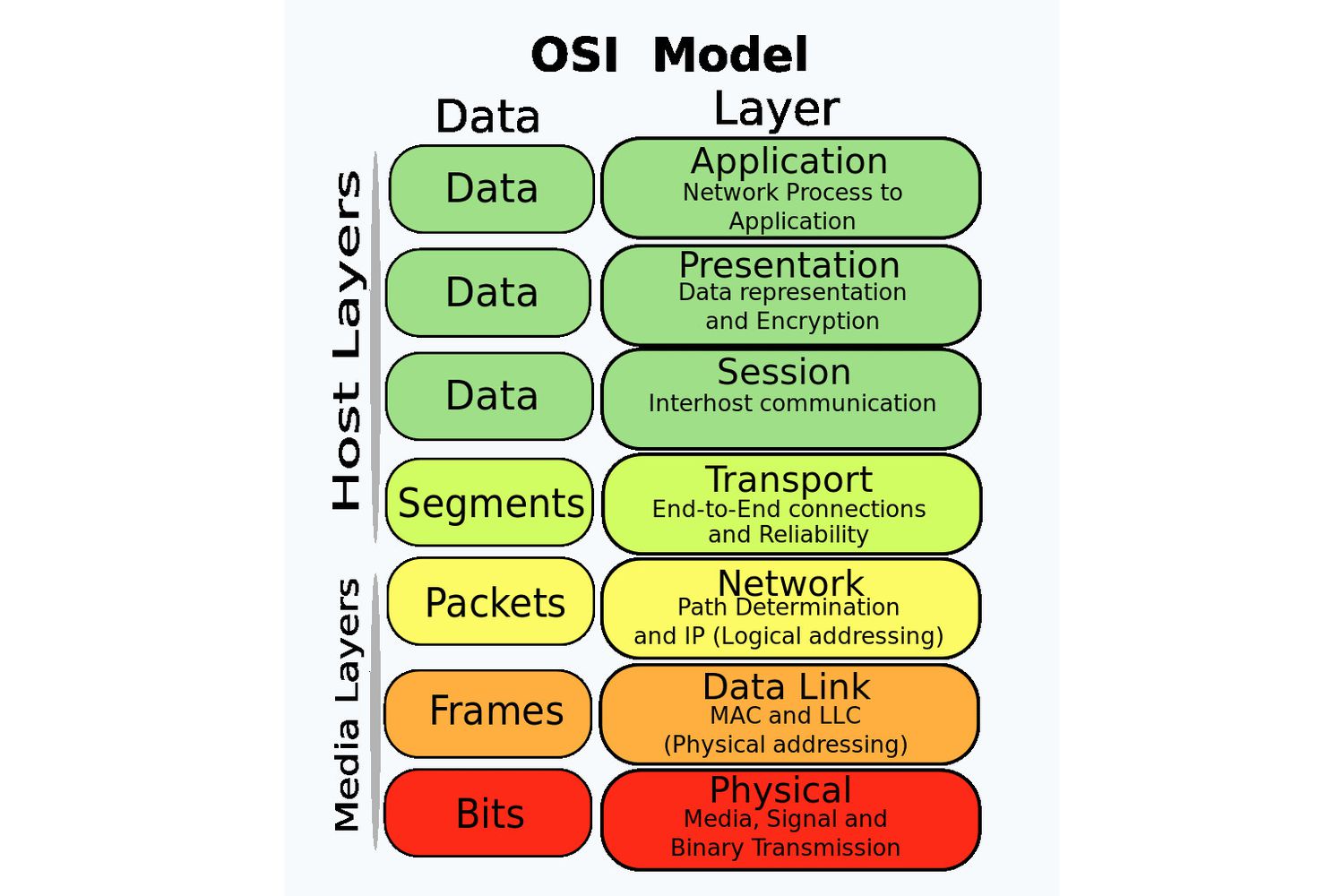
OSI (Open Systems Interconnection) is a reference framework that explains the process of transmitting data between computers. It is divided into seven layers that work together to carry out specialised network functions, allowing for a more systematic approach to networking.



Has 7 layers:

1. Physical – physical transmission of signals such as with Cable, network interface cards, hubs
2. Data Link – Mac addresses, switches, Ethernet
3. Network – IP addresses, switches, responsible for addressing and routing packets
4. Transport – end to end connections TCP, UDP, port numbers
5. Session – connection maintenance, start & stop sessions | API sockets
6. Presentation – data representation and encryption | SSH, SSL, IMAP, FTP, JPEG
7. Application – Human computer interaction | HTTP, DNS, FTP, POP3

OSI model is a framework not a communication environment.

Each layer communicates with three other layers (except for 1 and 7):

* Layer above it
* Layer below it
* Corresponding layer on another computer

A diagram of a data link

Description automatically generated

**Base definitions**

Information and service data when transferred can be represented in a different form.

