Module IV- Cloud Providers & Software Platforms

Dr. Madhukrishna Priyadarsini

Assistant Professor

NIT Trichy

Outline:

- Globally available public clouds
  - AWS
  - Azure
  - GCP
- Introduction to open source software
  - Eucalyptus
  - Open Nebula
  - OpenStack
  - Apache Cloudstack

# Cloud platforms overview and comparison:

Amazon Web Services (AWS) is a cloud management platform that manages and maintains hardware and infrastructure, reducing the expense and complexity

Microsoft Azure is a cloud computing service that offers a collection of cloud computing services for building, testing, deploying, and managing applications in cloud

Google Cloud Platform (GCP) offers a variety of cloud computing services for building, deploying, scaling, monitoring, and operating a cloud

Comparison parameters

Origin:
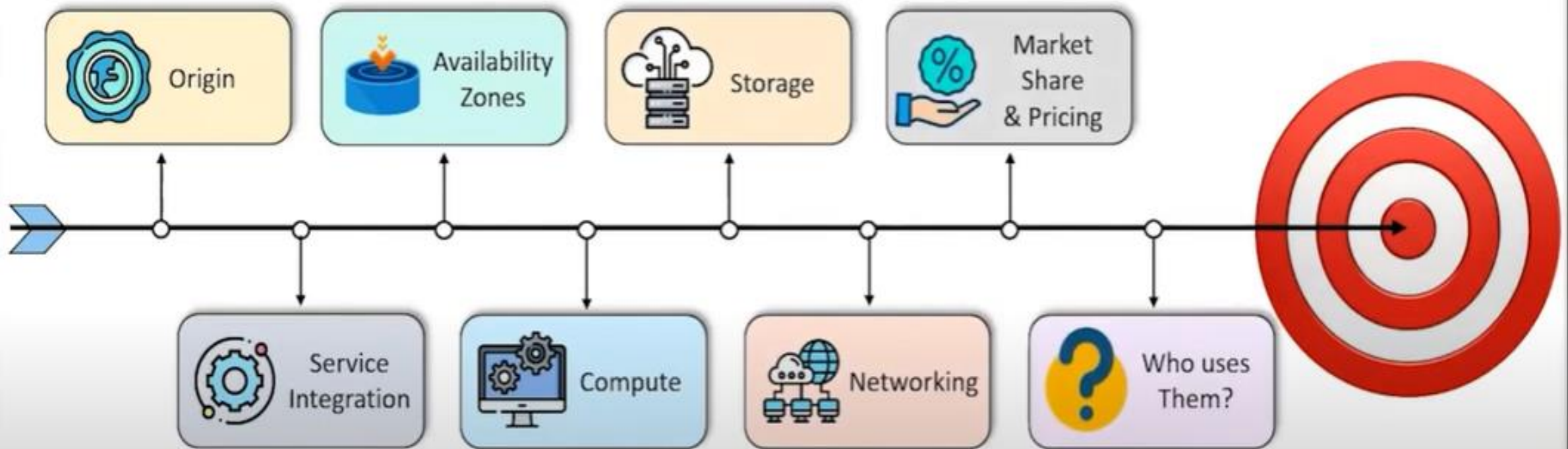
aws	Azure	Google Cloud
Amazon Web Services (AWS) was introduced in **2006** to the market.	• Azure launched its services in **2009**.	• GCP was established in the year **2008**.
It has been **supportive** of the open-source concept.	• The open-source community has a **tense relationship** with it.	• It provides Google Cloud with managed open-source services that are **tightly linked**.
Offers services on a large and complex scale that can be manipulated.	• Support is comparatively low quality.	• Its monthly support is quite expensive.

Service Integration:

aws	Azure	Google Cloud
AWS makes it simple for users to combine services such as Amazon EC2, Amazon S3, Beanstalk, and others.	Azure allows customers to effortlessly combine Azure VMs, Azure App Service, SQL databases, and other services.	Users can utilize GCP to combine services such as Compute Engine, Cloud Storage, and Cloud SQL.

Availability Zone:

aws

AWS being the **first cloud domain**, had more time to build and extend their network.

AWS now offers **66 availability zones**, with an additional 12 on the pipeline.

Azure

- Azure have various locations around the world but has comparatively **less amount of availability zones**.

- Azure is available in 140 countries and is available in **54 regions throughout the world**.

- GCP also have various locations around the world but differs in the no. of availability zones they have.

- Google Cloud Platform is now available in 20 global areas with **73 zones**, with three more on the way.

Compute/Instance:

aws

Elastic Compute Cloud (EC2) offers a wide range of features including many instances, support for both Windows and Linux, and more.

Amazon's container services has Docker, Kubernetes, which automates server and cluster management.

Azure

- **Virtual Machines** includes Linux, Windows Server, operating systems, with better security and Microsoft program integration.

- **Azure uses Virtual Machine Scale Sets** of two container services: Azure Container Service and Container Services.

- **Compute Engine** offers custom and predefined machine types, per-second invoicing, Linux and Windows support, etc.

- For enterprises interested in deploying containers, Google offers the **Kubernetes Engine**.

Storage:

aws

Simple Storage Service (S3), Elastic Block Storage (EBS), and Elastic File System (EFS) are among AWS' storage offerings.

Relational Database Service (RDS), DynamoDB NoSQL database, etc are all SQL-compatible databases offered by Amazon.

Azure

- Blob Storage, Queue Storage, File Storage, and Disk Storage are among Microsoft Azure's core storage services.

- SQL Database, MySQL Database, and PostgreSQL Database are the three SQL-based choices for Azure.

- GCP offers an increasing number of storage options like Cloud Storage.

- It offers the SQL-based Cloud SQL and Cloud Spanner, a relational database built for mission-critical workloads.

Networking:

AWS uses Amazon Virtual Private Cloud(VPC).

Azure uses Azure Virtual Network(VNET).

GCP uses Cloud Virtual Network.

Market Share & Pricing:

aws	**Azure**	Google
AWS covers **32% of the market**.	• Azure at **19% covers the market**.	• And Google at **7% covers the global Cloud Market**.
In AWS, pricing is done on yearly basis. Basic instance costs around **US$69 per month** and the largest instance around **US$3.97/hour**.	• Here it is done in minutes basis. The smallest instance The largest instance costs around **US$70/month** and **US$6.79/hour**.	• Here also it is done in minute basis. It provides basic instance around **US$52/month** and the largest instance around **US$5.32/hour**.
It provides spot instances, reserved instances and dedicated hosts.	• It provides developers discount and hybrid benefits.	• It gives you sustained discount and preemptible discount.

Market shares of leading cloud service providers in Q2 2023*

Who Uses Them?
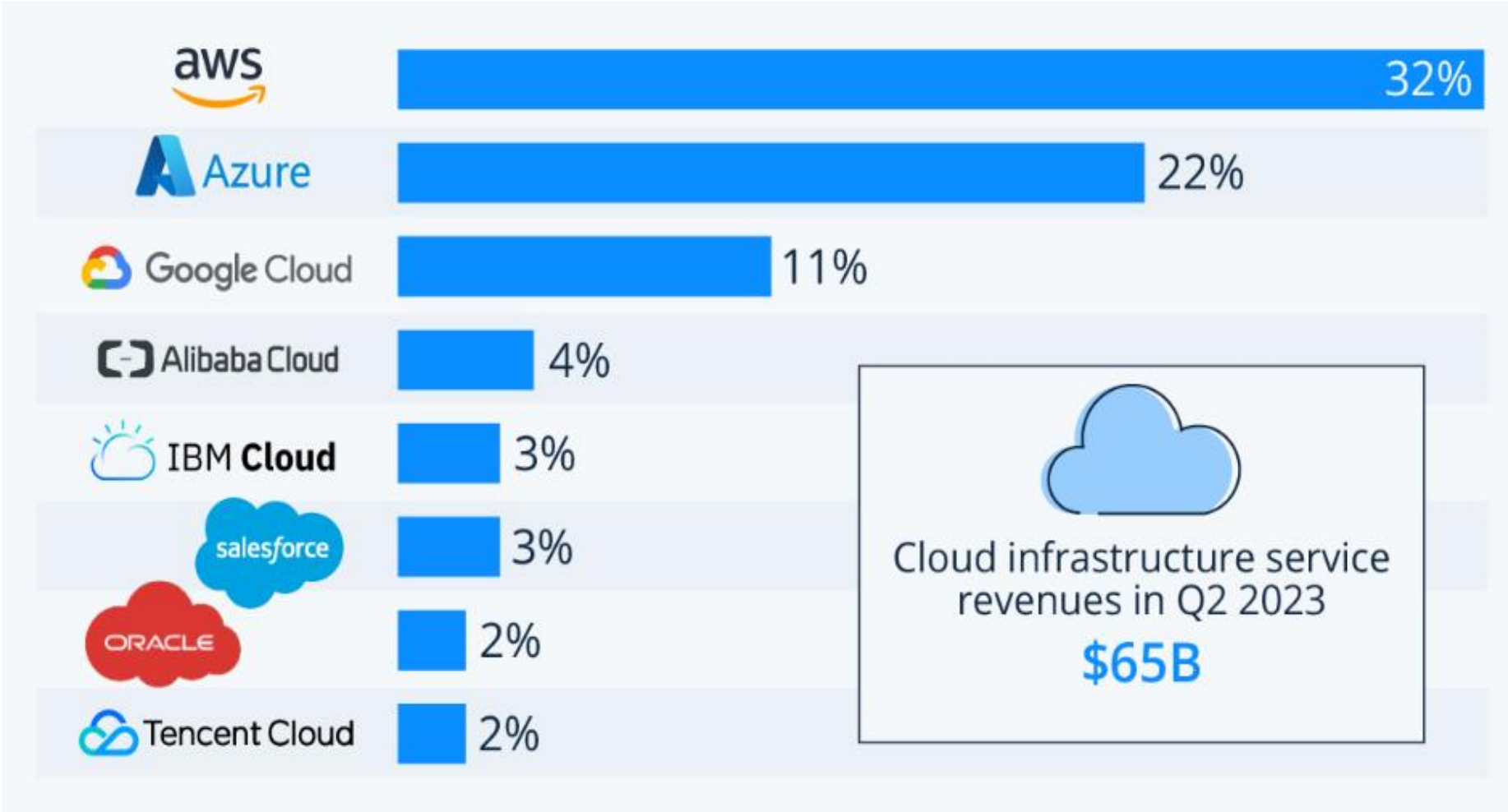
aws	Azure	Google Cloud
Netflix, Airbnb, Unilever, BMW, Samsung, MI, Zynga, and more high-profile and well-known customers use AWS.	Johnson Controls, Polycom, Fujifilm, HP, Honeywell, Apple, and others are among the key clients of Azure.	HSBC, PayPal, 20th Century Fox, Bloomberg, Dominos, and others are among Google Cloud's many clients.

