Application control :
Boundary control :
● **Access Controls:** These controls restrict use of computer system resources to authorized users, limit the actions authorized users can taker with these resources, and ensure that users obtain only authentic computer system resources.
An access control mechanism processes users' request for resources in three steps:
▪ **Identification:** First users identify themselves to the mechanism, thereby indicating their intent to request system resources
▪ **Authentication:** It is a two way process wherein users must authenticate themselves, and the mechanism in turn must authenticate itself.
▪ **Authorization:** Users must request specific resources and specify the actions they intend to take with the resources.
● **Digital Signatures:** In computer system, Digital Signatures establish the authenticity of persons and prevent the denial of messages or contracts when data is exchanged electronically

Communication Controls: Components in the communication subsystem are responsible for transporting data among all the other subsystems within a system and for transporting data to or receiving data from another system. Three types of exposure arise in the communication subsystem.
(a) As data is transported across a communication subsystem, it can be impaired through attenuation, and noise.
(b) The hardware and software components in a communication subsystem can fail.
(c) The communication subsystem can be subjected to passive or active subversive attacks.

● **Physical Component Controls:** One way to reduce expected losses in the communication subsystem is to choose physical component that have characteristics that make them reliable These controls involve Transmission Media - Bounded (Guided) Media or Unbounded (Unguided) Media; Communication Lines – Private (Leased) or Public; Modems; Port Protection Devices; Multiplexors .
● **Line Error Controls:** Whenever data is transmitted over a communication line, it can be received in error because of attenuation, distortion, or noise that occurs on the line. Error Detection (using Parity Checking, Loop Check) and Error Correction (using forward Error Correcting Codes and Backward Error Correction) are the two major approaches under Line Error Controls.
● **Flow Controls:** These are needed because two nodes in a network can differ in terms of the rate at which they can send receive and process data. The simplest form of flow control is "Stop-and-Wait Flow Control" in which the sender transmits a frame of data only when the receiver is ready to accept the frame.
● **Link Controls:** This involves two common protocols – HDLC (Higher Level Data Control) and SDLC (Synchronous Data Link Control); the study of these is beyond the scope of this book.
● **Topological Controls:** A communication network topology specifies the location of nodes within a network, the ways in which these nodes will be linked. Some of the four basic topologies include Bus, Ring, Star and Tree Topology.
● **Channel Access Controls:** Two different nodes in a network can compete to use a communication channel. Whenever the possibility of contention for the channel exists, some type of channel access control technique must be used. These techniques fall into two classes – Polling methods and Contention methods. Polling techniques establish an order in which a node can gain access to channel capacity; whereas in Contention methods, nodes in a network must compete with each other to gain access to a channel.

● **Internetworking Controls:** Internetworking is the process of connecting two or more communication networks together to allow the users of one network to communicate with the users of other networks. Three types of devices are used to connect sub-networks in an Internet: Bridge, Router and Gateway.

Some common types of Virtualization :
Hardware Virtualization: Hardware Virtualization or Platform Virtualization refers to the creation of a virtual machine that acts like a real computer with an operating system. Software executed on these virtual machines is separated from the underlying hardware resources. For example, a computer that is running Microsoft Windows may host a virtual machine that looks like a computer with the Linux operating system; based software that can be run on the virtual machine.

Network Virtualization: Network virtualization is a method of combining the available resources in a network by splitting up the available bandwidth into channels, each of which is independent from the others, and each of which can be assigned (or reassigned) to a particular server or device in real time. Various equipment and software vendors offer network virtualization by combining any of the Network hardware such as switches and network interface cards (NICs); Network virtualization is intended to optimize network speed, reliability, flexibility, scalability, and security.

Storage Virtualization: Storage virtualization is the apparent pooling of data from multiple storage devices, even different types of storage devices, into what appears to be a single device that is managed from a central console. Storage virtualization helps the storage administrator perform the tasks of backup, archiving, and recovery more easily.

