Physical Design of IoT

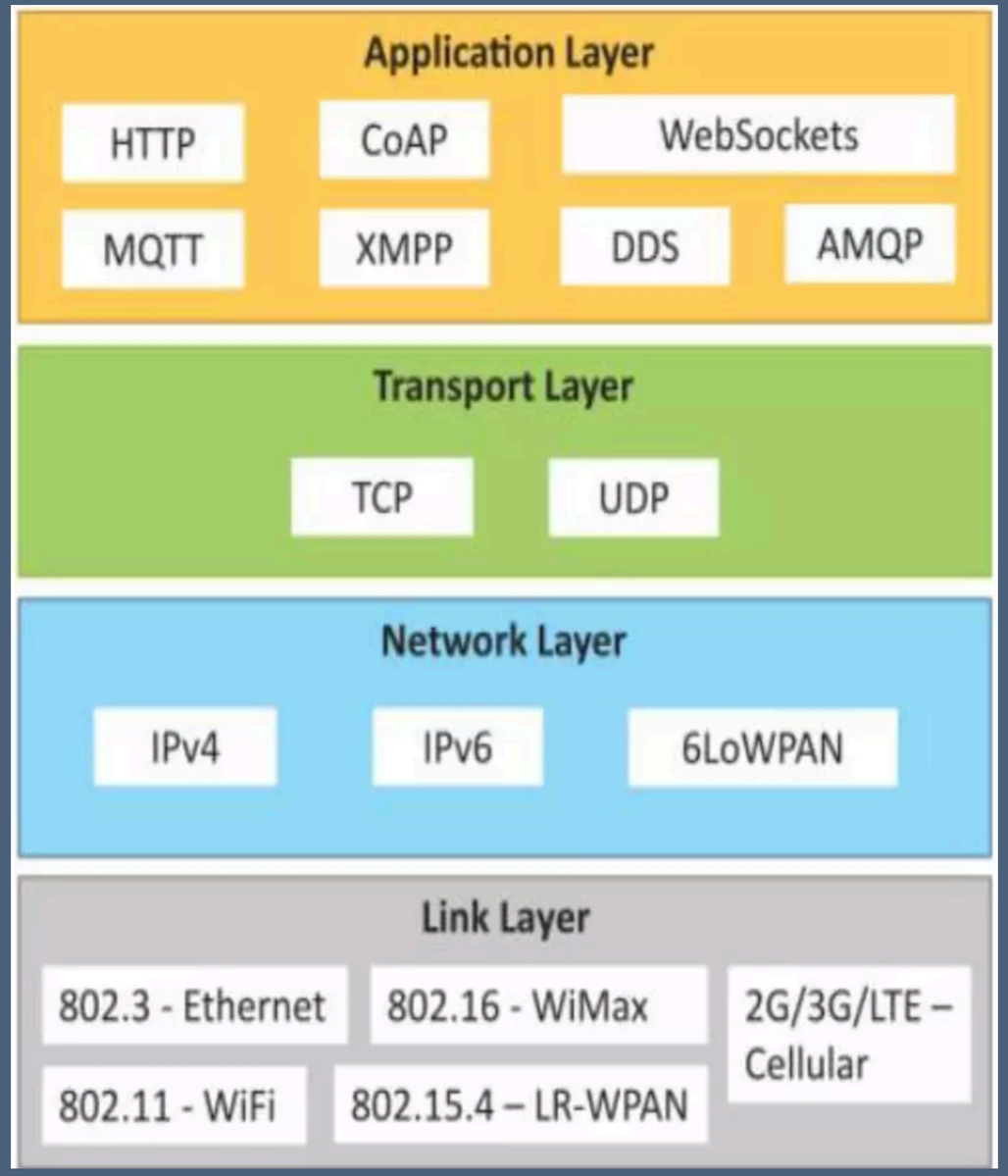
IoT Protocols

Raj Ramchandani

IoT Protocols

Link Layer

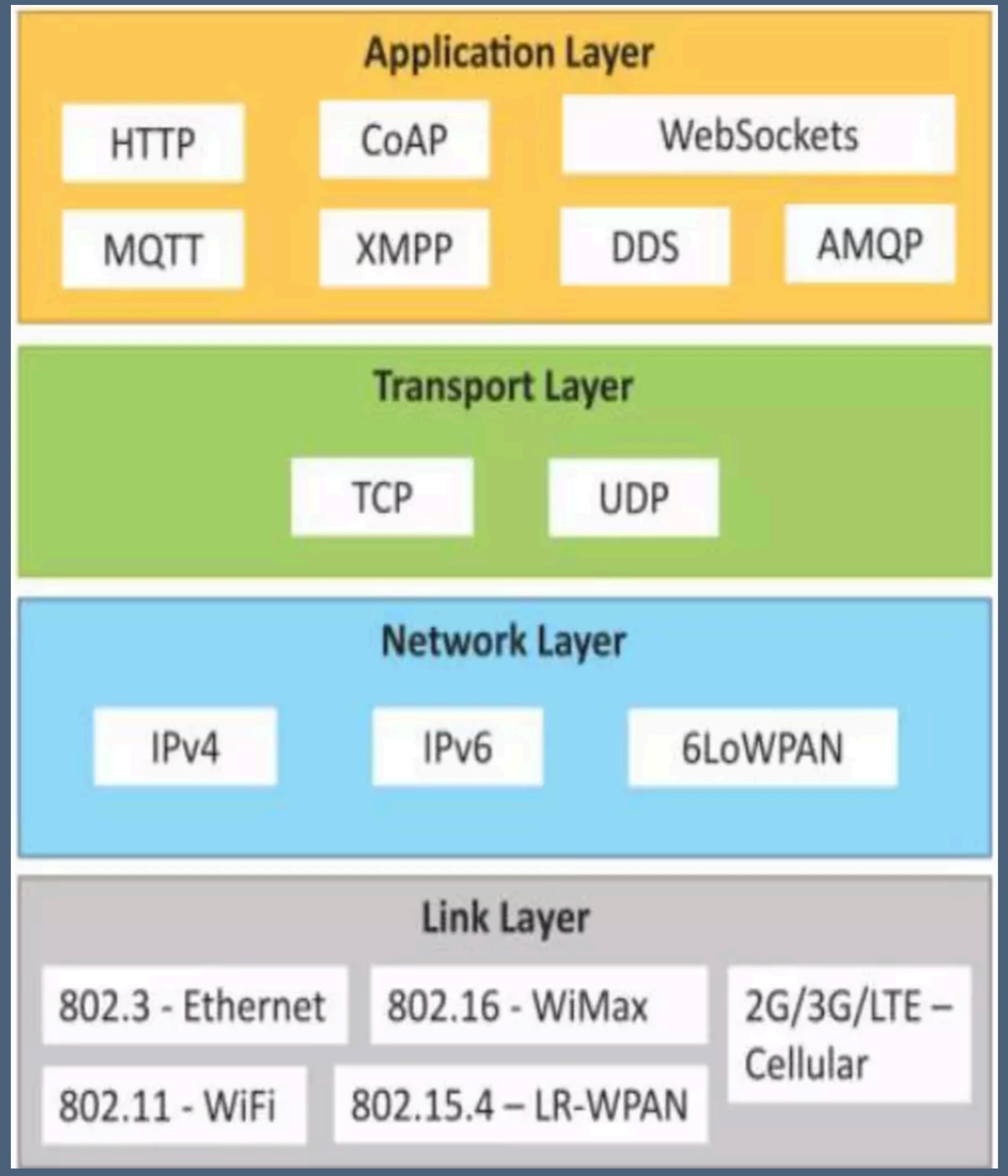
- These protocols determine how data is physically sent over the network
- It determines how the packets are coded and signalled by the hardware device over the medium to which host is attached



Link layer

802.3 Ethernet

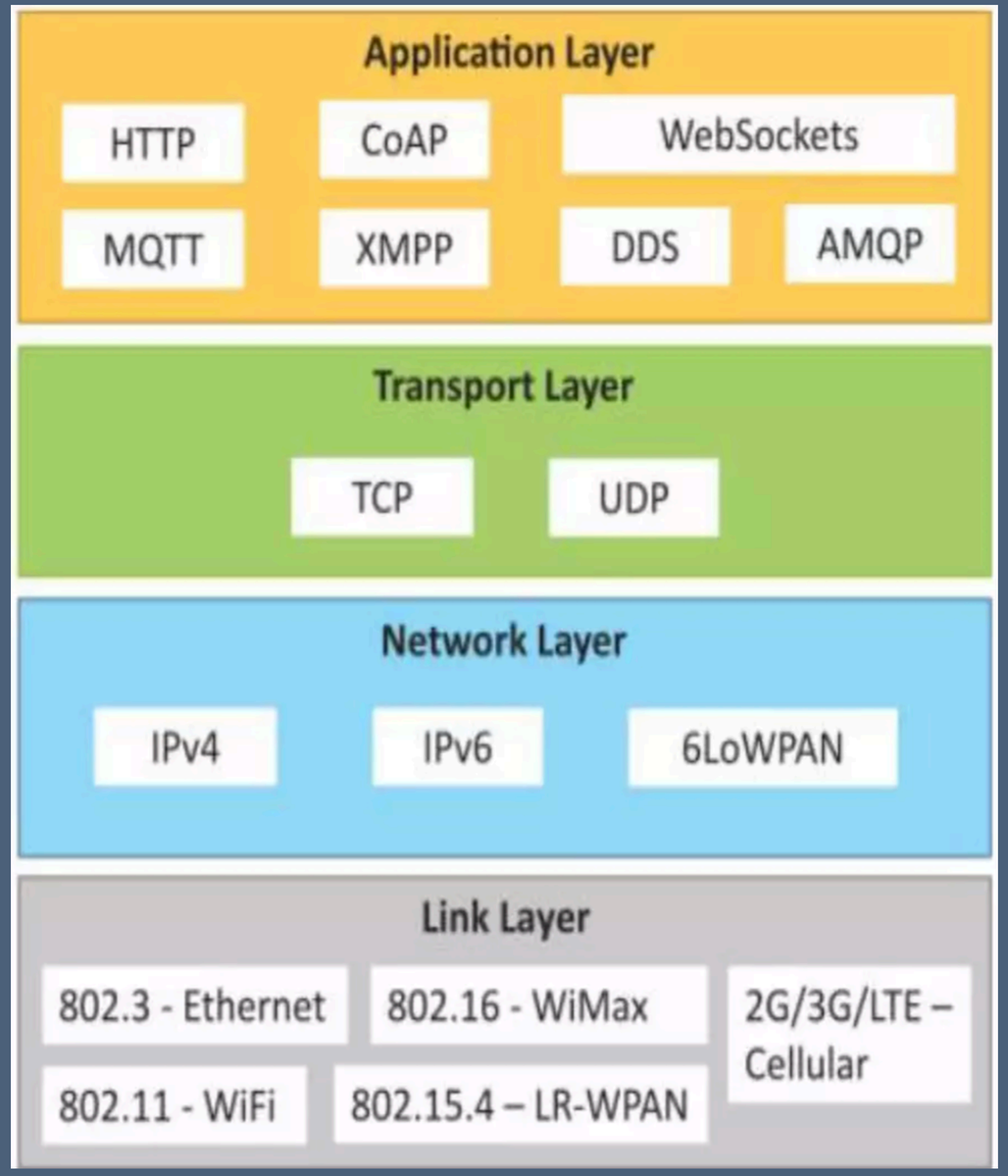
- Collection of wired ethernet standards in the link layer
- Example:
  802.3 is the standard for 10BASE5 Ethernet
  802.3.i is the standard for 10BASE-T Ethernet
  802.3.j is the standard for 10BASE-F Ethernet
  802.3ae is the standard for 10 Gbit/s Ethernet over fiber.
- These standards provide data rates from 10 Mb/s to 40 Gb/s and higher.