AWS Vs. Azure Vs. GCP Pros

aws	Azure	Google Cloud Platform
AWS is a Stablish market leader.	Azure is Open to Hybrid Cloud systems.	GCP specializes in high compute offerings like Big data, Machine Learning.
High Transfer Stability: Minimal data loss during server and storage transfer.	Easy integration with Microsoft tools and software.	Easy integration with other GCP Services like Compute Engine, Kubernetes Engine, or App Engine.
Easy Availability of Data: users can choose to store data close to their location.	Azure has a more profound knowledge of enterprise needs.	Well-detailed documentation, including an API reference guide.
78 ~~regions~~ regions available across 140 countries (which is higher in all cloud giants)	Higher Availability Zones (106) with 24 Countries.	Outstanding reputation in the open-source community

AWS Vs. Azure Vs. GCP Cons

AWS vs Azure vs GCP Cons.		
aws	Azure	Google Cloud Platform
Incomplete and weak Hybrid Strategy	Integration with non-Microsoft is complex.	Google's App Engine is limited to Java, Python, PHP and Google Go.
No demonstrated support for hybrid cloud outputs is still in its nascence	Restrictive Platform, Less flexible with non-windows server platforms	The cost of downloading data from Google Cloud Storage is relatively high
Large and Complex scale offerings that can potentially manipulate.	Low-Quality Support	Quite costly support fee of about $150 per month for the silver class, which is the most basic of services

Open Source Software:

- Open Source Cloud Software is used to build private, public, and hybrid clouds which are compatible with the already available cloud platforms in market.

- In other words, the open source cloud software is interoperable with any back-end platform and can easily be migrated to a different IT infrastructure/environment.

- Majorly used open source cloud software are:

1. Eucalyptus
2. Open Nebula
3. Open Stack
4. Apache Cloudstack

Open Source Cloud Infrastructure

- Simple
  - ❖ Transparent-> need to see in to the 'cloud'
  - ❖ Scalable-> complexity often limits scalability

- Extensible
  - ❖ New application classes and service classes may require new features
  - ❖ Clouds are new-> needs to extend while retaining useful features

- Commodity-based
  - ❖ Must leverage extensive catalog of open source software offerings

- Easy
  - ❖ To install-> system administration time is expensive
  - ❖ To maintain-> system administration time is really expensive

Open Source Cloud Anatomy

- Extensibility
  - Simple architecture and open internal APIs
- Client-side interface
  - Amazon's AWS interface and functionality (familiar and testable)
- Compute
  - Multiple hypervisors and image formats
- Networking
  - Virtual private network (VPN) per cloud allocation
- Storage
  - Cloud storage models using generic OS functionality
- Security
  - Must be compatible with local security policies
- Packaging, installation, maintenance
 - Minimal intrusiveness, maximal portability

1- Eucalyptus (Elastic Utility Computing Architecture Linking Your Programs To Useful Systems)

- Eucalyptus is open source software for building AWS-compatible private and hybrid clouds.

- As an Infrastructure as a Service product, Eucalyptus allows you to flexibly provision your own collections of resources (both compute and storage), on an as-needed basis.

- It is a web services based implementation of elastic/utility/cloud computing infrastructure
  - Linux image hosting ala Amazon

Goals of Eucalyptus:

- Foster greater understanding and uptake of cloud computing
  - Provide a vehicle for extending what is known about the utility model of computing
- Experimentation vehicle prior to buying commercial services
  - Provide development, debugging, and "tech preview" platform for public clouds
- Homogenize local IT environment with public clouds
  - AWS functionality locally makes moving using Amazon AWS easier, cheaper, and more sustainable
- Provide a basic software development platform for the open source community
  - E.g., the "Linux experience"
- Not designed as a replacement technology for AWS or any other public cloud service

Architecture:
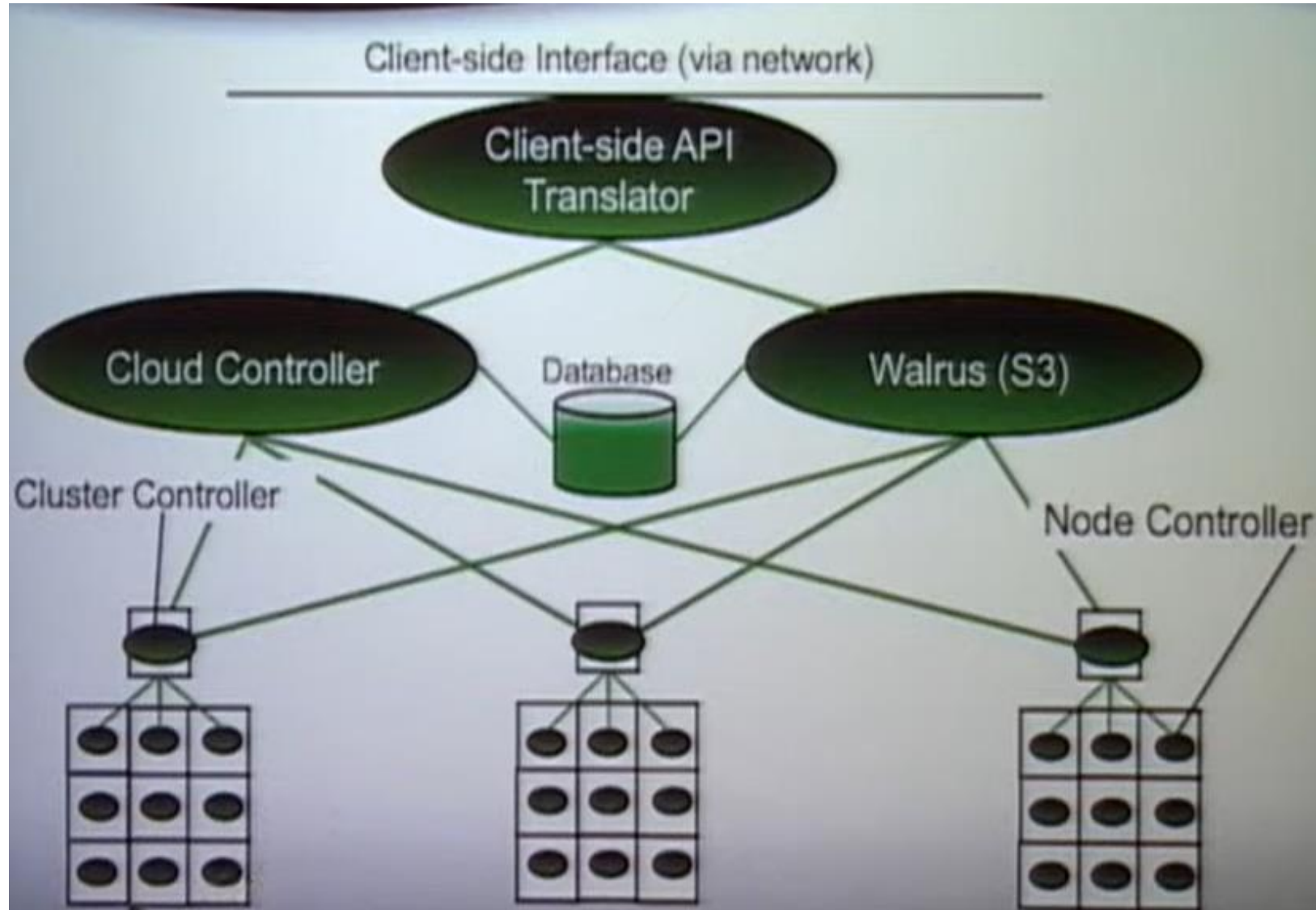
Client side Interface

- Interface is based on Amazon's published WSDL
  - EC2, S3, EBS
  - Availability zones correspond to individual clusters
  - Uses either Amazon tool or Eucalyptus replacements
  - Web services and REST interfaces
- Looking at Sun's new API
  - Initial set of 14 calls looks interesting
  - Need well-defined error codes
- System administration is different
  - Eucalyptus defines its own cloud admin tool set for user accounting and cloud management

Networking

- Eucalyptus does not assume that all worker nodes will have publicly routable IP addresses.
  - All cloud images have access to a private network interface
- Three networking modes:
  - System Mode-> hosted VMs use local DHCP to get IP addresses (Eucalyptus does not control the routing)
  - Static Mode-> cloud administrator configures a set of usable public IP addresses (Eucalyptus hands them out via DHCP)
  - Managed Mode-> all VMs get a private address and Eucalyptus uses IP tables for local and remote routing (Elastic Ips and security groups only work in managed mode)
- Availability zone approach fits with Amazon's high level semantics

Security

- All Eucalyptus components use WS-security for authentication
  - Encryption of inter component communication is not enabled by default (configuration option)
- SSH key generation and installation is implemented
- User sign-up is web-based
  - User specifies a password and submits sign-up request
  - Certificate is generated but withheld until admin approves the request
  - User gains access to the certificate through password-protected web page

# Packaging and installation

- From source
  - Tarball download
  - Ubuntu universe (sudo apt-get install Eucalyptus)
    - Distributing updates via apt-get as well
- From binary
  - Ubuntu
  - Debian "lenny" and "squeeze"
  - CentOS RPM (catches most of the RedHat derivatives)
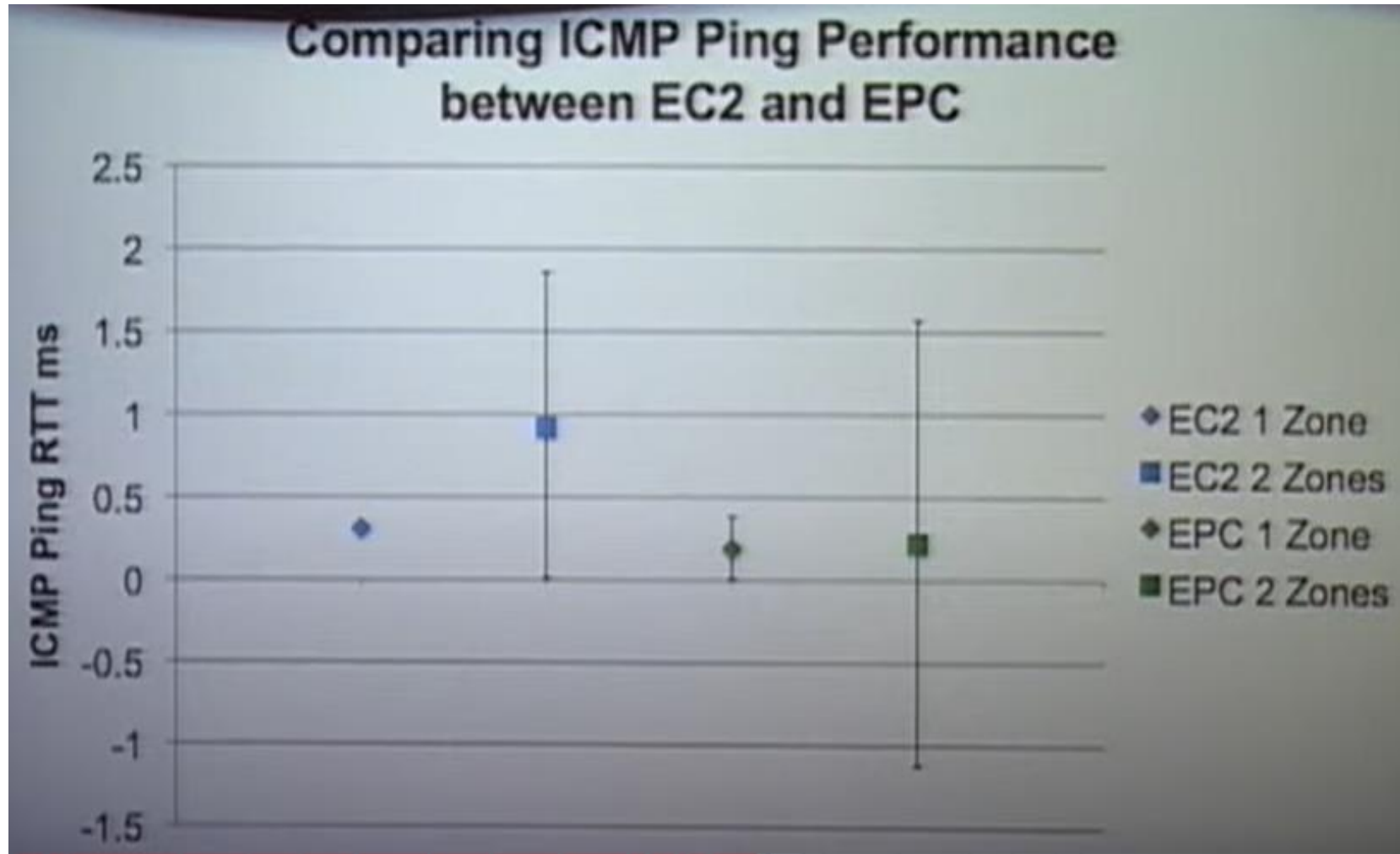 - OpenSUSE (poll based)

