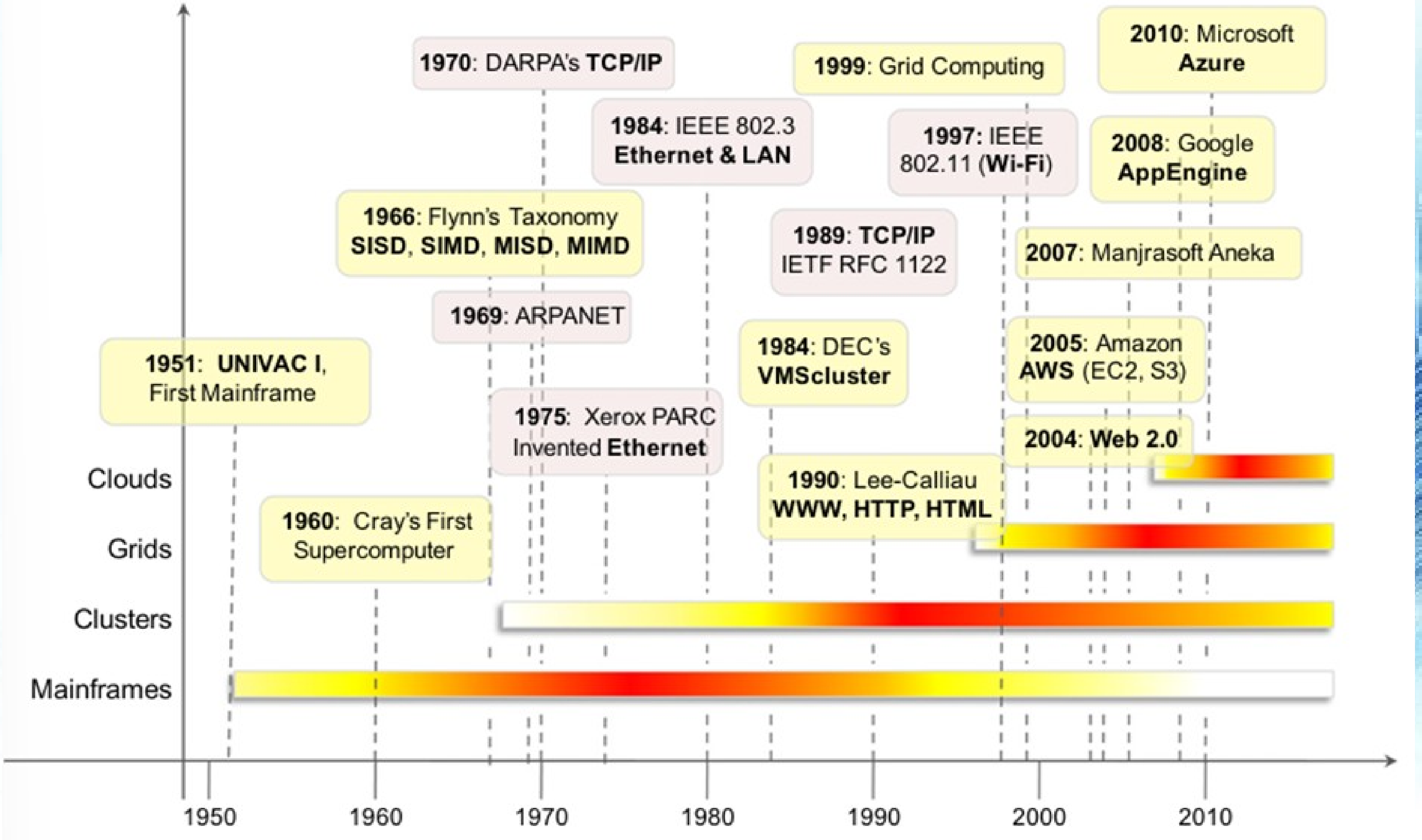
HISTORICAL DEVELOPMENT - CLOUD COMPUTING