Link layer

802.11 WiFi

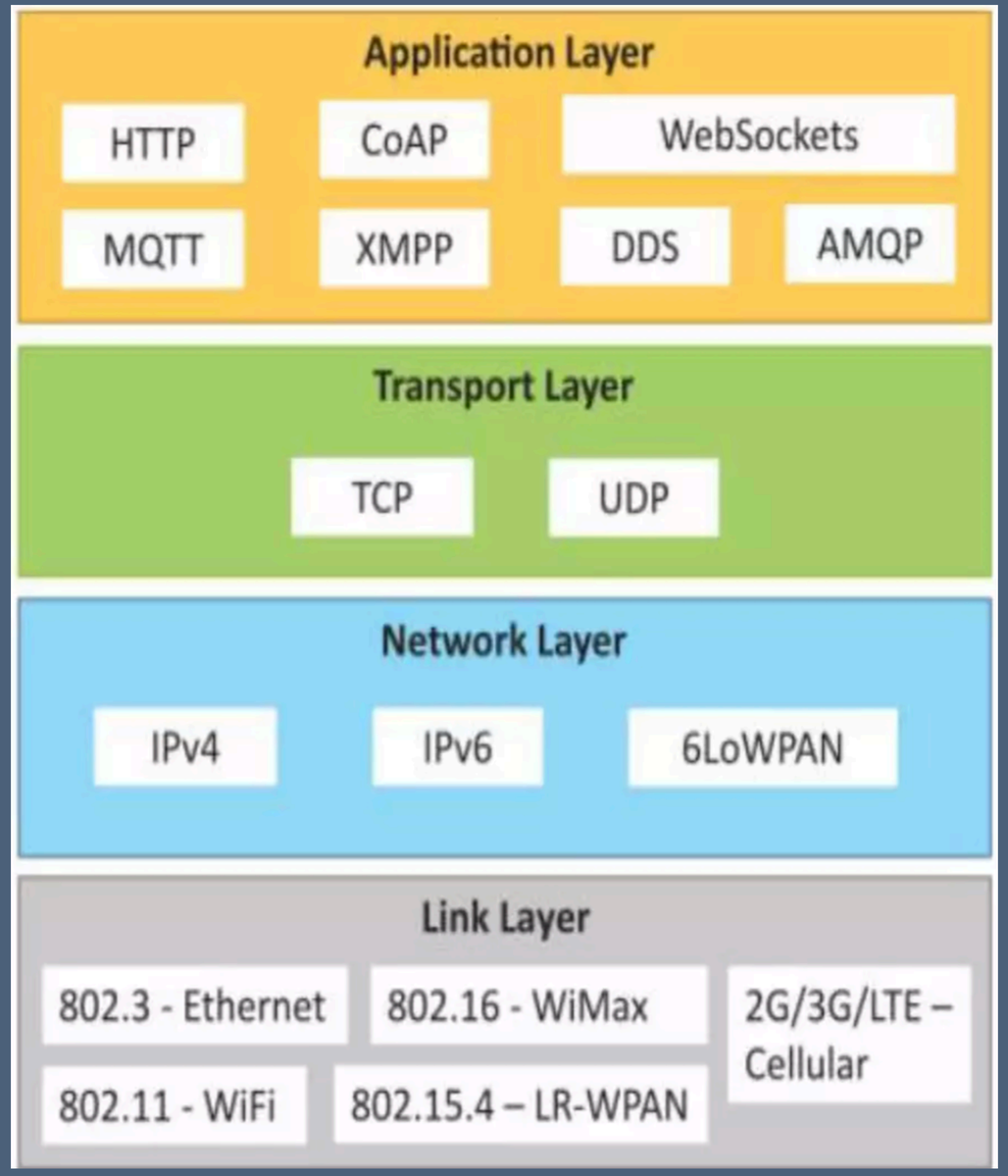
- 802.11 is a collection of wireless local area network (WLAN) communication standards
- Examples:
- 802.11a operates in the 5 GHz band,
- 802.1 lb and 802.11g operate in the 2.4 GHz band,
- 802.11n operates in the 2.4/5 GHz bands,
- 802.11ac operates in the 5GHz band
- 802.11ad operates in the 60 GHz band.



Link layer

802.16 WiMax

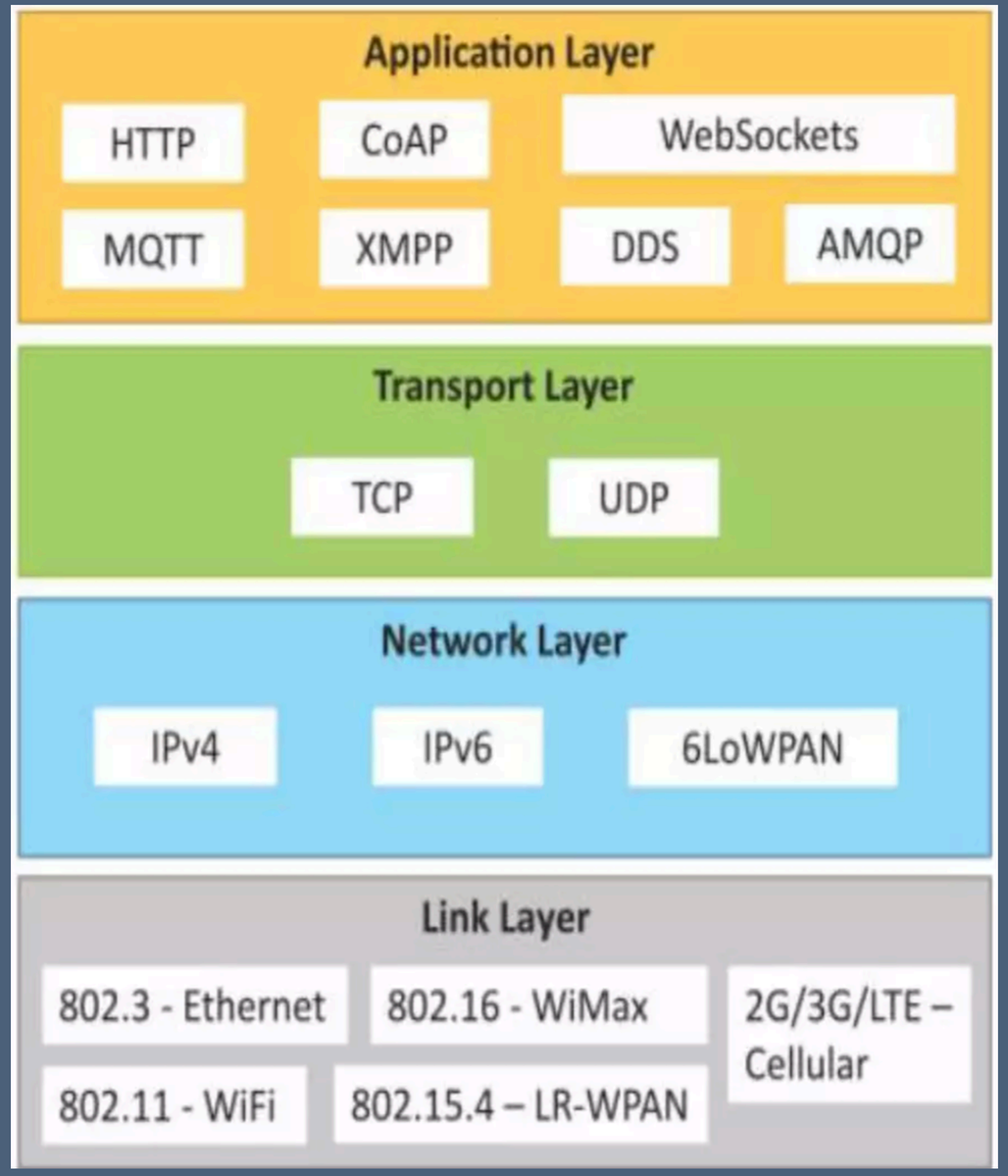
- 802.16 is a collection of wireless broadband standards
- WiMax standards provide data rates from 1.5 Mb/s to 1 Gb/s.
- The recent update (802.16m)provides data rates of 100 Mbit/s for mobile stations and 1 Gbit/s for fixed stations.



