitude of industries and businesses.

Grid computing is defined as flexible, secure, coordinated resource sharing among a dynamic collection of individuals, institutions, and resources. The sharing of these resources must be tightly controlled, with resource providers and consumers defining clearly and carefully just what is shared, who is allowed to share, and the conditions under which sharing occurs. Therefore, Grid computing presents unique authentication, authorization, resource access, resource discovery, and resource management challenges.

Grid computing enables the virtualization of distributed computing and data resources such as processing, network bandwidth, and storage capacity to create a single system image, granting users and applications seamless access to vast IT capabilities. Just as an Internet user views a unified instance of content via the Web, a Grid user essentially sees a single, large virtual computer, which provides us a technology to optimize resources.

Grid computing aims to create a single virtualized system by pooling undedicated, disparate resources. Typical resource classes include computational, network connectivity, and data storage. These resources are combined, potentially in parallel, into tasks such as hosting Web applications, processing batch jobs, and hosting file downloads. The goal of an intelligent Grid is to:

_ Optimize the use of these resources based on constraints such as locality, availability of free resources, and the business value (or cost) of individual jobs.

_ Provide a sharable hosting environment for easier software installation and infrastructure maintenance.

Table of Contents

Part 1

Fundamentals of Grid & Grid Computing

1

Introduction to Grid

1. Introduction

Over the last few years we have seen grid computing evolve from a niche technology associated with scientific and technical computing, into a business-innovating technology that is driving increased commercial adoption. Grid deployments accelerate application performance, improve Productivity and collaboration, and optimize the resiliency of the IT infrastructure. By accelerating application performance, companies can more quickly deliver business results; achieving greater productivity, faster time to market, and increased customer satisfaction.

Grid technology also provides the ability to store, share and analyze large volumes of data, ensuring that people have access to information at the right time, which can improve decision making, employee productivity and collaboration. Grid technology improves resource utilization and reduces costs, while maintaining a flexible infrastructure that can cope with changing business demands, yet remain reliable, resilient and secure.

At its core, grid is about virtualization, of both information and workload. In non-grid environments, existing infrastructures are very much "siloed;" resources are dedicated to applications and information. Many such dedicated infrastructures exist for common applications such as HR, Payroll, etc. and for data/information mining purposes. System response is limited by server capacity – and access to the data stored. It is very difficult to dynamically respond to new requirements, as a new infrastructure would be required, inefficiencies would predominate, and full utilization across the many silos would be difficult to achieve.

In a grid environment, resources are virtualized to create a pool of assets. Workload is spread across servers and data can be seamlessly retrieved. By separating applications and information from the infrastructure they run on, and providing this abstract, "virtualized" view, a new level of infrastructure flexibility can be achieved. Infrastructures can now dynamically adapt to business requirements, instead of the other way around. Resources are more fully utilized, resulting in decreased infrastructure costs, reduced processing time, increased responsiveness and faster time-to-market.

IBM has been a strong advocate and practitioner in facilitating the commercial adoption of grid computing, even before the topic became a focus of media hype and analyst attention. Over the years, IBM has made investments in all aspects of the grid domain: standards definitions, technical development, open-source contributions, deployment of innovative business solutions, and nurturing of a robust ecosystem that extends to software developers and business partners.

We view grid as a game-changing technology that challenges basic assumptions of ownership, access, usage, operating efficiency, utilization of assets and total operating costs. A technology that fosters innovation and collaboration while helping customers establish a competitive advantage in their market. With particular emphasis on the future of the grid marketplace, this paper investigates the roots and evolution of grid computing germane to customer needs for associated business solutions that span the grid spectrum. Starting with a quick view into the origins of grid computing, we proceed to analyze its value proposition with respect to innovation and collaboration.

Furthermore, we present a holistic view of technologies associated with the grid and virtualization domain, and focus upon the importance of developing and adopting industry standards. We also explore significant market trends and dynamics and investigate critical challenges, especially those that extend beyond the compute elements of grid into data and information management. Throughout this paper, we define IBM's vision for grid computing and position grid relative to other IBM initiatives.

What is Grid?

Grid is a shared collection of reliable (cluster-tightly coupled) & unreliable resources (loosely coupled machines) and interactively communicating researchers of different virtual organizations (doctors, biologists, physicists). Grid System controls and coordinates the integrity of the Grid by balancing the usage of reliable and unreliable resources among its participants providing better quality of service. It means Grid System is synonymous to operating systems that control, co-ordinates and schedules the resources of a system.

"In 1854, George Boole (1815-1864), Professor of Mathematics at Cork from 1849 despite having no first degree, formalized a set of such rules in the seminal work entitled, perhaps optimistically, An Investigation of the Laws of Thought. Boole's aim was to identify the rules of reasoning in a rigorous framework and revolutionized formal logic after thousands of years of little progress. They transformed logic from a philosophical into a mathematical discipline. These rules have subsequently become known as Boolean algebra and the design of all modern binary digital computers has depended on the results of this work. These logical operations, normally implemented as electronic gates, are all that are required to perform more complicated operations such as arithmetic." The Virtual Museum of Computing.

A grid in the computing world, and particularly in Open Science Grid, is generally accepted as meaning a collection of networks, software, computers and possibly data intended for shared use by organizations of people. It links many people in many places to many computing resources, also in many places, in a regulated and secure way. Hence grid computing is computing that uses the resources managed by a grid. Ideally, grid users run their computing applications as needed without worrying about where the computers are, akin to the way we plug in a household appliance and access the electric power grid.

Grids are particularly well-suited to organizations that consist of a large number of geographically distributed members, all working on a common project or towards a common goal, and who require shared computing resources in order to accomplish their work. These organizations often span institutional and regional boundaries and experience membership changes frequently. Large scientific research collaborations exemplify these characteristics and in fact have provided a driving force behind grid development. Organizations that participate in grids must form "Virtual Organizations" (VOs), which is a grid term for an organization on which certain requirements have been imposed to facilitate access to the grid's resources.

A grid is made up of four layers of resources (hardware) and software. Each layer of grid architecture depends on the one underneath it:
The network layer is the base. It connects...

- ...the grid's resources, which make up the second layer. In addition to processing power and files, this layer may include data storage, databases, software repositories, and even sensors like telescopes, microscopes, and weather balloons!
- On top of the resources sits the third layer, the middleware, the "brains" of the grid. This software does all the work to connect users' jobs to computing resources, thereby hiding the grid's complexity from the user. It has the unenviable job of making many different networks and resources appear unified.

- Most people only interact with the fourth and uppermost layer, the application. This is the most diverse layer, as it includes virtually any program an end user wishes to run.

The structure and layers may vary somewhat from grid to grid, but these are the essential components.
Stringent requirements for security and accounting differentiate grid computing from other distributed computing models. Grid user authorization, for example, is handled through Virtual Organizations. A VO must authenticate and register its members, and enter into agreements with the other VOs on the target grid to define which resources are shared, who is allowed to share them, and the conditions under which sharing occurs. The middleware implements these agreements.

Grid is a software environment based on open standards and protocols that make it possible to share disparate, loosely coupled IT resources across organizations and geographies. IT resources are freed from their physical boundaries and offered as services. They can potentially include almost any IT component -- computer cycles, storage spaces, databases, applications, files, sensors or scientific instruments.

Web standards make it work. In grid computing, resources can be dynamically provisioned to users or applications that need them. Resources can be shared within a workgroup, department or enterprise; among different organizations and geographies; and even with groups outside the enterprise in collaborative projects. Grids can be designed to support various business processes. Grid technologies use emerging Web services standards such as XML, SOAP and WSDL. Bringing grid to the enterprise
Grid technologies have long been used for scientific and technical work, where dispersed computers are linked to create virtual supercomputers that rapidly process vast amounts of information. Now, with the success of e-commerce and the Internet, the commercial enterprise is moving to an IT model based on Web services, in which software can be offered and consumed as services -- a service-oriented architecture.

Grid is not a ready-made solution, but rather a set of components and protocols pulled together to create a solution. HP views grid computing as a powerful way to virtualizes resources and create a service-oriented architecture, where IT provides resources to business on demand, like a utility.

As an enabler of the Adaptive Infrastructure, grid links the IT infrastructure dynamically to business process software, so that changing business needs can be met in real time.

2

The Evolution of Grid Computing

2. The Evolution of Grid Computing

The phrase grid computing implies different technologies, markets and solutions to different people. Early on, much of the available literature focused on the compute-intensive problems made tractable by grid, often associating it with cycle-scavenging or job scheduling technologies. Although these are important and useful components of a grid, they do not by themselves deliver the complete grid vision.

The real “innovation” in grid comes from the combination of technology domains that include workload virtualization, information virtualization, system virtualization, storage virtualization, provisioning, and orchestration. From this statement, one may already conclude that no single technology constitutes a grid, but, instead, the method with which broad sets of resources are accessed and combined. Grid computing is not about a specific hardware platform, a database or a particular piece of job management software, but the way in which IT resources dynamically interact to address changing business requirements.

The grid domain has developed over a relatively short time period, fueled by significant technology advancements. Many grid technology roots can be traced back to the late 1980s in areas related to distribute supercomputing for numerically intensive applications, with particular emphasis on scheduling algorithms (e.g. Condor1, Load Sharing Facility2). By the late 1990s, a more generalized framework for accessing high-performance computing systems and distributed data (e.g. Globus3) began to emerge, and then, at the turn of the millennium, the pace of change quickened with the recognition of the potential synergies between grid and the emerging Service Oriented Architectures (in particular through the creation of the Open Grid Services Architecture – OGSA).

2.1 The Grid Value Proposition

The dynamic and flexible properties of the grid establish a more competitive and innovative business by enabling a more responsive IT infrastructure. IBM believes that this is not best achieved through the deployment of proprietary protocols or technologies. Invariably, these lead to ‘brittle’ architectures and solutions. Businesses need to be able to address their business issues of the day, leveraging assets they possess. Grid establishes a common vision and method for managing, referencing, and accessing the valuable IT resources available in an enterprise. This capability typically does not result from having applications locked to particular platforms, data to particular databases and user-access unnecessarily to particular systems.

The inherent flexibility that users and administrators/operators can derive is, perhaps, the most subtle and least quantifiable benefit availed by grid computing. Traditionally, IT infrastructure has been procured and deployed with a single purpose or function in mind. Systems have been established and grown in alignment with particular vertical businesses. Corporations have grown accustomed to managing a set of relatively distinct business units. As the global market encourages the lowering of barriers to international trade, deregulation fosters competition. The cost of entry into existing markets drops dramatically, the pace of market change increases, and the need to combine cross-organization resources, products and services grows. With this context, the traditional or "stove-pipe" IT solution is inefficient – in some cases we see it as a support for significant inertia which inhibits changes in business design.

There are different implications and derived benefits to the different functions within an organization. Understandably, there are distinctions between the implications to a Line of Business (LOB) executive, a Chief Information or Operation Officer (CIO or COO) and the Chief Operating Officer (CEO) of a corporation. LOB executives suffer from time, performance and quality pressures when driving to deliver to the business particular results or products. In this case grid provides a potential avenue for improving time to market and/or enhancing product quality – the highest value properties of grid are those linked to: process or application acceleration, improved access to underutilized IT resources, and system scalability. Embracing grid can give a "quick win" for those applications or problem sets which are easily subdivided to be run concurrently across the distributed IT infrastructure within the LOB department – and this is a characteristic of much of the early commercial interest in grid. IBM discussions with customers and LOBs often begin with a business requirement to more quickly process a job, transaction, or set of tasks to meet a particular business deadline. The processing or analysis may be anything from wanting to know the level of risk associated with an investment portfolio, which is needed during the trading day through to the execution of multiple Computational Fluid Dynamics (CFD) simulations to meet a particular design revision. With good reason, CIOs/COOs are more concerned with cross-organization integration and utilization of resources, enabling better access to information, mitigating business operating risk, more easily introducing new applications and systems across the company, and finally, managing heterogeneous systems while "keeping the whole thing running" at a reasonable total cost of ownership. The grid benefits that are most often aligned with their needs are:

a) improved sharing of all IT resources offered into the grid and greater opportunity for cross-organizational collaboration;

b) greater scalability of infrastructure by removing limitations inherent in the artificial IT boundaries existing between separate groups or departments;

c) increased ability to launch new projects or initiatives without being limited by what systems are available to a single group or department;

d) Reduction of overall IT costs.

But probably the biggest benefits of grids are derived from client ability to achieve new levels of innovation that can differentiate their business by implementing new business processes and applications that they would have been unable to accomplish using conventional information technology. Grid provides a virtual, resilient, responsive, flexible and cost effective infrastructure that fosters innovation and collaboration. In that regard, grid addresses the higher needs of organizations often associated with a CEO's agenda to find innovative ways to grow the business, improve the productivity of employees, and provide a sustainable competitive advantage.

2.2 Customer Adoption

Along with our profile of the motivations for grid computing, it is helpful to understand where the technology has gained greatest traction. Being intensely attuned to the commercial sectors which have showed the greatest affinity for grid computing, IBM has, correspondingly, aligned its efforts with five grid focus areas

1. Business Analytics Grid: Enabling faster and more comprehensive business planning and analysis through the sharing of data and computing power;

2. Engineering and Design Grid: Sharing data and computing power, for computing intensive engineering and scientific applications, to accelerate product design;
3. Research and Development Grid: Accelerate and enhance the R&D process by enabling the sharing data and computing power seamlessly for research intensive applications;
4. Government Development Grid: Create large-scale IT infrastructures to drive economic development and/or enable new government services;
5. Enterprise Optimization Grid: Optimize computing and data assets to improve utilization, efficiency and business continuity.

Each of these areas may span multiple sectors and industry applications (see Figure 1). Some of the early commercial adopters can be found in industries that exhibit close affinity to the grid paradigm such as Aerospace, Automotive, Agriculture, Petroleum, Electronics, Financial Services, Government, Higher Education, and Life Sciences. Furthermore, grid deployments have started to gain more traction in new industries such as Media & Entertainment for digital rendering and gaming applications.