- GEETHA PK 22Z219
  MITHRA KM 22Z238
  MOUMITHA K 22Z241
  SRUTHI S 22Z264
  THARIGALAKSHMI S 22Z268
  VARSHINI R 22Z270

HISTORICAL EVOLUTION AND DISTRIBUTED SYSTEMS

- SRUTHI S (22Z264)


1970: DARPA's TCP/IP
1999: Grid Computing
2010: Microsoft Azure
1984: IEEE 802.3 Ethernet & LAN
1997: IEEE 802.11 (Wi-Fi)
2008: Google AppEngine
1966: Flynn's Taxonomy SISD, SIMD, MISD, MIMD
1989: TCP/IP IETF RFC 1122
2007: Manjrasoft Aneka
1969: ARPANET
1984: DEC's VMScluster
2005: Amazon AWS (EC2, S3)
1951: UNIVAC I, First Mainframe
1975: Xerox PARC Invented Ethernet
2004: Web 2.0
1990: Lee-Calliau WWW, HTTP, HTML
1960: Cray's First Supercomputer
Clouds
Grids
Clusters
Mainframes
1950
1960
1970
1980
1990
2000
2010

Five core technologies that played an important role in the realization of cloud computing:

1. Distributed systems
2. Virtualization
3. Web 2.0
4. Service Orientation
5. Utility Computing

Distributed Systems

Clouds:

- ➢ Large distributed computing facilities
- ➢ Provides services to third parties on demand.

Distributed System:

- ➢ Collection of independent computers
- ➢ Appears to its users as a single coherent system

How Distributed Systems Paved the Way for Cloud Computing

- ➢ Multiple Independent Components and Single System Perception
- ➢ Resource Sharing & Optimization
- ➢ Scalability
- ➢ Concurrency
- ➢ Continuous Availability
- ➢ Fault Tolerance
- ➢ Cloud Evolution

Limitation

- Systems were required to be present at the same geographical location.

Three major milestones have led to cloud computing

- Mainframe computing
- Cluster computing
- Grid computing

MAJOR TECHNOLOGIES IN DISTRIBUTED SYSTEMS

- MOUMITHA K (22Z241)

Mainframes - 1950s

- ❑ Specialized in massive data movement and I/O operations
- ❑ Main application - Batch processing
- ❑ Still used today in transaction processing (banking, ticket booking, telecom, government)

Advantages:

- ➢ Always on
- ➢ High fault tolerance - no shutdown for failures

Disadvantages:

- ➢ Extremely expensive - not affordable for smaller organizations

➔ Led to Cluster Computing

Cluster Computing - 1980s

- ❑ Multiple commodity machines connected via high-bandwidth network
- ❑ Used for parallel & high-performance computing
- ❑ Enabled distributed computing frameworks - MPI, PVM, Condor

Advantages:

- ➢ Low cost - cheaper than mainframes
- ➢ Scalable - easy to add new machines

Disadvantages:

- ➢ Geographical restrictions - machines must be close together

➔ Led to Grid Computing

Grid Computing - 1990s

- ❑ Connects machines across locations via Internet
- ❑ Users consume resources like utilities - pay-per-use model
- ❑ Heterogeneous nodes - machines from different organizations

Advantages:

- ➢ Greater flexibility - no need for uniform hardware

Disadvantages:

- ➢ Network limitations - depends on high-bandwidth connections

➔ Led to Cloud Computing

Cloud Computing

Successor of Grid computing

Combining the Best of All

- ❑ Mainframes → Always on, fault-tolerant
- ❑ Clusters → Uses commodity machines, scalable
- ❑ Grids → Pay-per-use resource sharing

