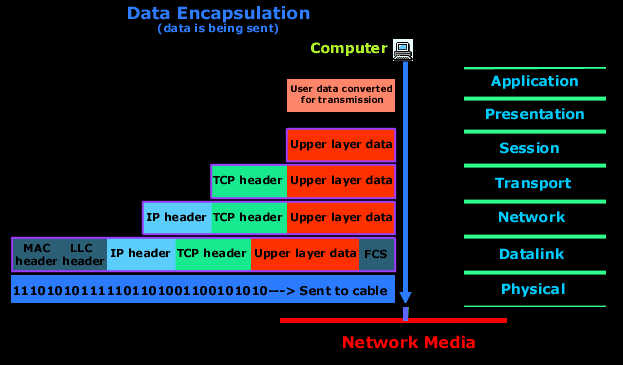
# **Data Encapsulation And De-Encapsulation**

* **Data Encapsulation :**
* Encapsulation is the process of adding additional information when data is traveling in an OSI or TCP/IP model. The information has been added on the sender's side, from the Application layer to the Physical layer.
* **OSI Model and Working of Encapsulation :**

**🔸 Step-by-Step Process:**

* **Application, Presentation, Session Layers (Layer 7–5):**
  + Data is created by the user or application (like email, message, etc.).
  + No extra information is added here — it just prepares data for transmission.
* **Transport Layer (Layer 4):**
  + Breaks large data into smaller pieces (segments).
  + Adds a **header** containing: source port, destination port, sequence number, etc.
  + These details help in **data delivery and reassembly** at the receiver’s side.
  + The encapsulated data here is called **Segments** (for TCP) or **Datagrams** (for UDP).
* **Network Layer (Layer 3):**
  + Adds an **IP header** containing source and destination IP addresses.
  + Helps route data across different networks.
  + Data here is called a **Packet**.
* **Data Link Layer (Layer 2):**
  + Adds a **header** (source and destination MAC address) and a **trailer** (for error checking).
  + Ensures that the packet reaches the correct physical device.
  + Data here is called a **Frame**.
* **Physical Layer (Layer 1):**
  + Converts frames into **bits (0s and 1s)** and transmits them over cables or wireless signals.
  + Data here is called **Bits**.

✅ **In short:**  
**Data → Segment → Packet → Frame → Bits**

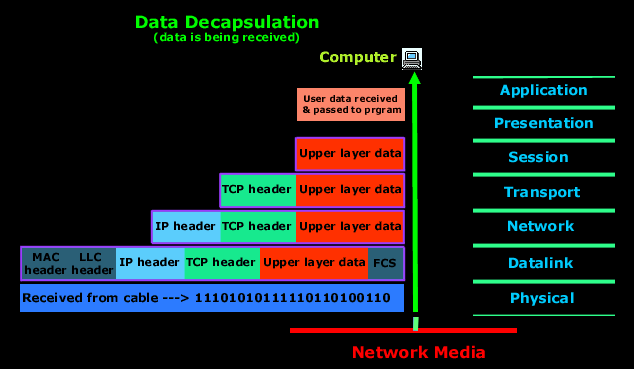


**🔄 De-Encapsulation :**

* **De-Encapsulation is the reverse of encapsulation. It happens at the receiver’s side when data moves up the OSI layers.**

1. **Physical Layer:  
   Receives bits and converts them back into frames.**
2. **Data Link Layer:  
   Checks the MAC address and removes the layer 2 header and trailer if correct.  
   Passes the packet to the Network layer.**
3. **Network Layer:  
   Checks the IP addresses. If matched, it removes the layer 3 header and sends the data to the Transport layer.**
4. **Transport Layer:  
   Removes TCP/UDP header and reassembles the data pieces in correct order.**
5. **Session, Presentation, and Application Layers:  
   The original message or file is delivered to the user or application.**

**✅ In short:  
Bits → Frame → Packet → Segment → Data**



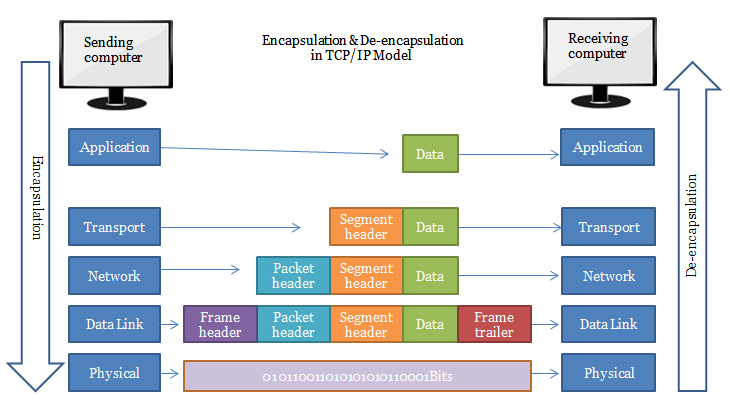
**Data Encapsulation in TCP/IP Model :**

* **Encapsulation in the TCP/IP model means adding headers (and sometimes trailers) to data as it moves from the Application layer down to the Network Access layer before transmission.**

**🔸 Step-by-Step Process:**

1. **Application Layer (Layer 4)**
   * **The user creates data (for example, an email or web request).**
   * **Protocols like HTTP, FTP, DNS, SMTP, etc., work here.**
   * **Data is simply called Data.**
2. **Transport Layer (Layer 3)**
   * **Breaks data into smaller units and adds a Transport Header (TCP or UDP).**
   * **Includes source port, destination port, and sequence number.**
   * **🔹 Data here is called Segment (TCP) or Datagram (UDP).**
3. **Internet Layer (Layer 2)**
   * **Adds an IP Header containing source and destination IP addresses.**
   * **Responsible for logical addressing and routing.**
   * **🔹 Data here is called a Packet.**
4. **Network Access Layer (Layer 1)**
   * **Adds MAC Header and Trailer for physical addressing and error checking.**
   * **Converts packets into bits for transmission.**
   * **🔹 Data here is called a Frame, and finally Bits on the wire.**

**✅ Summary (Encapsulation order):  
Data → Segment → Packet → Frame → Bits**



**🔄 Data De-Encapsulation in TCP/IP Model :**

**De-Encapsulation is the reverse process that happens at the receiver’s side.  
As the data moves up the layers, each layer removes its corresponding header and processes the information.**

**🔸 Step-by-Step Process:**

1. **Network Access Layer:**
   * **Receives bits and converts them into frames.**
   * **Checks MAC address, removes header/trailer, and passes packet to the Internet layer.**
2. **Internet Layer:**
   * **Checks the IP address in the header.**
   * **Removes the IP header and passes the segment to the Transport layer.**
3. **Transport Layer:**
   * **Uses TCP/UDP information to reassemble data.**
   * **Removes the transport header and sends data to the Application layer.**
4. **Application Layer:**
   * **Final destination where the original message or file is delivered to the user.**

**✅ Summary (De-Encapsulation order):  
Bits → Frame → Packet → Segment → Data**