Internet of Things

DR. RAIHAN UL ISLAM

ASSOCIATE PROFESSOR

DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING

ROOM NO# AB3-1003
EMAIL: RAIHAN.ISLAM@EWUBD.EDU

MOBILE: +8801992392611

Internet of things

- Internet Of Things is Fully Networked and Connected Devices sending analytics data back to cloud or data center.
- The definition of Internet of things is that it is the network in which every object or **thing is provided unique identifier** and data is transferred through a network without any verbal communication.
- **Scope of IoT is not just limited to just connecting things to the internet, but it allows these things to communicate and exchange data, process them as well as control them while executing applications.**

Formal Definition of IoT

- A **dynamic global network** infrastructure with **self- configuring capabilities** based on standard and **interoperable communication protocols,** where physical and virtual "things" have **identities,** physical attributes, and use intelligent interfaces, and are seamlessly **integrated into information network** that communicate data with users and environments.

Characteristics of IoT

- **Dynamic Global network & Self-Adapting** : Adapt the changes w.r.t changing contexts
- **Self Configuring** : E.g. Fetching latest s/w updates without manual intervention.
- **Interoperable Communication Protocols** : Communicate through various protocols
- **Unique Identity** : Such as Unique IP Address or a URI
- **Integrated into Information Network** : This allows to communicate and exchange data with other devices to perform certain analysis.

Physical Design of IoT

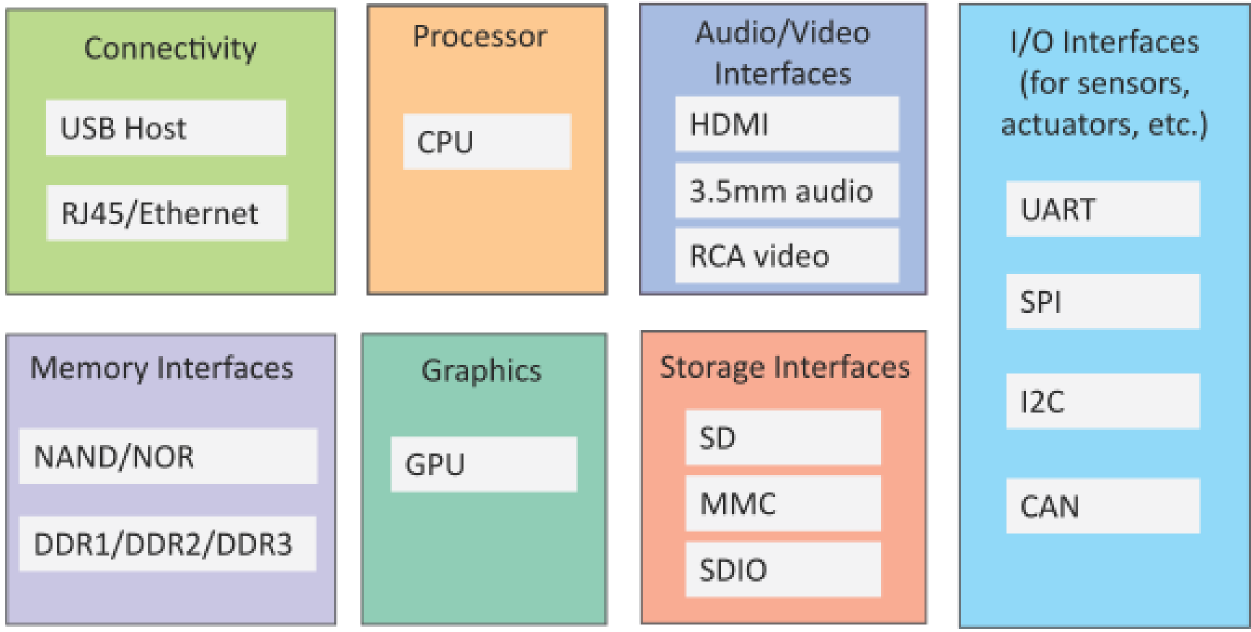
- Things in IoT
- IoT Protocols

Things in IoT

- Refers to IoT devices which have **unique identities** that can perform **sensing, actuating and monitoring capabilities.**
- IoT devices can exchange data with other connected devices or collect data from other devices and process the data either locally or send the data to centralized servers or cloud – based application back-ends for processing the data.

Generic Block Diagram of an IoT Device

- An IoT device may consist of several interfaces for connections to other devices, both wired and wireless.
  - I/O interfaces for sensors
  - Interfaces for internet connectivity
  - Memory and storage interfaces
  - Audio/video interfaces



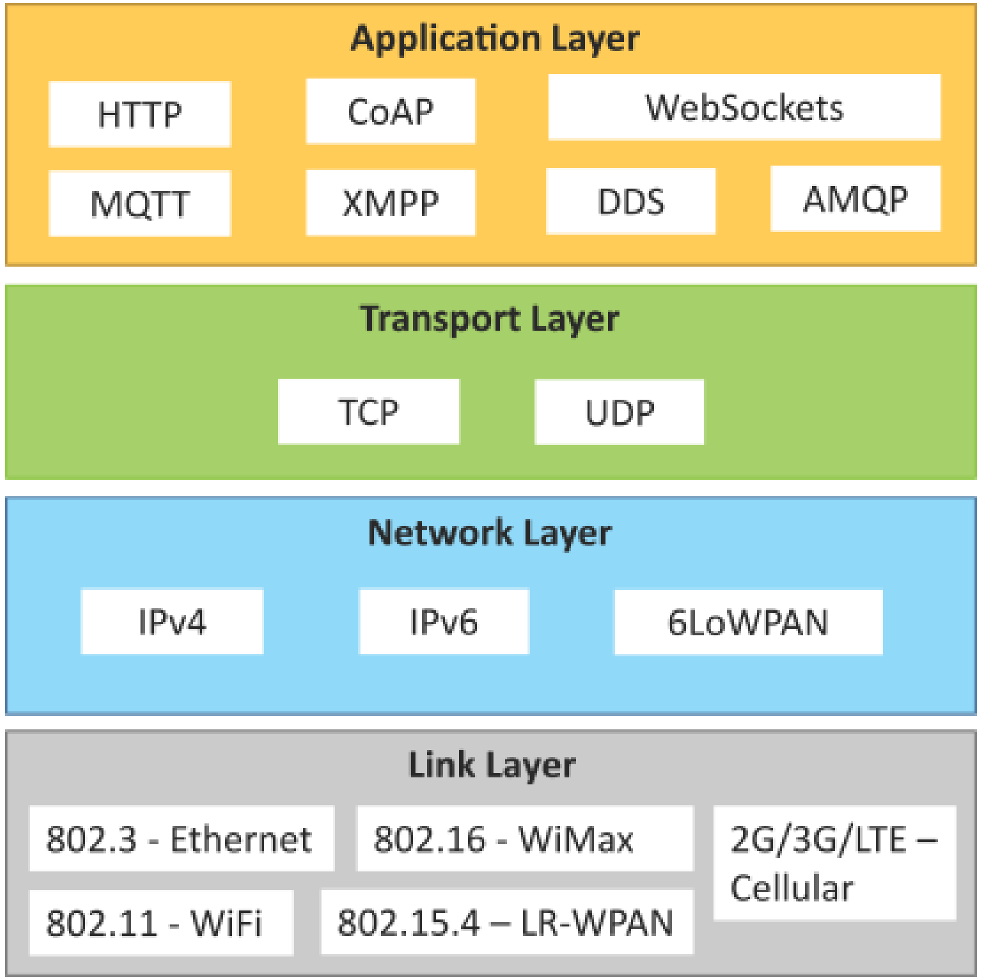
IoT Protocols

Link Layer
- 802.3 – Ethernet
- 802.11 – WiFi
- 802.16 – WiMax
- 802.15.4 – LR-WPAN
- 2G/3G/4G

Network/Internet Layer
- IPv4
- IPv6
- 6LoWPAN

Transport Layer
- TCP
- UDP

Application Layer
- HTTP
- CoAP
- WebSocket
- MQTT
- XMPP
- DDS
- AMQP



IoT Protocols…Link Layer…Ethernet

Sr.No	Standard	Shared medium
1	802.3	Coaxial Cable…10BASE5
2	802.3.i	Copper Twisted pair …..10BASE-T
3	802.3.j	Fiber Optic……10BASE-F
4	802.3.ae	Fiber…..10Gbits/s

