NPTEL ONLINE CERTIFICATION COURSES

Cloud Computing

Prof. Soumya K Ghosh

Department of Computer Science and Engineering

Module 09: Cloud Computing Paradigm

Lecture 41: Cloud–Fog Computing - Overview

CONCEPTS COVERED

- Cloud-Fog Paradigm - Overview
- Cloud-Fog-Edge/IoT
- Case Study: Health Cloud-Fog Framework
- Case Study: Performance

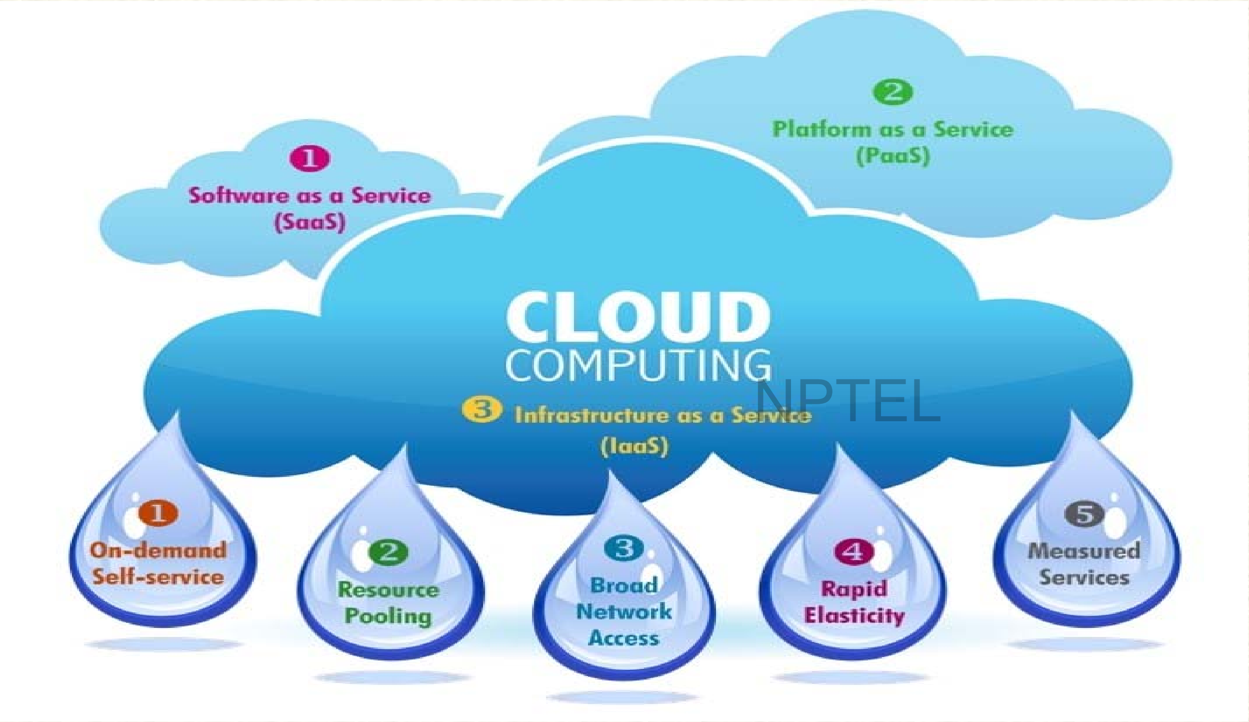
KEYWORDS

- Cloud Computing
- Fog Computing
- Edge, IoT, Sensors
- Performance analysis

Cloud – Fog Computing Paradigm

NPTEL

Cloud Computing: “Anything”-as-a-Service



Fog Computing

- Fog computing a model in which data, processing and applications are concentrated in devices at the network edge rather than existing almost entirely in the cloud.
- The term "Fog Computing" was introduced by the Cisco Systems as new model to ease wireless data transfer to distributed devices in the Internet of Things (IoT) network paradigm
- Vision of fog computing is to enable applications on billions of connected devices to run directly at the network edge.

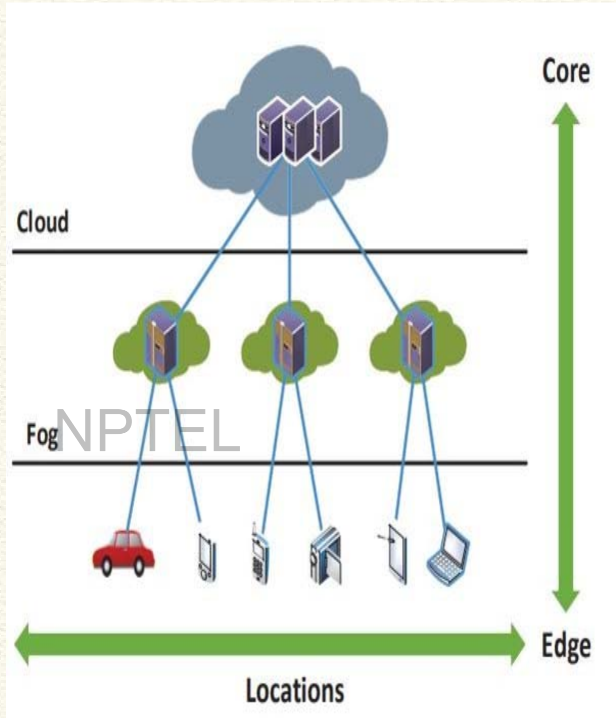
Cloud vs Fog Computing

Requirement	Cloud computing	Fog computing
Latency	high	low
Delay jitter	High	Very low
Location of server nodes	With in internet	At the edge of local n/w
Distance between the client and server	Multiple hops	One hop
Security	Undefined	Can be defined
Attack on data enrouter	High probability	Very Less probability
Location awareness	No	Yes

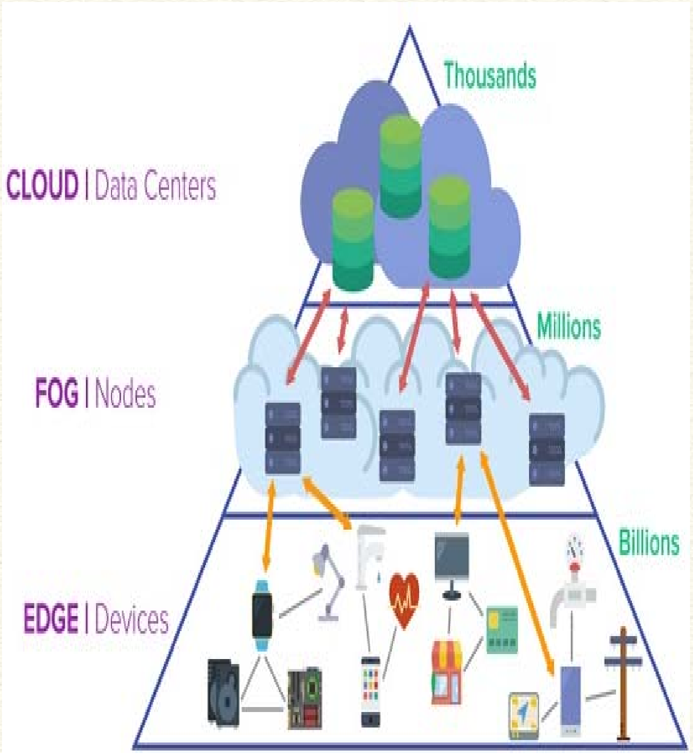
Requirement	Cloud computing	Fog computing
Geographicaldistribution	Centralized	Distributed
No. of server nodes	Few	Very large
Support for Mobility	Limited	Supported
Real time interactions	Supported	Supported
Type of last mile connectivity	Leased line	Wireless

Cloud-Fog-Edge Computing

- Bringing intelligence down from the cloud close to the ground/ end-user.
- Cellular base stations, Network routers, WiFi Gateways will be capable of running applications.
- End devices, like sensors, are able to perform basic data processing.
- Processing close to devices lowers response time, enabling real-time applications.



Cloud – Fog – Edge Computing



Source: https://www.learntechnology.com/network/fog-computing/

Cloud Issues Limitations

- Latency
- Large volume of data being generated.
- Bandwidth requirement
- Not designed for volume, variety and velocity of data generated by IoT devices

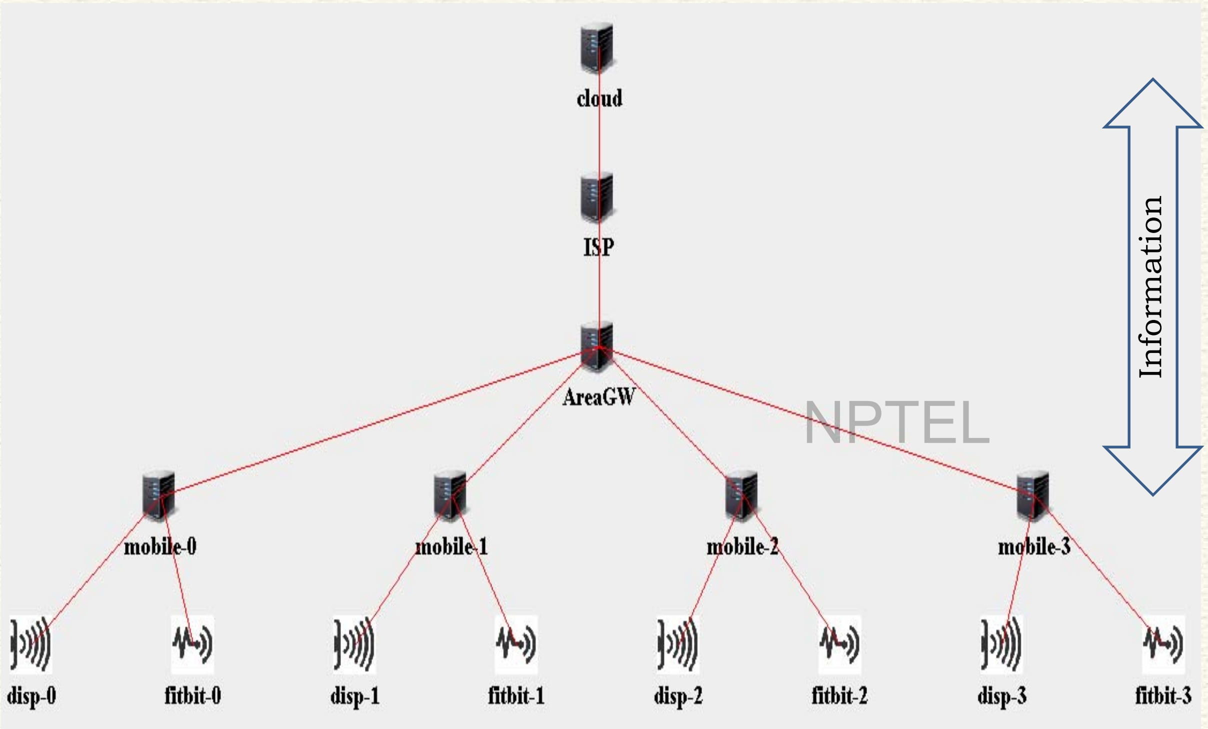
IoT Device (Edge) Issues

- Processing
- Storage
- Power requirement

Fog layer

- Much lesser **latency** permits usage in Real-time applications
- Less n**etwork congestion**
- Reduced **cost of execution** at cloud
- More of data location awareness
- Better handling of colossal data generated by sensors

