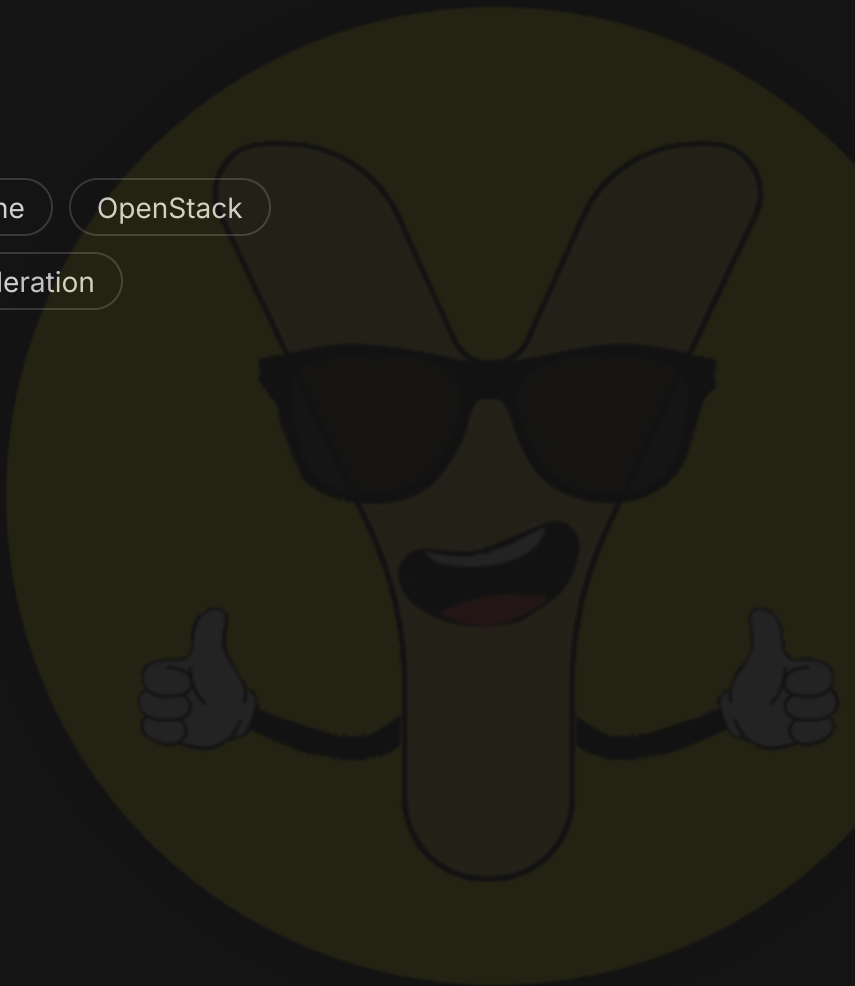
AKTU · 3RD YEAR · SEMESTER 5 · BCAM051

Cloud Computing

Unit 5 — Cloud Technologies and Advancements

- Hadoop: MapReduce
- VirtualBox
- Google App Engine
- Programming Environment for Google App Engine
- OpenStack
- Federation in the Cloud
- Four Levels of Federation
- Federated Services and Applications
- Future of Federation



DECODED BY

Chirag Jain

Founder, Ycotes · Gold Medalist in Artificial Intelligence, IIT Madras · taught 10,000+ students

theycotes.com

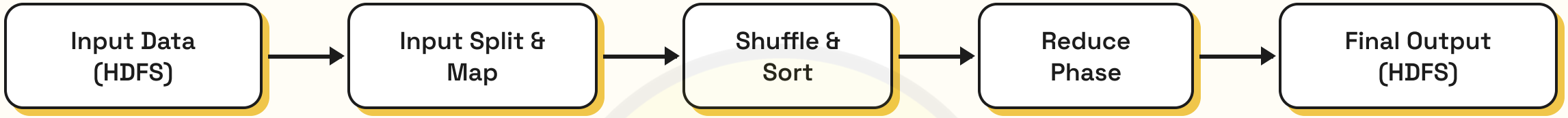
Hadoop and MapReduce Overview

HADOOP MAPREDUCE

A software framework and programming model for processing massive datasets in parallel across distributed clusters using fault-tolerant, commodity hardware.

