MPSSDEGB
कौशल से रोजगार, समृद्धि अपार

FITT
Foundation For Innovation And Technology Transfer

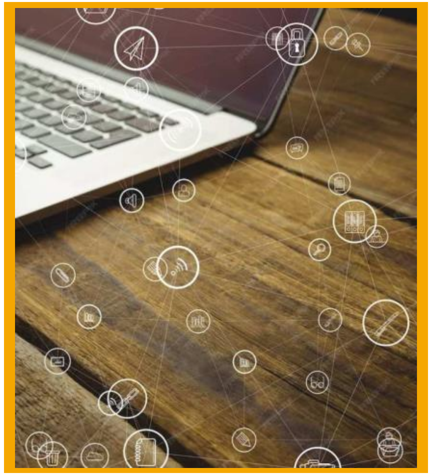
SANKALP
Ministry of Skill Development
& Entrepreneurship

IoT Communication Protocols

Objectives:

1. Overview of IoT Communication Protocols
2. Types of IoT Communication Protocols(Based on Range and Layer)
3. Learn About Constrained Devices and Network
4. Some widely used Application Layer protocols :MQTT, CoAP
5. Comparison of MQTT and CoAP Protocols
6. Future Challenges in developing IoT Communication Protocols

Overview of IoT Communication Protocols



Iot Communication:

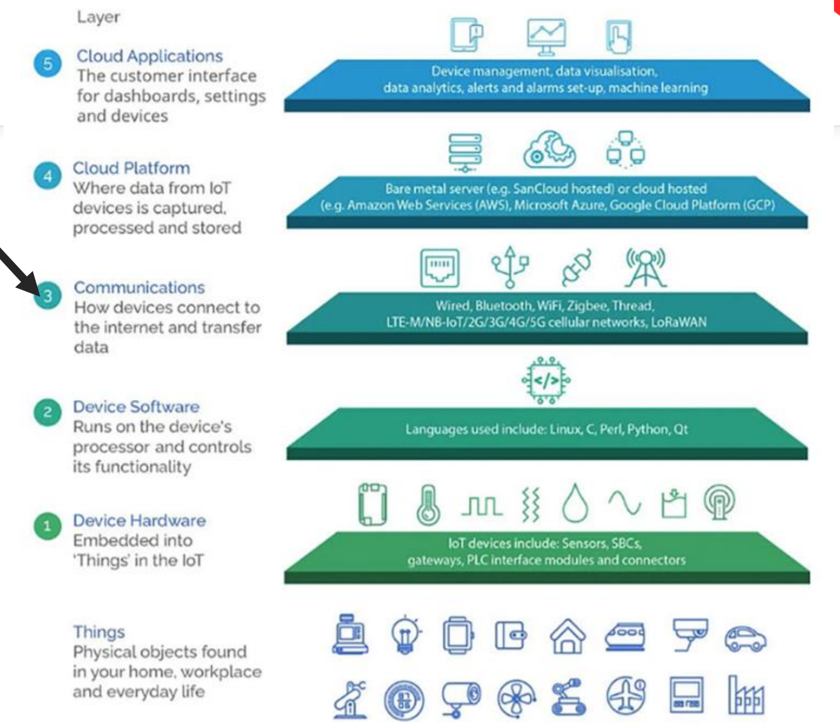
IoT **communication** refers to the exchange of data between various devices and systems in the context of the Internet of Things. It establishes a network where devices can send, receive, and process data.

IoT communication **involves** understanding the fundamental concepts and principles that govern the exchange of data between interconnected devices (sensors, actuators, industrial machinery etc) and also the systems (like servers, clouds, etc) in the Internet of Things (IoT).

IoT devices communicate using standardized communication protocols. These **protocols** define the rules and conventions for data exchange. Common IoT protocols include MQTT, CoAP, HTTP, and others.

Why IoT Communication Protocol?

- IoT communication protocols are necessary to standardize and optimize the exchange of data among diverse devices, ensuring interoperability, efficiency, and security in the Internet of Things ecosystem. These protocols simplify development, enhance scalability, and address resource constraints, facilitating seamless and reliable communication between IoT devices.



Layer

5 Cloud Applications
The customer interface for dashboards, settings and devices

Device management, data visualisation, data analytics, alerts and alarms set-up, machine learning

4 Cloud Platform
Where data from IoT devices is captured, processed and stored

Bare metal server (e.g. SanCloud hosted) or cloud hosted (e.g. Amazon Web Services (AWS), Microsoft Azure, Google Cloud Platform (GCP)

3 Communications
How devices connect to the internet and transfer data

Wired, Bluetooth, WiFi, Zigbee, Thread, LTE-M/NB-IoT/2G/3G/4G/5G cellular networks, LoRaWAN

2 Device Software
Runs on the device's processor and controls its functionality

Languages used include: Linux, C, Perl, Python, Qt

1 Device Hardware
Embedded into 'Things' in the IoT

IoT devices include: Sensors, SBCs, gateways, PLC interface modules and connectors

Things
Physical objects found in your home, workplace and everyday life

Source: https://iotbusinessnews.com/WordPress/wp-content/uploads/2022/07/IoT-technology-stack-diagram.jpg

The Role of Protocols in IoT:

GIF Reference : https://www.einfochips.com/blog/hardware-design-challenges-of-the-embedded-internet-of-things-iot/

Unified Communication:

Ensuring devices from different manufacturers can work together seamlessly.

Standardization:

Implementing common protocols for consistent data exchange.

Seamless Integration:

Enabling interconnected systems to function harmoniously.

Types of IoT Communication Protocols

In the world of IoT, communication protocols are essential for enabling devices to connect and exchange data. Understanding the types of IoT communication protocols is crucial for determining the best approach for different applications.

They are

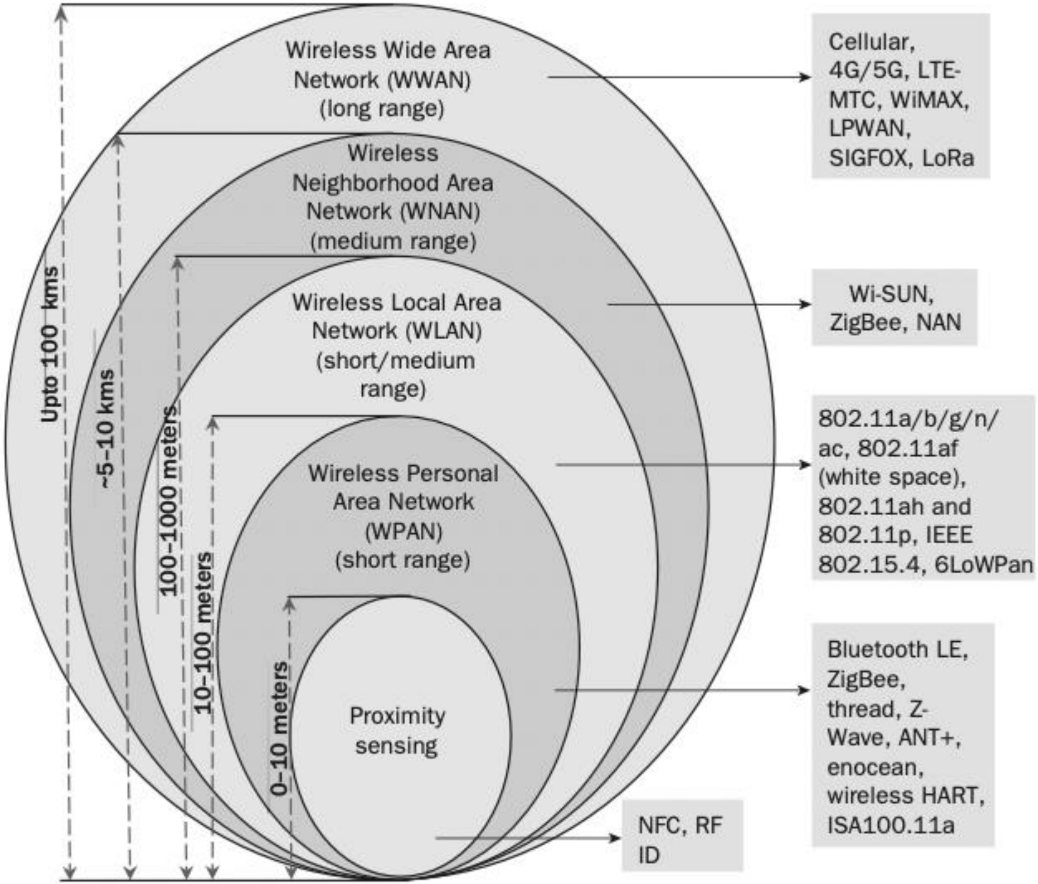
Categorised based

on

Range........

Layer ....