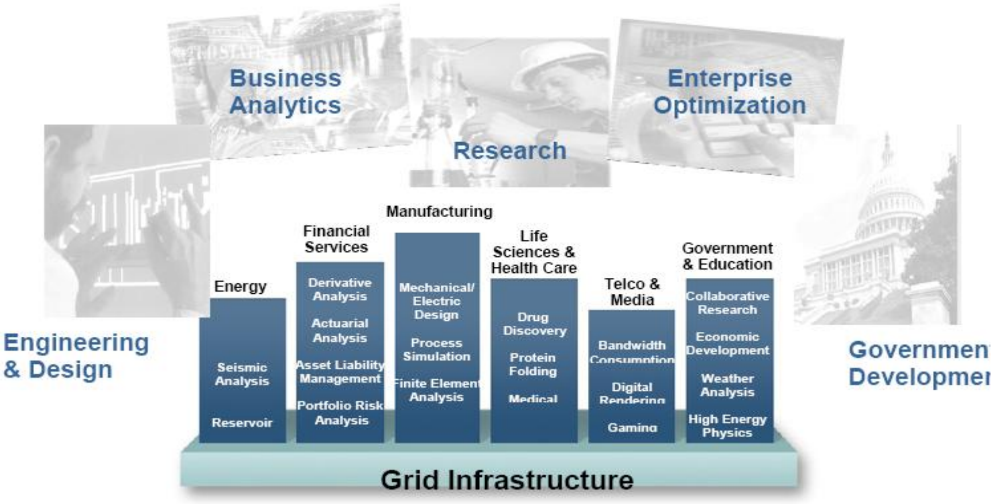


Figure 1. Grid Computing Focus Areas & Industry Applications

3

The Grid Spectrum

3. The Grid Spectrum

Grid is best understood as part of a continuum representing virtualization technologies and solutions (See Figure 2).

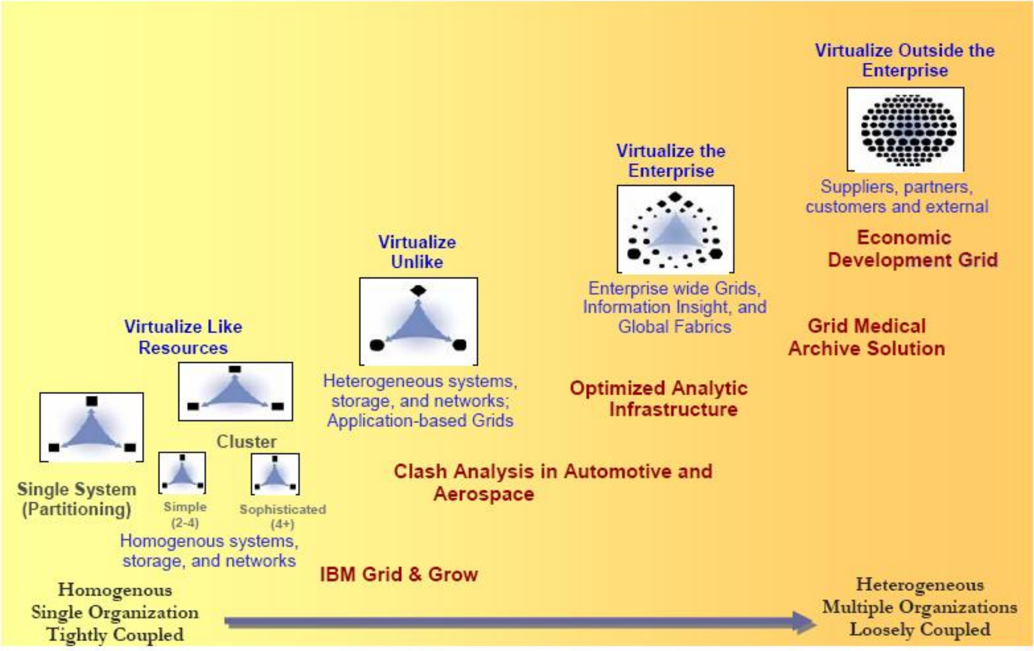

Virtualize Outside the Enterprise
Suppliers, partners, customers and external
Economic Development Grid
Virtualize the Enterprise
Enterprise wide Grids, Information Insight, and Global Fabrics
Grid Medical Archive Solution
Virtualize Unlike
Heterogeneous systems, storage, and networks; Application-based Grids
Optimized Analytic Infrastructure
Virtualize Like Resources
Cluster
Single System (Partitioning)
Simple (2-4)
Sophisticated (4+)
Homogenous systems, storage, and networks
Clash Analysis in Automotive and Aerospace
IBM Grid & Grow
Homogenous Single Organization Tightly Coupled
Heterogeneous Multiple Organizations Loosely Coupled

Figure 2. The Grid Spectrum

At their simplest form, grid technology and solutions can be deployed within a homogeneous environment or within a single organization in a tightly coupled manner. Virtualizing “like” resources often entails deploying grid and virtualization functionality on cluster environments or multiple single systems with partitioning. As discussed earlier, a common entry point for grid deployments is a single line of business or single department using grids or clusters to increase business value; application acceleration, meeting service level agreements, or gaining insights to data. Such adoption has been fueled by the ongoing proliferation of clusters within enterprises, which often allows for a smoother transition using parallel applications. Another driver in this space is the continued acceptance of Linux and open source for multiple applications and workloads within enterprises. Indeed, many grid implementations in this space do rely on Linux and open source software functionality. The next level of virtualization extends the grid concept – still within the domain of a single department or organization – to “unlike” resources. As a matter of fact, most departments often run applications and processes that are composed of unlike resources, whether these are servers, storage or other operating systems and software. The same principles of seamless integration using grid technology still apply, despite the loss of homogeneity.

Where we “cross the chasm” is when we move from virtualizing unlike resources within a specific organization or line of business to virtualizing across multiple lines of business (or departments) within an enterprise. That is the point where (real and perceived) issues regarding security, ownership, and governance surface. Grid is a powerful technology that challenges long held assumptions over ownership, access and usage. Clients are increasingly confronted with the prospect of allowing their databases, their storage devices, their system processors to be leveraged and used by others who did not pay for those assets. For a challenged LOB, grid can be mistakenly viewed as a method by which their department’s valued IT assets are wrestled from their local control and made available to other users in the enterprise.

We should point out that while such political struggles are often viewed as barriers to “crossing the chasm” into enterprise, adopting grid technology does not dictate that a department administrator loses authority or control over the resource. To the contrary, the resources are offered into the grid under a set of policies or conditions determined by the owner. For the grid to be effective and scale across the enterprise, it is important that enterprise IT establishes, up front, a base set of resource-allocating policies. As examples: in offering a computing cluster into a grid, the owner may declare the hours during which grid workloads will be able to run or what percentage of the systems will be available to the grid; a database owner is free to determine what data services are to be allowed; a storage system administrator is able to insist on particular security requirements users must meet to be able to access its files. To assist in this process, IBM offers grid middleware solutions that provide provisioning, orchestration, billing and service level agreement type solutions.

In short, there is no specific reason why allowing grid-based access to a resource dilutes the local administrator/operator control. But rather, one can manage the grid, get the benefit of leveraging the additional resources as needed, and transfer that benefit to others within the organization. That is when organizations are able to get much better balanced resource utilizations across the entire IT environment. Once organizations are able to get past the political barriers to enterprise optimization, they begin to attain previously unavailable business value from grid by starting to integrate horizontally. In that journey – which often equates to becoming an *on demand* business – organizations should consider implementing potential new governance models, new businesses and processes, and financial/accounting systems that provide incentives to participating organizations.

Finally, at the top right side of the spectrum, organizations start to virtualize with others outside the boundaries of the enterprise. A grid environment can leverage resources with suppliers and business partners and integrate business processes with the rest of the value net. These types of grids exist today primarily in the public sector where governments and academic institutions link their resources and share information to support collaboration and relationships across organizations, countries, states or local governments. There is incredible potential, however, in other industries such as automotive and aerospace where the design time and quality of an automobile or plane, for example, can be significantly improved by sharing design data across all the suppliers associated with the product.

To summarize, as customers move from left to right on the spectrum, they are moving from a homogeneous environment within a single organization that is very tightly coupled to one that is heterogeneous, distributed and loosely-coupled. It is important to note that there are *multiple potential customer entry points*, thus the spectrum does not necessarily imply a sequential progression. What is very important, however, is for the customers to preserve their ability to expand on their grid implementations and grow as their business requirements grow.

4

Grid Solutions

4. Grid Solutions

IBM has the expertise, technical capability and solutions, supported by relationships with business partners, to help clients achieve immediate value with grid with solutions that are right for them. In addition, IBM can provide customers a clear path for grid expansion along the progression map. IBM’s solutions strategy offers a set of repeatable solutions that can reduce the time and risk of implementation. These solutions are based on repeatable patterns found across numerous client engagements. Although there is a plethora or grid offerings in targeted market segments and industries, in the following sections, we just highlight a few key solutions.

4.1 IBM Grid and Grow

The IBM Grid and Grow offering provides an easy to deploy, integrated solution for customers interested in beginning the grid journey. It includes industry-leading IBM and business partner technology along with a "get started" services package to help first time grid customers maximize the benefits of grid computing. Grid and Grow leverages a customer’s existing investment and lays the foundation to expand to larger, more robust grid implementations, further optimizing their infrastructure as their needs grow. This solution can also be leveraged to expand capacity or build redundancy or to existing resources. The offering is also part of the IBM “Express” solutions portfolio.

The offering features the IBM BladeCenter with seven blades and a choice of three server architectures: Intel HS20, AMD LS20, or POWER JS20. These blades can be mixed and matched in a single chassis and run one or more of RedHat or Novell SUSE Linux OS, AIX 5L, or Windows. Additional blades can be easily added to fill out the existing chassis, or expanded to multiple chassis for optimal scalability.

Customer choice continues with grid scheduler options: Altair's PBS Professional™, Platform Computing's LSF, DataSynapse’s GridServer or IBM’s LoadLeveler. These enable application scheduling, efficient dynamic resource allocation and resource sharing. Rounding out the offering is a services package that includes: a site readiness assessment, hardware and software installation and tailoring, grid application readiness assessment, testing, documentation and client training. Customers can expand the initial Grid and Grow implementation to a larger, more robust deployment to address additional business needs. Tools that could add significant value are included in products from the IBM Tivoli and WebSphere portfolios. Some examples include dynamic provisioning, dynamic software license tracking, and a standards-based secure portal.

IBM has designed the Grid and Grow platform as a solution building block that application vendors can built upon with key applications that may be important to client business.

4.2 IBM Grid Offering for Engineering Design: Clash Analysis in Automotive and Aerospace

In an intensely competitive marketplace, automotive and aerospace companies must achieve faster time to market by decreasing the turnaround time for product design. By significantly decreasing the amount of time required to assess whether new designs affect, or *clash*, with existing product structures, designers can more quickly evaluate alternatives. In addition, automobile and aircraft companies face great pressure to reduce IT investments and increase return on investment (ROI), requiring them to maximize the use of their existing infrastructures. Meanwhile, their heterogeneous environments – spanning many departments and partners – are inherently complex. In this case, optimizing existing compute resources holds the key to balancing market needs and costs.

The IBM Grid Offering for Engineering Design: Clash Analysis in Automotive and Aerospace helps automotive and aerospace design engineers use grid technology for more rapid evaluation of design alternatives during sub-assembly clash analysis. Developed in cooperation with Platform Computing, the offering includes CATIA® and ENOVIA® application software. It helps reduce the time required to capture, compile and analyze clash research data and can accelerate product development and time to market. The offering also includes a Grid Innovation Workshop for assessing and planning a grid network, a pilot design and implementation services and comprehensive portfolio of IBM Global Services product lifecycle management (PLM) for implementing and tuning product design, data management and clash analysis software.

4.3 IBM Optimized Analytic Infrastructure

To be competitive in today's financial services marketplace requires performance of complex analytics and computations in near real-time for a broad portfolio of products and activities such as derivatives, structured and fixed-income products, risk management, program trading, actuarial analysis, hedging and portfolio rebalancing. From a business perspective, firms need to accelerate development of complex financial products with shorter life cycles. A repeatable and consistent process for global deployment of applications is also critical. This enables the pursuit of higher margins and revenue growth while meeting client demands for innovation and specialization. From a technical perspective, financial services firms demand a dynamic IT infrastructure that rapidly responds to changing business needs and requirements. This requires an operationally efficient analytics infrastructure scalable for increasing data volumes, complexity, and a spectrum of computational profiles. As well, this infrastructure needs to be inherently low latency, extremely fast, standards-based and highly flexible, and address data center constraints of space, power and cooling. IBM Optimized Analytic Infrastructure (OAI) responds to these requirements and has been designed to help financial services firms address their business and technical concerns so they can compete more successfully in today's environment. The IBM OAI addresses a primary requirement for financial market firms, to make rapid and extremely accurate decisions in a stable, scalable and robust environment to support trading, analytics and risk-management. The IBM OAI supports a broad spectrum of numerically intensive business processes and applications by leveraging grid computing, HPC, Linux and blade technologies from both IBM and Business Partners.

The IBM OAI includes products such as IBM’s GPFS for data management, IBM LoadLeveler for workload management, Cluster System Management for centralized administration, and the IBM ApplicationWeb. These solutions have been used by some of the largest supercomputing labs for over a decade to support a range of applications, such as high-energy physics, search analytics, weather modeling, and electronic chip design on geographically distributed systems in very large and dispersed user communities. The IBM Optimized Analytic Infrastructure solution complements IBM technologies with products from ISVs such as Altair PBS Professional (highly scalable scheduling environment), Scali MPI Connect™ (MPI programming model) and GemStone Systems (message board/virtual shared memory application environment) and the Linux operating systems (Red Hat and Novell SUSE). All the IBM and ISV products have been tested using representative workloads to help ensure full interoperability. Solutions like the IBM OAI will enable businesses to significantly improve the speed and accuracy of decisions through the use of grid and HPC technologies.

4.4 IBM Grid Medical Archive Solution

The IBM Grid Medical Archive Solution (GMAS) allows multi-campus hospitals and imaging centers to link geographically disparate sites – and modalities – together, helping optimize storage utilization and eliminate redundancy. IBM GMAS brings together shared pools of storage via deployment of intelligent grid software. Grid middleware create a “virtual” medical imaging archive that pools distributed storage, yet does not require the consolidation of physical images. By applying configurable business rules, storage grids can keep multiple copies of images in geographically distributed sites, eliminating the need for a physical disaster recovery site.

