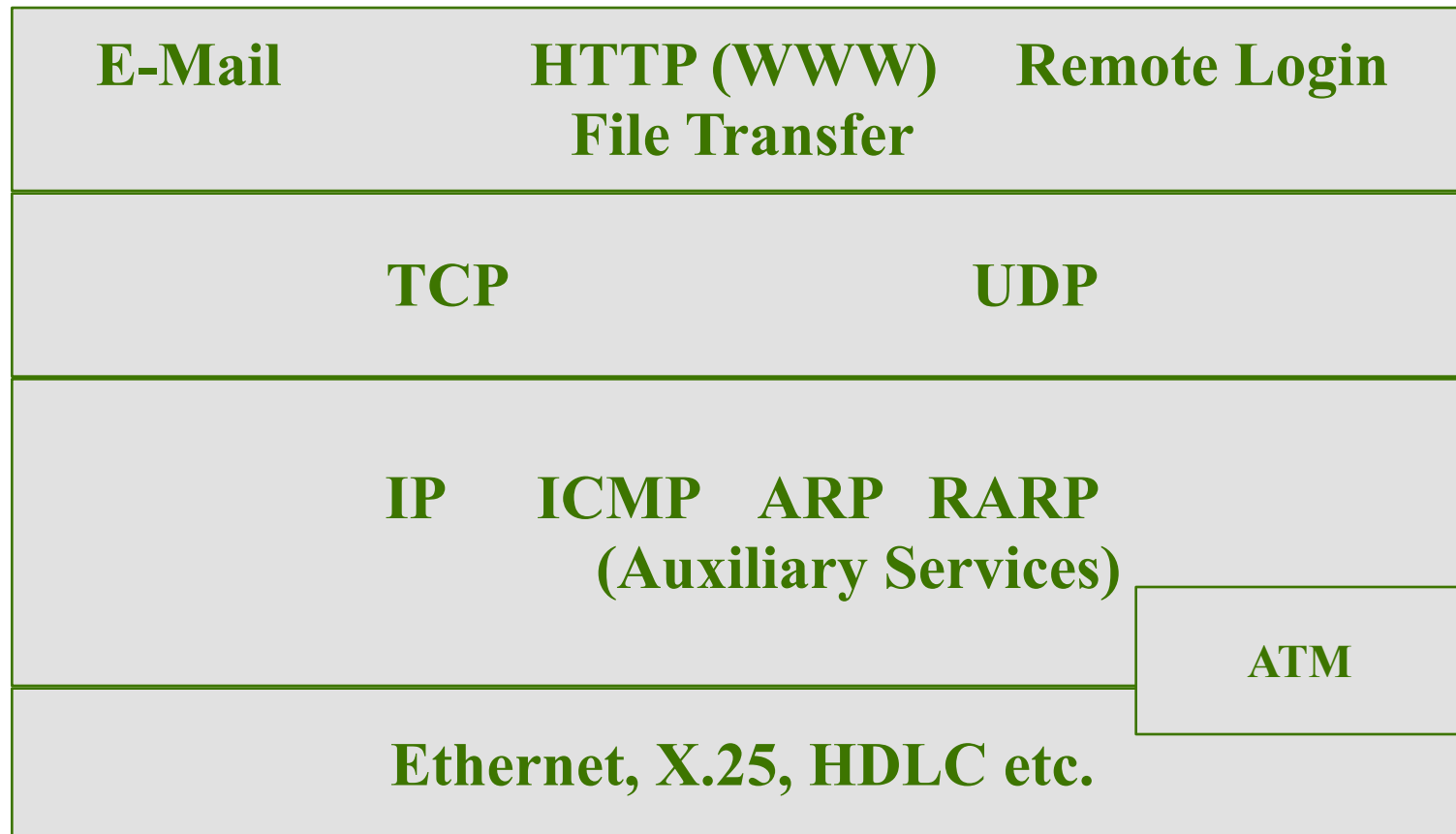


# Performance “Just enough TCP”

ITL

# Protocol Overview

---



# Connection Types in TCP/IP

---

**Transport Layer**

**TCP: Connection Oriented**

**UDP: Connection-less**

**Network Layer**

**Connection-less**

**Data Link Layer and Physical Network**

**Depends on the network**

# What is Performance?

---

- What data rate can I sustain (on average) between two nodes on the Internet?
  - Throughput
  - “Speed”
- How long does it take to get a reply for a request I send out?
  - Latency
  - Round Trip Time

