Sharing & Evaluating Networks

CPSC 433/533, Spring 2021

Anurag Khandelwal

Administrivia

- I hope by now, you have:
  - **All:** Checked Canvas for all the course policies and administrivia
  - **CPSC 533:** Found resources on how to read/critique papers

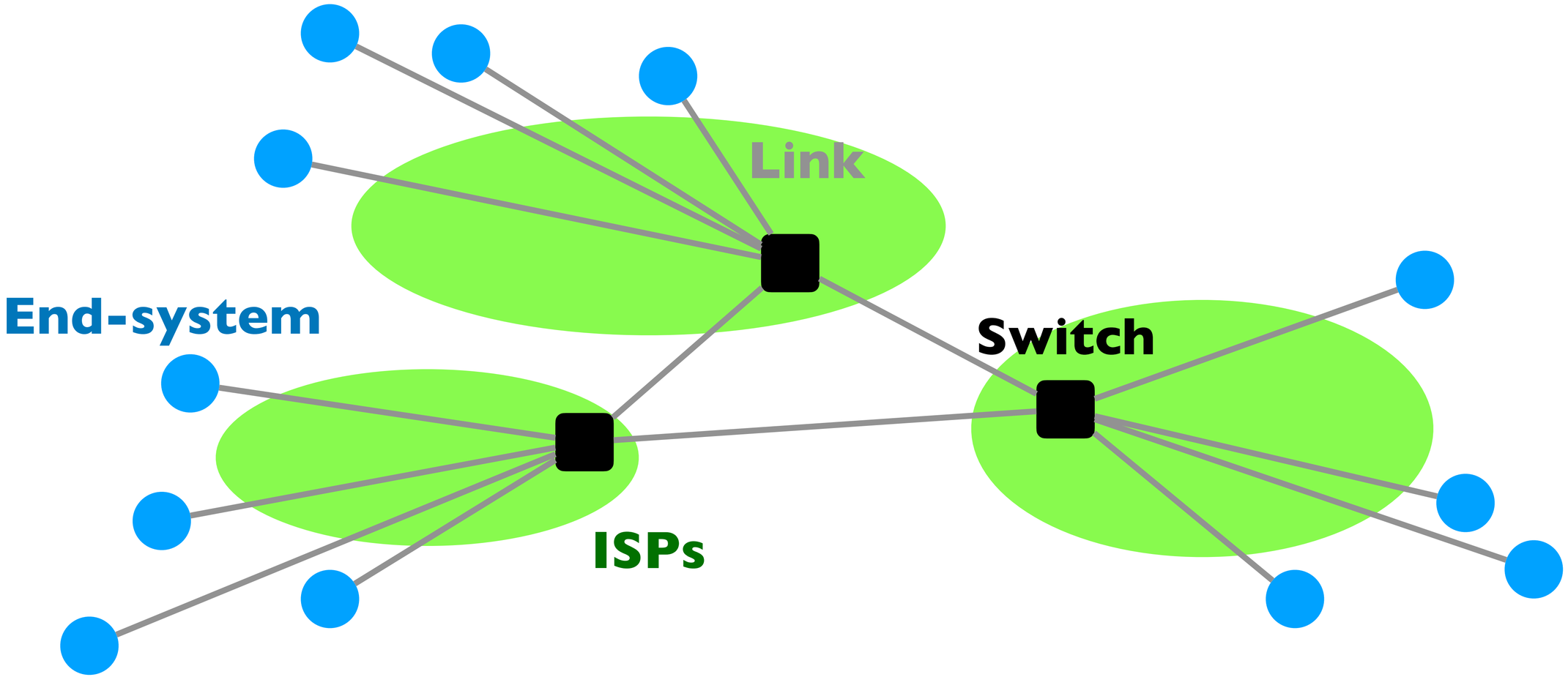
Today's Agenda

- More details on What & How of the Internet
- Sharing a network
- Evaluating a network

Today's Agenda

- More details on What, How and Why of the Internet
- Sharing a network
- Evaluating a network

The What



Digging Deeper into the “What”

The Last Mile

Digging Deeper into the “What”