IBM GMAS is designed to be self-healing and self optimizing. It can also be deployed in support of multiple applications and across heterogeneous storage hardware. Healthcare providers can potentially gain significant ROI through increased storage utilization and simplified hardware administration.

Based on Bycast StorageGRID software, IBM GMAS is designed to cost effectively deliver enterprise-wide medical image access, regardless of the image’s physical location or sourcing system in an environment rich with security features. IBM GMAS delivers a unified storage system that can support multiple picture archiving and communication systems (PACS), enabling clinicians to view and share patient images at any time, from any location, using familiar PACS interface.

4.5 IBM Economic Development Grid

IBM has launched an initiative to enable communities worldwide to stimulate economic growth through the use of grid computing and other open standard technologies, such as Linux. Cleveland is the first region to benefit from this Economic Development Grid initiative, which is part of IBM's government development focus area to allow state and local governments, higher education establishments, and local businesses to share information by leveraging computing power and resources that benefit communities. State and local governments continue to be challenged to find ways to collaborate and attract new businesses to their communities to support economic growth, deliver and improve essential services such as education and health care for their citizens, and create a climate for innovation and growth to address future needs. Grid computing allows organizations to dynamically share information and computer resources, because it provides benefits that are not available with traditional IT infrastructure strategies. Grid computing applications in healthcare, life sciences, software development, digital media, manufacturing and petroleum can all enable economic benefits.

There are many types of grid computing implementations that can help communities drive economic growth. A couple of examples include compute-intensive grids for software development and medical research, as well as more data intensive grids that deliver collaboration benefits for healthcare and education.

4.6 IBM Global Services Grid Offerings

IBM Global Services offers a variety of offerings aimed at moving customers beyond the concept phase or helping customers expand their existing implementations. Some key offering include:

• Accelerated Design Services for Grid: Developed by the IBM Grid Integration Center in Austin, Texas - which integrates best-of-breed technology from IBM and IBM Business Partners - the Grid Accelerated Design Services enable clients to build grid solutions faster and more efficiently based on the experiences and results of other clients that have successfully deployed grid solutions. The offering supports multiple applications in a heterogeneous environment integrating grid middleware packages, workload virtualization, storage virtualization, orchestration and provisioning, and license management. In addition, the offering can help define and enforce policies and priorities to control resource sharing across organizations.

- Grid Innovation Workshops: These two- and three-day workshops introduce grid computing concepts, benefits and adoption frameworks, along with industry-specific value propositions. Initial opportunities to leverage grid computing technologies are identified, including business process considerations, top-line economics, technology architecture and potential risks.
- Grid Value At Work Tool: Developed by financial optimization experts from IBM Research and IBM Global Services, this tool provides detailed and quantifiable business value output for grid computing. Using industry templates, it examines multiple grid and non-grid scenarios prior to implementation to predict application performance and return on investment.
- Grid Computing Application Enablement: A service that enables applications to be adapted to operate in a heterogeneous environment and to exploit grid computing resources. The service also includes porting efforts to one or more of the platforms running the grid for improved processing performance.
- Grid Solution Deployment Services: A service that includes the implementation of infrastructure, application software, middleware, management tools and management processes as needed for a successful grid deployment.

5

Information Grid:

An Information Infrastructure for Grid Computing

5. Information Grid: An Information Infrastructure for Grid Computing

Every year, data volumes increase 800 MB per user. The sheer quantity of information can overwhelm the IT systems that must collect, store, retrieve, manage and protect it. Nearly onethird of an IT staff's time is spent searching for relevant data. Although the data may be timely, is it easy to use? Is it integrated? Is it tailored to business needs, and is it cost-effective to manage and retrieve? With a dynamic infrastructure, enabled by IBM grid computing technology, customers can deploy capabilities that allow them to capture and analyze customer information to increase the speed and accuracy of business decisions. IBM calls this capability "Information on Demand". IBM can help turn disparate data into true business insights. As a result, more useful business information can be gained from the raw data that is stored.

5.1 Information Infrastructure

Establishing an information infrastructure for grid is a core component of the grid computing model. It allows end users and applications secure access to any information source – regardless of where it exists – over a local and/or distributed network in intranet, internet or even extranet environments. It provides access to heterogeneous files, databases, storage systems and supports data sharing for processing and/or large-scale collaboration. The initial focus of a grid implementation may be on shortening the processing time of a single application. However, as more applications and system resources become associated with a grid environment, the need to consider how data is accessed and managed must be taken into consideration. While a grid may be optimally constructed to intelligently schedule and manage workloads, a poor information-access and storage-system deployment scheme could significantly reduce the benefits that can be derived from a grid implementation. In a grid environment, applications are not necessarily dedicated to running on specific processors (or nodes). Applications can be provisioned onto different processors within the grid at any given time, depending on their business priority. Moreover, nodes in the grid can be geographically dispersed. The challenge is making sure that data is easily accessible and doesn't create network bandwidth problems as a result of transporting it to computing locations in a distributed environment. Customers would like to be able to have an easier method of pulling together data and information from multiple "business" areas within and beyond the enterprise, without disturbing the original format of the data (or how the data is managed at its source). Therefore, it is necessary to ensure that any node in the grid can access the data ubiquitously, without having to build a new path to access it. Physically consolidating data can be incredibly expensive, time consuming and negatively impact performance of applications. Information-grid solutions may also address the following customer challenges:

- Fragmentation of data resources and assets due to a heterogeneous environment or underutilized compute and storage resources
- Cumbersome data access and poor integration
- Data security and protection
- Complex management of decentralized systems and resources

5.2 Challenges and Solutions

The information grid solves the problem of managing information, which may include databases, files, storage spanning across heterogeneous resources and software and hardware. The following are some common computing challenges and their solutions.

- **Challenge No. 1:** Accessing “heterogeneous data” stored in different formats across multiple “business” areas. The application must perform multiple I/O requests to retrieve the data that slows down the execution of the job. Programmers that build and maintain these applications must be aware of different formats and determine how to transform and join the data within their applications.

Solution: Data-access virtualization technology across diverse data formats is instrumental in helping solve the challenge of reading data stored in different formats. Programmers can simplify access to data that are stored in mixed formats (e.g., multivendor rational databases, flat files) by enabling these data to be accessed with a single structured query language instruction. Such access also helps reduce the need to move remote files.

The data’s virtual view is also known as federated access to the data. That is, making the data appear as one source even though the data are distributed and stored in mixed formats. In the event that large volumes of data need to be tailored for an application, specific extraction, transformation and loading functions can be preformed on the grid. Once the data has been prepared into a proper format for an application, the data can be temporally “transported” to and cashed at a location where the processing will take place. The ability to cleanse, transform, federate and analyze data across “heterogeneous data” sources is crucial to gaining information insight and making more effective business decisions. WebSphere DataStage, ProfileStage, QualityStage and WebSphere Information Integrator address this "heterogeneous data" challenge. These products are being integrated into the “IBM WebSphere Information Server”.

- **Challenge No. 2:** Data discovery and information delivery in a grid environment with mixed file system types. It is difficult for application developers and end-users to locate and access data because data are stored under multiple directories that are associated with each file system type.

Solution: Poor storage-resource utilization can be resolved through the use of SAN technology. Optimal solutions would include SAN software that enables system administrators to create a virtual view of all of the SAN storage, making them appear to be one homogeneous set of files with a common name space. Also, it is necessary to move large data volumes across a network to facilitate remote processing. A software solution that addresses this challenge should enable data to be cached close to where distributed processing occurs. An ideal solution would include global naming, secure wide-area access to consistent, current data and distributed data access including POSIX/NFS interface, access control and remote-data caching. Similarly, virtualization of heterogeneous file systems can help manage a complex SAN environment. Creating a single name space for the file system helps programmers and administrators locate and access data more easily versus having to identify files individually and determine what access path is required to reference the data. IBM's General Parallel File System or SAN File System can be used as foundations for this solution

- **Challenge No. 3:** Mixed vendor storage is common within any given enterprise. It is costly for administrators to manually manage data placement across heterogeneous sets of storage devices. In many cases, bottlenecks occur in retrieving data from these devices due to congestion of over utilized devices even though there may be space available on underutilized storage resources.

Solution: Frequently, customers have heterogeneous (multi-vendor) storage devices installed. Each vendor's storage device comes with its own management console, which makes it difficult to efficiently manage data placement across the various devices and ensure that there isn't uneven data loading. Uneven data distribution could cause some of the devices to be over utilized while others remain underutilized. This unbalanced condition could lead to bottlenecks when attempting to retrieve data, thus slowing the application processing.
A virtualization portal that consolidates the view across all of the SAN devices allows a single administrator to see how data is being loaded on these devices. Administrators are then able to shift data from over utilized devices to those that are underutilized without disrupting how applications access the data. IBM has the ability to deliver the building blocks for such solution having industry-leading storage virtualization products such as SAN Volume Controller and SAN File System. Other considerations in a SAN environment include error detection and data resiliency. It is important that data are protected and secure while providing the right data to applications.

5.3 The Optimal Information Infrastructure

Once the aforementioned solutions have been applied, the information grid has been established. These solutions address many potential problems of accessing data, managing heterogeneous file and storage systems and removing the network effect of supplying data for remote processing. By applying each of these solutions to create a virtual environment, distributed computing in a grid environment can achieve its maximum benefits. The enterprise has the flexibility to utilize all of its computing assets by enabling information to be virtually managed and presented.

6

The Importance of Standards

6. The Importance of Standards

IBM is a strong advocate for open industry standardization in information technology. Not only do open standards enhance interoperability, integration and customer choice, but also create an important opportunity for communities of vendors, governments, universities and researchers to innovate in the development of new computing paradigms. The very nature of grid computing, which tries to take broadly distributed, heterogeneous computing and data resources and aggregate them into an "abstracted" set of capabilities, almost demands open standards for integration and interoperability.

Most organizations that "build" grids do not purposefully acquire new hardware, operating systems and software to create their solutions, but rather integrate existing heterogeneous systems into a fabric over which they can deploy applications. While many vendors have successfully implemented distributed systems with proprietary interfaces, it can be argued that only "true grid" systems based on standards are capable of achieving the "scale out" promised by the "grid vision" – where an application can exploit any processing capability required, access any data it needs, and not be concerned with the specifics of configuration, management, or infrastructure.

6.1 Web Services: A Foundation for Grid Computing

Our fundamental approach over the past three years has been to base grid computing architecture and standards on the emerging foundation of a "services oriented" model that is increasingly being adopted by the IT industry. Web services provide a component model for the composition of functions that make up and support distributed systems and grids. The loosely coupled model of constructing applications, management functions or infrastructure using a collection of web services is ideal for building scalable, flexible, dynamic systems like grids. IBM has provided significant technical leadership to develop the WS-Resource Framework (WSRF) and WS-Notification Framework under the auspices of the Organization for the Advancement of Structured Information Standards (OASIS). WS-RF and WS-Notification provide the needed "stateful" web services environment on top of which other grid specific standards can be implemented. More recently, IBM in partnership with Microsoft, Intel, and Hewlett Packard has been working on a "convergence" plan for these and a number of other overlapping specifications.

In addition to these very fundamental web services standards, there are additional standards being worked on that add important functional capabilities like security (WS-Security), service level management (WS-Agreement) , policy expression (WS-Policy), etc.

6.2 Higher Level of Grid Specific Standards

Grid Standards

Beyond these fundamental and very general purpose web services specifications, is another group of standards that builds on top of them, to define more functional protocols and operations (e.g. scheduling and workload management, application deployment, resource provisioning, data movement, and data access). The Global Grid Forum (GGF) is involved in the definition of a number of important standards at this level. Most notable is a broad architectural focus called the Open Grid Services Architecture (OGSA) which defines a very rich vision of execution, management and data services to enable the creation of grids. OGSA is not a single specification, but rather a set of related standards in several areas. Some important OGSA-related specifications include:

- OGSA Basic Profile
- OGSA Security Profile
- Basic Execution Services (OGSA-BES)
- Job Submission Description Language (JSDL)
- Data Access and Integration Services (DAIS)
- Configuration Description, Deployment, and Lifecycle Management (CDDLM)
- OGSA Byte I/O (ByteIO)

Information Model

While several existing standards have embedded within them some form of resource representation, it has become apparent that a truly common framework for an abstract information model is needed. To this end, the Global Grid Forum (GGF) and other standards bodies have increasingly turned to the Distributed Management Task Force (DMTF) Common Information Model (CIM) as a touchstone. CIM has been developed over a number of years to describe all kind of IT resources, from very high level conceptual capabilities to very specific low level components. While the GGF and OGSA working groups have not yet formally identified DMTF CIM as the information model that they will use for grid computing, they are working towards that direction.

Management Standards

Along with the push to develop a common modeling approach for web services based standards, there has been a strong motivation to establish common high level protocols and operations for managing resources that are exposed as web services. OASIS' Web Services Distributed Management (WS-DM) is an industry-wide standard for management both *using* web services and *managing* web services. WS-DM attempts to exploit web services technology to create a universal and consistent abstraction for management and manageability interfaces that leverage key features of web services protocols. The specific types of management capabilities exposed by WS-DM include:

- Monitoring the quality of a service associate with a service
- Enforcing a service level agreement (quality of service)
- Querying or controlling the basic operational state of a resource
- Managing a resources lifecycle (create / destroy)

As in the case of the foundational web services standards, there is an ongoing effort to develop a “convergence” plan for overlapping management standards such as WS-DM and WSManagement. In summary, there is a growing collection of related standards and architecture being developed in open standards bodies like IETF, W3C, OASIS, DMTF and GGF that are all based on web services and can be composed to help develop interoperable grid middleware and infrastructure. IBM has been a leader in the definition and development of these open standards and is driving its important implementations along this standards roadmap.

Grid standards and requirements.

