Unit-2

The Evolution of Cloud Computing

- To get an appreciation of how we got into the cloud environment
    - understand the evolution of computing
- Looking at the evolution of the computing hardware itself, from the first generation to the current (fourth) generation of computers, shows how we got from there to here.
- The hardware, however, was only part of the evolutionary process. As hardware evolved, so did software. As networking evolved, so did the rules for how computers communicate. The development of such rules, or protocols, also helped drive the evolution of Internet software
- Establishing a common protocol for the Internet led directly to rapid growth in the number of users online. This has driven technologists to make even more changes in current protocols and to create new ones.
    - The use of IPv6 (Internet Protocol version 6) to mitigate addressing concerns and for improving the methods we use to communicate over the Internet.
- The ability to build a common interface to the Internet has evolved with the improvements in hardware and software. Using web browsers has led to a steady migration away from the traditional data center model to a cloud-based model.
- Using technologies such as server virtualization, parallel processing, vector processing, symmetric multiprocessing, and massively parallel processing has fueled radical change.

1. **Hardware Evolution**

- Computerization has permeated nearly every facet of our personal and professional lives.
- Computer evolution has been both rapid and fascinating.
    - The first step along the evolutionary path of computers occurred in 1930, when binary arithmetic was developed and became the foundation of computer processing technology, terminology, and programming languages.
    - Calculating devices date back to at least as early as 1642, when a device that could mechanically add numbers was invented. Adding devices evolved from the abacus. It was a significant milestone in the history of computers.
    - In 1939, the Berry brothers invented an electronic computer capable of operating digitally. Computations were performed using vacuum-tube technology.
    - In 1941, the introduction of Konrad Zuse's Z3 at the German Laboratory for Aviation in Berlin was one of the most significant events in the evolution of computers because this machine supported both floating-point and binary arithmetic. Because it was a "Turing-complete" device, it is considered to be the very first computer that was fully operational. A programming language is considered Turing-complete if it falls into the same computational class as a Turing machine, meaning that it can perform any calculation a universal Turing machine can perform.

1.1 First-Generation Computers

- The first generation of modern computers can be traced to 1943, when the Mark I and Colossus computers were developed.
- With financial backing from IBM (then International Business Machines Corporation), the Mark I was designed and developed at Harvard University.
    - It was a general-purpose electromechanical programmable computer.
- Colossus, on the other hand, was an electronic computer built in Britain at the end 1943.
    - Colossus was the world's first programmable, digital, electronic, computing device.

- First-generation computers were built using hard-wired circuits and vacuum tubes (thermionic valves). Data was stored using paper punch cards.
- Colossus was used in secret during World War II to help decipher teleprinter messages encrypted by German forces using the Lorenz SZ40/42 machine. British code breakers referred to encrypted German teleprinter traffic as "Fish" and called the SZ40/42 machine and its traffic "Tunny." To accomplish

its deciphering task, Colossus compared two data streams read at high speed from a paper tape. Colossus evaluated one data stream representing the encrypted “Tunny,” counting each match that was discovered based on a programmable Boolean function. A comparison with the other data stream was then made. The second data stream was generated internally and designed to be an electronic simulation of the Lorenz SZ40/42 as it ranged through various trial settings. If the match count for a setting was above a predetermined threshold, that data match would be sent as character output to an electric typewriter