Link layer

802.15.4 LR-WPAN

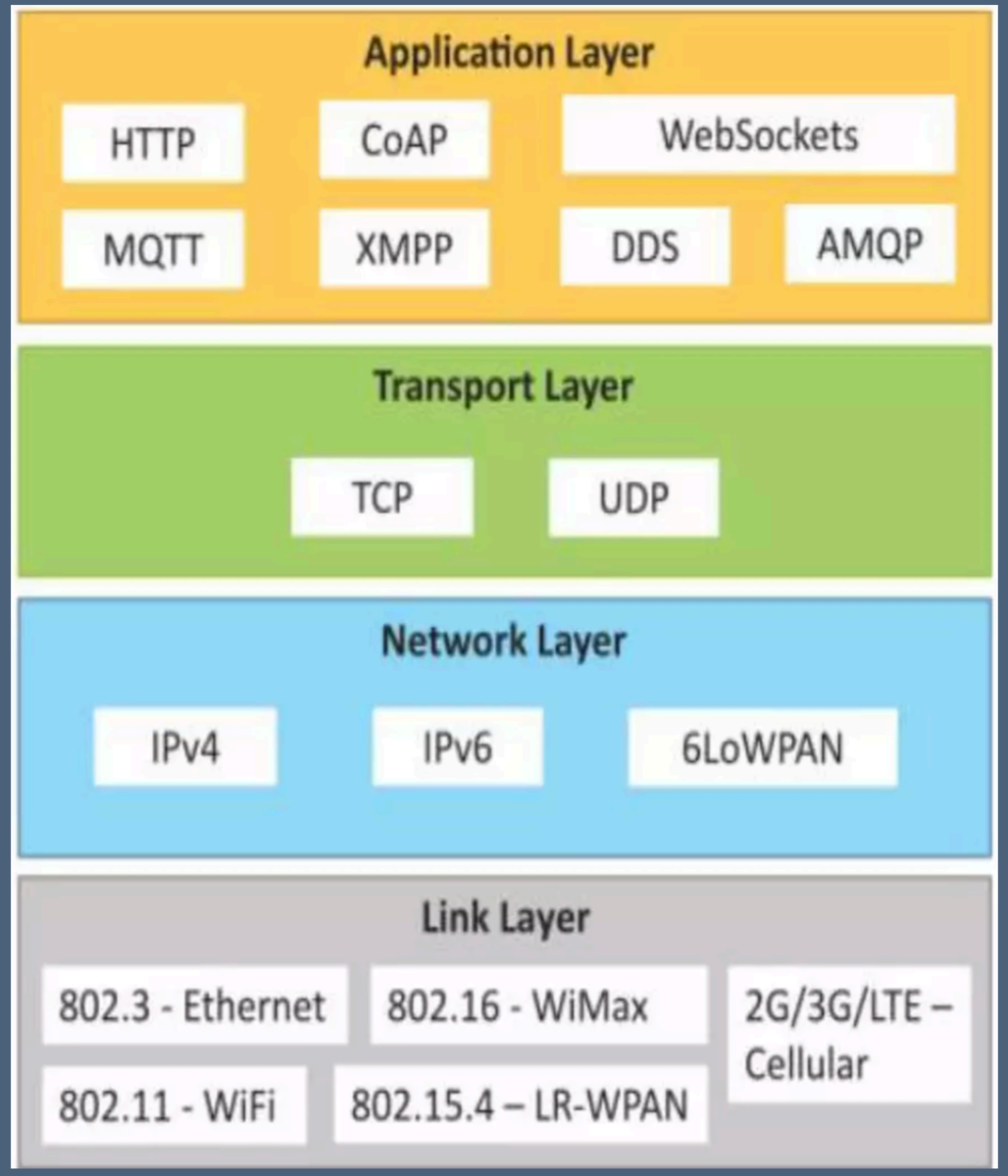
- 802.15.4 is a collection of standards for low-rate wireless personal area networks (LR-WPANs).
- These standards form the basis of specifications for high level communication protocols such as ZigBee.
- data rates from 40 Kb/s 250 Kb/s.
- low-costand low-speed communication for power constrained devices



Link layer

2G/ 3G/ 4G - Mobile Communication

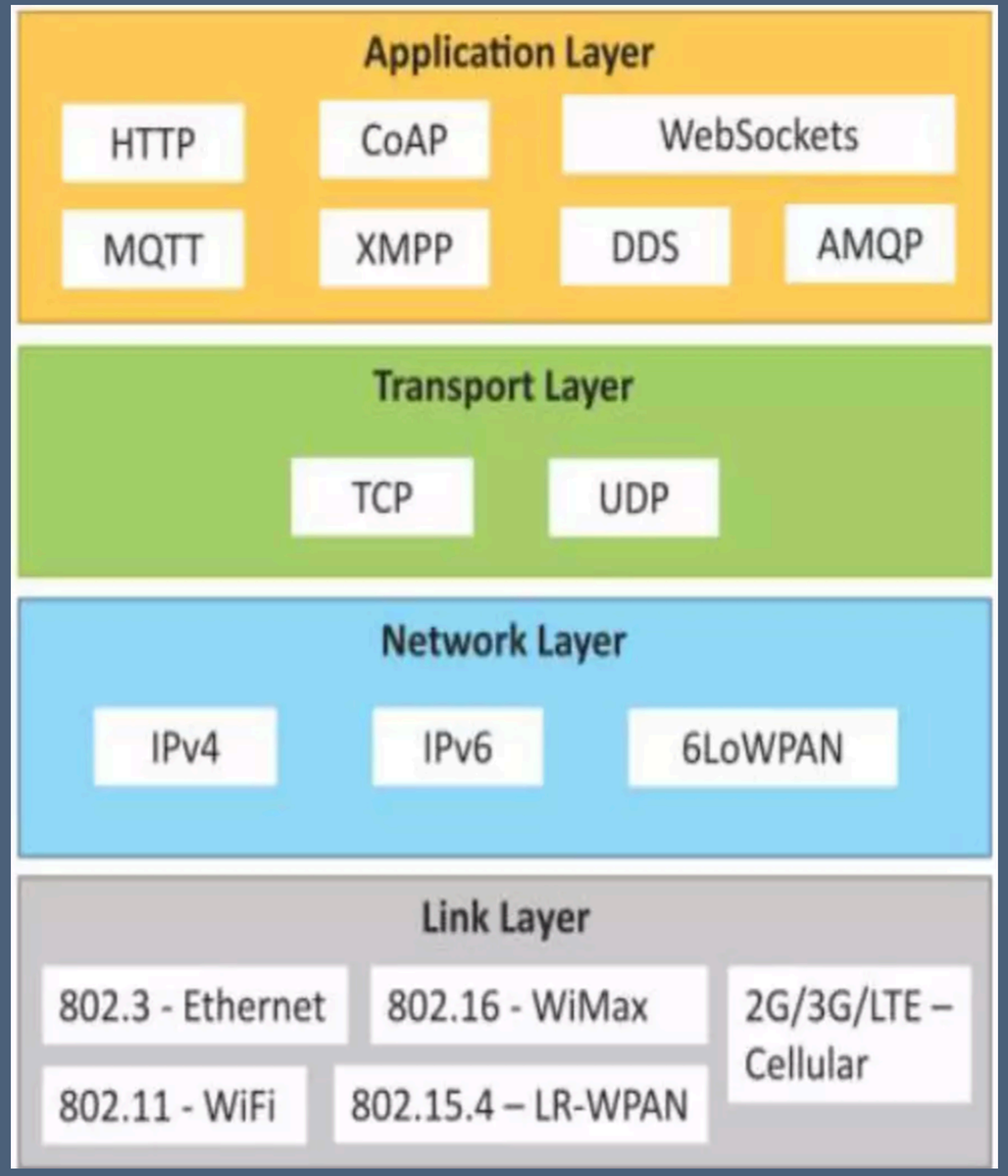
- There are different generations of mobile communication standards
- Includes second generation (2G including GSM and CDMA)
- third generation (3G - including UMTS and CDMA2000)
- fourth generation(4G - including LTE).
- Data rates for these standards range from 9.6 Kb/s (for 2G) to upto 100 Mb/s (for 4G)



IoT Protocols

Network / Internet Layer

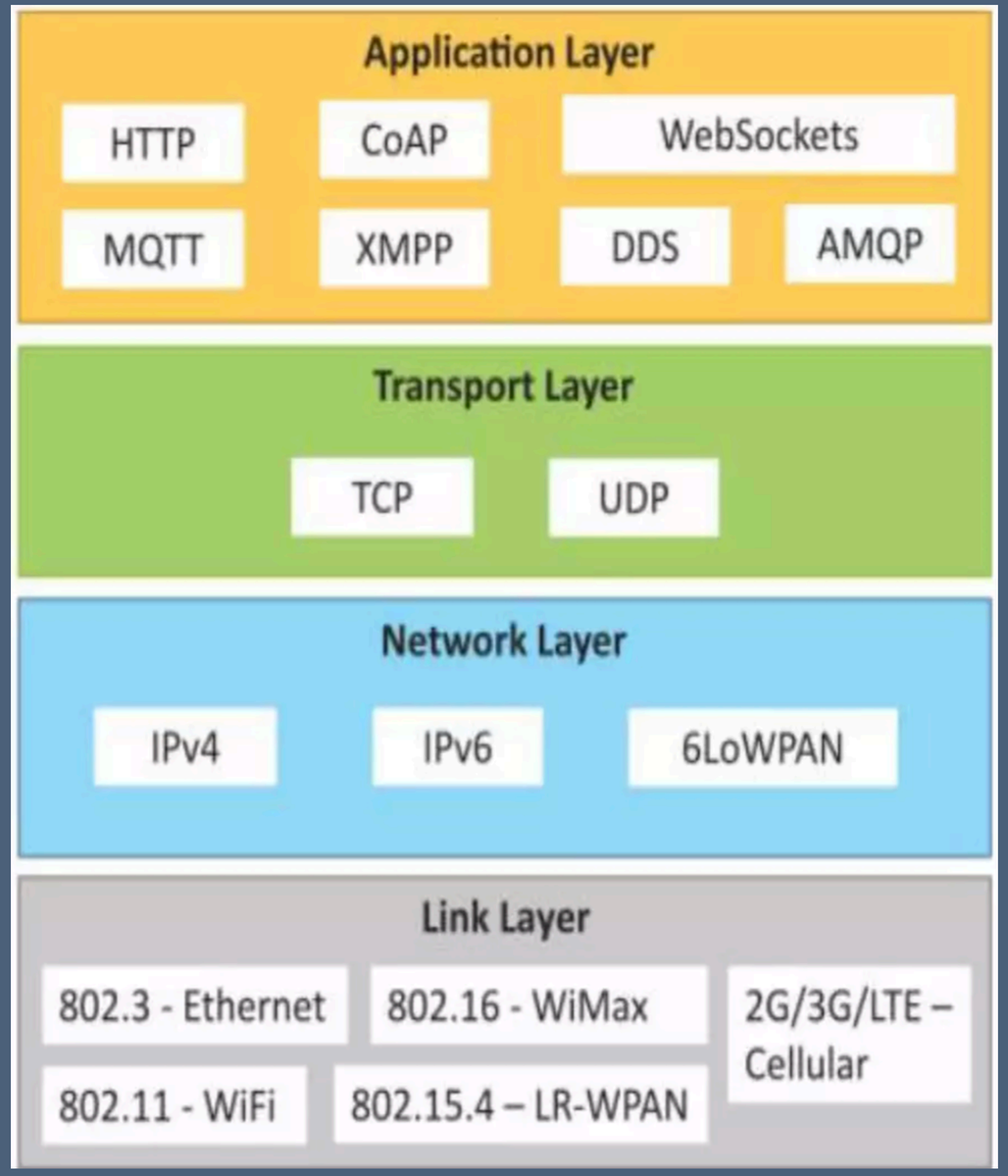
- The network layers are responsible for sending of IP datagrams from the source network to the destination network.
- This layer performs the host addressing and packet routing.
- The datagrams contain the source and destination addresses which are used to route them from the source to destination across multiple networks.
- identification is done using hierarchical IP addressing schemes such as IPv4 or IPv.