Area	**Requirements**	**Existing Standards**
Policy and service-level agreements	• Policy-based negotiation to establish grid service integration • Service-level agreement (SLA) management across heterogeneous grid providers • Advance reservation mechanisms based on customer requirements for QoS	OGSA Policy, WS-Policy, and WS-Agreements Advance Reservation Application Programming Interfaces
Job management	• Identification of individual schedulable entities • Definition of a common information model for job execution requirements, workflow characteristics • Definition of scalable and extendable mechanisms for job descriptions	Resource Specification Language (RSL), Job Submission Description Language (JSDL), Job Submission Information Model (JSIM), and Business Process Execution Language (BPEL4WS)
Grid scheduling	• Scheduling and executing individual jobs, workflow management • Meta-scheduler interfaces with	Grid local scheduling, Meta-scheduling, and Unified scheduler

	capabilities for resource discovery, provisioning, resource scheduling, and job execution • Creating a unified scheduler that acts as a provider for all the grid resources in a virtual organization and as a meta-scheduler for execution of jobs	
Grid service security	• Sandbox (protected domain that places tight controls around the execution of downloaded code) security model for grid service execution • Federation of security across heterogeneous resource providers • Trust model and certificate management in a federated environment	WS-Security, WS-Trust, WS-Federation, Grid Security Infrastructure (GSI), and Generic Security Service extensions
Grid management model	• A common management model to describe and interact with heterogeneous resources • A standards-based monitoring system with systems management capabilities • Performance management to meet SLA requirements for resources • Alignment with systems management concepts	Common Management Model (CMM), Grid Monitoring Architecture (GMA), and Web Services Distributed Management (WSDM)
Grid data management	• Access to and integration of structured and semi-structured data across grid • Discovery and storage of multitudes of data • Enhanced data mining for global information exchange • Virtual file system directory service and grid file system • Replica management	DAIS (Grid Data Access and Integration), XQuery, Virtual File System, Directory Service (VFSD), Grid File System (GFS), Local replica catalog services, and GridFTP
Grid and network	• Handling varying load on the network infrastructure due to heterogeneity of resources and policies applied to resources • Handling the varying type data stream and its impact on the network	Grid High-Performance Network (GHPN) and Open Service Gateway Initiative (OSGi)

	• Managing dynamic network conditions • Managing the SLAs among customers and service providers • Aligning with emerging network standards such as IP-V6 • Utilizing mobile network standards	
Grid architecture and programming models	• Defining an open architecture model for the grid • Aligning the architecture with the existing/emerging programming models and standards • Defining use cases, interoperable solutions and best practices for grid adoption	Open Grid Services Architecture (OGSA), Open Grid Services Infrastructure (OGSI), Web Service Resource Framework (WSRF), Java** Network Initiative (JNI), Semantic Grid, and New Productivity Initiative (NPI)

7

Technology Development and Integration

7. Technology Development and Integration

IBM is integrating grid functionality (provisioning, workload scheduling, resource management and information virtualization) across the software portfolio, and taking a leadership position in accelerating grid standards development and adoption across the industry. IBM's grid technical strategy extends the company's leadership in software, systems, and storage virtualization by focusing on three main areas: workload virtualization, information virtualization, and grid management. In addition, IBM is actively working on new programming models, tools, and techniques for developing and enabling distributed grid applications. For all of these areas, IBM is actively involved and is a leader in creating and developing the appropriate grid and Web services standards. IBM's workload virtualization strategy is to create a single, logical view of workload scheduling, differentiated with IBM automation and management technologies. This will enable clients to dramatically accelerate performance of multiple large application workloads across their enterprise, leveraging and orchestrating IT resources in a more flexible and dynamic fashion than ever before. Interoperability, driven by standards, is important in this space to provide a cross organization workload management capability spanning different types of scheduling environments and domains.

IBM’s information virtualization strategy is to create an integrated view of storage, file systems, and databases driven by standards, interoperability and advanced technologies like data transformation, security, caching and replication. The end result of this strategy will be the ability for clients to gain insight from disparate information federated systems in ways never before envisioned. The integration of these core grid technologies enables HPC solutions to deliver accelerated application performance, increased asset utilization, and information insight across disparate IT resources, data centers and geographies.

8

Grid and Service Oriented Architectures

8. Grid and Service Oriented Architectures

Customers are implementing Service Oriented Architectures (SOAs) to reduce complexity, to enable flexibility, and to streamline business processes. A SOA is a framework that lets customers build, deploy and integrate services for IT resources, applications and business process flows. It facilitates integration, offers modularization of applications and provides a coherent view of a business process as a set of coordinated services. Therefore, customers can build and integrate applications and business processes across their enterprise, as well as with partners, suppliers and customers, in a much more on demand fashion. In order to realize these benefits, companies need to establish a flexible infrastructure capable of supporting dynamic operations with flexibility inherent throughout. As companies continue to implement SOAs throughout their enterprise, there are going to be a lot of services, some very fine grained and some larger, that will require a very responsive, dynamic and scalable infrastructure. These services can be mobile. They need to be resilient. And the whole process needs to be accomplished in a simple manner. The underlying infrastructure needs to be adaptable and autonomic in providing the right resources to the right application services and business policies to meet service level agreements and business performance needs. The necessary infrastructure consists of our leading capabilities across system and resource virtualization, and grid computing. IBM views grid computing as critical to the ongoing development of a dynamic and flexible infrastructure that enables SOA:

- Just as an SOA allows customers to separate applications from services, grids allow customers to separate both applications and services from the infrastructure and systems resources. Scheduling and workload management are key capabilities here supporting the placement and mobility of services and composite applications to the appropriate resources.

Grids provide an underlying foundation to support the dynamic nature of SOA.

- Companies can pool resources for services, improve availability and reliability, and rapidly deploy and scale this new class of composite applications.
- With a virtualized infrastructure, customers can much more easily scale their support for SOAs. They can harness all of their resources to accelerate time to results and to better align their infrastructure performance to their business goals.

None of this will be possible at the SOA level without a "commitment to openness". It is the only way to link virtualization and grids with SOAs in a consistent and uniform manner. The infrastructure has to thoroughly support the ability to be managed in real-time, with sophisticated monitoring capabilities.

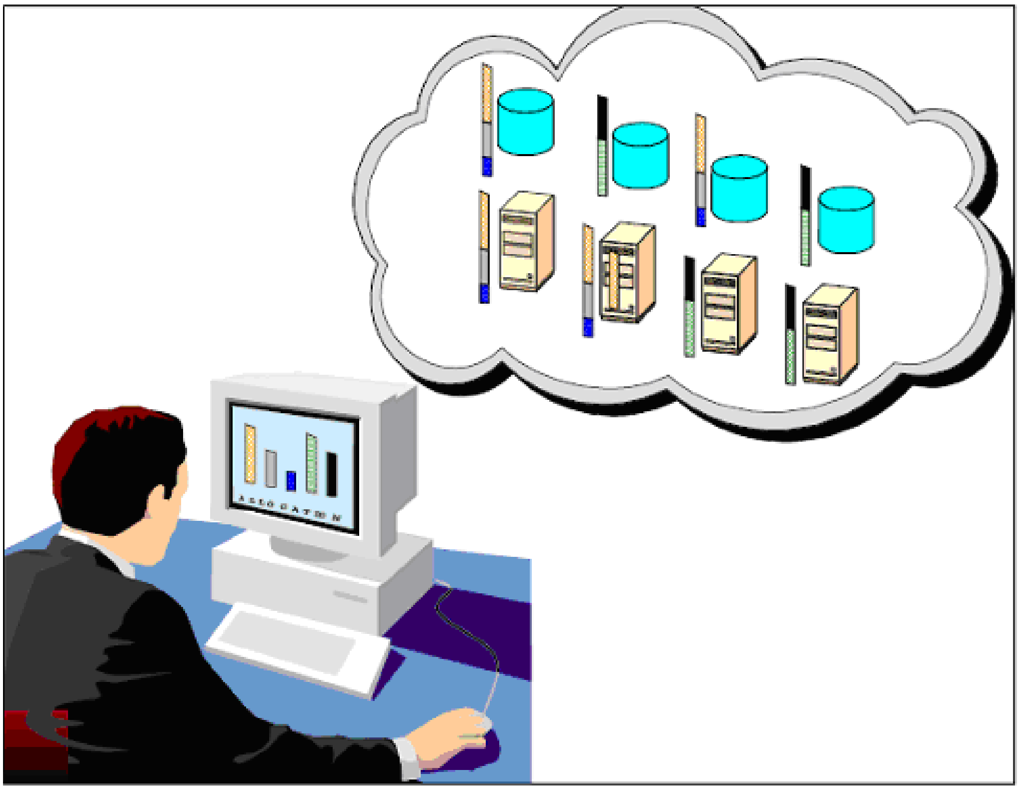
Finally, the SOA and virtualization landscapes require a significant ecosystem of partners and capabilities. We need to see cross industry collaboration to drive the innovation required for SOA. In simple terms, grid computing, based on open standards and supported by collaborative communities, delivers the underlying dynamic infrastructure that enables competitive advantage for customers implementing SOA-based solutions.

9

Grid Ecosystem

9. Grid Ecosystem

The primary basis of grid is that it can accommodate highly-heterogeneous technologies. Therefore, the viability of such technology requires the emergence of a strong, and viable, ecosystem that includes software vendors and business partners. By abstracting the physical resources, as well as the virtual resources, software developers can widen their target customer base. Specifically, by preparing their applications to run across infrastructures, as opposed to being the infrastructure, they make a better case for utilizing equipment that their application process requires.



To ensure that software developers and partners are able to adapt their products and services for use in the grid, IBM is nurturing an ecosystem to assist the grid solution creation process. Our efforts range from educating, enabling and assisting partners and independent software vendors (ISVs) on when and where grid technologies apply, through to helping create reference implementations for particular grid solutions. Some key examples include:

- *developerWorks GridZone*: The developerWorks GridZone provides software developers with tools, online training, IBM Redbooks, articles, emerging technologies from IBM Research, and more, to help them develop grid computing applications.
- *IBM Innovation Centers*: At the centers, the technical consultants work with the ISVs to help them implement their application topologies on a grid infrastructure. The infrastructure can consist of any hardware platforms or any of the supported operations systems.

• *The Solutions Enablement Virtual Loaner Program (VLP):* The VLP uses grid computing and other on demand technologies, such as the IBM Tivoli Provisioning Manager, to provide a rich and flexible software development environment for remote-access use by ISVs. ISVs are able to reserve, in advance, resources on the grid to satisfy their need for low-cost access to current IBM hardware and middleware to develop, port, test and validate their applications.

• *IBM Ready for Grid Program:* IBM's Ready for Grid computing program validates that an application is capable of executing and realizing benefits from running in a grid computing environment. The new program also includes "The Ready for IBM GRID Computing" mark, which is a critical component of IBM's strategy to create a robust ecosystem with our partners around open grid standards.

• *Value Network Initiative:* The Value Network Initiative builds networks of partners who can effectively deliver grid solutions. This program offers select partners access to enhanced PartnerWorld Industry Network (PWIN) co-marketing benefits.

Conclusion: ***Innovation that Matters*!**

Throughout the last decade, grid computing has emerged as a promising virtualization paradigm capable of providing a flexible, dynamic, resilient and cost effective infrastructure that can promote both collaboration and innovation within enterprises. Advancements in grid computing are driving increased adoption into commercial lines of business spearheaded by customer desire to create a competitive advantage through more responsive SOAs.

IBM is poised to extend its leadership in grid computing in order to advance innovation that matters across a much broader commercial marketplace, extending its benefits to community and society. Life sciences organizations, for example, already are leveraging the new efficiency afforded by grid technology to advance human health. In one example, scientists at the University of Pennsylvania have found a way to use grid computing to promote early detection of a disease that affects one out of every eight women today: breast cancer. The technology solution is known as the National Digital Mammography Archive (NDMA). The product uses grid technology to facilitate the complicated data access and analysis required for timely and accurate breast cancer screening.

In another example, the LA Grid program will link faculty, students and researchers from the world renowned IBM T.J. Watson Research Centers across the United States, Latin America and Spain to collaborate on innovative industry projects for applications in areas such as health care, life sciences and nanotechnology; and in regionally-specific concerns like hurricane mitigation.

Furthermore, IBM announced a new research effort to help battle AIDS using the massive computational power of World Community Grid, a global community of computer users who have joined the philanthropic technology initiative by simply donating unused time on their personal computers. The new World Community Grid initiative will deploy massive computer power to develop novel chemical strategies effective in the treatment of HIV-infected individuals in the face of evolving drug resistance in the virus. Developing new, more robust therapies to prevent the onset of AIDS in individuals infected with HIV will be the focus of this innovative project.

Grid computing continues IBM's history of IT innovation for business. IBM has the skills, experience and resources to attack the most challenging customer problems in the industry – on a global scale, with a deep understanding of clients' environments and requirements.

Part 2

Grid Architectural Design

10

The Grid Design

10.1) Building a grid architecture

10.2) Grid architecture models

10.3) Grid topologies

10.4) Phases and activities

10.5) A conceptual architecture

Design

This chapter provides architectural design considerations for grid computing. Other design topics that will be discussed are different grid topologies, grid infrastructure design, and grid architecture models.

At a glance, the following topics are discussed:

_ Grid architecture design concepts

_ Different grid topologies

_ Grid architecture models

_ Building grid architecture

_ Grid architecture conceptual model

10.1) Building a grid architecture

The foundation of a grid solution design is typically built upon an existing infrastructure investment. However, a grid solution does not come to fruition by simply installing software to allocate resources on demand. Given that grid solutions are adaptable to meet the needs of various business problems, differing types of grids are designed to meet specific usage requirements and constraints. Additionally, differing topologies are designed to meet varying geographical constraints and network connectivity requirements. The success of a grid solution is heavily dependant on the amount of thought the IT architect puts into the solution design. Once the functional and non-functional requirements are known, the IT architect should readily be able to select the type of grid and the best topology required to satisfy the majority of the business requirements. When armed with this information, the high-level grid design will be easier to complete, and by leveraging the use of known grid types and topologies, articulating the solution design will require much less effort.

It is important to focus on starting small and to begin building the basic framework of the design. Rather than setting out to build the desired end state grid solution all at once, consider building the grid solution in a phased approach.

