Fog Computing & Data Management

Amir Masoud Rahmani

May 2018

Content

1. Introduction and basic concepts
2. Data Management
3. Case Study
4. Conclusion

1

Introduction & Basic Concept

Introduction

- What is “Fog Computing”?
- Fog computing is a new computational paradigm, which extends the traditional cloud computing and services to the edge of network. It provides the computation, communication, controlling, storage and services capabilities at the edge of network.

Fog Node vs Cloud

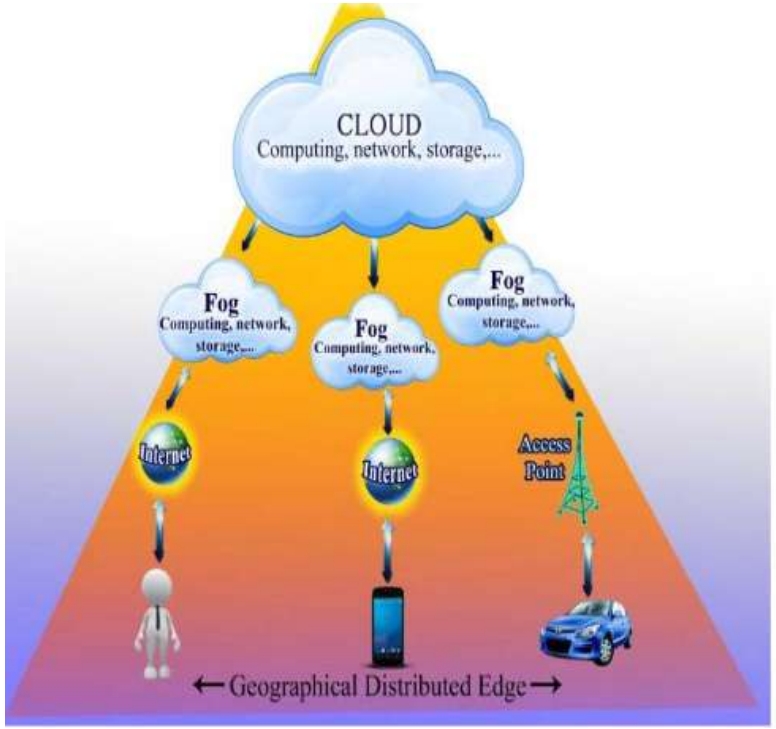
Fog nodes:

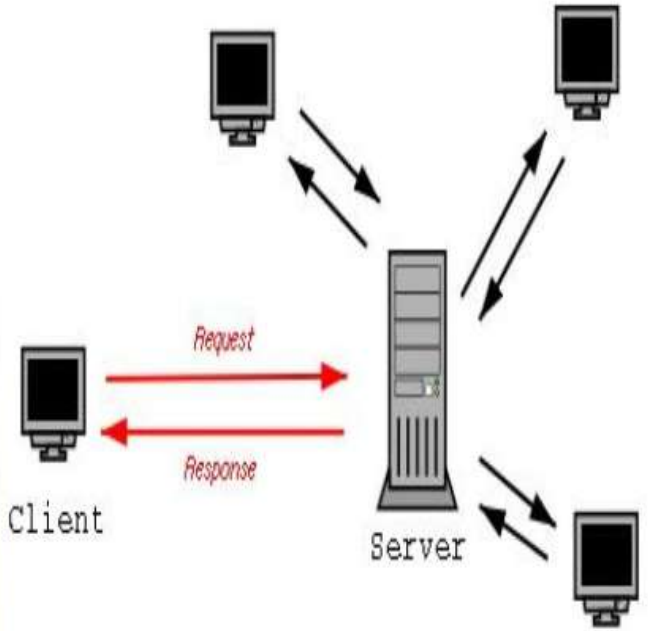
- Receive feeds from IoT devices using any protocol, in real time
- Run IoT-enabled applications for real-time control and analytics, with millisecond response time
- Provide transient storage, often 1–2 hours
- Send periodic data summaries to the cloud

The Cloud platform:

- Receives and aggregates data summaries from many fog nodes
- Performs analysis on the IoT data and data from other sources to gain business insight
- Can send new application rules to the fog nodes based on these insights