Various range Connectivity in IoT



Wireless Wide Area Network (WWAN) (long range)

Wireless Neighborhood Area Network (WNAN) (medium range)

Wireless Local Area Network (WLAN) (short/medium range)

Wireless Personal Area Network (WPAN) (short range)

Proximity sensing

Upto 100 kms

~5–10 kms

100–1000 meters

10–100 meters

0–10 meters

Cellular, 4G/5G, LTE-MTC, WiMAX, LPWAN, SIGFOX, LoRa

Wi-SUN, ZigBee, NAN

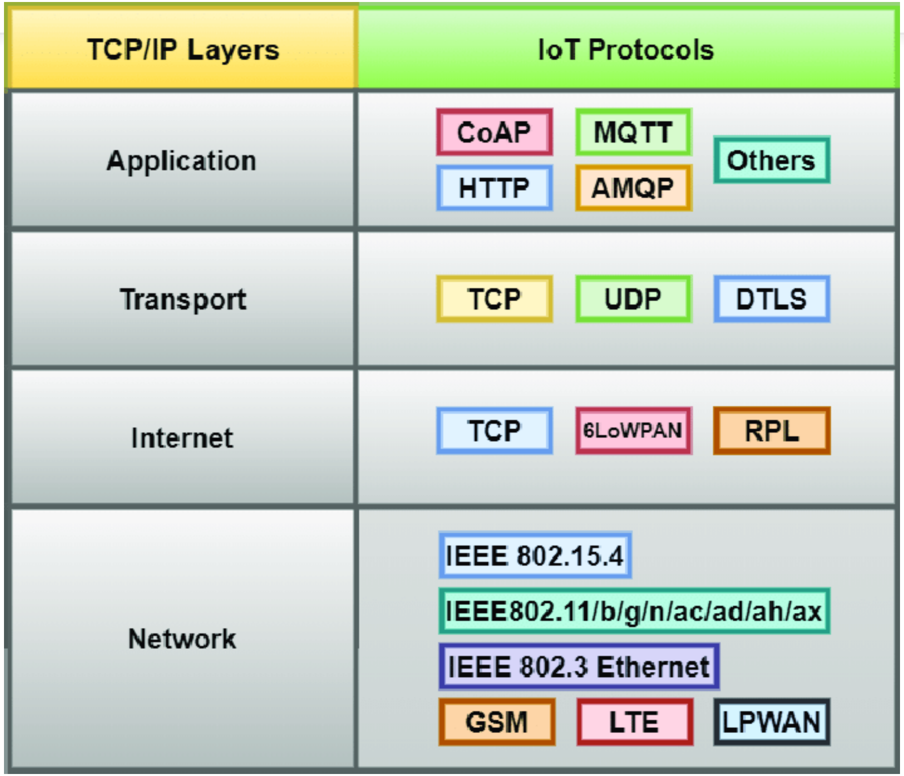
802.11a/b/g/n/ac, 802.11af (white space), 802.11ah and 802.11p, IEEE 802.15.4, 6LoWPan

Bluetooth LE, ZigBee, thread, Z-Wave, ANT+, enocean, wireless HART, ISA100.11a

NFC, RF ID

Source: [Internet of Things, Durbha et.al.](#)

Layer-wise Protocols

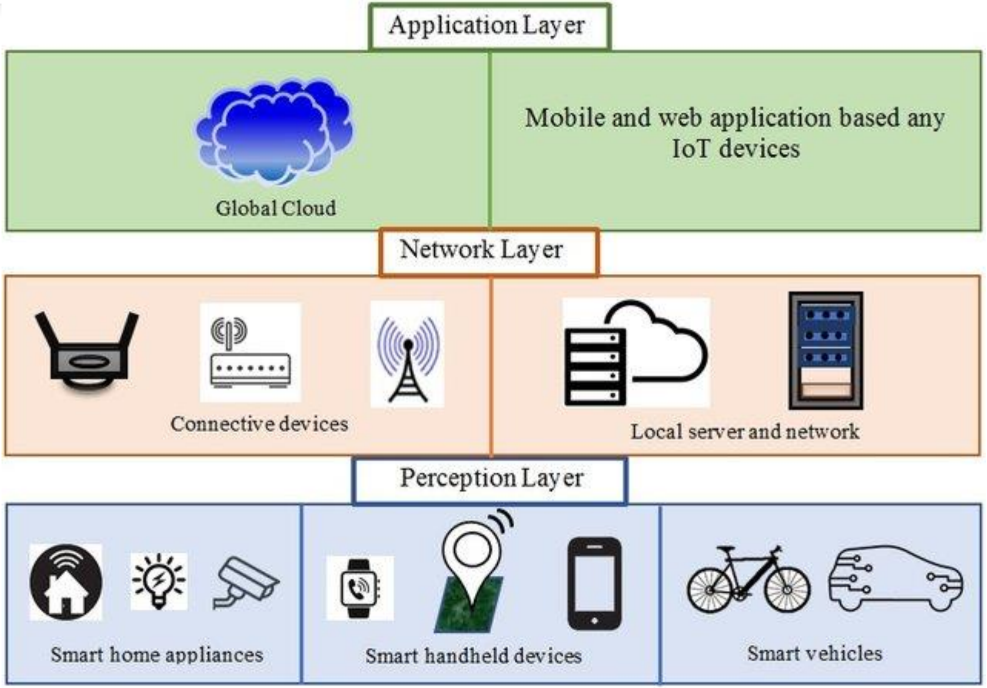


TCP/IP Layers	IoT Protocols
Application	CoAP MQTT HTTP AMQP Others
Transport	TCP UDP DTLS
Internet	TCP 6LoWPAN RPL
Network	IEEE 802.15.4 IEEE802.11/b/g/n/ac/ad/ah/ax IEEE 802.3 Ethernet GSM LTE LPWAN

Fig Reference: https://www.researchgate.net/figure/TCP-IP-Model-Main-IoT-Protocols_fig3_346414150

Application Layer in IoT.

In IoT communication, the application layer refers to the top layer of the communication stack, where the specific functionalities and services for IoT applications are defined. The application layer plays a crucial role in facilitating communication between IoT devices and applications, allowing them to exchange meaningful data and commands



https://www.researchgate.net/publication/340497604_Machine_learning_based_solutions_for_security_of_Internet_of_Things_IoT_A_survey/figures?lo=1&utm_source=google&utm_medium=organic

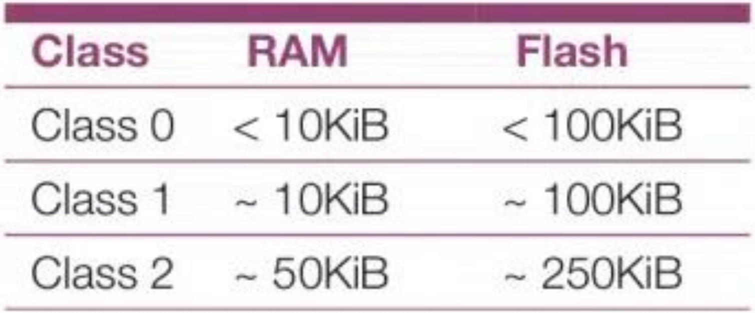
Resource Constraints:

Data protocols are used in resource constrained situations.

Resource constraints can occur in ,

1. IoT Nodes
2. IoT Network

Resource constrained nodes



Class	RAM	Flash
Class 0	< 10KiB	< 100KiB
Class 1	~ 10KiB	~ 100KiB
Class 2	~ 50KiB	~ 250KiB

Resource Constrained Network:

A constrained network exhibits below characteristics:

- Low bit-rate/throughput
- High packet loss and high variability of packet loss
- Highly asymmetric link characteristics
- Lack of advanced network services like multicast
- Power Supply and Processing limitations

The possible reasons for the characteristics of constrained network are below:

- Cost
- Physical constraints, e.g., power constraints or environmental constraints

Techniques to Conserve Power in Constrained Network:

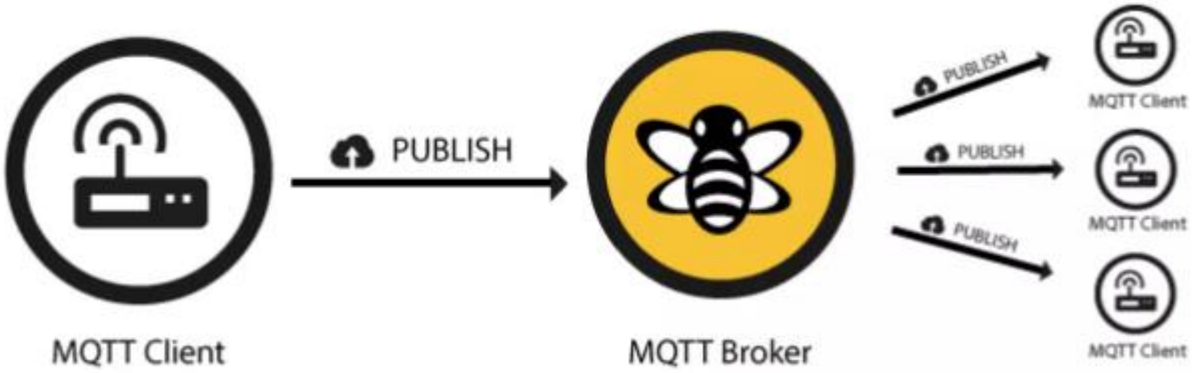
- Communication and mediation schemes that allow nodes to sleep
- Smaller (compressed) headers
- Smaller payloads/packets to keep bit error rates low and permit media sharing
- Limited bandwidth
- Nodes with different capability, some with reduced functionality and dependence on more powerful peers

MQTT Protocol

MQTT Protocol:

MQTT, which stands for Message Queuing Telemetry Transport, is a key protocol IoT. It operates on a publish-subscribe model. This lightweight and efficient protocol is widely utilized in IoT for constrained network applications.Works on top of the TCP protocol stack.

