Cloud Computing for IoT

Bibhas Ghoshal

IIIT Allahabad

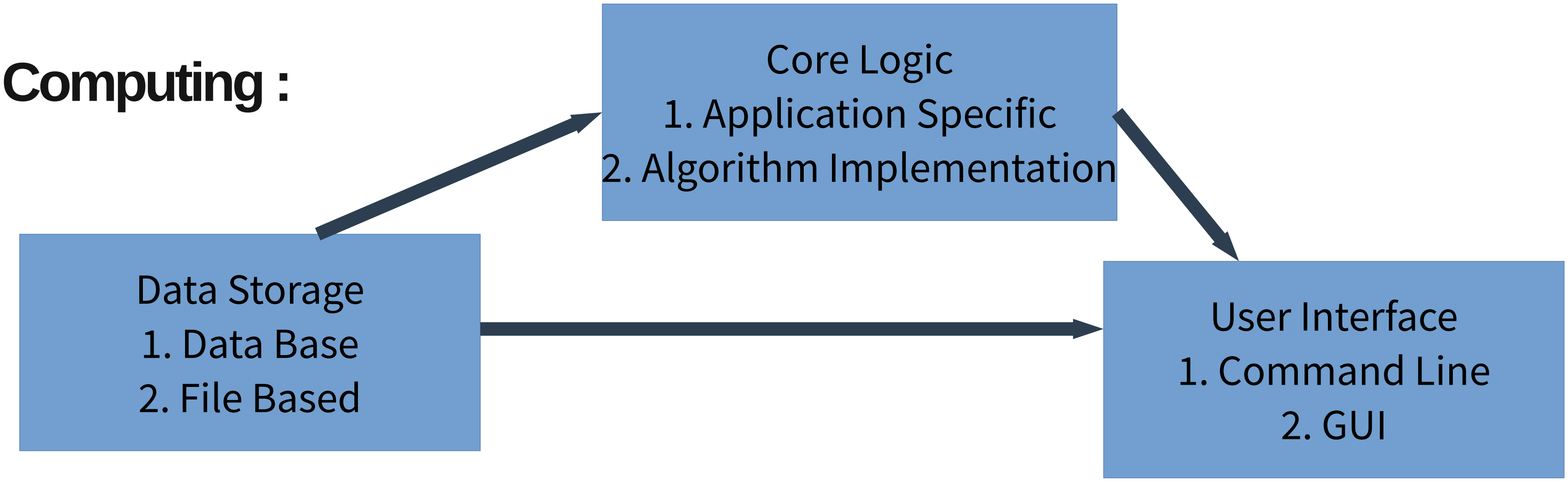
Traditional Computing

Traditional Computing Components:

Data Storage	Core Logic	User Interface
1. Data Base	1. Application Specific	1. Command Line
2. File Based	2. Algorithm Implementation	2. GUI

All components sit on the same computer

Distributed Computing :



Distributed Computing

Distributed System : A collection of independent computers that appears to its users as a single coherent system

Ex : Internet, computing cluster, LAN, Data centre, web

Distributed Application :

- One single system or one of several subsystems
- Collection of processors – parallel processing (increased performance, reliability and fault tolerance)
- Partitioned or replicated data

Why Distribution?

- Sharing of information and services
- Increase in reliability, fault tolerance and improved performance

Goals and Challenges for Distributed Systems

Goals :

Making Resources Available

Distribution Transparency

Openness

Scalability

Security

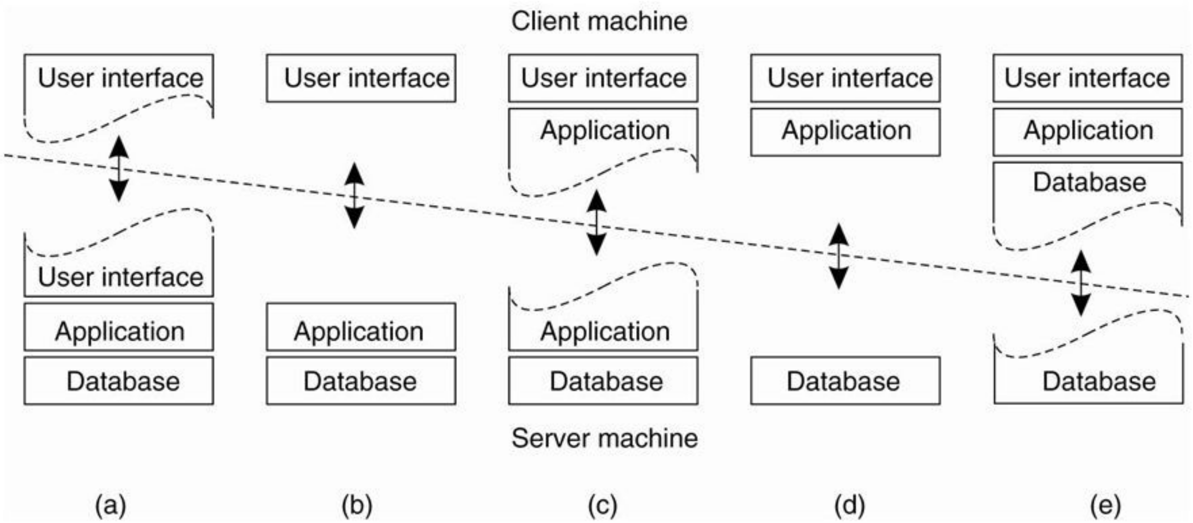
System Design requirements

Challenges :

Naming and Access control, Security, Performance, Mutual exclusion of users, replication and migration, failure modes, concurrency, heterogeneity

Client Server Architectures in Distributed Systems

Multitiered Architectures

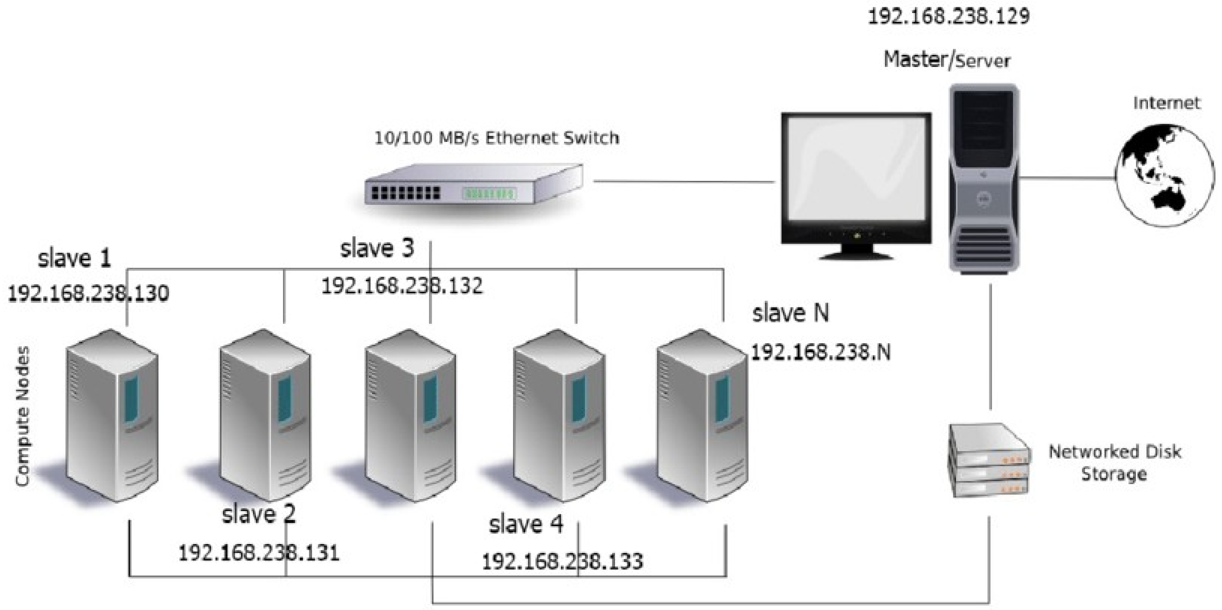


Alternative client-server organizations (a)–(e).