- Originates from Google's MapReduce and distributed computing principles.
- Divides computation into two core functional primitives: Map and Reduce.
- Employs data locality: computation is scheduled near the node storing data blocks.
- Automates fault tolerance, data distribution, and inter-node network communications.

MapReduce Execution Pipeline



End-to-end data processing lifecycle from raw HDFS blocks to final aggregated results.

MapReduce Execution: Word Count

Calculate the frequency of each word in the input text: 'cloud computing cloud federation'.

1. Splitting: Divide input text into individual lines or records.
2. Map Phase: Emit key-value pairs (k_1, v_1): ('cloud', 1), ('computing', 1), ('cloud', 1), ('federation', 1).
3. Shuffle & Sort: Group identical keys across nodes: 'cloud' → [1, 1]; 'computing' → [1]; 'federation' → [1].
4. Reduce Phase: Sum list values per key: $'cloud', \sum(1,1) = 2, 'computing', 1, 'federation', 1$.

Final Result: {('cloud', 2), ('computing', 1), ('federation', 1)}

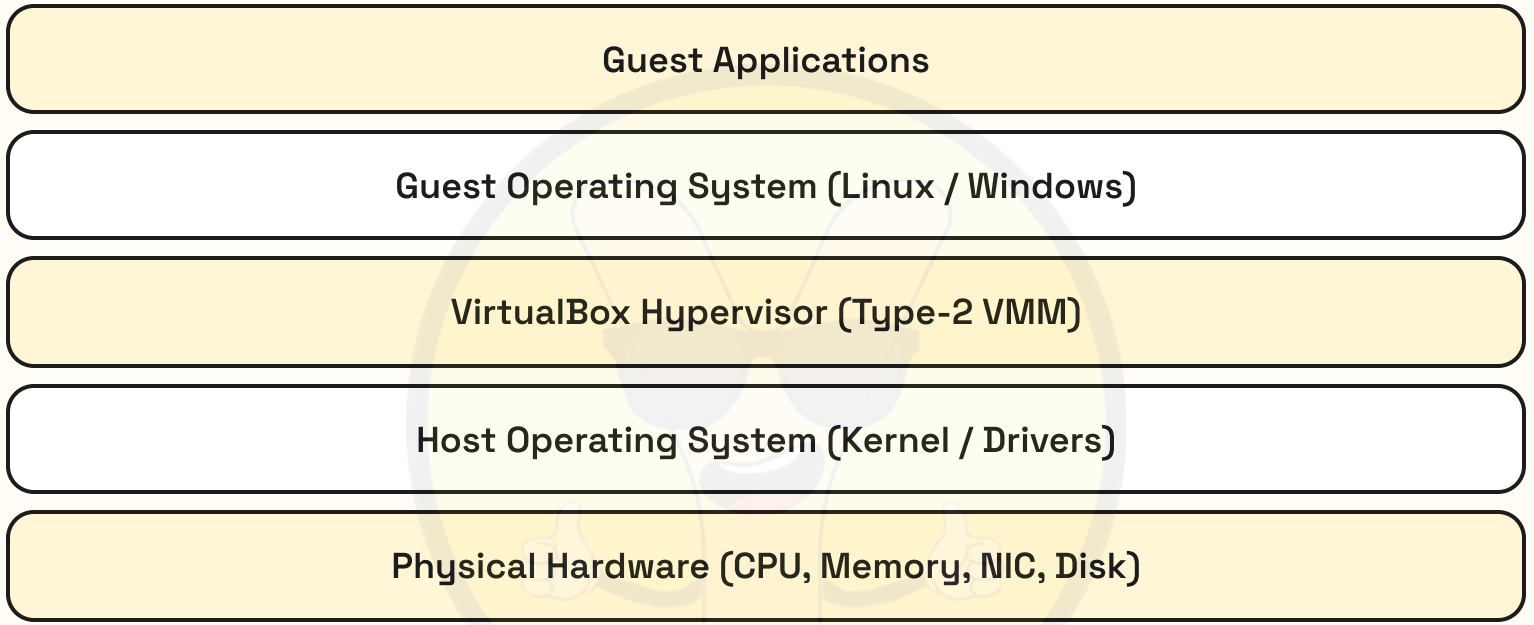
Oracle VirtualBox Overview

ORACLE VM VIRTUALBOX

A cross-platform, open-source Type-2 (hosted) hypervisor that enables users to create and run multiple x86 and AMD64 virtual machines concurrently on a single physical host.

