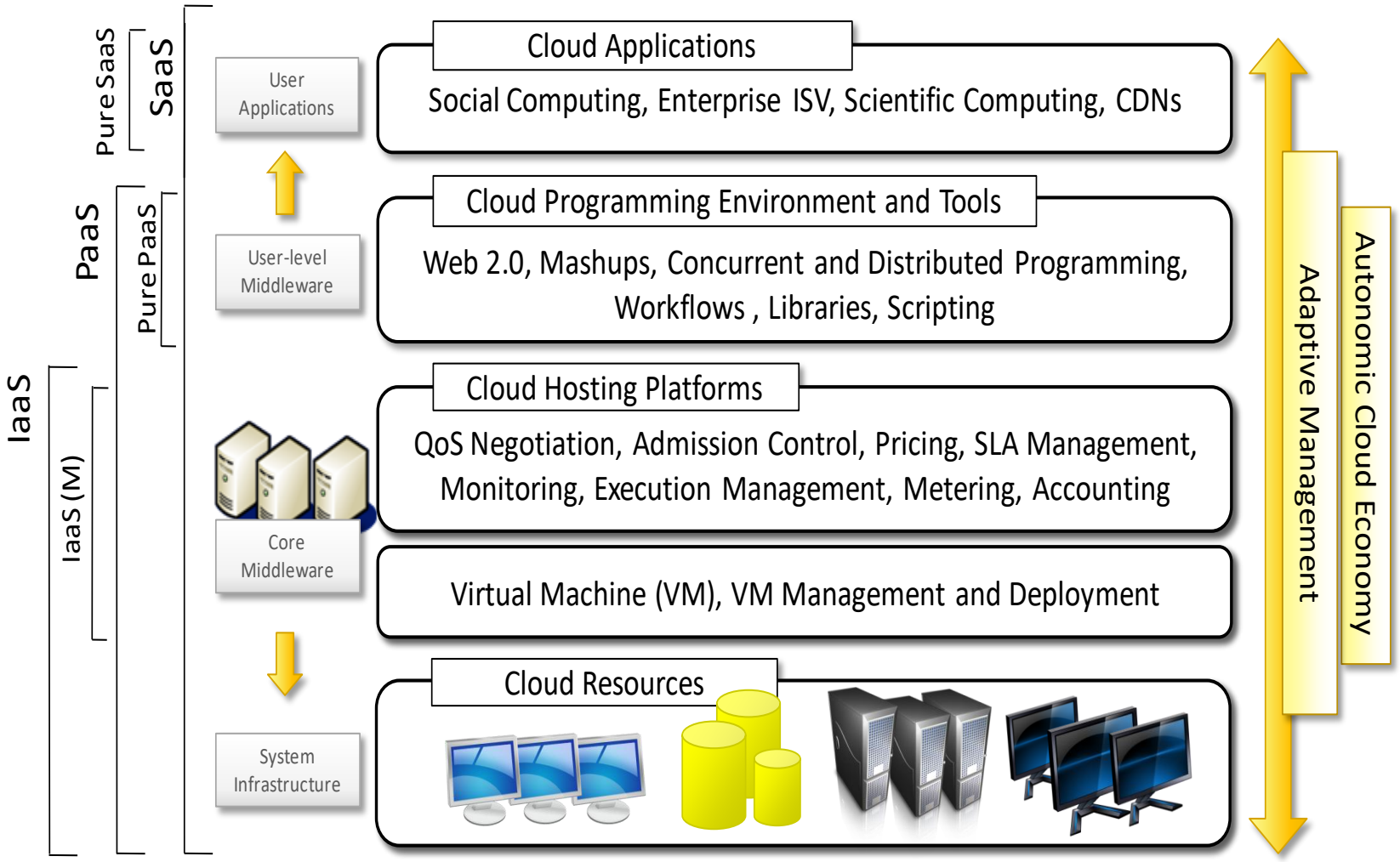
Cloud Computing Architecture

Dr. Shreeya Swagatika Sahoo

Introduction

- Cloud computing is a utility-oriented and Internet-centric way of delivering IT services on demand.
- These services cover the entire computing stack: from the hardware infrastructure packaged as a set of virtual machines to software services such as development platforms and distributed applications.
- It is possible to organize all the concrete realizations of cloud computing into a layered view covering the entire stack rom hardware appliances to software systems.
- Cloud resources are harnessed to offer “computing horsepower” required for providing services.

Cloud Reference Model



Cloud Reference Model

- The combination of cloud hosting platforms and resources is generally classified as a **Infrastructure-as-a-Service (IaaS)** solution.
- The management layer is often integrated with other IaaS solutions **IaaS(M)** that provide physical infrastructure and adds value to them.
- Users develop their applications specifically for the cloud by using the API exposed at the user level middleware which is also known as **Platform-as-a-Service(PaaS).**
- PaaS solutions generally include the infrastructures well, which is bundled as part of the service provided to users. In the case of **Pure PaaS**, only the user-level middle ware is offered, and it has to be complemented with a virtual or physical infrastructure.
- The top layer of the reference model contains services delivered at the application level referred as **Software-as-a-Service(SaaS).**