Clients: thin → fat

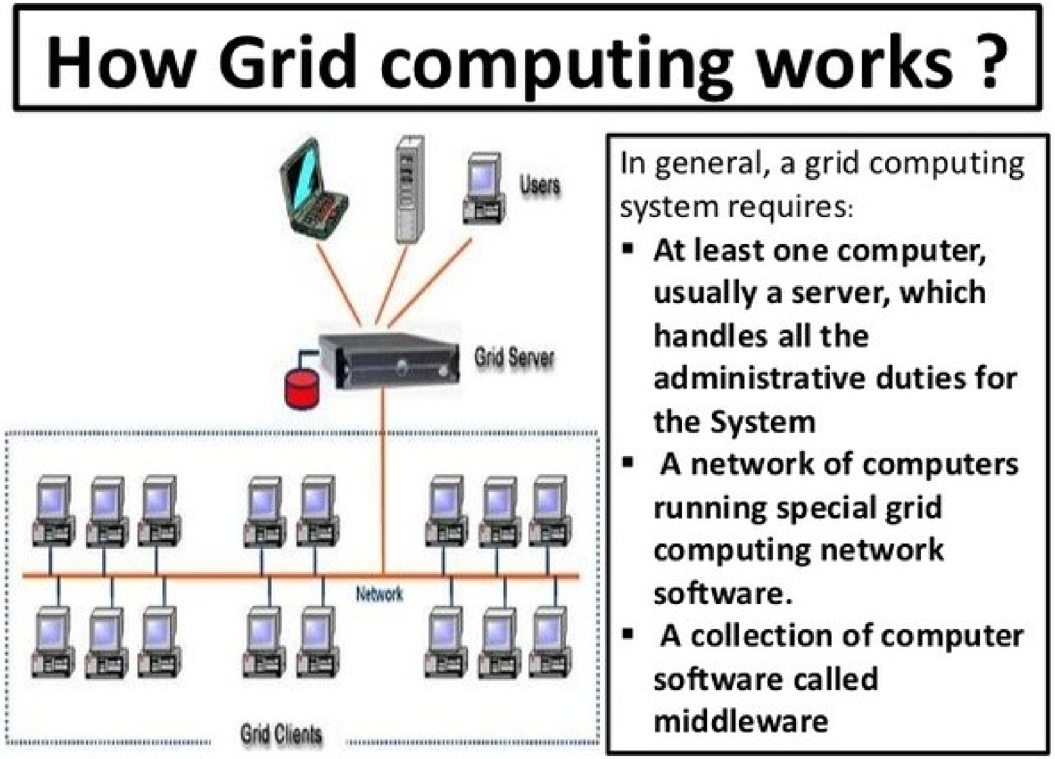
Trends in Distributed Computing

Cluster Computing : Homogeneous computing nodes (connected loosely or tightly) working together



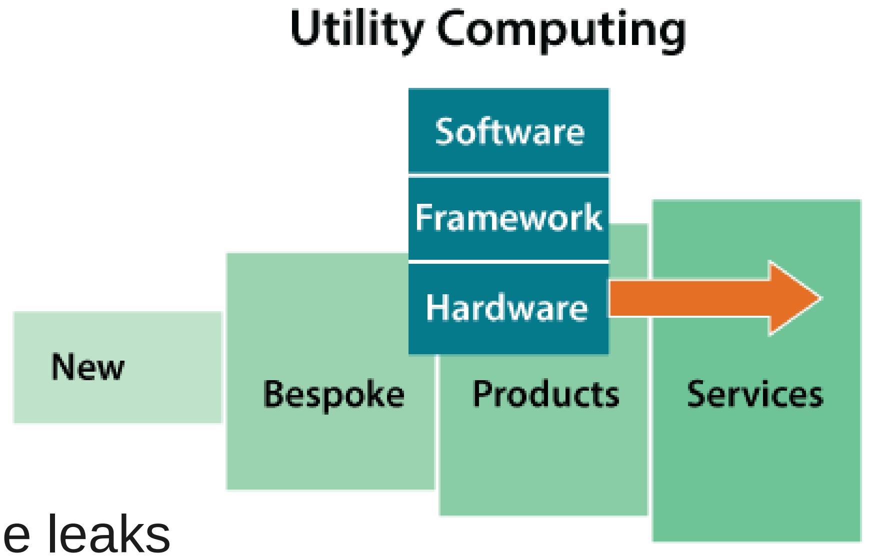
Trends in Distributed Computing

Grid Computing : Heterogeneous computing nodes distributed over a wide area to perform very large tasks



Trends in Distributed Computing

Utility Computing : Packaged resources available for computing and storage (on rent)



Q: What do you look out for ?

Ans :

1. Can I afford the rent?
2. Is there enough computing and power
   And memory?
3. Is it safe from other users - memory / code leaks
4. will the application use affect me?
5. Can I pay for what I use?

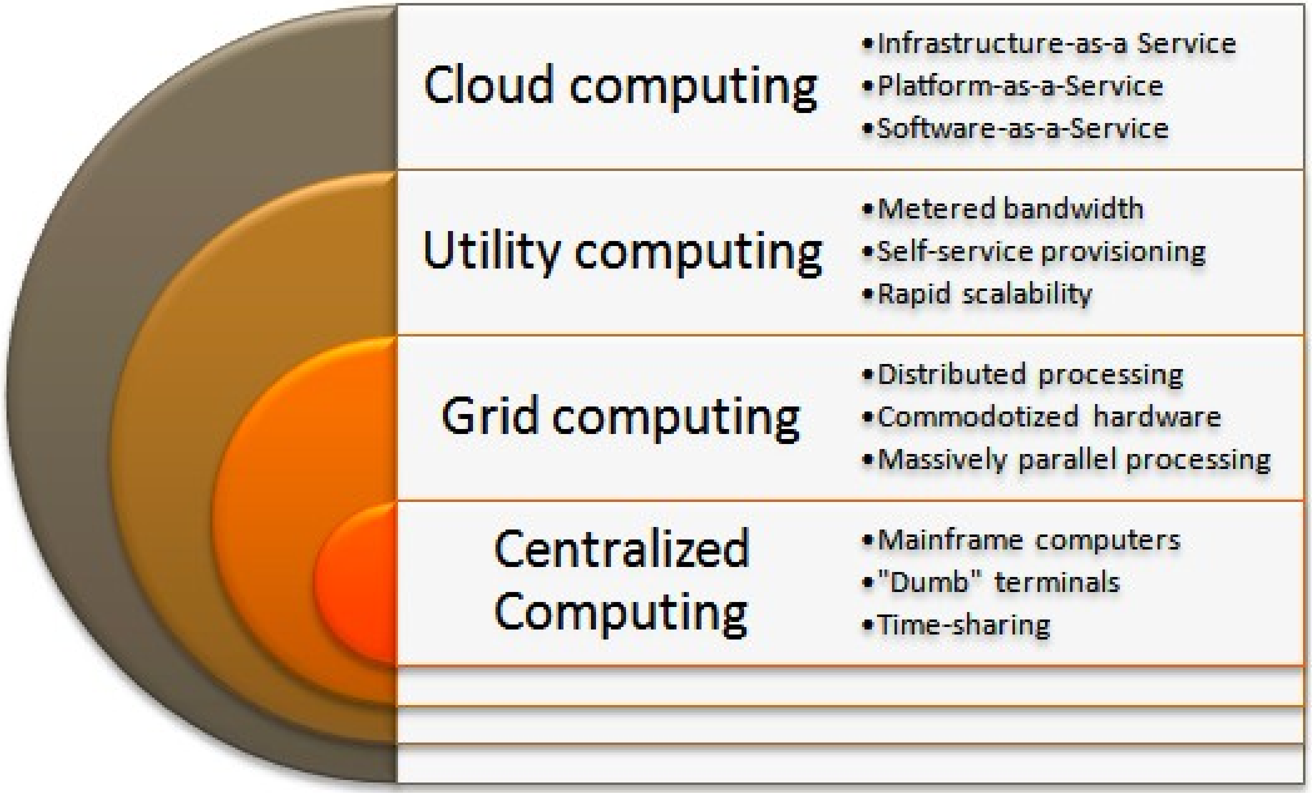
Trends in Distributed Computing

Cloud Computing : Model for enabling convenient, on-demand network access to shared pool of configurable computing resources.