Eucalyptus Components
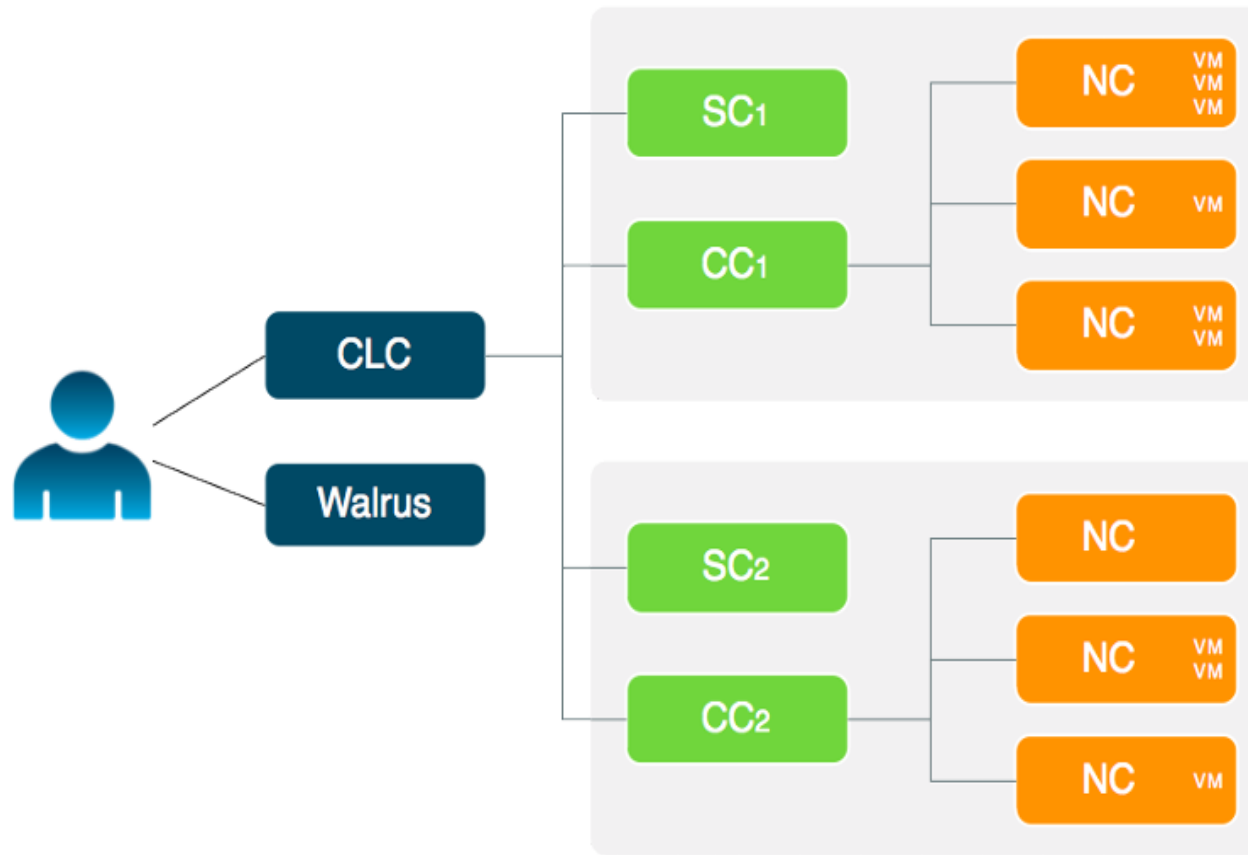
1. Cloud Controller (CLC)
2. Walrus
3. Cluster Controller (CC)
4. Storage Controller (SC)
5. VM Broker
6. Node Controller(NC)

Bandwidth Performance
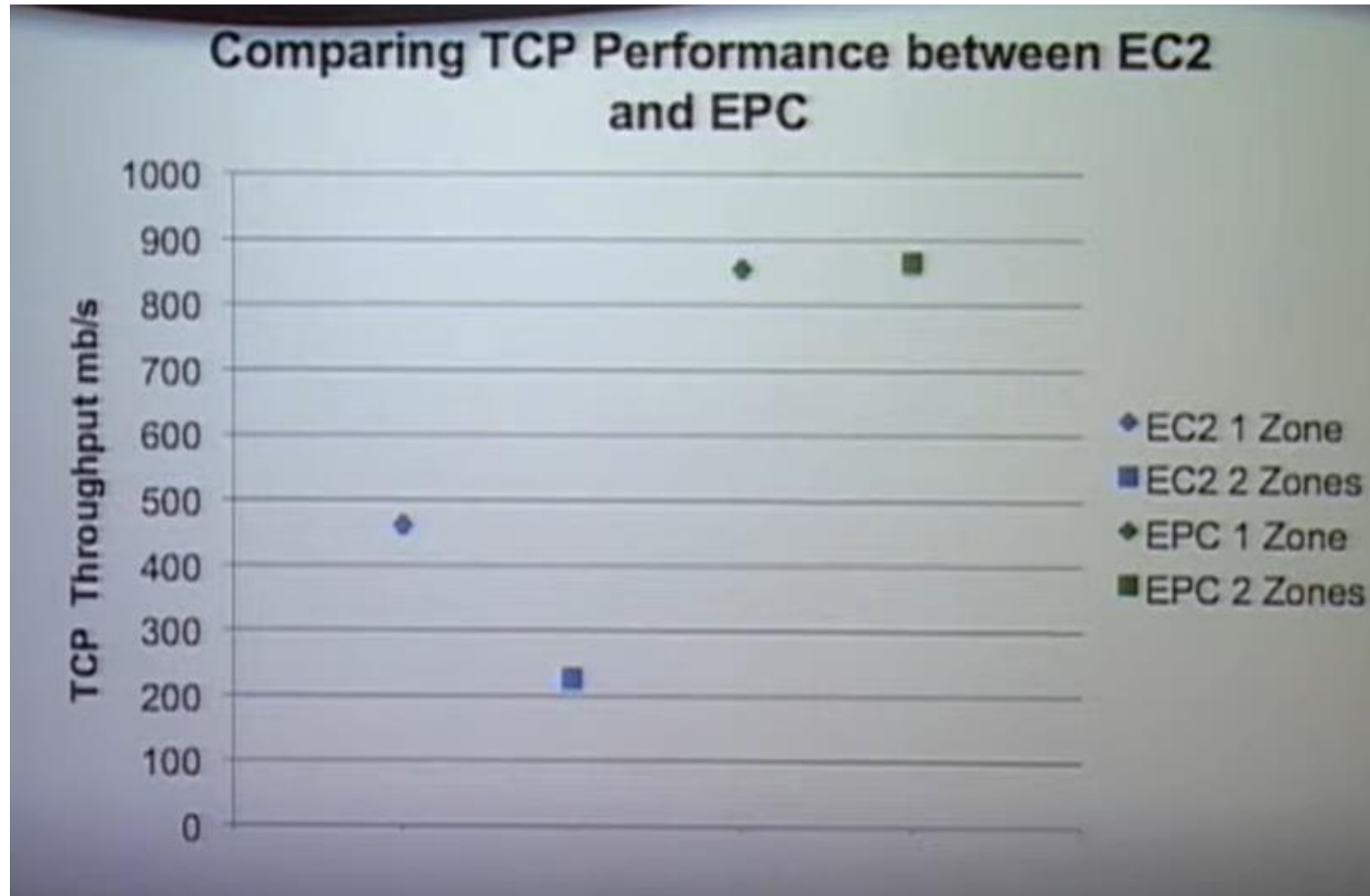
RTT Performance

Eucalyptus Ecosystem

- AppScale
  - Google App engine atop Eucalyptus
  - Multiple scalable database backends
- RightScale
  - Local enterprises focused on providing client tools as SaaS hosed in AWS
  - Current release includes registration protocol for RightScale
- CohesiveFT
  - Elastic server, personal edition targets Eucalyptus as the on-premise platform
- rPath
  - "Appliancizing" Eucalyptus

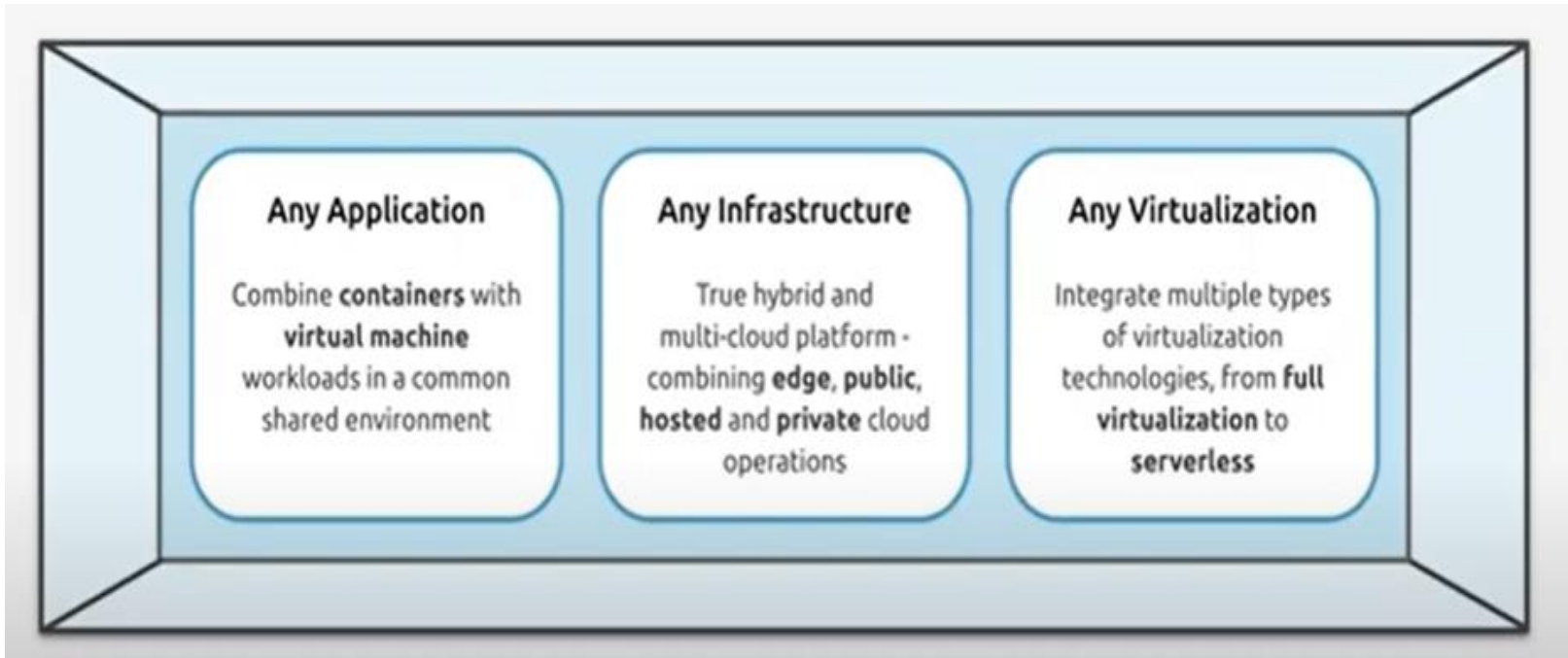# Installation, Configuration, and Working Environment of Eucalyptus

1. Configure the dependencies
2. Disable Firewall
3. Configure Network Time Protocol (NTP)
4. Configure Java
5. Configure Mail Transport Agent (MTP) server
6. Install Repository
7. Install Eucalyptus Release Package
8. Configure Eucalyptus
9. Configure Network modes
10. Create Network JSON file
11. Create scheduling policy
12. Start Eucalyptus

2- Open Nebula (Sponsored by Open Nebula Systems)

- It is an open source cloud computing platform for managing heterogeneous data center, public cloud, and edge computing infrastructure resources. Basically it is developed for managing the enterprise.