Network / Internet Layer

IPv4

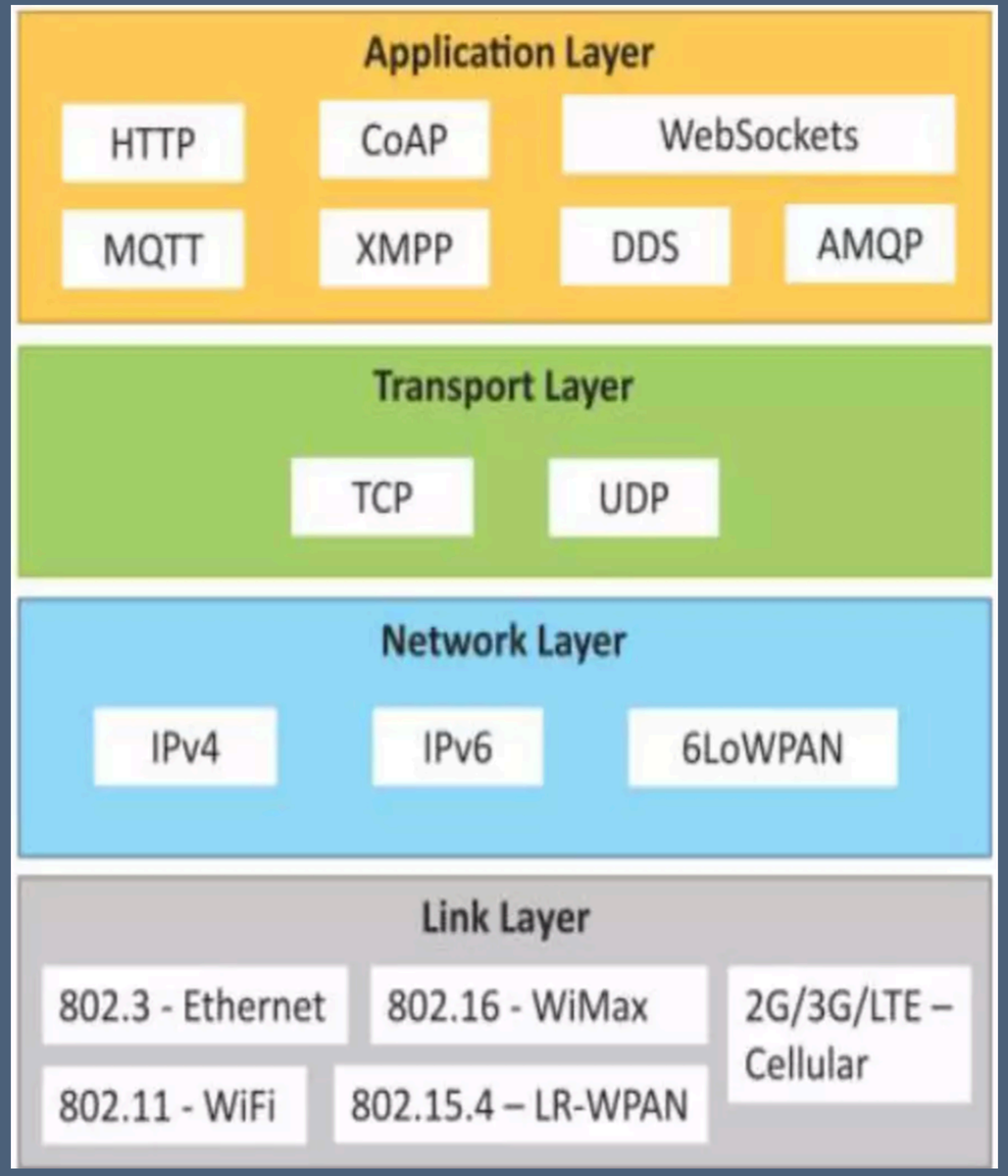
- most deployed Internet protocol
- used to identify the devices on a network using a hierarchical addressing scheme.
- IPv4 uses a 32-bit address scheme that allows total of 232 or 4,294,967,296 addresses.
- As more and more devices got connected to the Internet, these addresses got exhausted in the year 2011.
- The IP protocols establish connections on packet networks, but do not guarantee delivery of packets.



Network / Internet Layer

IPv6

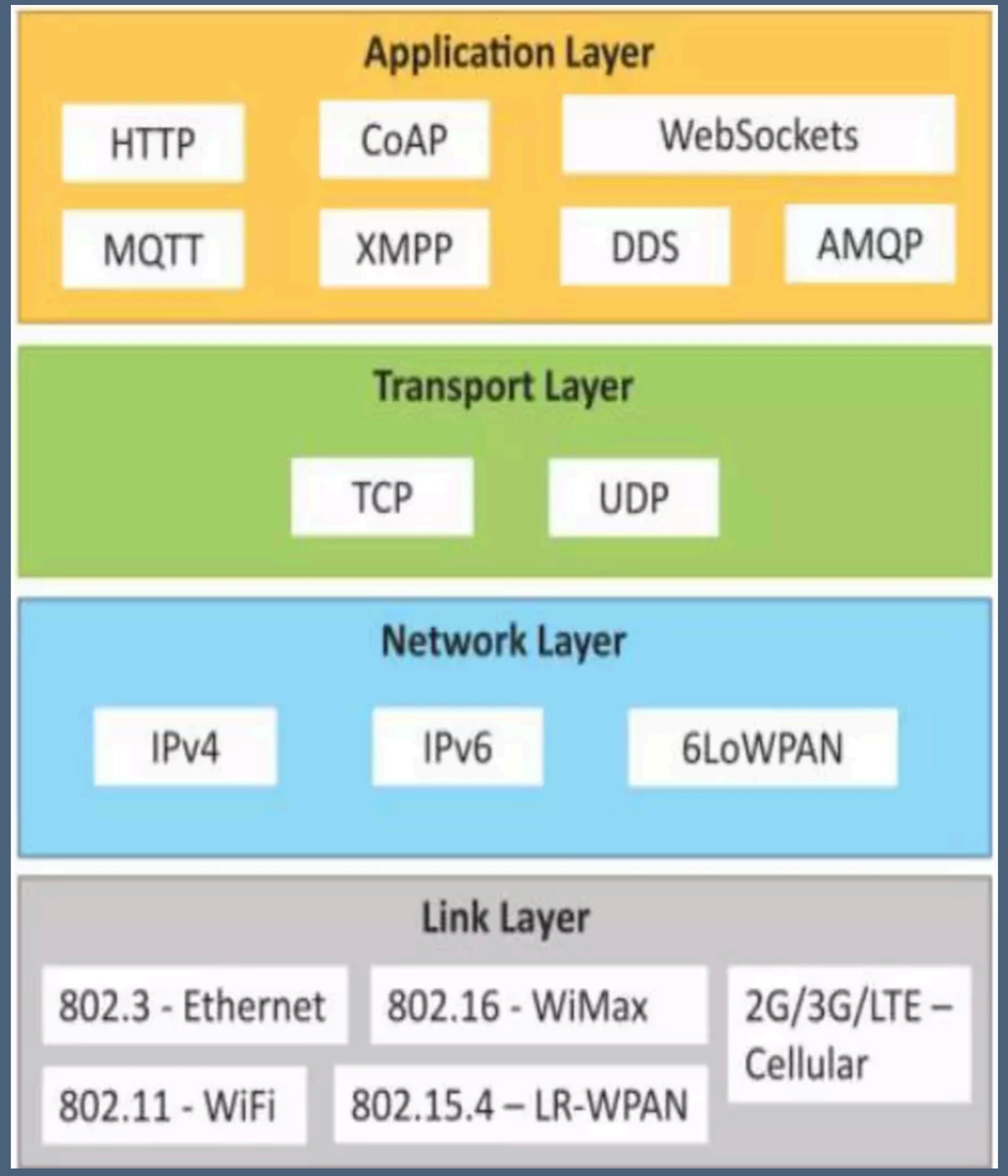
- Internet Protocol version 6 (IPv6) is the newest version of Internet protocol and successor to IPv4.
- Pv6 uses 128-bit address scheme that allows total of 2^128 or 3.4 × 10^38 addresses
- Sample address: 2001:0db8:0001:0000:0000:0ab9:C0A8:0102