Ex: network infrastructures, servers, storage, application

- Step from utility computing
- High level generalization of computing and storage model
- It can be rapidly allocated and released with low management effort
- Some essential characteristics, service models and deployment models

Trends in Distributed Computing

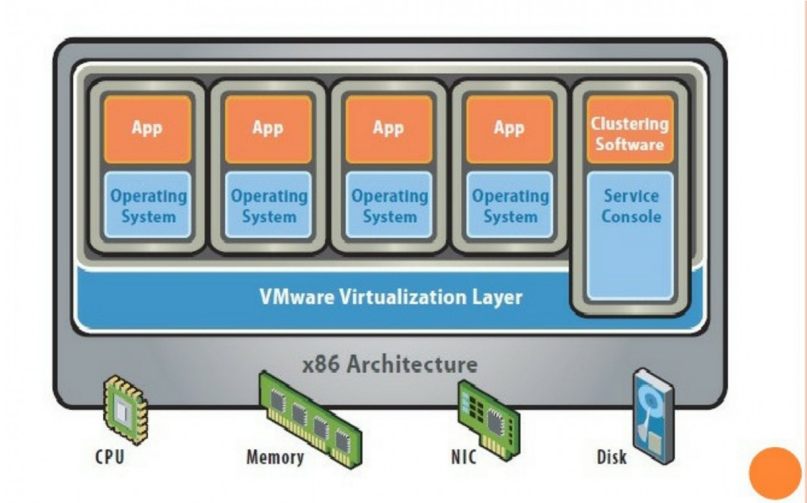


Adapted from a graphic by Kent Langley

Virtualization

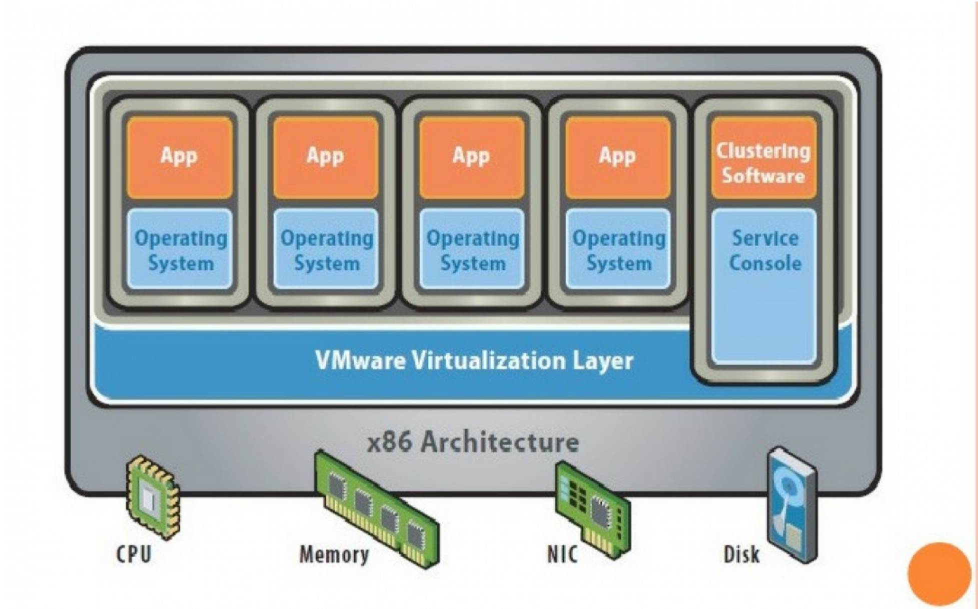
- Share same hardware among independent users
- Increases degree of hardware parallelism
- Easier management and energy usage
- Flexible allocation and utilization
- Decouple apps from underlying hardware

  H/W upgrades without impact on OS

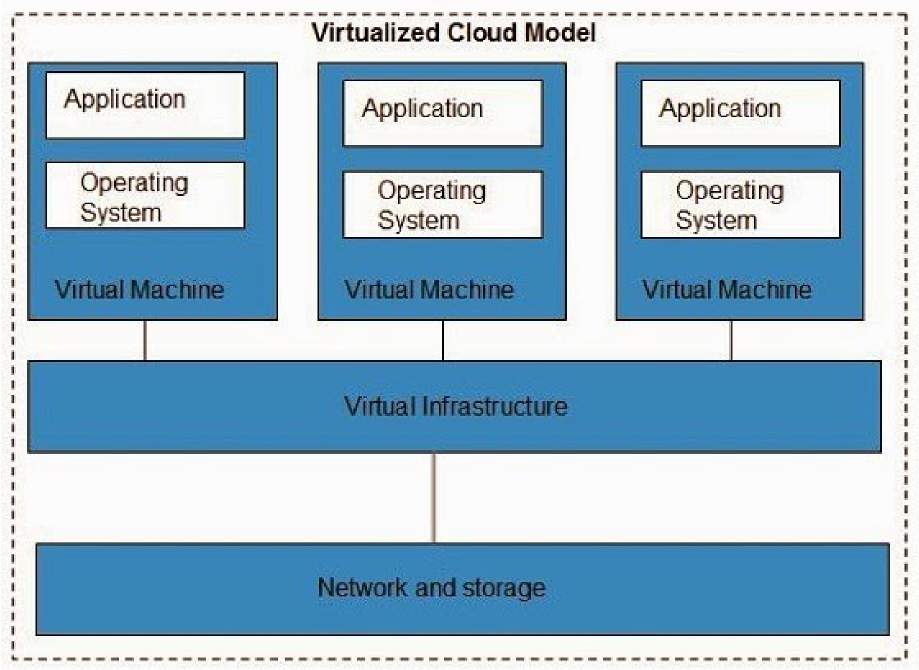


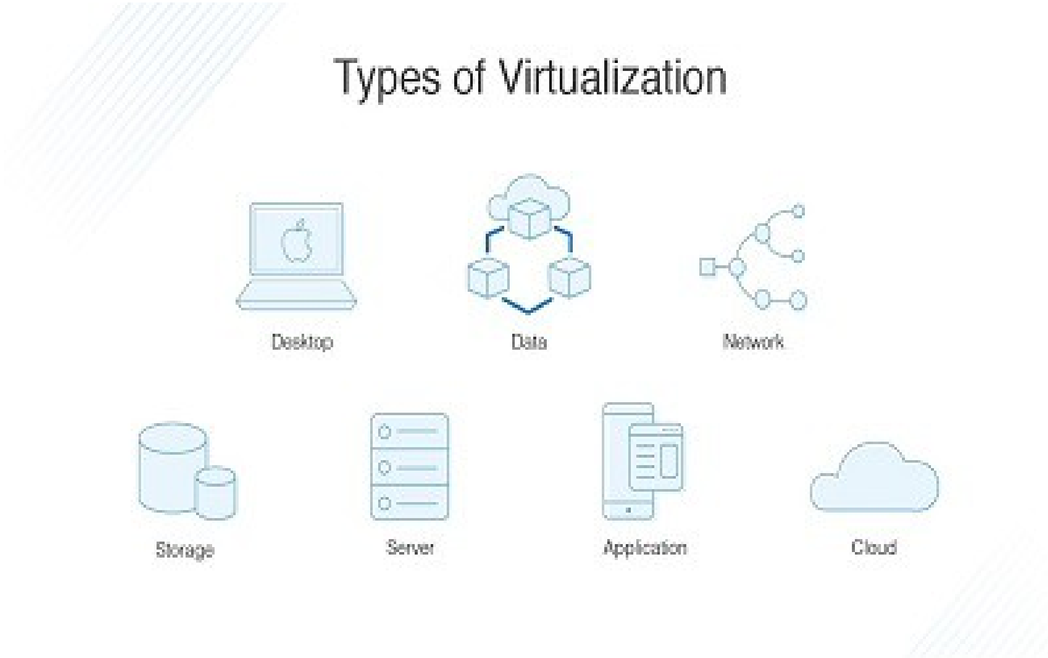
Virtualization Raises Abstraction

- Virtual Memory to access large address space (Physical memory mapping is hidden by OS using paging)
- Allow code on one architecture to run on another
- Physical devices to Virtual Devices