Cloud Computing Services Classification

Category	Characteristics	Product Type	Vendors & Products
SaaS	Customers are provided with applications that are accessible anytime and from anywhere.	Web applications and services (Web 2.0)	SalesForce.com (CRM); Clarizen.com (Project Management); Google Apps…
PaaS	Customers are provided with a platform for developing applications hosted in the Cloud.	Programming APIs and frameworks; Deployment Systems.	Google AppEngine; Microsoft Azure; Manjrasoft Aneka; Data Synapse…
IaaS/HaaS	Customers are provided with virtualized hardware and storage on top of which they can build their infrastructure.	Virtual machines management infrastructure; Storage management; Network management.	Amazon EC2 and S3; GoGrid; Nirvanix…

Cloud Reference Model

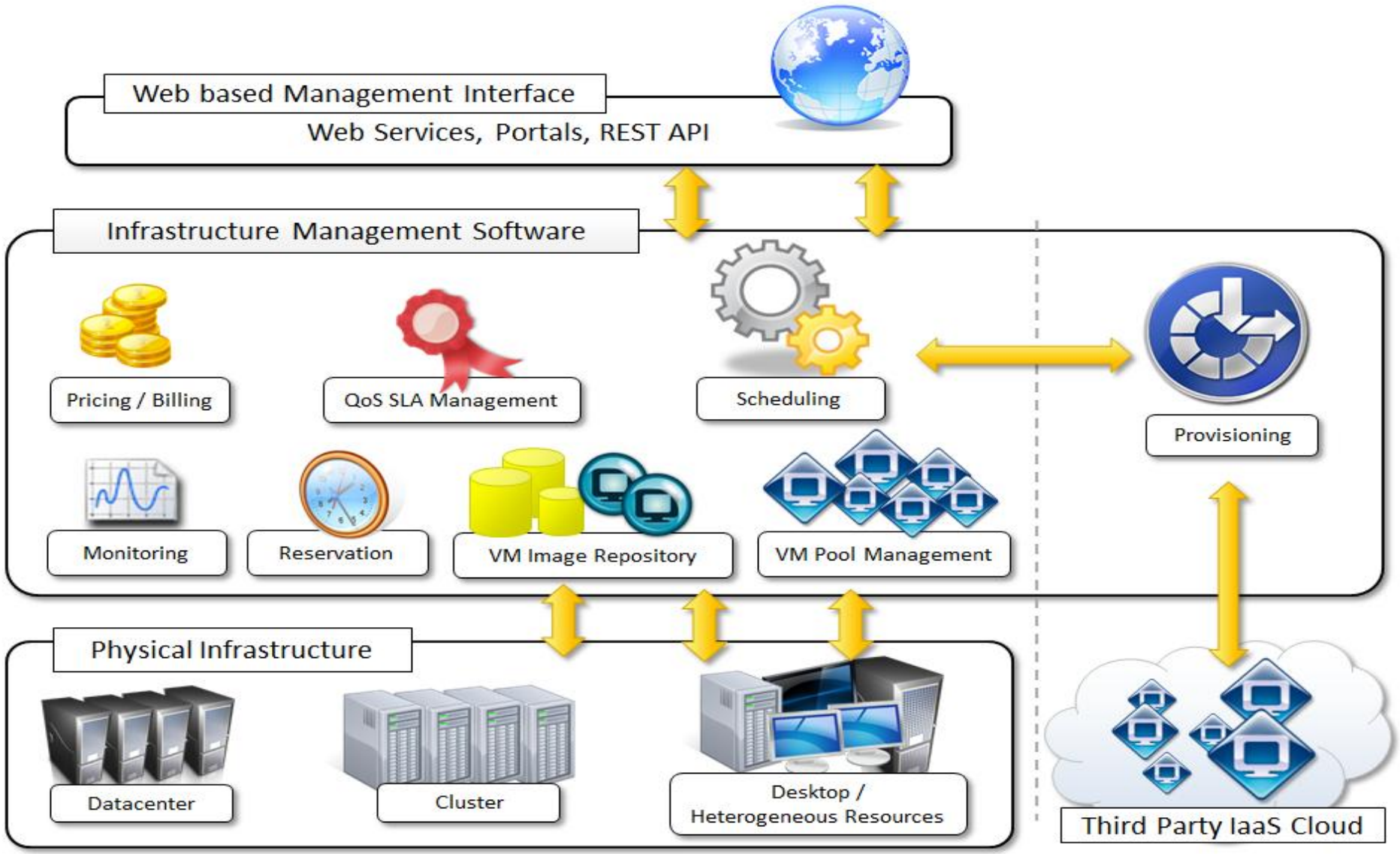
- The model identifies three major service models, which are layered one upon the other, from the base of the computing stack to the top.

1. **Infrastructure-as-a-Service (IaaS)**
2. **Platform-as-a-Service (PaaS)**
3. **Software-as-a-Service (SaaS)**

Infrastructure- and hardware-as-a-service (IaaS/HaaS)

- This layer is at the base of the stack and is supported by Cloud Resources and System Infrastructure.
- **Function**: IaaS solutions deliver infrastructure on demand, typically in the form of virtual hardware, storage, and networking. The combination of cloud hosting platforms and resources is generally classified as an IaaS solution.
- **Components:** Cloud resources are harnessed to provide the necessary "computing horsepower," often implemented using datacenters where resources are virtualized to provide isolation and effective utilization. Virtual hardware is used to provide compute on demand through virtual machine instances.
- **User Interaction**: Users are given tools and interfaces to configure the software stack on the virtual instances. This infrastructure management often relies on software providing storage and network virtualization.
- Examples: Amazon EC2, S3, Rightscale, and vCloud