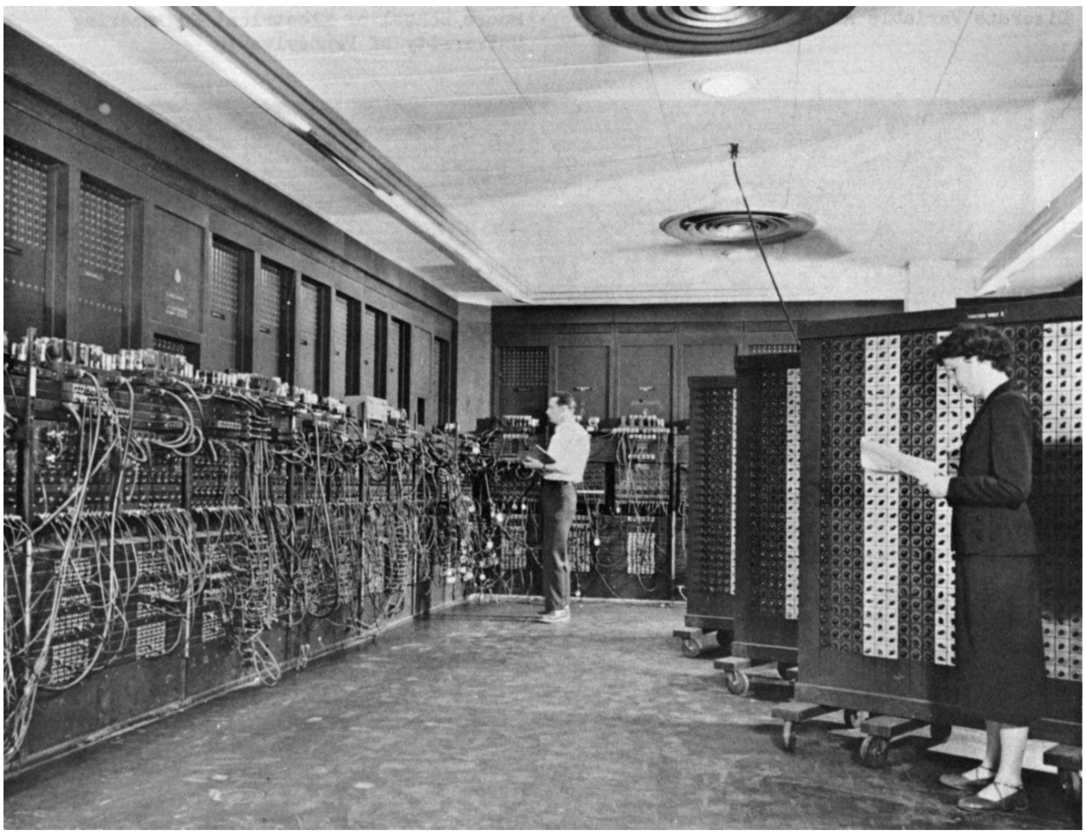
- In this generation mainly batch processing operating system were used. Punched cards, paper tape, and magnetic tape were used as input and output devices. The computers in this generation used machine code as programming language

1.2 Second-Generation Computers

- Another general-purpose computer of this era was ENIAC (Electronic Numerical Integrator and Computer, which was built in 1946.
- This was the first Turing-complete, digital computer capable of being reprogrammed to solve a full range of computing problems.
- ENIAC’s original purpose was to calculate artillery firing tables for the U.S. Army’s Ballistic Research Laboratory.
- ENIAC contained 18,000 thermionic valves, weighed over 60,000 pounds, and consumed 25 kilowatts of electrical power per hour.
- ENIAC was capable of performing 100,000 calculations a second. Within a year after its completion, however, the invention of the transistor meant that the inefficient thermionic valves could be replaced

with smaller, more reliable components, thus marking another major step in the history of computing.

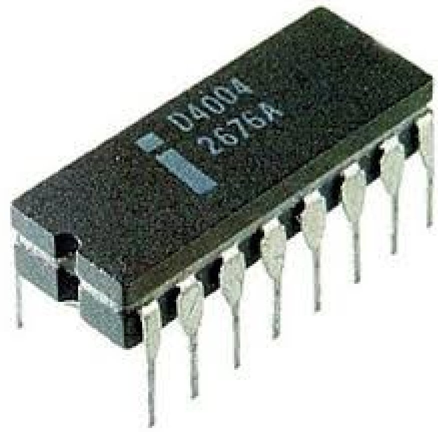
- Transistorized computers marked the advent of second-generation computers, which dominated in the late 1950s and early 1960s.
- Despite using transistors and printed circuits, these computers were still bulky and expensive. They were therefore used mainly by universities and government agencies.



- In this generation assembly language and high-level programming languages like FORTRAN, COBOL were used. The computers used batch processing and multiprogramming operating system.

1.3Third-Generation Computers

- The integrated circuit or microchip was developed by Jack St. Claire Kilby, an achievement for which he received the Nobel Prize in Physics in 2000.
- Kilby's invention started an explosion in third-generation computers. Even though the first integrated circuit was produced in September 1958, microchips were not used in computers until 1963.
- While mainframe computers like the IBM 360 increased storage and processing capabilities even further, the integrated circuit allowed the development of minicomputers that began to bring computing into many smaller businesses.
- Large-scale integration of circuits led to the development of very small processing units, the next step along the evolutionary trail of computing.
- In November 1971, Intel released the world's first commercial microprocessor, the Intel 4004. The 4004 was the first complete CPU on one chip and became the first commercially available microprocessor. It was possible because of the development of new silicon gate technology that enabled engineers to integrate a much greater number of transistors on a chip that would perform at a much faster speed. This development enabled the rise of the fourth-generation computer platforms.



- In this generation remote processing, time-sharing, multi-programming operating system were used. High-level languages (FORTRAN-II TO IV, COBOL, PASCAL PL/1, BASIC, ALGOL-68 etc.) were used during this generation.