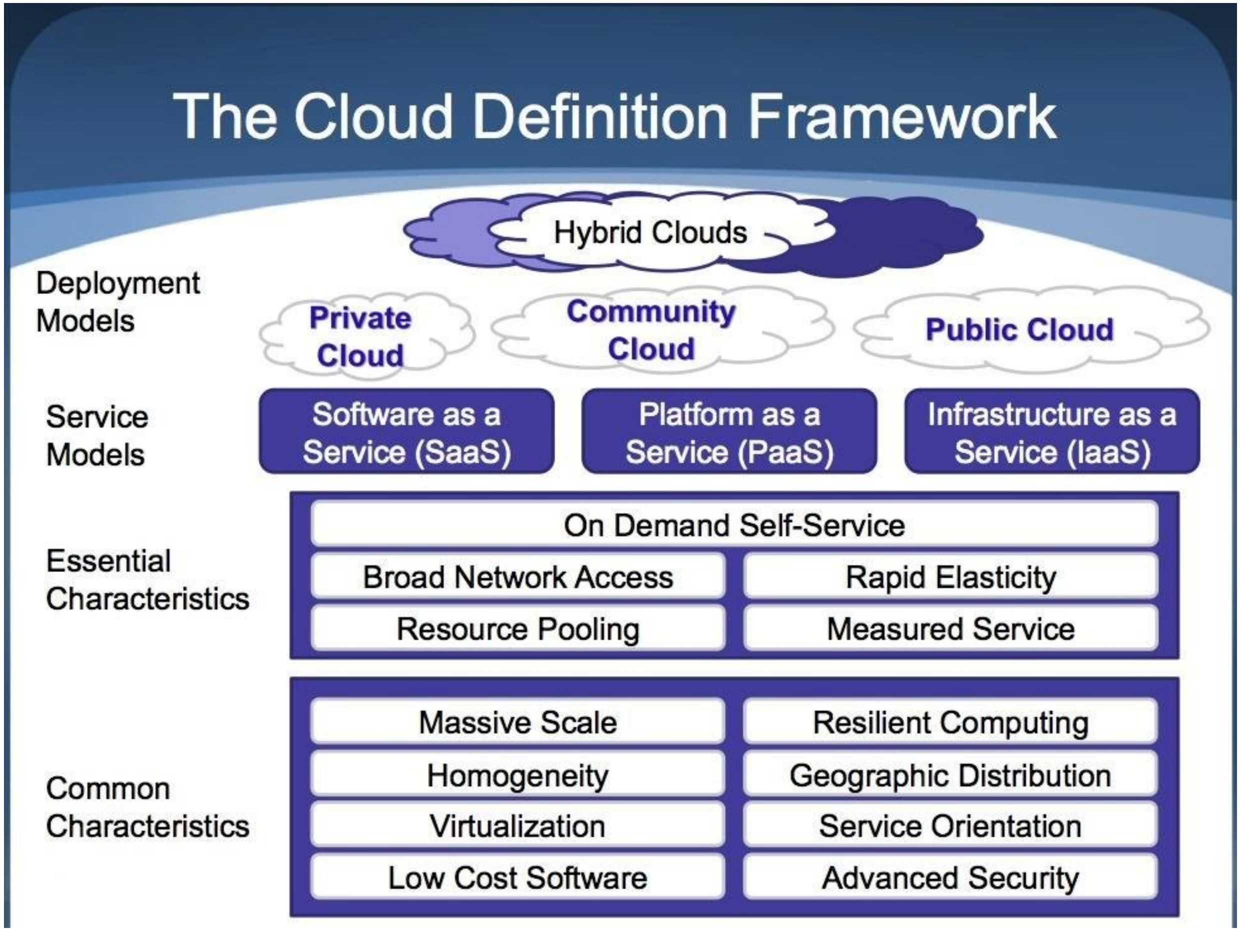


Virtualization in Cloud





Cloud Computing Framework



Cloud Computing : Characteristics

Improved agility in resource provisioning

Ubiquitous – independent of device and location

Multi tenancy – sharing of resources and costs accross a large pool of users

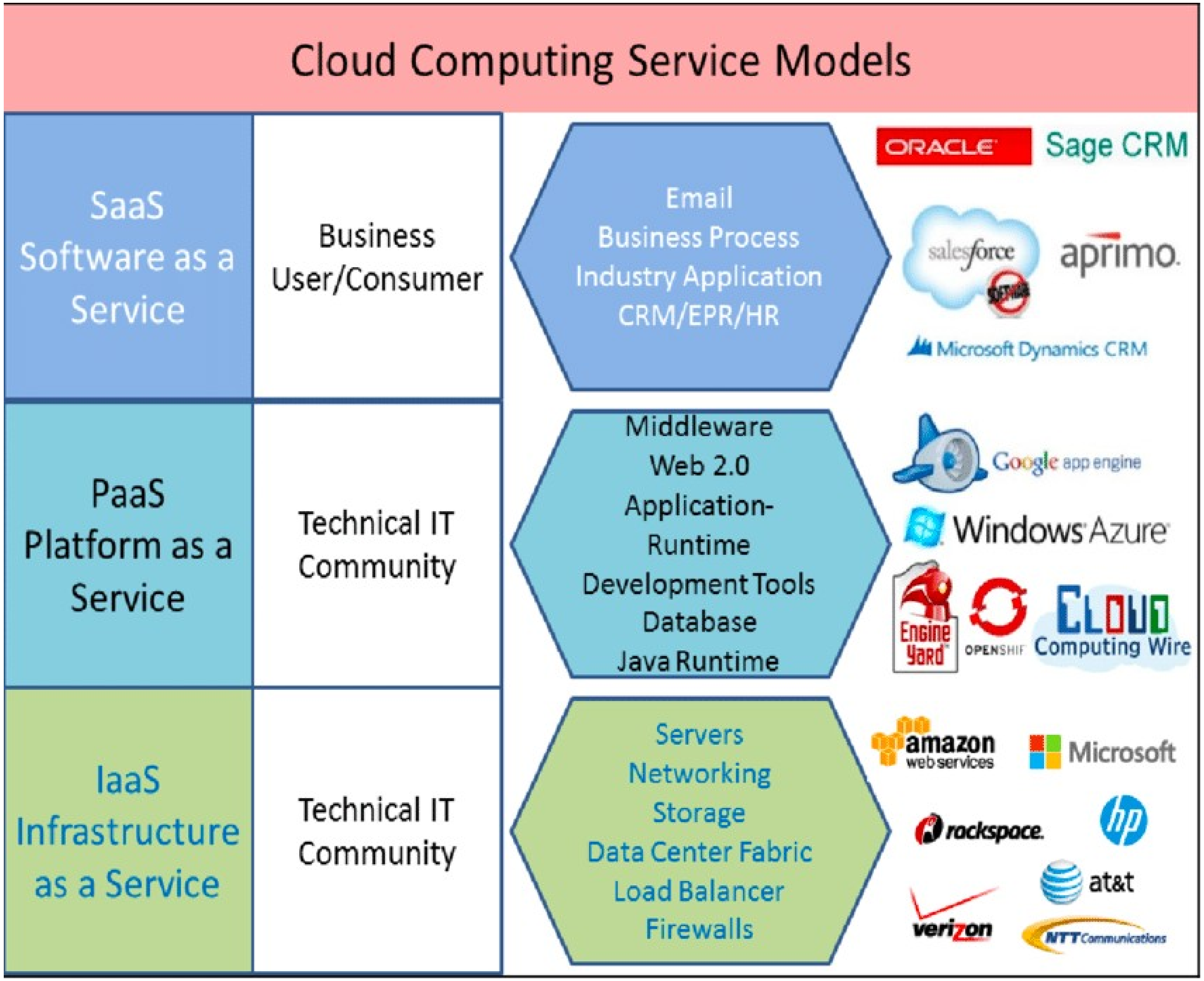
Dynamic load balancing

Highly reliable and scalable

Low cost and low maintenance