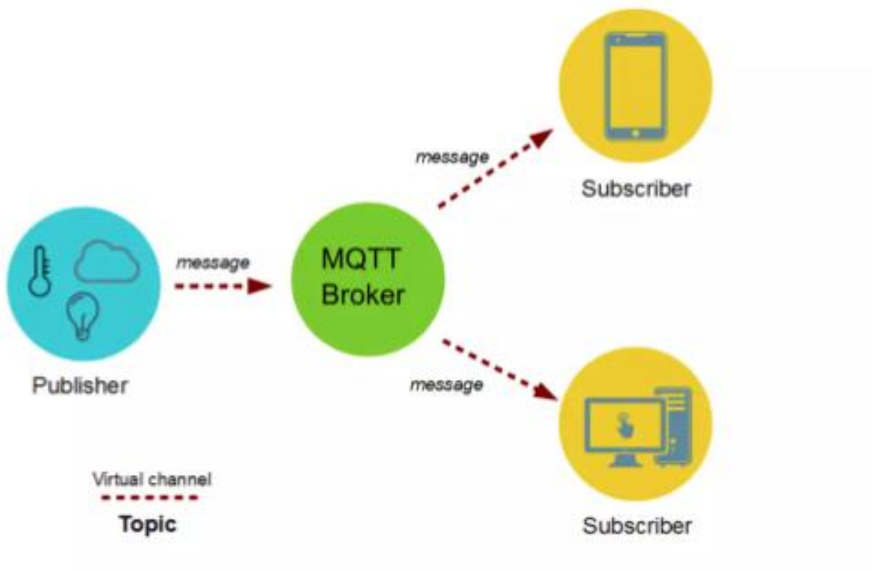
Publish-Subscribe model of MQTT:



https://www.slideshare.net/PremSanil/mqtt-iot-protocol-introduction

Publisher:

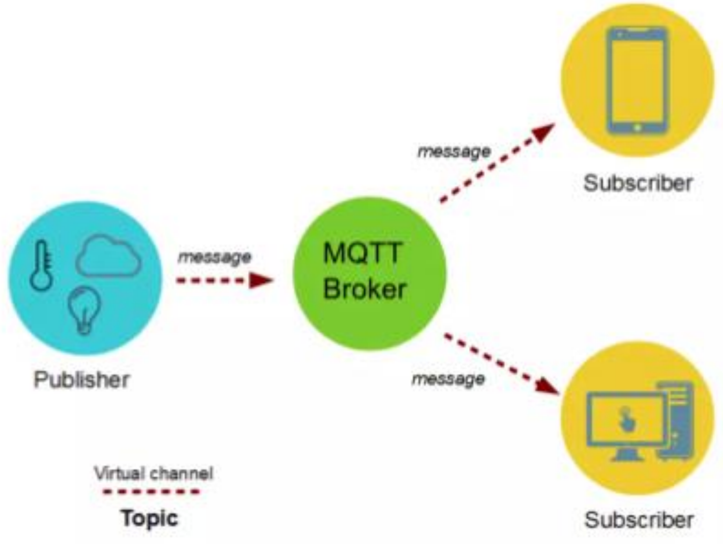
- System or sensors which collects data and sends it to broker which further sends it to multiple subscribers.
- Example motion sensor, water level Sensor, etc.
- Publisher publish data in following format:
  - Binary
  - JSON
  - SDC Record
  - Text



https://www.slideshare.net/vgitcsofficialedupoi/mqtt-and-coap

Subscriber:

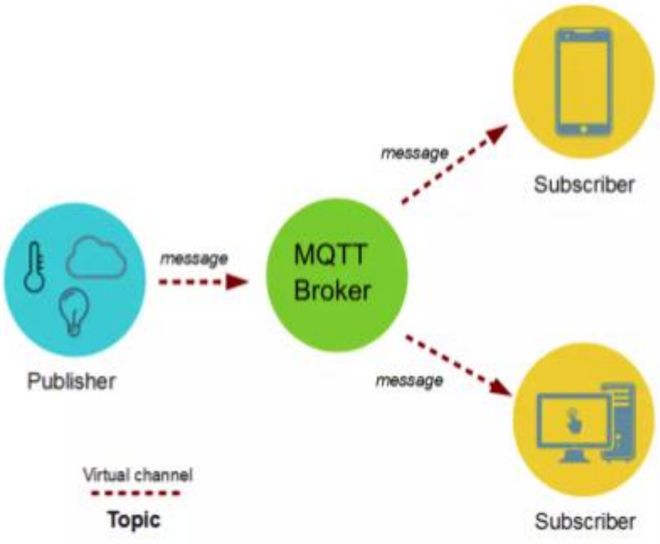
- Subscriber can be a mobile device, data server, monitoring stations, etc. which receives publish data from broker so that it can act according to it or monitor it or store it.
- Subscriber send request to broker to send required publisher's data.
- Broker has the table in which it maintains all subscriptions request, and send publish data according to it.
- Example Mobile devices, Monitoring system, etc.



https://www.slideshare.net/vgitcsofficialedupoi/mqtt-and-coap

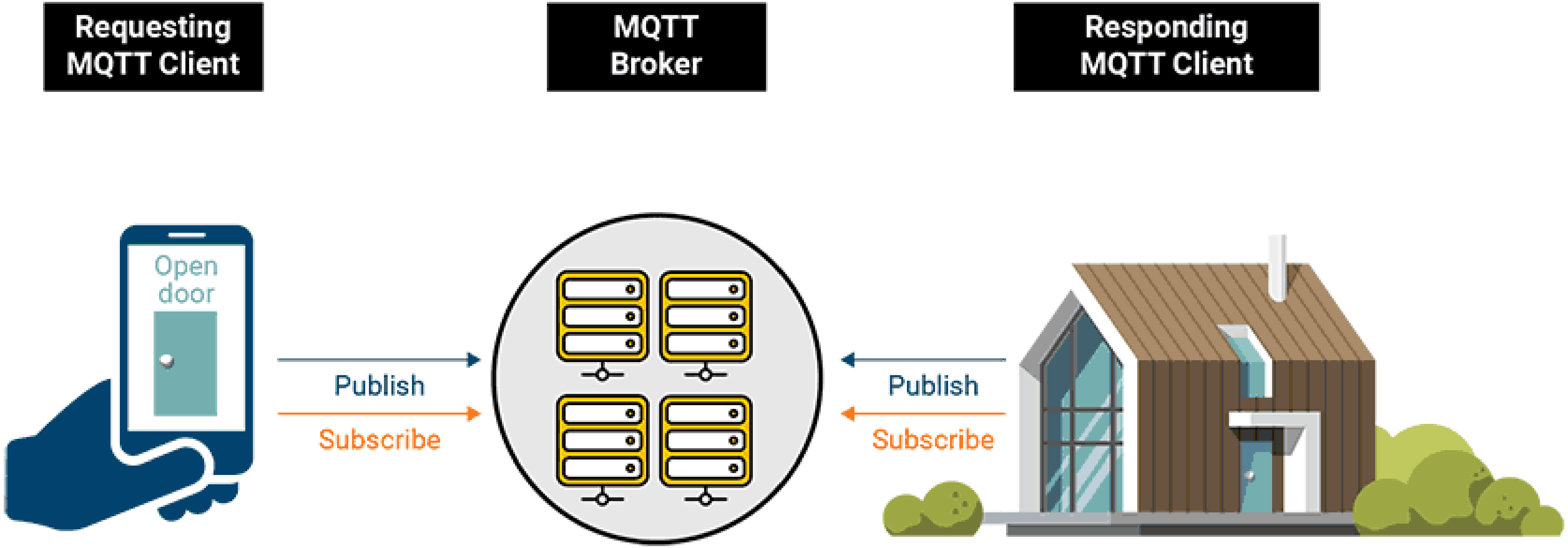
Broker:

- Broker is the component which take care of receiving data from Publisher and sending it to Subscriber accordingly.
- Broker has to filter messages, broker can filter messages in following ways.
  - Subject based
  - Content based
  - Type based
- Broker has a subscription table in which it store all the request from the subscriber for the publisher's data.
- Broker sends publish data to multiple subscriber according to this list.



https://www.slideshare.net/vgitcsofficialedupoi/mqtt-and-coap

Home Automation.