Fog Computing Vs. Client-Server





Compare & Contrast

equirements	Cloud Computing	Fog Computing
tency	High	Low
elay Jitter	High	Very low
cation of Servers	Within Internet	At the edge close to Nodes
stance between e client & server	Multiple hops	One hop
curity	Varies amongst providers	Can be more defined and customized
tack on Data-in-ght	High probability	Limited with less probability
cation awareness	No	Yes

Compare & Contrast

equirements	Cloud Computing	Fog Computing
eo. Distribution	Centralized	Distributed
o. of Server Noes	Few	Very large
ıpport for Mobility	Limited	Supported
eal Time teractions	Supported but may be difficult to achieve & Costly	Supported
ype of last mile onnectivity	Leased line	wireless

Characteristics of Fog Computing

- A paradigm that extends Cloud computing to the edge of the network
- Low latency & location awareness
- Send the right data to the cloud for big data analytics and storage
- Wide-spread geographical distribution
- Strong presence of streaming and real-time applications
- Handle an unprecedented volume, variety, and velocity of data Heterogeneity of connected objects
- Fog applications to communicate directly with mobile
- Devices Predominant role of wireless access

Benefits of Fog Computing

- The obvious benefits of using fog computing includes minimizing application latency and improving the Quality-of-Service (QoS) and Experience (QoE) for users while leveraging hierarchical networking and tapping into resources that are traditionally not employed for general purpose computing.

Benefits of Fog Computing

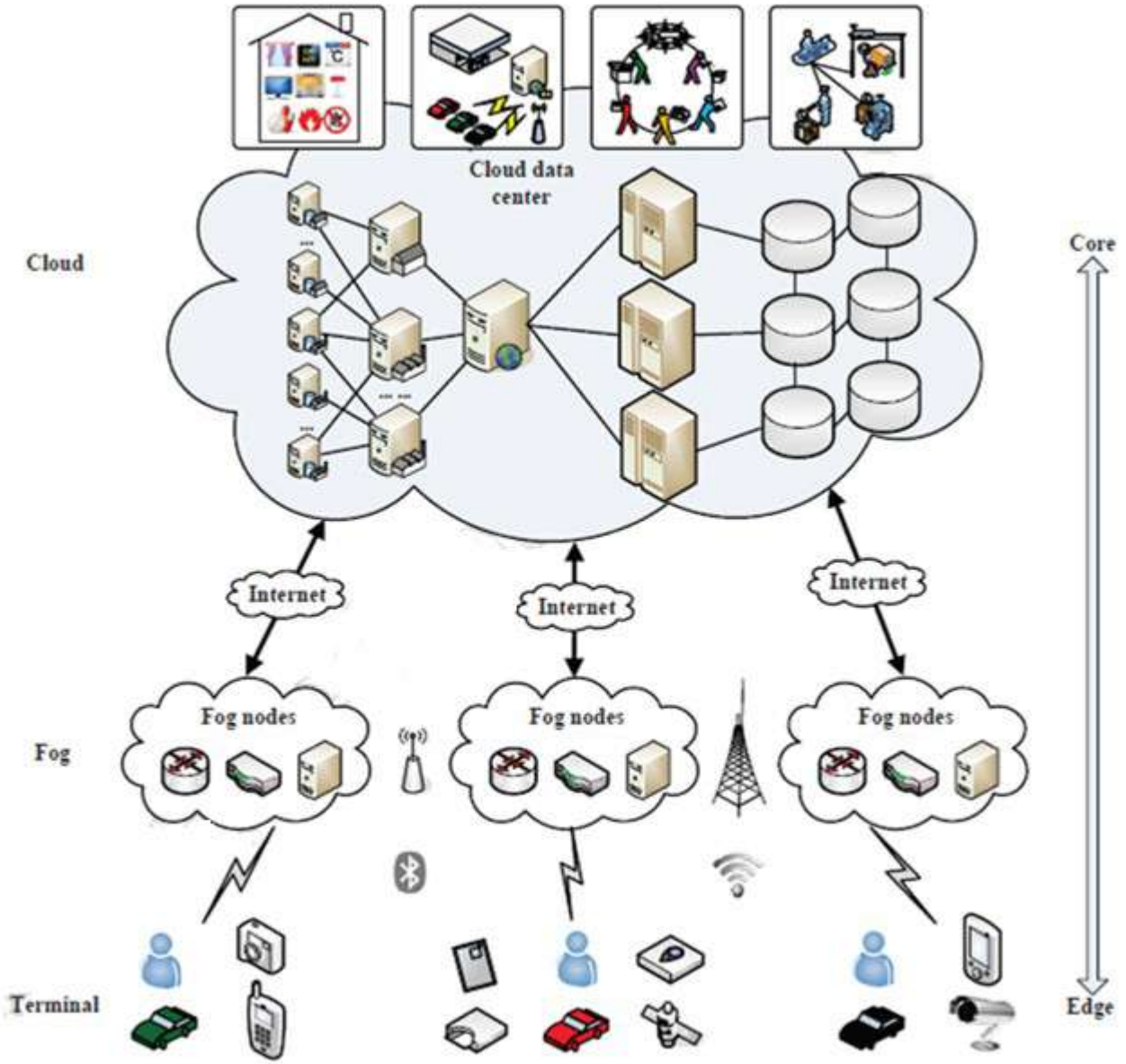
- **Greater business agility**: With the right tools, developers can quickly develop fog applications and deploy them where needed. Machine manufacturers can offer MaaS to their customers. Fog applications program the machine to operate in the way each customer needs.
- **Better security**: Protect your fog nodes using the same policy, controls, and procedures you use in other parts of your IT environment. Use the same physical security and cybersecurity solutions.
- **Deeper insights, with privacy control**: Analyze sensitive data locally instead of sending it to the cloud for analysis. Your IT team can monitor and control the devices that collect, analyze, and store data.
- **Lower operating expense**: Conserve network bandwidth by processing selected data locally instead of Sending it to the cloud for analysis.