Network / Internet Layer

6LoWPAN

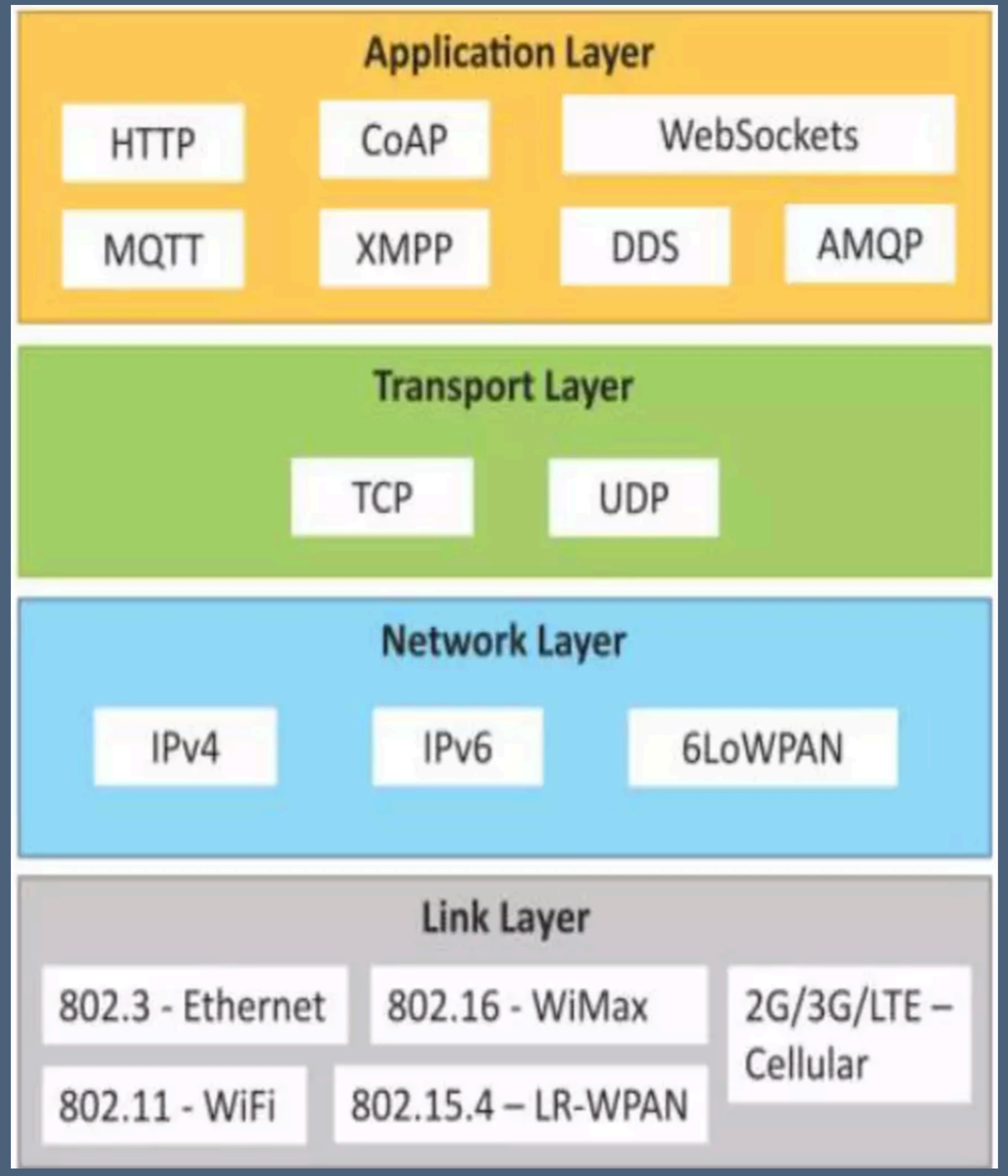
- IPv6 over Low power Wireless Personal Area Networks
- brings IP protocol to the low-power devices which have limited processing capability.
- 6LoWPAN operates in the 2.4 GHz frequency range and provides data transfer rates of 250 Kb/s.
- 6LoWPAN works with the 802.15.4 link layer protocol and defines compression mechanisms for IPv6 datagrams



IoT Protocols

Transport Layer

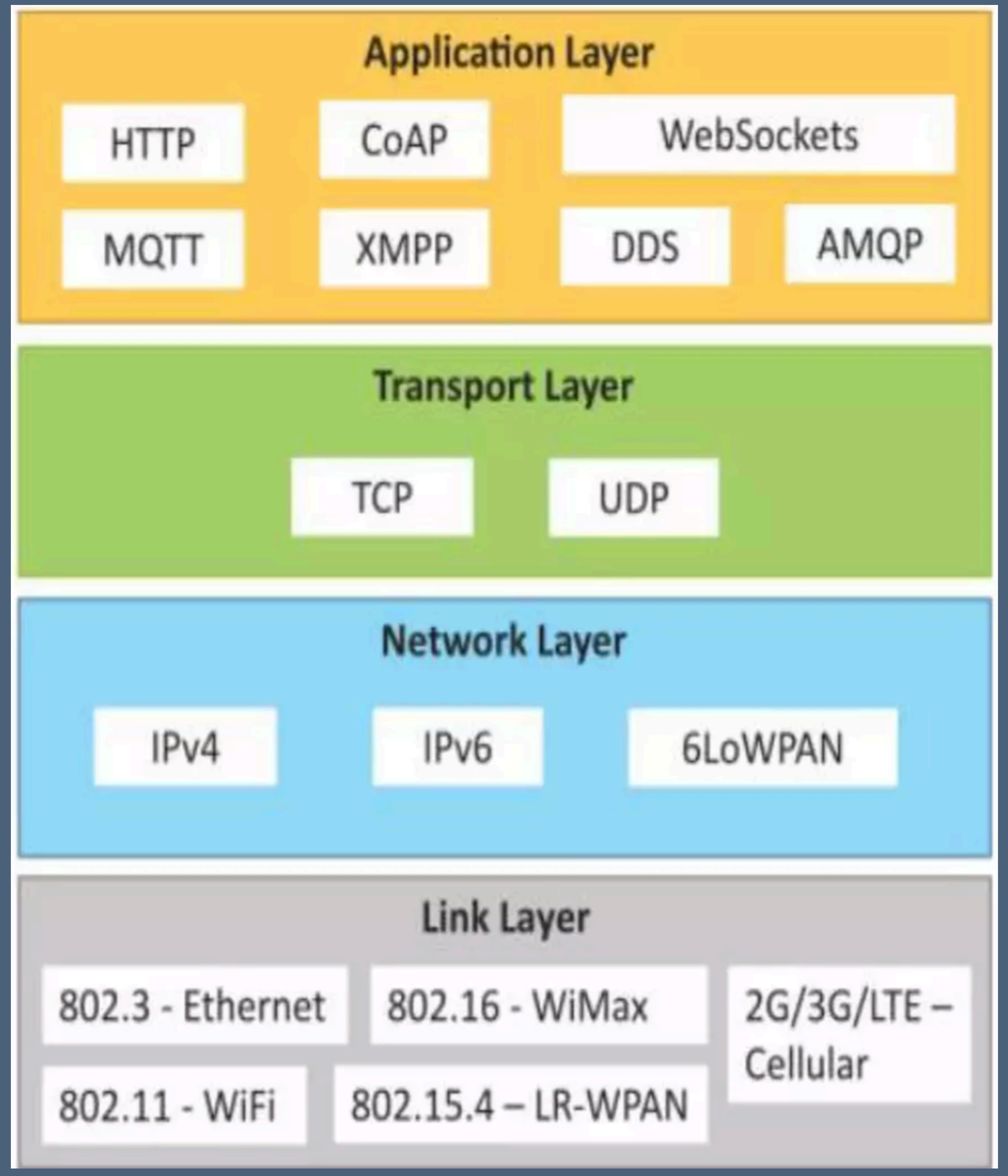
- The transport layer protocols provide end-to-end message transfer capability independent of the underlying network.
- The message transfer capability can be set up on connections, either using handshakes (as in TCP)
- without handshakes/acknowledgements (as in UDP)
- provides functions such as error control, segmentation, flow control and congestion control.



Transport Layer

Transmission control Protocol (TCP)

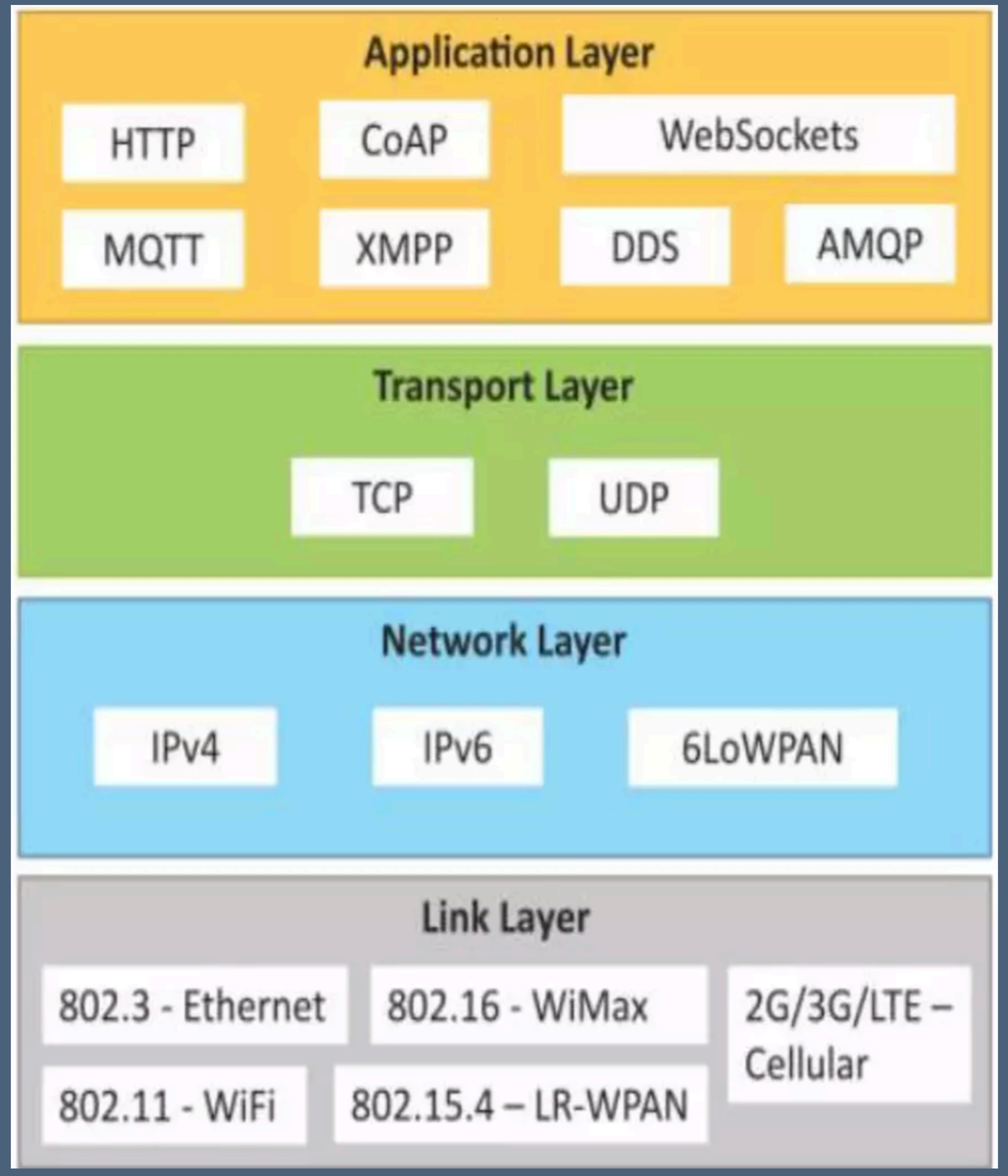
- Transmission Control Protocol (TCP) is the most widely used transport layer protocol, that is used by web browsers.
- TCP is a connection oriented and stateful protocol.
- While IP protocol deals with sending packets, TCP ensures reliable transmission of packets in-order.
- TCP also provides error detection capability so that duplicate packets can be discarded and lost packets are retransmitted.
- The flow control capability of TCP ensures that rate at which the sender sends the data is not too high for the receiver to process.
- The congestion control capability of TCP helps in avoiding network congestion and congestion collapse which can lead to degradation of network performance.