The milestone for the initial phase is to provide an *intragrid* solution, which is essentially a grid sandbox that supports a basic set of Grid services. This solution would support a single location built upon the core grid components, such as a security model, information services, workload management, and the host devices. As long as this model supports the same protocols and standards, this design can be expanded as needed.

The first step of the design process is to build a graphical representation of the grid components. The subsequent phases of the design will be focused on the next level of architecture. This phase of the design is a starting point for architects, technical managers, and executives to understand the overall structure of the architecture.

At a glance, the grid architecture design should offer the following:

_ The “blueprint” for the detailed conceptual design

_ The use of open standards prescribed by the grid framework

_ A multi-dimensional tiered and layered view of the grid infrastructure, which demonstrates the ability to logically partition grid resources so that their service consumption does not impact other grid locations

_ The middleware components and subsystems for a grid infrastructure integration

_ A design for communication to both business and technical personnel, for budget and planning purposes, and to provide application development an illustration of how the shared grid infrastructure will impact the middleware solution design

_ The distribution of applications and subsystems

_ A means for identifying the necessary technical, infrastructural, and other middleware components and subsystems for a grid infrastructure

10.1.1) Solution objectives

The design objectives provide a basic framework for building the grid infrastructure. The advantage of using design solution objectives is to start documenting certain areas that can affect the overall design. Within your design, you are going to need to make sure that the grid can provide a certain amount of security, availability, and performance. By documenting these different objectives or requirements, it will make your design a lot easier to follow. You will also be able to justify some of your decisions during the course of the design by being able to come back to certain objectives and making sure they were met. Once the design objectives have been defined, you can separate them into individual subsystems. This allows each design objective to be worked on in parallel, while at the same time providing a cohesiveness for the overall architecture. Once you have documented the core subsystems of the design, you can focus on the different requirements that your grid design will comprise.

When you start building the initial pieces of your design, you need to make sure that your solution objectives line up with the customer’s requirements. For a grid design, this is especially important, as there are not only the standard infrastructure components to consider, but specialized middleware and application integration issues as well. Making sure that your solution objectives satisfy your stated requirements will allow you to design a working grid.

Security

Within any networked environment, there is going to be some risk and exposure involved with the security of your infrastructure. Unless the computers are unplugged in a locked room, there is the potential that someone may bypass the security and get access to protected resources. Whether the weaknesses are exploited in the infrastructure, application, configuration, or administration, there is some level of risk.

Security objectives are put in place to help to reduce that risk to an acceptable level. While no design is 100 percent secure, the level of risk is reduced and controlled through the use of security controls. The goal of the security objectives is to examine the security requirements and implement the necessary tools and processes to reduce the risk involved.

The degree of security involved is based on the type of grid topology and the data the security will be protecting. The security requirements for a grid design within a bank will be completely different from those of an academic institution doing research. Whatever the security requirements may be, the security design objectives for the grid design need to be a central focus for the conceptual architecture.

Considering that the basic grid security model is based on PKI, it is imperative that the security components are designed and thought out carefully. While PKI has been around for a while, there are different components and necessary processes that should be identified. Rushing this process could lead to many problems in the future.

With the PKI architecture being the focus of the initial design, there are still areas that need attention. The infrastructure components (firewalls, IDS, anti-virus, and encryption) and the processes to manage these pieces are all part of the security objectives. Knowing which areas match up with your existing environment is the first step to robust security. The following bullet points are an example of some security questions that will be answered during the course of the design. The first three assume that the enterprise will provide its own certificate authority, which is not usually recommended:

_ Where will my CA be deployed and how will we manage it?

_ Do I have the necessary processes in place to administer my own CA?

_ What are the responsibilities for managing my own CA?

_ How will I administer security on the local servers?

_ Are my servers of a uniform build or common operating environment?
_ Do I have a consistent software build across critical grid infrastructure systems?
_ Which processes are running on my servers?
_ Will any existing applications conflict with or further expose my grid to any vulnerabilities?

Availability

Availability in its simplest terms commonly refers to the percentage of time that a site is up and servicing job requests. Determining how much availability should be built into the design is part of the availability objectives. This leads down the path of discovering how many potential single points of failure exist and how much redundancy should be built into the design. It is inevitable that some components will fail during a lifetime of usage, but this can be managed by using redundant components where possible.

Whenever you review various availability scenarios, there are always discussions about the amount of availability that is required. In this respect, a grid design is no different from any other infrastructure. A good start is to list the potential components within the design that should be resilient to failure. Once these components have been identified, you can seek out the specific availability options for those components. In the following examples, some different infrastructure options are described.

An important point that needs to be discussed is the availability of dynamic resources within a grid environment. Grid is not like a standard environment where resources are fixed and do not change regularly. Within grid environments, resources are constantly changing according to the membership and participation in the grid. When grid resources are active, they can register with information services within the grid to alert the system of their state. It is important to make sure that when you design your grid, you keep this in mind.

Besides the grid middleware components, the different infrastructure components will also require different levels of availability. Some components will be more critical than others, and it will be up to your design to make sure that you account for this. When going through the different availability requirements, make sure that you account for both the grid and infrastructure components. The following lists are some examples of availability resources that should be accounted for:

_ Grid middleware
– Workload management
– Grid directory and indexing service

– Security services

– Data storage

– Grid software clustering

_ Networks

– Load-balancing

– High-availability routing protocols

– Redundant and diverse network paths

_ Security

– Redundant firewalls

_ Datastore

– Mirroring

– Data replication

– Parallel processing

_ Systems management

– Backup and recovery

– LDAP replicas

- Alerts and monitoring to signal a failure within the environment
-

Every so often, different components necessary to the workflow process fail periodically and disrupt availability of the system. You can help mitigate the risk involved by eliminating the single points of failure within your environment through the use of redundant software or hardware components.

To give you a better idea of some different availability targets, the following list presents an example of the expected system availability in a whole year:

_ Normal commercial availability (single node): 99–99.5 percent, 87.6–43.8 hours of system down

_ High availability: 99.9 percent, 8.8 hours of system down

_ Fault resilient: 99.99 percent, 53 minutes of system down

_ Fault tolerant: 99.999 percent, 5 minutes of system down

_ Continuous processing: 100 percent, 0 minutes of system down

Keep in mind, however, that the redundancy that is added to the grid infrastructure will normally increase the costs within the infrastructure. It is up to the business to help justify the costs that would bring an environment from 99.9 percent availability per year up to 99.99 percent per year. While the difference in time between those two numbers is about eight hours, the costs associated may be too much to justify the increased availability.

Performance

The performance objective for a grid environment is to most efficiently utilize the various resources within the grid. Whether that includes spare CPU cycles, access to a federated databases, or application processing, it is up to you to match the performance goals of the business and design accordingly.

If your application can take advantage of multiple resources, you can design your grid to be broken up into smaller instances and have the work distributed throughout the grid. The goal is to take advantage of the grid as a whole in order to increase the performance of the application. Through intelligent workload management and scheduling, your application can take advantage of whatever resources within the grid are available. Part of the performance is based on the form of workload management to make sure that all resources within the grid are actively servicing jobs or requests within the grid.

10.2) Grid architecture models

There are different types of grid architectures to fit different types of business problems. Some grids are designed to take advantage of extra processing resources, whereas some grid architectures are designed to support collaboration between various organizations.

The type of grid selected is based primarily on the business problem that is being solved. Taking the goals of the business into consideration will help you choose the proper type of grid framework. A business that wants to tap into unused resources for calculating risk analysis within their corporate data center will have a much different design than a company that wants to open their distributed network to create a federated database with one or two of their main suppliers.

Such different types of grid applications will require different designs, based on their respective unique requirements. The selection of a specific grid type will have a direct impact on the grid solution design. Additionally, it should be mentioned that grid technologies are still evolving and tactical modifications to a grid reference architecture may be required to satisfy a particular business requirement.

10.2.1) Computational grid

A computational grid aggregates the processing power from a distributed collection of systems. A well-known example of a computational grid is the SETI@home grid. This type of grid is primarily comprised of low-powered computers with minimal application logic awareness and minimal storage capacity.

Rather than simply painting images of flying toasters, the idle cycles of the personal computers on the SETI@home grid are combined to create a computational grid used to analyze radio transmissions received from outer space in the "Search for Extra Terrestrial Intelligence." Most businesses interested in computational grids will likely have similar IT initiatives in common. While they probably will not want to search for

Extraterrestrials, there will likely be a business initiative to expand abilities and maximize the computer utilization of existing resources through aggregation and sharing. The business may require more computer capacity than is available. The business is interested in modifying specific vertical applications for parallel computing opportunities.

Additional uses for a computational grid include mathematical equations, derivatives, pricing, portfolio valuation, and simulation (especially risk measurement). Note that not all algorithms are able to leverage parallel processing, data intensive and high throughput computing, order and transaction processing, market information dissemination, and enterprise risk management.

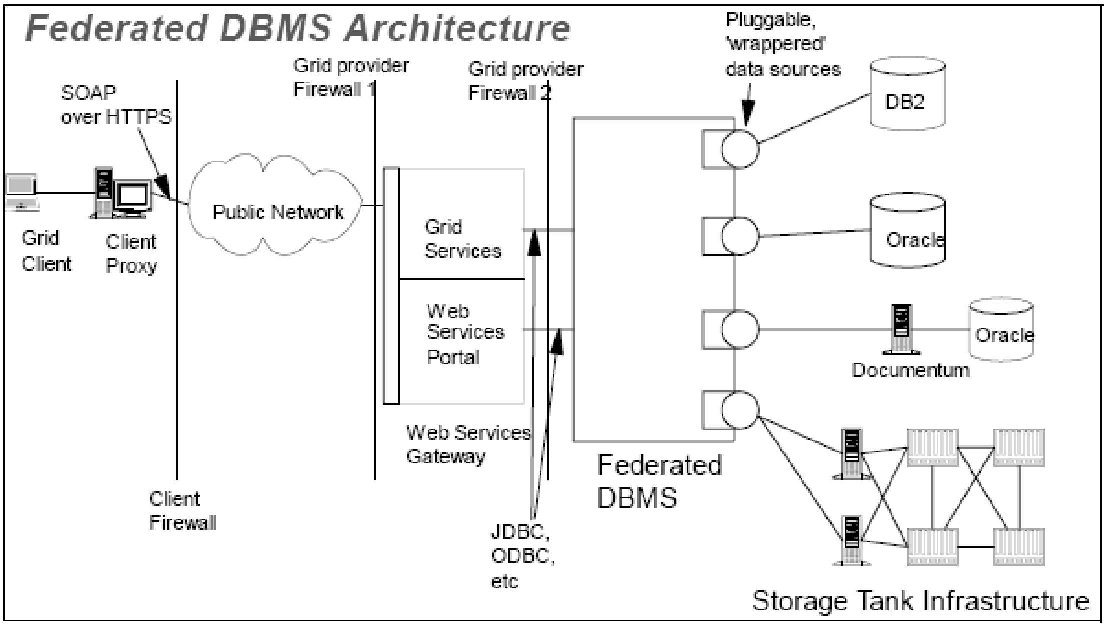
In many cases, the grid architecture model is not (yet) suitable for real-time applications. Computational grids can be recognized by these primary characteristics:

_ Made up of clusters of clusters
_ Enables CPU scavenging to better utilize resources
_ Provides the computational power to process large-scale jobs
_ Satisfies the business requirement for instant access to resources on demand

The primary benefits of computational grids are a reduced Total Cost of Ownership (TCO) and shorter deployment life cycles. Besides the SETI@home grid, the World Community Grid™, the Distributed Terascale Facility (TeraGrid), and the UK and Netherlands grids are all different examples of deployed computational grids. The next generation of computational grid computing will shift focus towards solving real-time computational problems.

10.2.2) Data grid

While computational grids are more suited for aggregating resources, data grids focus on providing secure access to distributed, heterogeneous pools of data. Through collaboration, data grids can also include resources such as a federated database. Within a federated database, a data grid makes a group of databases available that function as a single virtual database. Through this single interface, the federated database provides a single query point, data modeling, and data consistency. Data grids also harness data, storage, and network resources located in distinct administrative domains, respect local and global policies governing how data can be used, schedule resources efficiently (again subject to local and global Constraints), and provide high speed and reliable access to data. Businesses interested in data grids typically have IT initiatives to expand data-mining abilities while maximizing the utilization of an existing storage infrastructure investment, and to reduce the complexity of data management.