Case Study: Health Cloud-Fog-Edge



Level0: Cloud

Level1: ISP

Level2: AreaGW

Level3: Mobile

Level4: IoT devices

Case Study: Health Cloud-Fog-Edge

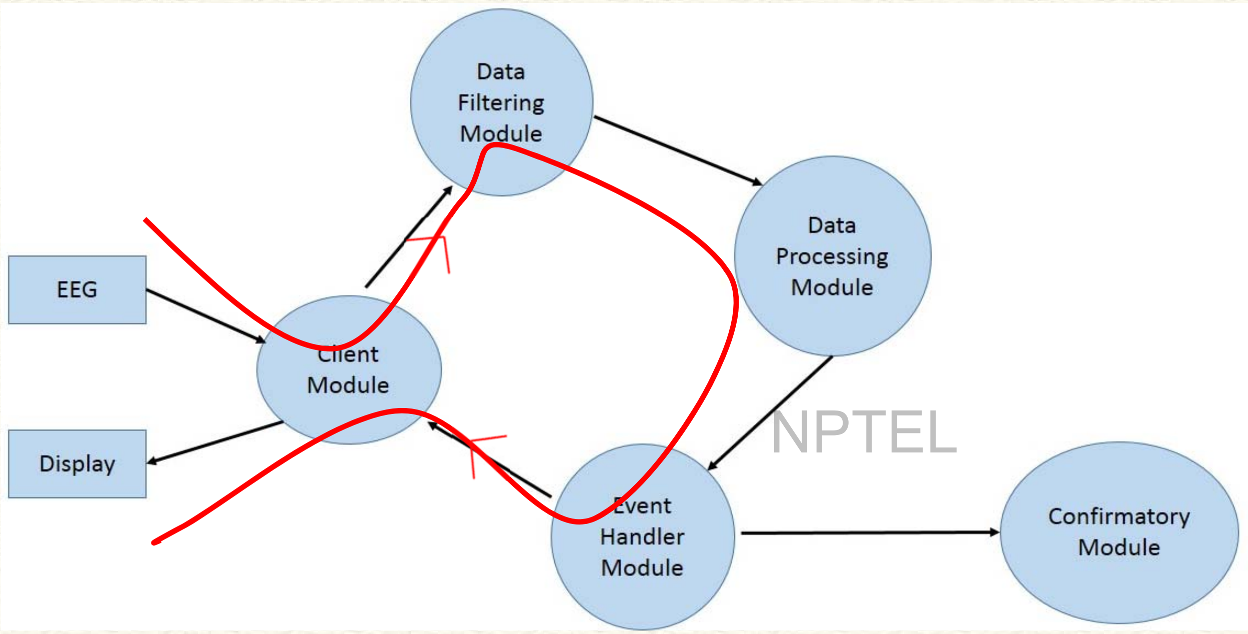
Device Configuration

Device	MIPS	RAM	Up Bw	Down Bw	Level	Cost/ MIPS	Busy Power	Idle Power
Cloud	44800	40000	100	10000	0	0.01	16*103	16*83.25
ISP	2800	4000	10000	10000	1	0	107.339	83.4333
AreaGW	2800	4000	10000	10000	2	0	107.339	83.4333
Mobile	350	1000	10000	270	3	0	87.53	82.44

Latency

Source	Destination	Latency
Fitbit	Mobile	1
Mobile	Area GW	2
Area GW	ISP GW	2
ISP GW	Cloud	100
Mobile	Display	1

Case Study: Health Cloud-Fog-Edge



Typical Application Components and Flow

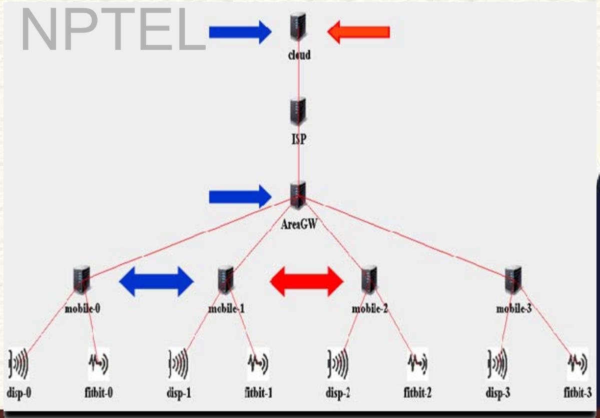
Case Study: Simulation of Health Cloud-Fog Model

Placement obtained for different Application Modules for Fog and Cloud architecture:

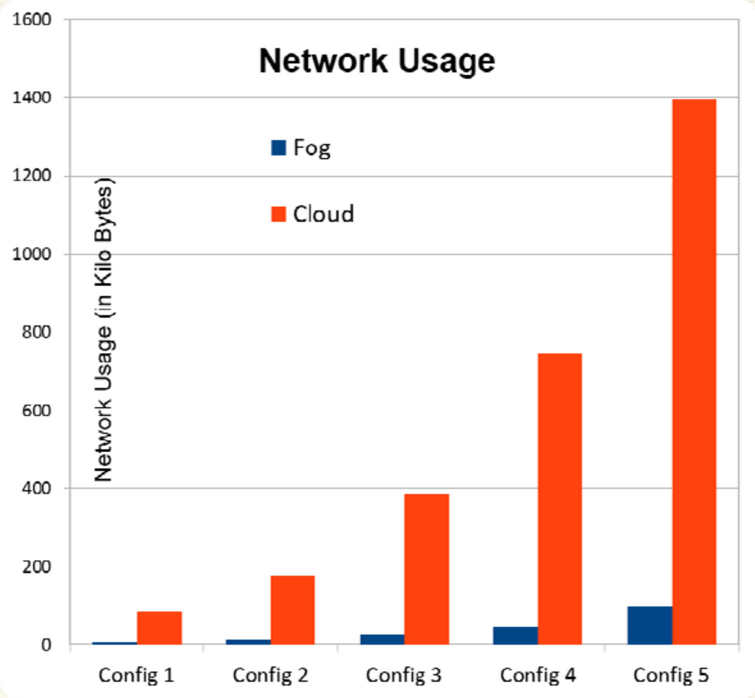
Application Module	Placement in Fog based Model	Placement in Cloud based Model
Client Module	Mobile	Mobile
Data Filtering Module	Area Gateway	Cloud
Data Processing Module	Area Gateway	Cloud
Event Handler Module	Area Gateway	Cloud
Confirmatory Module	Cloud	Cloud

Simulation of different configurations in iFogSim:

Configuration	No. of AreaGW	Total No. of Users
1	1	4
2	2	8
3	4	16
4	8	32
5	16	64



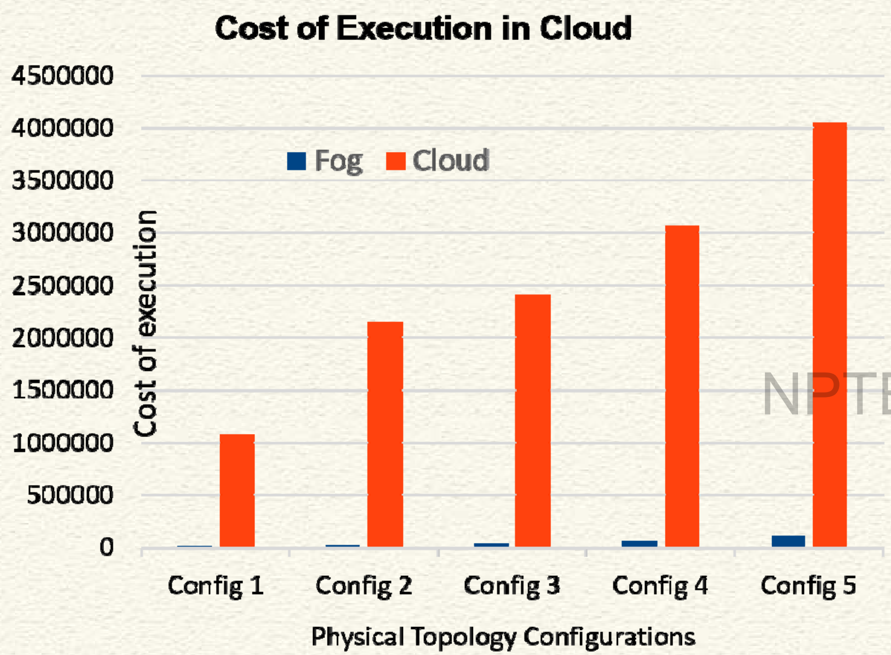
Case Study: Performance Analysis - Network Usage



Physical Topology Configurations

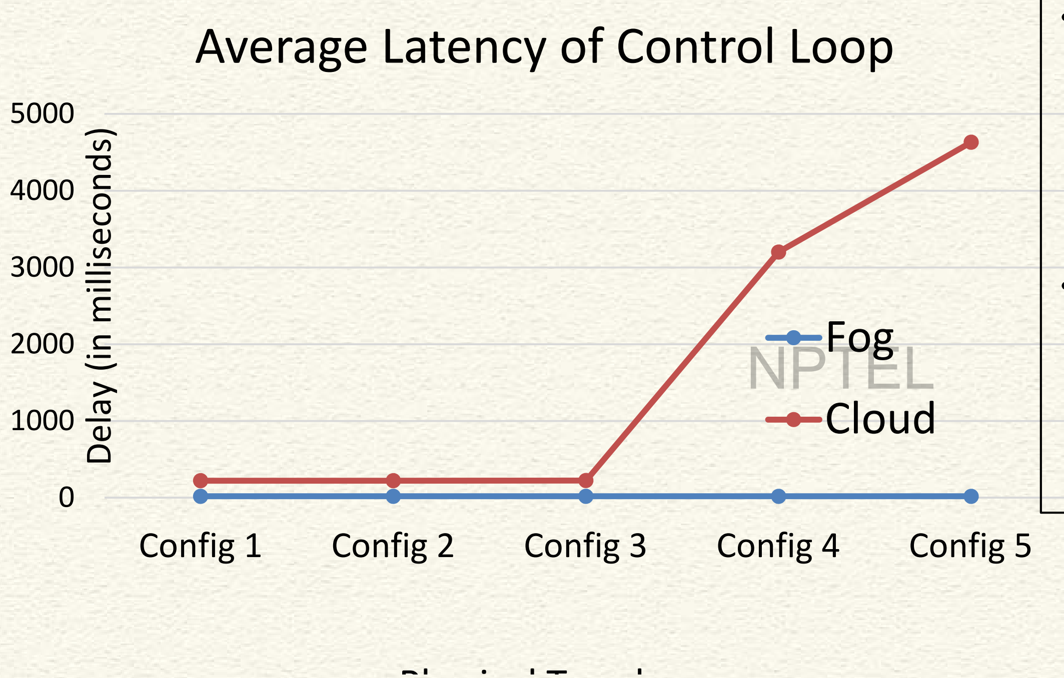
- Network usage is very low in case of Fog architecture as only for few positive cases, the Confirmatory module residing on Cloud is accessed.
- In case of Cloud based architecture, the usage is high as all modules are now on Cloud.