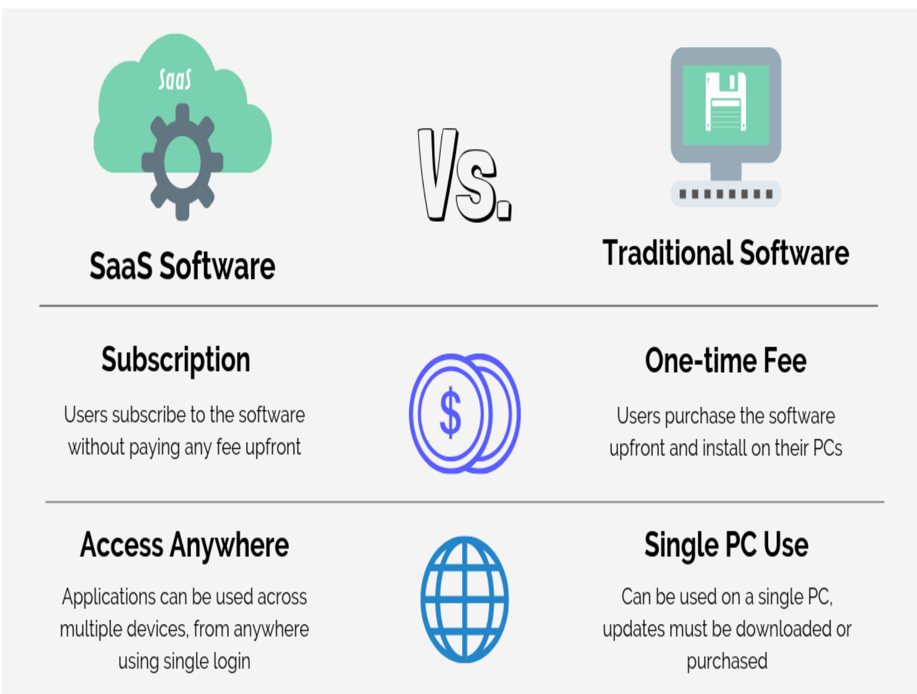
Improve security and access control

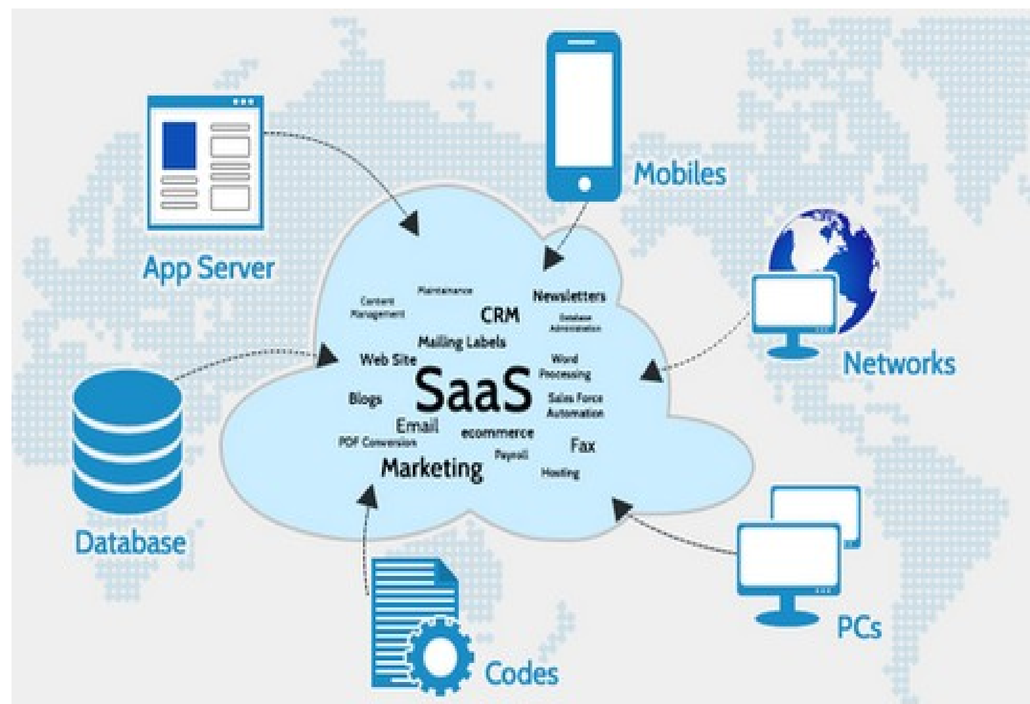
Cloud Computing : Service Models



Software-as-a Service (SaaS)

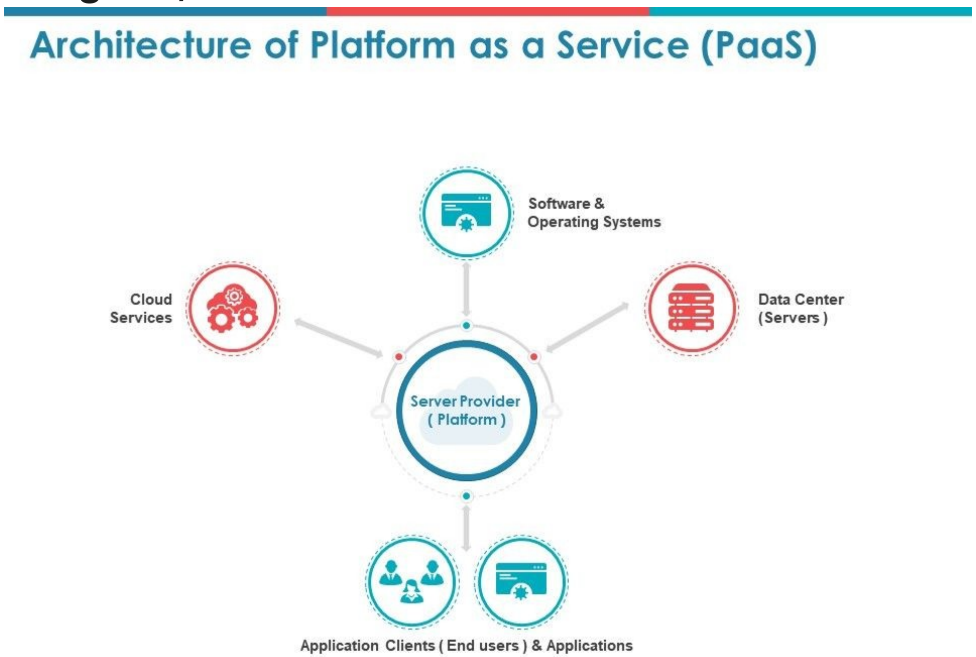
- Facility to execute service provider’s application at users end
- Services can be accessed via different types of client devices such as web browser or app
- End users do not have control of cloud platform
- Example : Google Apps