Transport Layer

UDP

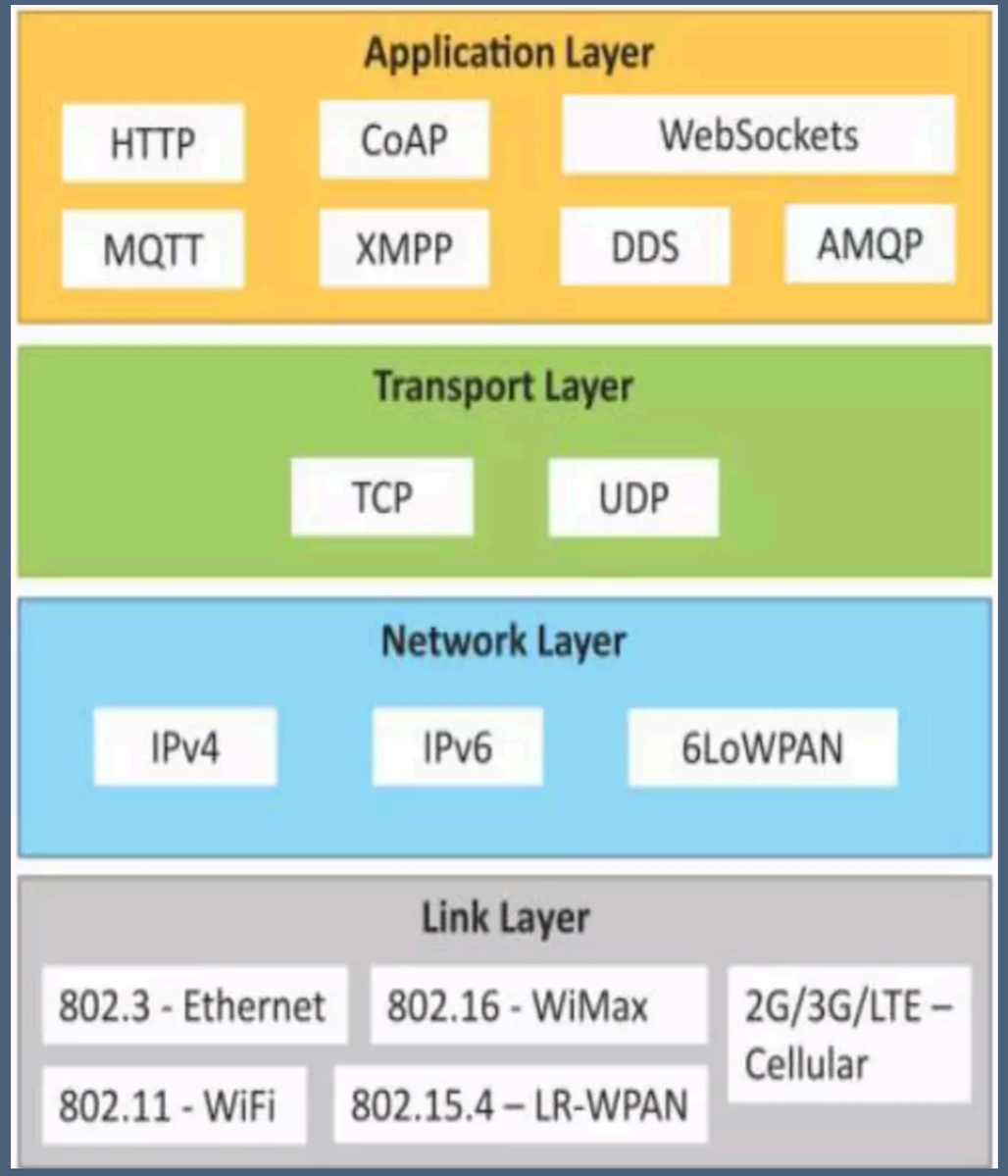
- UDP is a connectionless protocol.
- UDP is useful for time-sensitive applications that have very small data units to exchange and do not want the overhead of connection setup.
- UDP does not provide guaranteed delivery, ordering of messages and duplicate elimination.



IoT Protocols

Application layer

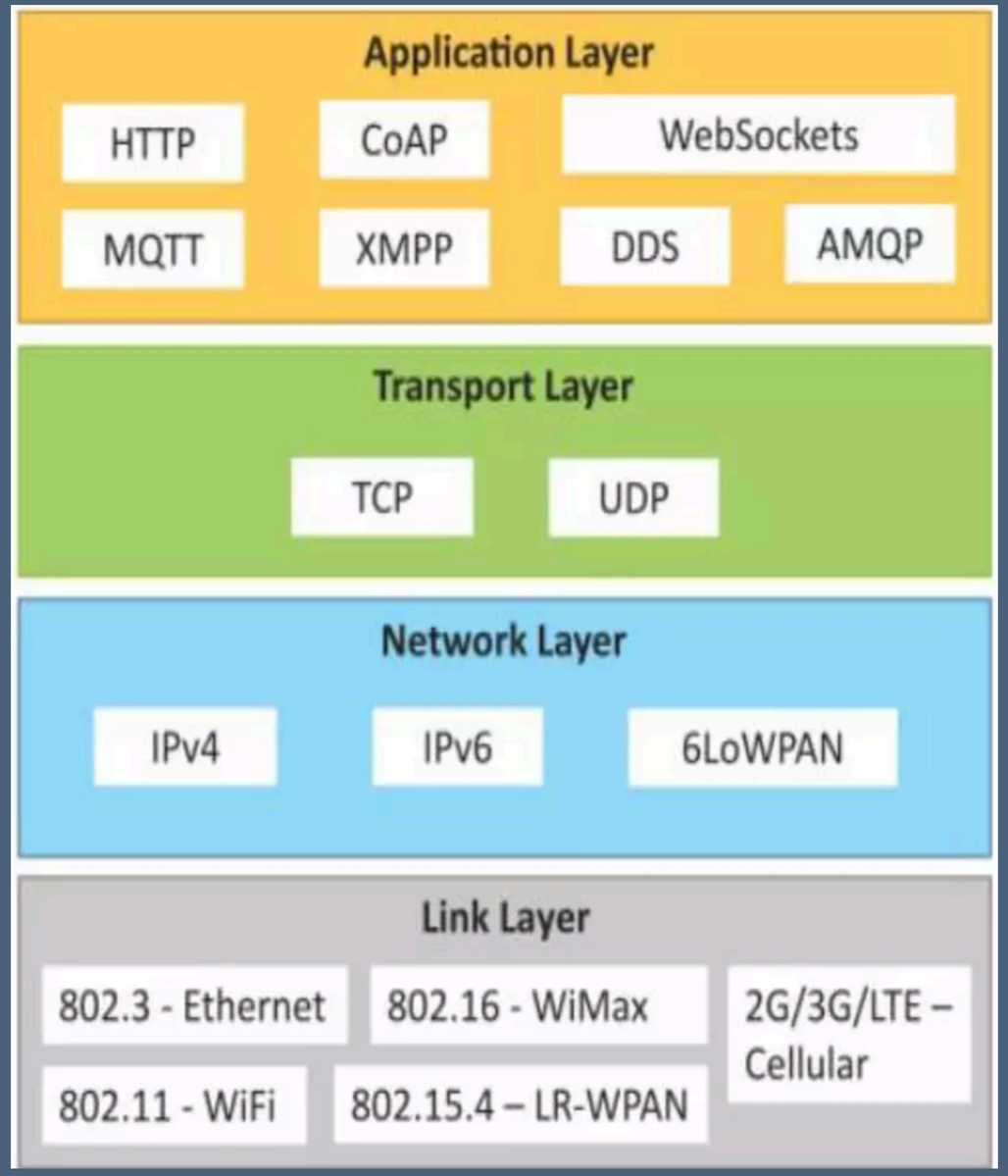
- Application layer protocols define how the applications interface with the lower layer protocols to send the data over the network.
- The application data, typically in files, is encoded by the application layer protocol and encapsulated in the transport layer protocol which provides connection or transaction oriented communication over the network.
- Port numbers are used for application addressing (for example port 80 for HTTP, port 22 for SSH, etc.).



