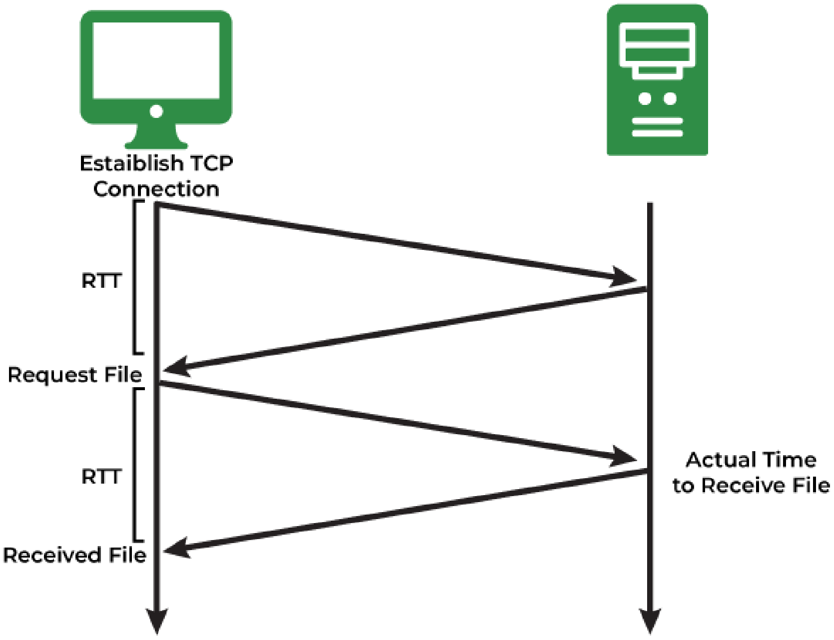
What is round-trip time?

Round-trip time (RTT) is the duration in milliseconds (ms) it takes for a network request to go from a starting point to a destination and back again to the starting point. RTT is an important metric in determining the health of a connection on a local network or the larger Internet, and is commonly utilized by network administrators to diagnose the speed and reliability of network connections.

ignal and reverts it back.



RTT(Round Trip Time) Measurement

What Are Common Factors that Affect RTT?

There are certain factors that can bring huge changes in the value of RTT. These are enlisted below:

- **Distance:** It is the length in which a signal travels for a request to reach the server and for a response to reach the browser,
- **Transmission medium:** The medium which is used to route a signal, which helps in faster transfer of request is transmitted.
- **Network hops:** It is the time that servers take to process a signal, on increasing the number of hops, RTT will also increase.
- **Traffic levels:** Round Trip Time generally increases when a network is having huge traffic which results in that, for low traffic RTT will also be less.
- **Server response time:** It is the time taken by a server to respond to a request which basically depends on the capacity of handling requests and also sometimes on the nature of the request.

Delays in Computer Network

The delays, here, means the time for which the processing of a particular packet takes place. We have the following types of delays in computer networks:

1. Transmission Delay:
The time taken to transmit a packet from the host to the transmission medium is called Transmission delay.

For example, if bandwidth is 1 bps (every second 1 bit can be transmitted onto the transmission medium) and data size is 20 bits then what is the transmission delay? If in one second, 1 bit can be transmitted. To transmit 20 bits, 20 seconds would be required.

Let B bps is the bandwidth and L bit is the size of the data then transmission delay is,

```
Tt = L/B
```

This delay depends upon the following factors:

- If there are multiple active sessions, the delay will become significant.
- Increasing bandwidth decreases transmission delay.
- MAC protocol largely influences the delay if the link is shared among multiple devices.
- Sending and receiving a packet involves a context switch in the operating system, which takes a finite time.

2. Propagation delay:
After the packet is transmitted to the transmission medium, it has to go through the medium to reach the destination**. Hence the time taken by the last bit of the packet to reach the destination is called propagation delay**.

Factors affecting propagation delay:

1. **Distance –** It takes more time to reach the destination if the distance of the medium is longer.
2. **Velocity –** If the velocity(speed) of the medium is higher, the packet will be received faster.

```
Tp = Distance / Velocity
```

Note:

```
Velocity =3 X 10^8 m/s (for air)

Velocity= 2.1 X 10^8 m/s (for optical fibre)
```

3. Queueing delay:
Let the packet is received by the destination, the packet will not be processed by the destination immediately. It has to wait in a queue in something called a buffer. **So the amount of time it waits in queue before being processed is called queueing delay**.
In general, we can’t calculate queueing delay because we don’t have any formula for that.

This delay depends upon the following factors:

- If the size of the queue is large, the queuing delay will be huge. If the queue is empty there will be less or no delay.
- If more packets are arriving in a short or no time interval, queuing delay will be large.
- The less the number of servers/links, the greater is the queuing delay.

4. Processing delay:
Now the packet will be taken for the processing which is called processing delay.
Time is taken to process the data packet by the processor that is the time required by intermediate routers to decide where to forward the packet, update TTL, perform header checksum calculations.

It also doesn’t have any formula since it depends upon the speed of the processor and the speed of the processor varies from computer to computer.