- Runs on top of an existing host operating system such as Windows, Linux, or macOS.
- Abstracts host physical CPU, RAM, disk, and network interfaces for guest OS environments.
- Supports hardware-assisted virtualization extensions including Intel VT-x and AMD-V.
- Supports virtual disk formats including VDI, VMDK, VHD, and raw disk images.

VirtualBox Architecture



Layered architecture of VirtualBox showing hypervisor dependency on the host OS.

VirtualBox Networking and Storage Features

VirtualBox delivers configurable virtualization capabilities for testing and enterprise development:

- Network Modes
  - NAT (Network Address Translation): Default mode providing external internet access.
  - Bridged Networking: Assigns guest an IP directly on physical network.
  - Host-Only: Creates isolated internal network between host and specific guests.
- Management Features
  - Snapshots preserve complete VM state, memory, and disk at a point in time.
  - Guest Additions provide shared clipboards, folder sharing, and auto-resizing.

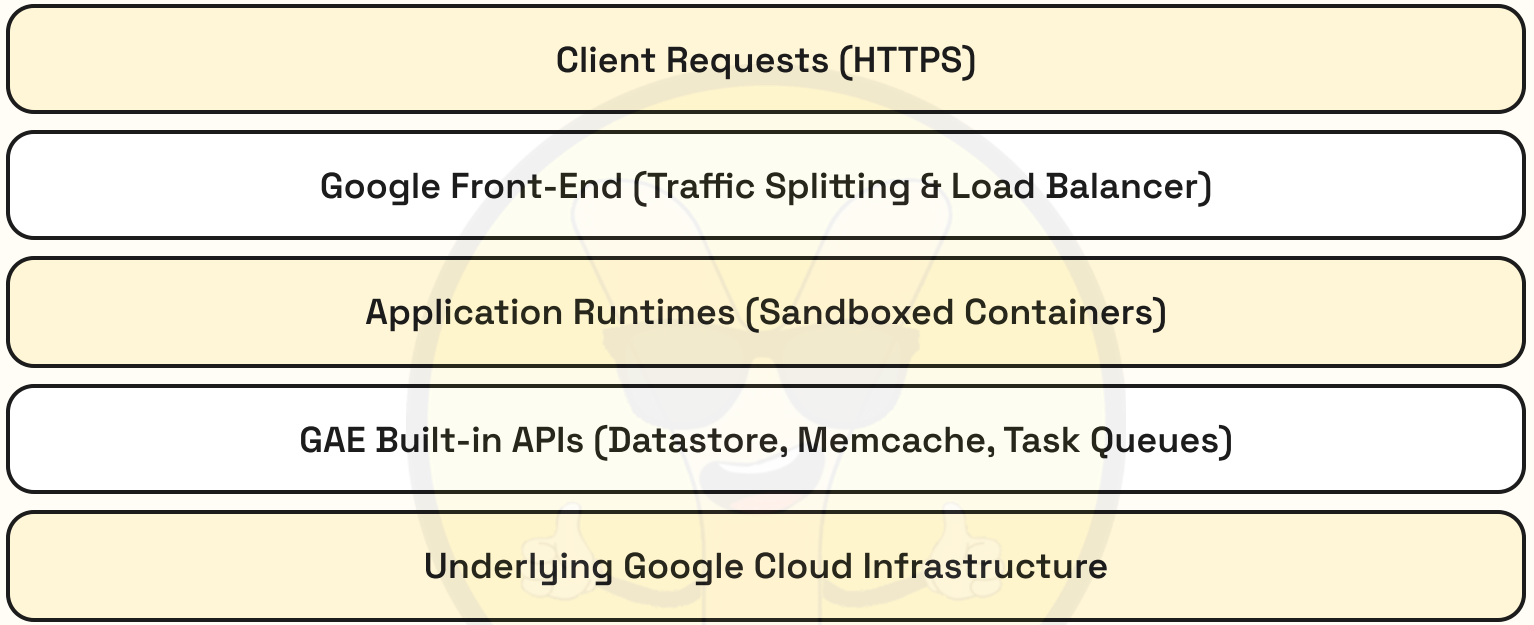
Google App Engine (GAE) Overview

GOOGLE APP ENGINE (GAE)

A fully managed, serverless Platform as a Service (PaaS) offering by Google Cloud that enables developers to build and deploy scalable web applications without managing underlying infrastructure.