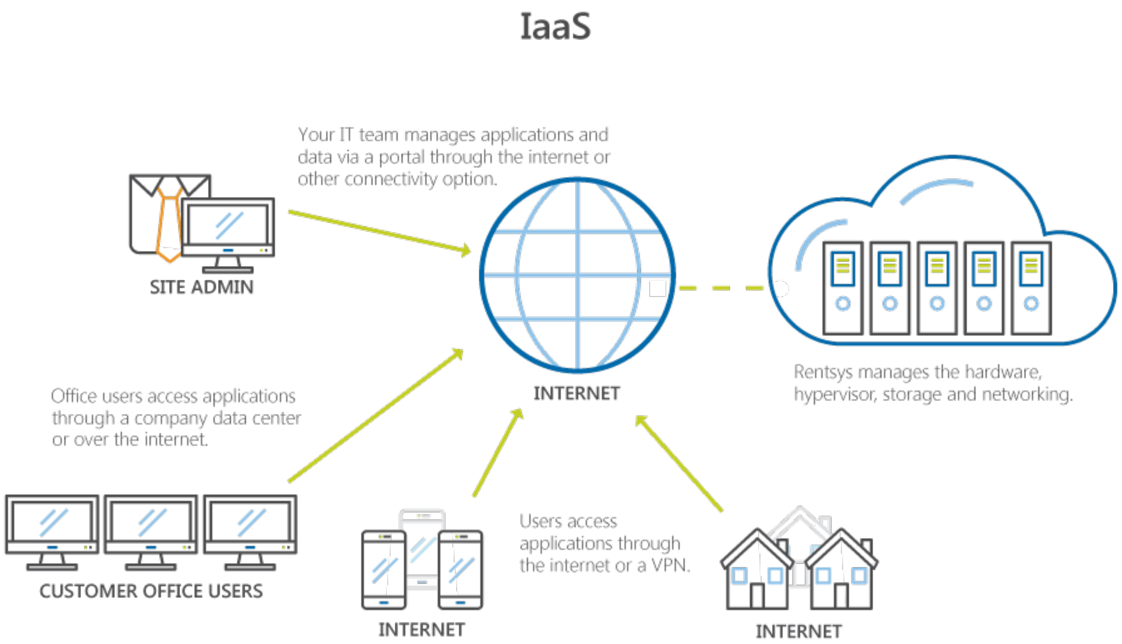
Platform-as-a-Service(PaaS)

- Facilitates consumer created or acquired applications onto cloud infrastructure
- Support for deployment of such applications
- Users do not control cloud infrastructure
- Users can control deployed apps using given configuration
- Ex : Google App Engine , Windows Azure

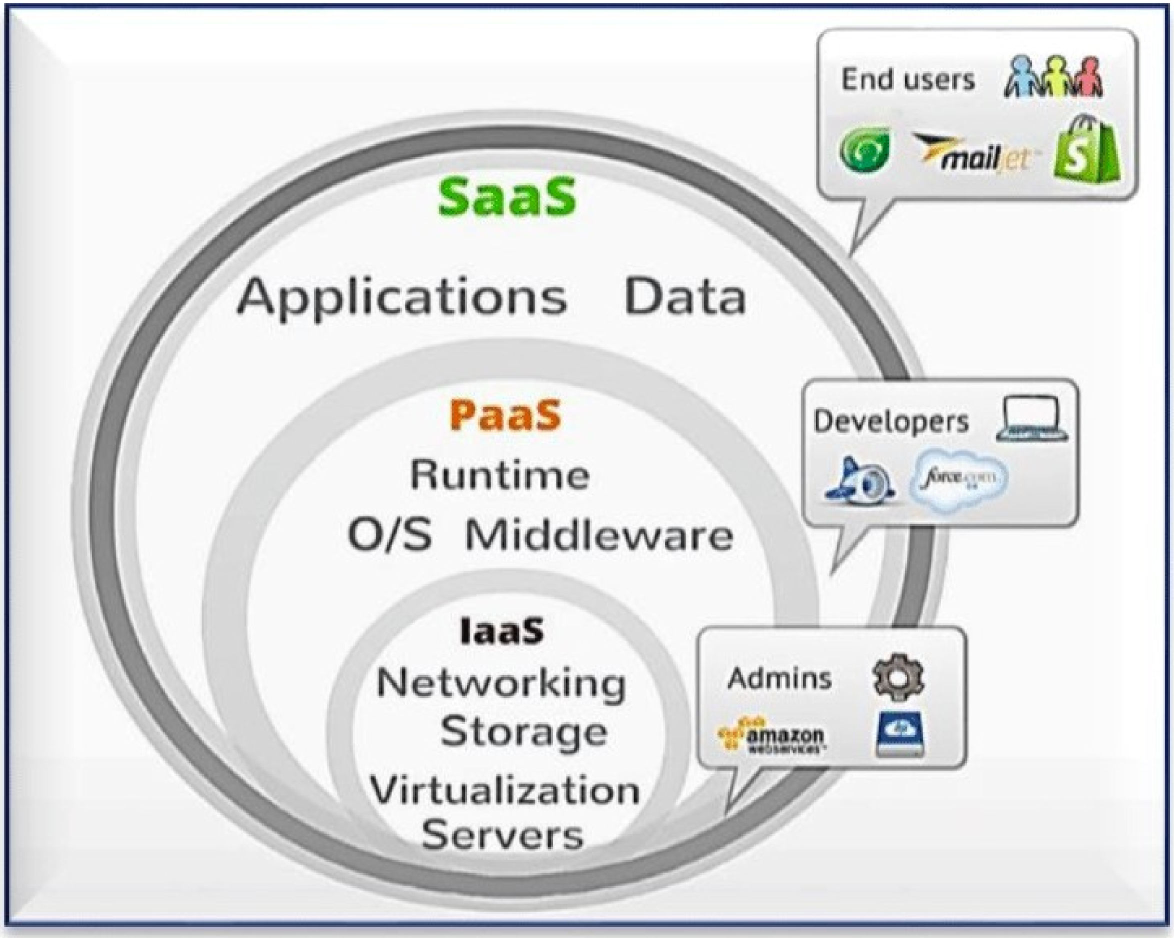


Infrastructure-as-a Service(Iaas)

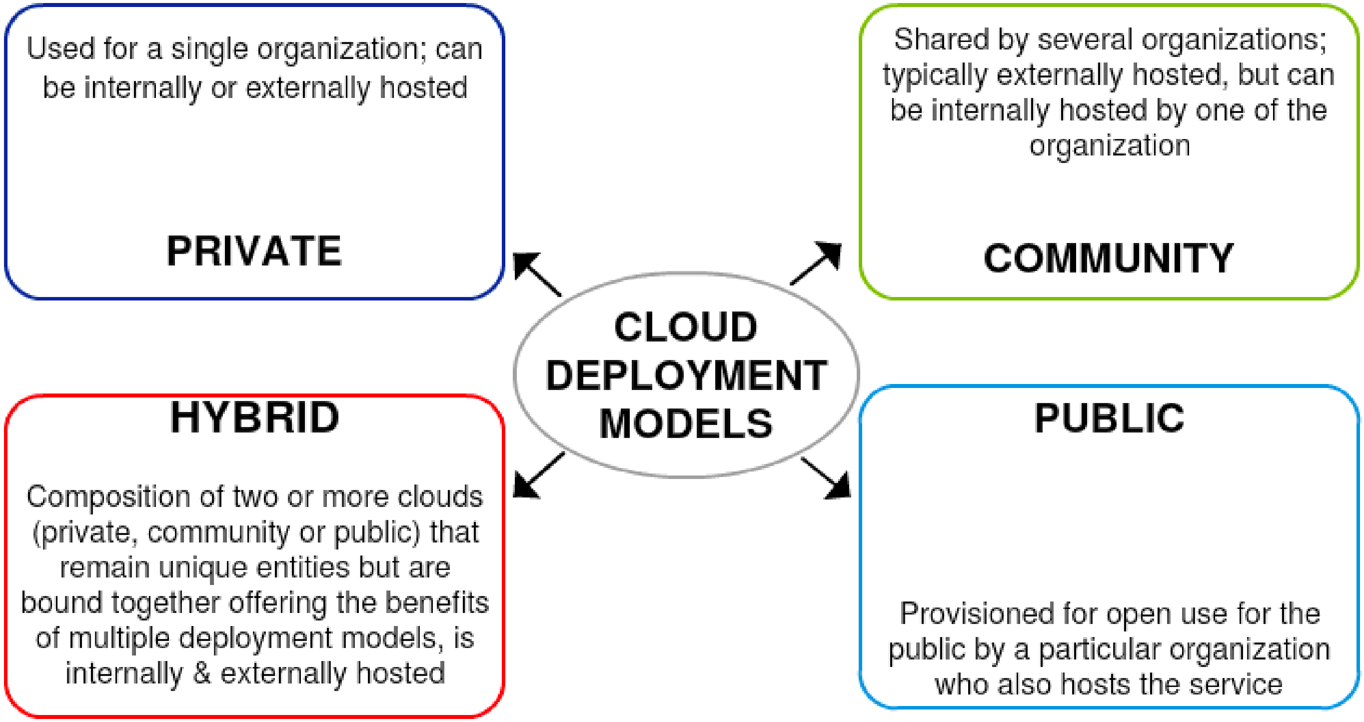
- Access of computing resources such as Operating System, Storage and Network
- Users can deploy and execute any software
- In some cases, users can control selected network components
- Ex : Amazon EC2, GoGrid