Infrastructure-as-a-Service reference implementation



Infrastructure-as-a-Service reference implementation

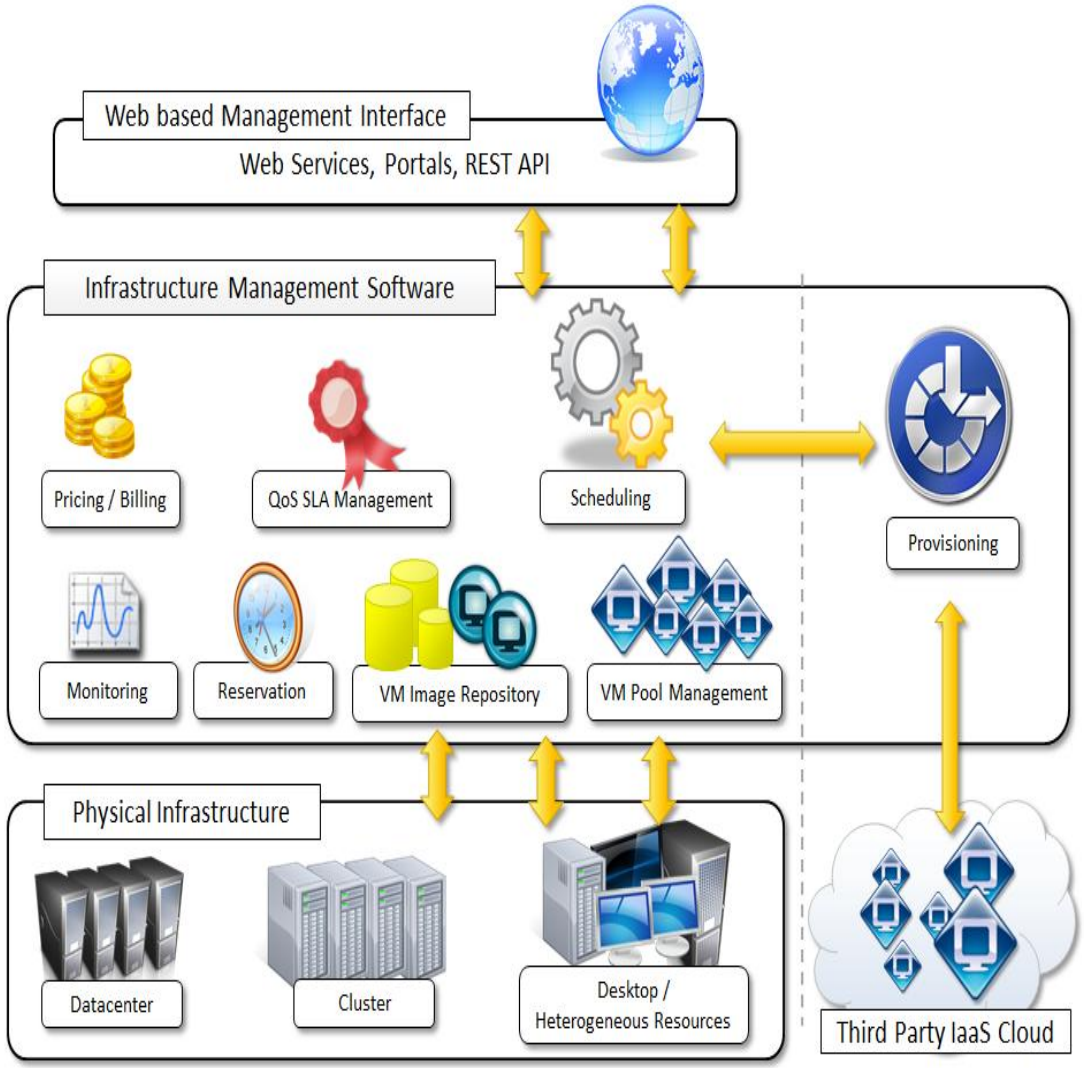
- It provides an overall view of the components forming an Infrastructure-as-a-Service solution.
- It is possible to distinguish three principal layers: the physical infrastructure, the software management infrastructure, and the user interface.
- At the top layer the user interface provides access to the services exposed by the software management infrastructure. Such interface is generally based on Web 2.0 technologies: web services, RESTful APIs, and mash-ups.
- The core features of an Infrastructure-as-a-Service solution are implemented in the infrastructure management software layer.
- In particular, the management of the virtual machines is the most important function performed by this layer. A central role is played by the scheduler, which is in-charge of allocating the execution of virtual machine instances.

Infrastructure-as-a-Service reference implementation

Web-based Management Interface (Top Layer)

- **Components:**
  - Web Services, Portals, REST API
- **Purpose**

  This layer provides the **user interface** for cloud administrators and customers to interact with the system. Through web portals or APIs, users can:
  - Request new virtual machines (VMs)
  - Monitor usage
  - Manage resources and configurations



Contd.

Infrastructure Management Software (Middle Layer)

- This is the core management layer that controls all cloud operations and automates infrastructure management.
- It includes several functional modules:

a. Pricing / Billing

- Calculates cost based on resource usage (e.g., CPU hours, storage, bandwidth).
- Generates bills for users.

b. **QoS (Quality of Service) and SLA Management**

- Ensures services meet predefined performance and reliability standards.
- Manages Service Level Agreements (SLAs) between users and providers.

c. **Scheduling**

- Allocates physical or virtual resources efficiently to users' tasks or VMs.
- Ensures load balancing and optimized performance.