- Provides dynamic auto-scaling from zero instances up to thousands based on traffic volume.
- Abstracts server provisioning, OS patching, system updates, and network topology.
- Includes built-in load balancing, health checks, and application version management.
- Offers integrated developer services: logging, task queues, NoSQL Datastore, and Memcache.

Google App Engine System Stack



Request routing and service isolation inside Google App Engine.

Programming Environment for Google App Engine

GAE provides two distinct runtime operational environments for application hosting:

- Standard Environment
  - Runs inside heavily sandboxed, pre-configured runtimes (Java, Python, Node.js, Go).
  - Rapid scaling from zero to match spikes; strict limits on system calls and write access.
- Flexible Environment
  - Runs inside custom Docker containers on managed Compute Engine virtual machines.
  - Supports arbitrary programming runtimes, native OS libraries, and background threads.
- Developers declare configuration, environment variables, and scaling in an app.yaml file.
- Deployment is managed using the Google Cloud CLI with the single command 'gcloud app deploy'.

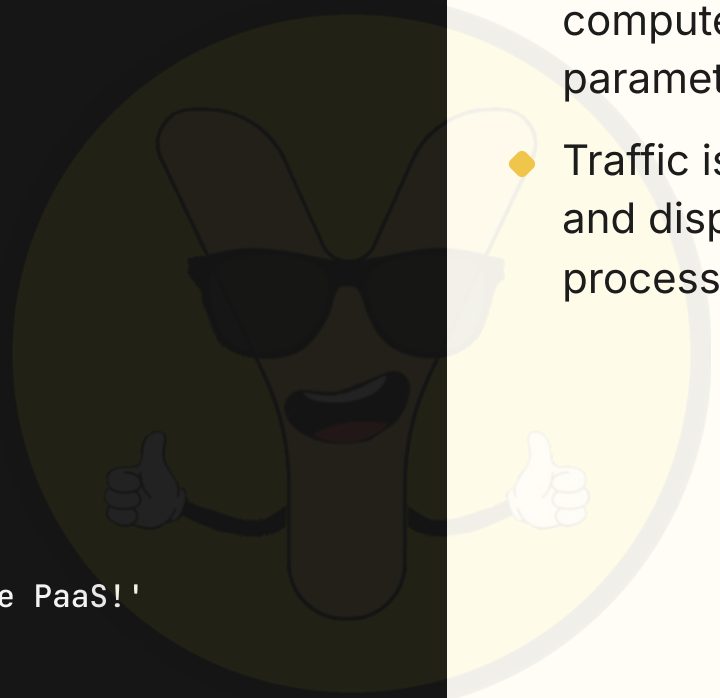
GAE Deployment Configuration and Handler

```
# app.yaml: Configuration file
runtime: python39
instance_class: F1
automatic_scaling:
  min_instances: 0
  max_instances: 5

# main.py: Flask application entrypoint
from flask import Flask
app = Flask(__name__)

@app.route('/')
def index():
    return 'Hello from Google App Engine PaaS!'

if __name__ == '__main__':
    app.run(host='0.0.0.0', port=8080)
```



- The app.yaml file defines runtime language, compute tier, and automated scaling parameters.
- Traffic is received by the managed front-end and dispatched to stateless Python worker processes.

OpenStack Architecture and Overview

OPENSTACK

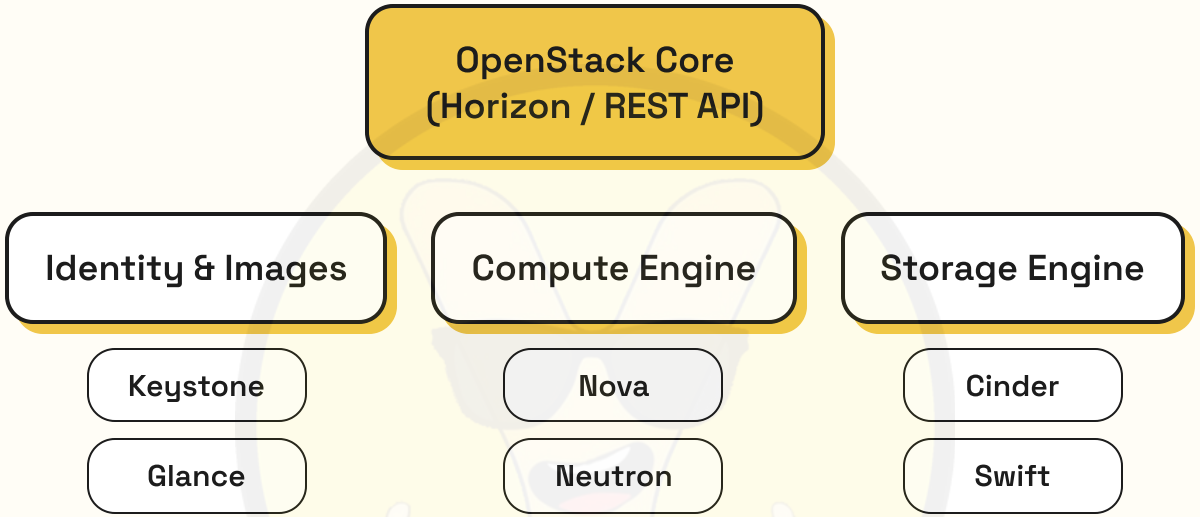
An open-source cloud computing platform that provides Infrastructure as a Service (IaaS) by pooling virtual resources to build and manage public and private clouds.