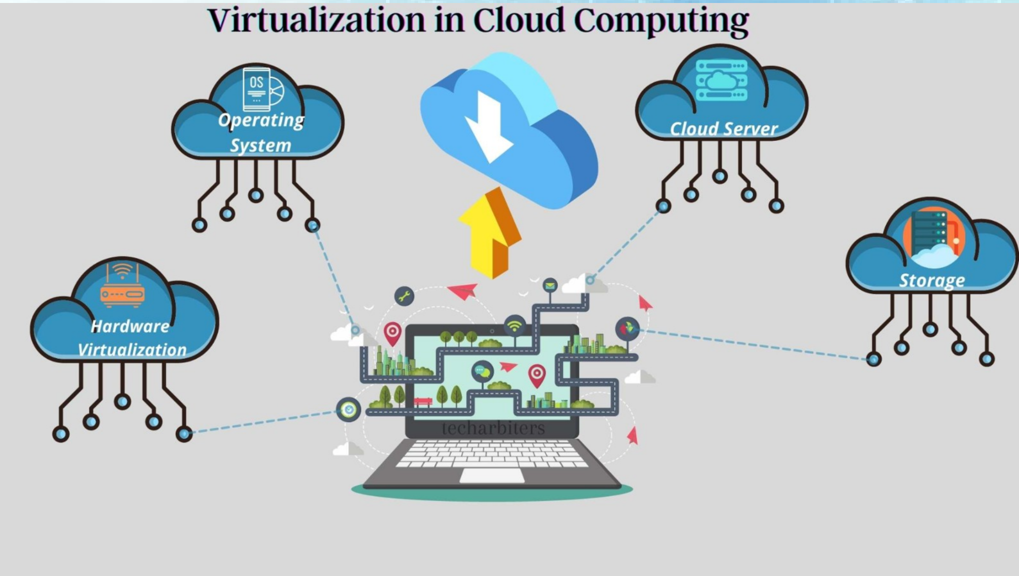
- ❖ On-demand computing, accessible worldwide
- ❖ Scalable, cost-efficient, and reliable

VIRTUALIZATION

- GEETHA PK 22Z219

What is Virtualization ?

Virtualization is the foundation of cloud computing, used to create a virtual version of an underlying service With the help of Virtualization, multiple operating systems and applications can run on the same machine and its same hardware at the same time.Virtualization allows sharing of a single physical instance of a resource or an application among multiple customers and organizations at one time.



Types of Virtualization

- ✓ **Data virtualization**:brings data from different sources together in one place
- ✓ **Storage Virtualization** :combines storage from different servers into a single system
- ✓ **Network Virtualization** :allows multiple virtual networks to run on the same physical network
- ✓ **Application Virtualization**:use an application on your local device while it’s actually hosted on a remote server

Uses of Virtualization

✓ **Resource Optimization**:Maximizes hardware utilization

✓ **Cost Reduction**:Reduces hardware and maintenance costs

✓ **Scalability**:Enables quick scaling

✓ **Flexibility**:Dynamically allocates resources

✓ **Disaster Recovery**:Simplifies backup

✓**Energy Efficiency**:Reduces power consumption

Future Trends in Virtualization

✓**Serverless Computing** – Eliminates the need to manage infrastructure (AWS Lambda, Azure Functions).

✓**Edge Computing** – Virtualized resources closer to users for real-time processing(Google Distributed Cloud Edge).

✓**AI-Driven Virtualization** – Smart resource allocation for better efficiency(VMware AI-Driven Hypervisors).

WEB 2.0

- MITHRA K M 22Z238


Web 2.0
future
http
apps
now
groups
storage
blog
version
software
browser
wiki
server
welcome
share
forum
network
website
protection
domain
online
global
blogging
usability
e-business
accessibility
www
access
modern
speed
connection
data
engine
tags
world
news
chat
computing
yes
user
go
revolution
ready
friendly
login
cloud
updates
smart
rss
download
feed
start
flexibility
new
search
social networking
upgrade
encryption
next
security
recommmend
upload
media
compatibility
standard
internet
webpage
reliability
programs
open
like
simplicity
communication
click here
easy
generation