# “Frame Transmission Time”

---

- Depends on
  - Frame Length
  - Channel Bit Rate

$$t_{frame} = \frac{L}{R}$$

$L$  = Frame Length in Bits

$R$  = Channel Data Rate in Bits per Second

# “Frame Propagation Time”

---

- Depends on
  - Physical distance between stations
  - Signal propagation speed
    - $2 \times 10^8$  to  $3 \times 10^8$  meters/second

$$t_{prop} = \frac{d}{v}$$

$d$  = distance between stations in meters

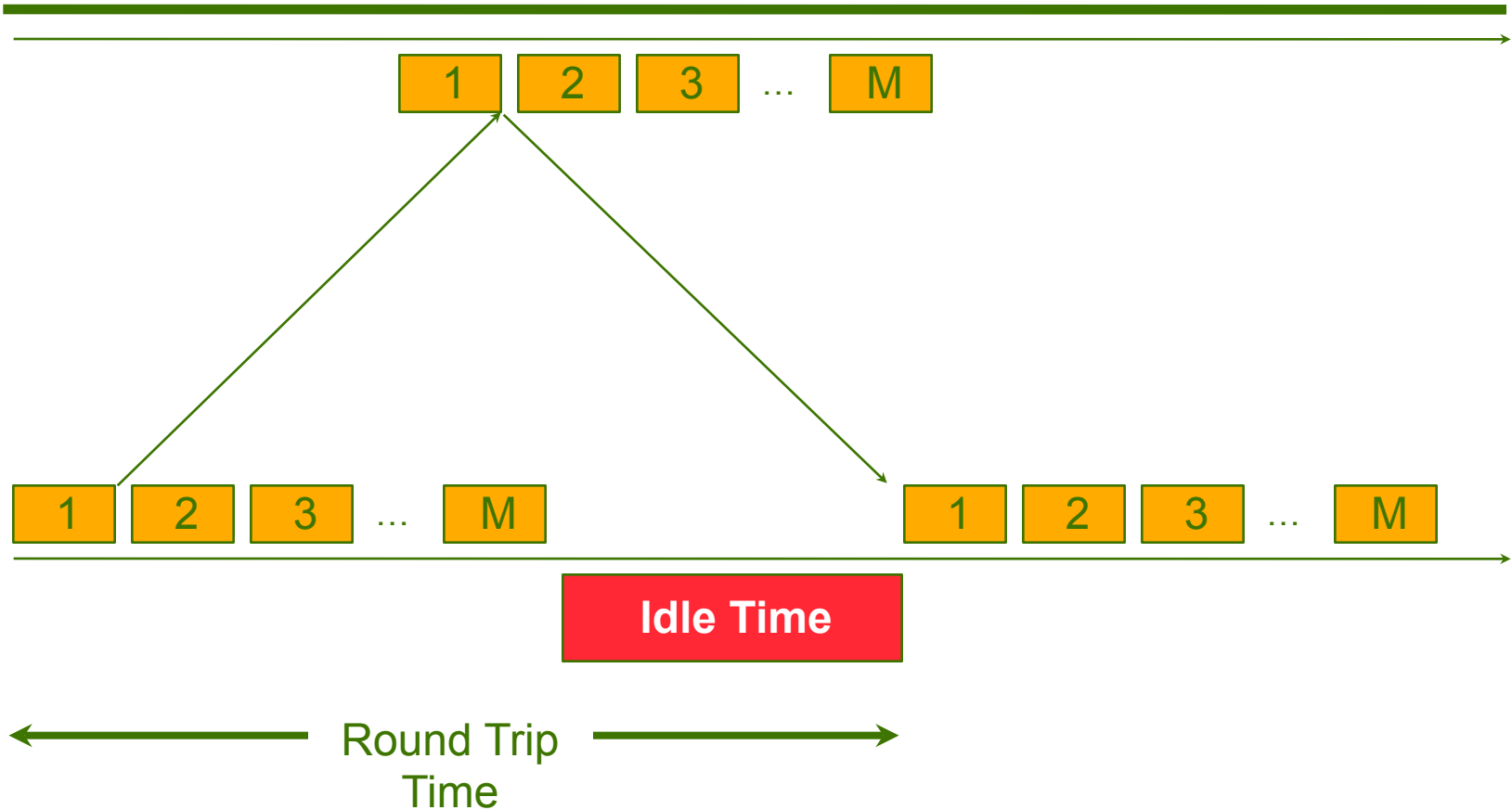
$v$  = speed of signal propagation

# Round Trip Time (RTT)

---

- Combination of
  - Frame Transmission times on intermediate links
  - Frame propagation times on all links
  - **Queue wait times** in all intermediate nodes

# “Sliding Window”



# Window Size and Throughput

---

- Maximum throughput is
  - (window size)/RTT
  - Actual throughput may of course be less.
- Bandwidth-Delay-Product
  - $BDP = RTT * (\text{effective bandwidth})$
  - Measures the amount of data “in the network” at any given time.