1.4 Fourth-Generation Computers

- The fourth-generation computers that were being developed at this time utilized a microprocessor that put the computer's processing capabilities on a single integrated circuit chip.
- By combining random access memory (RAM), developed by Intel, fourth-generation computers were faster than ever before and had much smaller footprints.
- The 4004 processor was capable of "only" 60,000 instructions per second. As technology progressed, however, new processors brought even more speed and computing capability to users.
- The microprocessors that evolved from the 4004 allowed manufacturers to begin developing personal computers small enough and cheap enough to be purchased by the general public.

- The first commercially available personal computer was the MITS Altair 8800, released at the end of 1974. What followed was a flurry of other personal computers to market, such as the Apple I and II, the Commodore PET, the VIC-20, the Commodore 64, and eventually the original IBM PC in 1981.
- The PC era had begun in earnest by the mid-1980s. During this time, the IBM PC and IBM PC compatibles, the Commodore Amiga, and the Atari ST computers were the most prevalent PC platforms available to the public.
- Computer manufacturers produced various models of IBM PC compatibles. Even though micro processing power, memory and data storage capacities have increased by many orders of magnitude since the invention of the 4004 processor, the technology for large-scale integration (LSI) or very-large-scale integration (VLSI) microchips has not changed all that much. For this reason, most of today's computers still fall into the category of fourth-generation computers.
- In this generation time sharing, real time, networks, distributed operating system were used. All the high-level languages like C, C++, DBASE etc., were used in this generation.

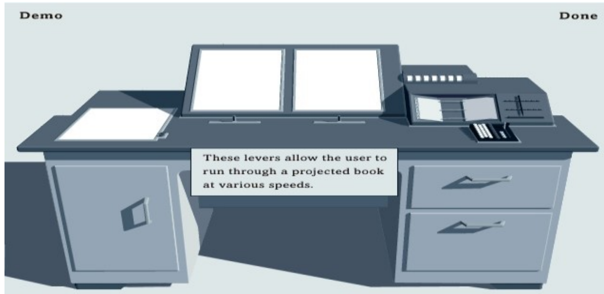
1.5Fifth Generation Computers

- The period of fifth generation is 1980-till date.
- In the fifth generation, the VLSI technology became ULSI (Ultra Large Scale Integration) technology, resulting in the production of microprocessor chips having ten million electronic components.
- This generation is based on parallel processing hardware and AI (Artificial Intelligence) software.

- AI is an emerging branch in computer science, which interprets means and method of making computers think like human beings. All the high-level languages like C and C++, Java, .Net etc., are used in this generation.
- AI includes:
    - Robotics
    - Neural Networks
    - Game Playing
    - Development of expert systems to make decisions in real life situations.
    - Natural language understanding and generation.

2. Internet Software Evolution

- The Internet is named after the Internet Protocol, the standard communications protocol used by every computer on the Internet.
- The conceptual foundation for creation of the Internet was significantly developed by three individuals.
    - The first, Vannevar Bush, wrote a visionary description of the potential uses for information technology with his description of an automated library system named MEMEX. Bush introduced the concept of the MEMEX in the 1930s as a microfilm-based "device in which an individual stores all his books, records, and communications, and which is mechanized so that it may be consulted with exceeding speed and flexibility."



- The second individual to have a profound effect in shaping the Internet was Norbert Wiener. Wiener was an early pioneer in the study of stochastic and noise processes. His work in stochastic and noise processes was relevant to electronic engineering, communication, and control systems. He also founded the field of cybernetics. This field of study formalized notions of feedback and influenced research in many other fields, such as engineering, systems control, computer science, biology, philosophy, etc. His work in cybernetics inspired future researchers to focus on extending human capabilities with technology. Influenced by Wiener, Marshall McLuhan put forth the idea of a global village that was interconnected by an electronic nervous system as part of our popular culture
- In 1957, the Soviet Union launched the first satellite, Sputnik I, prompting U.S. President Dwight Eisenhower to create the Advanced Research Projects Agency (ARPA) agency to regain the technological lead in the arms race. ARPA (renamed DARPA, the Defense Advanced Research Projects Agency, in 1972) appointed J. C. R. Licklider to head the new Information Processing Techniques Office (IPTO). Licklider was given a mandate to further the research of the SAGE system. The SAGE system was a continental air-defense network commissioned by

the U.S. military and designed to help protect the United States against a spacebased nuclear attack. SAGE stood for Semi-Automatic Ground Environment. (SAGE was the most ambitious computer project ever undertaken at the time, and it required over 800 programmers and the technical resources of some of America's largest corporations. SAGE was started in the 1950s and became operational by 1963. It remained in continuous operation for over 20 years, until 1983)