Main formats:

* Frames
* Packets
* Datagrams
* Segments
* Messages
* Cells
* Data units

Frame- A block of information on the data link layer.

Packet – A block of information on the network layer.

Datagram – block of information when data exchange is NOT connection oriented.

Segment – A block of information on the transport layer.

Message – A block of information OVER the network layer.

Cell – A block of information of a fixed size on the data link layer. Used in ATM and SMDS networks.

Data unit – general term, covering several types of information blocks:

* Service data units (SDU) – data blocks, defining requests from higher layers to lower layers
* Protocol data units (PDU) – equivalent to packet
* Bridge control data units (BPDU) – hello message

Connection-oriented and connectionless network services

Connection oriented (e.g TCP):

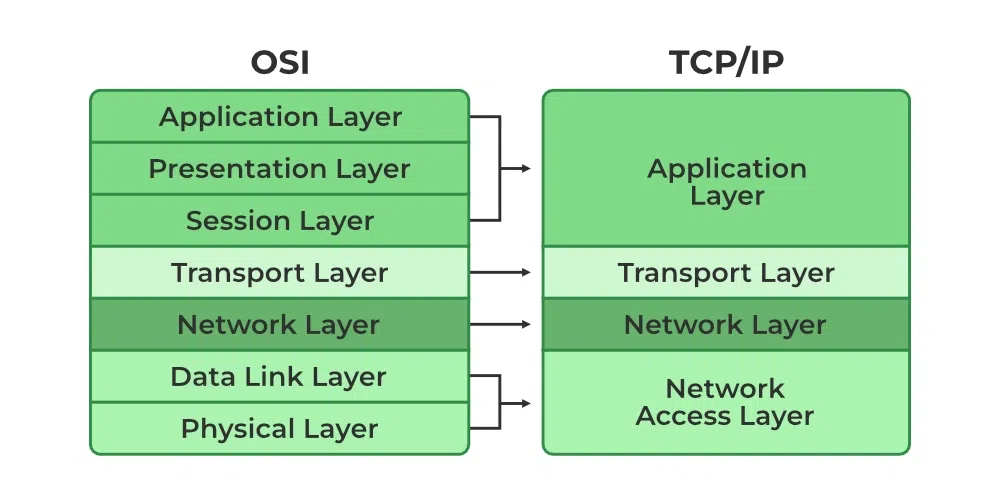
It is necessary to establish a connection

Connectionless (e.g UDP)

Data is sent without connection establishment.

**TCP/IP model**

The OSI model has 7 layers (Physical, Data Link, Network, Transport, Session, Presentation, Application), while the TCP/IP model has 4 layers (Network Interface, Internet, Transport, Application). The OSI model is a theoretical framework developed by ISO, while the TCP/IP model is a practical framework developed by the U.S. Department of Defense for internet communication. The OSI model's layers each have specific functions, whereas the TCP/IP model's layers combine some of these functions for simplicity. Protocols in the OSI model include X.25 and FTAM, whereas the TCP/IP model includes IP, TCP, and HTTP. The OSI model is more detailed, but the TCP/IP model is more commonly used.



The TCP/ip model can sometimes is also sometimes described as a 5-layer model instead of 4.

A blue screen with white text

Description automatically generated

encapsulation involves wrapping data with protocol-specific headers at each layer as it moves from the application to the physical transmission medium:

1. **Application Layer**: The data is generated by an application and an application header is added.
2. **Transport Layer**: The transport layer (TCP/UDP) adds a transport header, creating a segment.
3. **Network Layer**: The network layer (IP) adds an IP header, creating a packet.
4. **Data Link Layer**: The data link layer adds a frame header and trailer, creating a frame.
5. **Physical Layer**: The frame is converted into bits for transmission over the physical medium.

At the destination, the process is reversed: the physical layer receives the bits, which are then de-encapsulated by each successive layer to retrieve the original application data.

**Encapsulation in OSI**

Encapsulation in the OSI model involves adding protocol-specific headers (and sometimes trailers) to data as it moves down the layers, just like in the TCP/IP model:

1. **Application Layer**: Data is created by the application.
2. **Presentation Layer**: Data is formatted, encrypted, or compressed, with a presentation header added if necessary.
3. **Session Layer**: Manages sessions and may add a session header.
4. **Transport Layer**: Adds a transport header (e.g., TCP/UDP), creating a segment.
5. **Network Layer**: Adds a network header (e.g., IP), creating a packet.
6. **Data Link Layer**: Adds a frame header and trailer, creating a frame.
7. **Physical Layer**: Converts the frame into bits for transmission over the physical medium.

At the receiving end, the data is de-encapsulated layer by layer, removing headers and trailers, to retrieve the original application data.