Cloud Computing Concepts

CS 3132

Dr. Anand Kumar Mishra

NIIT University

Historical Developments

Historical developments

- 5 core technologies that played an important role in the realization of cloud computing
  - Distributed systems
  - Virtualization
  - Web 2.0
  - Service-oriented computing
  - Utility-oriented computing

What is a Distributed System?

- A distributed system is a collection of multiple physically separated servers and data storage that reside in different systems worldwide.
- These components can collaborate, communicate, and work together to achieve the same objective, giving an illusion of being a single, unified system with powerful computing capabilities.

- A distributed computing server, databases, software applications, and file storage systems can all be considered distributed systems

Distributed systems must have a network that connects all components (machines, hardware, or software) together so they can transfer messages to communicate with each other.

Examples of Distributed Systems

- Networks: The internet (World Wide Web) itself
- Telecommunication networks with multiple antennas, amplifiers, and other networking devices appear as a single system to end-users
- Distributed Real-time Systems
  - Airlines use flight control systems
  - Uber and Lyft use dispatch systems
  - Manufacturing plants use automation control systems
  - Logistics and e-commerce companies use real-time tracking systems
- Content Delivery Networks (CDNs) utilize geographically separated regions to store data locally in order to serve end-users faster

Historical developments - Distributed systems

- Clouds are essentially large distributed computing facilities that make available their services to third parties on demand

- A distributed system is a collection of independent computers that appears to its users as a single coherent system:
  - composed of multiple independent components
  - components are perceived as a single entity by users

Historical developments - Distributed systems

Two general ways that distributed systems function:

- Each machine works toward a common goal and the end-user views results as one cohesive unit
- Each machine has its own end-user and the distributed system facilitates sharing resources or communication services

Important functions of distributed computing

- Resource sharing - whether it's the hardware, software or data that can be shared
- Openness - how open is the software designed to be developed and shared with each other
- Concurrency - multiple machines can process the same function at the same time
- Scalability - how do the computing and processing capabilities multiply when extended to many machines
- Fault tolerance - how easy and quickly can failures in parts of the system be detected and recovered
- Transparency - how much access does one node have to locate and communicate with other nodes in the system.

Historical developments - Virtualization

- Virtualization is another core technology for cloud computing
- It encompasses a collection of solutions allowing the abstraction of some of the fundamental elements for computing, such as
  - hardware, runtime environments, storage, and networking

Historical developments – Web 2.0

- The Web is the primary interface through which cloud computing delivers its services
- Web 2.0 brings interactivity and flexibility into Web pages
  - providing enhanced user experience by gaining Web-based access to all the functions that are normally found in desktop applications
- These capabilities are obtained by integrating a collection of standards and technologies such as XML, Asynchronous JavaScript and XML (AJAX), Web Services, and others
- These technologies allow to build applications leveraging the contribution of users, who now become providers of content

Historical developments – Service-oriented computing

- Service orientation is the core reference model for cloud computing systems
- Adopts the concept of services as the main building blocks of application and system development
- Service-oriented computing (SOC) supports:
  - the development of rapid, low-cost, flexible, interoperable, and evolvable applications and systems

Historical developments – Service-oriented computing - [Service]

- Virtually any piece of code that performs a task can be turned into a service and expose its functionalities through a network-accessible protocol
- A service is supposed to be:
  - loosely coupled, reusable,
  - programming language independent, and
  - location transparent
- Loose coupling allows services to serve different scenarios more easily and makes them reusable
- Independence from a specific platform increases services accessibility
- Services are composed and aggregated into a service-oriented architecture (SOA)

Historical developments – Service-oriented computing

- Service-oriented computing introduces and diffuses two important concepts, which are also fundamental to cloud computing:
  - Quality of service (QoS)
    - identifies a set of functional and nonfunctional attributes that can be used to evaluate the behavior of a service from different perspectives
  - Software-as-a-Service (SaaS)
    - delivery model for applications

Historical developments: Utility-oriented computing

- Utility computing is a vision of computing that defines a **service-provisioning model** for compute services in which resources such as storage, compute power, applications, and infrastructure are packaged and offered on a pay-per-use basis