Contd.

d**. Monitoring**

- Tracks system performance, health, and resource utilization in real time.

e. Reservation

- Reserves specific resources or time slots for users or high-priority tasks.

f. **VM Image Repository**

- Stores templates or images of virtual machines that can be quickly deployed when needed.

g**. VM Pool Management**

- Manages a pool of virtual machines available for allocation to users.
- Handles creation, migration, and termination of VMs.

h**. Provisioning**

- Responsible for creating and deploying virtual machines from available templates or images.
- Connects with both internal and third-party infrastructure.

Contd.

Physical Infrastructure (Bottom Layer)

- This layer represents the hardware resources that the cloud software manages.

It includes:

  - Datacenters: Physical buildings housing computing and storage servers.
  - Clusters: Grouped servers that work together to provide computing power.
  - Desktop/Heterogeneous Resources: Different types of systems (e.g., Windows, Linux) contributing resources.

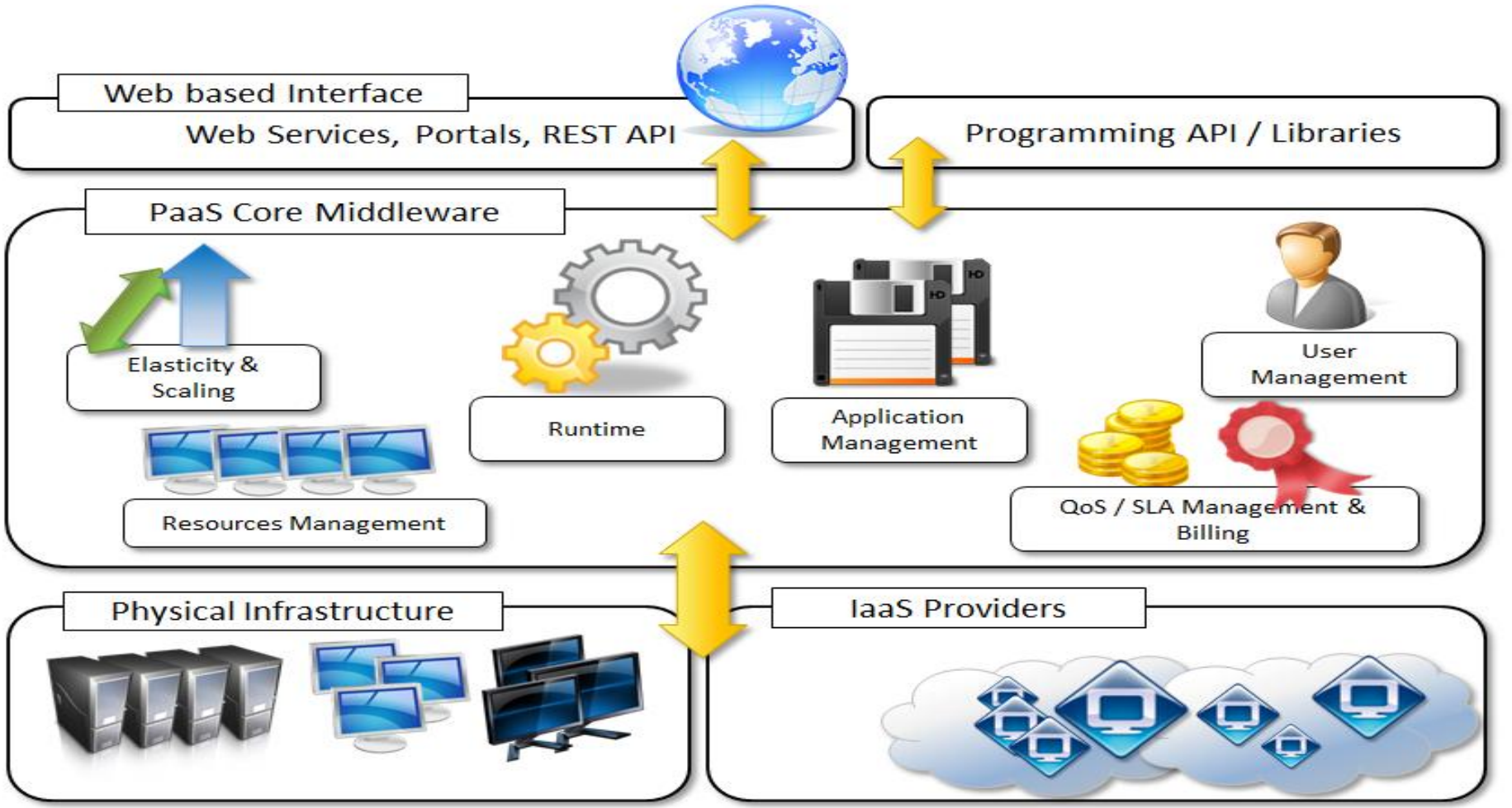
Third Party IaaS Cloud

- Refers to integration with external cloud providers (e.g., AWS, Azure, Google Cloud).
- Enables hybrid cloud or multi-cloud setups where resources can be provisioned from external infrastructure when needed.