Case Study: Performance Analysis – Execution Cost



- Only the resources on Cloud incur cost, other resources are owned by the organization.
- More processing at Cloud leads to higher costs in case of Cloud based architecture.

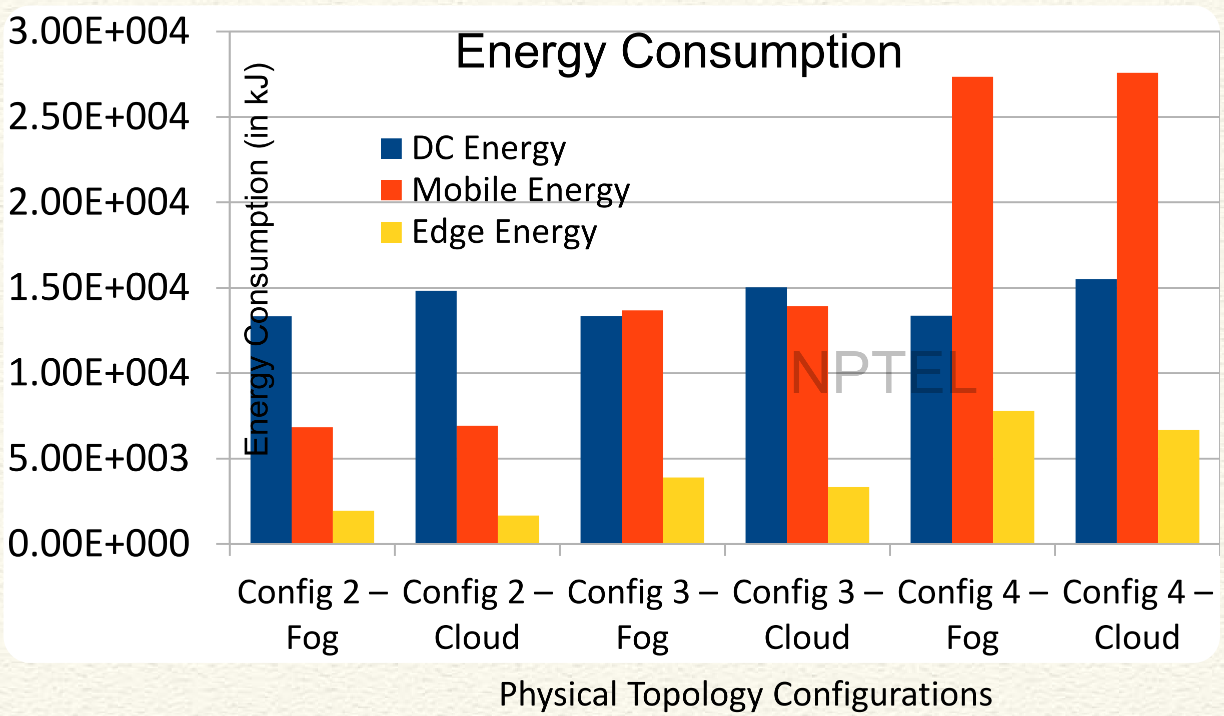
Case Study: Performance Evaluation - Latency



In this case:

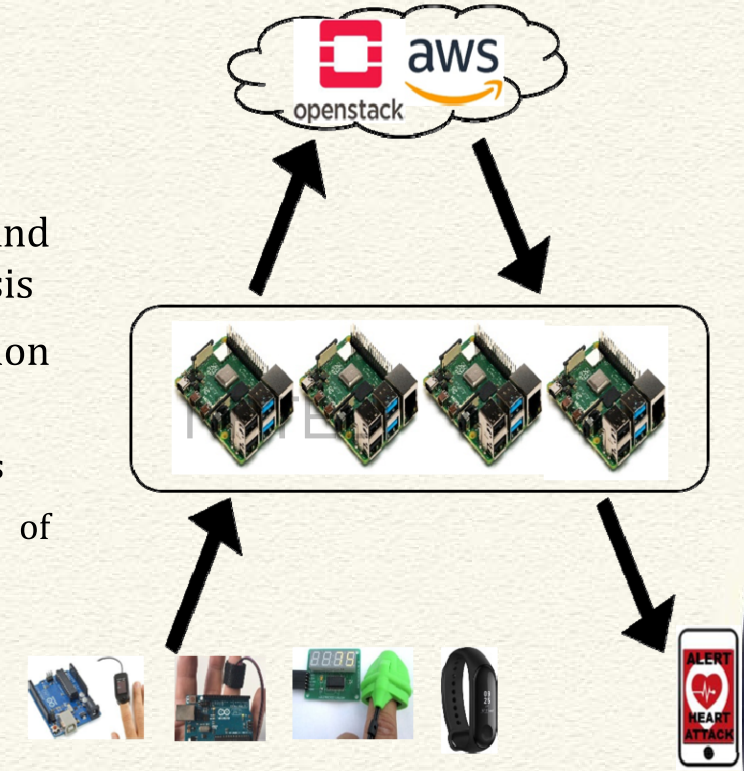
- Latency is fixed in case of Fog architecture as the application modules which form part of the control loop are located at Area Gateway itself.
- The modules are located at the Datacenter in case of Cloud based architecture.

Case Study: Energy Consumptions



Case Study: Prototype Implementation

- Lab based setup:
  - Raspberry Pi as Fog Devices
  - AWS as Cloud
- Use different dataset and customize formulae for analysis
- Changes in Resource Allocation Policy in terms of:
  - Customized physical devices
  - Customized requirements of Application Modules
  - Module Placement policy



REFERENCES

- Cisco White Paper. 2015. Fog Computing and the Internet of Things: Extend the Cloud to Where the Things Are.
- Gupta H, Vahid Dastjerdi A, Ghosh SK, Buyya R. iFogSim: A toolkit for modeling and simulation of resource management techniques in the Internet of Things, Edge and Fog computing environments. *Softw Pract Exper. 2017;47:1275-296. https://doi.org/10.1002/spe.2509*
- Mahmud, Md & Buyya, Rajkumar. (2019). Modeling and Simulation of Fog and Edge Computing Environments Using iFogSim Toolkit: Principles and Paradigms. 10.1002/9781119525080.ch17
- Mahmud, Md and Koch, Fernando and Buyya, Rajkumar. (2018). Cloud-Fog Interoperability in IoT-enabled Healthcare Solutions. 10.1145/3154273.3154347. In proceedings of 19th International Conference on Distributed Computing and Networking, January 4–7, 2018, Varanasi, India. ACM, NewYork, NY, USA.

Thank you

NPTEL

NPTEL ONLINE CERTIFICATION COURSES

Cloud Computing

Prof. Soumya K Ghosh

Department of Computer Science and Engineering

Module 09: Cloud Computing Paradigm
Lecture 42: Resource Management - I

CONCEPTS COVERED

- Cloud-Fog Paradigm – Resource Management Issues
- Service Placement Problem

KEYWORDS

- Cloud Computing
- Fog - Edge Computing
- Resource Management
- Service Placement

Resource Management - I

NPTEL

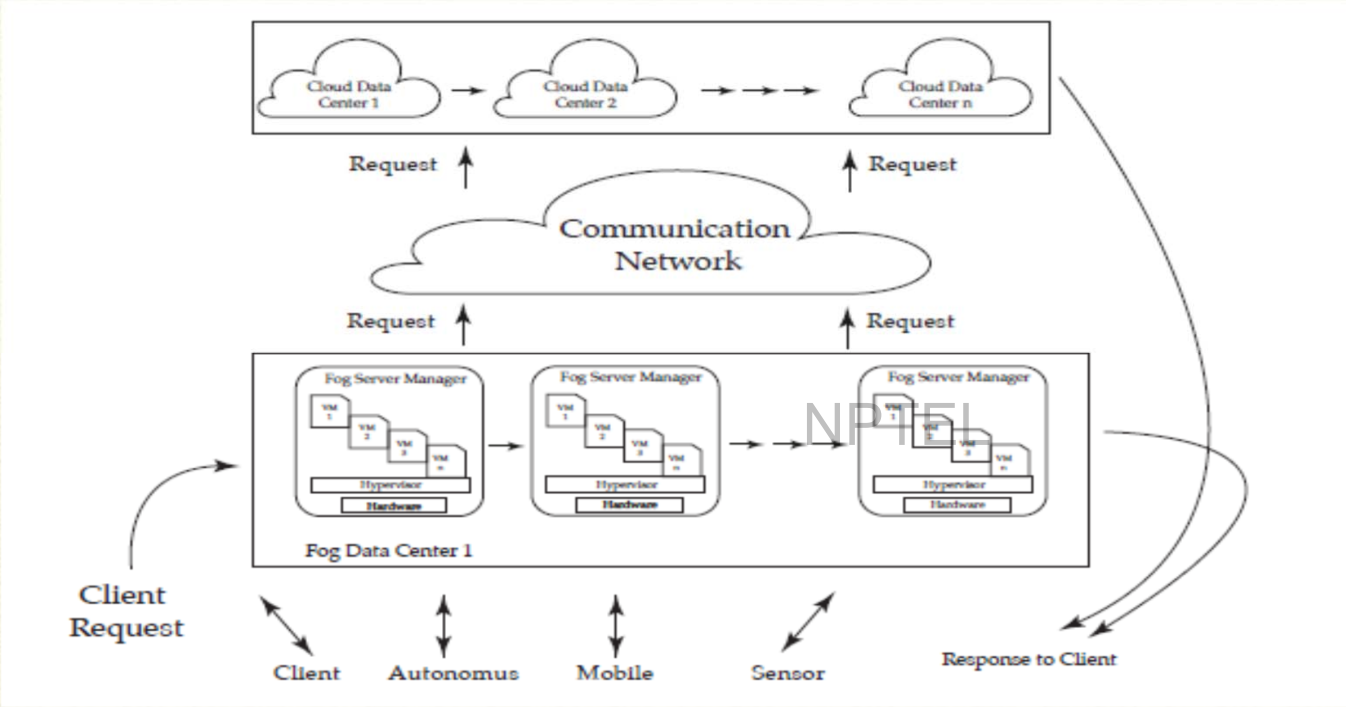
Challenges in “Cloud-only” scenario

- Processing IoT applications directly in the cloud may not be the most efficient solution for each IoT scenario, especially for time-sensitive applications.
- A promising alternative is to use fog and edge computing, which address the issue of managing the large data bandwidth needed by end devices.
- These paradigms impose to process the large amounts of generated data close to the data sources rather than in the cloud.
- One of the considerations of cloud-based IoT environments is resource management, which typically revolves around resource allocation, workload balance, resource provisioning, task scheduling, and QoS to achieve performance improvements.

Fog-Edge to support Cloud Computing

- Latency issue: May involve transport of data from each single sensor to a data center over a network, process these data, and then send instructions to actuators.
- Fog and edge computing may aid cloud computing in overcoming these limitations.
- Fog computing and edge computing are no substitutes for cloud computing as they do not completely replace it.
- Oppositely, the three technologies can work together to grant improved latency, reliability, and faster response times.

Cloud-Fog Paradigm



Ref: Agarwal, S.; Yadav, S.; Yadav, A.K. An efficient architecture and algorithm for resource provisioning in fog computing. Int. J. Inf. Eng. Electronic Bus. (IJIEEB) 2016, 8, 48–61.