Data Rates are provided from 10Gbit/s to 40Gb/s and higher

IoT Protocols…Link Layer…WiFi

Sr.No	Standard	Operates in
1	802.11a	5 GHz band
2	802.11b and 802.11g	2.4GHz band
3	802.11.n	2.4/5 GHz bands
4	802.11.ac	5GHz band
5	802.11.ad	60Hz band

- Collection of Wireless LAN
- Data Rates from 1Mb/s to 6.75 Gb/s

IoT Protocols…Link Layer…WiMax

Sr.No	Standard	Data Rate
1	802.16m	100Mb/s for mobile stations 1Gb/s for fixed stations

- Collection of Wireless Broadband standards
- Data Rates from 1.5Mb/s to 1 Gb/s

IoT Protocols…Link Layer…LR-WPAN

- Collection of standards for low-rate wireless personal area networks
- Basis for high level communication protocols such as Zigbee
- Data Rates from 40Kb/s to 250Kb/s
- Provide low-cost and low-speed communication for power constrained devices

IoT Protocols…Link Layer…2G/3G/4G/5G – Mobile Communication

Sr.No	Standard	Operates in
1	2G	GSM-CDMA
2	3G	UMTS and CDMA 2000
3	4G	LTE
4	5G	**NR (New Radio)**, supports **NB-IoT & LTE-M**

- Data Rates from 9.6Kb/s (for 2G) to up to 100Mb/s (for 4G)

IoT Protocols...Network/Internet Layer

- Responsible for sending of IP datagrams from source to destination network
- Performs the host addressing and packet routing
- Host identification is done using hierarchical IP addressing schemes such as IPV4 or IPV6

IoT Protocols…Network Layer

- IPV4
  - Used to identify the devices on a network using hierarchical addressing scheme
  - Uses 32-bit address scheme
- IPV6
  - Uses 128-bit address scheme
- 6LoWPAN (IPV6 over Low power Wireless Personal Area Network)
  - Used for devices with limited processing capacity
  - Operates in 2.4 Ghz
  - Data Rates of 250Kb/s

IoT Protocols…Transport Layer

- Provide end-to-end message transfer capability independent of the underlying network
- It provides functions such as error control, segmentation, flow-control and congestion control

IoT Protocols…TCP

- Transmission Control Protocol
- Connection Oriented
- Ensures Reliable transmission
- Provides Error Detection Capability to ensure no delicacy of packets and retransmit lost packets
- Flow Control capability to ensure the sending data rate is not too high for the receiver process
- Congestion control capability helps in avoiding congestion which leads to degradation
- of n/w performance

IoT Protocols…UDP

- User Datagram Protocol
- Connectionless
- Does not ensures Reliable transmission
- Does not do connection before transmitting
- Does not provide proper ordering of messages
- Transaction oriented and stateless

IoT Protocols…Application Layer…Hyper Transfer Protocol

- Forms foundation of World Wide Web(WWW)
- Includes commands such as GET,PUT, POST, HEAD, OPTIONS, TRACE..etc
- Follows a request-response model
- Uses Universal Resource Identifiers(URIs) to identify HTTP resources

IoT Protocols…Application Layer…CoAP

- Constrained Application Protocol
- Used for Machine to machine (M2M) applications meant for constrained devices and n/w's
- Web transfer protocol for IoT and uses request-response model
- Uses client –server architecture
- Supports methods such as GET,POST, PUT and DELETE

IoT Protocols…Application Layer

WebSocket

- Allows full-duplex communication over single socket
- Based on TCP
- Client can be a browser, IoT device or mobile application

MQTT

- Message Queue Telemetry Transport , light-weight messaging protocol
- Based on publish-subscribe model
- Well suited for constrained environments where devices have limited processing, low memory and n/w bandwith requirement

IoT Protocols…Application Layer…XMPP

- Extensible messaging and presence protocol
- For Real time communication and streaming XML data between n/w entities
- Used for Applications such as Multi-party chat and voice/video calls.
- Decentralized protocol and uses client server architecture.

IoT Protocols…Application Layer…DDS

- Data Distribution service is a data-centric middleware standard for device-to-device or machine-to-machine communication.
- Publish subscribe model where publishers create topics to which subscribers can use.
- Provides Quality-of-service control and configurable reliability.

IoT Protocols…Application Layer…AMQP

- Advanced Messaging Queuing Protocol used for business messaging.
- Supports both point-to-point and publisher/subscriber models, routing and queuing
- Broker here receives messages from publishers and route them over connections to consumers through messaging queues.

Comparisons of IoT Protocols

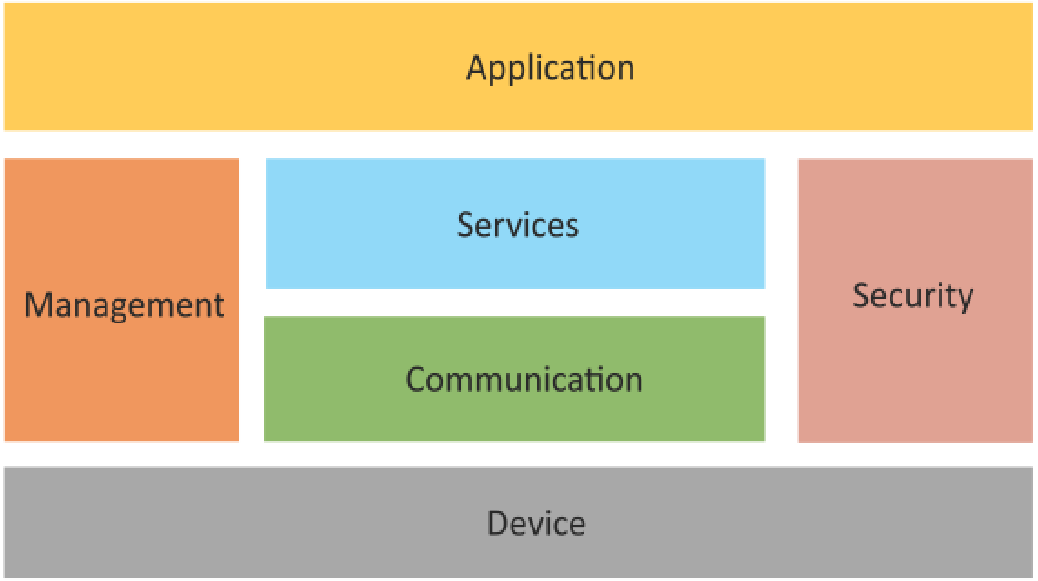
	DDS	AMQP	CoAP	MQTT	REST / HTTP	XMPP
TRANSPORT	UDP/IP (unicast + multicast) TCP/IP	TCP/IP	UDP/IP	TCP/IP	TCP/IP	TCP/IP
INTERACTION MODEL	Publish-and-Subscribe, Request-Reply	Point-to-Point Message Exchange	Request-Reply (REST)	Publish-and-Subscribe	Request-Reply	Point-to-Point Message Exchange
SCOPE	Device-to-Device Device-to-Cloud Cloud-to-Cloud	Device-to-Device Device-to-Cloud Cloud-to-Cloud	Device-to-Device	Device-to-Cloud Cloud-to-Cloud	Device-to-Cloud Cloud-to-Cloud	Device-to-Cloud Cloud-to-Cloud
AUTOMATIC DISCOVERY	✓	-	✓	-	-	-
CONTENT AWARENESS	Content-based Routing Queries	-	-	-	-	-
QoS	Extensive (20+)	Limited	Limited	Limited	-	-
INTEROPERABILITY LEVEL	Semantic	Structural	Semantic	Foundational	Semantic	Structural
SECURITY	TLS, DTLS, DDS Security	TLS + SASL	DTLS	TLS	HTTPS	TLS + SASL
DATA PRIORITIZATION	Transport Priorities	-	-	-	-	-
FAULT TOLERANCE	Decentralized	Implementation-Specific	Decentralized	Broker is SPoF	Server is SPoF	Server is SPoF