Platform-as-a-Service (PaaS)

- This layer represents the next step up the stack and is generally characterized by Cloud Hosting Platforms and User-level Middleware.
- **Function**: PaaS solutions deliver scalable and elastic runtime environments on demand for hosting the execution of applications. They expose APIs and libraries that applications leverage.
- **Core Middleware**: This middleware platform is responsible for creating the abstract environment where applications are deployed and executed. It manages resource and application management, QoS/SLA management, and handles elasticity and scaling.
- **User Focus**: Users are requested to focus on the logic of the application, while the service provider is responsible for ensuring scalability and fault tolerance.
- **Pure PaaS**: Some vendors deliver only the middleware (referred to as Pure PaaS). Manjrasoft Aneka is characterized as a pure PaaS solution.
- Examples: Windows Azure, Google AppEngine, and Aneka

Platform-as-a-Service (PaaS)



Physical Infrastructure & IaaS Providers (Bottom Layer)

- **Physical Infrastructure:**
  Represents the actual hardware resources such as servers, storage, and networking devices that form the backbone of the cloud environment.
- **IaaS Providers:**
  These are Infrastructure-as-a-Service platforms (like AWS EC2, Google Compute Engine, Microsoft Azure Infrastructure).
  They virtualize and manage physical resources, providing compute, storage, and networking capabilities to higher layers.

PaaS Core Middleware (Middle Layer)

- This is the **core** of the PaaS architecture — the layer that abstracts infrastructure and provides an environment for developing, deploying, and managing applications.

Key Components:

- **Elasticity & Scaling:**
  Automatically adjusts resources (like CPU, memory, or storage) based on application demand that is scaling up or down as needed.
- **Resources Management:**
  Handles the allocation and monitoring of computing resources efficiently.

- **Runtime:**
  Provides the necessary runtime environment for executing applications (e.g., Java, Python, .NET).
- **Application Management:**
  Tools and services for deploying, configuring, updating, and managing applications.
- **User Management:**
  Handles authentication, authorization, and account management for developers and end users.
- **QoS / SLA Management & Billing:**
  Ensures Quality of Service (QoS) according to Service Level Agreements (SLA) and manages billing based on usage.

Web-Based Interface & Programming API (Top Layer)

- This is the **user interaction layer**, where developers and users engage with the PaaS platform.

Components

- **Web-Based Interface:**
  Includes web portals, dashboards, and REST APIs that allow users to access and manage platform services.
- **Programming API / Libraries:**
  Provides APIs and SDKs that developers use to build, deploy, and manage applications programmatically.

Platform as a Service Offering Classification

Category	Description	Product Type		Vendors & Products
PaaS-I	Runtime environment with web hosted application development platform. Rapid application prototyping.	Middleware Middleware	+ +	Force.com; Longjump;
PaaS-II	Runtime environment for scaling web applications. The runtime could be enhanced by additional components which provide scaling capabilities.	Middleware Middleware Middleware Middleware Middleware Middleware	+ + + + 	Google AppEngine; AppScale; Heroku; Engine Yard; Joyent Smart Platform; GigaSpaces XAP;
PaaS-III	Middleware and programming model for developing distributed applications in the Cloud.	Middleware Middleware Middleware Middleware Middleware Middleware	+ 	Microsoft Azure; DataSynapse; Cloud IQ; Manjrasof Aneka; Apprenda SaaSGrid; GigaSpaces DataGrid;

Essential Characteristics of PaaS solution

- *Runtime framework*. It represents the "software stack" of the PaaS model and the most intuitive aspect that comes to the mind of people when referring to Platform-as-a-Service solutions. The runtime framework executes end-user code according to the policies set by the user and the provider.
- *Abstraction*. PaaS solutions are distinguished by the higher level of abstraction that they provide. PaaS the focus is on the applications the Cloud must support.
- *Automation*. PaaS environment automate the process of deploying applications to the infrastructure, scaling them by provisioning additional resources when needed.
- *Cloud services*. PaaS offerings provide developers and architects with services and APIs helping them to simplify the creation and delivery of elastic and highly available Cloud applications.

Software-as-a-Service (SaaS)

- This layer sits at the top of the stack and consists of Cloud Applications.
- **Function**: SaaS solutions provide applications and services directly to end users on demand. They leverage the horsepower provided by IaaS and PaaS solutions.
- **Scope**: They commonly offer functionalities similar to desktop applications, such as office automation, customer relationship management (CRM), and photo editing.
- **Characteristics:** These applications are typically Web-based and often utilize a multitenant architecture, where costs are shared across a large user base.
- Examples: Google Documents, Facebook, Flickr, and Salesforce

Types of Clouds

- The four major deployment models are
  - Public Cloud
  - Private Cloud
  - Hybrid Cloud
  - Community cloud