Application layer

HTTP

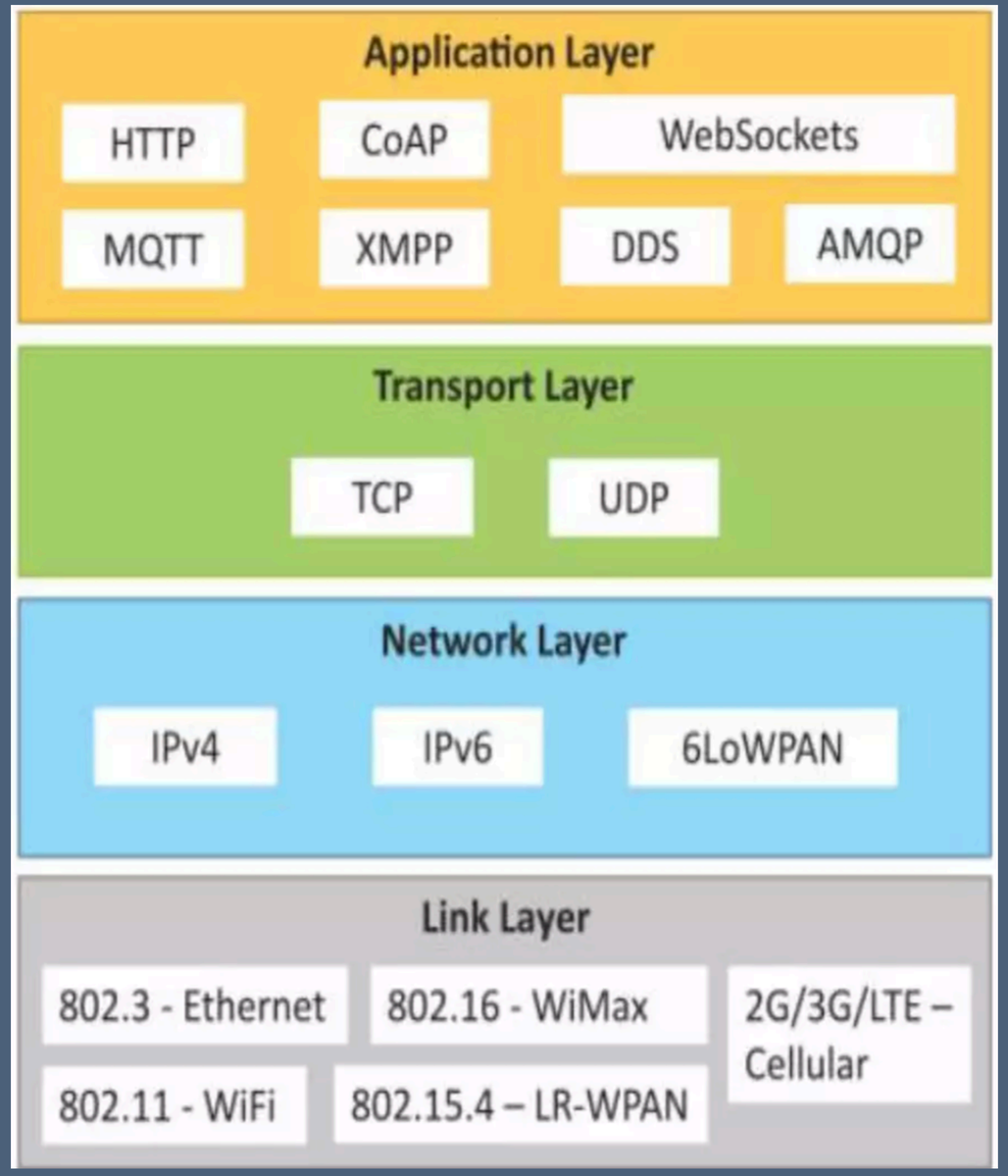
- Hypertext Transfer Protocol (HTTP) is the application layer protocol that forms the foundation of the World Wide Web
- HTTP includes commands such as GET, PUT, POST, DELETE, HEAD, TRACE, OPTIONS, etc
- HTTP is a stateless protocol and each HTTP request is independent of the other requests.
- An HTTP client can be a browser or an application running on the client



Application layer

CoAP

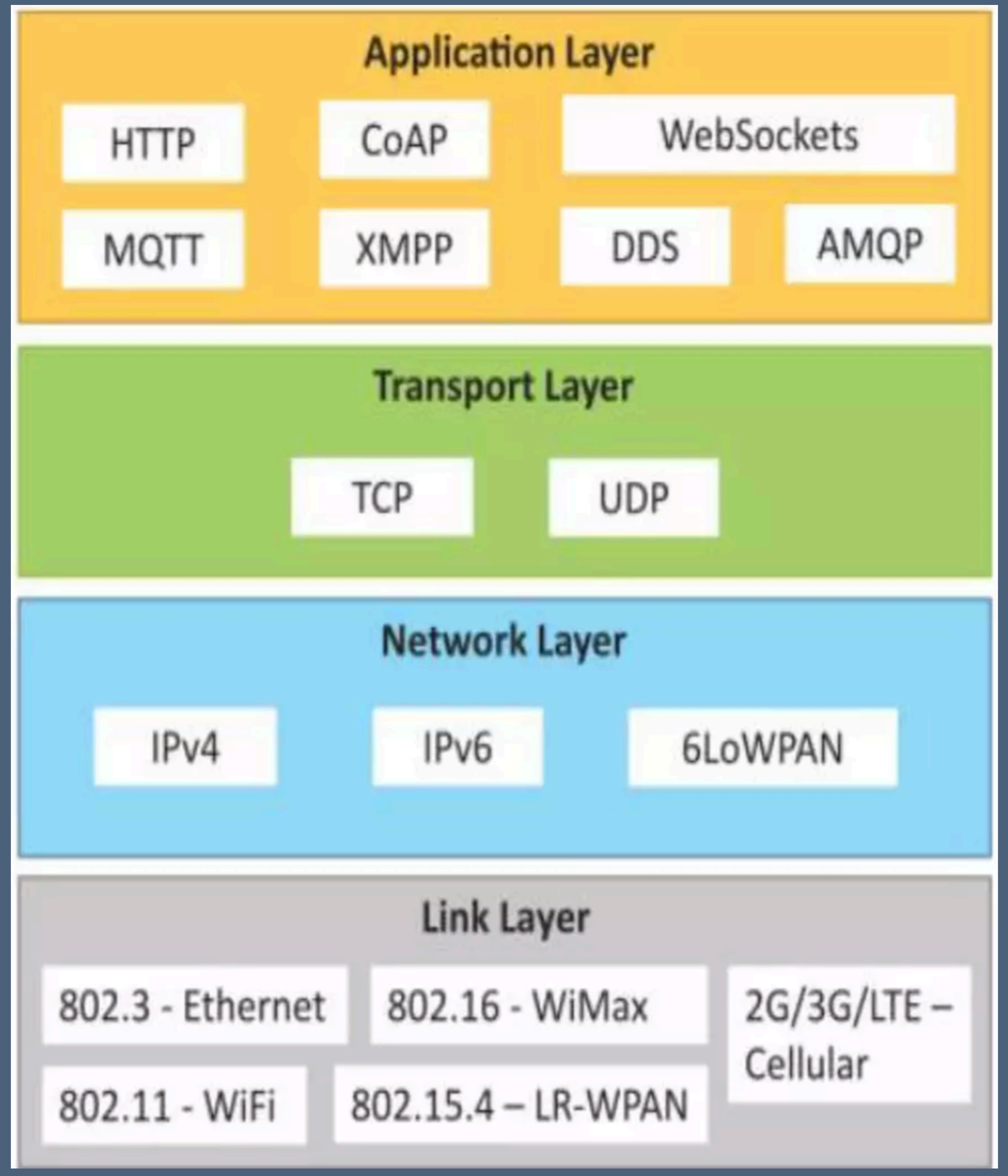
- Constrained Application Protocol is an application layer protocol for machine-to-machine (M2M) applications
- meant for constrained environments with constrained devices and constrained networks.
- is a web transfer protocol and uses a request-response model, however it runs on top of UDP instead of TCP.
- CoAP is designed to easily interface with HTTP.
- Like HTTP, CoAP supports methods such as GET, PUT, POST, and DELETE



Application layer

Web Sockets

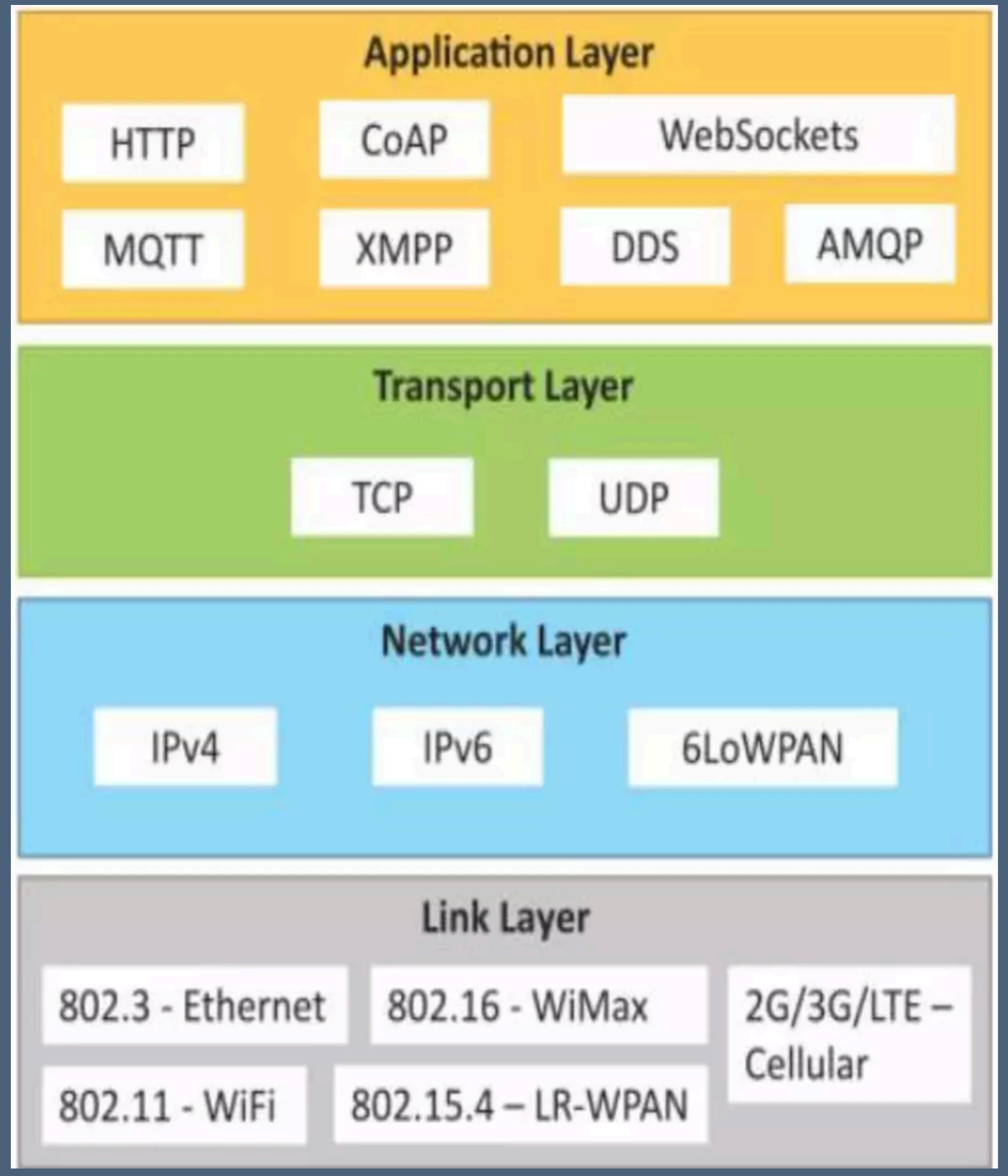
- allows full-duplex communication over a single socket connection for sending messages between client and server.
- based on TCP
- allows streams of messages to be sent back and forth between the client and server while keeping the TCP connection open.
- The client can be a browser, a mobile application or an IoT device.