Source:https://www.hivemq.com/blog/mqtt5-essentials-part9-request-response-pattern/

Publish-Subscribe Mechanism Characteristics

Useful in IoT including:

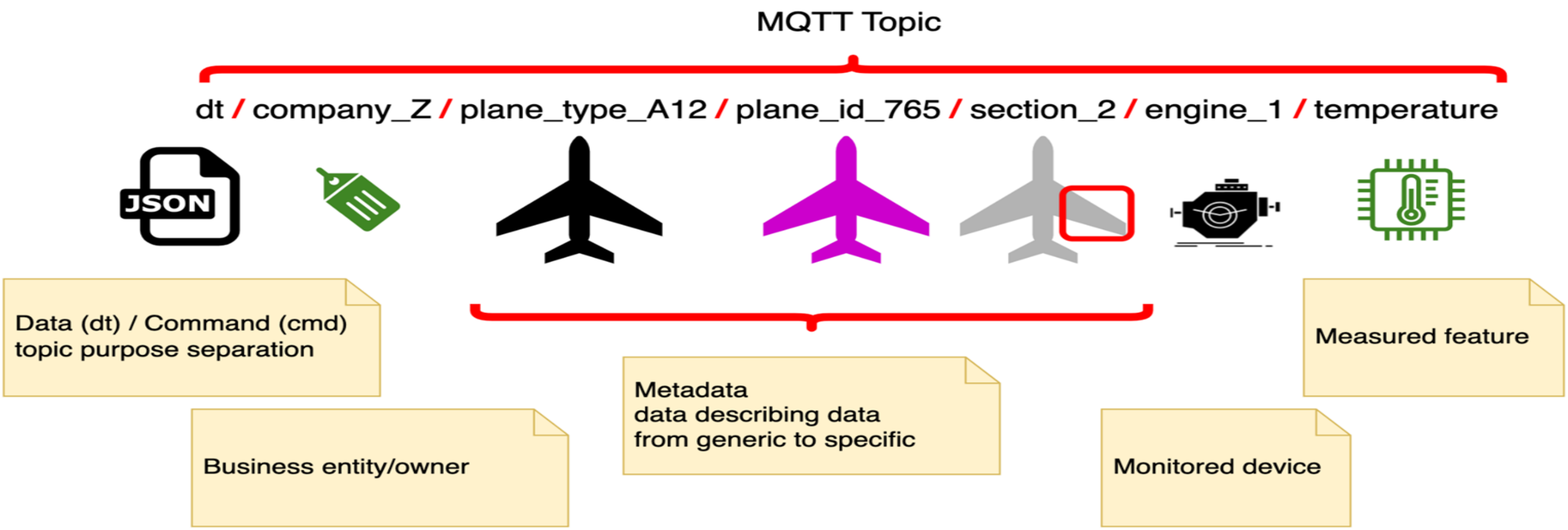
- **Decouples publishers from subscribers**
- **Allows for asynchronous communication**
- **Facilitates scalability**

MQTT QoS

There are three QoS levels in MQTT:

- QoS 0, at most once.
- QoS 1, at least once.
- QoS 2, exactly once.

MQTT Topic structure



MQTT Topic

dt / company_Z / plane_type_A12 / plane_id_765 / section_2 / engine_1 / temperature

Data (dt) / Command (cmd) topic purpose separation

Business entity/owner

Metadata
data describing data
from generic to specific

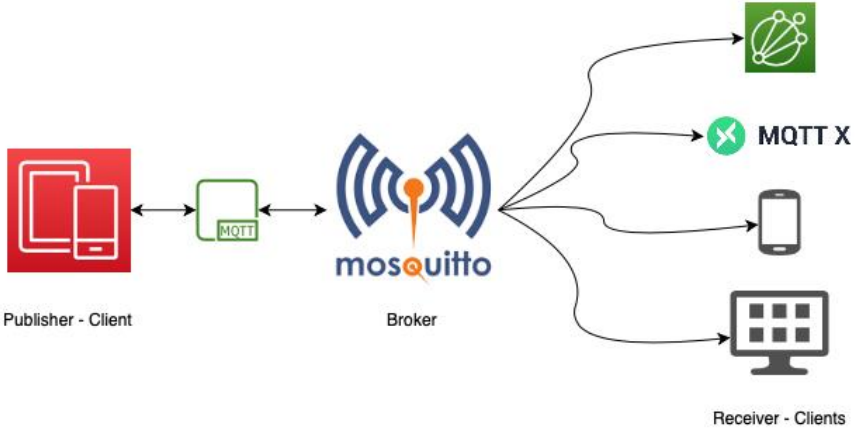
Monitored device

Measured feature

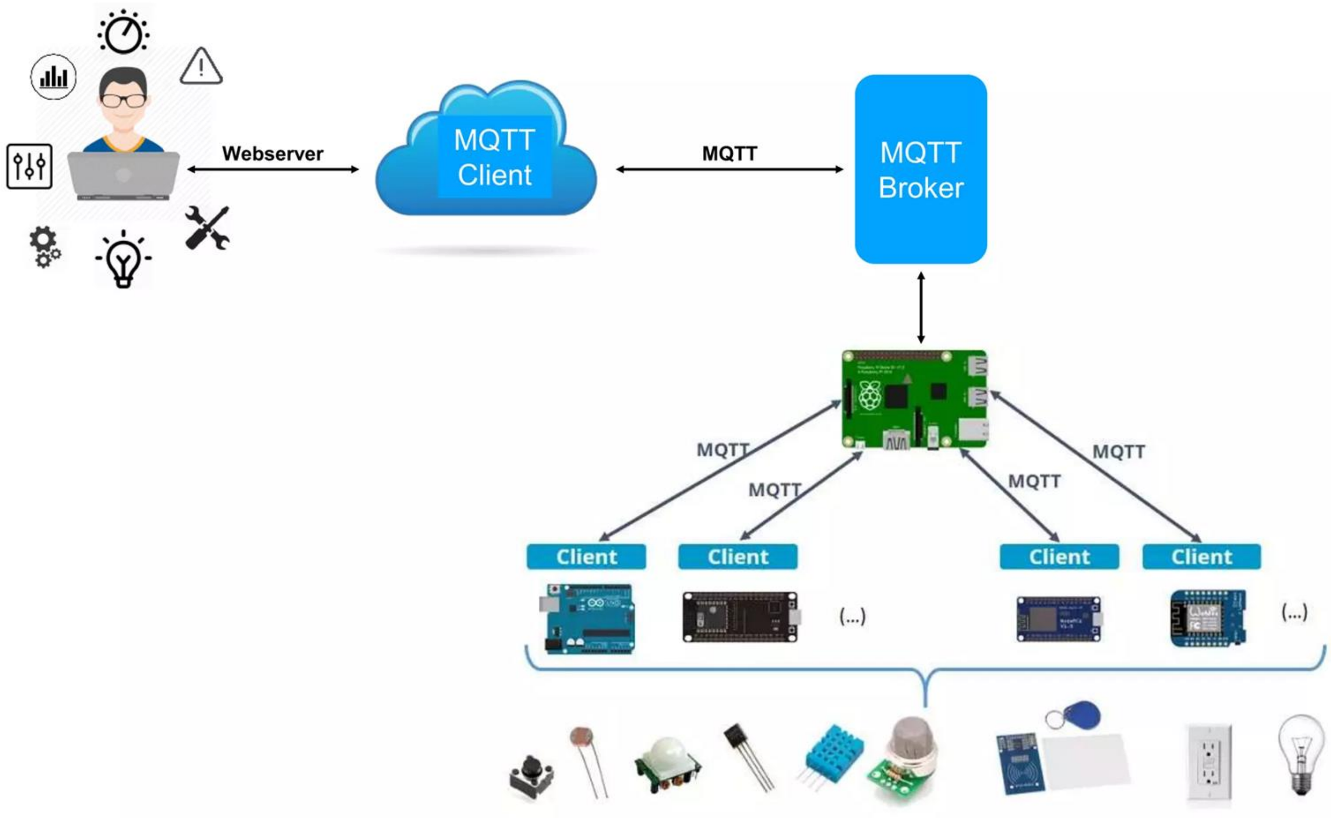
Source: https://www.thingrex.com/mqtt_topic_structure/

Mosquitto in MQTT

- Mosquitto in the context of MQTT protocol is an open-source message broker that implements the MQTT protocol.
- It is a server that receives all messages from the clients and then routes these messages to the appropriate destination clients.
- Key Features:
  - Lightweight and Efficient
  - Supports MQTT Protocol Versions
  - Cross-Platform
  - Security Features