The Last Mile

Edge Networks: The Last Mile

Telephony Access Technology

Edge Networks: The Last Mile

Telephony Access Technology

Edge Networks: The Last Mile

Telephony Access Technology

Edge Networks: The Last Mile

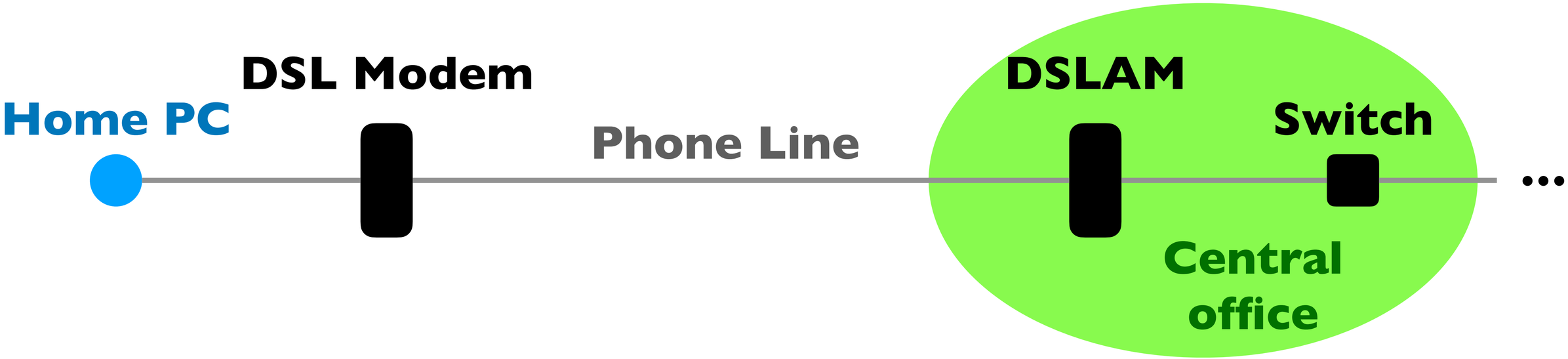