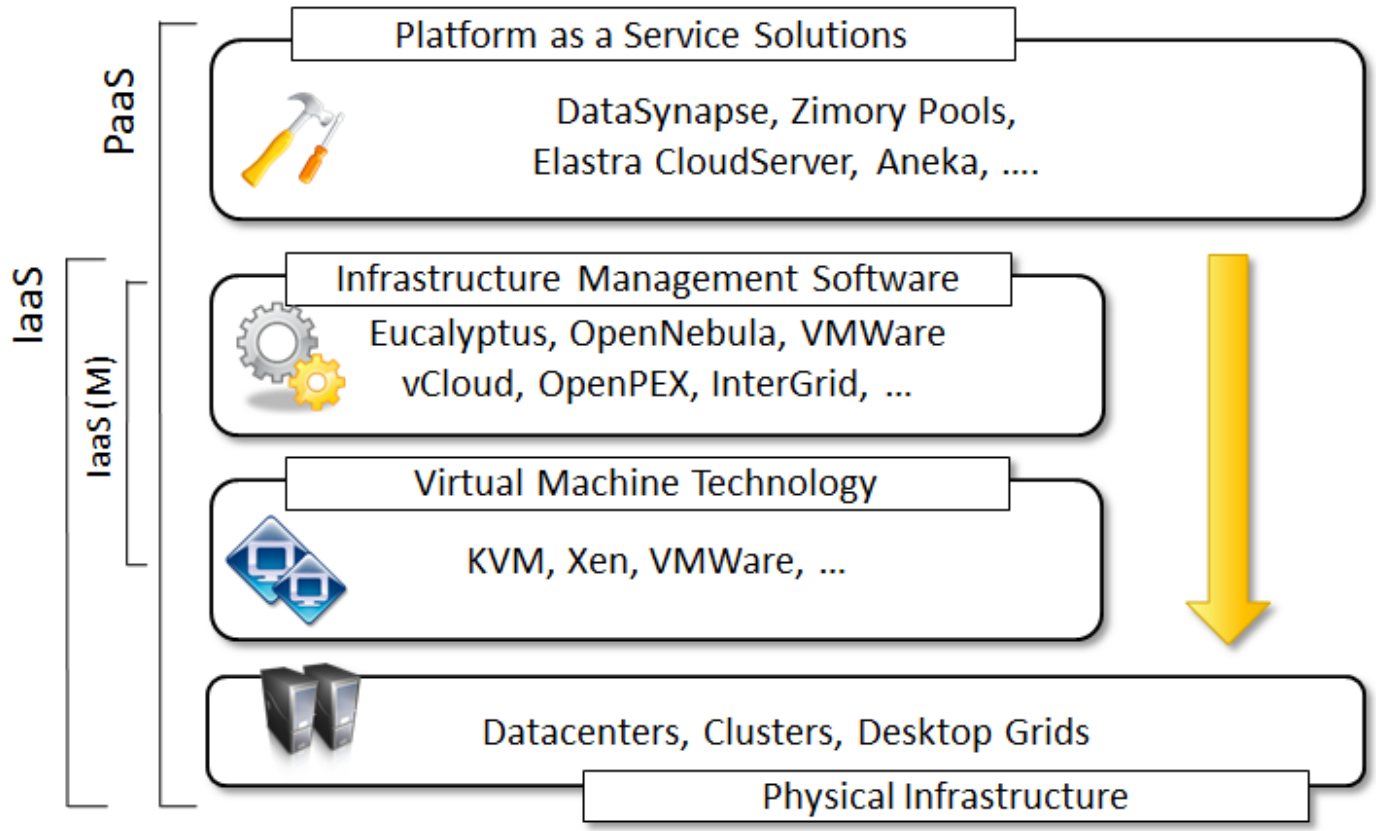
Public Cloud

- **Definition** :
  - The cloud is open to the wider public and its services are generally made available to anyone, from anywhere, and at any time through the Internet.
  - Customers can easily subscribe, enter credentials and billing details, and immediately use the services offered.
- **Structure**: They are typically implemented as a distributed system composed
- **Economics and Benefits:**
  - They offer solutions for minimizing IT infrastructure costs.
  - They are a viable option for handling peak loads on local infrastructure and are attractive for small enterprises as they can start businesses without large up-front investments.
  - Customers can dynamically upsize or downsize their IT according to business demands.

Public Cloud

- **Key Characteristic**: A fundamental characteristic is multi-tenancy, meaning the cloud is designed to serve a multitude of users, not just a single customer.
- **Service Offerings**: Public clouds can offer any type of service, including Infrastructure-as-a-Service (IaaS) like Amazon EC2, Platform-as-a-Service (PaaS) like Google AppEngine, and Software-as-a-Service (SaaS) like SalesForce.com.