MQTT-IOT-Example



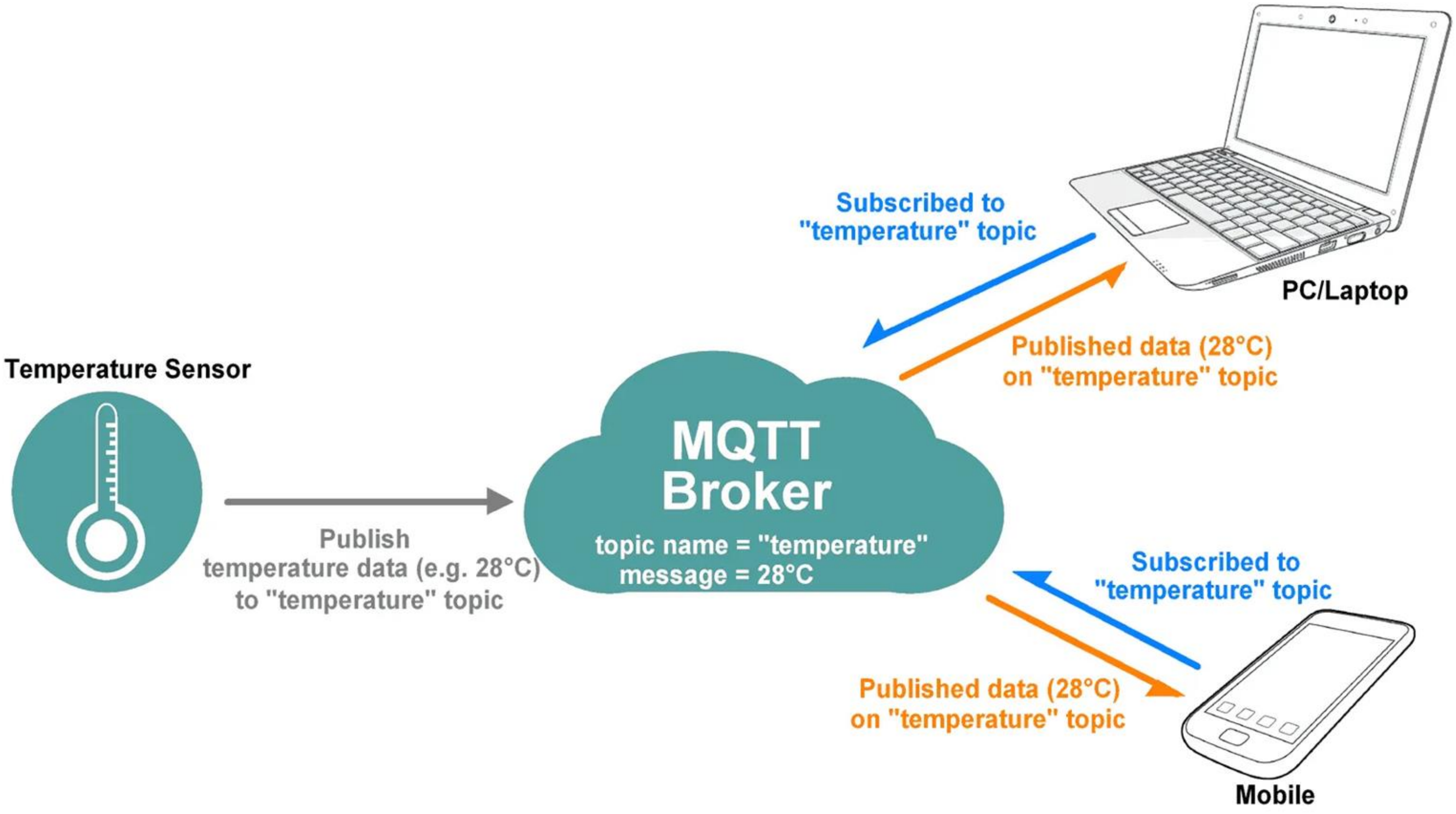

Subscribed to "temperature" topic
PC/Laptop
Published data (28°C) on "temperature" topic
Temperature Sensor
MQTT Broker
topic name = "temperature"
message = 28°C
Publish temperature data (e.g. 28°C) to "temperature" topic
Subscribed to "temperature" topic
Published data (28°C) on "temperature" topic
Mobile

Fig reference: https://www.mdpi.com/1424-8220/22/6/2174

CoAP Protocol

CoAP: The Key Protocol for Resource-Constrained Devices

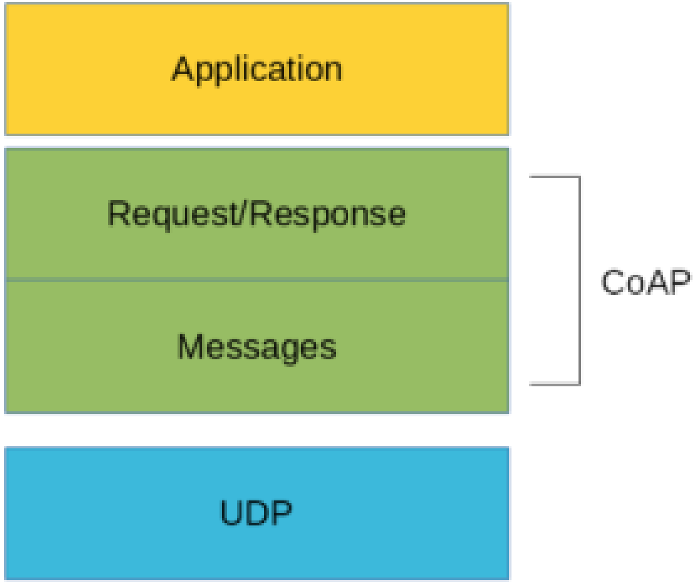
CoAP (Constrained Application Protocol) is specifically designed to facilitate communication for resource-constrained devices. It is built on the principles of RESTful communication, providing a lightweight and efficient protocol for constrained environments. CoAP operates over UDP, minimizing the overhead and making it suitable for devices with limited resources using client server architecture.

Coap Architecture

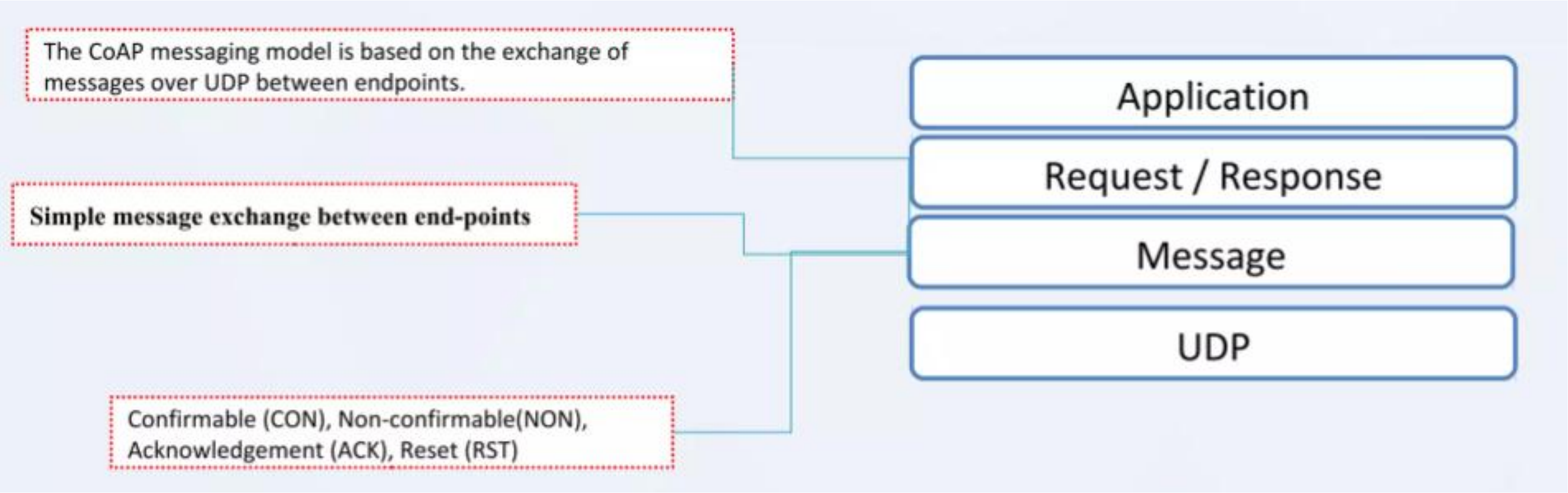
There are two different layers that make CoAP protocol: Messages and Request/Response.

The Messages layer deals with UDP and with asynchronous messages.

The Request/Response layer manages request/response interaction based on request/response messages



CoAP Protocol Layering



The CoAP messaging model is based on the exchange of messages over UDP between endpoints.