Fog computing architecture

Fog Nodes Extend the Cloud to the Network Edge.

	Fog Nodes Closest to IoT Devices	Fog Aggregation Nodes	Cloud
onse time	Milliseconds to subsecond	Seconds to minutes	Minutes, days, weeks
ication nples	M2M communication Haptics[2], including telemedicine and training	Visualization Simple analytics	Big data analytics Graphical dashboards
long IoT data ored	Transient	Short duration: perhaps hours, days, or weeks	Months or years

Fog computing architecture



1st Layer

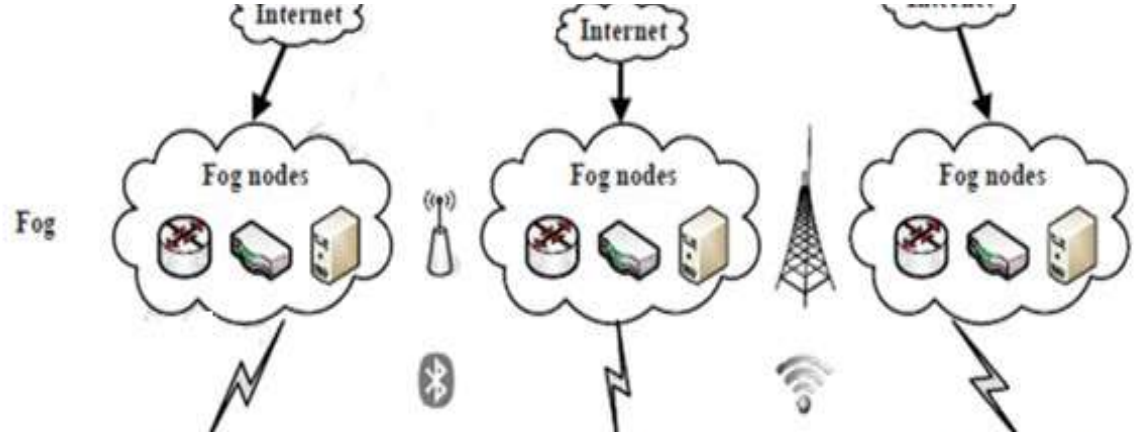
Terminal layer

- This is the layer closest to the end user and physical environment. It consists of various IoT devices, for example, sensors,mobile phones, smart vehicles, smart cards, readers, and so on.