Cloud Computing Services



Cloud Computing : Deployment Models



Cloud Computing for IoT

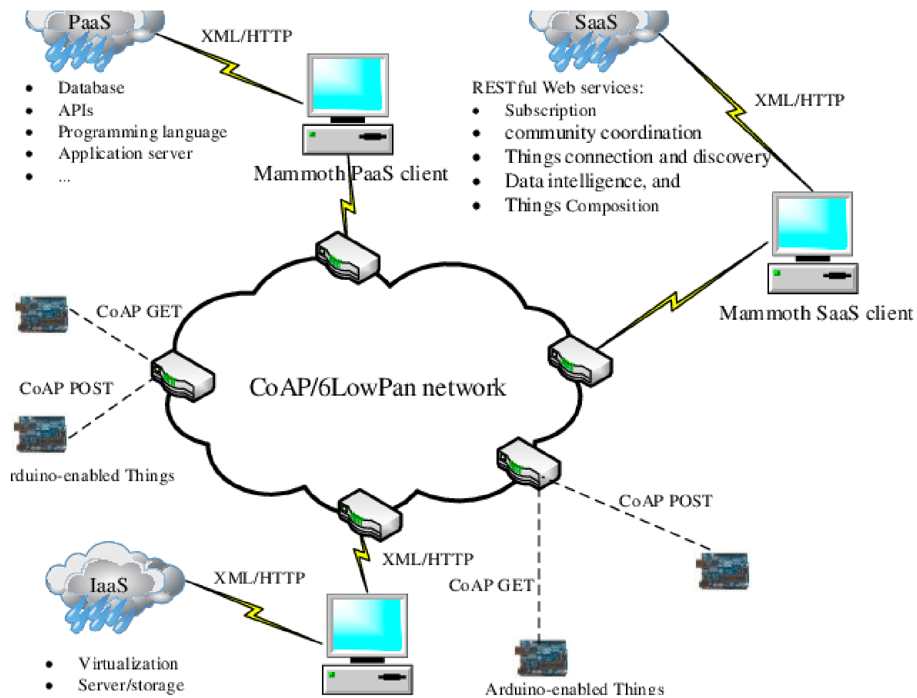
Cloud is an IoT enabler :

Huge amount of IoT Data needs storage, retrieval and management (sustained services)

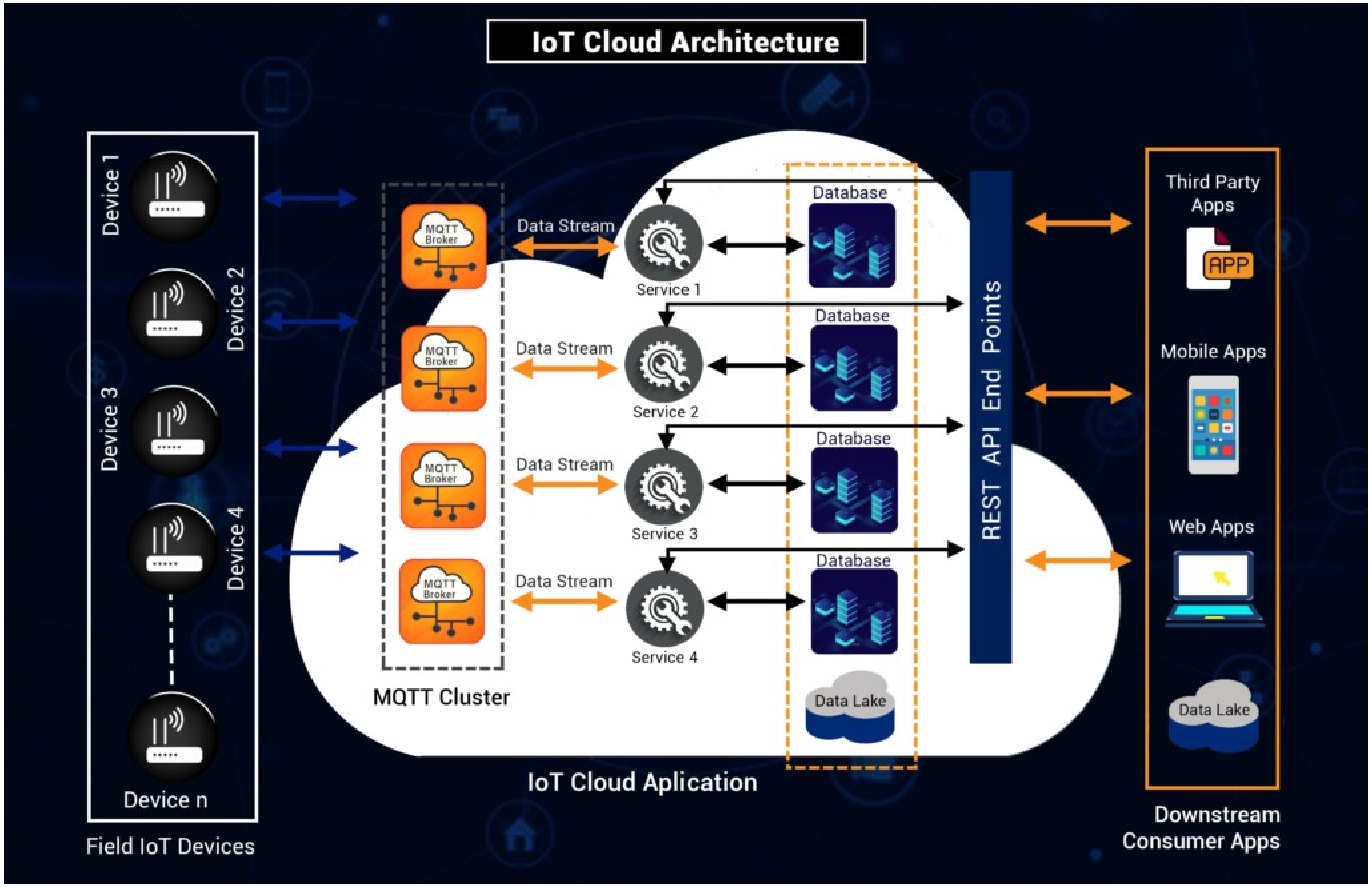
Fast Analytics for Prediction and Critical Decision making

Benefits of Cloud in IoT :