- Controls large pools of compute, storage, and networking resources via a central dashboard or APIs.
- Allows enterprises to operate private clouds with functionality comparable to public IaaS providers.
- Modular architecture where independent microservices communicate via RESTful APIs and message brokers.
- Natively supports multiple hypervisors including KVM, QEMU, and VMware ESXi.

OpenStack Core Components

Component	Service Name	Role / Functional Purpose
Nova	Compute	Manages lifecycle, scheduling, and provisioning of virtual machines.
Neutron	Networking	Provides Software-Defined Networking, IP management, and VLANs.
Cinder	Block Storage	Provides persistent block volumes to Nova instances.
Swift	Object Storage	Stores unstructured distributed data with high fault tolerance.
Glance	Image Service	Stores and retrieves virtual machine disk and operating system images.
Keystone	Identity	Provides authentication, authorization, and centralized service catalog.
Horizon	Dashboard	Web-based graphical user interface for administration.

OpenStack Architecture Hierarchy



Modular service relationships and logical categorization within OpenStack.

Federation in the Cloud

CLOUD FEDERATION

The interconnection and aggregation of heterogeneous computing resources, platforms, and services across two or more distinct cloud service providers to satisfy dynamic user demand.

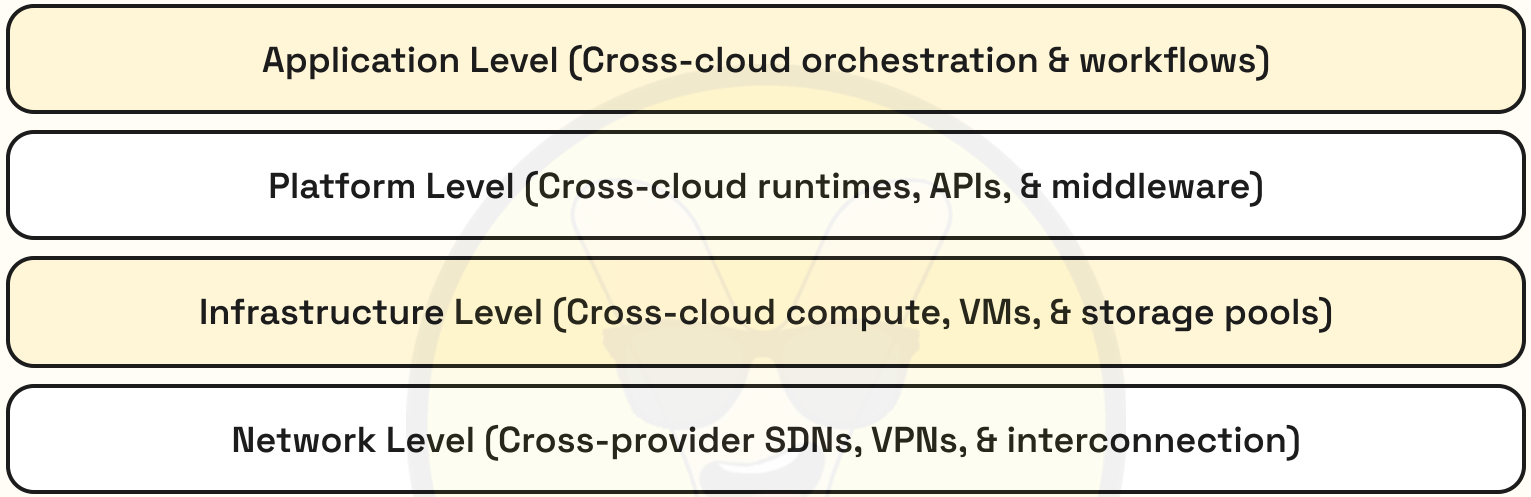
- Prevents vendor lock-in by maintaining standardized interfaces across independent clouds.
- Solves capacity limits by enabling burst workloads across secondary provider boundaries.
- Enables geo-diversity, reducing latency and satisfying regulatory data sovereignty rules.
- Relies on agreed trust models, federated identity, and automated Service Level Agreements (SLAs).