Private Cloud



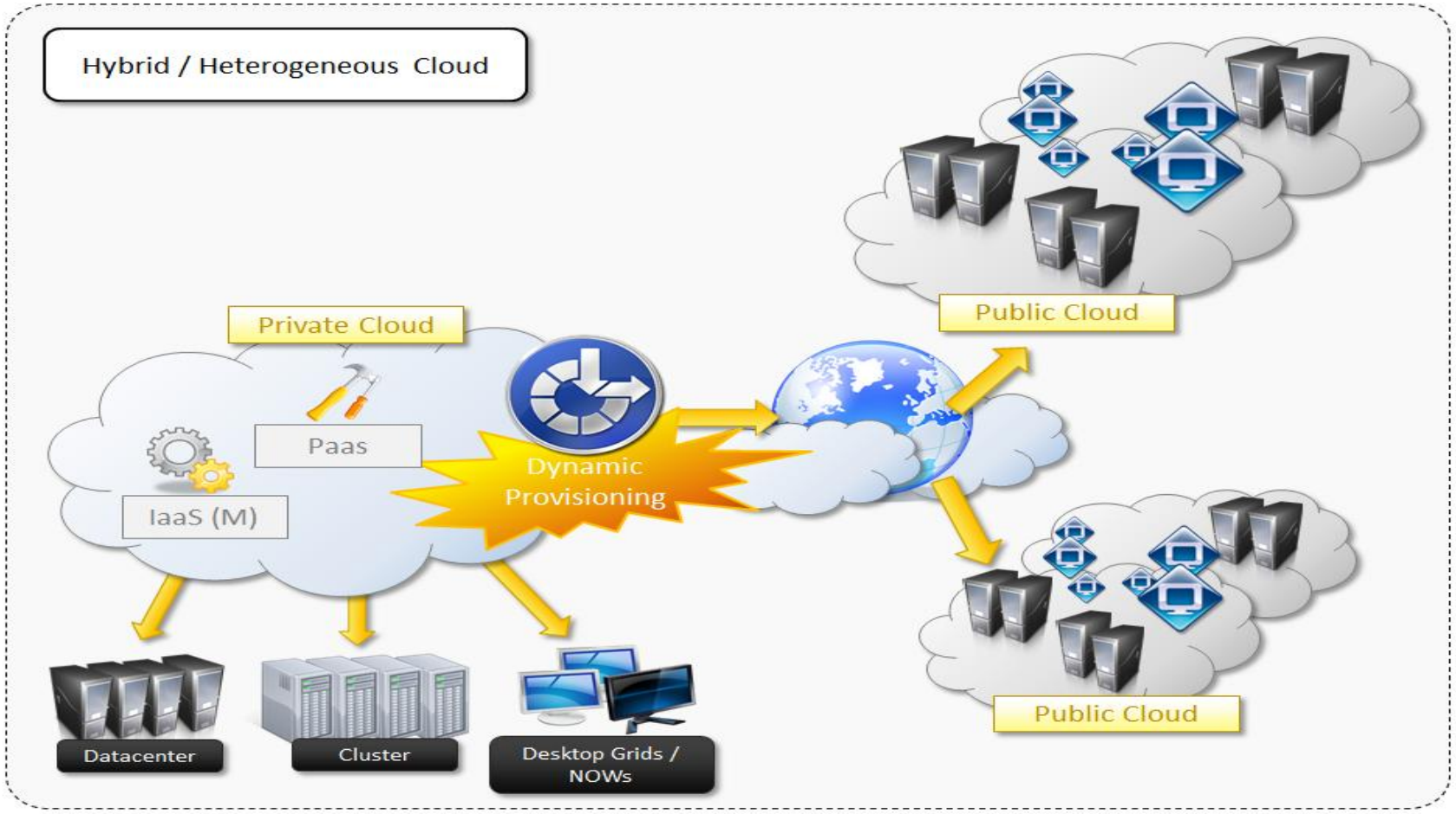
Private Cloud

- The cloud is implemented within the private premises of an institution and is generally made accessible only to the members of that institution or a subset of them. They rely on a private infrastructure.
- Differently from Public Clouds, instead of a pay-as-you-go model, there could be other schemes in place, which take into account the usage of the Cloud and proportionally bill the different departments or sections of the enterprise.

Benefits: Private clouds help in keeping core business operations in-house and reducing security concerns. They also allow for better utilization of existing IT resources and enable testing applications at a lower cost compared to public clouds.

Limitations: Compared to public clouds, private clouds exhibit a more limited capability to scale elastically on demand

Hybrid Cloud



Hybrid Cloud

- A hybrid cloud is a mixed environment resulting from a private cloud that integrates additional services or resources from one or more public clouds.
- It consists of hybrid computing systems partially composed of privately owned infrastructures and partially of public cloud resources. Hybrid clouds are also referred to as **heterogeneous clouds.**

Purpose and Benefits:

- Hybrid clouds are a common way for many stakeholders to start exploring the possibilities offered by cloud computing.
- They are utilized whenever private cloud resources are unable to meet users' quality-of-service (QoS) requirements.
- They allow enterprises to exploit existing IT infrastructures while maintaining sensitive information within their own premises.