Telephony Access Technology

Edge Networks: The Last Mile

Telephony Access Technology

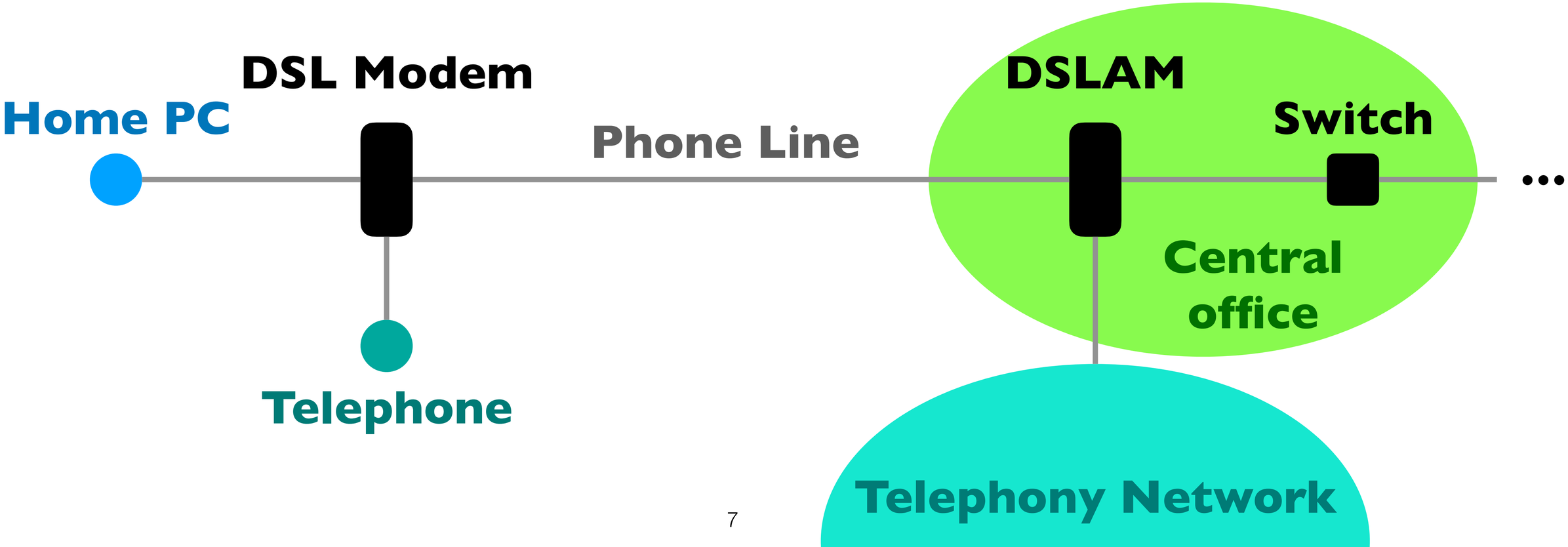
Edge Networks: The Last Mile

Telephony Access Technology



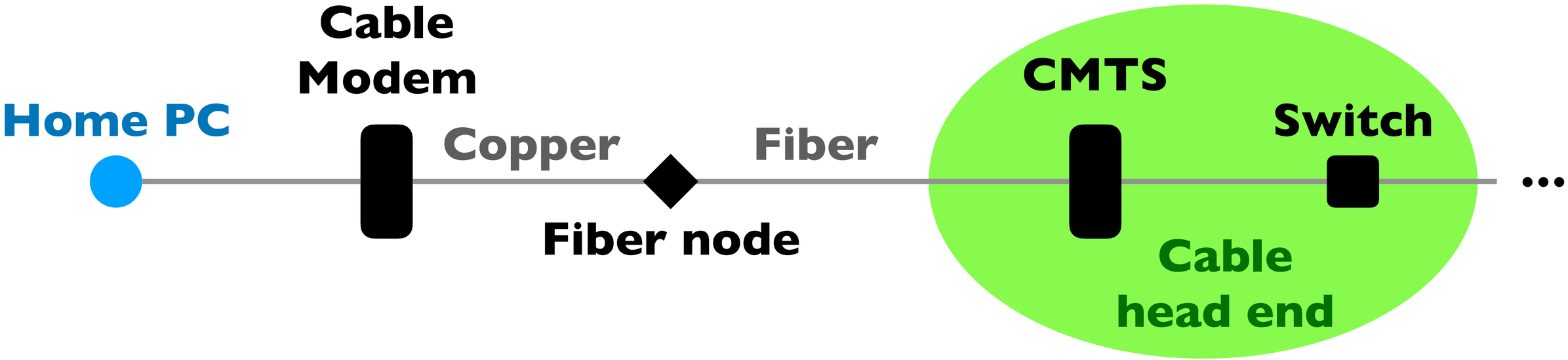