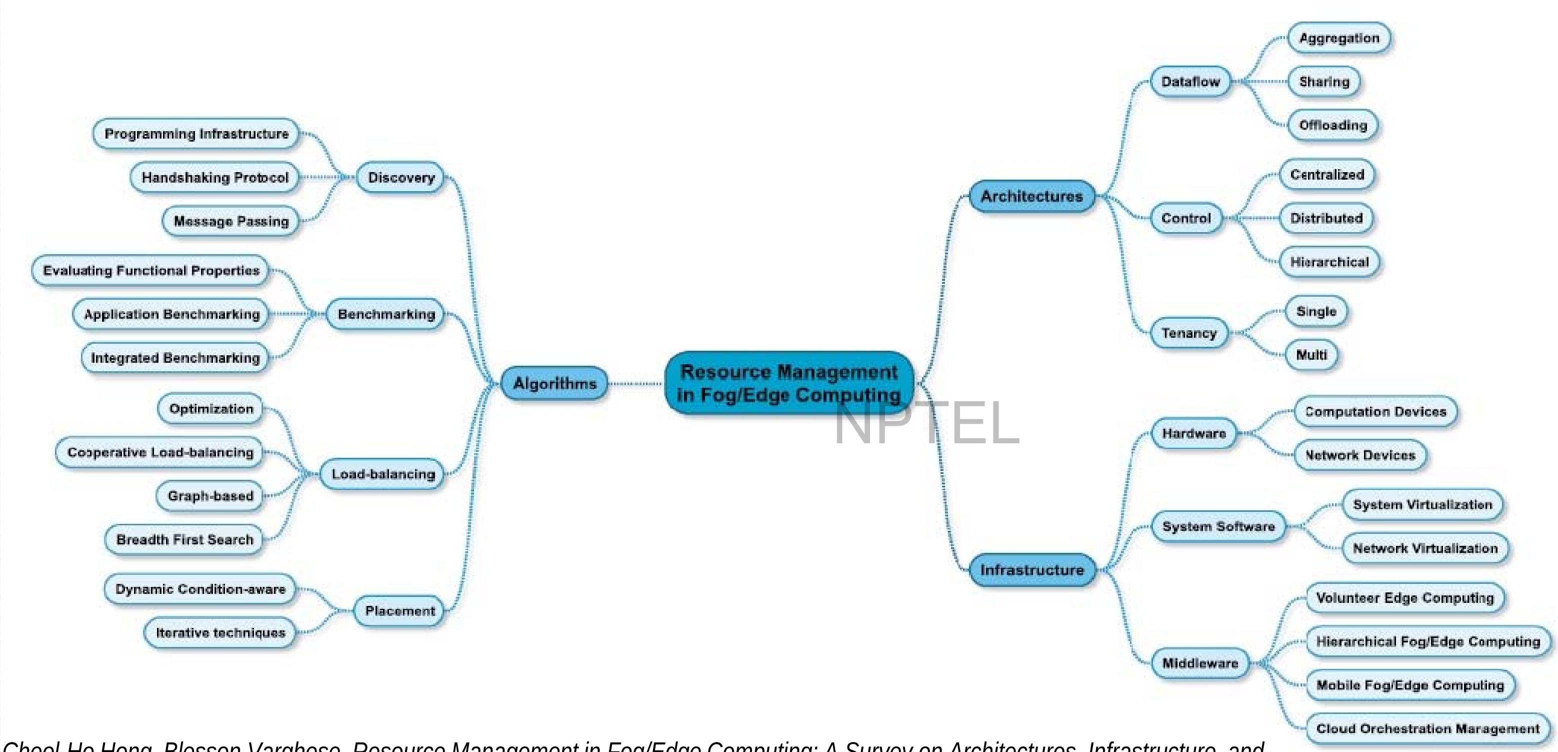
Fog-Edge to support Cloud Computing

- Cloud–fog environment model, typically, is composed of three layers: a client layer (edge), a fog layer, and a cloud layer.
- Fog layer is to accomplish the requirement of resources for clients.
- If there is no/limited availability of resources in the fog layer, then the request is passed to the cloud layer.
- Main functional components of this model are:
  - Fog server manager employs all the available processors to the client.
  - Virtual machines (VMs) operate for the fog data server, process them, and then deliver the results to the fog server manager.
  - Fog servers contain fog server manager and virtual machines to manage requests by using a 'server virtualization technique'.

Fog-Edge to support Cloud Computing

- Trend is to decentralize some of the computing resources available in large Cloud data centers by distributing them towards the edge of the network closer to the end-users and sensors
- Resources may take the form of either (i) dedicated “micro” data centers that are conveniently and safely located within public/private infrastructure or (i) Internet nodes, such as routers, gateways, and switches that are augmented with computing capabilities.
- *A computing model that makes use of resources located at the edge of the network is referred to as “edge computing”.*
- *A model that makes use of both edge resources and the cloud is referred to as “fog computing”*

Resource Management in Cloud-Fog-Edge



Cheol-Ho Hong, Blesson Varghese, Resource Management in Fog/Edge Computing: A Survey on Architectures, Infrastructure, and Algorithms, ACM Computing Surveys, Vol 52(5), October 2019, pp 1–37.

Resource Management Approaches

- **Architectures** - the architectures used for resource management in fog/edge computing are classified on the basis of data flow, control, and tenancy
- **Infrastructure** - The infrastructure for fog/edge computing provides facilities composed of hardware and software to manage the computation, network, and storage resources for applications utilizing the fog/edge.
- **Algorithms** - There are several underlying algorithms used to facilitate fog/edge computing.

Resource Management Approaches - Architectures

Architectures

- Data Flow
- Control
- Tenancy

- **Data Flow:** Based on the direction of movement of workloads and data in the computing ecosystem. For example, workloads could be transferred from the user devices to the edge nodes or alternatively from cloud servers to the edge nodes.
- **Control**: Based on how the resources are controlled in the computing ecosystem. For example, a single controller or central algorithm may be used for managing a number of edge nodes. Alternatively, a distributed approach may be employed.
- **Tenancy**: Based on the support provided for hosting multiple entities in the ecosystem. For example, either a single application or multiple applications could be hosted on an edge node.

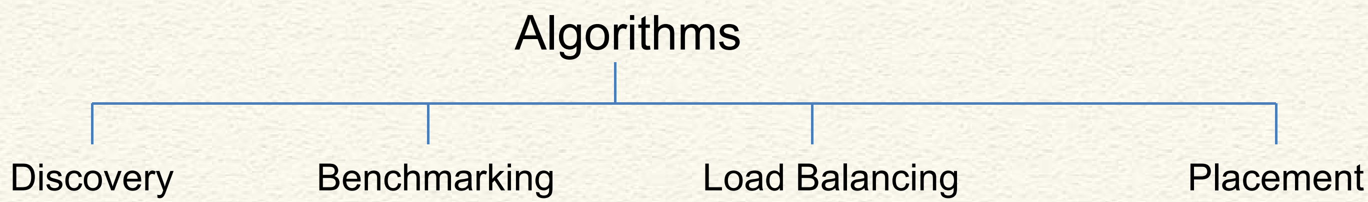
Resource Management Approaches - Infrastructure

Infrastructure

- Hardware
- System Software
- Middleware

- **Hardware:** Fog/ edge computing exploits small-form-factor devices such as network gateways, WiFi Access Points (APs), set-top boxes, small home servers, edge ISP servers, cars, and even drones as compute servers for resource efficiency. also utilizes commodity products such as desktops, laptops, and smartphones.
- **System Software**: Runs directly on fog/edge hardware resources such as the CPU, memory, and network devices. It manages resources and distributes them to the fog/edge applications. E.g., operating systems and virtualization software.
- **Middleware**: Runs on an operating system and provides complementary services that are not supported by the system software. The middleware coordinates distributed compute nodes and performs deployment of VMs or containers to each fog/edge node.

Resource Management Approaches – Algorithms



- **Discovery:** Identifying edge resources so workloads from the clouds or from user devices/ sensors can be deployed on them
- **Benchmarking**: Capturing the performance (of entities like, CPU, storage, network, etc.) of a computing system
- **Load balancing**: As edge data centers are deployed across the network edge, the issue of distributing tasks using an efficient load-balancing algorithm has gained significant attention. Typical techniques are, namely, optimization techniques, cooperative load balancing, graph-based balancing, and using breadth-first search.
- **Placement**: Addresses the issue of place incoming computation tasks on suitable fog/edge resources, considering the availability of resources in the fog/edge layer and the environmental changes. Dynamic condition-aware techniques and iterative techniques.

REFERENCES

- Cheol-Ho Hong, Blesson Varghese, Resource Management in Fog/Edge Computing: A Survey on Architectures, Infrastructure, and Algorithms, ACM Computing Surveys, Vol 52(5), October 2019, pp 1–37.
- Farah Aït Salaht, Frédéric Desprez, and Adrien Lebre. 2020. An Overview of Service Placement Problem in Fog and Edge Computing. ACM Comput. Surv. 53, 3, Article 65 (June 2020), 35 pages.
- Agarwal, S.; Yadav, S.; Yadav, A.K. An efficient architecture and algorithm for resource provisioning in fog computing. Int. J. Inf. Eng. Electronic Bus. (IJIEEB) 2016, 8, 48–61.