# Sliding Window flow control

---

- Invented to control flow to slower devices (i.e. end-to-end)
- Almost accidentally provides a cap on data flow rate
- TCP uses a sliding window mechanism for multiple purposes
  - Rate Control
  - Congestion Control/Avoidance
- The window size is dynamically adjusted

# UDP Header

---

**Source Port**

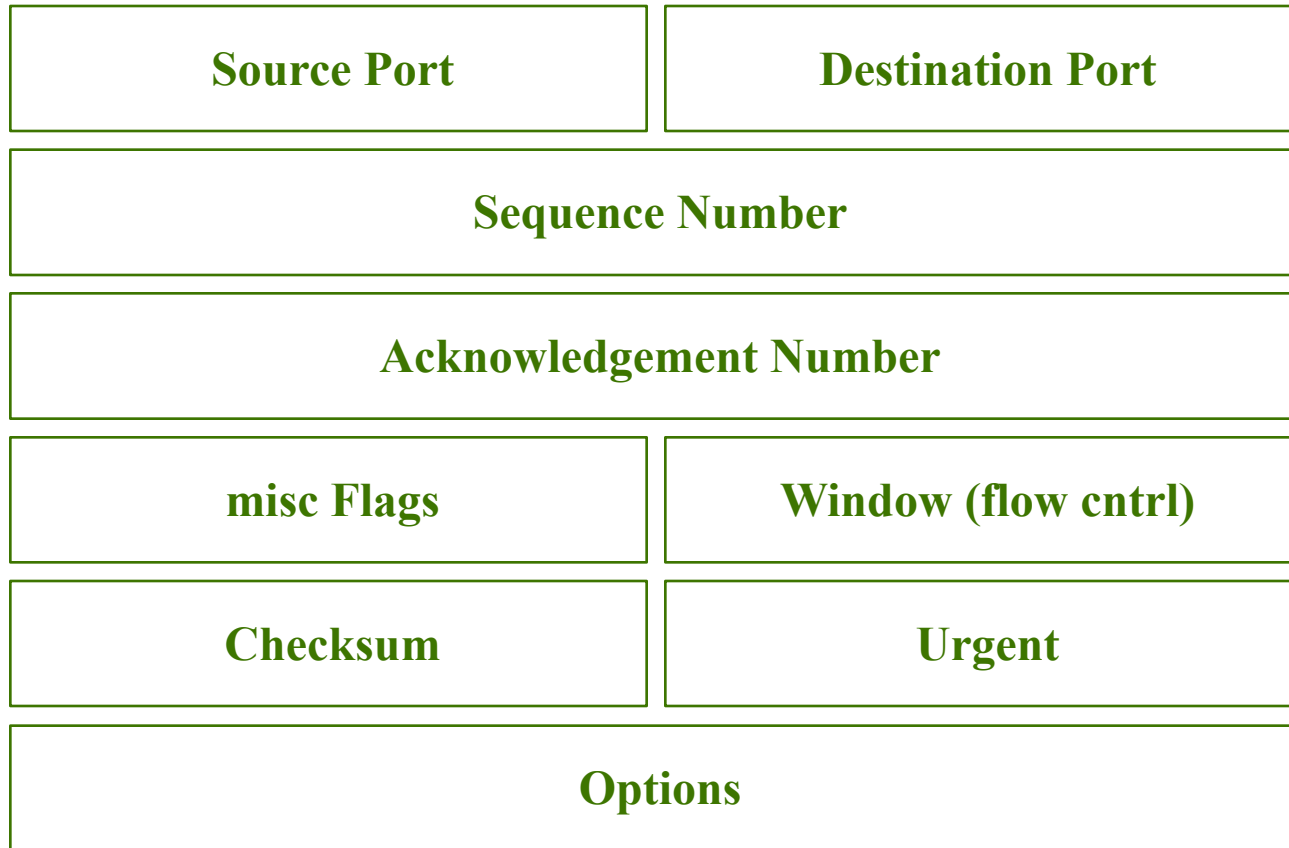
**Destination Port**

**Length**

**Checksum**

# TCP Header

---



# TCP Connection Setup

---

- “Three-Way Handshake”
  - Send SYN packet
  - Wait for peer to return a SYN/ACK packet
  - Acknowledge the SYN/ACK packet

# TCP Connection Termination

---

- Send a FIN packet
- Wait to receive acknowledgement of FIN

# TCP Data Exchange

---

Sequence Number

Acknowledgement Number

- Sequence Numbers - Sliding Window
  - Arbitrary initial setting
  - Labels the first byte of the segment
- Acknowledgements
  - Indicate the next byte the receiver is looking for, all previous bytes have been received.