Edge Networks: The Last Mile

Telephony Access Technology



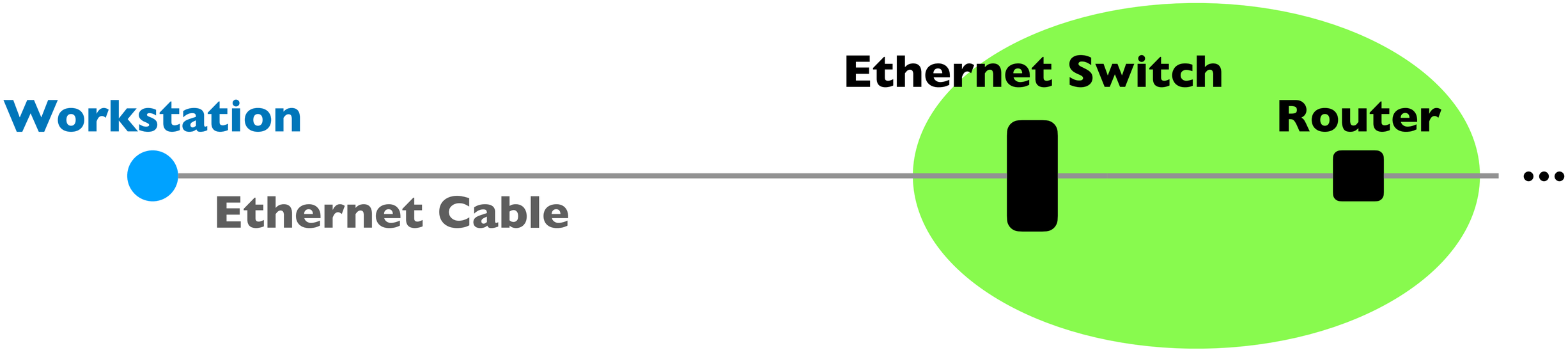
Edge Networks: The Last Mile

Cable Access Technology



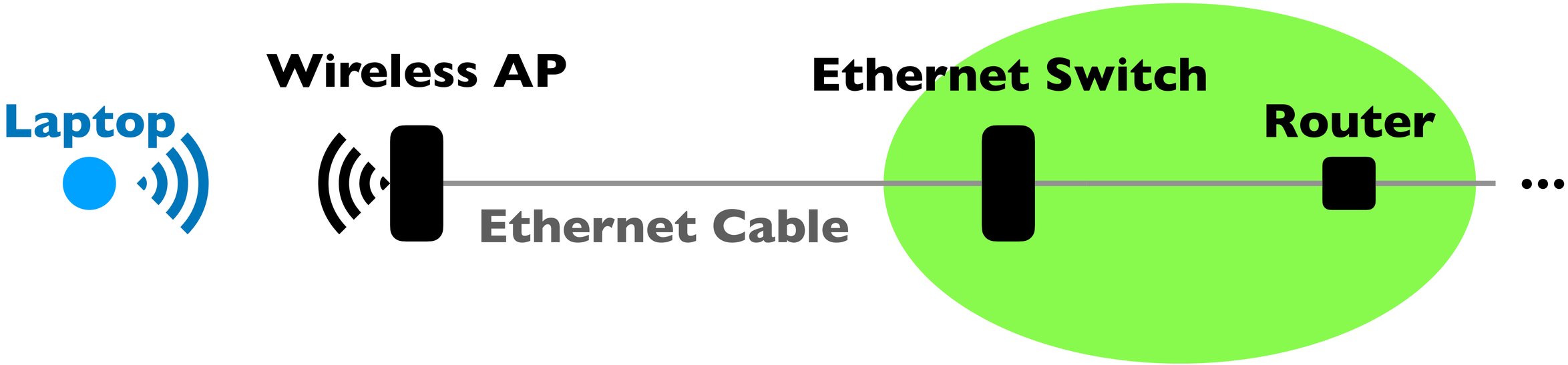
Edge Networks: The Last Mile

Ethernet Access Technology