Comparisons of IoT Protocols

Features	MQTT (MQTT-SN)	CoAP	XMPP	AMQP
Transport	TCP/IP UDP/IP (MQTT-SN)	UDP/IP	TCP/IP	TCP/IP
Communication model	Publish/subscribe	Request/Response	Request/Response Publish/subscribe	Publish/subscribe
Security	Medium-optional	Medium-optional	High-Mandatory	High-Mandatory
QoS Level	High	Moderate	Moderate	High
Header size	2 bytes	4 bytes	-	8 bytes
Constrained devices	Yes	Yes	No	No
Power consumption	Less	Medium	High	Medium
Assured delivery level	High	Medium	-	High

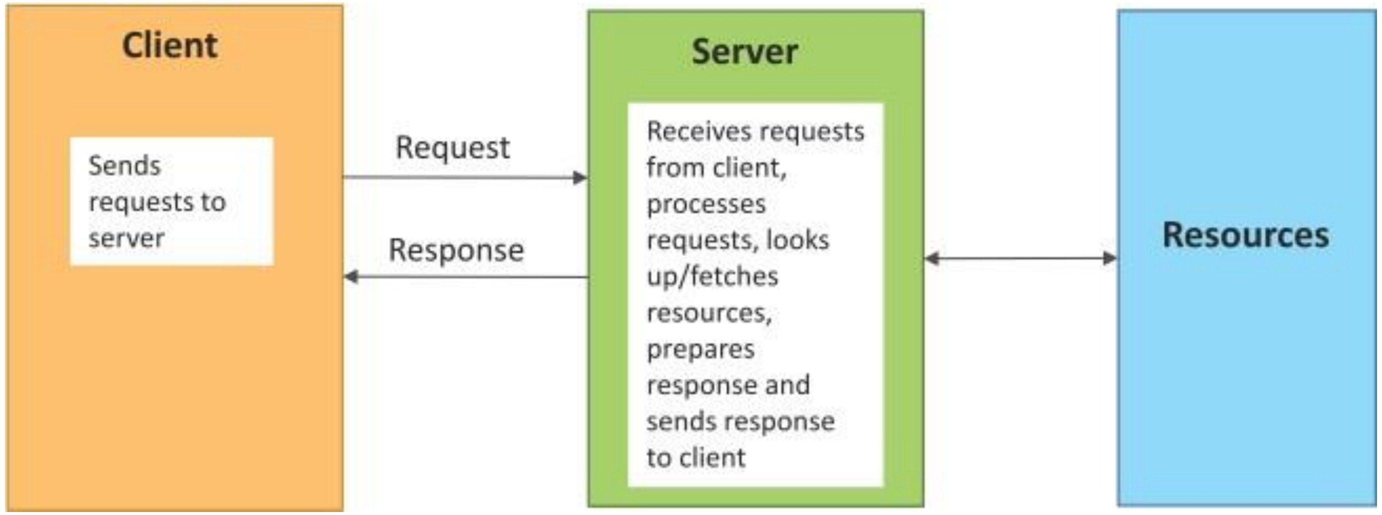
Logical Design of IoT

- Device : Devices such as sensing, actuation, monitoring and control functions.
- Communication : IoT Protocols
- Services like device monitoring, device control services, data publishing services and device discovery
- Management : Functions to govern the system
- Security : Functions as authentication, authorization, message and content integrity, and data security
- Applications



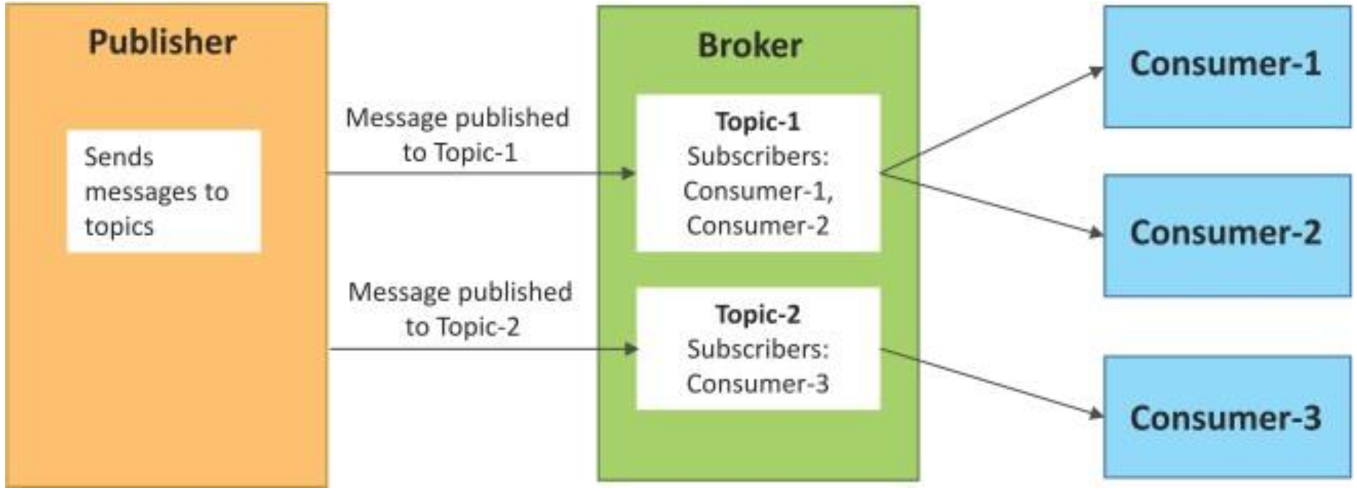
Request–Response Communication Model

- Request–Response is a communication model in which the client sends requests to the server and the server responds to the requests.
- When the server receives a request, it decides how to respond, fetches the data, retrieves resource representations, prepares the response and then sends the response to the client.
- Stateless communication model



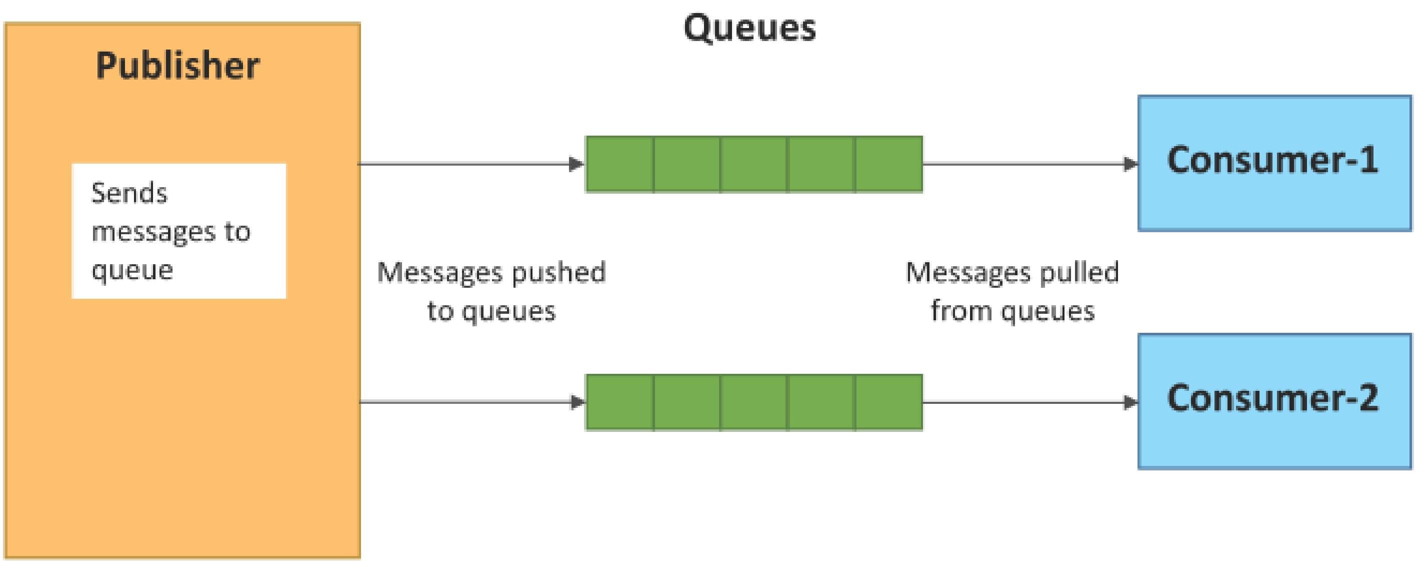
Publish–Subscribe Communication Model

- Publish–Subscribe is a communication model that involves publishers, brokers and consumers.
- Publishers are the source of data. Publishers send the data to the topics which are managed by the broker. Publishers are not aware of the consumers.
- Consumers subscribe to the topics which are managed by the broker.
- When the broker receives data for a topic from the publisher, it sends the data to all the subscribed consumers.



