INTERNET OF THINGS

- What is IoT
- Advantages of IoT
- Features of IoT
- IoT architecture & protocol
- IoT platforms
- Device selection for IoT
- Some applications

KUNTAL**MAJHI**

WHAT IS **IOT** ?

IOT is basically connecting things(sensors, electronics device, complex Data model which used in industry) with internet.

IOT framework is designed to connect the information from devices which are interconnected. The process has been classified into five phases.

1. Create Phase (sensors collect the data from the environment)
2. Communicate phase(data generated in the first phase are communicated)
3. Aggregate phase (collected data aggregate with device itself)
4. Analyse phase (data are used to generate pattern)
5. Act phase (action is taken on the basic of information)

IOT **ADVANTAGE**

1. Improve control of operation processes
2. Improve monitoring
3. Achieve Customer-Centricity
4. New capabilities to predict and act
5. Improve automation and saves time
6. Rapid response
7. Reduction of human errors

IOT **FEATURES**

1. Connectivity
2. Sensing
3. Active Engagements
4. Dynamic Nature
5. Intelligence
6. Energy Saving
7. Integration

Kuntal Majhi

IOT **ARCHITECTURE** 1

IoT architecture comprised with many components:

1. Things(sensor & actuator)
2. Gateway(data processing, filtering, cloud communication)
3. Streaming data processor(distribute sensor data)
4. Data warehouse or lake (store data)
5. Machine learning and control application(Generate data models with developing algorithm)
6. Application and Analytics(make decision with available data)

IOT **ARCHITECTURE** 2

The perception layer is the physical layer, which has sensors for sensing and gathering information about the environment. It senses some physical parameters or identifies other smart objects in the environment.

The transport layer transfers the sensor data from the perception layer to the processing layer and vice versa through networks such as wireless, 3G, LAN, Bluetooth, RFID, and NFC.

The processing layer is also known as the middleware layer. It stores, analyzes, and processes huge amounts of data that comes from the transport layer. It can manage and provide a diverse set of services to the lower layers. It employs many technologies such as databases, cloud computing, and big data processing modules.

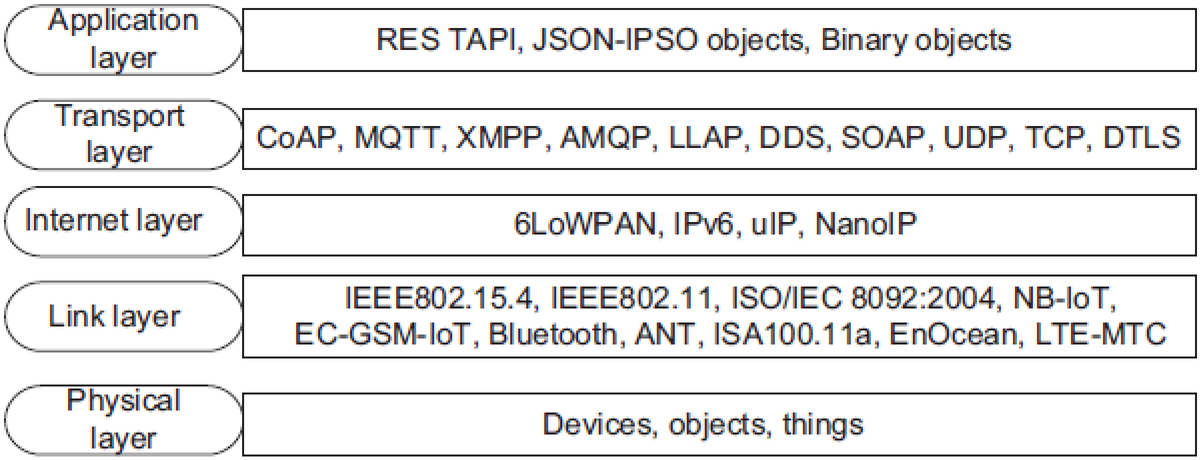
The application layer is responsible for delivering application specific services to the user. It defines various applications in which the Internet of Things can be deployed, for example, smart homes, smart cities, and smart health.