- While working at ITPO, Licklider evangelized the potential benefits of a country-wide communications network. His chief contribution to the development of the Internet was his ideas, not specific inventions. He foresaw the need for networked computers with easy user interfaces. His ideas foretold of graphical computing, point-and-click interfaces, digital libraries, e-commerce, online banking, and software that would exist on a network and migrate to wherever it was needed. Licklider worked for several years at ARPA, where he set the stage for the creation of the ARPANET. He also worked at Bolt Beranek and Newman (BBN), the company that supplied the first computers connected on the ARPANET.
- Roberts led the development of the network. His efforts were based on a novel idea of "packet switching" that had been developed by Paul Baran while working at RAND Corporation. The idea for a common interface to the ARPANET was first suggested in Ann Arbor, Michigan, by Wesley Clark at an ARPANET design session set up by Lawrence Roberts in April 1967. Roberts's implementation plan called for each site that was

to connect to the ARPANET to write the software necessary to connect its computer to the network. There were so many different kinds of computers and operating systems in use throughout the DARPA community that every piece of code would have to be individually written, tested, implemented, and maintained. Clark told Roberts that he thought the design was "bass-ackwards"

- Clark elaborated the concept to deploy a minicomputer called an Interface Message Processor at each site. The IMP would handle the interface to the ARPANET network. The physical layer, the data link layer, and the network layer protocols used internally on the ARPANET were implemented on this IMP. Using this approach, each site would only have to write one interface to the commonly deployed IMP. The host at each site connected itself to the IMP using another type of interface that had different physical, data link, and network layer specifications. These were specified by the Host/IMP Protocol in BBN Report 1822.14.
- The first networking protocol that was used on the ARPANET was the Network Control Program (NCP). The NCP provided the middle layers of a protocol stack running on an ARPANET-connected host computer. The NCP managed the connections and flow control among the various processes running on different ARPANET host computers. An application layer, built on top of the NCP, provided services such as email and file transfer. These applications used the NCP to handle connections to other host computers.

- A minicomputer was created specifically to realize the design of the Interface Message Processor. This approach provided a system-independent interface to the ARPANET that could be used by any computer system. Because of this approach, the Internet architecture was an open architecture from the very beginning

Cloud-Based Service Offerings

Cloud computing may be viewed as a resource available as a service for virtual data centers, but cloud computing and virtual data centers are not the same. For example, consider Amazon's S3 Storage Service. This is a data storage service designed for use across the Internet (i.e., the cloud). It is designed to make web-scale computing easier for developers. According to Amazon:

Amazon S3 provides a simple web services interface that can be used to store and retrieve any amount of data, at any time, from anywhere on the web. It gives any developer access to the same highly scalable, reliable, fast, inexpensive data storage infrastructure that Amazon uses to run its own global network of web sites. The service aims to maximize benefits of scale and to pass those benefits on to developers.

Amazon.com has played a vital role in the development of cloud computing. In modernizing its data centers after the dot-com bubble burst in 2001, it discovered that the new cloud architecture it had implemented resulted in some very significant internal efficiency improvements. By providing access to its systems for third-party users on a utility computing basis, via Amazon Web Services, introduced in 2002, a revolution of sorts began. Amazon Web Services began implementing its model by renting computing cycles as a service outside a given user's domain, wherever on the planet that domain might be located. This approach modernized a style of computing whereby IT-related capabilities could be provided "as a service" to users. By allowing their users to access technology-enabled services "in the cloud," without any need for knowledge of, expertise with, or control over how the technology infrastructure that supports those services worked, Amazon shifted the approach to computing radically. This approach transformed cloud computing into a paradigm whereby data is permanently stored in remote servers accessible via the Internet and cached temporarily on client devices that may include desktops, tablet computers, notebooks, hand-held devices, mobile phones, etc., and is often called Software as a Service (SaaS).

SaaS is a type of cloud computing that delivers applications through a browser to thousands of customers using a multiuser architecture. The focus for SaaS is on the end user as opposed to managed services (described below). For the customer, there are no up-front investment costs in servers or software licensing. For the service provider, with just one product to maintain, costs are relatively low compared to the costs incurred with a conventional hosting model. Salesforce.com3 is by far the best-known example of SaaS computing among enterprise applications. Salesforce.com was founded in 1999 by former Oracle executive Marc Benioff, who pioneered the concept of delivering enterprise applications via a simple web site. Nowdays, SaaS is also commonly used for enterprise resource planning and human resource applications. Another example is Google Apps, which provides online access via a web browser to the most common office and business applications used today, all the while keeping the software and user data stored on Google servers. A decade ago, no one could have predicted the sudden rise of SaaS applications such as these.