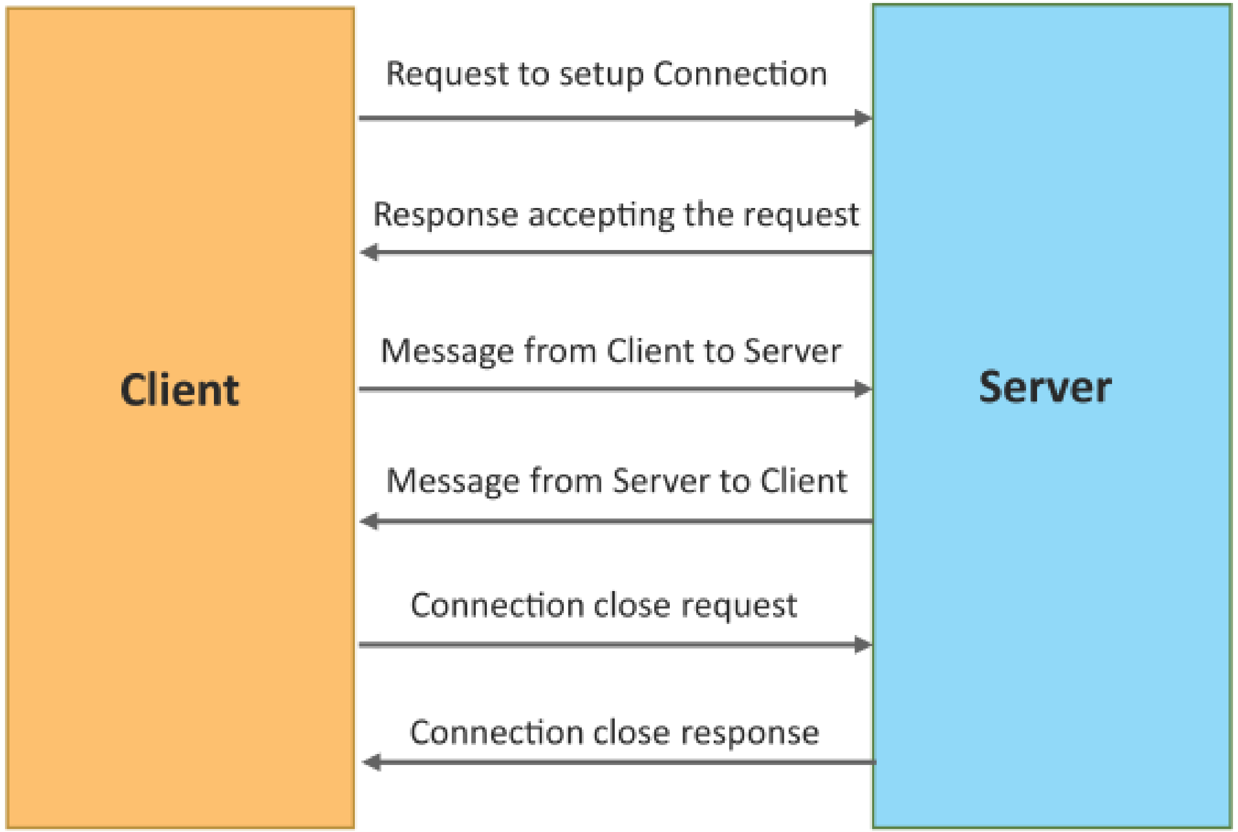
Push–Pull Communication Model

- Push–Pull is a communication model in which the data producers push the data to queues and the consumers pull the data from the queues. Producers do not need to be aware of the consumers.
- Queues help in decoupling the messaging between the producers and consumers.
- Queues also act as a buffer which helps in situations when there is a mismatch between the rate at which the producers push data and the rate at which the consumers pull data.



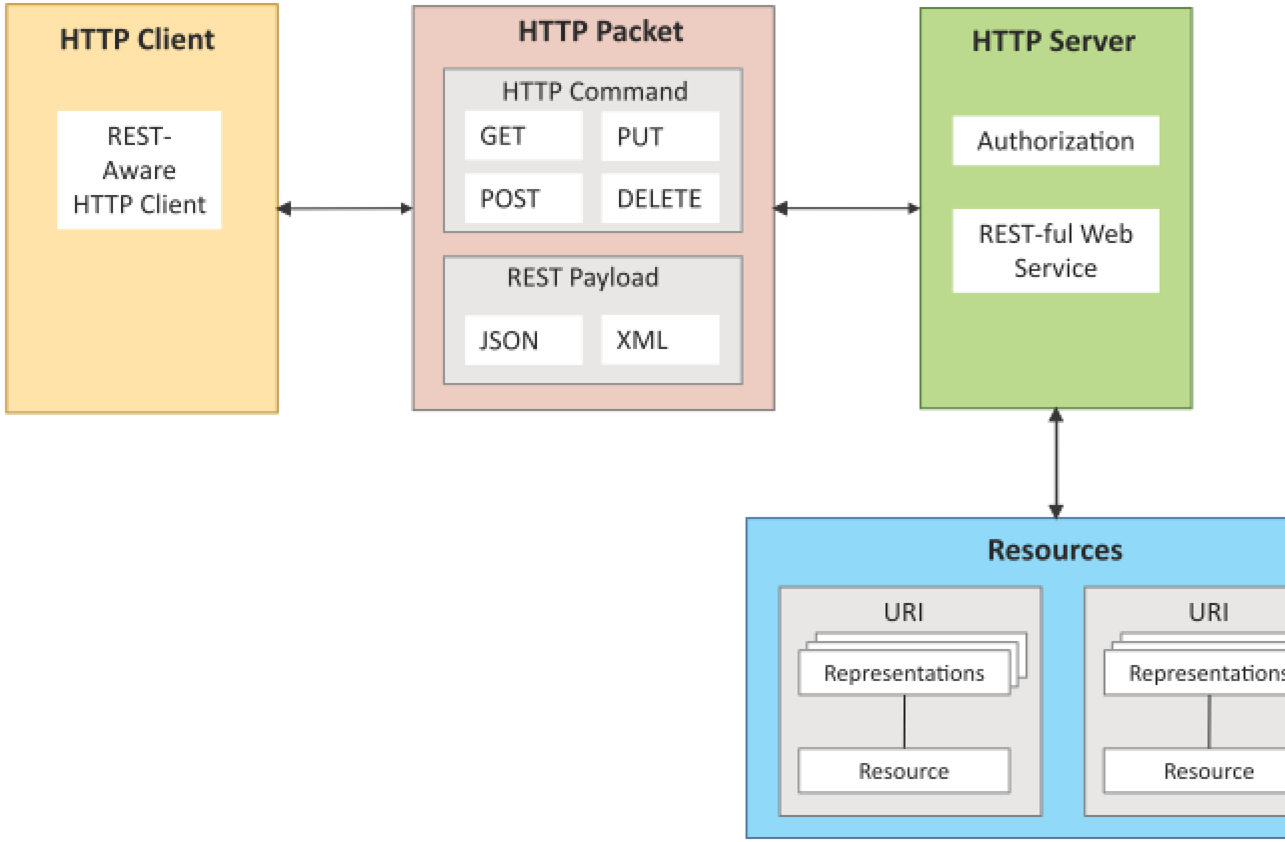
Exclusive Pair Communication Model

- Exclusive Pair is a bidirectional, fully duplex communication model that uses a persistent connection between the client and the server.
- Once the connection is set up it, remains open until the client sends a request to close the connection.
- Client and server can send messages to each other after connection setup.