2nd Layer

Fog layer

- This layer is located on the edge of the network. Fog computing layer is composed of a large number of fog nodes, which generally including routers, gateways, switchers, access points, base stations, specific fog servers, etc.
- These fog nodes are widely distributed between the end devices and cloud



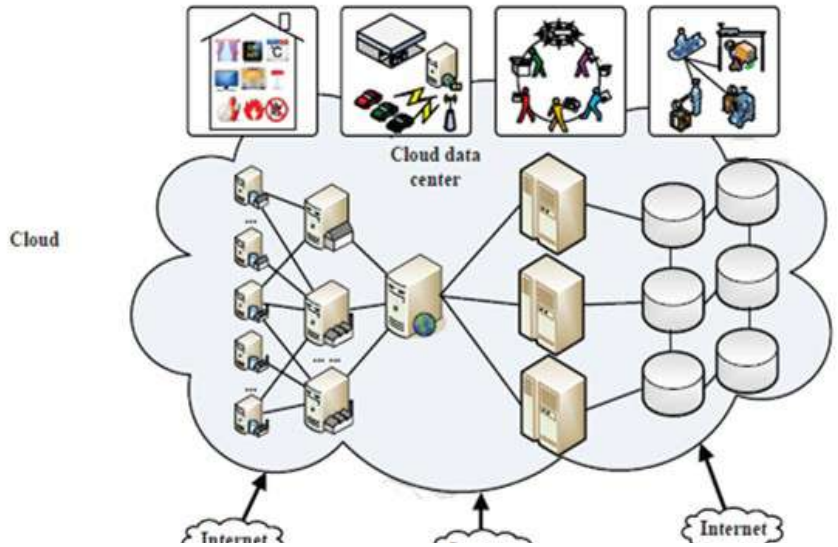
Fog Node

- Heterogeneous devices such as Cisco IOx networking equipment, micro-datacenter, Nano-server, smart phone, personal computer and Cloudlets, commonly known as Fog node.
- Fog nodes create a wide distribution of services to process IoT-data closer to the source.
- Any device with computing, storage, and network connectivity can be a fog node. (such as industrial controllers, switches, routers, embedded servers, cameras, and etc.)