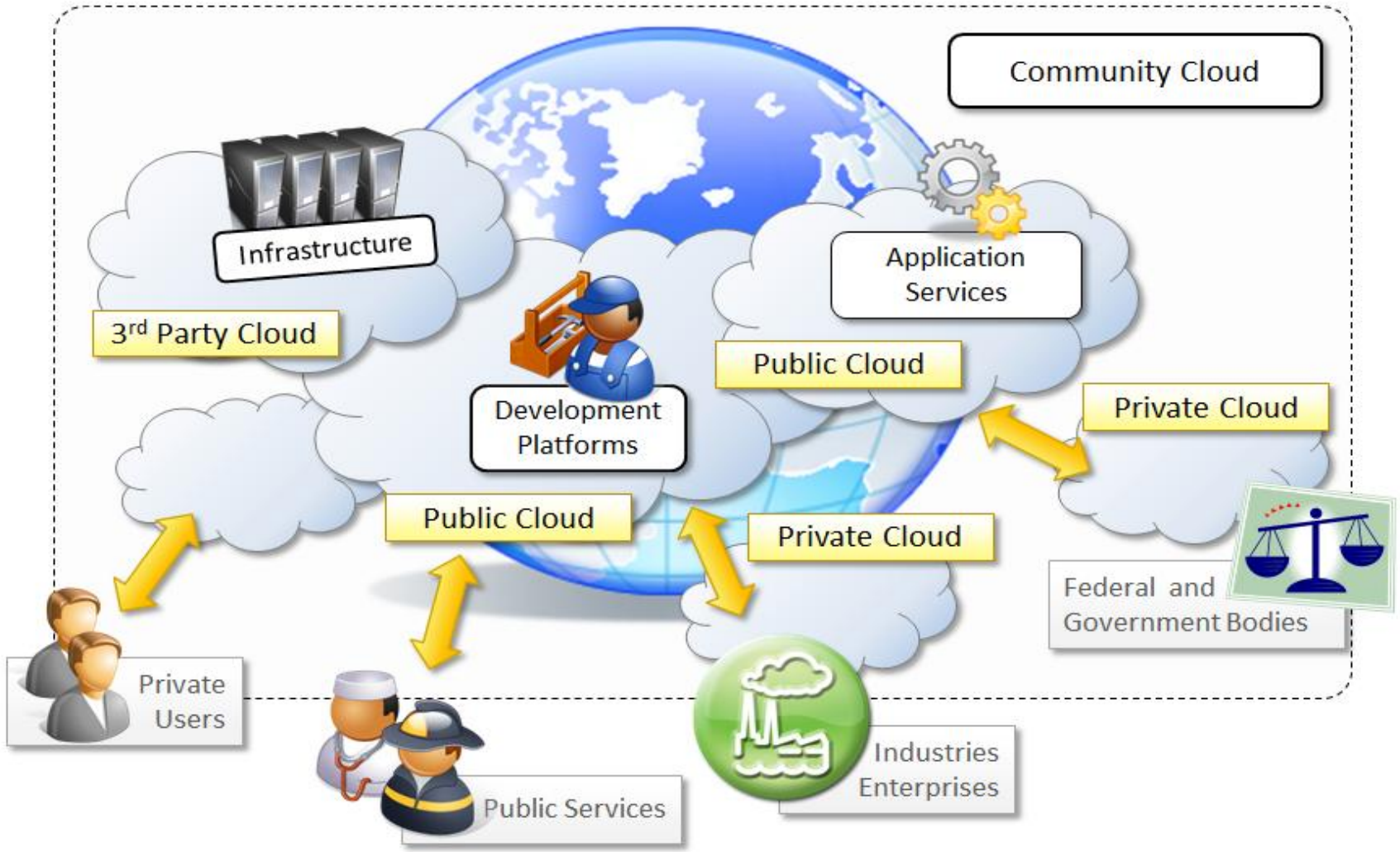
Hybrid Cloud

Scalability and Dynamics:

- The hybrid model allows organizations to naturally **grow and shrink** capacity by provisioning external resources and releasing them when they are no longer needed.
- Hybrid clouds address **scalability** issues by leveraging external resources for exceeding capacity demand.
- Dynamic provisioning is a fundamental component in the hybrid cloud scenario.
- The practice of temporarily leasing and releasing these external resources or services when required is also known as **cloudbursting.**

- Infrastructure management software and Platform-as-a-Service (PaaS) solutions are key components for deploying and managing hybrid clouds.
- PaaS solutions, such as Manjrasoft's Aneka, support hybrid cloud deployment mode. Aneka's provisioning service leverages different IaaS providers to scale the existing cloud infrastructure.
- Infrastructure management software like OpenNebula can integrate resources from public clouds such as Amazon EC2

Community Clouds



Community Clouds

- A community cloud is one of the four major deployment models for cloud computing, alongside public, private, and hybrid clouds.
- It is defined as an infrastructure shared by several organizations.
- It supports a specific community that has shared concerns (such as mission, security requirements, policy, and compliance considerations).
- Community clouds are characterized by a multi-administrative domain.
- They are designed specifically to address the needs of a specific industry.

Architecture

- Community clouds are created by integrating the services of different clouds.
- Architecturally, they are typically implemented over multiple administrative domains.
- These domains involve contributions from different organizations, such as government bodies, private enterprises, research organizations, and public virtual infrastructure providers, to build the cloud infrastructure.
- This model often represents a mixed deployment involving public, private, and hybrid cloud models.
- Examples: Scientific Research, Healthcare Industry, Public Sector, Media Industry etc.

Economics of the Cloud

- The main drivers of Cloud computing are: economy of scale and simplicity of software delivery and its operation.
- The biggest benefit of this phenomenon is financial: the *pay-as-you-go* model offered by Cloud providers. In particular, Cloud computing allows:
  - reducing the capital costs associated to the IT infrastructure;
  - eliminating the depreciation or lifetime costs associated with IT capital assets
  - replacing software licensing with subscriptions
  - cutting down the maintenance and administrative costs of IT resources
-

Economics of the Cloud

- A *capital cost* is the cost occurred in purchasing an asset that is useful in the production of goods or the rendering of services.
- The amount of cost savings that Cloud computing can introduce within an enterprise is related to the specific scenario in which Cloud services are used and how they contribute to generate a profit for the enterprise.
- In the case of a small startup starting its business it is possible to completely leverage the Cloud for many aspects such as:
  - IT infrastructure;
  - Software development;
  - CRM and ERP;

Open Challenges

- **Cloud Definition** : Defining cloud computing remains an ongoing challenge because the phenomenon is constantly evolving
- **Cloud Interoperability and Standards**: To fully realize cloud computing as a utility model, introducing standards and ensuring interoperability among solutions from different vendors is of fundamental importance
- **Scalability and Fault Tolerance**: While the ability to scale on demand is an attractive feature, ensuring this capability is a significant difficulty for those who develop, manage, and maintain the cloud middleware
- **Security, Trust, and Privacy**: Traditional cryptographic methods secure data transit, but information must be **decrypted in memory** for processing
- **Organizational Aspects**: Cloud computing fundamentally changes how IT services are consumed and managed, raising organizational issues. Key questions arise concerning the **new role of the IT department** in an enterprise that significantly relies on the cloud

Thank You