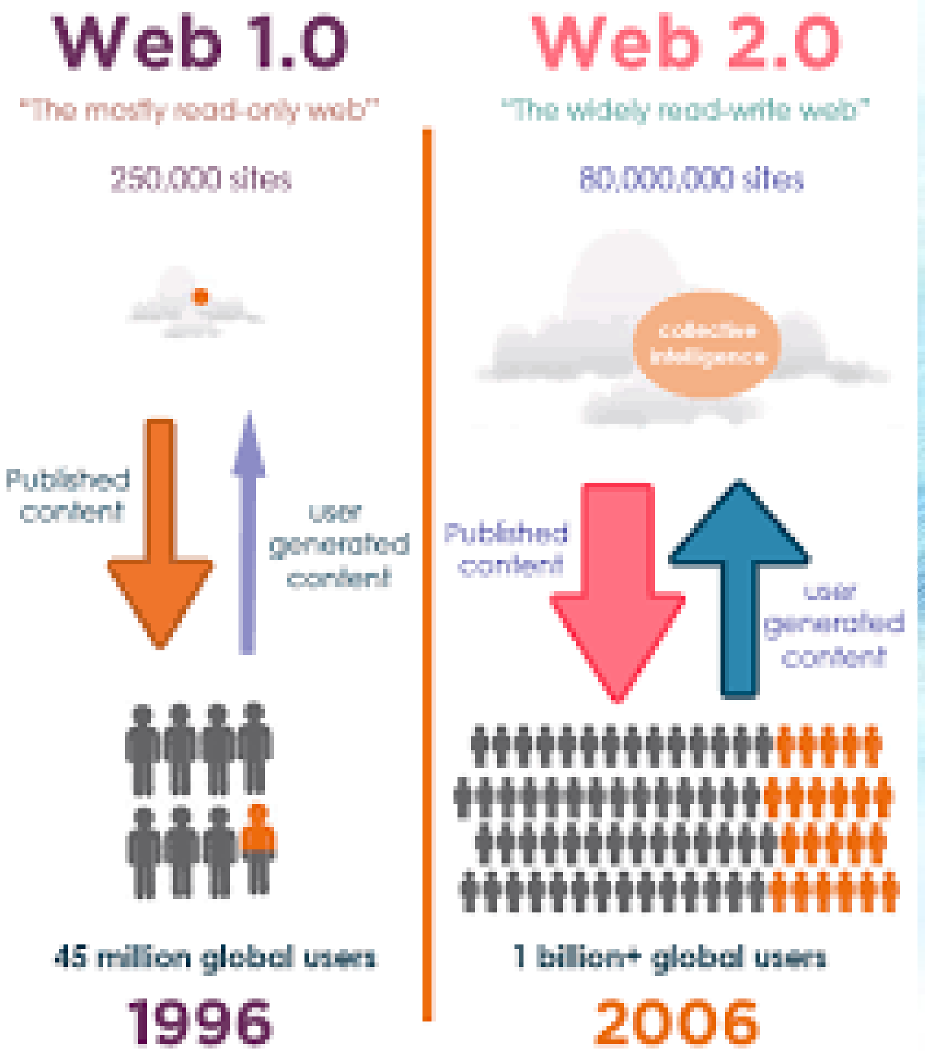
Examples of Web 2.0 Applications

Google Docs

YouTube

WIKIPEDIA
The Free Encyclopedia

- Web 2.0 enables user-generated content, unlike Web 1.0
- Web 1.0 had static pages, whereas Web 2.0 supports dynamic content
- Web 1.0 follows a linear format, while Web 2.0 is more interactive and non-linear.
- Web 2.0 enhances functionality with web APIs and cloud-based applications.



Key Features

- **Interactivity & Flexibility**: Enhanced user experience similar to desktop applications.
- **User Contribution:** Users are content providers (e.g., Wikipedia, YouTube, Flickr).
- **Device Accessibility:** Services are accessible via mobile phones, car dashboards, TV sets, etc.
- **Continuous Updates:** Applications are updated frequently without requiring user intervention.