The business layer manages the whole IoT system, including applications, business and profit models, and users' privacy. The business layer is out of the scope of this paper. Hence, we do not discuss it further.

IOT **PROTOCOL** 1

OSI MODEL:

The open system interconnection(OSI) model for IoT protocol having five Layers –



Application layer	RES TAPI, JSON-IPSO objects, Binary objects
Transport layer	CoAP, MQTT, XMPP, AMQP, LLAP, DDS, SOAP, UDP, TCP, DTLS
Internet layer	6LoWPAN, IPv6, uIP, NanoIP
Link layer	IEEE802.15.4, IEEE802.11, ISO/IEC 8092:2004, NB-IoT, EC-GSM-IoT, Bluetooth, ANT, ISA100.11a, EnOcean, LTE-MTC
Physical layer	Devices, objects, things

IOT **PROTOCOL** 2

Organizational Levels:

1. Infrastructure (IPv4/IPv6, 6LowPAN, RPL)
2. Identification (EPC, IPv6, uCode, URIs)
3. Communication (Bluetooth, Wi-Fi, LPWAN)
4. Discovery (DNS-SD, mDNS, Physical Web)
5. Data Protocols (AMQP, MQTT, Websocket, CoAP, Node)
6. Device Management (TR-069, OMA-DM)
7. Semantic (Web Thing Model, JSON-LD)
8. Multi-layer Frameworks (Weave, IoTivity, Alljoyn, Homekit)

IOT COMMUNICATION MEDIA

It may be wireless or wired depend upon availability –

A) Wireless:

1. Short range (BLE, Wi-Fi , Li-Fi, NFC, RFID , Z-wave, ZigBee)
2. Medium range (HaLow, LTE- advance)
3. Long range (LPWAN, VSAT)

B) Wired:

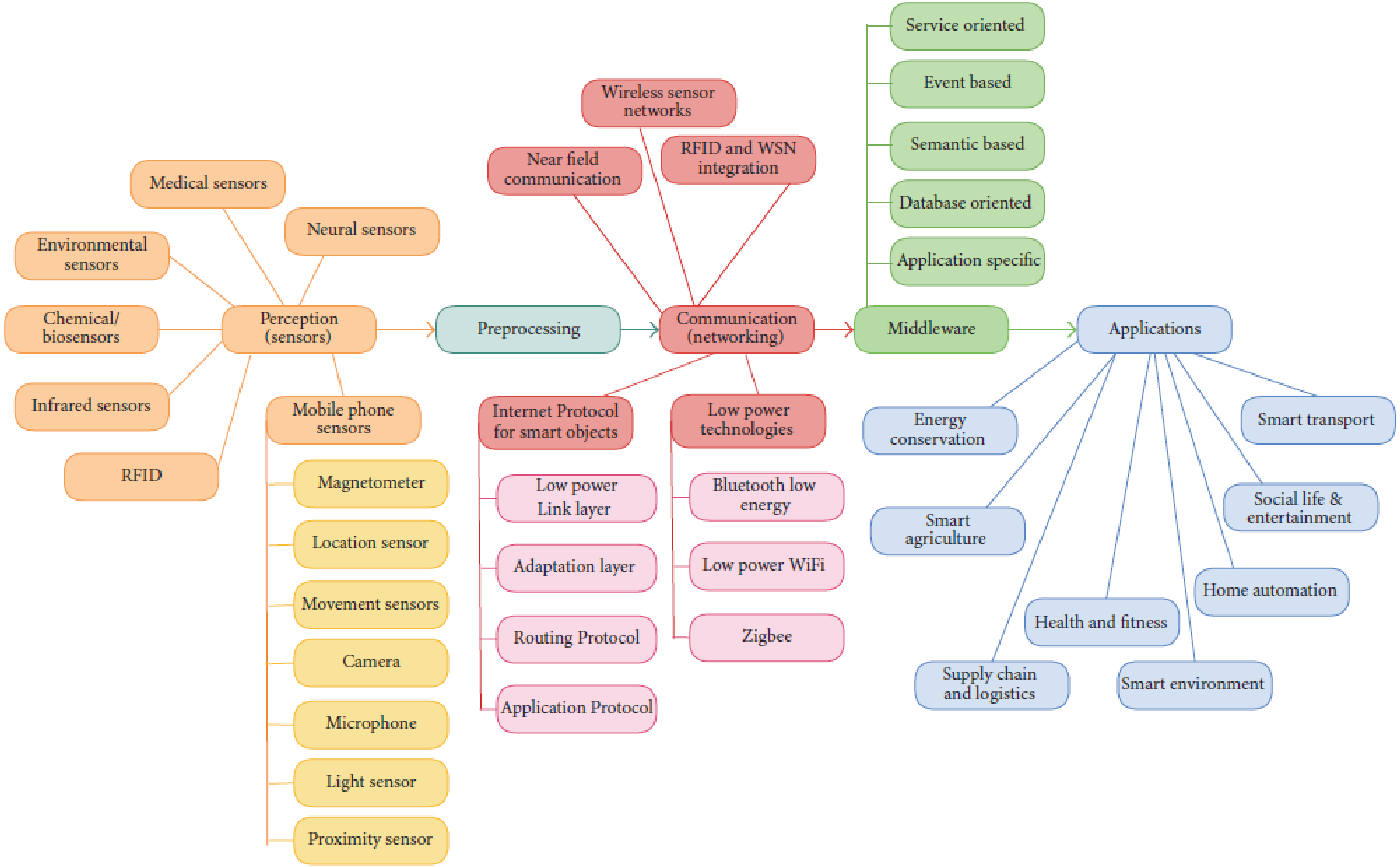
Ethernet, Multimedia over Coax Alliance (MoCA), Power-line communication(PLC)