Thank you

NPTEL ONLINE CERTIFICATION COURSES

Cloud Computing

Prof. Soumya K Ghosh

Department of Computer Science and Engineering

Module 09: Cloud Computing Paradigm
Lecture 43: Resource Management - II

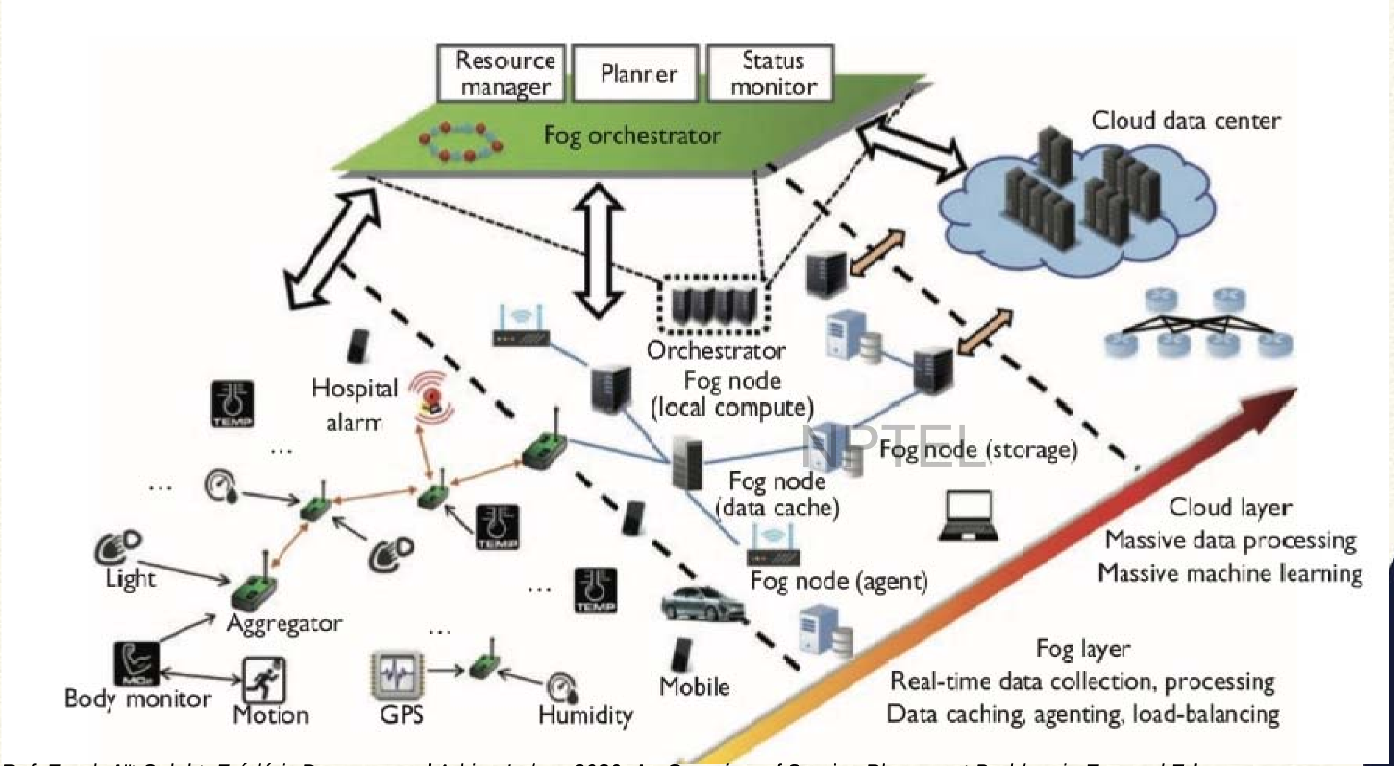
CONCEPTS COVERED

- Cloud-Fog Paradigm – Resource Management Issues
- Service Placement Problem
- Service and Data Offloading
- Hardware and Software

KEYWORDS

- Cloud Computing
- Fog - Edge Computing
- Resource Management
- Service Placement

Service Placement Problem in Fog and Edge Computing

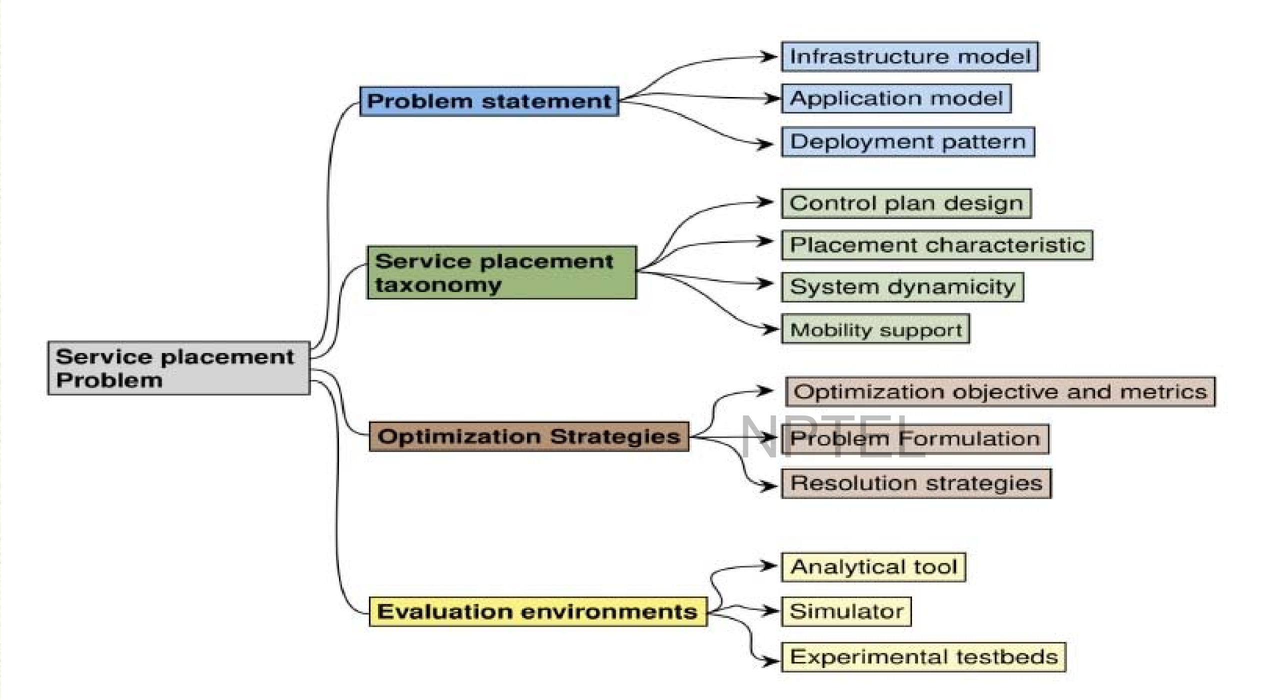


Ref: Farah Aït Salaht, Frédéric Desprez, and Adrien Lebre. 2020. An Overview of Service Placement Problem in Fog and Edge Computing. ACM Comput. Surv. 53, 3, Article 65 (June 2020), 35 pages.

Service Placement Problem in Fog and Edge Computing

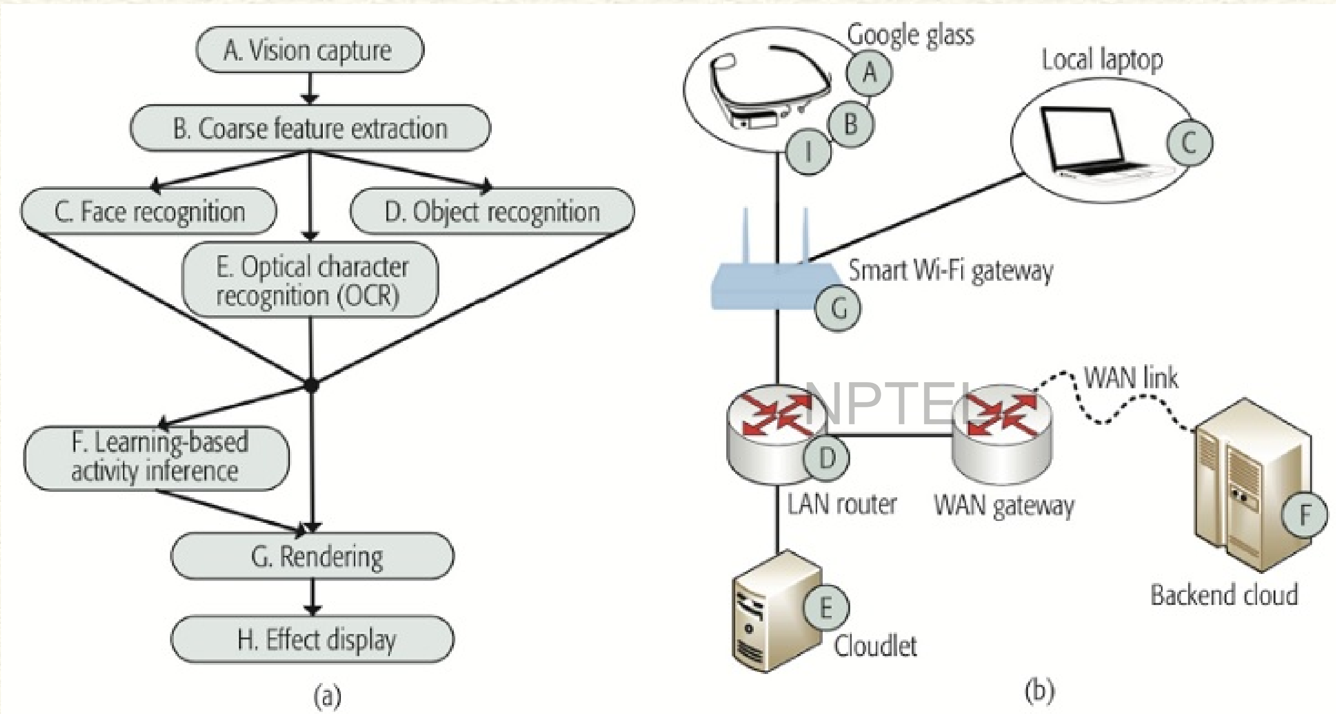
- Fog Computing is a highly virtualized platform that offers computational resources, storage, and control between end-users and Cloud servers.
- It is a new paradigm in which centralized Cloud coexists with distributed edge nodes and where the local and global analyses are performed at the edge devices or forwarded to the Cloud.
- Fog infrastructure consists of IoT devices (End layer), Fog Nodes, and at least one Cloud Data Center (Cloud layer), with following characteristics:
  - Location awareness and low latency
  - Better bandwidth utilization
  - Scalable
  - Support for mobility

Service Placement Problem in Fog and Edge Computing



Ref: Farah Aït Salaht, Frédéric Desprez, and Adrien Lebre. 2020. An Overview of Service Placement Problem in Fog and Edge Computing. ACM Comput. Surv. 53, 3, Article 65 (June 2020), 35 pages.

Deployment (Application Placement) on Cloud-Fog-Edge framework



Ref: *Farah Aït Salaht, Frédéric Desprez, and Adrien Lebre. 2020. An Overview of Service Placement Problem in Fog and Edge Computing. ACM Comput. Surv. 53, 3, Article 65 (June 2020), 35 pages.*

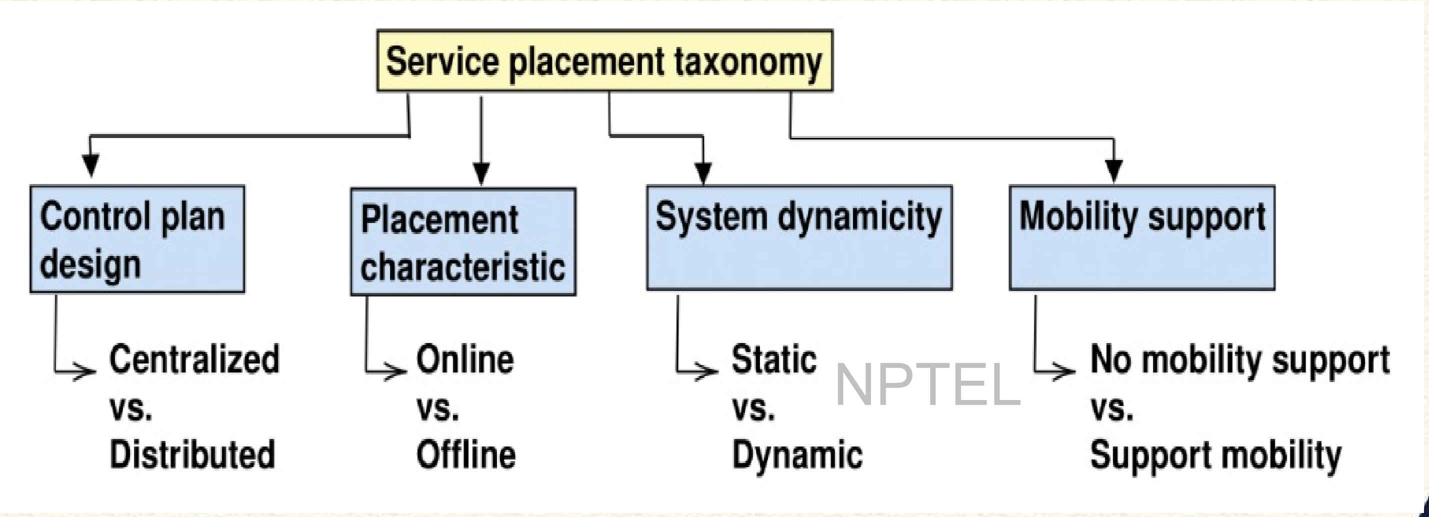
Application Placement on Cloud-Fog-Edge framework