Advantages and Challenges of Cloud Federation

Federated cloud ecosystems balance scalability benefits against systemic complexity:

- Core Advantages
  - Elastic expansion: Absorb sudden traffic spikes via cloud bursting.
  - High availability: Automated disaster recovery and cross-cloud failover.
  - Cost optimization: Dynamic bidding and resource brokerage across providers.
- Implementation Challenges
  - Interoperability: Heterogeneous APIs, data formats, and VM hypervisors.
  - Security: End-to-end identity propagation, data isolation, and shared trust.
  - Performance variance: Unpredictable WAN latency and egress bandwidth costs.

Four Levels of Cloud Federation



Standard four-level abstraction model for multi-provider cloud federation.

Detailed Breakdown of Federation Levels

Federation functions across four operational layers of the cloud stack:

- Network Level: Creates unified, low-latency communication planes using overlay VPNs, BGP peering, and SDNs.
- Infrastructure Level (IaaS): Pools virtual machines, raw block storage, and compute nodes across disparate data centers.
- Platform Level (PaaS): Provides shared runtime environments, common database queries, and distributed middleware.
- Application Level (SaaS): Orchestrates end-to-end user workflows, global microservice calls, and multi-cloud web apps.
- Each level requires distinct security boundaries, translation layers, and identity protocols.

Federated Services and Applications

Key services and use cases deployed across federated cloud architectures:

- Federated Identity Management (FIM)
  - Enables Single Sign-On (SSO) across independent providers using SAML, OAuth 2.0, or OpenID Connect.
- Cloud Bursting
  - Private cloud handles baseline load; bursts overflow traffic to public clouds during peaks.
- Disaster Recovery as a Service (DRaaS)
  - Continuously mirrors critical data and system states to a distinct secondary cloud provider.
- Cloud Brokers
  - Intermediary software dynamically selects optimal cloud vendors based on price, latency, and SLA.

Future of Cloud Federation

Emerging technological directions shaping next-generation federated platforms:

- AI-Driven Brokering: Machine learning algorithms dynamically predict load and allocate workloads across cheapest providers.
- Blockchain for Governance: Smart contracts enforce SLAs, verify billing, and establish trustless multi-party audit trails.
- Edge-to-Cloud Federation: Integrating 5G edge computing nodes, IoT gateways, and centralized hyperscalers.
- Kubernetes Multi-Cluster Orchestration: Container federation (KubeFed) replacing proprietary VM-level federation.
- Sovereign Clouds: Legal-aware federations ensuring data residency compliance across global regulatory zones.

Exam Corner: Unit 5 High-Yield Questions

- **Q1** Explain the MapReduce computational model with its architecture, phases, and a worked Word Count example.
- **Q2** Describe the architectural difference between Type-1 and Type-2 hypervisors, detailing VirtualBox's internal architecture.
- **Q3** Discuss Google App Engine (GAE). Differentiate clearly between its Standard and Flexible programming environments.
- **Q4** List and explain the core components of OpenStack (Nova, Swift, Cinder, Neutron, Glance, Keystone).
- **Q5** Define Cloud Federation. Explain the motivation, benefits, and architectural challenges of federated cloud systems.
- **Q6** Draw and explain the Four Levels of Cloud Federation (Network, Infrastructure, Platform, Application).
- **Q7** Write short notes on: (a) Cloud Bursting, (b) Federated Identity Management, and (c) Role of Cloud Brokers.

Unit 5 done. You are exam-ready.

Revise the Exam corner once more, then open Unit 1 on theycotes.com. Every AKTU subject, unit by unit, free. Notes by Chirag Jain, Ycotes.