Grid Computing :
Grid Computing is a computer network in which each computer's resources are shared with every other computer in the system. It is a distributed architecture of large numbers of computers connected to solve a complex problem. In the grid computing model, servers or personal computers run independent tasks and are linked by the Internet or low-speed networks.
A typical Grid Model is shown in Fig. 5.7.1.
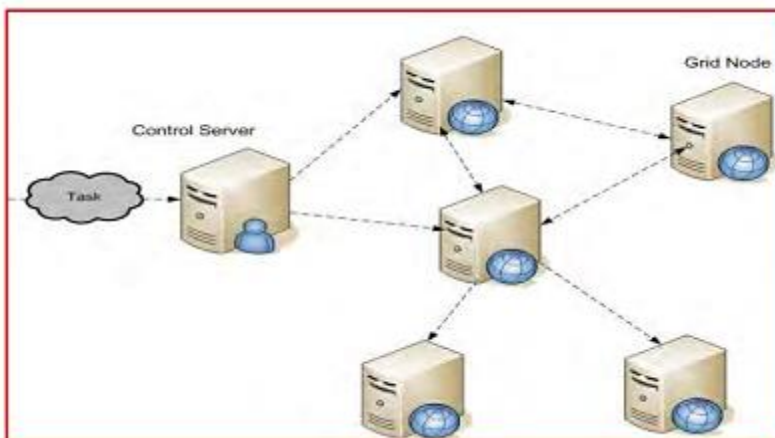
(i) Benefits of Grid Computing :

• **Making use of Underutilized Resources**: In most organizations, there are large amounts of underutilized computing resources. In some organizations, even the server machines can often be relatively idle. Grid computing provides a framework for exploiting these underutilized resources and thus has the possibility of substantially increasing the efficiency of resource usage.

• **Resource Balancing**: For applications that are grid-enabled, the grid can offer a resource balancing effect by scheduling grid jobs on machines with low utilization. An unexpected peak can be routed to relatively idle machines in the grid; and 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.

• **Parallel CPU Capacity**: The potential for usage of massive parallel CPU capacity is one of the most common visions and attractive features of a grid. A CPU-intensive grid application can be thought of as many smaller sub-jobs, each executing on a different machine in the grid.

• **Virtual resources and virtual organizations for collaboration**: Another capability enabled by grid computing is to provide an environment for collaboration among a wider audience. 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 such as data, specialized devices, software, services, licenses, and so on, collectively as a larger grid.

• **Access to additional resources**: In addition to CPU and storage resources, a grid can provide access to other resources as well. 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.

• **Reliability:** High-end conventional computing systems use expensive hardware to increase reliability. The machines also use duplicate processors in such a way 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.

• **Management:** The goal to virtualizes the resources on the grid and more uniformly handle heterogeneous systems create new opportunities to better manage a larger, more distributed IT infrastructure. Aggregating utilization data over a larger set of projects can enhance an organization's ability to project future upgrade needs.

(ii) Types of Resources :

A grid is a collection of machines, sometimes referred to as nodes, resources, clients, hosts and many other such terms. Some resources may be used by all users of the grid, while others may have specific restrictions.

• **Computation:** The most common resource is Computing Cycles provided by the processors of the machines on the grid where processors can vary in speed, architecture, software platform, and other associated factors such as memory, storage, and connectivity. There are three primary ways to exploit the computation resources of a grid.

o To run an existing application on an available machines on the grid rather than locally;

o To use an application designed to split its work in such a way that the separate parts can execute in parallel on different processors; and

o To run an application, that needs to be executed many times,

● **Storage:** The second most common resource used in a grid is Data Storage. **A grid providing an integrated view of data storage is sometimes called a Data Grid.** Each machine on the grid usually provides some quantity of storage for grid use, even if temporary. Storage can be memory attached to the processor or it can be secondary storage, using hard disk drives or other permanent storage media.

● **Communications:** Communications within the grid are important for sending jobs and their required data to points within the grid. The bandwidth available for such communications can often be a critical resource that can limit utilization of the grid. In some cases, higher speed networks must be provided to meet the demands of jobs transferring larger amounts of data.

● **Software and Licenses:** The grid may have software installed that may be too expensive to install on every grid machine. Some software licensing arrangements permit the software to be installed on all of the machines of a grid but may limit the number of installations that can be simultaneously used at any given instant.

● **Special equipment, capacities, architectures, and policies:** Platforms on the grid will often have different architectures, operating systems, devices, capacities, and equipment. Each of these items represents a different kind of resource that the grid can use as criteria for assigning jobs to machines.

(iii) Using a Grid: User's Perspective
● **Enrolling and installing Grid Software:** A user may first have to enroll his machine as a donor on the grid and install the provided grid software on his own machine that may require authentication for security purposes. The user positively establishes his identity with a Certificate Authority who must take steps to assure that the user is in fact who he claims to be. Once the user and/or machine are authenticated, the grid software is provided to the user for installing on his machine for the purposes of using the grid as well as donating to the grid.
● **Logging onto the grid:** Most grid systems require the user to log on to a system using an ID that is enrolled in the grid. Once logged on, the user can query the grid and submit jobs.
● **Queries and submitting jobs:** Grid systems usually provide command-line tools as well as graphical user interfaces (GUIs) for queries. Job submission usually consists of three parts.
o First, some input data and executable program are sent to the machine to execute the job.
o Second, the job is executed on the grid machine. The grid software running on the donating machine executes the program on the user's behalf.
o Third, the results of the job are sent back to the submitter.