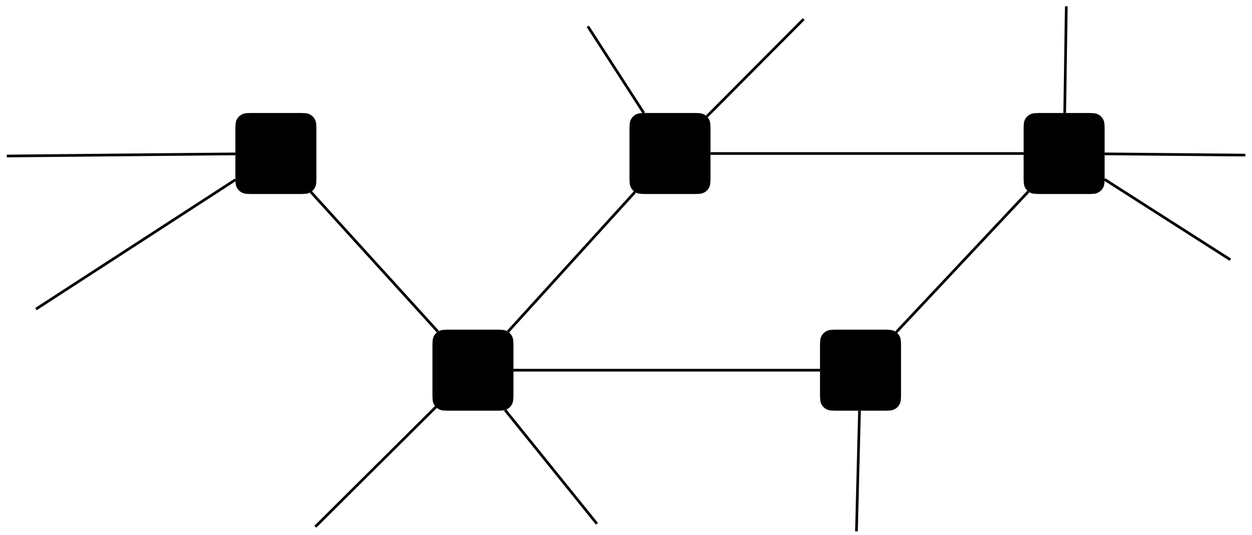


Edge Networks: The Last Mile

Ethernet Access Technology

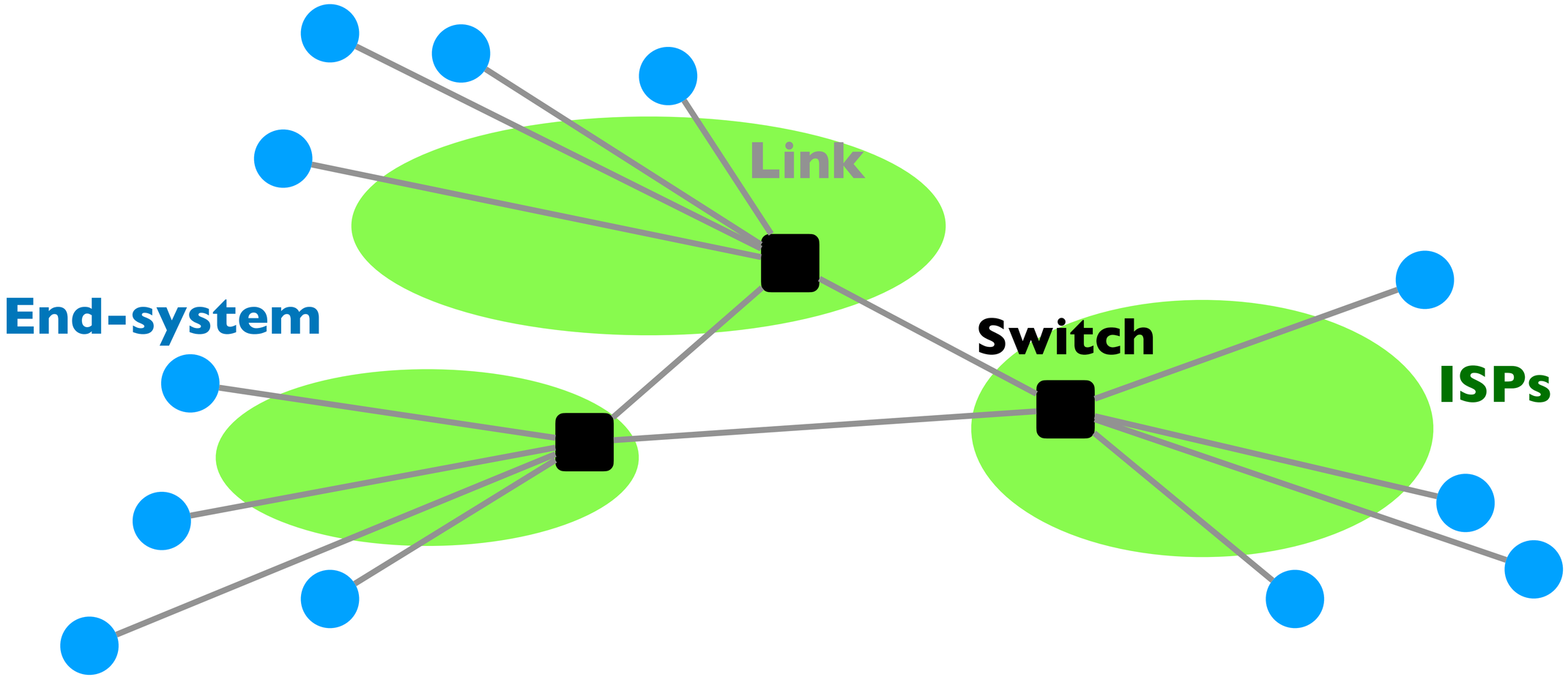


Core Network

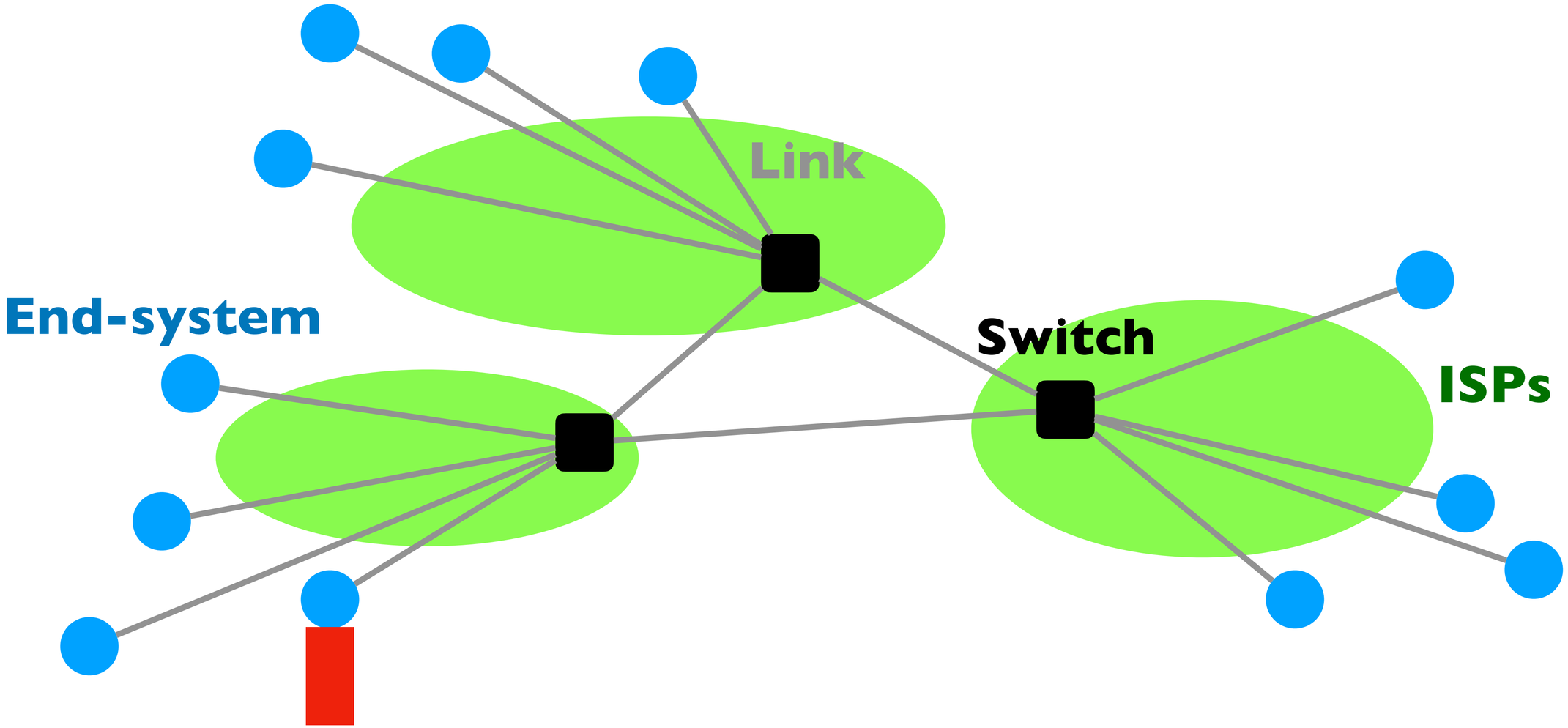


Switched networks enable efficient scaling!