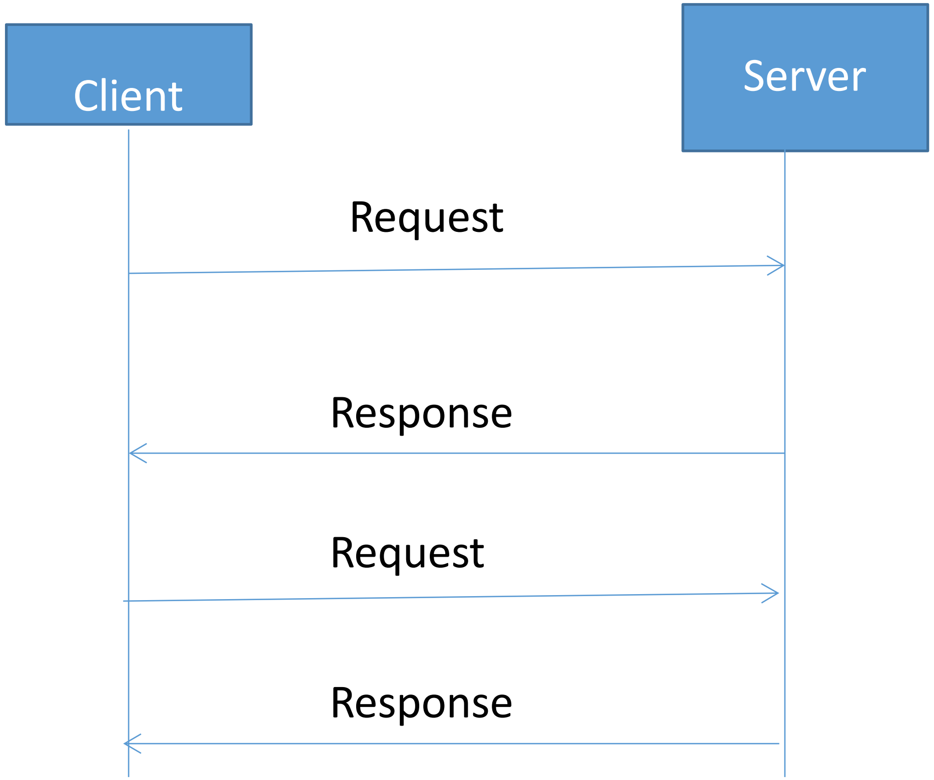
REST-based Communication APIs

- Representational State Transfer (REST) is a set of architectural principles by which you can **design web services and web APIs** that focus on a system's resources and how resource states are addressed and transferred.
- REST APIs **follow the request–response communication model.**
- REST architectural constraints apply to the components, connectors and data elements within a distributed



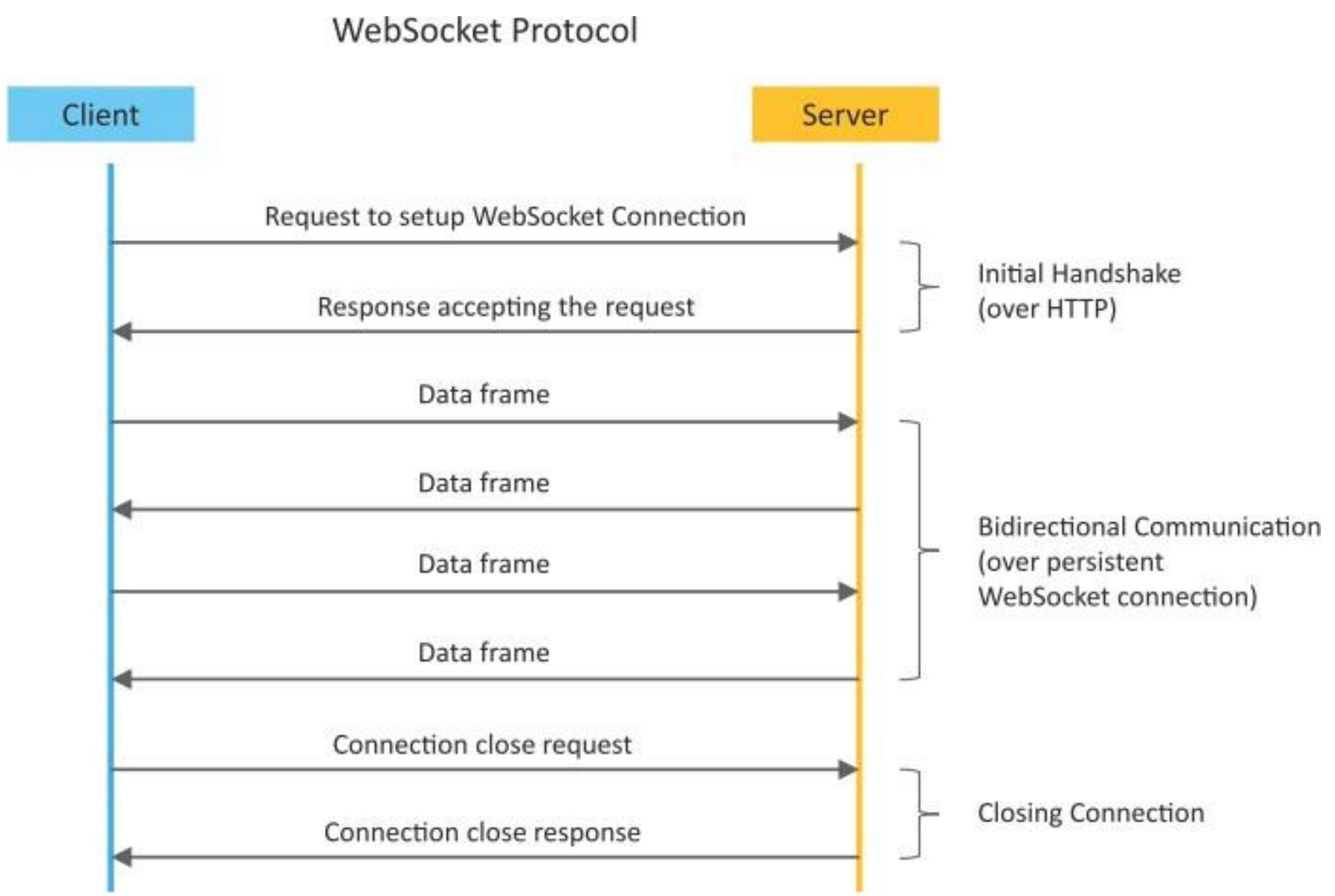
REST-based Communication APIs Constraints

- **Client – Server**
- **Stateless**
- **Cacheable**
- **Layered System**
- **Uniform Interface**
- **Code on demand**



WebSocket-based Communication APIs

- WebSocket APIs **allow bi-directional, full duplex communication** between clients and servers.
- WebSocket APIs follow the **exclusive pair communication model**.