Application layer

MQTT

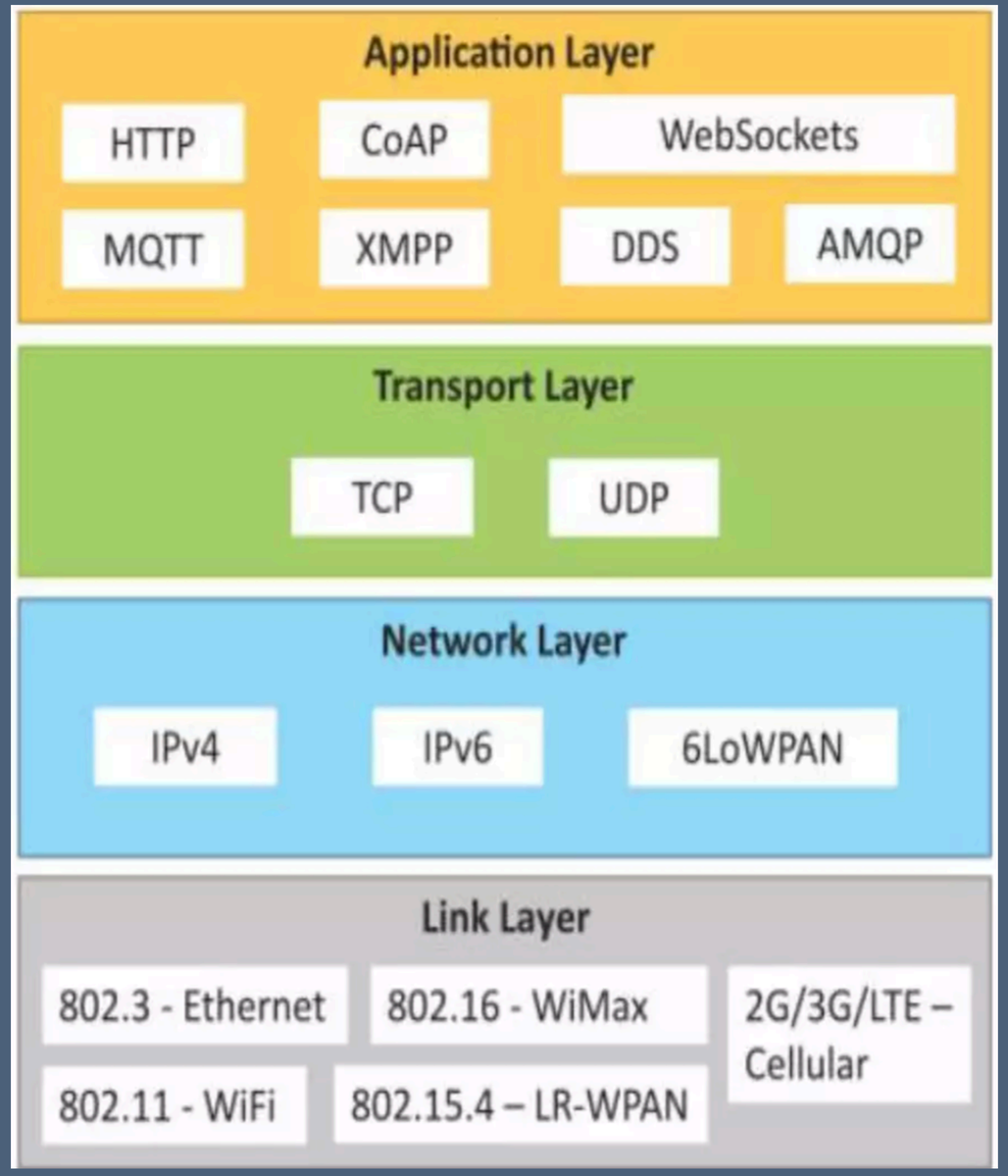
- Message Queue Telemetry Transport (MQTT) is a light-weight messaging protocol based on the publish-subscribe model.
- uses a client-server architecture
- where the client (such as an IoT device) connects to the server (also called MQTT Broker) and publishes messages to topics on the server.
- well suited for constrained environments where the devices have limited processing and memory resources and the network bandwidth is low.



Application layer

XMPP

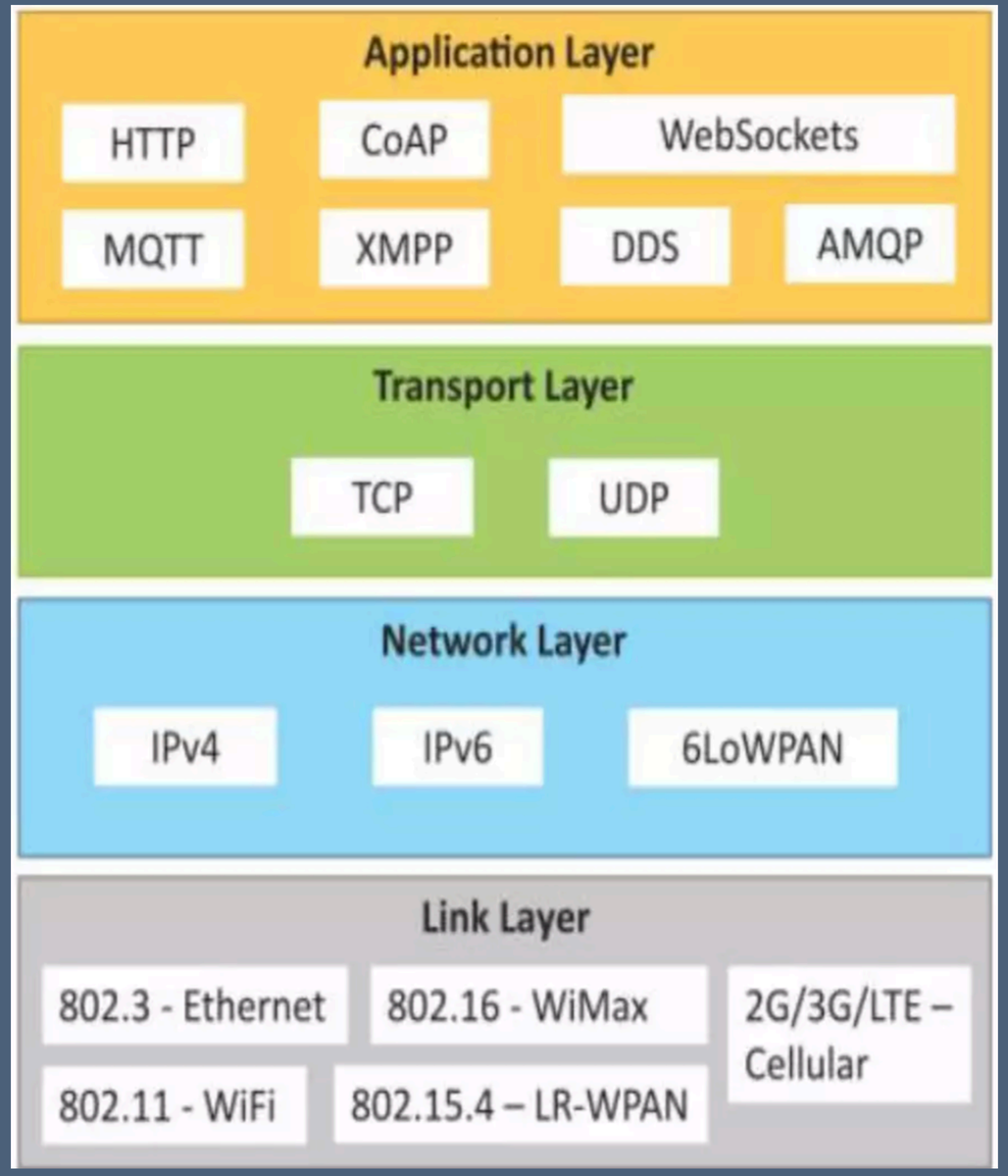
- Extensible Messaging and Presence Protocol is a protocol for real-time communication and streaming XML data between network entities
- powers wide range of applications including messaging, presence, data syndication, gaming, multi-party chat and voice/video calls.
- allows sending small chunks of XML data from one network entity to another in near real-time.
- supports both client-to-server and server-to-server communication paths.



Application layer

DDS

- Data Distribution Service is a data-centric middleware standard for device-to-device or machine-to-machine communication
- uses a publish-subscribe model
- Publisher is an object responsible for data distribution and the subscriber is responsible for receiving published data.
-



Application layer

AMQP

- Advanced Message Queuing Protocol (AMQP) is an open application layer protocol for business messaging.
- supports both point-to-point and publisher/subscriber models, routing and queuing
- Publishers publish the messages to exchanges which then distribute message copies to queues.
- Messages are either delivered by the broker to the consumers which have subscribed to the queues
- or the consumers can pull the messages from the queues.