- Open Nebula manages on-premise and remote virtual infrastructure to build private, public, or hybrid implementations of IaaS platform.

- Two main implementations of Open Nebula are; data center virtualization and cloud deployments.

- The major features of Open Nebula are:
  - Self-service
  - Provisioning
  - Multi-tenancy
  - Elasticity

What is Open Nebula?

A "Single Pane of Glass" to bring real freedom to your enterprise cloud.

Any Application

Combine **containers** with **virtual machine** workloads in a common shared environment

Any Infrastructure

True hybrid and multi-cloud platform - combining **edge, public, hosted** and **private** cloud operations

Any Virtualization

Integrate multiple types of virtualization technologies, from **full virtualization** to **serverless**

✓ Avoids "Vendor Lock-in"
✓ Minimizes complexity

✓ Reduces resource consumption
✓ Slashes operating cost

Who is using Open Nebula?

A widely adopted solution to build clouds
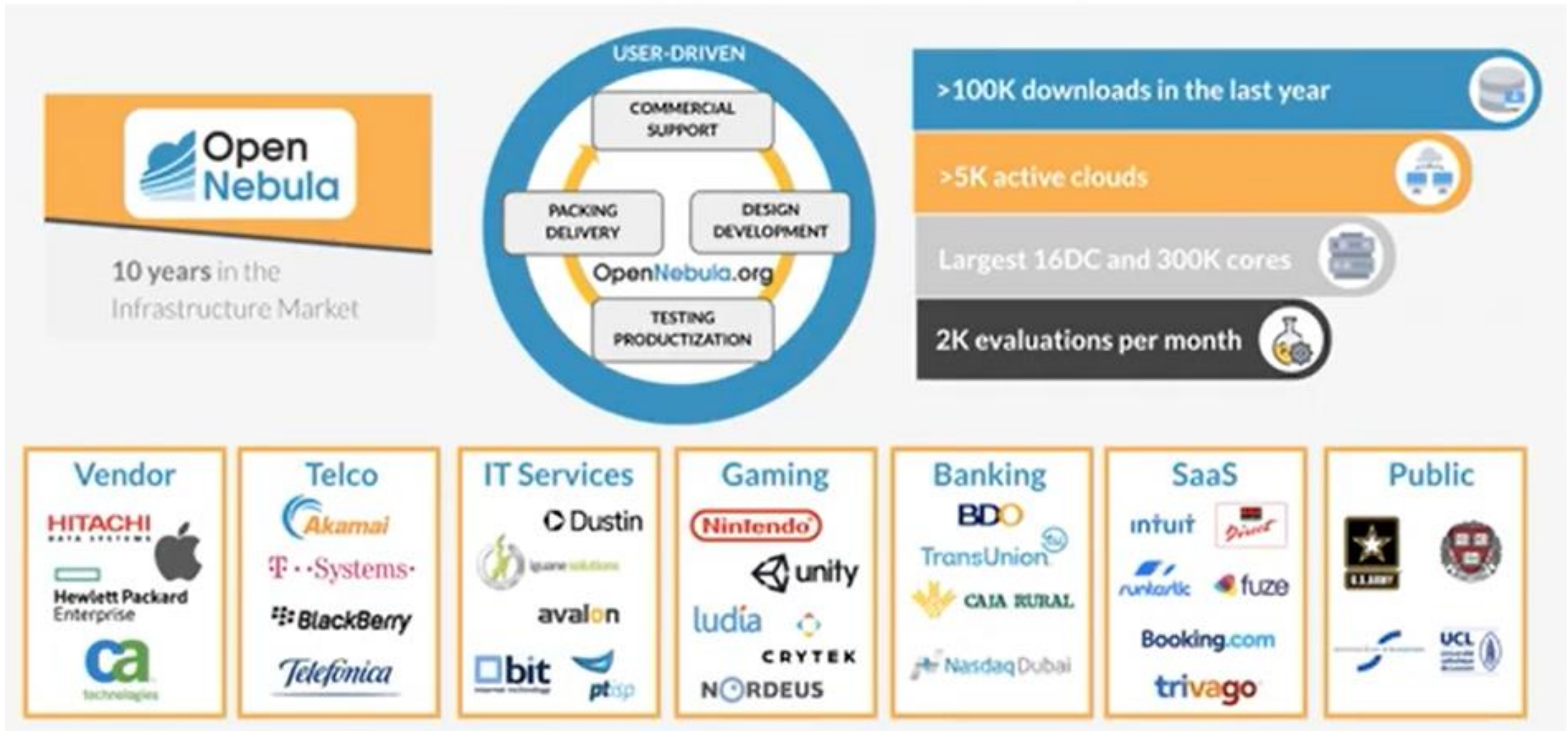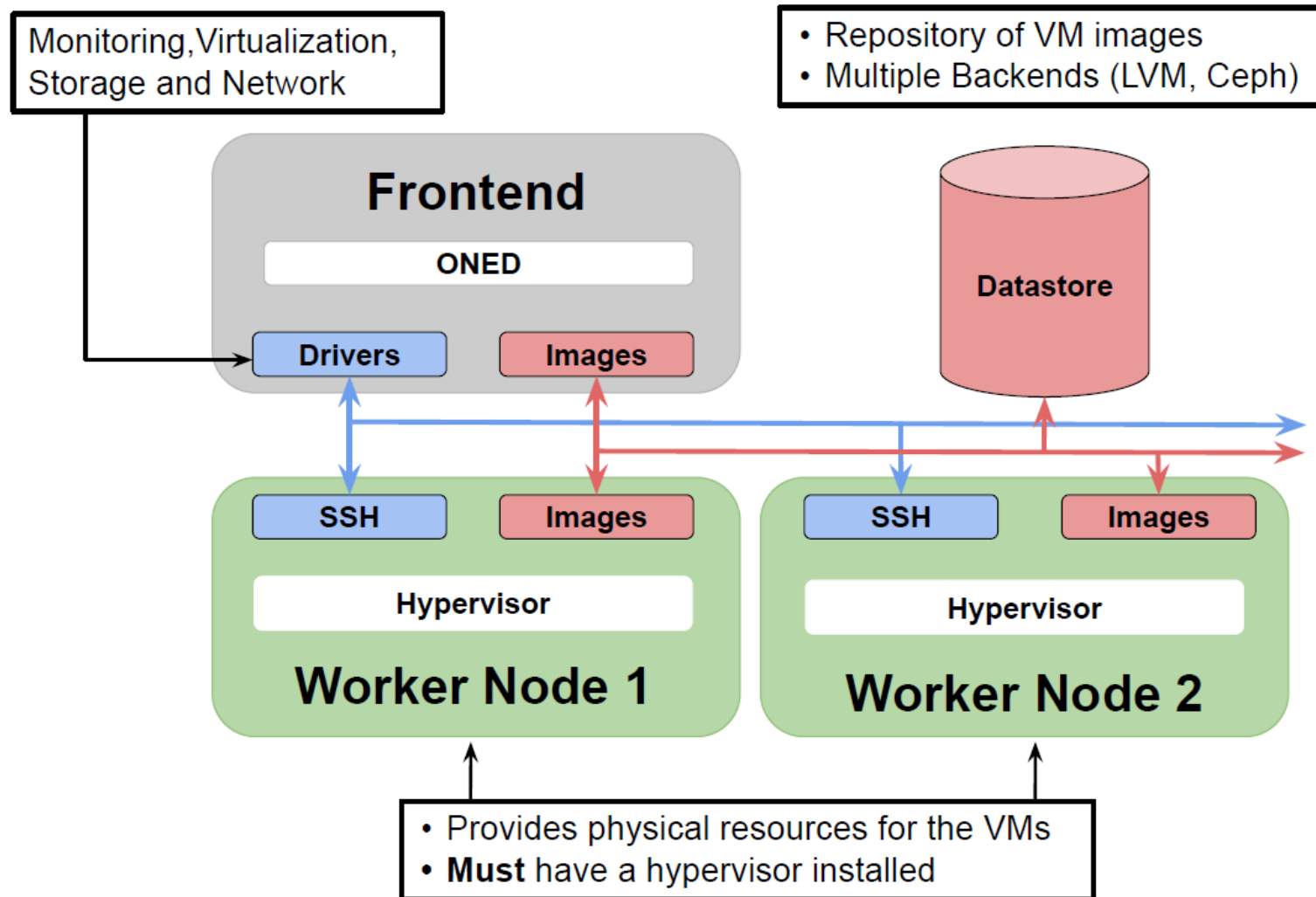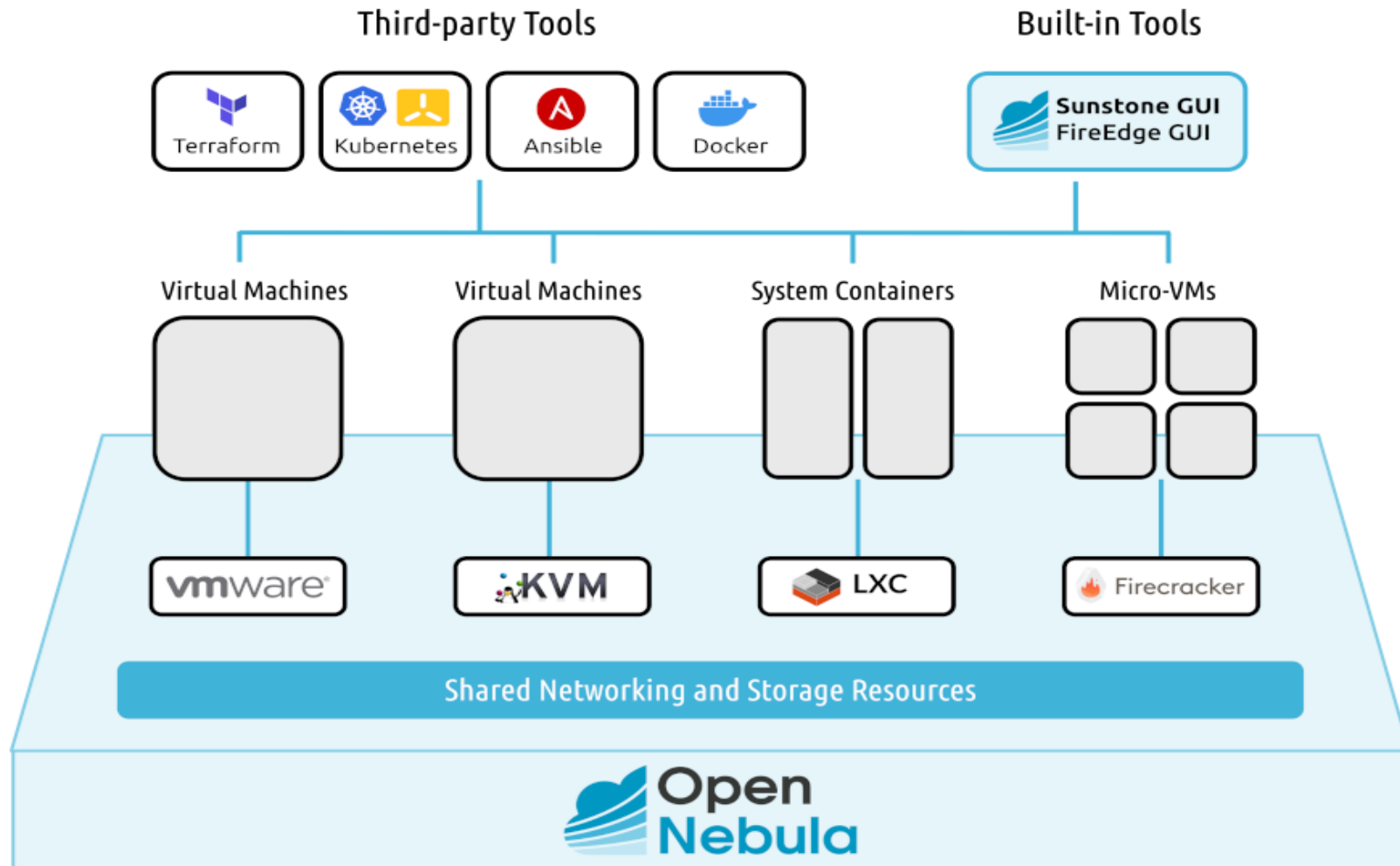
Open Nebula Subscription