● **Data configuration:** The data accessed by the grid jobs may simply be staged in and out by the grid system. However, depending on its size and the number of jobs, this can potentially add up to a large amount of data traffic. This is preferable to using a networked file system to share this data, because in such a file system, the data would be effectively moved from a central location every time the application is run. This type of analysis is necessary for large jobs to better utilize the grid and not create unnecessary bottlenecks.
● **Monitoring progress and recovery:** The user can query the grid system to see how his application and its sub-jobs are progressing. A grid system, in conjunction with its job scheduler, often provides some degree of recovery for sub-jobs that fail. A job may fail due to a Programming error, Hardware or power failure, Communications interruption etc.
● **Reserving resources:** To improve the quality of a service, the user may arrange to reserve a set of resources in advance for his exclusive or high priority use.

(iv) Using a Grid: An Administrative Perspective
● **Planning:** The administrator should understand the organization's requirements for the grid to better choose the grid technologies that satisfy grid's requirements. One of the first considerations is the hardware available and how it is connected via a LAN or WAN. Next, an organization may want to add additional hardware to supplement the capabilities of the grid.

o **Security:** Security is a much more important factor in planning and maintaining a grid where data sharing comprises the bulk of the activity. In a grid, the member machines are configured to execute programs rather than just move data. This makes an unsecured grid potentially fertile ground for viruses and Trojan horse programs. For this reason, it is important to understand the issues involved in authenticating users and providing proper authorization for specific operations.
o **Organization:** It is important to understand how the departments in an organization interact, operate, and contribute to the whole.

● **Installation:** First, the selected grid system must be installed on an appropriately configured set of machines. These machines should be connected using networks with sufficient bandwidth to other machines on the grid.

● **Managing enrollment of donors and users:** The administrator is responsible for controlling the rights of the users in the grid. Donor machines may have access rights that require management as well.. The rights of these grid user IDs must be properly set so that grid jobs do not allow access to parts of the donor machine to which the users are not entitled.

● **Certificate Authority:** It is critical to ensure the highest levels of security in a grid because the grid is designed to execute code and not just share data. Thus, viruses, Trojan horses, and other attacks cane affect the grid system. The Certificate Authority is one of the most important aspects of maintaining strong grid security. An organization may choose to use an external Certificate Authority or operate one itself. **The primary responsibilities of a Certificate Authority are:**
o Positively identifying entities requesting certificates;
o Issuing, removing, and archiving certificates;
o Protecting the Certificate Authority server;
o Maintaining a namespace of unique names for certificate owners;
o Serving signed certificates to those needing to authenticate entities; and
o Logging activity.

● **Resource Management:** Another responsibility of the administrator is to manage the resources of the grid. This includes setting permissions for grid users to use the resources as well as tracking resource usage and implementing a corresponding accounting or billing system. Usage statistics are useful in identifying trends in an organization that may require the acquisition of additional hardware; reduction in excess hardware to reduce costs etc.
● **Data sharing:** For small grids, the sharing of data can be fairly easy, using existing networked file systems, databases, or standard data transfer protocols. As a grid grows and the users become dependent on any of the data storage repositories, the administrator should consider procedures to maintain backup copies and replicas to improve performance.

(v) Application Areas of Grid Computing

• Civil engineers collaborate to design, execute, & analyze shake table experiments.

• An insurance company mines data from partner hospitals for fraud detection.

• An application service provider offloads excess load to a compute cycle provider.

• An enterprise configures internal & external resources to support e-Business workload.

• Large-scale science and engineering are done through the interaction of people, heterogeneous computing resources, information systems and instruments, all of which are geographically and organizationally dispersed.

(vi) Grid Computing Security

Grid systems and applications require standard security functions which are Authentication, Access Control, Integrity, Privacy, and No Repudiation.

To develop security architecture, following constraints are taken from the characteristics of grid environment and application.

Single Sign-on: A user should authenticate once and they should be able to acquire resources, use them, and release them and to communicate internally without any further authentication.

Protection of Credentials: User passwords, private keys, etc. should be protected.

Interoperability with local security solutions: Access to local resources should have local security policy at a local level. Despite of modifying every local resource there is an inter-domain security server for providing security to local resource.

● **Exportability:** The code should be exportable i.e. they cannot use a large amount of encryption at a time.

● **Support for secure group communication:** In a communication there are number of processes which coordinate their activities. This coordination must be secure .

● **Support for multiple implementations:** There should be a security policy which should provide security to multiple sources based on public and private key cryptography.