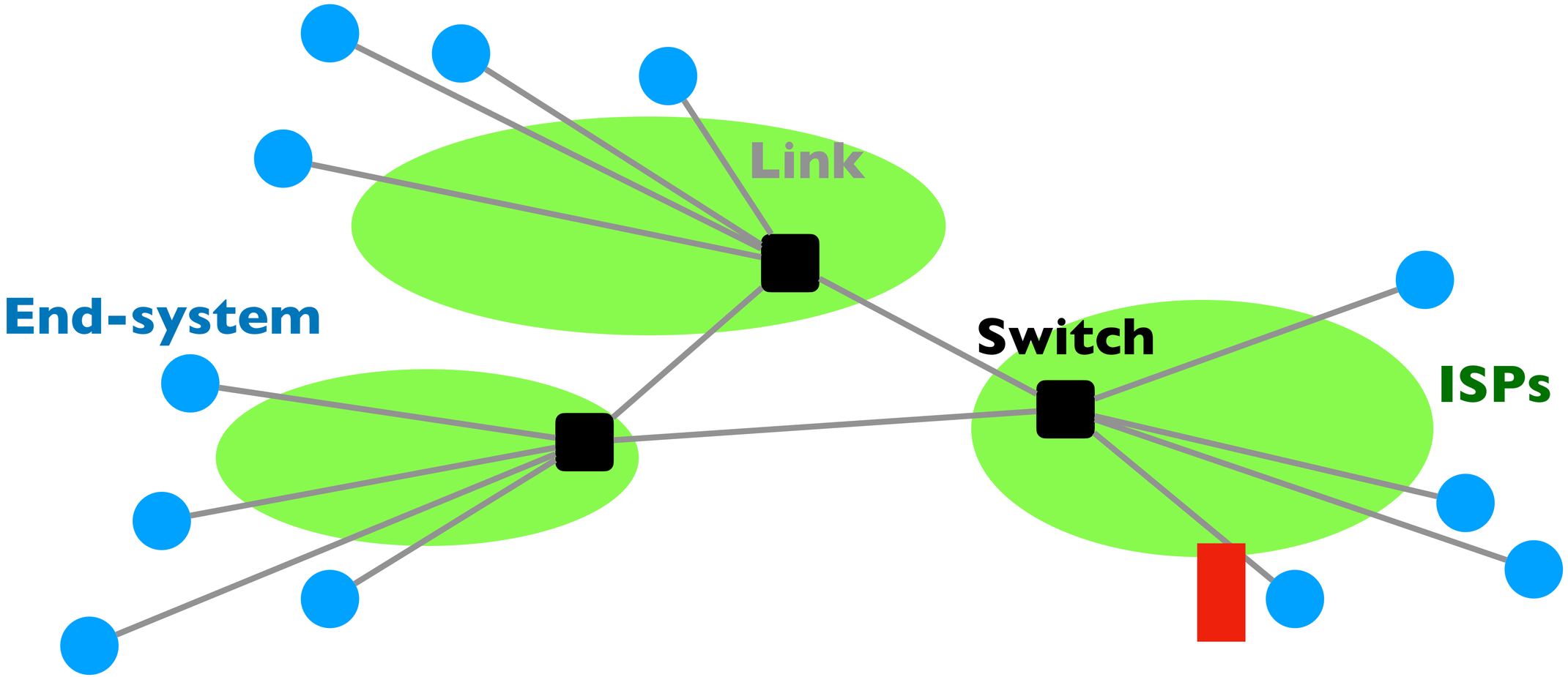
The How



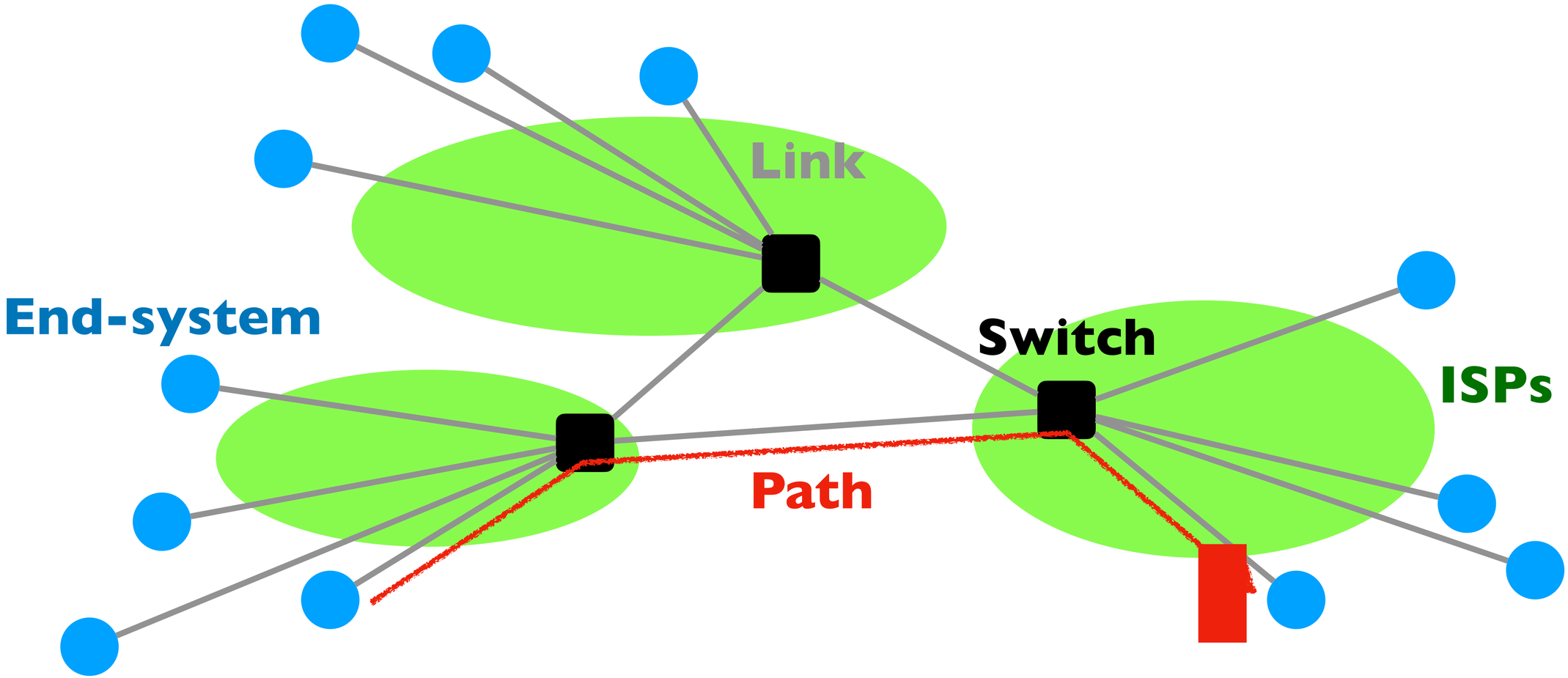
The How



The How



The How



Digging Deeper into the How

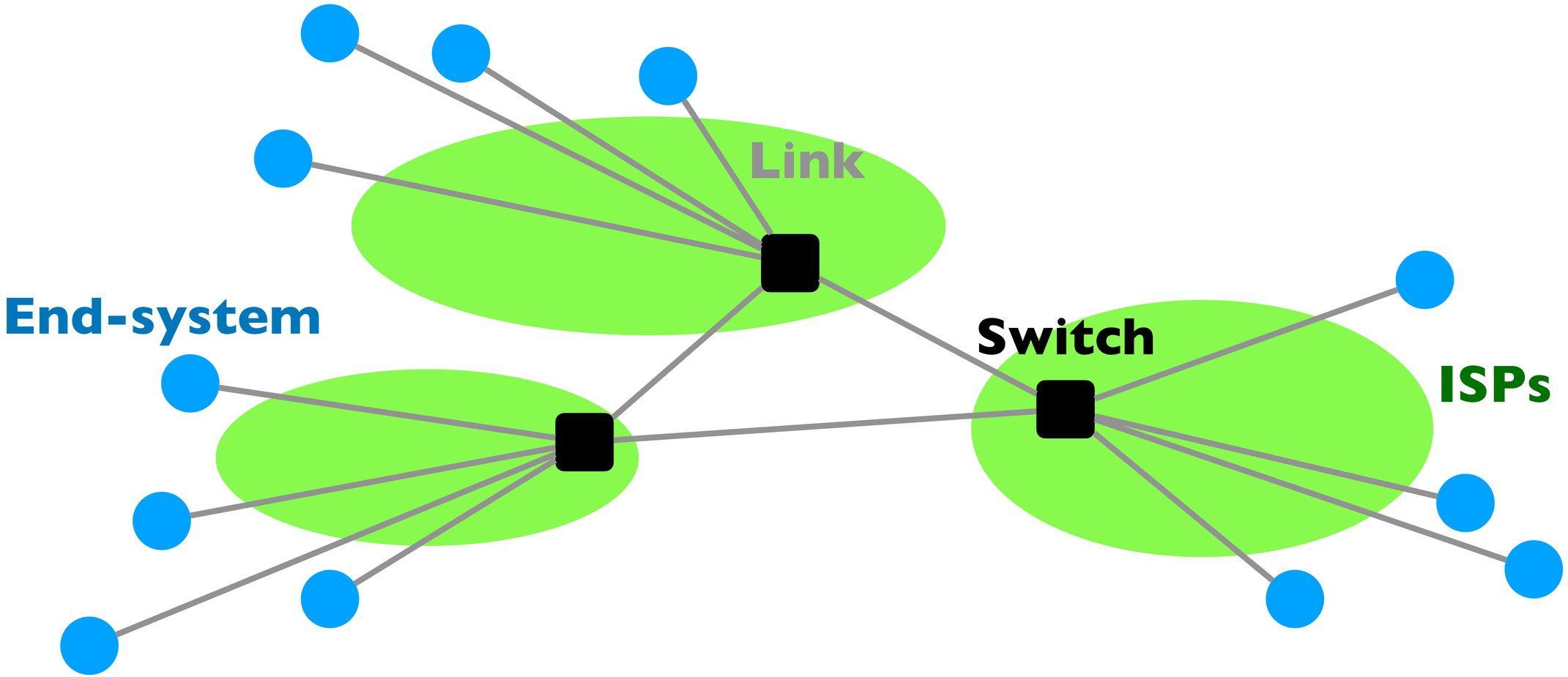
- **Locating the destination:** Naming, Addressing
- **Finding a path to the destination:** Routing
- **Sending data to the destination:** Forwarding
- **Handling cases where things go horribly wrong:** Distributed routing, congestion control
- We will spend (almost) the entire course to understand these concepts!