# Misc Slides

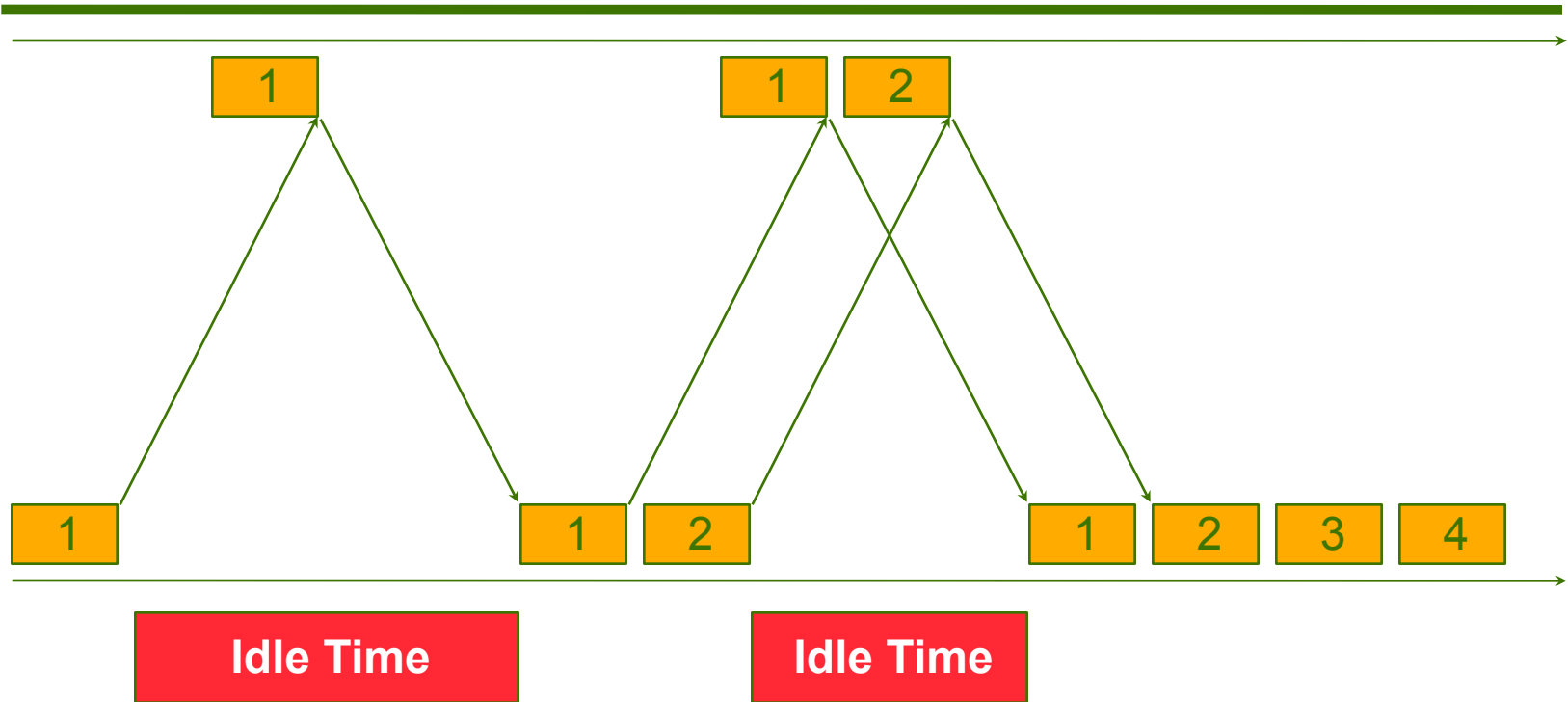
---

# Congestion Issues

---

- Slow Start - New Connection
  - Set send window to  $n \cdot \text{MSS}$  ( $n \leq 4$ )
  - Increase the window by MSS for each ack received
  - Exponential increase in send window size
- What is the limit?
  - Window size reached before full utilization
  - Path is overloaded and an intermediate router discards one or more packets

# Slow Start Congestion Control



Note: recent TCP amendments permit more than 1 initial segment

Window doubles in each "cycle"

# Controls and Measurements

---

- TCP Window
  - Most controls are automatic
  - Operating systems can and do limit the largest possible TCP window
- Measurements
  - Ping, Traceroute
  - Test tools that imitate real traffic flow
    - iperf (see the assigned reading)

# Control Mechanisms

---

- Connection-Less traffic
  - Packets representing overload are discarded
  - Retransmissions should not take place
  - Applications should adapt
- Connection-Oriented traffic
  - Packets representing overload are also discarded
  - Lost packets must be retransmitted
  - Hosts must adjust sending rate

# In practice

---

- How is the sliding window mechanism used in TCP
- What dynamic adjustments are made to the window size
- What control do we have over performance parameters
- Starting with a quick UDP and TCP review...