Platform-as-a-Service (PaaS) is yet another variation of SaaS. Sometimes referred to simply as web services in the cloud, PaaS is closely related to SaaS but delivers a platform from which to work rather than an application to work with. These service providers offer application programming interfaces (APIs) that enable developers to exploit functionality over the Internet, rather than delivering full-blown applications. This variation of cloud computing delivers development environments to programmers, analysts, and software engineers as a service. A general model is implemented under which developers build applications designed to run on the provider's infrastructure and which are delivered to users in via an Internet browser. The main drawback to this approach is that these services are limited by the vendor's design and capabilities. This means a compromise between freedom to develop code that does something other than what the provider can provide and application predictability, performance, and integration. An example of this model is the Google App Engine. According to Google, "Google App Engine makes it easy to build an application that runs reliably, even under heavy load and with large amounts of data."7 The Google App Engine environment includes the following features

- Dynamic web serving, with full support for common web technologies
- Persistent storage with queries, sorting, and transactions
- Automatic scaling and load balancing
- APIs for authenticating users and sending email using Google Accounts
- A fully featured local development environment that simulates Google App Engine on your computer

Server Virtualization

Virtualization is the creation of a virtual (rather than actual) version of something, such as an operating system, a server, a storage device or network resources.

Operating system virtualization is the use of software to allow a piece of hardware to run multiple operating system images at the same time. The technology got its start on mainframes decades ago, allowing administrators to avoid wasting expensive processing power.

There are three areas of IT where virtualization is making headroads, network virtualization, storage virtualization and server 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. The idea is that virtualization disguises the true complexity of the network by separating it into manageable parts, much like your partitioned hard drive makes it easier to manage your files.
- Storage virtualization is the pooling of physical storage from multiple network storage devices into what appears to be a single storage device that is managed from a central console. Storage virtualization is commonly used in storage area networks (SANs).
- Server virtualization is the masking of server resources (including the number and identity of individual physical servers, processors, and operating systems) from server users. The intention is to spare the user from having to understand and manage complicated details of server resources while increasing resource sharing and utilization and maintaining the capacity to expand later.

Virtualization is a method of running multiple independent virtual operating systems on a single physical computer.

The term was coined in the 1960s in reference to a virtual machine (sometimes called a pseudo-machine). The creation and management of virtual machines has often been called platform virtualization.

Platform virtualization is performed on a given computer (hardware platform) by software called a control program. The control program creates a simulated environment, a virtual computer, which enables the device to use hosted software specific to the virtual environment, sometimes called guest software.

The guest software, which is often itself a complete operating system, runs just as if it were installed on a stand-alone computer. Frequently, more than one virtual machine is able to be simulated on a single physical computer, their number being limited only by the host device's physical hardware resources. Because the guest software often requires access to specific peripheral devices in order to function, the virtualized platform must support guest interfaces to those devices. Examples of such devices are the hard disk drive, CD-ROM, DVD, and network interface card. Virtualization technology is a way of reducing the majority of hardware acquisition and maintenance costs, which can result in significant savings for any company.

Parallel Processing

In computers, parallel processing is the processing of program instructions by dividing them among multiple processors with the objective of running a program in less time.

In the earliest computers, only one program ran at a time. A computation-intensive program that took one hour to run and a tape copying program that took one hour to run would take a total of two hours to run.

An early form of parallel processing allowed the interleaved execution of both programs together. The computer would start an I/O operation, and while it was waiting for the operation to complete, it would execute the processor-intensive program. The total execution time for the two jobs would be a little over one hour.

The next improvement was multiprogramming. In a multiprogramming system, multiple programs submitted by users were each allowed to use the processor for a short time, each taking turns and having exclusive time with the processor in order to execute instructions. This approach is known as "round-robin scheduling" (RR scheduling. To users it appeared that all of the programs were executing at the same time.

Problems of resource contention first arose in these systems. Explicit requests for resources led to the problem of the deadlock. Competition for resources on machines with no tie-breaking instructions lead to the critical section routine.

Contention occurs when several processes request access to the same resource. In order to detect deadlock situations, a counter for each processor keeps track of the number of consecutive requests from a process that have been rejected. Once that number reaches a predetermined threshold, a state machine that inhibits other processes from making requests to the main store is initiated until the deadlocked process is successful in gaining access to the resource.