Federated DBMS architecture

10.3) Grid topologies

A topology view covers the following spectrum of grids:

_ **Intragrids**

– Single organizations

– No partner integration

– A single cluster

_ **Extragrids**

– Multiple organizations

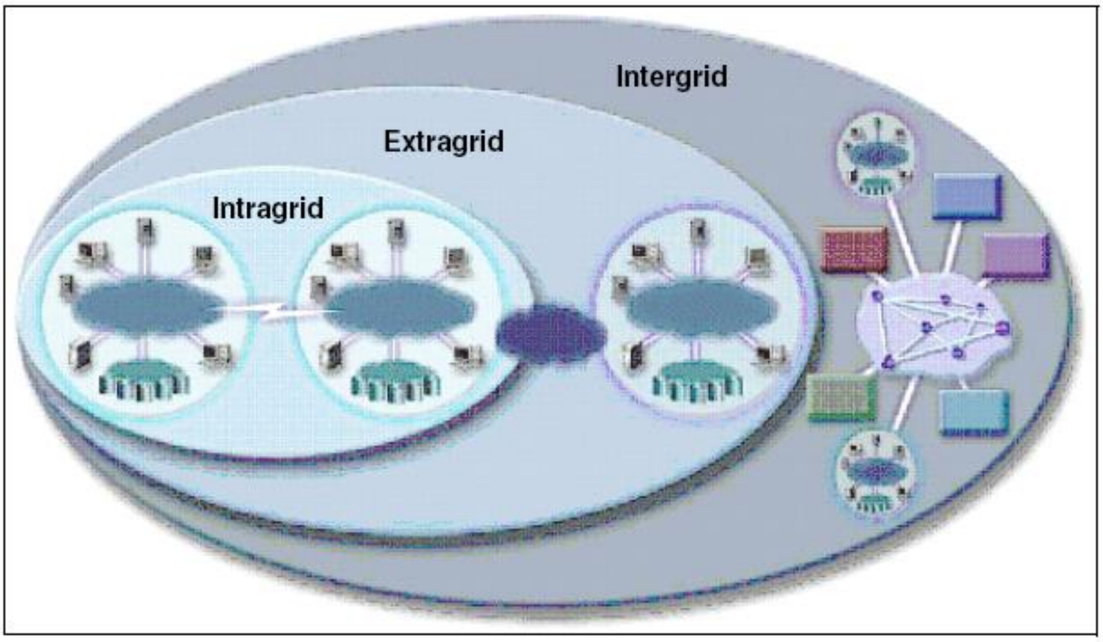
– Partner integration

– Multiple clusters

_ **Intergrids**

– Many organizations

- Multiple partners
- Many multiple clusters



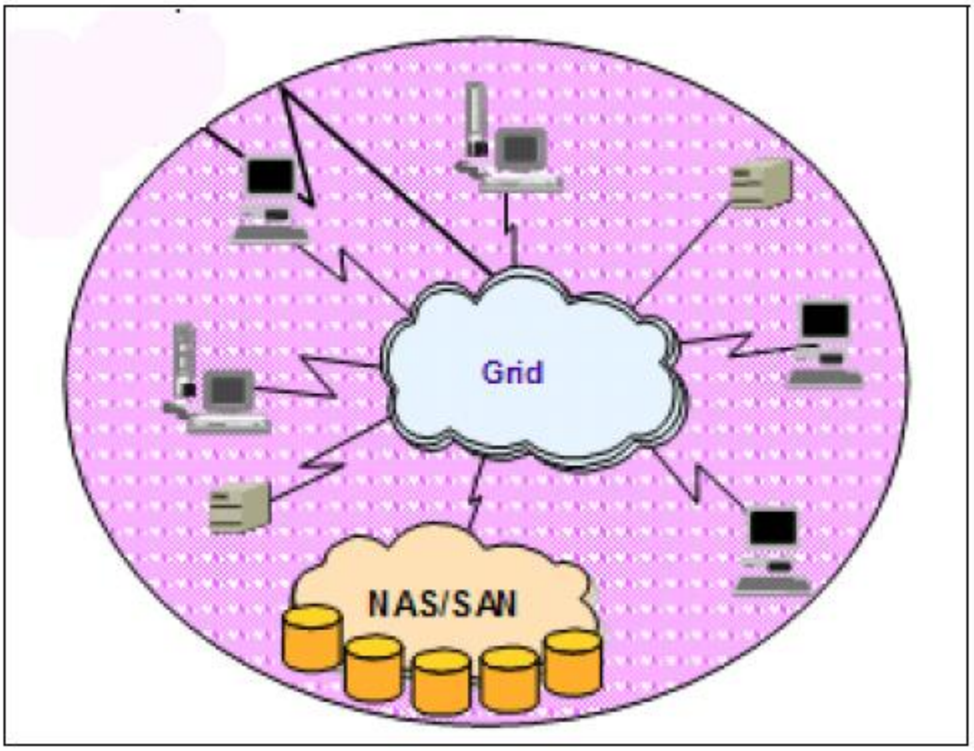
Intragrids, extragrids, and intergrids

The simplest of the three topologies is the intragrid, which is comprised merely of a basic set of Grid services within a single organization. The complexity of the grid design is proportionate to the number of organizations that the grid is designed to support, and the geographical parameters and constraints. As more organizations join the grid, the non-functional or operational requirements for security, directory services, availability, and performance become more complex.
As more organizations require access to grid resources, the requirements for increased application layer security, directory services integration, higher availability, and capacity become more complicated. The resource sharing alluded to is not primarily file exchange but rather direct access to computers, software, data, and other resources, as is required by a range of collaborative problem-solving and resource-brokering strategies emerging in industry, science, and engineering. This sharing is, necessarily, highly protected, with resource providers and consumers defining clearly and carefully just what is shared, who is allowed to share, and the conditions under which sharing occurs.

10.3.1) Intragrid

A typical intragrid topology, exists within a single organization, providing a basic set of Grid services. The single organization could be made up of a number of computers that share a common security domain, and share data internally on a private network. The primary characteristics of an intragrid are a single security provider, bandwidth on the private network is high and always available, and there is a single environment within a single network. Within an intragrid, it is easier to design and operate computational and data grids. An intragrid provides a relatively static set of

Computing resources and the ability to easily share data between grid systems. The business might deem an intragrid appropriate if the business has an initiative to gain economies of scale on internal job management, or wants to start exploring the use of a grid internally first by enabling vertical enterprise applications.



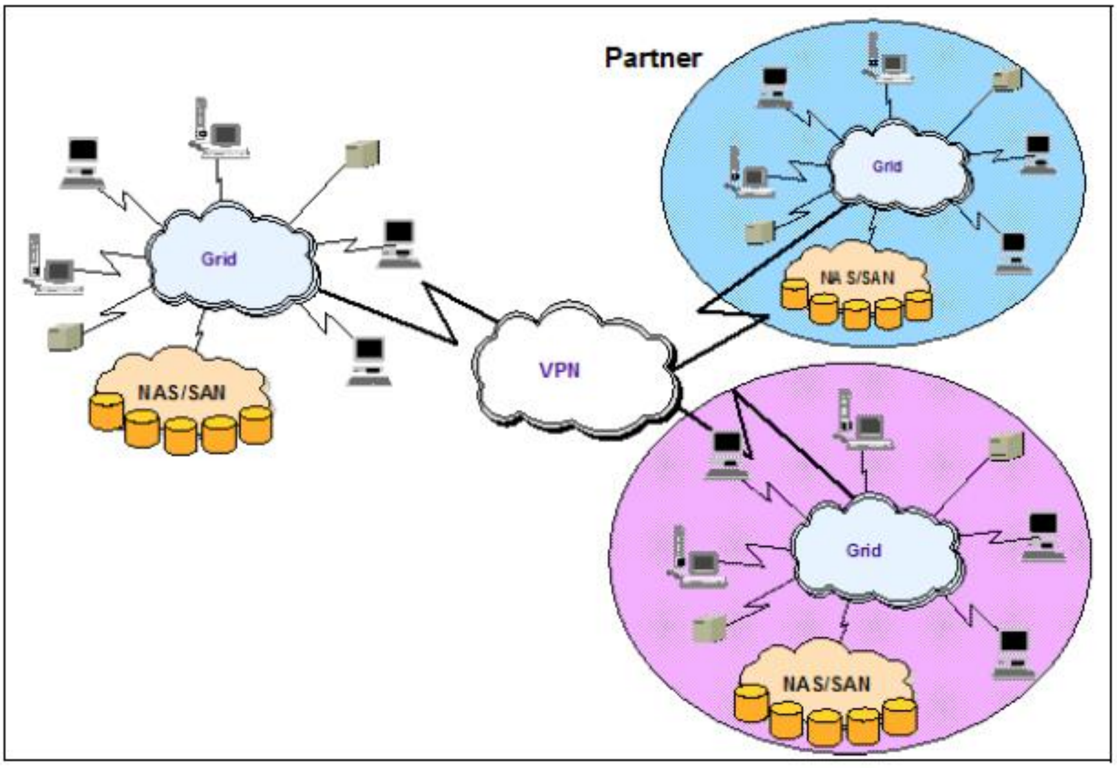
An INTRAGRID

10.3.2) Extragrid

Based on a single organization, the extragrid expands on the concept by bringing together two or more intragrids. An extragrid, typically involves more than one security provider, and the level of management complexity increases. The primary characteristics of an extragrid are dispersed security, multiple organizations, and remote/WAN connectivity.

Within an extragrid, the resources become more dynamic and your grid needs to be more reactive to failed resources and failed components. The design becomes more complicated and information services become relevant to ensure that grid resources have access to workload management at run time.

A business would benefit from an extragrid if there was a business initiative to integrate with external trusted business partners. An extragrid could also be used in a B2B capacity and/or to establish relationships of trust.

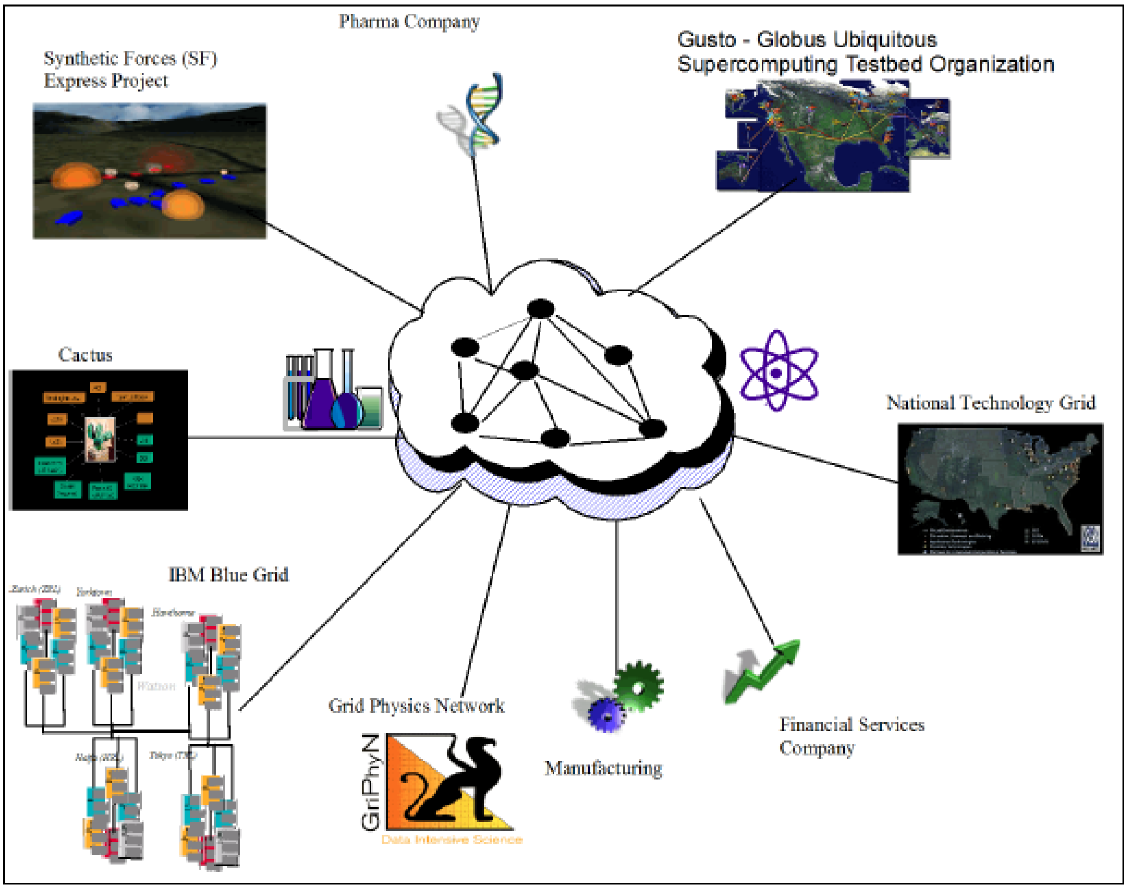


Extragrids can exist in several organizations and security providers

10.3.3) Intergrid

An intergrid requires the dynamic integration of applications, resources, and services with patterns, customers, and any other authorized organizations that will obtain access to the grid via the internet/WAN. An intergrid topology, is primarily used by engineering firms, life science industries, manufacturers, and by businesses in the financial industry.

The primary characteristics of an intergrid include dispersed security, multiple organizations, and remote/WAN connectivity. The data in an intergrid is global public data, and applications (both vertical and horizontal) must be modified for a global audience. A business may deem an intergrid necessary if there is a need for peer-to-peer computing, a collaborative computing community, or simplified end-to-end processes with the organizations that will use the intergrid.



An InterGrid

10.3.4) e-Utilities

One other type of grid that we should discuss is what we will call e-utility computing. Instead of having to buy and maintain the latest and best hardware and software, with this type of grid, customers will have the flexibility of tapping into computing power and programs as needed, just as they do gas or electricity. But enterprises are coming more and more to see the e-sourcing trend as a continuum—reaching beyond commonplace IT resources on demand to the delivery of business process and management functions integral to the way the organization works.

The *e-sourcing* business model is based on providing the components of IT function that are (largely) standardized and delivered through a service provider model. The attributes of this model include a distributed and shared environment, and generally standardized non-core business processes. The e-utility is used by consumers of the e-utility as building blocks for developing complex e-business solutions. The major properties of e-sourcing environments are a standard solution that requires minimal configuration; pooled resources used to serve multiple customers; capacity on demand; and scalable, 24x7, always on, high availability, rapidly deployable, minimal operations overhead; shared systems management; and flexible pricing and billing based on either actual usage/consumption of resources, or a calculated flat rate subscription.

10.4) Phases and activities

Deciding which grid type and topology to chose from is just the first step in the grid architecture design. A mature end-to-end design methodology is comprised of distinct phases and activities. The activities in the architecture design phase of the project include a review of the detailed architectural decisions and design documentation for the current infrastructure, conducting interviews and workshops, the modification of the initial high-level design based on new requirements and the results of the detailed assessment, the creation of a detailed modular architecture design, and the creation of the implementation and transition plan.

10.4.1) Basic methodology

For building a grid architecture, using a basic methodology allows the design to follow a consistent path from beginning to end. A methodology is not a cookbook for building a grid architecture, but a way to trace the progress of the design from the kickoff meeting to the final end state. The methodology follows a reproducible set of guidelines that can be used over again based on a set of successful guiding principals for architecture design. A methodology allows the architecture to follow a set of principals that can be documented from beginning to end throughout the design.

We define one such basic design methodology for developing the grid conceptual architecture in the next three sections.

Understanding the business drivers

The first step of any design is to identify and document the business drivers that are the foundation behind building the grid. The business drivers outline the investment and what the end state will accomplish. The business drivers or business strategy is the foundation or reasoning behind building the grid. Whether the goal is to tie together or build a federated database with your suppliers or tie together a set of computers to harness their overall processing power, you should have an end goal in mind before the design begins.

Requirements gathering

The requirements gathering process will help drive the architecture process by helping the technical team work within a set of guidelines for the architecture. By following this process, all of your decisions can be tied back to the basic requirements and business drivers for the design. Along with your solution objectives, the requirements will offer a road map for you to follow work through the design phases.