- ❖ **Enterprise Edition** provides customers a hardened LTS version with software patches, bug fixes, and minor enhancements
    - ✓ Enterprise tool-> OneScape

- ❖ **Support** provided through a private and secure support portal-> 9*5 or 24*7

- ❖ Professional Services
    - ✓ Cloud deployment
    - ✓ Upgrade
    - ✓ Training
    - ✓ Consulting/Engineering
- ❖ Knowledge Base with best practices, step-by-step guides, and use cases
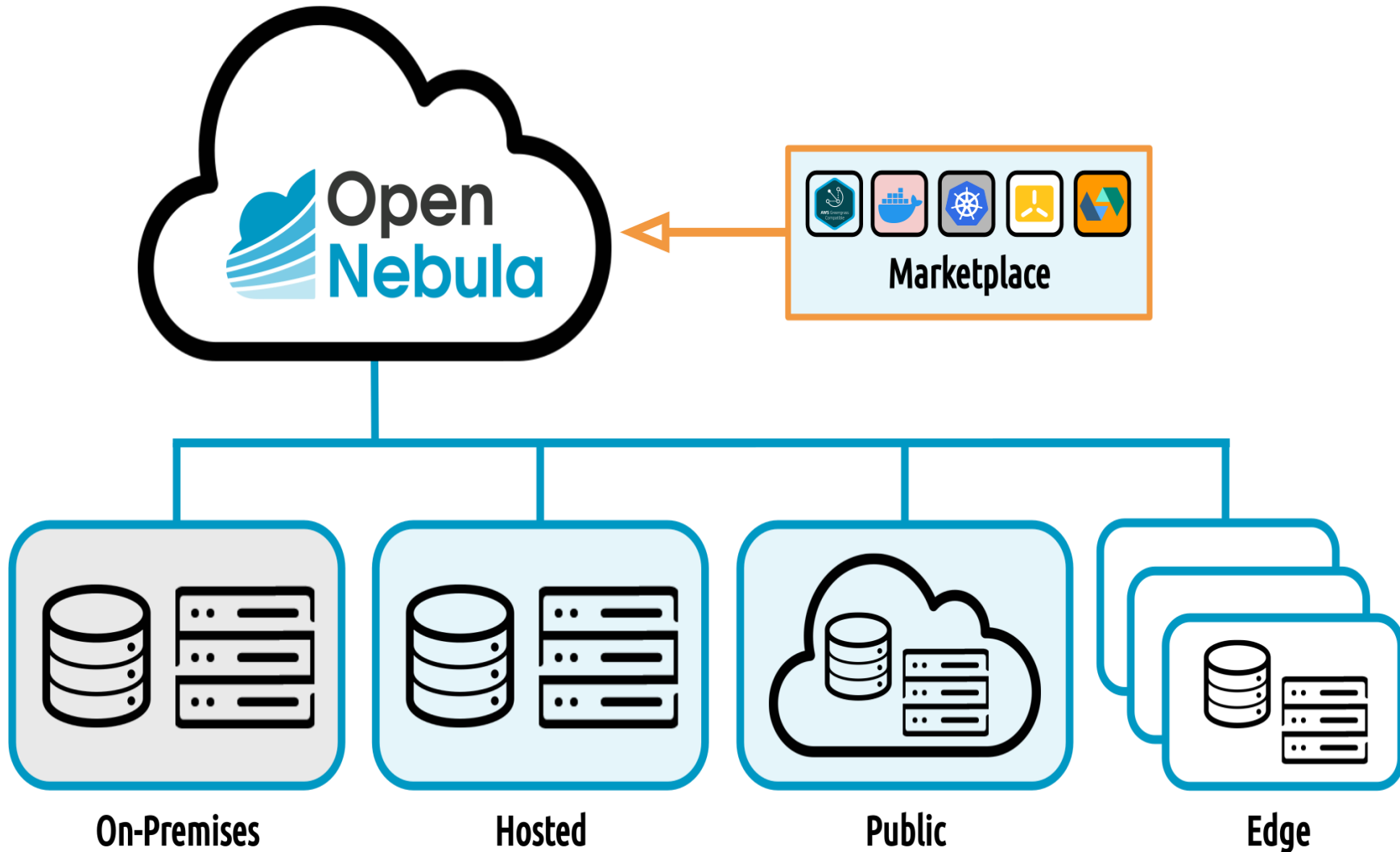
Open Nebula's Open Architecture

Software platform to build enterprise clouds for DevOps and Development workloads
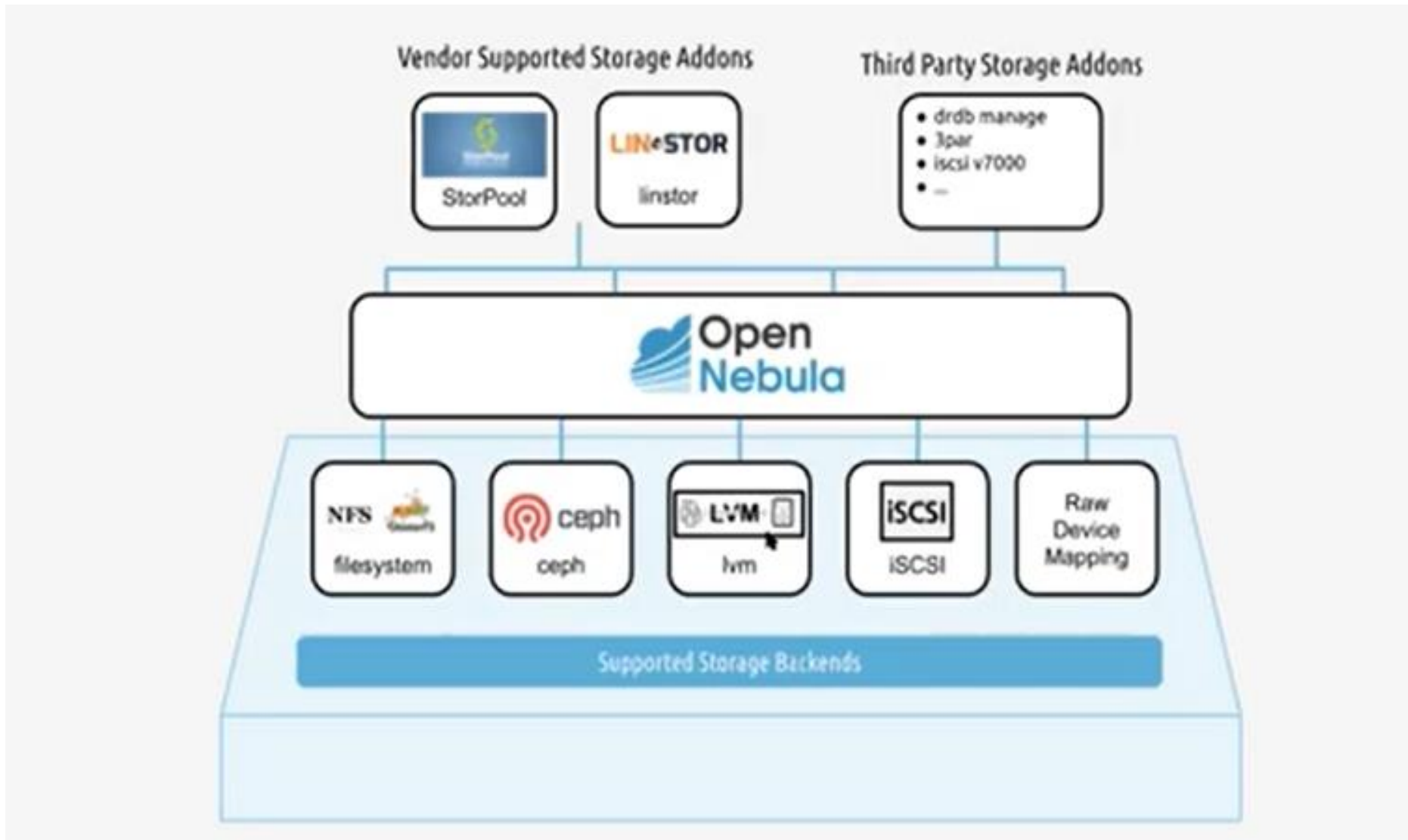
Third-party Tools

Terraform Kubernetes Ansible Docker

Built-in Tools

Sunstone GUI
FireEdge GUI

Virtual Machines Virtual Machines System Containers Micro-VMs

vmware KVM LXC Firecracker

Shared Networking and Storage Resources

Open
Nebula

How is Open Nebula being used?

