SYSTEM MODELS FOR DISTRIBUTED AND CLOUD COMPUTING

By: Neelam Singh

Introduction

- Distributed and cloud computing systems are built over a large number of autonomous computer nodes.
- These node machines are interconnected by SANs, LANs, or WANs in a hierarchical manner.
- With today's networking technology, a few LAN switches can easily connect hundreds of machines as a working cluster.
- A WAN can connect many local clusters to form a very large cluster of clusters.
- Massive systems are considered highly scalable, and can reach web-scale connectivity, either physically or logically.
- These can be classified into 4 groups: clusters, peer-to-peer networks, grids, and clouds.

Classification

Classification of Parallel and Distributed Computing Systems

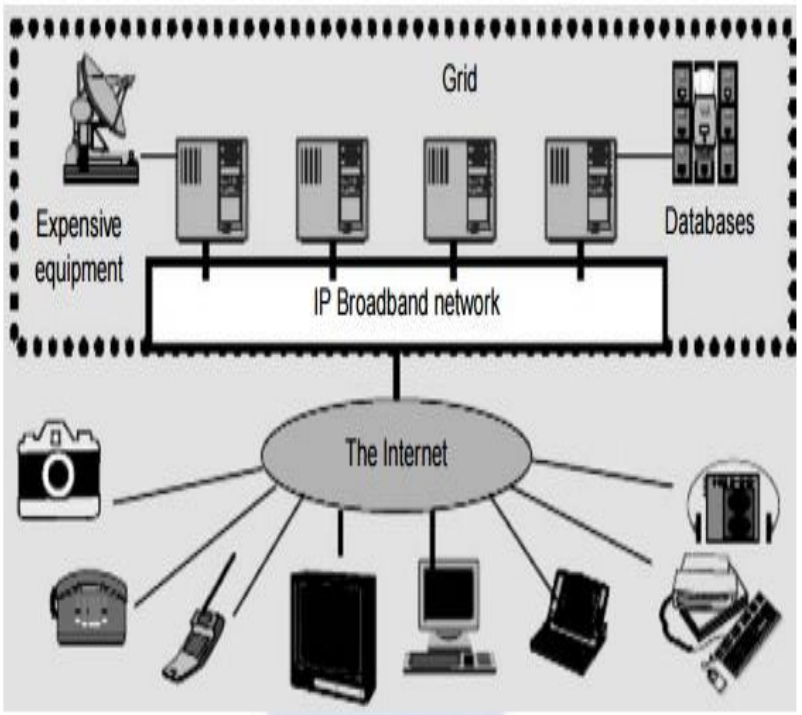
Functionality, Applications	Computer Clusters [10,28,38]	Peer-to-Peer Networks [34,46]	Data/ Computational Grids [6,18,51]	Cloud Platforms [1,9,11,12,30]
Architecture, Network Connectivity, and Size	Network of compute nodes interconnected by SAN, LAN, or WAN hierarchically	Flexible network of client machines logically connected by an overlay network	Heterogeneous clusters interconnected by high-speed network links over selected resource sites	Virtualized cluster of servers over data centers via SLA
Control and Resources Management	Homogeneous nodes with distributed control, running UNIX or Linux	Autonomous client nodes, free in and out, with self-organization	Centralized control, server-oriented with authenticated security	Dynamic resource provisioning of servers, storage, and networks
Applications and Network-centric Services	High-performance computing, search engines, and web services, etc.	Most appealing to business file sharing, content delivery, and social networking	Distributed supercomputing, global problem solving, and data center services	Upgraded web search, utility computing, and outsourced computing services
Representative Operational Systems	Google search engine, SunBlade, IBM Road Runner, Cray XT4, etc.	Gnutella, eMule, BitTorrent, Napster, KaZaA, Skype, JXTA	TeraGrid, GriPhyN, UK EGEE, D-Grid, ChinaGrid, etc.	Google App Engine, IBM Bluecloud, AWS, and Microsoft Azure

Peer-to-peer (P2P) Networks

- In a P2P network, every node (peer) acts as both a client and server.
- Peers act autonomously to join or leave the network. No central coordination or central database is needed.
- No peer machine has a global view of the entire P2P system. The system is self-organizing with distributed control. Unlike the cluster or grid, a P2P network does not use dedicated interconnection network.
- P2P Networks are classified into different groups:
  - Distributed File Sharing: content distribution of MP3 music, video, etc. E.g. Gnutella, Napster, BitTorrent.
  - Collaboration P2P networks: Skype chatting, instant messaging, gaming etc.
  - Distributed P2P computing: specific application computing such as SETI@home provides 25 Tflops of distributed computing power over 3 million Internet host machines.