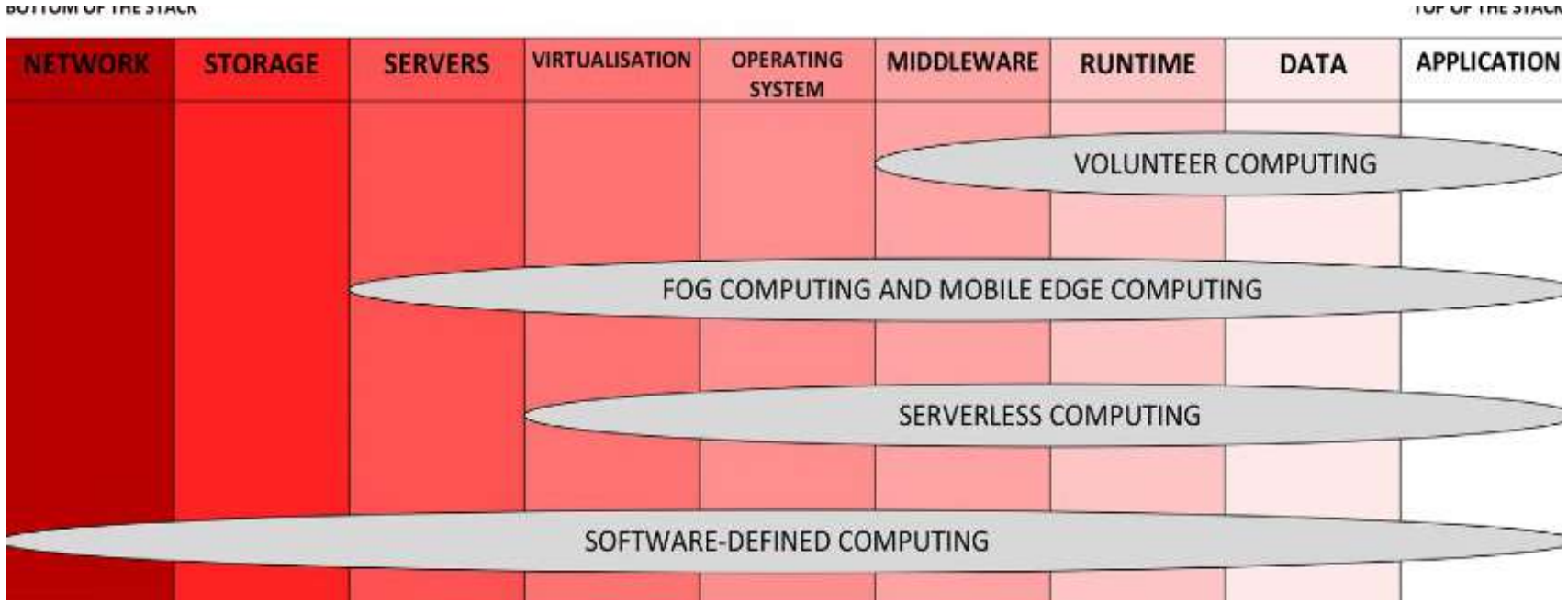
3rd Layer

Cloud layer

- The cloud computing layer consists of multiple **high-performance servers** and **storage** devices
- This layer provides various application services, such as smart home, smart transportation, smart factory, etc. It has powerful computing and storage capabilities to support for extensive computation analysis and permanently storage of an enormous amount of data.