Marketplace

On-Premises   Hosted   Public   Edge

Storage Backends

# ONEedge.io (Expanding open source cloud computing to the edge)
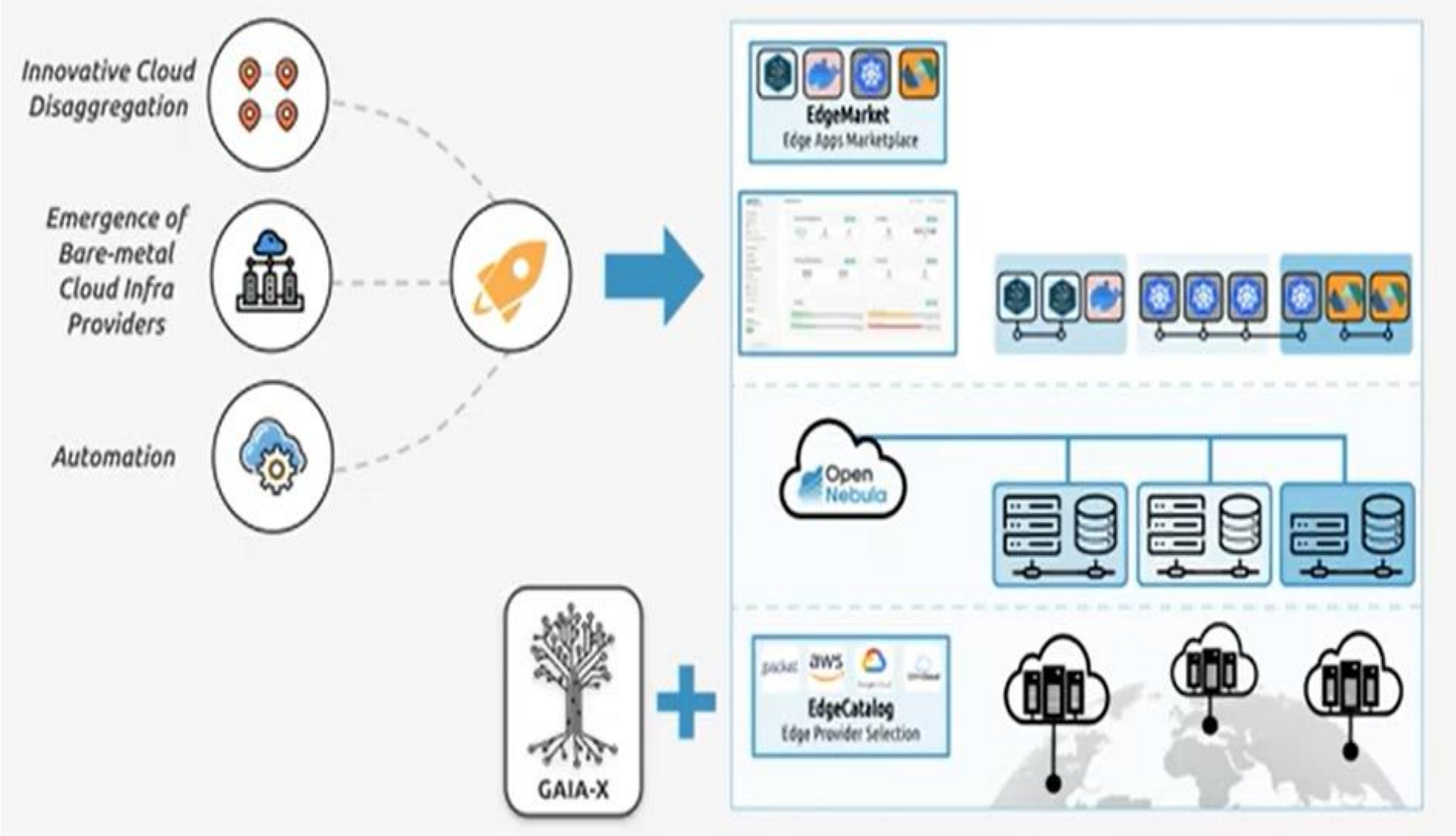
Installation, Configuration, and Working Environment of Open Nebula

- Install the repo: `/etc/yum.repos.d/opennebula.repo`
- Install packages: `yum install opennebula …`
- Install gems: `/usr/share/one/install_gems`
- Enable OpenNebula services
- Configure Storage
- Configure Network
- Configure passwordless ssh access
- OpenNebula configuration files

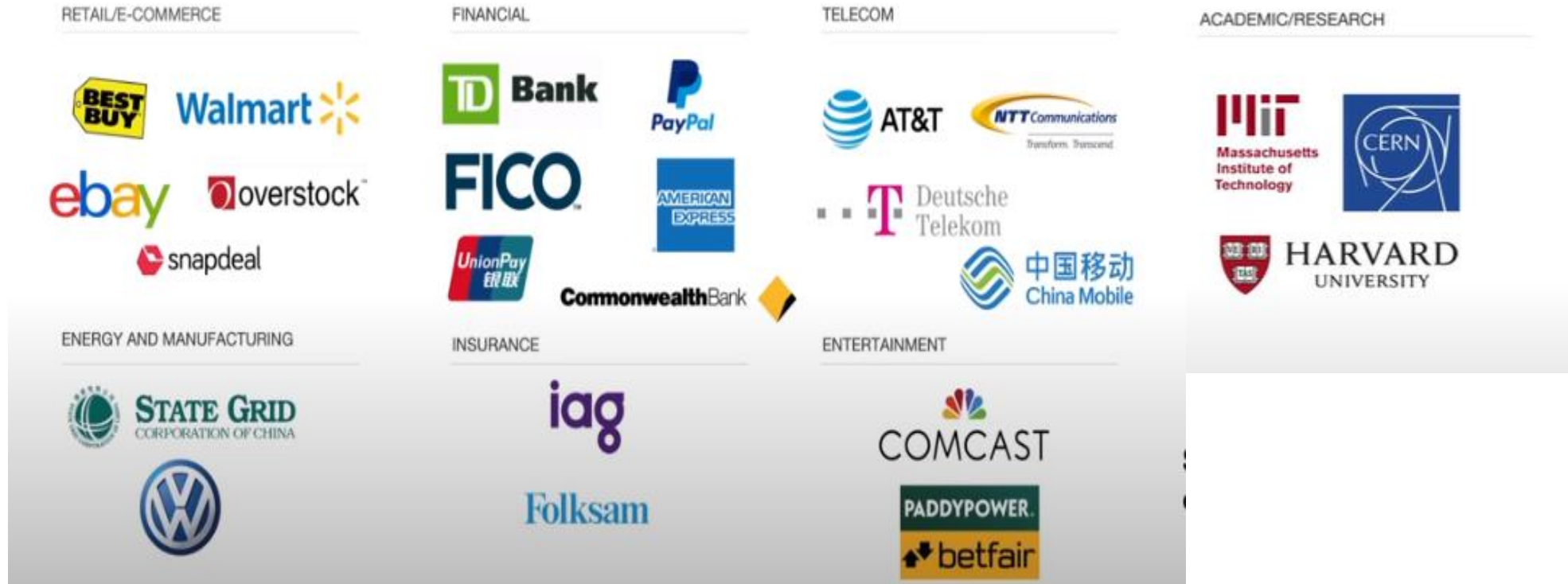# 3- OpenStack (By Rackspace Technology)
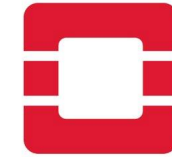
- OpenStack is an open source software that provides cloud infrastructure for virtual machines, bare metals, and containers.
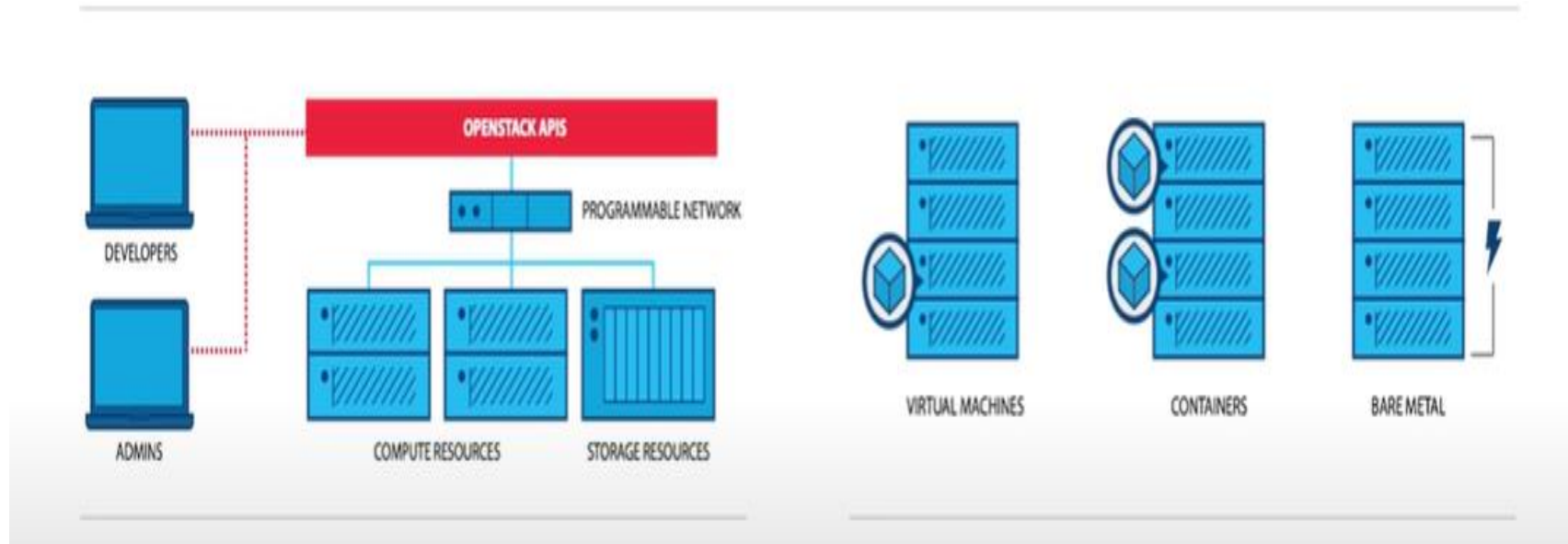
- Programmable infrastructure that lays a common set of APIs on top of compute, networking, and storage.

- One platform for virtual machines, containers, and bare metals.

Who uses OpenStack?