Simple message exchange between end-points

Confirmable (CON), Non-confirmable(NON), Acknowledgement (ACK), Reset (RST)

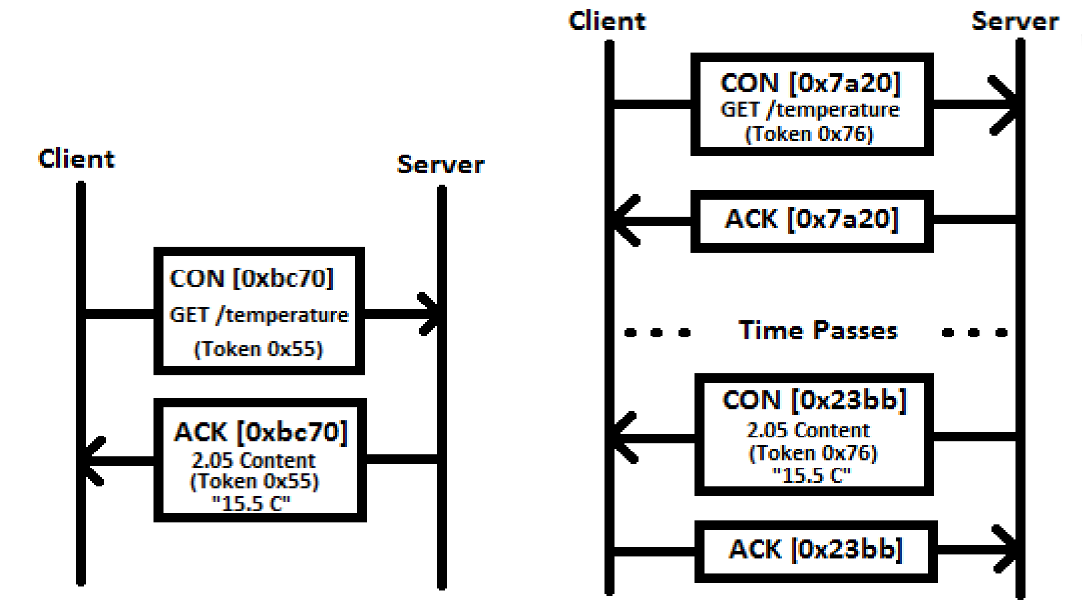
Application

Request / Response

Message

UDP

CoAP Asynchronous Message Exchange



a) Piggybacked response

b) Separate response.

Source: https://www.researchgate.net/figure/a-Piggybacked-response-b-Separate-response_fig3_335320124

CoAP using REST Architecture:

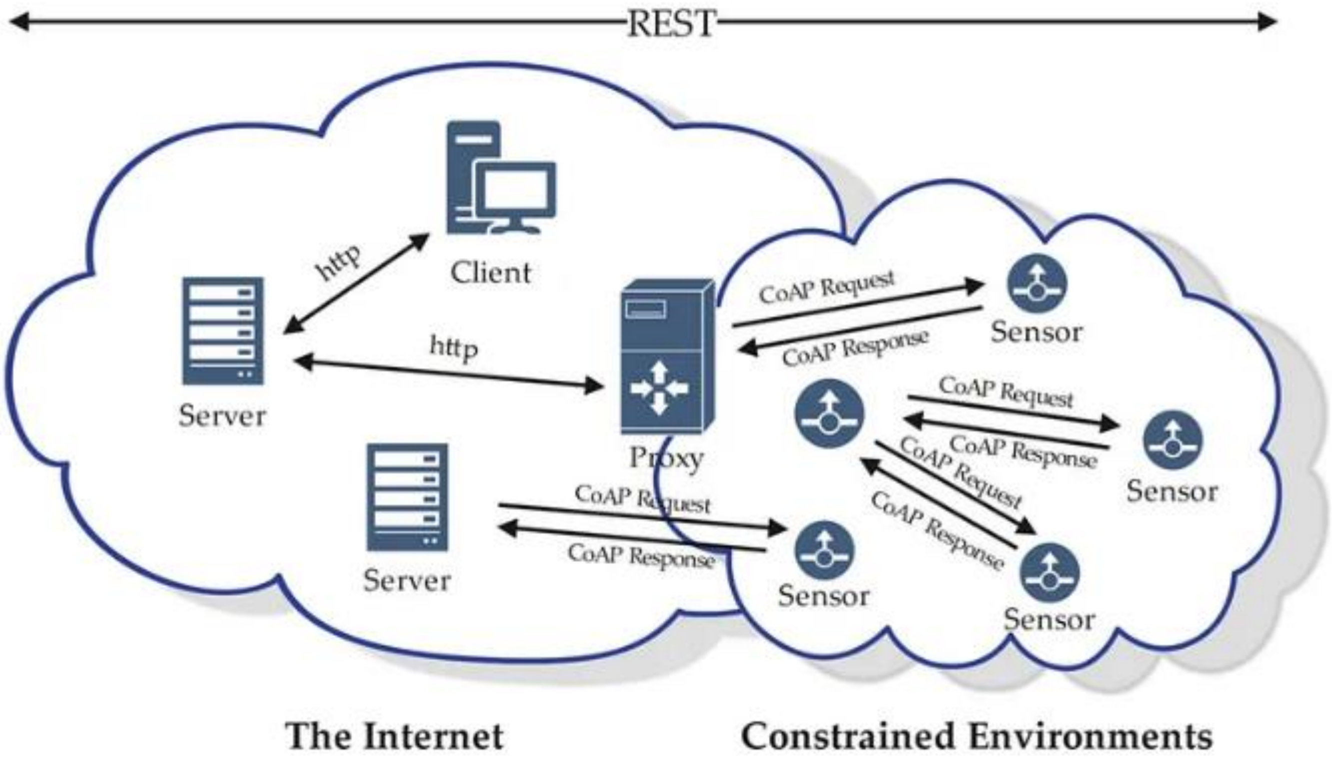


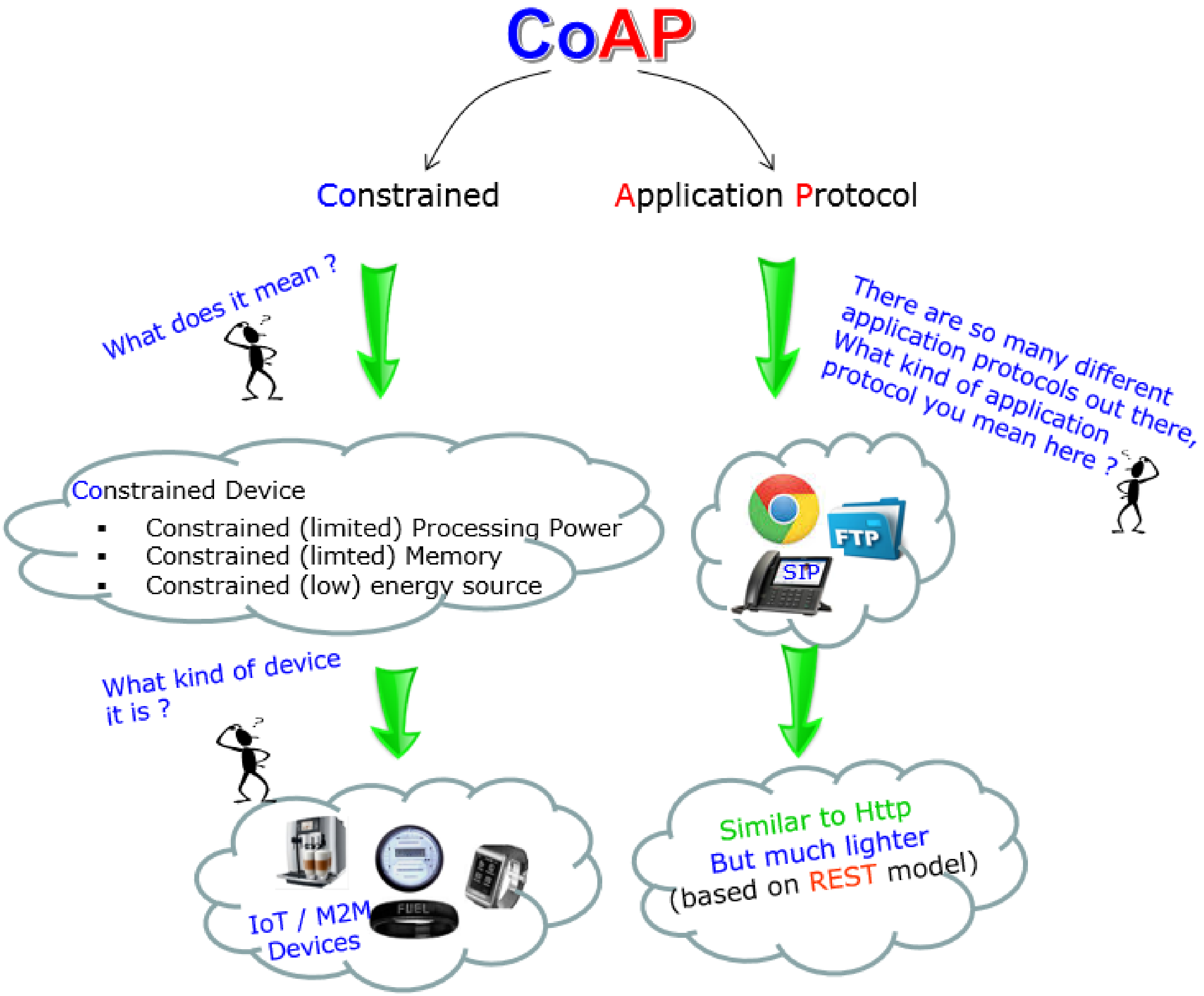
Image: ResearachGate

Key Features of CoAP

- CoAP is an easy-to-integrate protocol that can be quickly coupled with cross-protocol proxies applications. It works well with JSON, XML, CBOR, and various other data types. The web client is not informed that a sensor resource is being accessed throughout this operation.
- CoAP Protocol uses the tried and tested REST protocol is its main distinguishing characteristic. These allow developers to use a URL to conduct API-style calls to GET, PUT, POST, and DELETE data. This makes CoAP development a breeze.

Key Features of CoAP, contd

- The true power of adopting the entire REST protocol is the ease with which CoAP-enabled devices may be linked to existing applications. Existing REST-based apps may already receive input from CoAP-enabled devices with little or no changes.
- Because the resources required by CoAP are minimal, wastes are kept under control. There is no need to install a large transport stack for online transfers. The message processing header and encoding are concise and do not create fragmentation on the connection layer. It may support the functions of numerous servers at the same time.



CoAP

Constrained

Application Protocol

What does it mean ?

Constrained Device

- Constrained (limited) Processing Power
- Constrained (limted) Memory
- Constrained (low) energy source

What kind of device it is ?

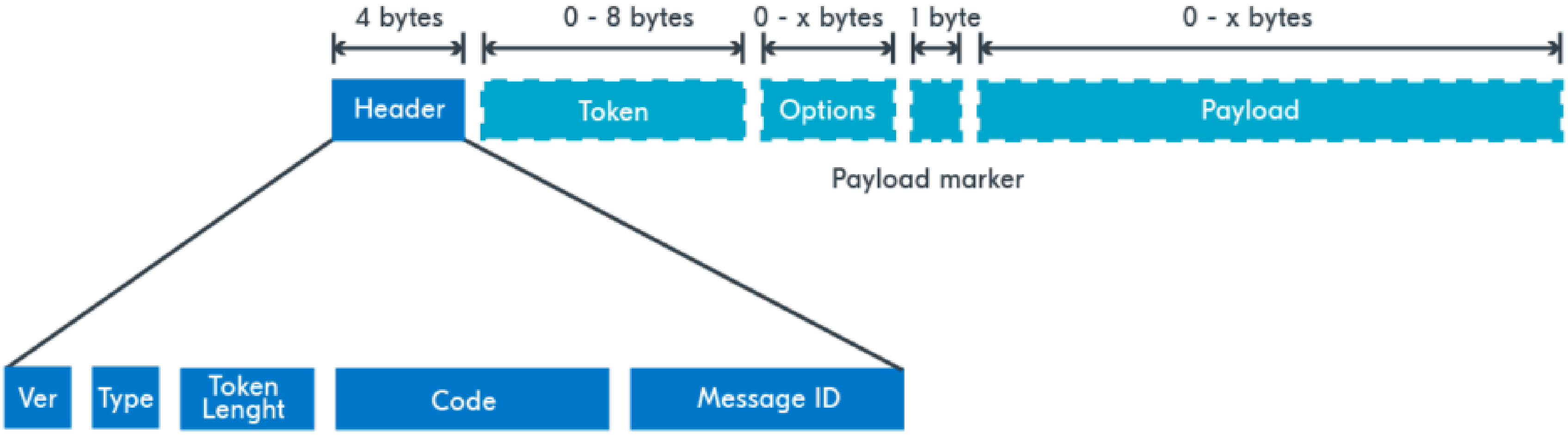
IoT / M2M Devices

There are so many different application protocols out there, What kind of application protocol you mean here ?

Similar to Http
But much lighter
(based on REST model)

Source: https://www.sharetechnote.com/html/IoT/image/IoT_CoAP_Concept_01.png

CoAP Message Model:



Source:https://academy.nordicsemi.com/courses/cellular-iot-fundamentals/lessons/lesson-5-cellular-fundamentals/topic/lesson-5-coap-protocol/

CoAP message header codes:

GET – retrieves the information that currently corresponds to the resource specified in the CoAP request

POST – requests that the server process the information enclosed within the request and either update the target resource or create a new resource with the information if the target resource didn’t already exist.

PUT – similar to a POST request, however, if the resource doesn’t exist, the server has the option to create the resource.

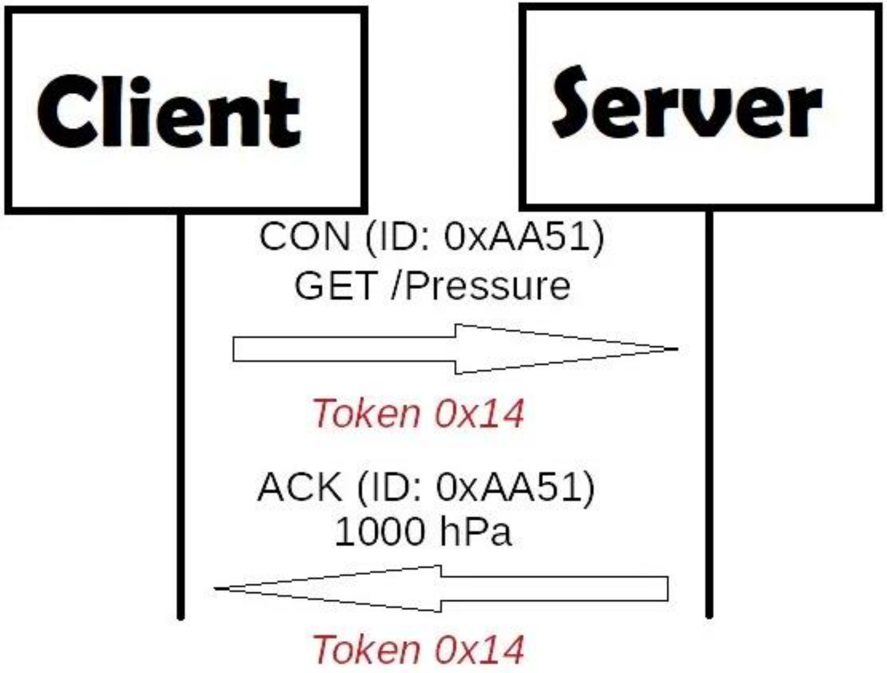
DELETE – requests that the resource specified in the CoAP request be deleted

CoAP Message types:

Confirmable:A CoAP confirmable message is a reliable message. In the CoAP protocol, a reliable message is obtained using a Confirmable message (CON). A CoAP Confirmable message is sent again and again until the other party sends an acknowledge message (ACK). The ACK message contains the same ID of the confirmable message (CON).

Non Conformable:These don’t require an Acknowledge by the server. These messages are unreliable messages means do not contain critical information that must be delivered to the server.These are denoted by (NON).

CoAP Confirmable message :



Client

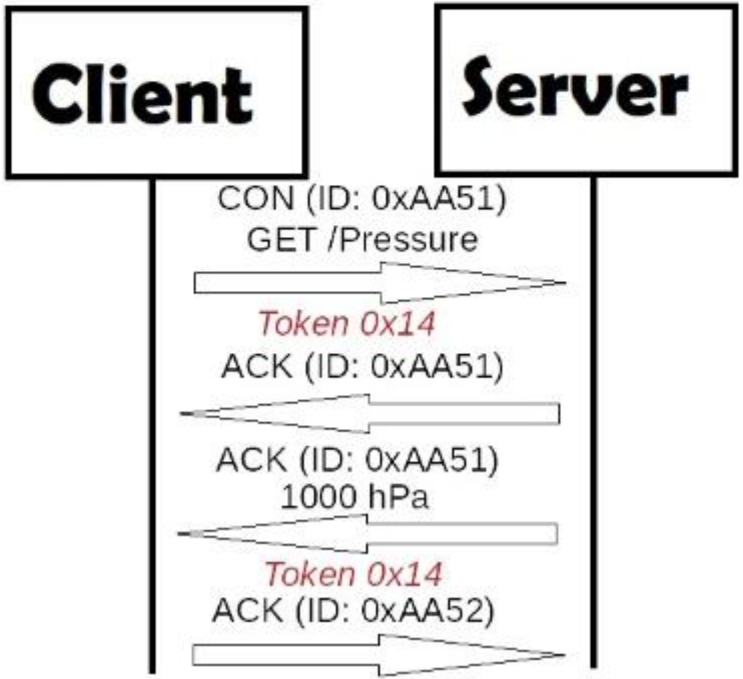
Server

CON (ID: 0xAA51)
GET /Pressure

Token 0x14

ACK (ID: 0xAA51)
1000 hPa

Token 0x14



Client

Server

CON (ID: 0xAA51)
GET /Pressure

Token 0x14

ACK (ID: 0xAA51)