- Application placement problem defines a mapping pattern by which applications components and links are mapped onto an infrastructure graph (i.e., computing devices and physical edges)
- Application placement involves finding the available resources in the network (nodes and links) that satisfy the application(s) requirements, satisfy the constraints, and optimize the objective.
- For instance, respect the applications (services) requirements, not exceed the resource capacities, satisfy the locality constraints, minimize the energy consumed, and so on.
- Service providers have to take into account these constraints to, (i) limit the research space and, (ii) provide an optimum or near-optimum placement

Application Placement - Constraints

- **Resource constraints**: An infrastructure node is limited by finite capabilities in terms of CPU, RAM, storage, bandwidth, etc. While placing application(s) (service components), the resource requirements need to be considered
- **Network constraints**: constraints such as latency, bandwidth, etc. and these constraints need to be considered when deploying applications.
- **Application constraints**:
  - Locality requirement: restricts certain services' executions to specific locations
  - Delay sensitivity: Some applications can specify a deadline for processing operation or deploying the whole application in the network

Service Placement Taxonomy



Ref: Farah Aït Salaht, Frédéric Desprez, and Adrien Lebre. 2020. An Overview of Service Placement Problem in Fog and Edge Computing. ACM Comput. Surv. 53, 3, Article 65 (June 2020), 35 pages.

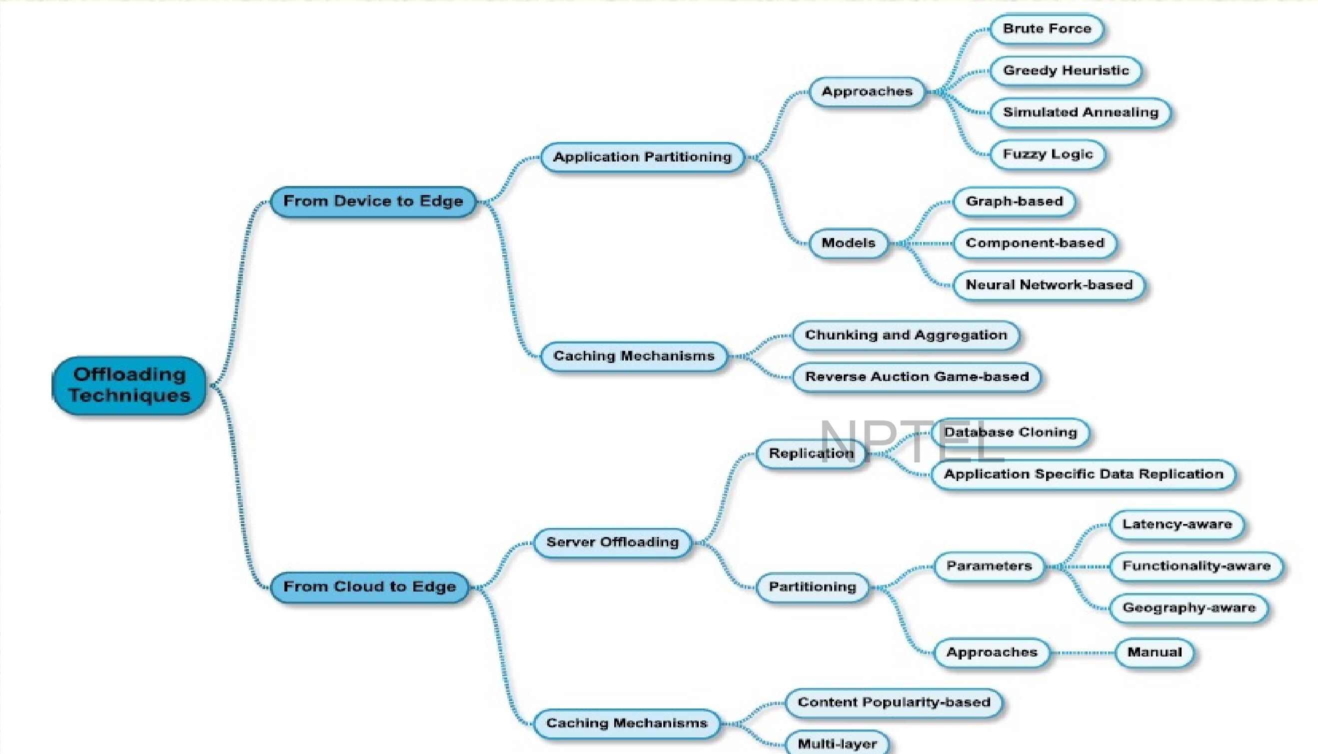
Service Placement – Optimization Strategies

- Optimizing the service placement problem in a Cloud-Fog infrastructure can have several different objectives, with different formulations and diverse algorithm proposals.
  - Optimization Objective and Metrics :
    - Latency
    - Resource utilization
    - Cost
    - Energy consumption

Offloading – Application and Data

- Offloading is a technique in which a server, an application, and the associated data are moved onto the edge of the network.
- Augments the (i) computing requirements of individual or a collection of user devices, (ii) brings services in the cloud that process requests from devices closer to the source.
- *Offloading from User Device to Edge*: Augments computing in user devices by making use of edge nodes (usually a single hop away)
  (i) Application partitioning, (ii) Caching mechanisms
- *Offloading from the Cloud to the Edge*: A workload is moved from the cloud to the edge.
  (i) Server offloading, (ii) Caching mechanisms

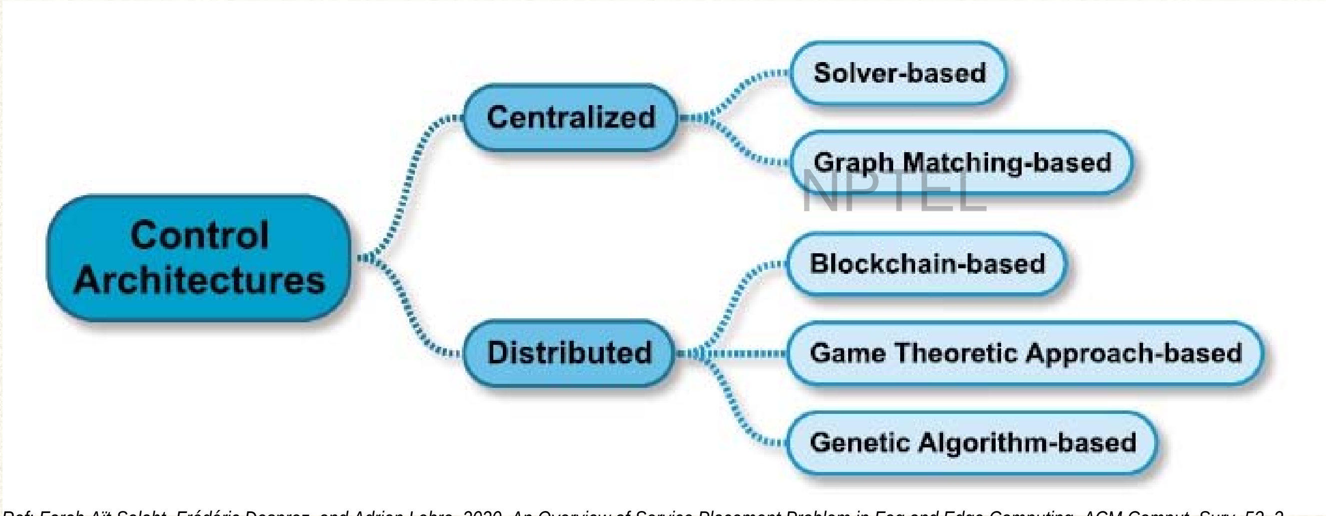
Offloading – Application and Data (contd.)



Ref: Farah Aït Salaht, Frédéric Desprez, and Adrien Lebre. 2020. An Overview of Service Placement Problem in Fog and Edge Computing. ACM Comput. Surv. 53, 3, Article 65 (June 2020), 35 pages.

Control

- Centralized
- Distributed

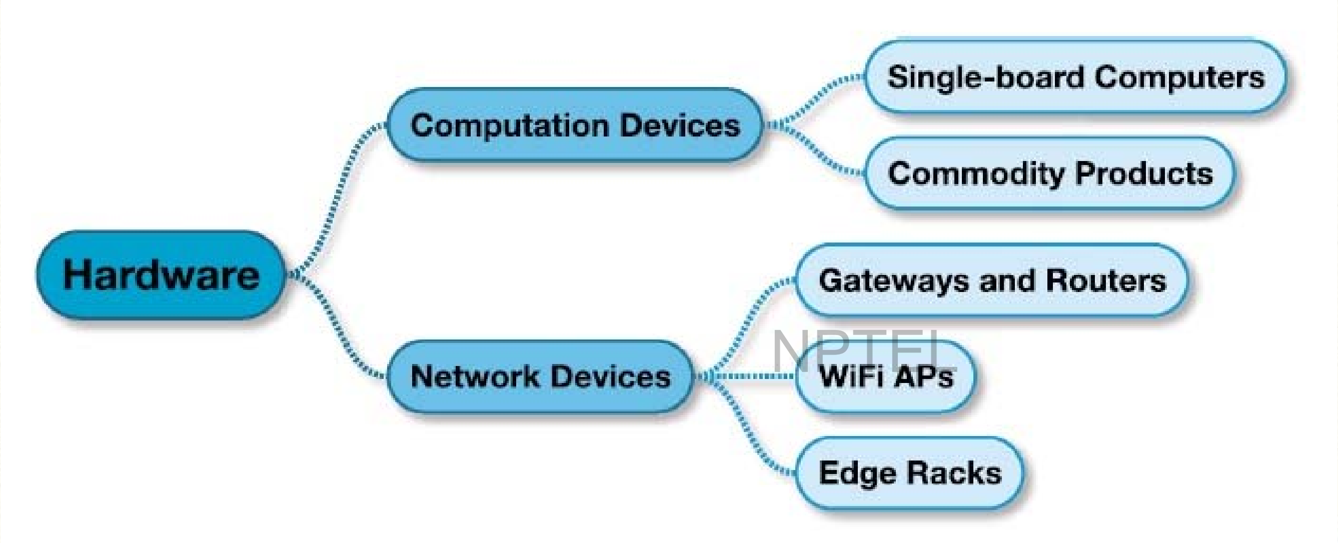


Ref: Farah Aït Salaht, Frédéric Desprez, and Adrien Lebre. 2020. An Overview of Service Placement Problem in Fog and Edge Computing. ACM Comput. Surv. 53, 3, Article 65 (June 2020), 35 pages.

Hardware

- Fog/edge computing forms a computing environment that uses low-power devices, namely, mobile devices, routers, gateways, home systems.
- Combination of these small-form-factor devices, connected to network, enables a cloud computing environment that can be leveraged by a rich set of applications processing Internet of Things (IoT) and cyber-physical systems (CPS) data.