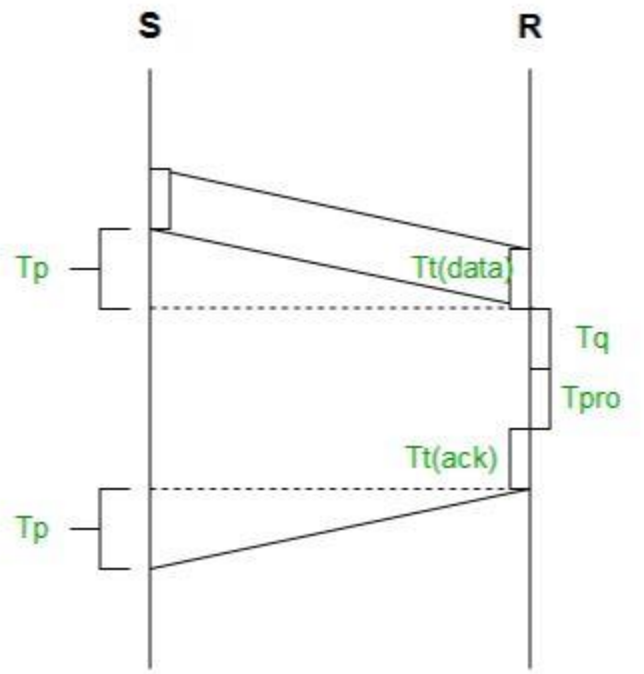
Note: Both queueing delay and processing delay doesn’t have any formula because they depend on the speed of the processor
This delay depends upon the following factors:

- It depends on the speed of the processor.

```
Ttotal = Tt + Tp + Tq + Tpro



Ttotal = Tt+Tp

(when taking Tq and Tpro equals to 0)
```

Efficiency of Stop and Wait Protocol

Stop and Wait is a flow control protocol. In which the sender sends one packet and waits for the receiver to acknowledge and then it will send the next packet. In case if the acknowledgement is not received, the sender will retransmit the packet. This is the simplest one and easy to implement. but the main disadvantage is the efficiency is very low.



Total time taken to send one packet,

```
= Tt(data) + Tp(data) + Tq + Tpro + Tt(ack) + Tp(ack)


Since,

Tp(ack) = Tp(data)


And,

Tt(ack) << Tt(data).


So we can neglect Tt(ack)

Tq = 0 and Tpro = 0
```

```
Hence,

Total time = Tt(data) + 2 * Tp
```

Where,

```
Tt(data) : Transmission delay for Data packet

Tp(data) : propagation delay for Data packet

Tq: Queuing delay

Tpro: Processing delay

Tt(ack): Transmission delay for acknowledgment

Tp(ack) : Propagation delay for acknowledgment
```

We know that the **Efficiency (η)**,

```
= Useful time / Total cycle time.

= Tt / (Tt + 2*Tp)

= 1 / (1+2*(Tp/Tt))

= 1 / (1+2*a)


where,

a = Tp / Tt
```

if Tt(ack) :Transmission delay for acknowledgment , Tq: Queuing delay and Tpro: Processing delay is mention

We know that the Efficiency (η),
=Useful time / Total cycle time
=Tt(data) /Tt(data) + Tp(data) + Tq + Tpro + Tt(ack) + Tp(ack)
Above formula is applicable for any condition, if any of the things are not given we assume it to be 0.

Other name of Efficiency are **Line Utilization or Link Utilization or Sender Utilization.**

Throughput: Number of bits send per second, which is also known as Effective Bandwidth or Bandwidth utilization.
Throughput=Length of frame/Total Cycle time
=L/(Tt(data) + Tp(data) + Tq + Tpro + Tt(ack) + Tp(ack))
=(L/BW)*BW)/(Tt(data) + Tp(data) + Tq + Tpro + Tt(ack) + Tp(ack))
=Tt(data)*BW/(Tt(data) + Tp(data) + Tq + Tpro + Tt(ack) + Tp(ack))
= η*BW

we know that Tt(data)=L/BW

where Tt(data)=Transmission Delay

BW=Band Width

Above formula is applicable for any condition, if any of the things are not given we assume it to be 0.

Other name of Throuhput are **Effective Bandwidth or Bandwidth Utilization or Maximum data rate possible.**

```
Throughput,
= L/(Tt + 2*Tp)
= ((L/BW)*BW)/(Tt + 2*Tp)
= Tt/(Tt + 2*Tp) * BW
= 1/(1 + 2a) * BW

Hence, Throughput
= η * BW

where,
BW : BandWidth
L : Size of Data packet
```

Factors affecting Efficiency:

```
n = 1/(1 + 2*(Tp/Tt)
= 1/(1 + 2*(d/v)*(BW/L))

where,
d = distance between source and receiver
v = velocity
```

Lets see an example. **Example:** Given,

```
Tt = 1ms
Tp = 2ms
Bandwidth = 6 Mbps

Efficiency(η)
= 1/(1 + 2*a)
= 1/(1 + 2*(2/1))
= 1/5
```

```
= 20 %


Throughput

= η * BW

= (1/5) * 6

= 1.2 Mbps
```

Note: As we can observe from the above given formula of Efficiency that:

1. On increasing the distance between source and receiver the Efficiency will decrease. Hence, Stop and Wait is only suitable for small area network like LAN. It is not suitable for MAN or WAN, as the efficiency will be very low.
2. If we increase the size of the Data packet, the efficiency is going to increase. Hence, it is suitable not for small packets. Big data packets can be send by Stop and Wait efficiently.