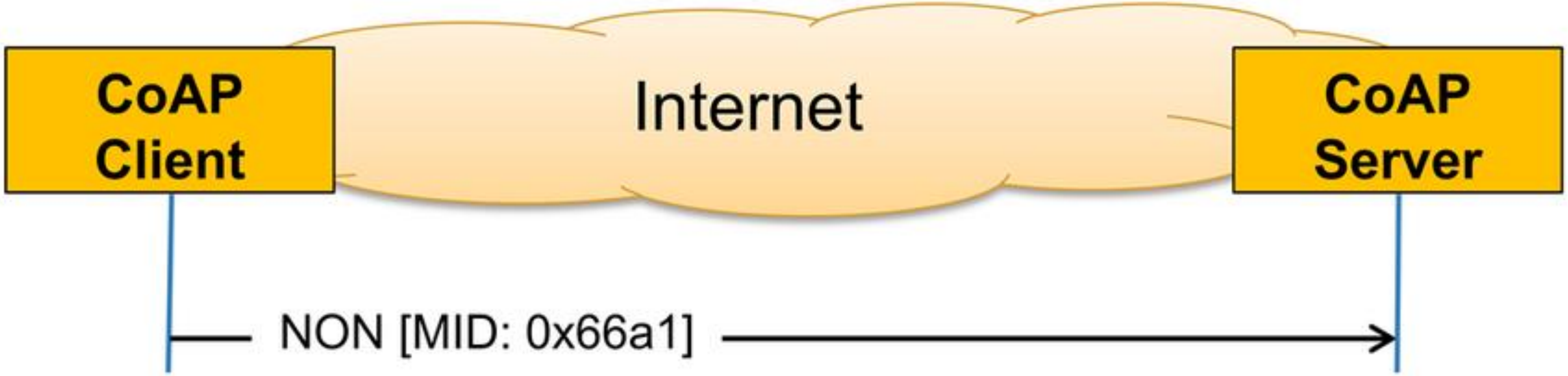
ACK (ID: 0xAA51)
1000 hPa

Token 0x14

ACK (ID: 0xAA52)

Reference:https://medium.com/@harshhvm/what-is-coap-protocol-coap-protocol-introduction-overview-3e8bac4d7f8e

Coap Non Confirmable message:



Reference:ResearchGate/305522152_Experimental_Evaluation_of_Unicast_and_Multicast_CoAP_Group_Communication

CoAP in Action

Smart City

CoAP plays a pivotal role in enabling smart city solutions, facilitating efficient communication and data exchange in urban environments.

Industrial IoT

Manufacturing and industrial setups leverage the benefits of CoAP for seamless and reliable device communication and control.

Agricultural IoT

In agricultural IoT deployments, CoAP ensures optimized communication and data exchange for precision farming and monitoring.

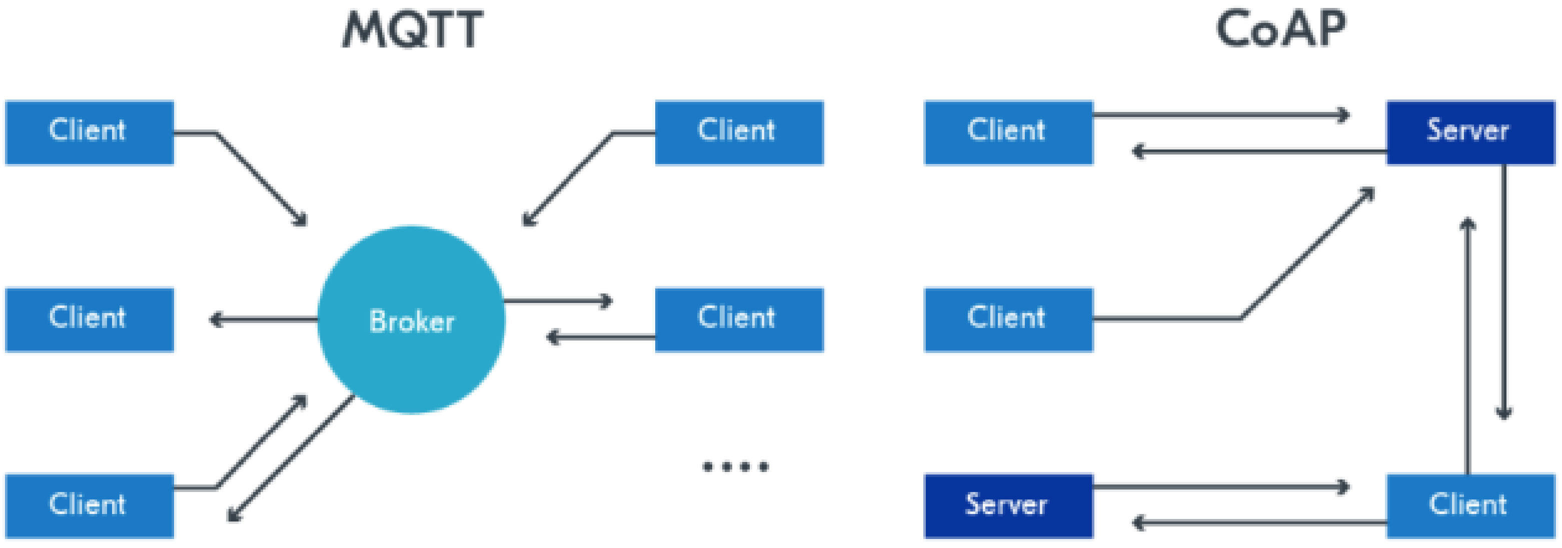
MPSSDEGB
कौशल से रोजगार, समृद्धि अपार

FITT
Foundation For Innovation And Technology Transfer

SANKALP
Ministry of Skill Development
& Entrepreneurship

Comparison of Key Protocols

Comparison of IoT Protocols, MQTT,CoAP



https://academy.nordicsemi.com/courses/cellular-iot-fundamentals/lessons/lesson-5-cellular-fundamentals/topic/lesson-5-coap-protocol/

Aspect	MQTT	CoAP
Latency	Low	Very Low
Bandwidth	Low	Low
Power Consumption	Low	Very Low
Scalability	High	Moderate
Advantages	Lightweight, efficient for low-bandwidth	RESTful, energy-efficient
Suitable Use Cases	IoT devices with low power and bandwidth, real-time updates	Constrained environments, M2M applications
Model	Publish/Subscribe	Full-duplex, Bidirectional
Addressing Mechanism	Topics	URIs
Transport Mechanism	TCP, UDP	UDP

Reference: https://bytebeam.io/blog/choosing-the-right-iot-protocol-a-comprehensive-guide-on-mqtt-coap-and-http/

Programming Exercise:

```
#defining a mean function
def mean(array):
    sum=0
    for x in array:#iterating through the entire array
      sum=sum+x
    return sum / len(array)

temperature=[25.5, 26, 27, 28, 29, 30, 25, 26, 27, 28]
pressure = [20, 21, 22, 23, 24, 25, 20, 21, 22, 23]
humidity = [40, 41, 42, 43, 44, 45, 40, 41, 42, 43]
print("Mean of Temperature is:",mean(temperature) )
print("Mean of Pressure is:",mean(pressure) )
print("Mean of Humidity is:",mean(humidity))
```

https://colab.research.google.com/drive/1PuG9nr4PbYvveveyxvGWETKiEp4W2qSi#scrollTo=XNNzx6nwYlkj

Homework Assignments 1

1. Design a smart classroom in your college.

    1a. Which sensors can be used?

    1b. Which IoT Communication protocol would you suggest?

    1c. How do you extend it to College and then University level?

Homework Assignment 2

2. Consider helping Pench Tiger reserve officials using IoT devices. Intruder and poaching detection and animal movement tracking are the sub tasks.

2a. What are the IoT devices required?

2b. Which communication protocol you will use?

2c. How to extend it to centralized monitoring system? Extending it to Full Madhya Pradesh.

2d. What are the challenges faced for interoperability when we try to extent it for entire Madhya Pradesh? How to enforce interoperability standards?

Homework Assignment 3

3a. Like IOT explore Web of Things.

3b. Basic protocols used in web of things.

3c. How can we work on Security issues and extensibility.

3d. What about interoperability?