Hardware (contd..)

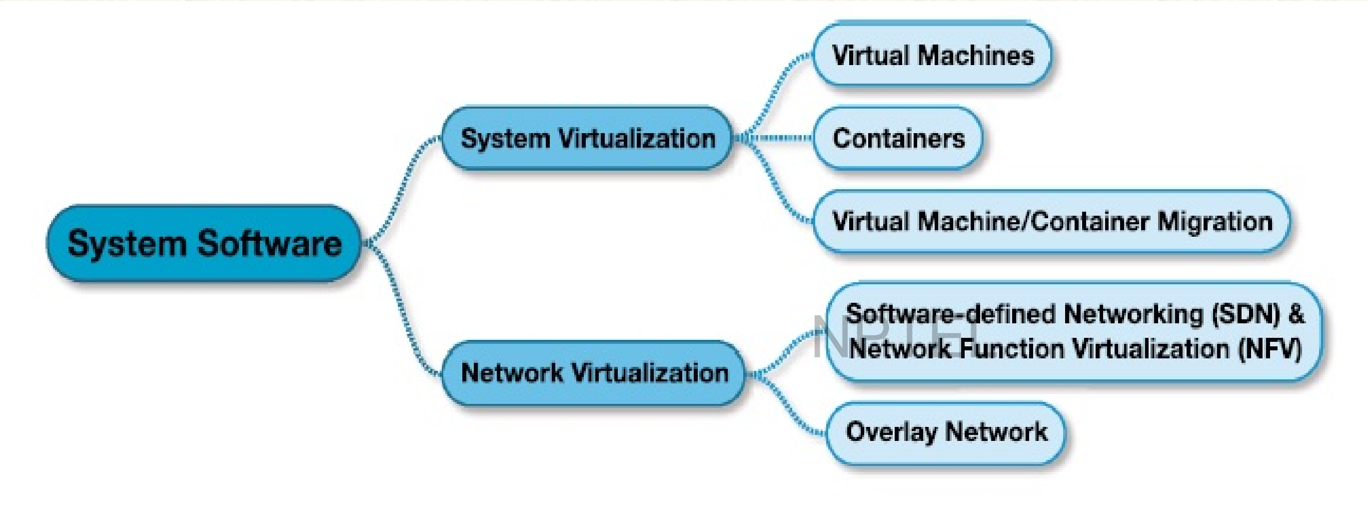


Ref: Farah Aït Salaht, Frédéric Desprez, and Adrien Lebre. 2020. An Overview of Service Placement Problem in Fog and Edge Computing. ACM Comput. Surv. 53, 3, Article 65 (June 2020), 35 pages.

System Software

- System software for the fog/edge is a platform designed to operate directly on fog/edge devices
- Manage the computation, network, and storage resources of the devices.
- System software needs to support multi-tenancy and isolation, because fog/edge computing accommodates several applications from different tenants.
- Two categories
  - System Virtualization
  - Network Virtualization

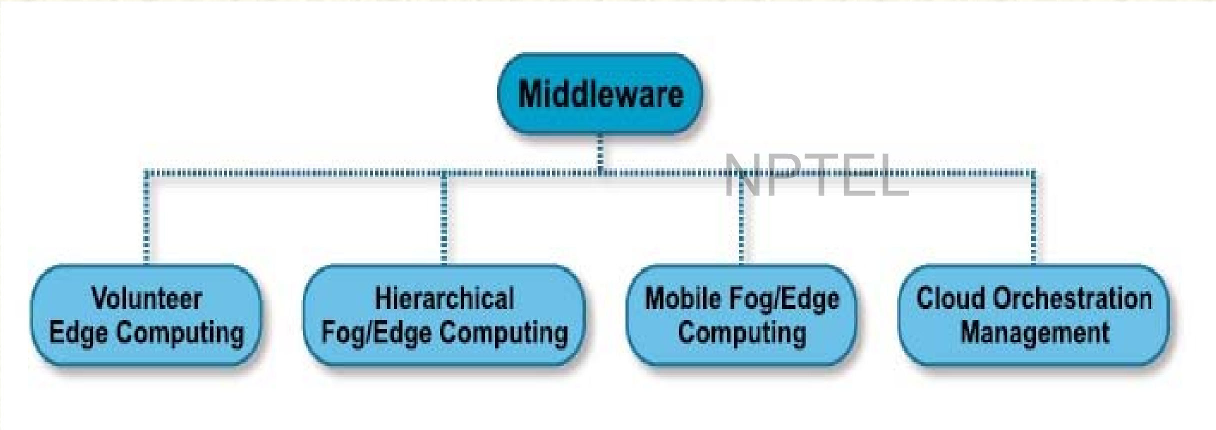
System Software (contd..)



Ref: Farah Aït Salaht, Frédéric Desprez, and Adrien Lebre. 2020. An Overview of Service Placement Problem in Fog and Edge Computing. ACM Comput. Surv. 53, 3, Article 65 (June 2020), 35 pages.

Middleware

- Middleware provides complementary services to system software.
- Middleware in fog/edge computing provides performance monitoring, coordination and orchestration, communication facilities, protocols etc.

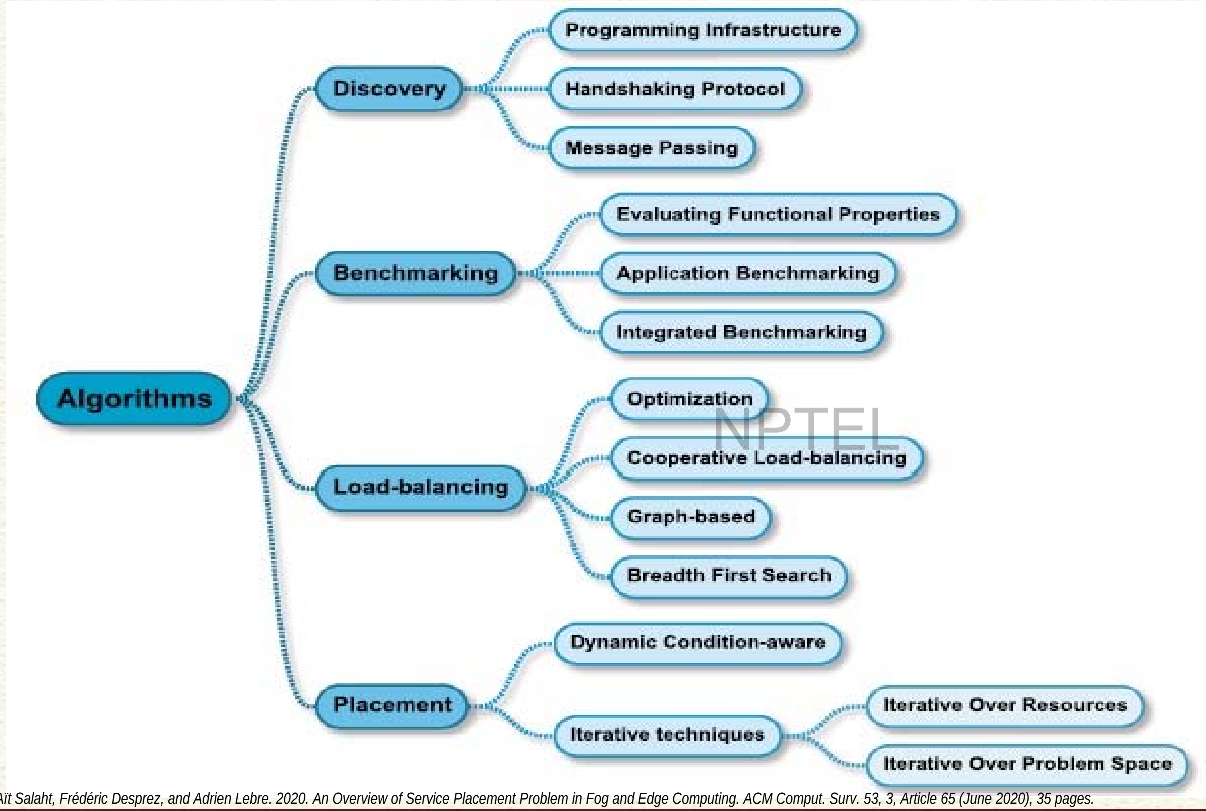


Ref: Farah Aït Salaht, Frédéric Desprez, and Adrien Lebre. 2020. An Overview of Service Placement Problem in Fog and Edge Computing. ACM Comput. Surv. 53, 3, Article 65 (June 2020), 35 pages.

ALGORITHMS

- Algorithms used to facilitate fog/edge computing. Four major algorithms.
- *Discovery*: identifying edge resources within the network that can be used for distributed computation
- *Benchmarking*: capturing the performance of resources for decision-making to maximize the performance of deployments
- *Load-balancing*: distributing workloads across resources based on different criteria such as priorities, fairness etc.
- *Placement*: identifying resources appropriate for deploying a workload.

ALGORITHMS (contd..)



Ref: Farah Aït Salaht, Frédéric Desprez, and Adrien Lebre. 2020. An Overview of Service Placement Problem in Fog and Edge Computing. ACM Comput. Surv. 53, 3, Article 65 (June 2020), 35 pages.

REFERENCES

- Cheol-Ho Hong, Blesson Varghese, Resource Management in Fog/Edge Computing: A Survey on Architectures, Infrastructure, and Algorithms, ACM Computing Surveys, Vol 52(5), October 2019, pp 1–37.
- Farah Aït Salaht, Frédéric Desprez, and Adrien Lebre. 2020. An Overview of Service Placement Problem in Fog and Edge Computing. ACM Comput. Surv. 53, 3, Article 65 (June 2020), 35 pages.
- Agarwal, S.; Yadav, S.; Yadav, A.K. An efficient architecture and algorithm for resource provisioning in fog computing. Int. J. Inf. Eng. Electronic Bus. (IJIEEB) 2016, 8, 48–61.

NPTEL

NPTEL ONLINE CERTIFICATION COURSES

Cloud Computing

Prof. Soumya K Ghosh
Department of Computer Science and Engineering

Module 09: Cloud Computing Paradigm
Lecture 44: Cloud Federation

CONCEPTS COVERED

- Cloud-Fog Paradigm – Resource Management Issues
- Service Placement Problem

KEYWORDS

- Cloud Computing
- Fog - Edge Computing
- Resource Management
- Service Placement

Cloud Federation

Cloud Federation?

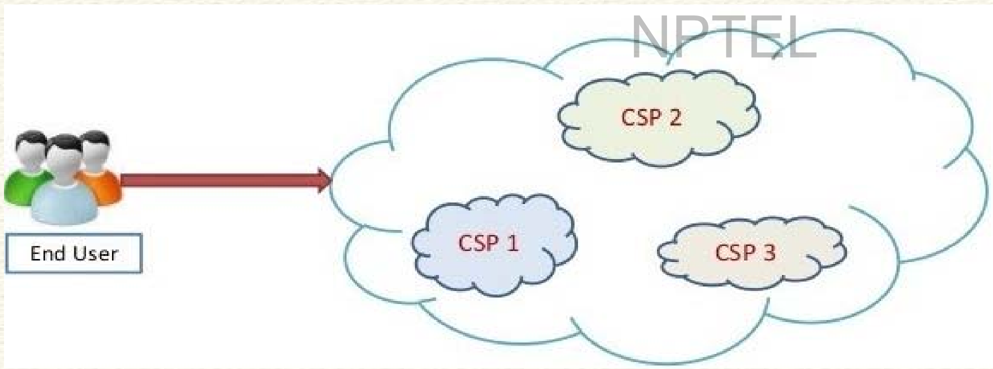
- A federated cloud (also called cloud federation) is the deployment and management of multiple external and internal cloud computing services to match business needs.
- A federation is the union of several smaller parts that perform a common action.