Scalability ; Data Mobility; Time to Market; Security; Cost effectiveness

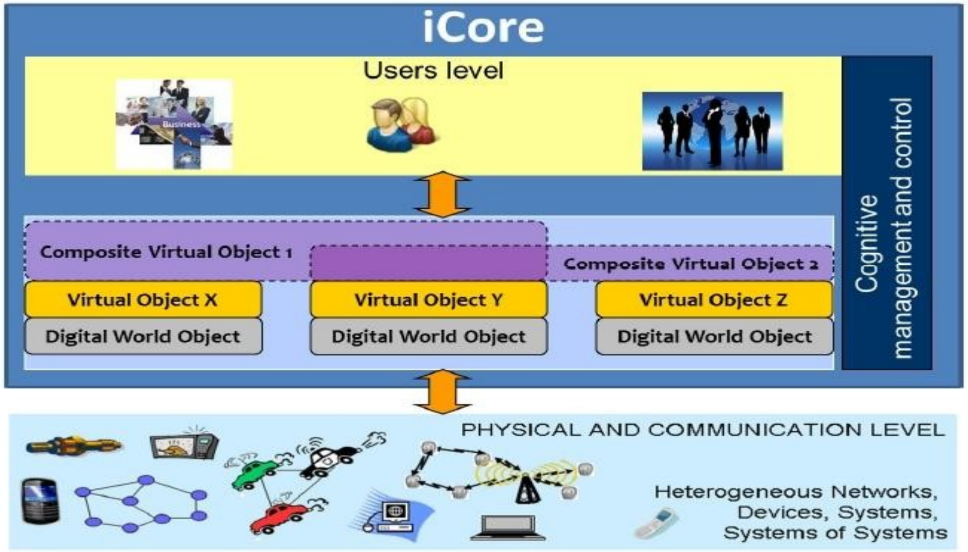


IoT Cloud Architecture



IoT Management

- **IoT-A Project [1] : physical entities represented as virtual entities access point to the real world by IoT apps through well defined and standardized interfaces**
- **European FP7 iCore project [2] : virtual object as alter ego of real world object, dynamically created and destroyed. VOs interact giving them some Cognitive capabilities**



[1] www.iot-a.eu

[2] Giaffreda : iCore: A Cognitive Management Framework for the Internet of Things, In: Galis A., Gavras A. (eds) The Future Internet. FIA 2013. Lecture Notes in Computer Science, vol 7858.

Cloud Based IoT Platforms - Characteristics

- Objects use HTTP protocol to send and receive data; allows interoperability among platforms
- Objects do no directly communicate with each other. An intermediate server is used
- Every object has a data point associated with it on the server side to keep track of the data sent
- The methods POST and GET are are used to send and request data
- *Tag* is assigned to every data point
- Data point discovery is performed through tags through an internal search engine
- System identifies every object with its *API key*

RESTful Architecture:

Lightweight, scalable and then fits perfectly with the principles and the current protocols of the Internet.

Resources as representation of the objects, that are uniquely identified through Uniform Resource Identifiers (URIs).

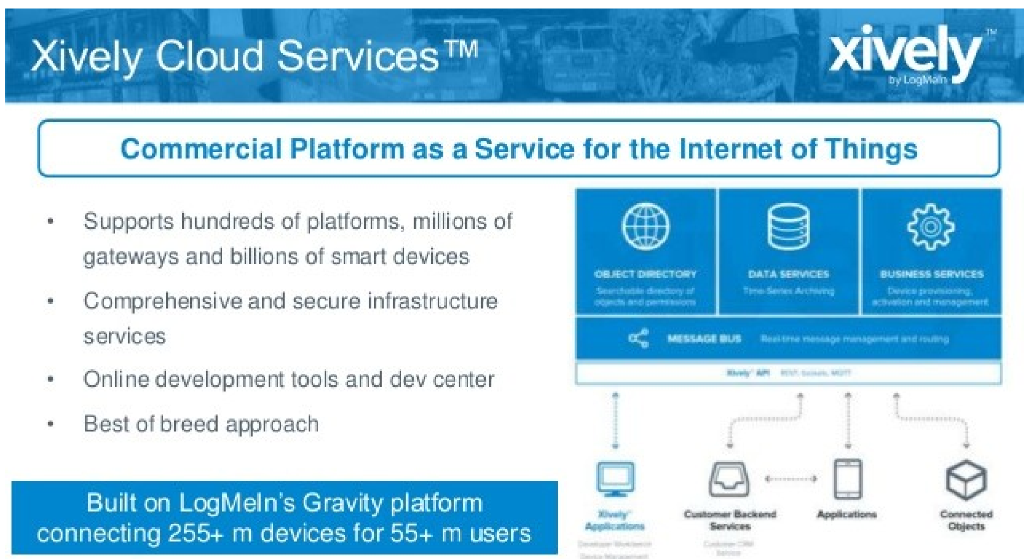
Object information can be obtained, deleted or posted through the HTTP protocol using a given method (GET, DELETE, POST, PUT).

The payload of the message can be incapsulated in a negotiated format such as XML or JSON.

IoT Platform Implementation - 1

Cosm (formerly Pachube) [6] : platform to store and redistribute real-time data, freely usable, which manages millions of devices per day. Following the nuclear accidents in Japan in 2011, Cosm was used by volunteers to interlink Geiger counters across the country to monitor the fallout.

Xively (formerly Cosm) [7] : commercial IoT platform to transmit, store and access to data generated by objects. It exploits PaaS partially (data stored in provider DB only). Now owned by Google.



[3] http://www.cosm.com
[7] http://www.xively.com

IoT Platform Implementation - 2

Nimbits [8] :

- **open source web application built on Google App engine**
- **Provides email alert, math calculations**
- **Storing and processing data**
- **User can define data points and use them to share several kinds of data**
- **Integrated with Twitter, Facebook**
- **Allows to manage data points, share sensor diagrams**

[8] http://nimbits.com

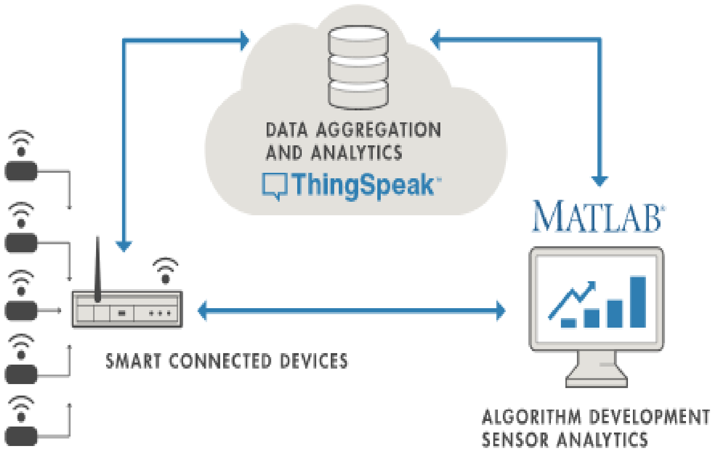
IoT Platform Implementation - 3

MATLAB ThinkSpeak

ThingSpeak™ is an IoT analytics service that allows you to aggregate, visualize, and analyze live data streams in the cloud. ThingSpeak provides instant visualizations of data posted by your devices to ThingSpeak. With the ability to execute MATLAB® code in ThingSpeak, you can perform online analysis and process data as it comes in. ThingSpeak is often used for prototyping and proof-of-concept IoT systems that require analytics.

You can send data from any internet-connected device directly to ThingSpeak using a Rest API or MQTT. In addition, cloud-to-cloud integrations with The Things Network, Senet, the Libelium Meshlium gateway, and Particle.io enable sensor data to reach ThingSpeak over LoRaWAN® and 4G/3G cellular connections.

With ThingSpeak, you can store and analyze data in the cloud without configuring web servers, and you can create sophisticated event-based email alerts that trigger based on data coming in from your connected devices.



https://in.mathworks.com/products/thingspeak.html