_ Business requirements

The business requirements are a subset of the business drivers that are focused on solving a specific business need. The business requirements drive important areas within the design, such as the performance and availability of the environment. Helping to understand these key service levels is an important part of the design.

_ Infrastructure requirements

The infrastructure requirements provide the basic framework for how the infrastructure will be designed. There are many different variables for how the grid architecture can be designed and, based on what the requirements will be, will shape how the environment will look.

_ Application requirements

There are many factors that need to be accounted for during the design, and the application is one of them. Possibly one of the most important requirements that must be validated is to ensure that the application in question can be made *grid-aware*. Unless the application can take advantage of the grid resources or split the workload across multiple components, the power of the grid is wasted.

Validate requirements

During the course of some designs, the requirements can change at the last minute or may go undiscovered. Requirements also have a way of changing when you least expect them to, so it is always a good idea to validate them before you proceed. Validating the requirements one last time before the design phase begins is a good way to ensure that all parties agree with the direction of the design.

10.4.2) Recommended steps

The following sections deal with additional recommended methods for developing an optimal grid design. These methods include attending grid design workshops and building prototypes once the design has been completed.

Grid design workshops

The purpose of the grid design workshops is to help all of the parties involved to better understand the variables, options, and considerations that need to be taken into account when developing a grid infrastructure design. Many or most of the grid middleware, technologies, and system components are probably new to many people within the design team and it is always a good idea to hear firsthand from experienced IT professionals the means by which grid infrastructures can be implemented, as well as any pitfalls to watch out for when designing environments for grid computing.

Documentation

An extremely critical means of communicating the design (your *solution*) of your grid infrastructure is via an architecture or solution document. The solution document should start with a high-level overview of the environment and subsequently should drill down into the most detailed configuration diagrams and descriptions possible. You will want to include things like IP addresses, network routes, server names, server architectures, network hardware, and essentially everything you know about the infrastructure at the time your design is completed. In truth, architecture documents are often dynamic, changing as the needs of the system users change and as technologies mature, become obsolete, and are replaced by newer technologies. You should revise your architecture document upon further hardware and software updates so that it accurately reflects the state of the system. Without an accurate architecture document, the system implementation team may get easily confused and not produce the system that was originally designed. Additionally, anyone adding further design changes to the system after the original system architect has moved on will appreciate an up-to-date architecture document, as it will save him or her countless hours of information gathering that would be necessary without an architecture document.

Prototype

Building a prototype of a grid system can save significant time that would otherwise be spent debugging and re-tooling unforeseen system incompatibilities. Your goal in building a prototype should be to produce a small-scale, end-to-end backbone of what your production environment will look like. It should include all interoperating technologies and/or architectures, so that if any incompatibility exists, it will be apparent before the production system is implemented. When all of the kinks are ironed out of your prototype, you will be confident that all of your components will work together properly in your designed infrastructure, and, additionally, you will have some experience in the implementation of such a system. Lessons learned from building the prototype should be reflected in your architecture document and any other directions provided to the implementation team.

10.5) A conceptual architecture

The purpose of the grid conceptual architecture is to establish a common understanding between the business owners and the people architecting and designing the grid infrastructure by describing the grid architecture that will support the client business requirements.

This section highlights some of the common components that you can choose from within the Globus Toolkit. If you are designing a grid architecture using different grid middleware software from Platform, DataSynapse, Avaki, or any other grid software provider, this section should still give you a head start on grid architecture. You will still be faced with decisions on the basic components, such as the security models, workload management, information services, and data sharing.

The conceptual model is a high-level framework consisting of the grid system components and nodes within the design. The nodes represent the different system components and grid middleware that make up the design. Normally, the conceptual model is the first graphical view of the grid infrastructure and is used as a stepping-stone to building a detailed configuration for the grid network. The graphic depiction of the grid environment will allow you to see how the requirements were gathered and how the many grid components will interact with one another.

10.5.1) Infrastructure

The infrastructure represents the physical hardware and software components used to interconnect different grid computers. These components help support the flow of information between grid systems and provide the basic set of services for connectivity, security, performance availability, and management. While many of these infrastructure components supply basic functionality to the grid, many are optional. It will be up to you to decide on the requirements and how well these components match up to the needs of your design.

Security

“Security” provides details about considerations related to security in a grid environment. Please refer to that chapter for more details on security.

One issue not addressed in detail in the chapter referenced above is the used of firewalls. The use of firewalls can provide logical and secure segmentation between grid systems. You might want to use firewalls to protect your networks and grid servers by limiting the types of services and protocols that connect to your computers. By using firewalls within your grid design, you can help limit the network communication between grid systems and only use protocols that you specify that the firewall will support.

Firewalls are not the only answer to protecting your grid servers, but they do add an additional layer of defense from internal or external users trying to access your systems. Firewalls work by controlling access to network services that your grid computers will be running. Since the network offers a gateway to your grid systems, you want to make sure that you control exactly the services and protocols that can be used to access your systems, as well as who can initiate communications.

For the most up-to-date information regarding the Globus Toolkit and firewalls, you should check out the firewall section on the Globus Web site at:

http://www.globus.org/security/

Some areas you may want to protect within your design are:

_ Certificate Authority/Registrant Authority

_ Globus Toolkit components, such as MDS, GRIS, and GIIS (For more information about these and other Globus Toolkit components "Components of Globus Toolkit")

_ Databases

_ All grid servers

Networks

The network design within the grid architecture can take on many different shapes. The networking components can represent the LAN or campus connectivity or even WAN communication between the grid networks. Whatever the case may be, the network's responsibility is to provide adequate bandwidth for any of the grid systems. Like many other components within the infrastructure, the networking can be customized to provide higher levels of availability, performance, or security.

Grid systems are for the most part network intensive due to security and other architectural limitations. For data grids in particular, which may have storage resources spread across the enterprise network, an infrastructure that is designed to handle a significant network load is critical to ensuring adequate performance.

Systems management

Any design will require a basic set of systems management tools to help determine availability and performance within the grid. A design without these tools is limited in how much support and information can be given about the health of the grid infrastructure. Some networks within a grid architecture can be dedicated to perform these functions as to not hamper the performance of the grid.

Storage

The storage possibilities are endless within a grid design. How that storage will be secured, backed up, managed, and replicated are some of the questions that the grid design will try to answer. Within a grid design, you want to make sure that your data is always available to the resources that need it. Besides availability, you want to make sure that your data is properly secured, as you would not want unauthorized access to sensitive data. Lastly, you want more than decent performance for access to your data. Obviously, some of this relies on the bandwidth and distance to the data, but you will not want any I/O problems to slow down your grid applications. For applications that are more disk-intensive, or for a data grid, more emphasis can be placed on storage resources, such as those providing higher capacity, redundancy, or fault-tolerance.

Summary

This section provided an overview of some of the key criteria and general methodologies that should be considered when designing a grid computing environment.

Part 3

Benefits of grid computing

Benefits of grid computing

When you deploy a grid, it will be to meet a set of business requirements. To better match grid computing capabilities to those requirements, it is useful to keep in mind some common motivations for using grid computing.

1) Exploiting under utilized resources

One of the basic uses of grid computing is to run an existing application on a different machine. The machine on which the application is normally run might be unusually busy due to a peak in activity. The job in question could be run on an idle machine elsewhere on the grid.

There are at least two prerequisites for this scenario. First, the application must be executable remotely and without undue overhead. Second, the remote machine must meet any special hardware, software, or resource requirements imposed by the application. For example, a batch job that spends a significant amount of time processing a set of input data to produce an output data set is perhaps the most ideal and simple use case for a grid. If the quantities of input and output are large, more thought and planning might be required to efficiently use the grid for such a job. It would usually not make sense to use a word processor remotely on a grid because there would probably be greater delays and more potential points of failure.

In most organizations, there are large amounts of under utilized computing resources. Most desktop machines are busy less than 5 percent of the time over a business day. In some organizations, even the server machines can often be relatively idle. Grid computing provides a framework for exploiting these under utilized resources and thus has the possibility of substantially increasing the efficiency of resource usage.

The processing resources are not the only ones that may be under utilized. Often, machines may have enormous unused disk drive capacity. Grid computing (more specifically, a *data grid*) can be used to aggregate this unused storage into a much larger virtual data store, possibly configured to achieve improved performance and reliability over that of any single machine. If a batch job needs to read a large amount of data, this data could be automatically replicated at various strategic points in the grid. Thus, if the job must be executed on a remote machine in the grid, the data is already there and does not need to be moved to that remote point. This offers clear performance benefits. Also, such copies of data can be used as backups when the primary copies are damaged or unavailable.

Another benefit of a grid is to better balance resource utilization. An organization may have occasional unexpected peaks of activity that demand more resources. If the applications are grid-enabled, they can be moved to under utilized machines during such peaks. In fact, some grid implementations can migrate partially completed jobs. In general, a grid can provide a consistent way to balance the loads on a wider federation of resources. This applies to CPU, storage, and any other types of resources that may be available on a grid.

2) Parallel CPU capacity

The potential for massive parallel CPU capacity is one of the most common visions and attractive features of a grid. In addition to pure scientific needs, such computing power is driving a new evolution in industries such as the bio-medical field, financial modeling, oil exploration, motion picture animation, and many others.

The common attribute among such uses is that the applications have been written to use algorithms that can be partitioned into independently running parts. A CPU-intensive grid application can be thought of as many smaller *subjobs*, each executing on a different machine in the grid. To the extent that these subjobs do not need to communicate with each other, the more *scalable* the application becomes. A perfectly scalable application will, for example, finish in one tenth of the time if it uses ten times the number of processors.

Barriers often exist to perfect scalability. The first barrier depends on the algorithms used for splitting the application among many CPUs. If the algorithm can only be split into a limited number of independently running parts, then that forms a scalability barrier. The second barrier appears if the parts are not completely independent; this can cause contention, which can limit scalability.

For example, if all of the subjobs need to read and write from one common file or database, the access limits of that file or database will become the limiting factor in the application’s scalability. Other sources of inter-job contention in a parallel grid application include message communications latencies among the jobs, network communication capacities, synchronization protocols, input-output bandwidth to storage or other devices, and other delays interfering with real-time requirements.

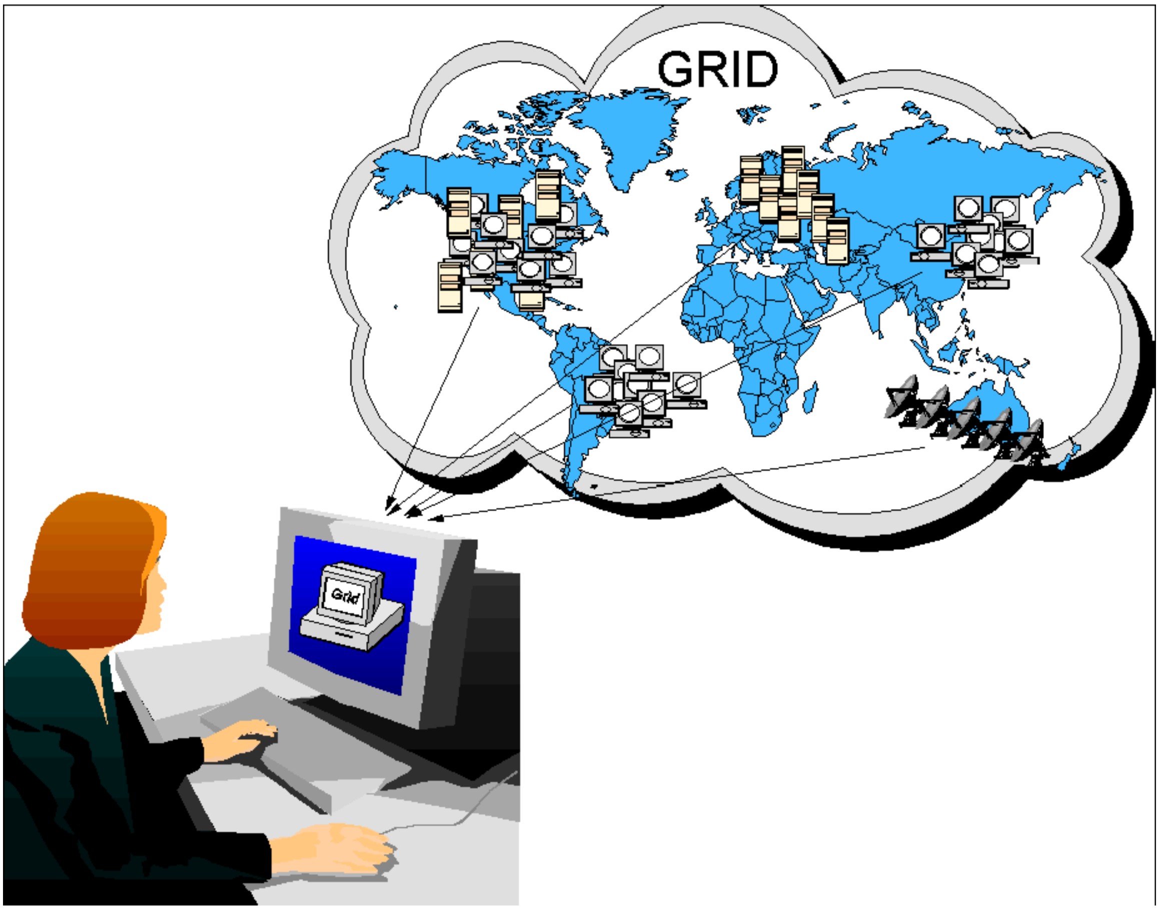
There are many factors to consider in grid-enabling an application. One must understand that not all applications can be transformed to run in parallel on a grid and achieve scalability. Furthermore, there are no practical tools for transforming arbitrary applications to exploit the parallel capabilities of a grid. There are some practical tools that skilled application designers can use to write a parallel grid application. However, automatic transformation of applications is a science in its infancy. This can be a difficult job and often requires mathematics and programming talents, if it is even possible in a given situation. New computation-intensive applications written today are being designed for parallel execution, and these will be easily grid-enabled, if they do not already follow emerging grid protocols and standards.