Difference between REST and WebSocket-based Communication APIs

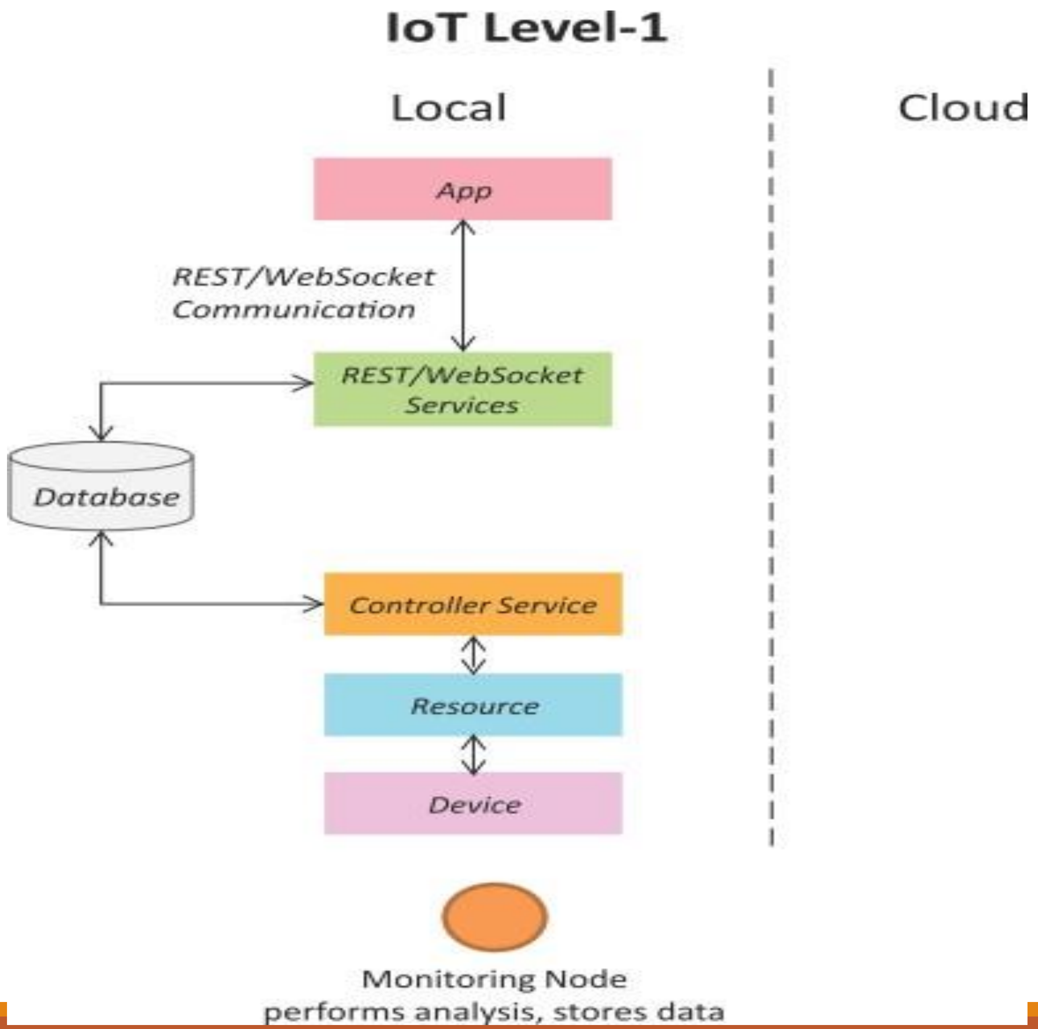
Comparison Based on	REST	Websocket
State	Stateless	Stateful
Directional	Unidirectional	Bidirectional
Req-Res/Full Duplex	Follow Request Response Model	Exclusive Pair Model
TCP Connections	Each HTTP request involves setting up a new TCP Connection	Involves a single TCP Connection for all requests
Header Overhead	Each request carries HTTP Headers, hence not suitable for real-time	Does not involve overhead of headers.
Scalability	Both horizontal and vertical are easier	Only Vertical is easier

IoT Levels and Deployment Templates

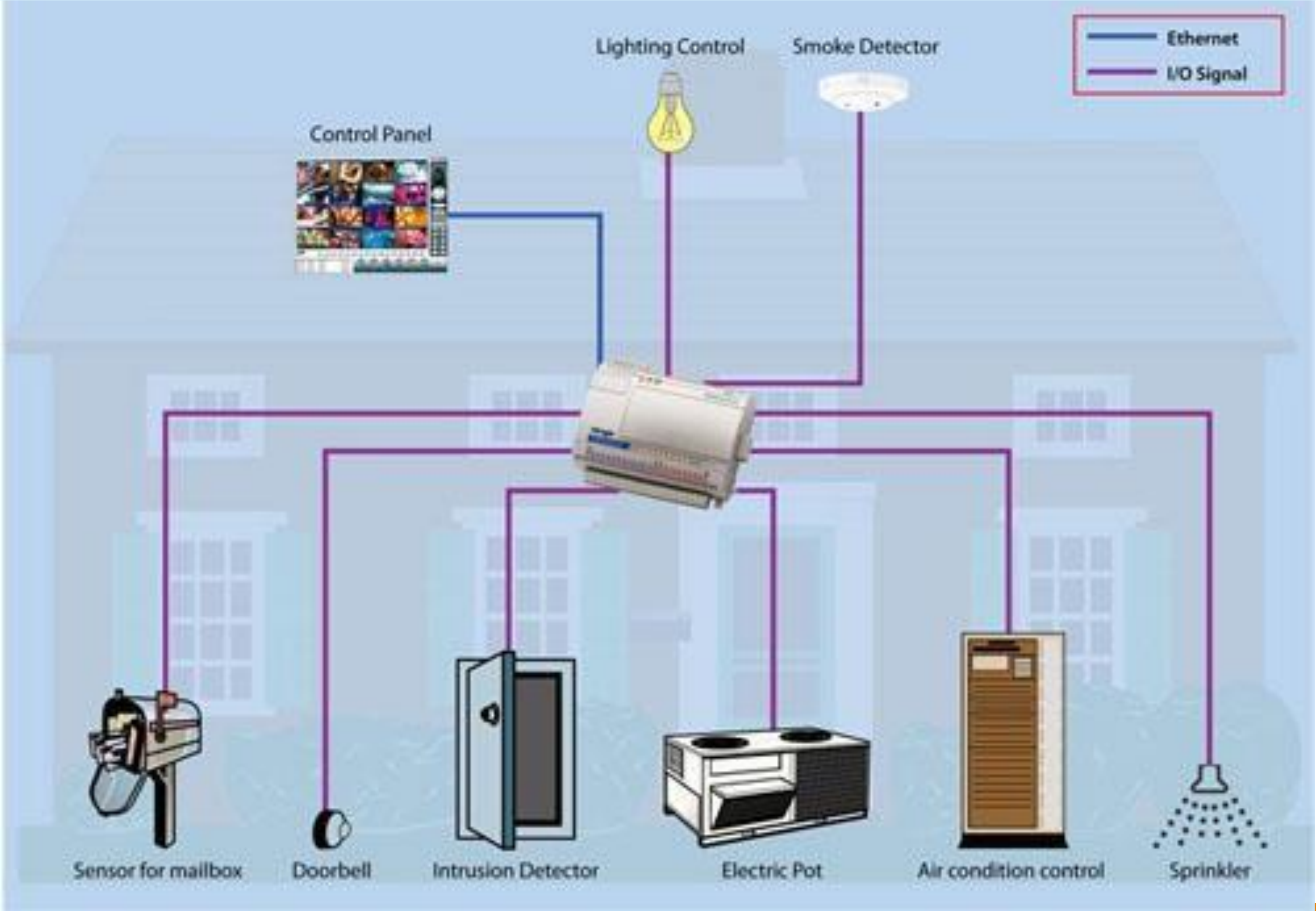
- **Database**: Database can be either local or in the cloud and stores the data generated by the IoT device.
- **Web Service**: Web services serve as a link between the IoT device, application, database and analysis components. Web service can be implemented using HTTP and REST principles (REST service) or using the WebSocket protocol (WebSocket service).
- **Analysis Component**: This is responsible for analyzing the IoT data and generating results in a form that is easy for the user to understand.
- **Application**: IoT applications provide an interface that the users can use to control and monitor various aspects of the IoT system. Applications also allow users to view the system status and the processed data.

IoT Level-1

- A level-1 IoT system **has a single node/device** that performs sensing and/or actuation, stores data, performs analysis and hosts the application.
- Level-1 IoT systems are suitable for **modelling low-cost and low-complexity solutions** where the data involved is not big and the **analysis requirements are not computationally intensive.**