Technologies Behind Web 2.0

01

AJAX

Enables dynamic web pages without refreshing.

02

XML

Standardized data exchange.

03

Web Services

Allows integration and communication between applications.

04

RSS

Enables content distribution and updates

SERVICE ORIENTED COMPUTING

- VARSHINI R 22Z270

SERVICE -ORIENTED CLOUD COMPUTING:

- SOC is the **core reference model** for cloud computing.
- Uses **services** as the building blocks for applications and systems.

Characteristics:

- **Self-describing** components.
- Can perform anything from **simple tasks to complex business processes**.
- **Loosely coupled** for reusability and flexibility.
- Forms the basis of **modern cloud services like SaaS, PaaS, and IaaS**.

QOS AND SAAS

1.Quality of Service (QoS):

Ensures **service performance** meets user expectations.

- **Key QoS attributes**:
  Performance: Response time & latency.
  Security: Data encryption & protection.

2.Software-as-a-Service (SaaS):

Cloud-based software delivery model.

- **Key Features:**
  Multi-tenancy: Multiple users share the same infrastructure.
  Subscription-based access: Pay-per-use pricing model.

Historical Developments of SOC:

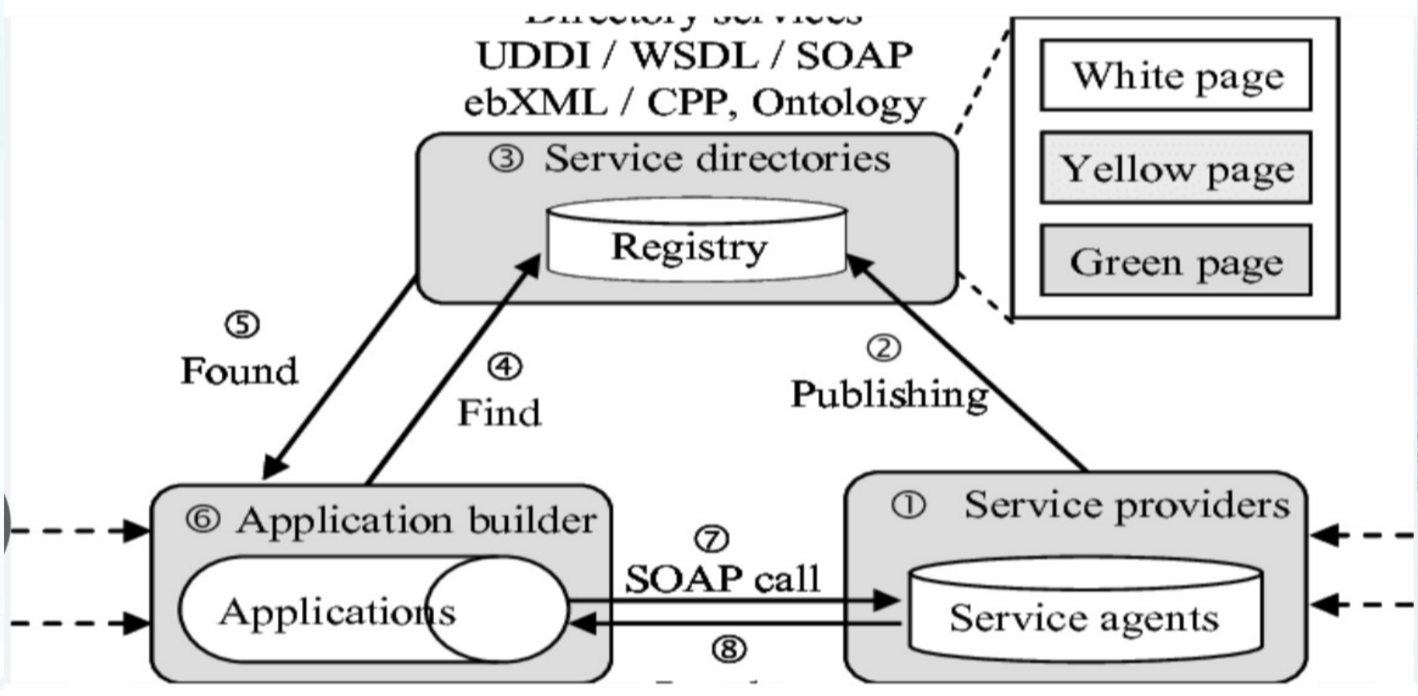
Introduction of Web Services (WS) in SOC