[*Ref: http://whatis.techtarget.com/definition/federated-cloud-cloud-federation*]

Cloud Federation?

Collaboration between Cloud Service Providers (CSPs) to achieve:

- Capacity utilization
- Inter-operability
- Catalog of services
- Insight about providers and SLA's



Federation - Motivation

- Different CSPs join together to form a federation
- Benefits:
  - Maximize resource utilization
  - Minimize power consumption
  - Load balancing
  - Global utility
  - Expand CSP's global foot prints

Federation - Characteristics

- To overcome the current limitations of cloud computing such as service interruptions, lack of interoperability and degradation of services.
- Many inter-cloud organizations have been proposed.
- Cloud federation is an example of an inter-cloud organization.

- ➢ It is a inter-cloud organization with voluntary characteristics.
- ➢ It should have maximum geographical separation.
- ➢ Well defined marketing system and regulated federal agreement.
- ➢ IT is an environment where multiple SP come together and share their resources.

Federation Architecture

- Cloud federation is associated with several portability and interoperability issues.
- Typical federation architectures: cloud bursting, brokering, aggregation, and multitier.
- These architectures can be classified according to the level of coupling or interoperation among the cloud instances involved, ranging from loosely coupled (with no or little interoperability among cloud instances) to tightly coupled (with full interoperability among cloud instances).

Loosely Coupled Federation

- Limited interoperation between CSPs / cloud instances.
  Example: a private cloud complementing its infrastructure with resources from an external commercial cloud

- A CSP has little or no control over remote resources (for example, decisions about VM placement are not allowed), monitoring information is limited (for example, only CPU, memory, or disk consumption of each VM is reported), and there is no support for advanced features such as cross-site networks or VM migration.

Partially Coupled Federation

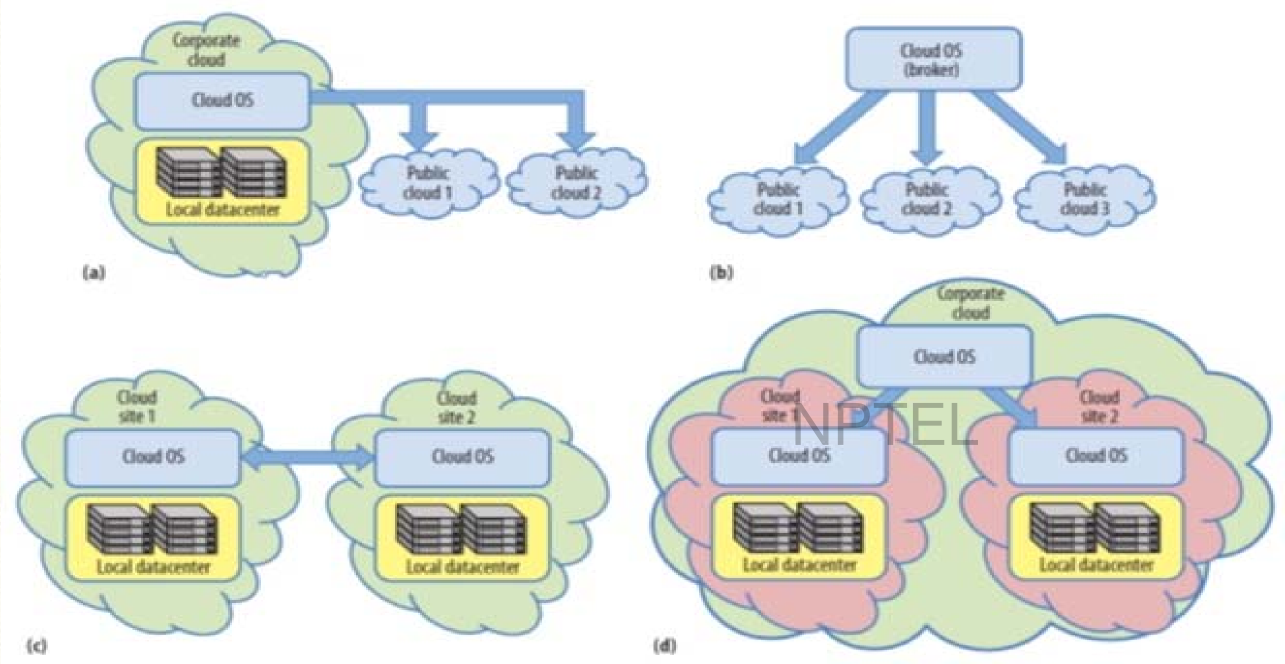
- Different CSPs (partner clouds) establish a contract or framework agreement stating the terms and conditions under which one partner cloud can use resources from another.

- This contract can enable a certain level of control over remote resources (for example, allowing the definition of affinity rules to force two or more remote VMs to be placed in the same physical cluster);
- May agree to the interchange of more detailed monitoring information (for example, providing information about the host where the VM is located, energy consumption, and so on);
- May enable some advanced networking features among partner clouds (for example, the creation of virtual networks across site boundaries).

Tightly Coupled Federation

- In this case the clouds are normally governed by the same cloud adminstration.
- A cloud instance can have advanced control over remote resources—for example, allowing decisions about the exact placement of a remote VM—and can access all the monitoring information available about remote resources.
- May allow other advanced features, including the creation of cross-site networks, cross-site migration of VMs, implementation of high availability techniques among remote cloud instances, and creation of virtual storage systems across site boundaries.

Cloud Federation Architectures



(a) Hybrid / Bursting, (b) Broker, (c) Aggregated, (d) Multiplier

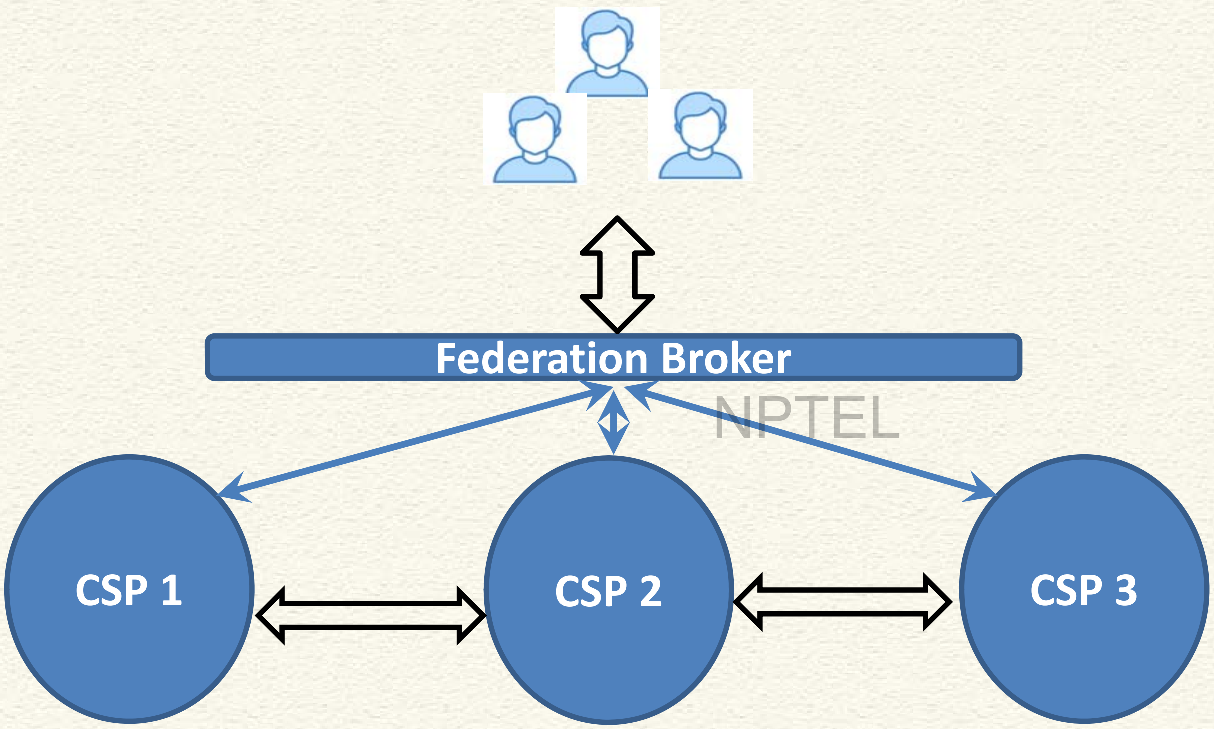
Hybrid / Bursting Architecture

- Cloud bursting or hybrid architecture combines the existing on-premise infrastructure (usually a private cloud) with remote resources from one or more public clouds to provide extra capacity to satisfy peak demand periods.
- As the local cloud OS has no advanced control over the virtual resources deployed in external clouds beyond the basic operations the providers allow, this architecture is loosely coupled.
- Most existing open cloud managers support the hybrid cloud architecture

Broker Architecture

- A broker that serves various users and has access to several public cloud infrastructures. A simple broker should be able to deploy virtual resources in the cloud as selected by the user.
- Brokering is the most common federation scenario.
- An advanced broker offering service management capabilities could make scheduling decisions based on optimization criteria such as cost, performance, or energy consumption to automatically deploy virtual user service in the most suitable cloud
- It may even distribute the service components across multiple clouds. This architecture is also loosely coupled since public clouds typically do not allow advanced control over the deployed virtual resources.

Broker Architecture



Aggregated Architecture

- Involves two or more partner clouds that interoperate to aggregate their resources and provide users with a larger virtual infrastructure.
- This architecture is usually partially coupled, since partners could be provided with some kind of advanced control over remote resources, depending on the terms and conditions of contracts with other partners.
- The partner clouds usually have a higher coupling level when they belong to the same corporation than when they are owned by different companies that agree to cooperate and aggregate their resources.

Multitier Architecture

- Involves two or more cloud sites, each running its own cloud OS and usually belonging to the same corporation, that are managed by a third cloud OS instance following a hierarchical arrangement.
- This root/top cloud OS instance has full control over resources in different cloud sites—a tightly coupled scenario—and it exposes the resources available in the different cloud sites as if they were located in a single cloud.
- This architecture is beneficial for corporations with geographically distributed cloud infrastructures because it provides uniform access.
- It may be useful for implementing advanced management features such as high availability, load balancing, and fault tolerance.

REFERENCES

- Moreno-Vozmediano, R., Montero, R., and Llorente, I., "IaaS Cloud Architecture: From Virtualized Data Centers to Federated Cloud Infrastructures", IEEE Computer Vol. 45 (12), Dec. 2012, pp. 65-72.
- Sanjay P. Ahuja, IaaS Cloud Architectures: Virtualized Data Centers to Federated Cloud Infrastructures, School of Computing, UNF

Thank you

NPTEL