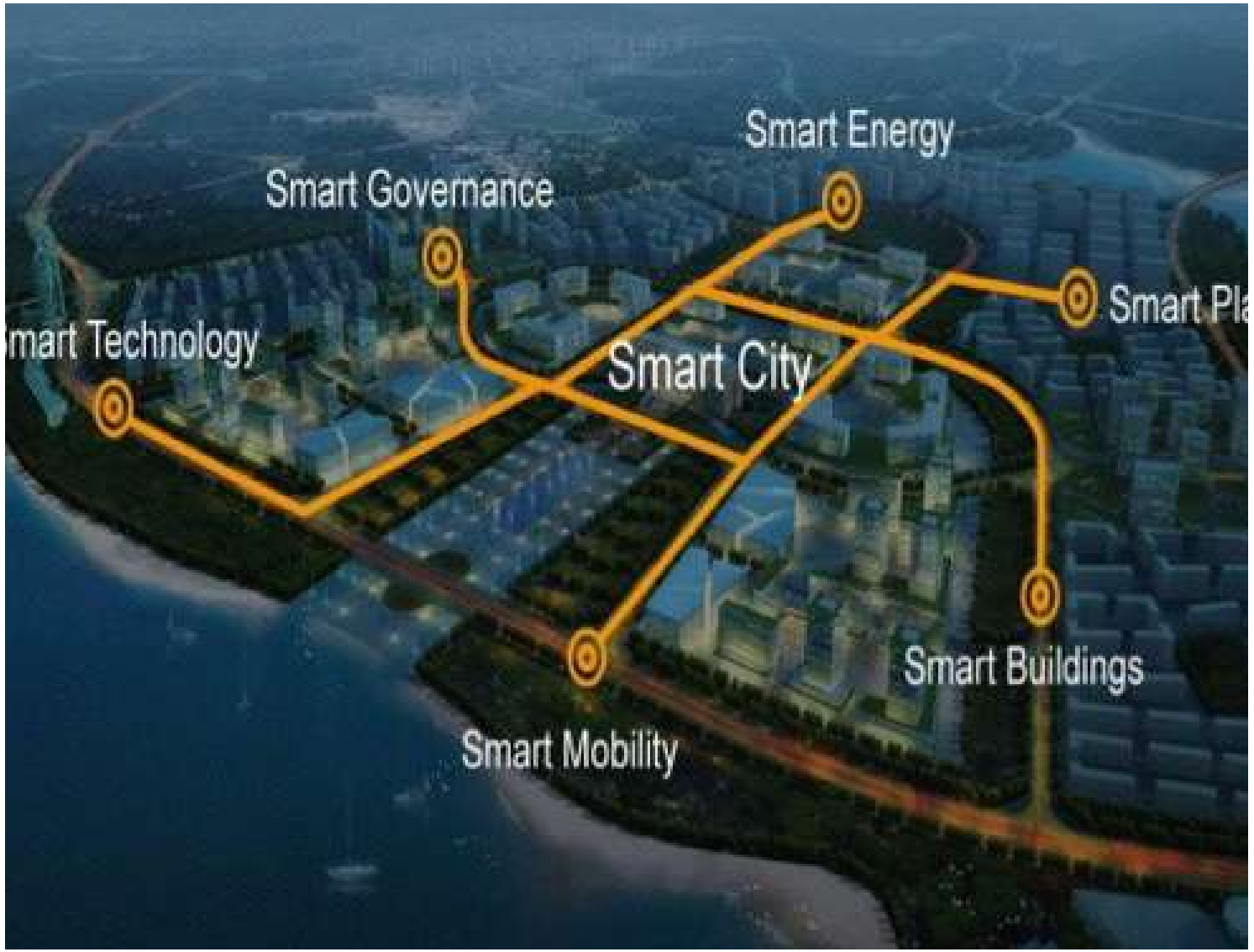
Layers of abstraction



Characteristics of fog computing

- Edge location, location awareness, and low latency : Fog computing support endpoints with finest services at the edge of the network.
- Geographical distribution : The services and application objective of the fog is widely distributed.
- Support for mobility : Using LISP protocol fog devices provide mobility techniques like decouple host identity to location identity.
- Real time interactions : fog computing requires real time interactions for speedy service.
- Heterogeneity : Fog nodes can be deployed in a wide variety of environments.
- Interoperability : Fog components must be able to interoperate in order to give wide range of services like streaming.

Fog computing application



Fog computing application

- Healthcare and activity tracking
- Smart utility services
- Smart transportation
- Smart Building
- Smart City
- Augmented reality, cognitive systems, and gaming

Fog computing application

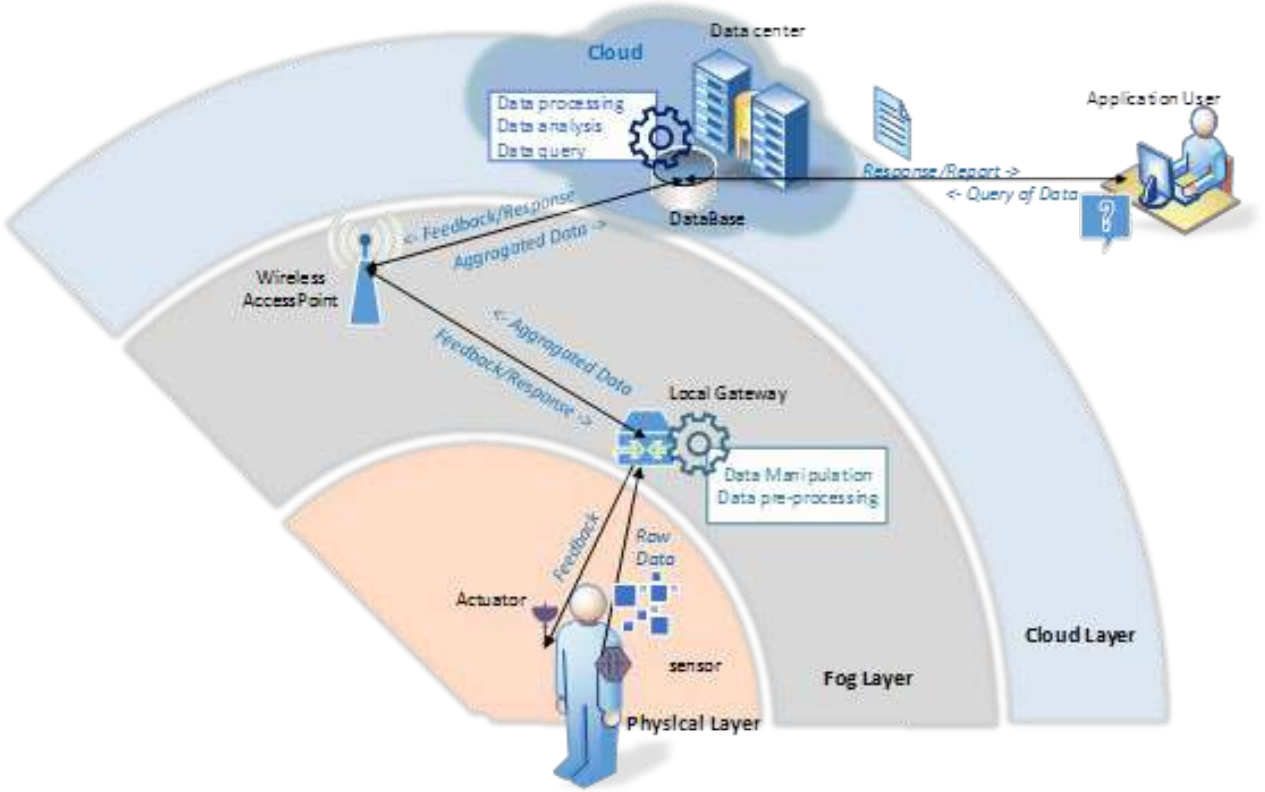
- Healthcare and activity tracking
- Smart utility services
- Augmented reality, cognitive systems, and gaming