- Web Services (WS) emerged as one of the **earliest implementations of Service-Oriented Computing (SOC)**.
- They introduced **machine-to-machine** communication over the **World Wide Web (WWW)**, allowing applications to interact seamlessly.

Technologies Enabling Web Services

1. **Web Service Description Language (WSDL)**
2. **Simple Object Access Protocol (SOAP)**
3. **Role of W3C in Standardizing Web Services**

THREE PARTIES OF SERVICE ORIENTED COMPUTING



UTILITY ORIENTED COMPUTING

- THARIGALAKSHMI S 22Z268

Utility-oriented computing

Utility computing is a service-provisioning model where computing resources such as storage, processing power, applications, and infrastructure are delivered on a **pay-per-use basis**, similar to utilities like electricity, water, or gas.

- In the early days, computing power was centralized in **mainframes**, where companies like IBM provided processing capabilities to organizations such as banks and government agencies. This introduced a **service-based model** where businesses paid for computing resources rather than owning them.

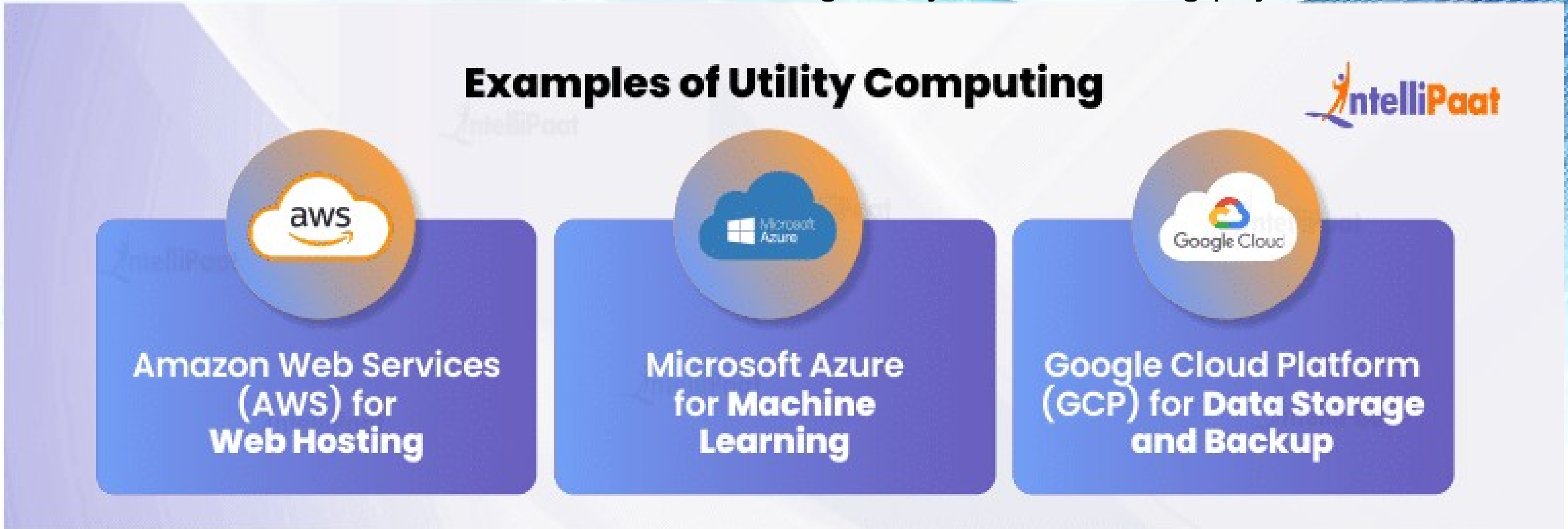
- With the need for more processing power, organizations moved towards **cluster computing**, where multiple computers worked together to solve complex computational problems. Research institutions started leveraging external computing clusters to handle their computational needs.
- The development of **computing grids** expanded utility computing on a global scale. Grids enabled **on-demand access** to storage, computation, and services, laying the foundation for **market-oriented computing**, where computing resources were **bought and sold like commodities**.
- The rise of e-commerce in the late 1990s made people comfortable with **buying online services**. This helped utility computing grow, as **infrastructure for online payments** became widely accessible.
- SOC and SOA introduced the idea that applications could be composed of **external services** rather than being built from scratch. This meant businesses could access services **on demand** and integrate them into their systems without managing infrastructure.