IOT **PLATFORMS**

The IoT can’t function without software, including middleware, known as an IoT or IoT cloud platform. As a form of middleware, an IoT platform, sits between the layers of IoT devices and IoT gateways.
Some names are-

- Google Cloud IoT
- AWS IoT
- Microsoft Azure IoT
- IBM WatsonIoT
- Oracle IoT
- SalesForse IoT
- Bosch IoT
- ThingsIO.AI
- Cisco IoT
- Thingworx
- Kaa
- Samsung Artik
- Zetta
- Particle
- ThingSpeak
- Siemens MindSphere

Kuntal Majhi



- Perception (sensors)
  - Medical sensors
  - Neural sensors
  - Environmental sensors
  - Chemical/ biosensors
  - Infrared sensors
  - RFID
  - Mobile phone sensors
    - Magnetometer
    - Location sensor
    - Movement sensors
    - Camera
    - Microphone
    - Light sensor
    - Proximity sensor
- Preprocessing
- Communication (networking)
  - Wireless sensor networks
  - Near field communication
  - RFID and WSN integration
  - Internet Protocol for smart objects
    - Low power Link layer
    - Adaptation layer
    - Routing Protocol
    - Application Protocol
  - Low power technologies
    - Bluetooth low energy
    - Low power WiFi
    - Zigbee
- Middleware
  - Service oriented
  - Event based
  - Semantic based
  - Database oriented
  - Application specific
- Applications
  - Energy conservation
  - Smart agriculture
  - Supply chain and logistics
  - Health and fitness
  - Smart environment
  - Home automation
  - Social life & entertainment
  - Smart transport

Kuntal Majhi

IOT DEVICE **SELECTION**

Depend upon project and needs or availability

Arduino	**Raspberry Pi**
It's a microcontroller	It's a mini computer
No operation system	Has it's own operating system
Low RAM	High RAM
8bit CPU	64bit CPU
Both analog and digital pins	Only digital gpio pins
i/o current drive strength 40mA	i/o current drive strength 16mA
Does not support audio & GUI	Support audio & GUI
Arduino IDE & compiler	Wide range of operating system
Best at controlling machines and -performing respective task	Best at logical processing of data and -communicating with other system

Kuntal Majhi

START WITH **RASPBERRY PI** (VNC) 1

1. first download the OS of RPi from its website extract the image.
2. download balenaEtcher to flash the OS in memory card.