2

Fog Computing & Data Management

Fog Computing & Data Management

- Fog plays as a mediator between devices and cloud
- Fog is responsible for temporary data storage, some preliminary processing, and analytics.
- Fog does some preliminary process and may store data for a while

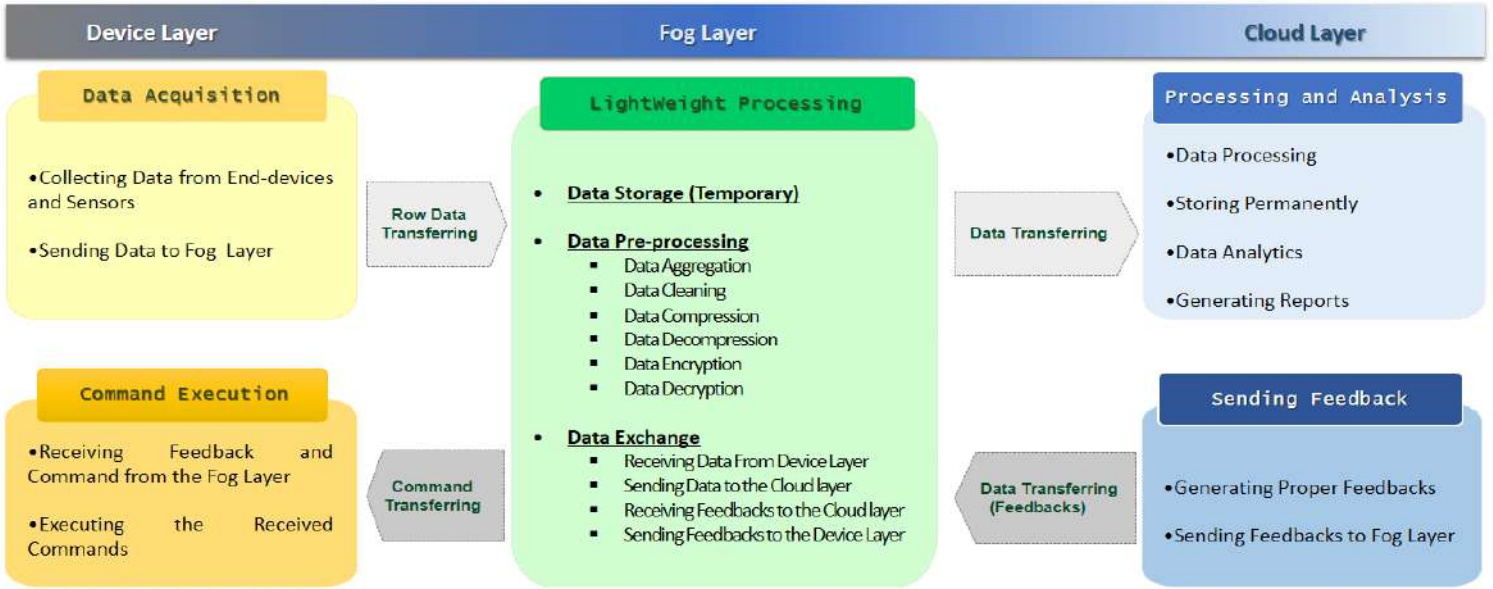


Advantages of Fog data management

- main advantages of data management in fog computing
- **Increasing efficiency**
- **Increasing the level of privacy**
- **Increasing data quality**
- **Decreasing the end-to-end latency**
- **Increasing dependability**
- **Decreasing cost**

Data Life Cycle

1. Data acquisition
2. Lightweight processing
3. Processing and analysis
4. Sending feedback
5. Command execution



Step1

Data acquisition

- Data from different types of end-devices should be acquired.
- Data must be sent to upper layers

Step2

Lightweight processing

- Lightweight data manipulation and local data processing on the collected data.
- Lightweight processing :
  - data aggregation
  - data filtering and elimination of unnecessary or repetitive data, data cleaning
  - compression/decompression
  - some lightweight data analysis
  - pattern extraction.