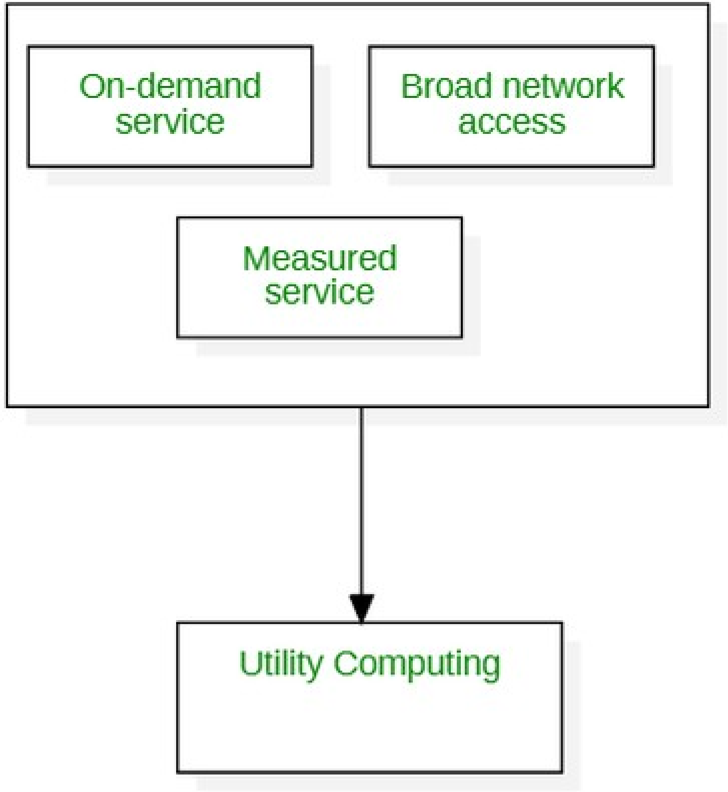
Cloud Computing: The Utility Computing

Cloud computing fully realizes the **utility computing vision**, where computing resources are provided **on-demand**, scalable, and charged **based on usage**. Cloud computing now dominates modern IT infrastructure.

Examples:

- **Microsoft Azure, Google Cloud, AWS** – Provide storage, computing, networking, and AI services as utilities.
- **Netflix** – Uses AWS cloud services to stream content globally without owning physical servers.




On-demand service
Broad network access
Measured service
Utility Computing

On-Demand Service

- Resources (e.g., computing power, storage, applications) are provided when needed.
- Users can **quickly scale up or down** based on their requirements.
- **Example:** Amazon EC2 providing virtual machines instantly upon request

Broad Network Access

- Services are accessible over the network (typically the internet) from **any device**.
- Enables users to **remotely access and manage resources** from various devices.
- **Example:** Google Drive, accessible via mobile devices and desktops.

Measured Service

- Resource usage is **monitored and billed according to actual consumption**.
- Users are charged for what they use, promoting **cost-efficiency**.
- **Example:** AWS charges users per hour of virtual machine usage or per GB of data stored.

THANK YOU