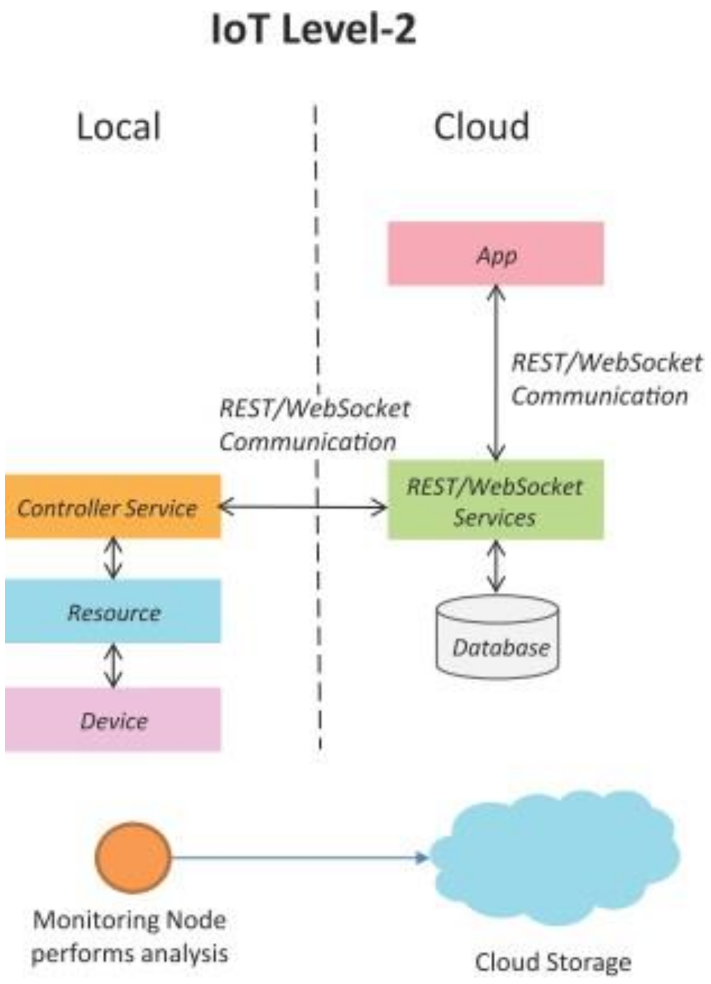


IoT – Level 1 Example ...Home Automation System



IoT Level-2

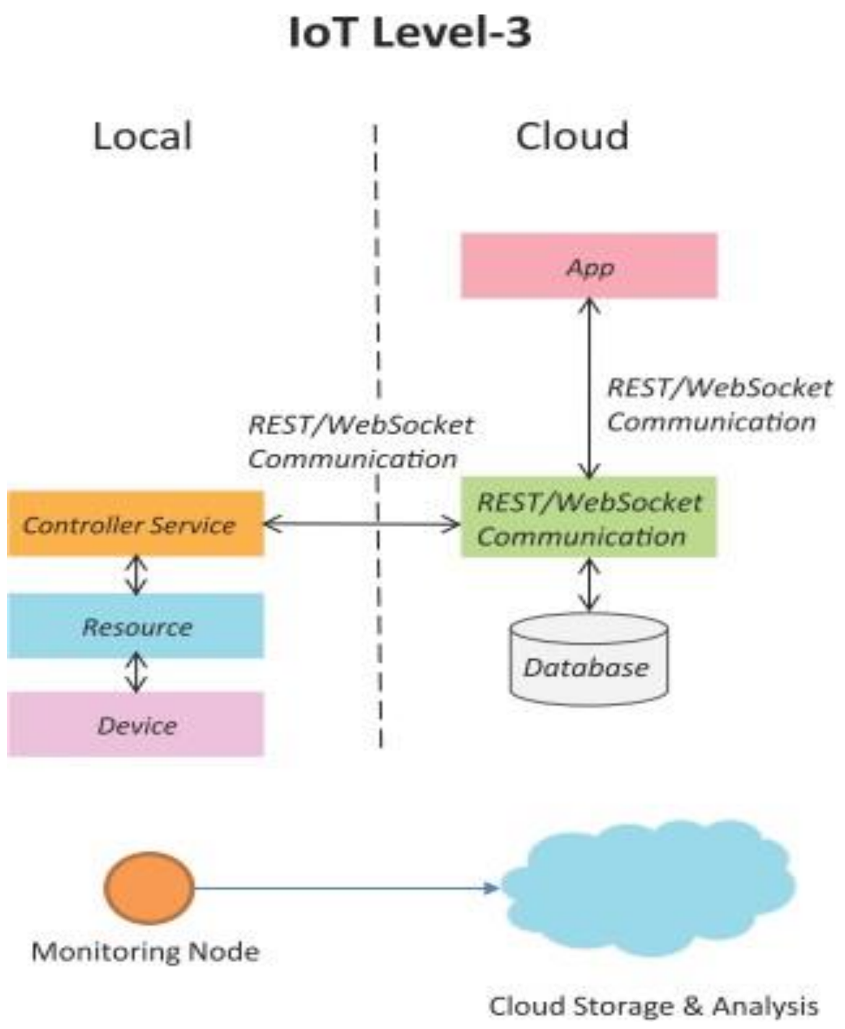
- A level-2 IoT system has a **single node that performs sensing and/or actuation and local analysis.**
- **Data is stored in the cloud** and the application is usually cloud-based.
- Level-2 IoT systems are **suitable for solutions where the data involved is big;** however, the primary **analysis requirement is not computationally intensive** and can be done locally.



IoT – Level 2 Example …Smart Irrigation

IoT Level-3

- A level-3 IoT system has a **single node. Data is stored and analyzed in the cloud** and the application is cloud-based.
- Level-3 IoT systems are suitable for solutions **where the data involved is big and the analysis requirements are computationally intensive.**

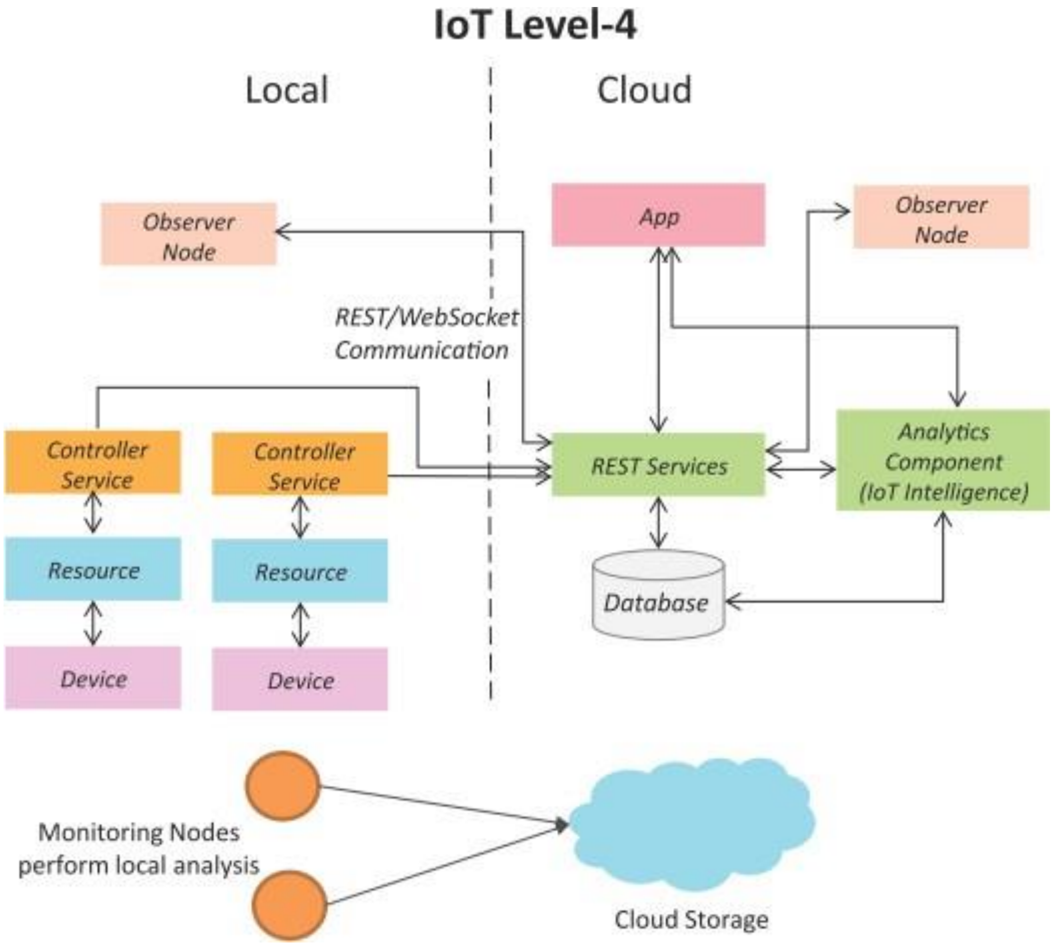


IoT – Level 3 Example …Tracking Package Handling

Sensors used accelrometer and gyroscope

IoT Level-4

- A level-4 IoT system has multiple nodes that perform local analysis. Data is stored in the cloud and the application is cloud-based.
- Level-4 contains local and cloud-based observer nodes which can subscribe to and receive information collected in the cloud from IoT devices.
- Level-4 IoT systems are suitable for solutions where multiple nodes are required, the data involved is big and the analysis requirements are computationally intensive.