Step4

Processing and analysis

- Received data may be stored permanently in the cloud layer, and are processed based on predefined requirements .
- Different types of analysis may have been required.
- These types of analysis almost are in the scale of big data and need big data platforms and technologies.

Step5

Sending feedback

- Based on previous step, proper feedbacks such as apporporiate commands or decisions are generated and sent to the fog layer.

Step6

Command execution

- Actuators must run the proper action based on the received data. By this way, proper feedback and responses are returned to the environment.

Data Characteristics

- Heterogeneity
- Inaccuracy
- Weak semantics
- Velocity
- Redundancy
- Scalability
- Inconsistency

Heterogeneity

- Distributed heterogeneous end-devices generate data with different formats, generated data may be diversely varying in terms of structure or for-mat

Inaccuracy

- Inaccuracy or uncertainties of the sensed data refer to the sensing precision, accuracy or misreading of data

Weak semantics

- The collected raw data that may be heterogeneous in terms of **data formats, data structure, data source**, etc. must be processed and managed.
- By using the concepts of semantic web and injecting some information and extra data to the raw data, make it readable and understandable for machines.
- Nevertheless, most of the collected data from the environment has weak semantics.

Velocity

- Data generation rates and sampling frequencies are varying in different types of end-devices

Redundancy

- Repetitive data that is sent by one or more end-devices leads to redundancy in the collected data

Scalability

- Large number of end-devices and high data sampling rate that may exist in different scenarios may lead to generate a huge amount of data

Inconsistency

- Low precision or misreading in the sensed data may cause in-consistency in the gathered data

Data Pre-processing and analytics

- Data clearing
  - Declarative data cleaning
  - Model-based data cleaning
- Data fusion
- Edge mining

Dirty Data

- Sensory data are not fully reliable, which is unpleasant for further processing and decision-making.
- “Dirty data” refer to missed readings and unreliable readings.

Data Cleaning

- Cleaning mechanism can be applied on the collected data in fog layer to reduce the effect of dirty and unreliable data, and increase the quality of them.

- Cleaning Approaches:

1. declarative data cleaning
2. model-based data cleaning

Declarative data cleaning

- High-level declarative queries such as CQL (continuous query language) are used to define the sensor values constraints.
- user can express the queries and control the system easily via the provided interface.
- Example: Extensible Sensor stream Processing (ESP)

- ESP is a declarative based and pipelined-framework for sensor data cleaning for use in pervasive applications.

Model-based data cleaning

- Anomalies are detected by comparison of raw values with the inferred values that are resulted as the most probable values based on selected models.
- The model-based approaches:

1. Regression models
   - Polynomial Regression
   - Chebyshev regression
2. proba-bilistic models
   - Kalman filter
   - outliner detection models

Data Fusion

- Data fusion refers to the **elimination of redundant and ambiguous data** and **integration of data** and can be done in the **fog layer** as one of the data management tasks to in-crease the accuracy and efficiency.

Data Fusion Models

- Data fusion models can be categorized in to three particular categories

1. data-based model
2. activity-based model
3. role-based model

Edge mining

- Edge mining refers to utilize mining approaches on row data which are produced by devices in the edge of the network (fog layer), by this way size of transferred data will be reduced and better energy saving can be achieved.

Data privacy

- Privacy preserving and **protection of data against unauthorized access** are considered as one of the fog computing functionalities to keeps malicious and unauthorized end-devices out of the system.

Case Study

e-Health

- Health-care applications such as ECG device may generate several GBs of a data in a day, transferring and processing it means that it conserves network bandwidth, storage, and processing cycles.
- In emergency conditions, fog computing performs faster rather than in the way that all processes are performed on the cloud layer.
- In comparison with nursing care, e-health application can monitor and control in 24/7 and low cost.

Case Study

e-Health

CHALLENGES

- Enabling real-time analytics
- Programming models and architectures
- Power consumption
- Security, reliability, and fault tolerance

Conclusion

- To decrease the response time of real-time systems for IoT applications and handle the huge amount of data in IoT systems, fog computing paradigm can be considered a good solution .

Question & Answer