3) Virtual resources and virtual organizations for Collaboration

Another capability enabled by grid computing is to provide an environment for collaboration among a wider audience. In the past, distributed computing promised this collaboration and achieved it to some extent. Grid computing can take these capabilities to an even wider audience, while offering important standards that enable very heterogeneous systems to work together to form the image of a large virtual computing system offering a variety of resources. The users of the grid can be organized dynamically into a number of virtual organizations, each with different policy requirements. These virtual organizations can share their resources collectively as a larger grid. Sharing starts with data in the form of files or databases. A *data grid* can expand data capabilities in several ways. First, files or databases can span many systems and thus have larger capacities than on any single system. Such spanning can improve data transfer rates through the use of striping techniques.

Data can be duplicated throughout the grid to serve as a backup and can be hosted on or near the machines most likely to need the data, in conjunction with advanced scheduling techniques. Sharing is not limited to files, but also includes other resources, such as specialized devices, software, services, licenses, and so on. These resources are *virtualized* to give them a more uniform interoperability among heterogeneous grid participants.

The participants and users of the grid can be members of several real and virtual organizations. The grid can help in enforcing security rules among them and implement policies, which can resolve priorities for both resources and users.



4) Access to additional resources

As already stated, in addition to CPU and storage resources, a grid can provide access to other resources as well. The additional resources can be provided in additional numbers and/or capacity. For example, if a user needs to increase their total bandwidth to the Internet to implement a data mining search engine, the work can be split among grid machines that have independent connections to the Internet. In this way, total searching capability is multiplied, since each machine has a separate connection to the Internet. If the machines had shared the connection to the Internet, there would not have been an effective increase in bandwidth.

Some machines may have expensive licensed software installed that users require. Users' jobs can be sent to such machines, more fully exploiting the software licenses.

Some machines on the grid may have special devices. Most of us have used remote printers, perhaps with advanced color capabilities or faster speeds. Similarly, a grid can be used to make use of other special equipment. For example, a machine may have a high speed, self-feeding DVD writer that could be used to publish a quantity of data faster. Some machines on the grid may be connected to scanning electron microscopes that can be operated remotely. In this case, scheduling and reservation are important. A specimen could be sent in advance to the facility hosting the microscope. Then the user can remotely operate the machine, changing perspective views until the desired image is captured. The grid can enable more elaborate access, potentially to remote medical diagnostic and robotic surgery tools with two-way interaction from a distance.

The variations are limited only by one's imagination. Today, we have remote device drivers for printers. Eventually, we will see standards for grid-enabled device drivers to many unusual devices and resources. All of these will make the grid look like a large system with a collection of resources beyond what would be available on just one conventional machine.

5) Resource balancing

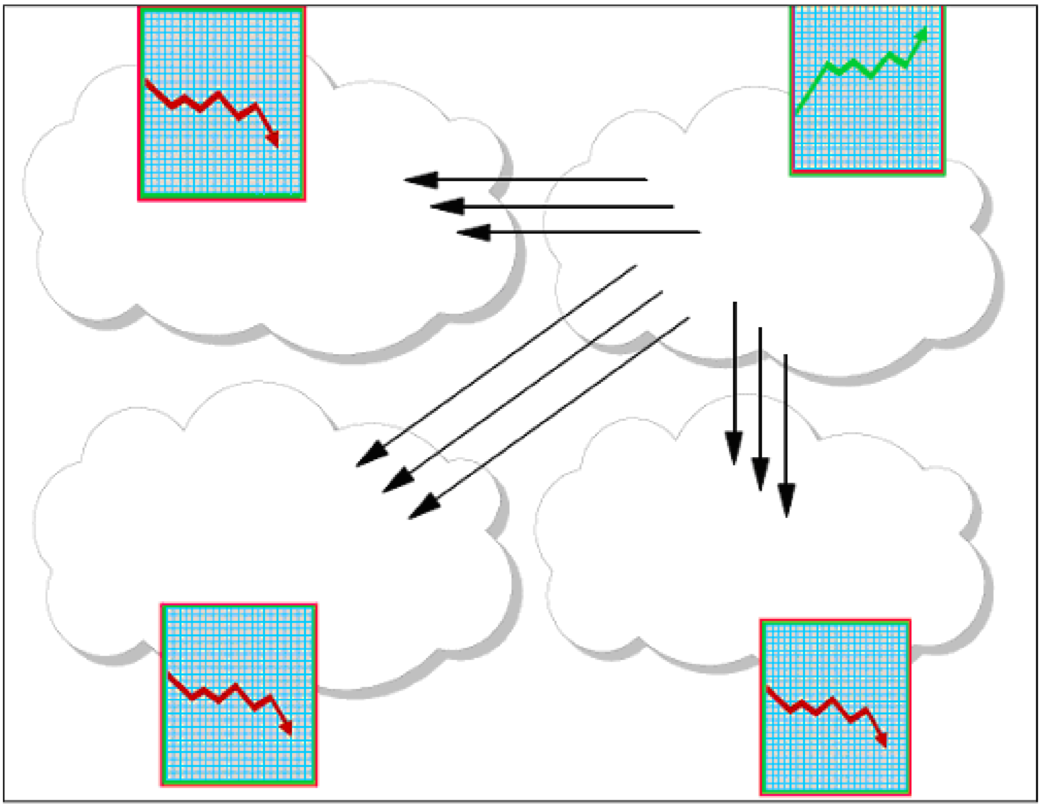
A grid federates a large number of resources contributed by individual machines into a large single-system image. For applications that are grid-enabled, the grid can offer a resource balancing effect by scheduling grid jobs on machines with low utilization This feature can prove invaluable for handling occasional peak loads of activity in parts of a larger organization. This can happen in two ways:

_ an unexpected peak can be routed to relatively idle machines in the grid.

_ if the grid is already fully utilized, the lowest priority work being performed on the grid can be temporarily suspended or even cancelled and performed again later to make room for the higher priority work.

Without a grid infrastructure, such balancing decisions are difficult to prioritize and execute.

Occasionally, a project may suddenly rise in importance with a specific deadline. A grid cannot perform a miracle and achieve a deadline when it is already too close. However, if the size of the job is known, if it is a kind of job that can be sufficiently split into subjobs, and if enough resources are available after preempting lower priority work, a grid can bring a very large amount of processing power to solve the problem.

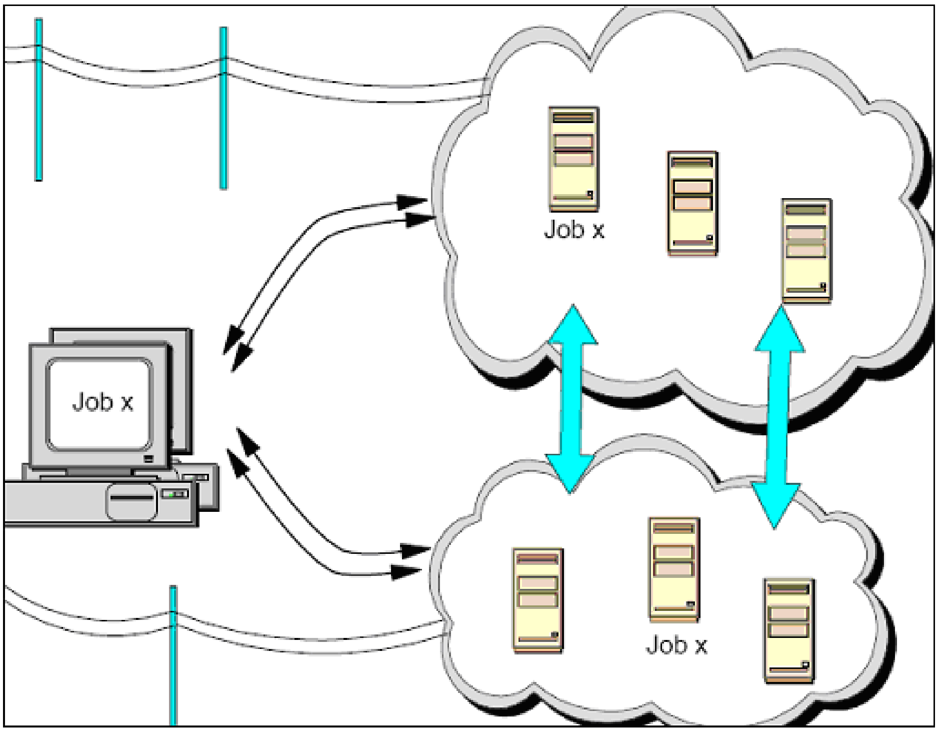


Other more subtle benefits can occur using a grid for load balancing. When jobs communicate with each other, the Internet, or with storage resources, an advanced scheduler could schedule them to minimize communications traffic or minimize the distance of the communications. This can potentially reduce communication and other forms of contention in the grid. Finally, a grid provides excellent infrastructure for brokering resources. Individual resources can be profiled to determine their availability and their capacity, and this can be factored into scheduling on the grid. Depending on the accounting facilities in place, different organizations participating in the grid can build up grid credits and use them at times when they need additional resources. This can form the basis for grid accounting and the ability to more fairly distribute work (and cost) on the grid.

6) Reliability

High-end conventional computing systems use expensive hardware to increase reliability. They are built using chips with redundant circuits that vote on results, and contain logic to achieve graceful recovery from an assortment of hardware failures. The machines also use duplicate processors with hot pluggability so that when they fail, one can be replaced without turning the other off. Power supplies and cooling systems are duplicated. The systems are operated on special power sources that can start generators if utility power is interrupted. All of this builds a reliable system, but at a great cost, due to the duplication of expensive components.

In the future, we will see a complementary approach to reliability that relies on software and hardware. A grid is just the beginning of such technology. The systems in a grid can be relatively inexpensive and geographically dispersed. Thus, if there is a power or other kind of failure at one location, the other parts of the grid are not likely to be affected. Grid management software can automatically resubmit jobs to other machines on the grid when a failure is detected. In critical, real-time situations, multiple copies of important jobs can be run on different machines throughout the grid. Their results can be checked for any kind of inconsistency, such as computer failures, data corruption, or tampering.



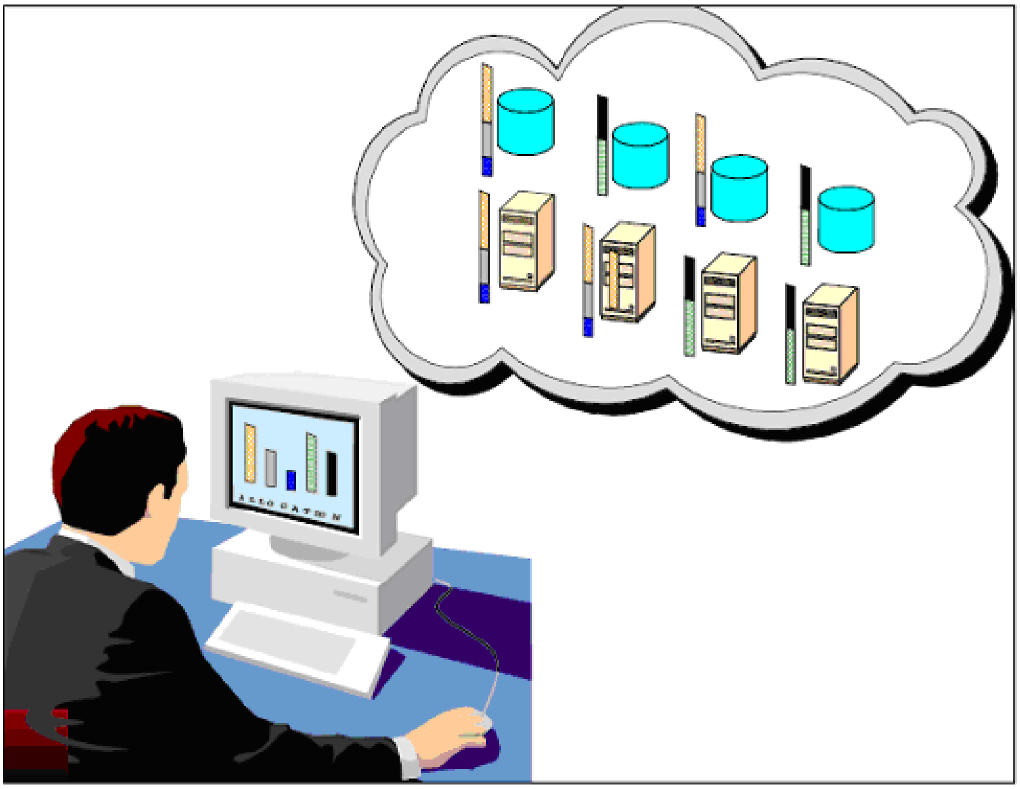
Such grid systems will utilize *autonomic computing*. This is a type of software that automatically heals problems in the grid, perhaps even before an operator or manager is aware of them. In principle, most of the reliability attributes achieved using hardware in today's high availability systems can be achieved using software in a grid setting in the future.

7) Management

The goal to virtualizes the resources on the grid and more uniformly handle heterogeneous systems will create new opportunities to better manage a larger, more distributed IT infrastructure. It will be easier to visualize capacity and utilization, making it easier for IT departments to control expenditures for computing resources over a larger organization.

The grid offers management of priorities among different projects. In the past, each project may have been responsible for its own IT resources and the associated expenses. Often these resources might be under utilized while another project finds itself in trouble, needing more resources due to unexpected events. With the larger view a grid can offer, it becomes easier to control and manage such situations. As illustrated in Figure 2-4, administrators can change any number of policies that affect how the different organizations might share or compete for resources. Aggregating utilization data over a larger set of projects can enhance an organization's ability to project future upgrade needs. When maintenance is required, grid work can be rerouted to other machines without crippling the projects involved.

Autonomic computing can come into play here too. Various tools may be able to identify important trends throughout the grid, informing management of those that require attention.



Grid computing enables organizations (real and virtual) to take advantage of various computing resources in ways not previously possible. They can take advantage of under utilized resources to meet business requirements while minimizing additional costs. The nature of a computing grid allows organizations to take advantage of parallel processing, making many applications financially feasible as well as allowing them to complete sooner. Grid computing makes more resources available to more people and organizations while allowing those responsible for the IT infrastructure to enhance resource balancing, reliability, and manageability.