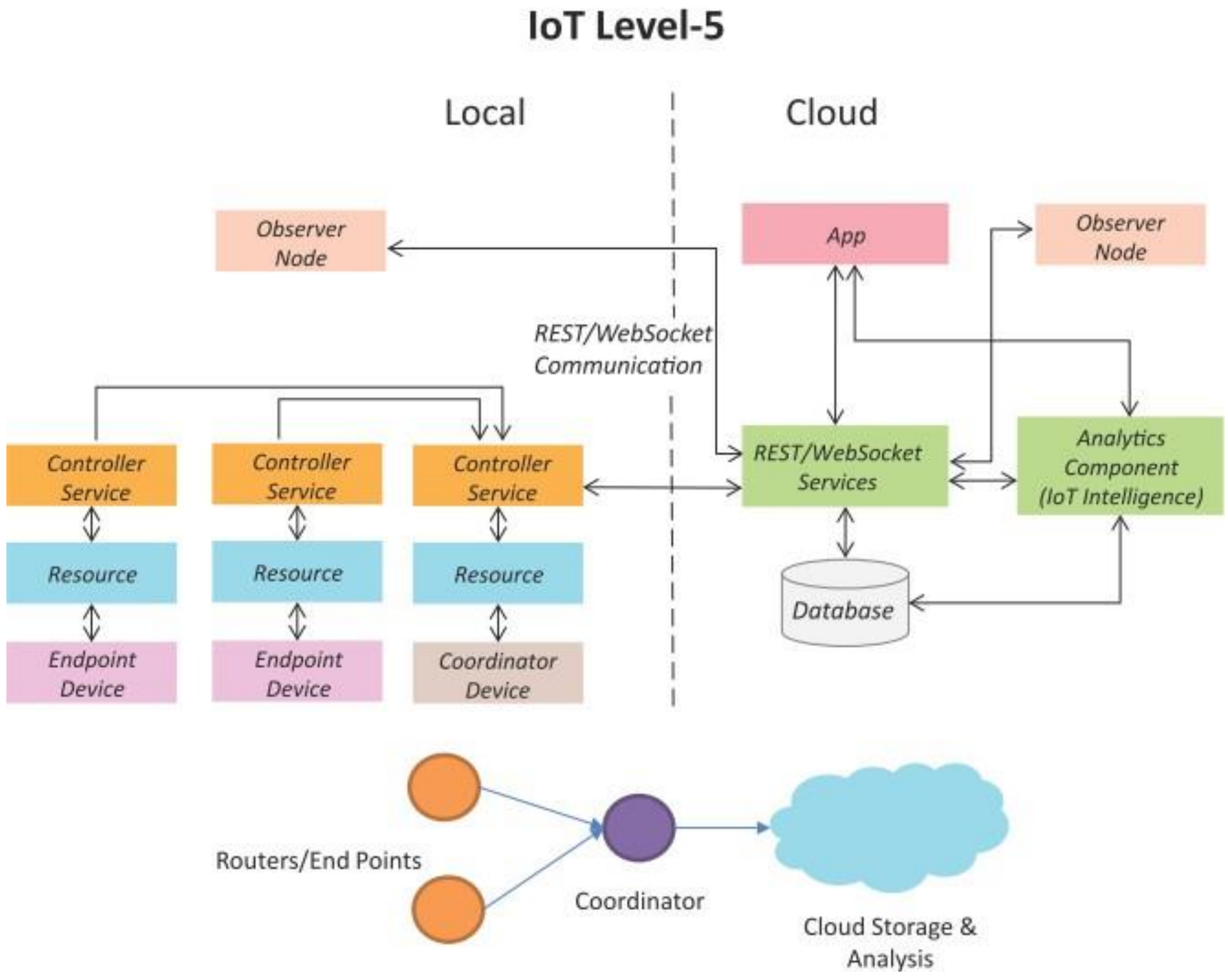


IoT – Level 4 Example …Noise Monitoring

Sound Sensors are used

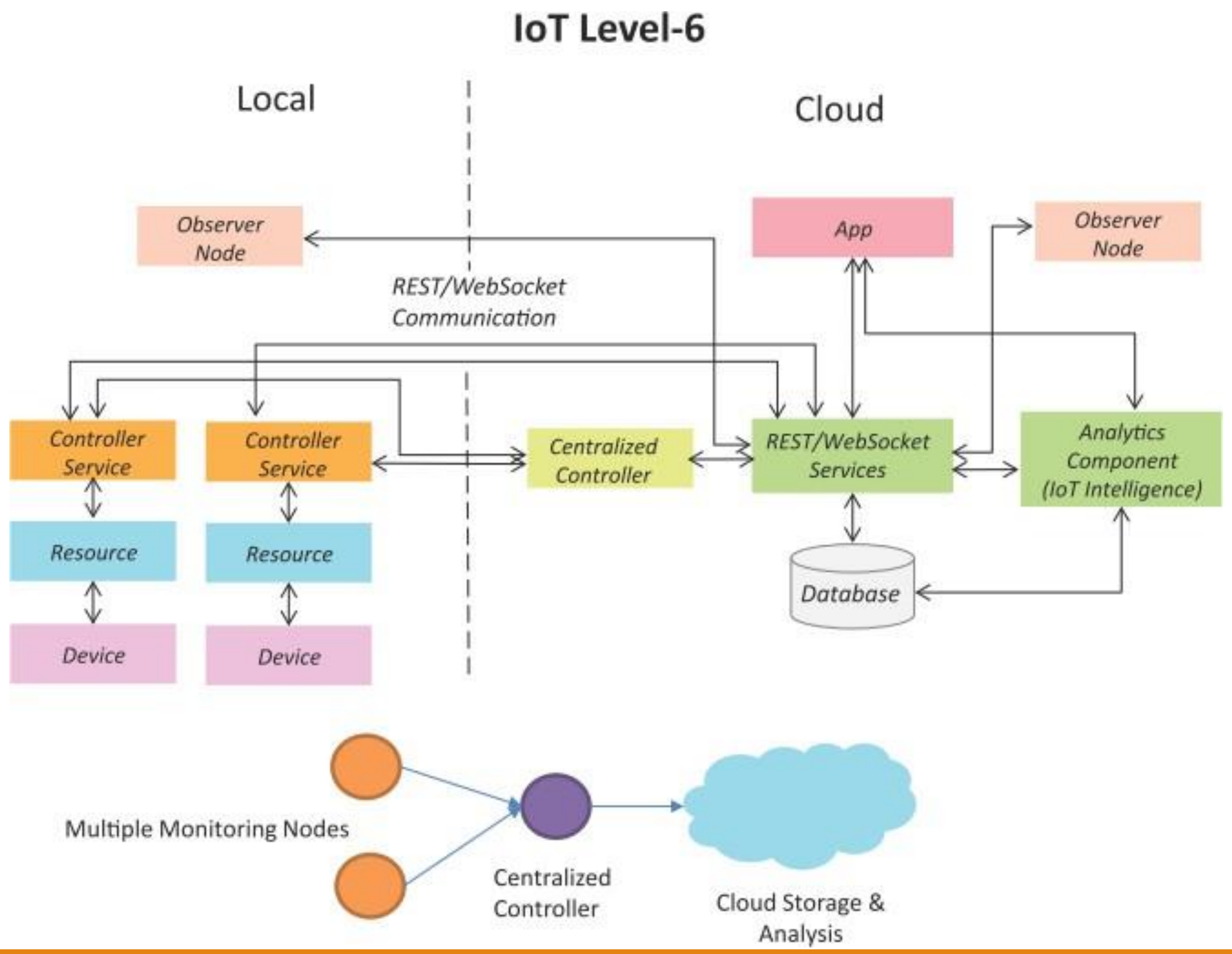
IoT Level-5

- A level-5 IoT system has multiple end nodes and one coordinator node.
- The end nodes perform sensing and/or actuation.
- The coordinator node collects data from the end nodes and sends it to the cloud.
- Data is stored and analyzed in the cloud and the application is cloud-based.
- Level-5 IoT systems are suitable for solutions based on wireless sensor networks, in which the data involved is big and the analysis requirements are computationally intensive.



IoT Level-6

- A level-6 IoT system has multiple independent end nodes that perform sensing and/or actuation and send data to the cloud.
- Data is stored in the cloud and the application is cloud-based.
- The analytics component analyzes the data and stores the results in the cloud database.
- The results are visualized with the cloud-based application.
- The centralized controller is aware of the status of all the end nodes and sends control commands to the nodes.



Thank you