- create a empty text file named "ssh" in the visible folder of memory card.
- for connecting network automatically create another text file and write the coding.

```
country=IN
update_config=1
ctrl_interface=/var/run/wpa_supplicant

network={
            scan_ssid=1
            ssid="network_name"
            psk="password"
           }
```

named it "wpa_supplicant".config and keep it in visible folder.

Kuntal Majhi

START WITH **RASPBERRY PI** (VNC) 2

1. Give power to RPi and scan the IP address with any IP scanner.
2. Download putty and type the IP address in there and login to the RPi

   username: pi
   password: raspberry port:22

The RPI prompt is open then type to enter the config menu:

```
sudo raspi-config
```

then on vnc and ssh from interfacing menu
change the resolution to full and reboot it.

Go to vnc_viewer and type the IP address and RPI desktop will open
In your pc.

Kuntal Majhi

CLIENT-SERVER **MODEL**

server have data client needs data. IP address is needed for client-sever talking

Rpi Terminal:

```
Server:   nc -l  1234
Client:   nc  127.0.0.1  1234
```

```
#Create a client
import  socket
my_soc = socket.socket()
my_soc.connect("127.0.0.1",1234)
my_soc.sendall(b"hello from client\n")
my_soc.close()
```

```
#create a server
import socket
server_soc = socket.socket()
server_soc.bind("ip address",1234)
server_soc.listen(5)
conn,adress = server_soc.accept()
conn.sendall(b"hi client\n")
Data = conn.recv(1000)
Data
conn.close()
serer_soc.close()
```

Kuntal Majhi

SENDING DATA TO **THINGSPEAK** 1

http protocol

```
#!/usr/bin/python3
from time import sleep
from urllib.request import urlopen

a = 1
baseURL = 'http://api.thingspeak.com/update?api_key=DX0GDLMQH4Z6ZBIU&field1='
while(a < 100):
    print (a)
    f = urlopen(baseURL +str(a))
    f.read()
    f.close()
    sleep(5)
    a += a
print ("Program has ended")
```

Kuntal Majhi

SENDING DATA TO **THINGSPEAK** 2

mqtt protocol

```
import paho.mqtt.client as mqtt

client = mqtt.Client()
client.connect("mqtt.thingspeak.com",1883,60)

channelId = "285697"
apiKey = "ZJJKFJNYVRQWJRFD"

client.publish("channels/%s/publish/%s" % (channelId,apiKey), "field1=26&field2=1013")
client.loop(2)
```

Kuntal Majhi

CGI SERVER USING **PYTHON** 1

1. First create a python file name as your choice and save it.in this case "test.py" Write

```
print("content-type:text/html\n")
print("this is my first server")
```

2. Create a new folder in your C drive and named any. In this case I named "my_server".

3. Then create another folder inside my_server and name "cgi-bin".

4. Move the python file inside it.

5. Open command window and change the directory to my_server.

```
Cd c:\my_server
```

Kuntal Majhi

CGI SERVER USING **PYTHON** 2

6. Then configure it as a server to do it type

```
Python –m http.server –cgi 8000
```

7. Type in your browser

```
Localhost:8000/cgi-bin/test.py
```

8. Now you can see in your browser the line is priented

this is my first server .

9. Done.

EXAMPLE:

```
print("content_type:text/html\n")
print()
print("<html>")
print("<head>")
print("<title> internet of things</title>")
print("</head>")
print("<body>")
print("<h2>hello welcome to iot training</h2>")
print("<h1>training will end on monday</h2>")
print("</body>")
```

UPDATE COMMAND ID **THROUGH URL**

copy and paste the url in browser it will show the current status.
Install REST api as an extension in chrome
Then go to that extension.
create new project >+sign >new request > change set to put>paste url

>>For mobile:
rest api client android>paste url id>remove http from url>select https> and change to put

>>to react automatically:
apps>react>new>fill the data accordingly >in action put thingshttp> set new thingshttp>paste url>save react

Kuntal Majhi