Computational and Data Grids

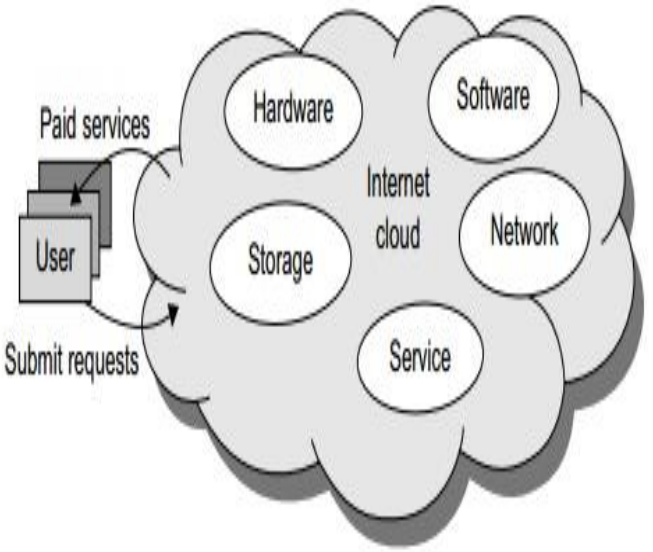
- Grid technology demands new distributed computing models, software/middleware support, network protocols, and hardware infrastructures.
- Grids are heterogeneous clusters interconnected by high-speed networks. They have centralized control, are server-oriented with authenticated security.
- They are suited to distribute supercomputing. E.g. TeraGrid. Like an electric utility power grid, a computing grid offers an infrastructure that couples computers, software/middleware, people, and sensors together.
- The grid is constructed across LANs, WANs, or Internet backbones at a regional, national, or global scale.
- The computers used in a grid include servers, clusters, and supercomputers. PCs, laptops, and mobile devices can be used to access a grid system.



Computational grid or data grid providing computing utility, data, and information services through resource sharing and cooperation among participating organizations.

Cloud

- A Cloud is a pool of virtualized computer resources. A cloud can host a variety of different workloads, including batch-style backend jobs and interactive and user-facing applications.
- Workloads can be deployed and scaled out quickly through rapid provisioning of VMs.
- Virtualization of server resources has enabled cost effectiveness and allowed cloud systems to leverage low costs to benefit both users and providers.
- Cloud system should be able to monitor resource usage in real time to enable rebalancing of allocations when needed.
- Cloud computing applies a virtualized platform with elastic resources on demand by provisioning hardware, software, and data sets dynamically.
- Desktop computing is moved to a service-oriented platform using server clusters and huge databases at datacenters.
- **Advantage of Clouds over Traditional Distributed Systems:** Traditional distributed computing systems provided for on-premise computing and were owned and operated by autonomous administrative domains (e.g. a company).
  - These traditional systems encountered performance bottlenecks, constant system maintenance, poor server (and other resource) utilization, and increasing costs associated with hardware/software upgrades.
  - Cloud computing as an on-demand computing paradigm resolves or relieves many of these problems.



Virtualized resources from data centers to form an Internet cloud, provisioned with hardware, software, storage, network, and services for paid users to run their applications.

Service-Oriented Architecture (SOA)Layered Architecture

- In web services, Java RMI, and CORBA, an entity is, respectively, a service, a Java remote object, And a CORBA object.
- These build on the TCP/IP network stack.
- On top of the network stack we have a base software environment, which would be .NET/Apache Axis for web Services, JVM for Java, and the ORB network for CORBA.
- On top of this base environment, a higher level environment with features specific to The distributed computing environment is built.
- Loose coupling and support of heterogeneous implementations make services more attractive than distributed objects.