Open Stack Features:
- ✓ Scalable
- ✓ Reliable
- ✓ Flexible
- ✓ Provides control

OpenStack Deployment Models
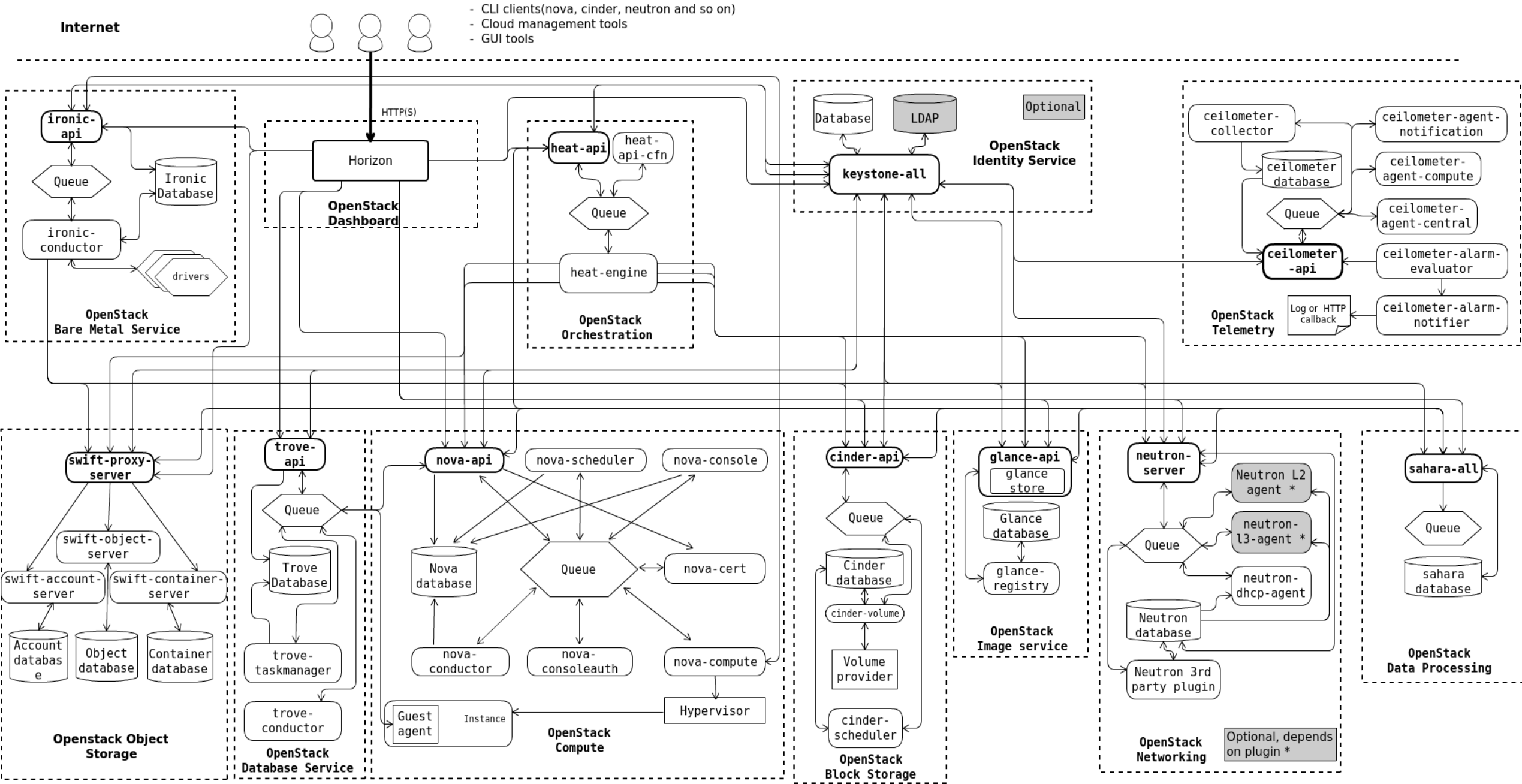
Open Stack on-premises

Open Stack Appliances

Open Stack Hosted Cloud

Open Stack-as-a-service

OpenStack Architecture

OpenStack Architecture

- OpenStack Components

Horizon (Dashboard)

- Horizon is a canonical implementation of OpenStack's dashboard, which provides a web-based user interface to OpenStack services.

Nova (Compute)

- Nova is the OpenStack project that provides a way to provision compute instances. Nova supports creating virtual machines, bare metal servers, and has limited support for system containers. Nova runs as a set of daemons on top of existing Linux servers to provide that service.

Swift (Object Store)

- Swift is a highly available, distributed, eventually consistent object/blob store. Organizations can use Swift to store lots of data efficiently, safely, and cheaply.

Cinder (Block Storage)

- Cinder is the OpenStack block storage service for providing volumes to Nova virtual machines, Ironic bare metal hosts, containers and more.

Neutron (Networking)

- Neutron is an OpenStack object to provide "network connectivity as a service" between interface devices managed by other OpenStack services (e.g., Nova). It implements the OpenStack networking API.

Heat (Orchestration)

- Heat is a service to orchestrate composite cloud applications using a declarative template format through an OpenStack-native REST API. A Heat template describes the infrastructure for a cloud application in text files which are readable and writable by humans, and can be managed by version control tools.

Glance (Image)

- The image service (Glance) project provides a service where users can upload and discovers data assets that are meant to be used with other services. This currently includes images and meta-data definitions. Glance image services include discovery, registering, and retrieving virtual machine (VM) images. Glance has a restful API that allows querying of VM image metadata as well as retrieval of the actual image.

Ceilometer (Telemetry)

- OpenStack ceilometer (Telemetry) offers a single point of contact for many billing systems, facilitating each counter they require to build customer billing around every future and current component of OpenStack.

Keystone (Identity)

- Keystone is the OpenStack service that provides API client authentication, service discovery, and distributed multi-tenant authorization by implementing OpenStack's identity API.

The OpenStack Landscape

CLIENT TOOLS

CLIs
OpenStackClient

SDKs
Python SDK

INTEGRATION ENABLERS

CONTAINER SERVICES
Kuryr

NFV
Tacker

OPENSTACK

WEB FRONTEND
Horizon

API PROXIES
EC2API

WORKLOAD PROVISIONING
Magnum   Trove
Sahara

APPLICATION LIFECYCLE
Murano   Freezer
Solum   Masakari

ORCHESTRATION
Heat   Mistral   Aodh
Senlin   Zaqar   Blazar

COMPUTE

VIRTUAL MACHINES
Nova

CONTAINERS
Zun

STORAGE

OBJECT
Swift

BLOCK
Cinder

FILE
Manila

NETWORKING

SDN   **LOAD BALANCING**   **DNS**
Neutron   Octavia   Designate

HARDWARE LIFECYCLE

BARE METAL
Ironic

ACCELERATORS
Cyborg

SHARED SERVICES

IDENTITY
Keystone

SCHEDULING
Placement

IMAGES
Glance

SECRETS
Barbican

OPERATIONS TOOLING

MONITORING SERVICES
Ceilometer
Monasca   Venus

RESOURCE OPTIMIZATION
Watcher   Vitrage

BILLING / BUSINESS LOGIC
Adjutant   CloudKitty

TESTING / BENCHMARK
Tempest   Rally

LIFECYCLE MANAGEMENT

DEPLOYMENT / LIFECYCLE TOOLS
Kolla-Ansible   OpenStack-Charms   Bifrost   Kayobe
OpenStack-Helm   OpenStack-Ansible   OpenStack-Chef

PACKAGING RECIPES FOR...
Puppet
Containers (LOCI, Kolla)

openstack

How to install and Configure OpenStack?

1. Create the environment for OpenStack

2. Install OpenStack packages

3. Finalize the installation

4. Install OpenStack services

5. Launch an OpenStack instance

6. Create the VM, storage, database etc.

4- Apache CloudStack

- Apache CloudStack is a scalable, multi-tenant, open source, purpose-built, cloud orchestration platform for delivering turnkey IaaS as a cloud.
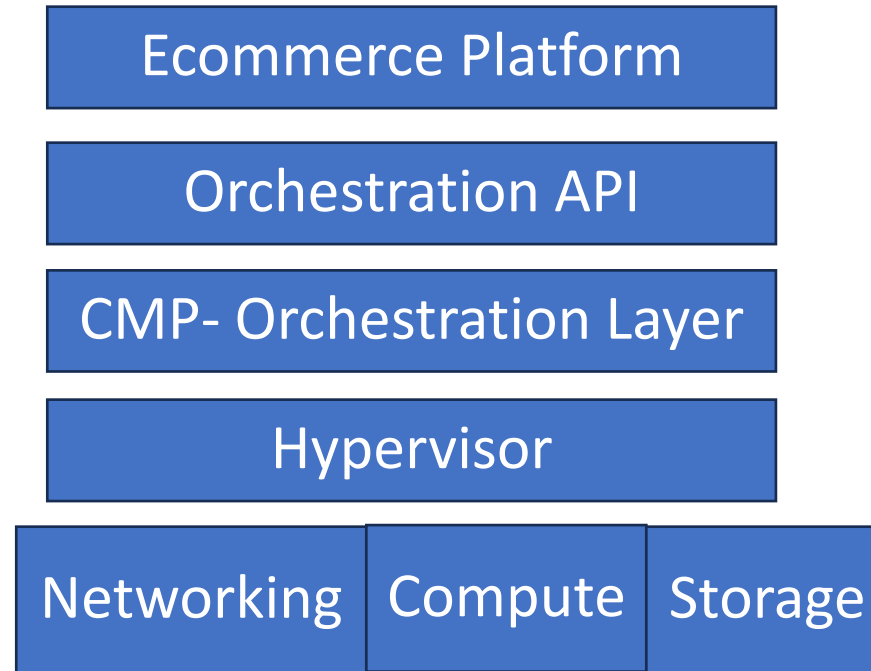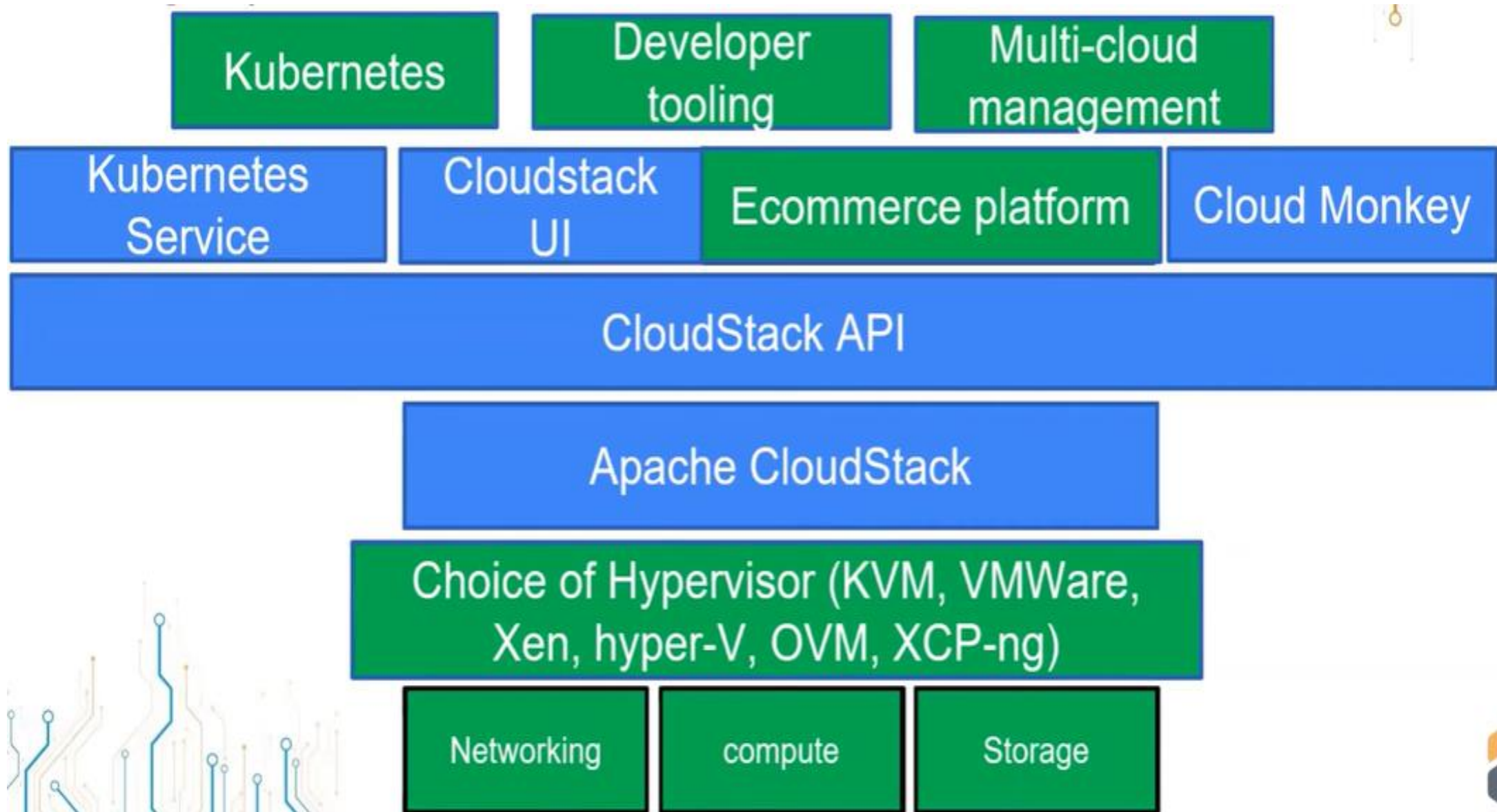
Key features:

- Integrated end-to-end IaaS product

- Proven at scale, wide-spread adoption

- Rapid time to value

- Low implementation and operation cost

- Truly multi-tenant

- Free and Open Source

- Easy integration

Available Functionalities:

- User self-service
  - Web UI, Command line, REST-based API
- Broad and deep hypervisor support
  - Xen-server, KVM, VMWare, Oracle VM, Hyper-V, XCP-ng
- Support for Kubernetes cluster
- Enterprise grade tenant virtual networking model
- Scalable architecture
  - Supports thousands of hosts and virtual machine guests
  - Largest known production cloud 35k+ physical hosts
- High availability
  - Configuration that provide automatic failover for virtual machines
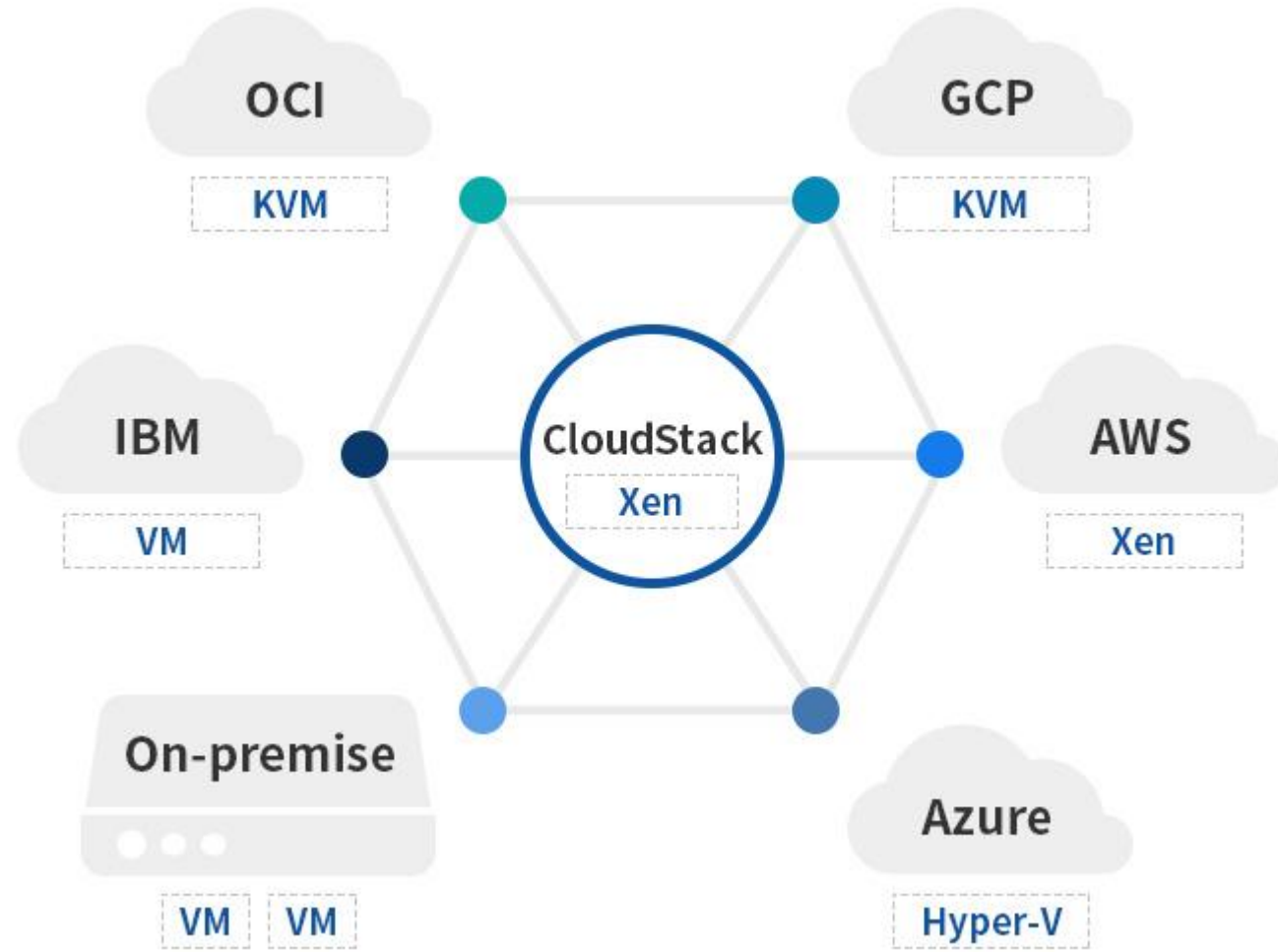
Architecture

| Ecommerce Platform |
| Orchestration API |
| CMP- Orchestration Layer |
| Hypervisor |

| Networking | Compute | Storage |

Organizations using Apache CloudStack

CloudStack Use-cases

What can we do with CloudStack?

- Self service of all resources- compute, storage, and networking- with no requirements of highly skilled technical staff.

- Automation of all provisioning and management through API/UI/CLI.

- Examples;
    - Create virtual machines from templates or ISOs
    - All virtual machine life cycle actions: start/stop/delete/storage/networking
    - Manage storage volumes
    - Create isolated, shared, and multi-tiered networks
    - Manage firewall and port forwarding rules
    - Manage network services such as load balancing, static and source NAT, VPNs, global load balancing, and auto-scaling
    - Create container clusters

Hype Cycle of Cloud Computing
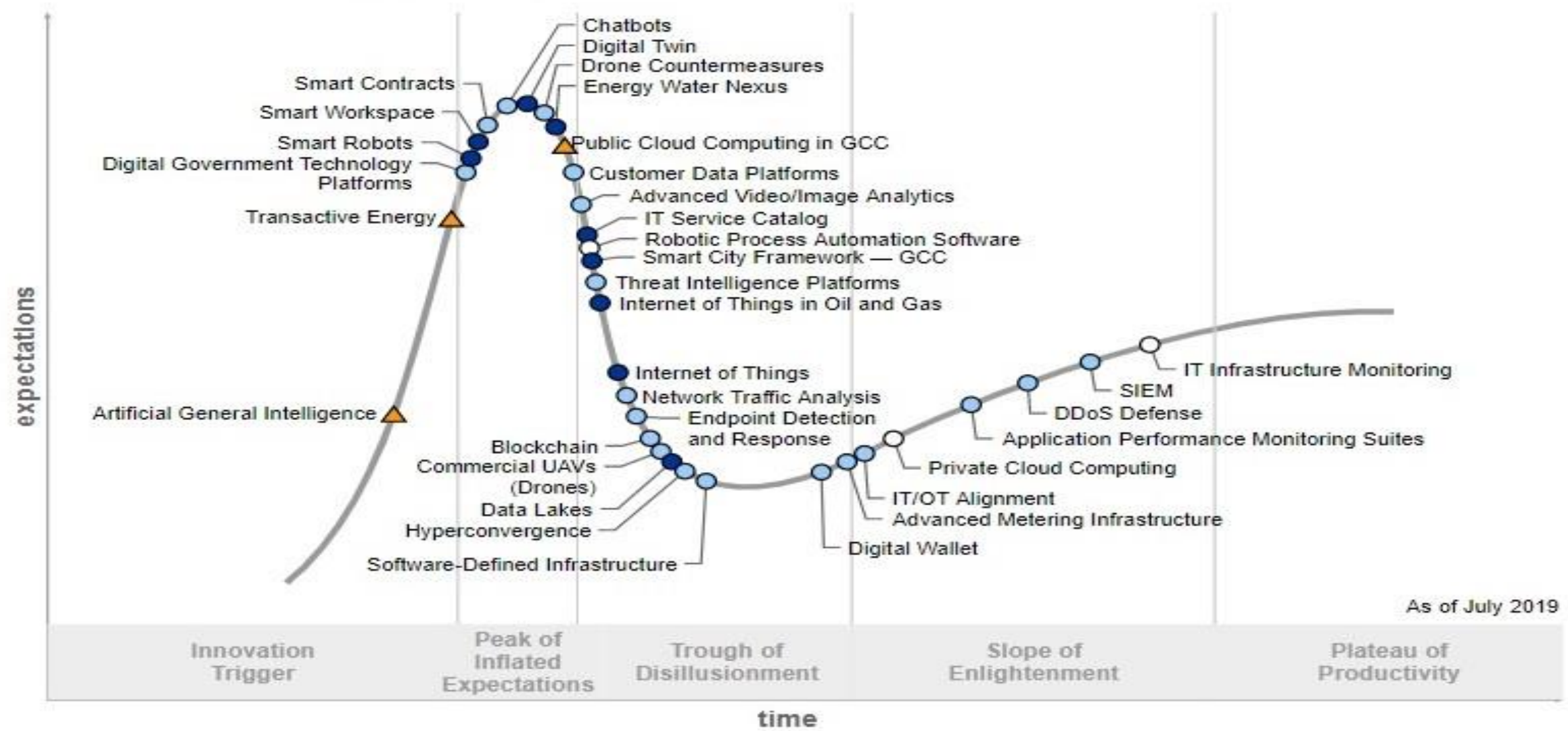
CloudStack Networking Models

- **Basic zone- Shared network model** (all VMs share a single L2 network, simple to set up, security groups are attached, scales very well, VMs route traffic directly to the gateway)

- **Advanced zone network model** (sophisticated network topologies, VMs share VPC, provides the most flexibility and network functions when defining guest networks, different tenant's networks use SDN technologies, virtual router per isolated network, complex to understand the set up)

- **Advanced zone- Shared network model** (similar to basic zone, usually without security groups, provisioned by cloud administrator)

- **Advanced zone-Isolated network model** (single network behind virtual router, most networking features present such as VPN, load balancer, private gateway, provisioned by end users, a number of specialized physical devices can be used for some of the networking functions)

- **Advanced zone-VPC** (richest network model, multiple network behind virtual router, provisioned by end users, special network cases such as private gateway networks (VPC internal network), network services supported such as ACL, port forwarding, VPN, load balancing, DHCP, DNS, routing etc.)

- **Advanced zone- L2 networks** (a special case, closed VLAN, CloudStack used for orchestration of everything-except networking, all services are provided on physical layer by datacenter administrator, no IPAM)

Installation and Configuration:

- **https://docs.cloudstack.apache.org/en/4.18.1.0/**

- **Quick Installation Guide:**

https://docs.cloudstack.apache.org/en/4.18.1.0/quickinstallationguide/qig.html