Questions?

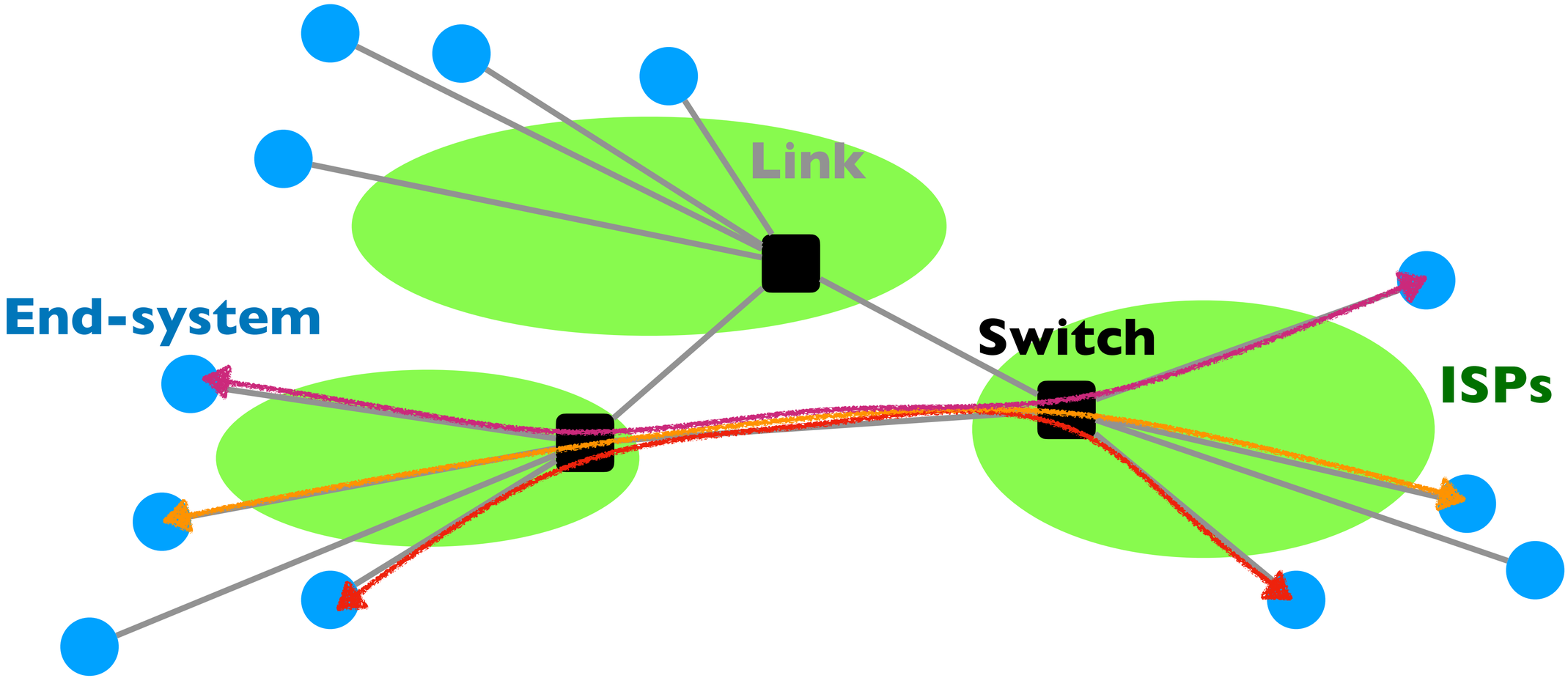
Today's Agenda

- More details on What & How of the Internet
- Sharing a network
- Evaluating a network

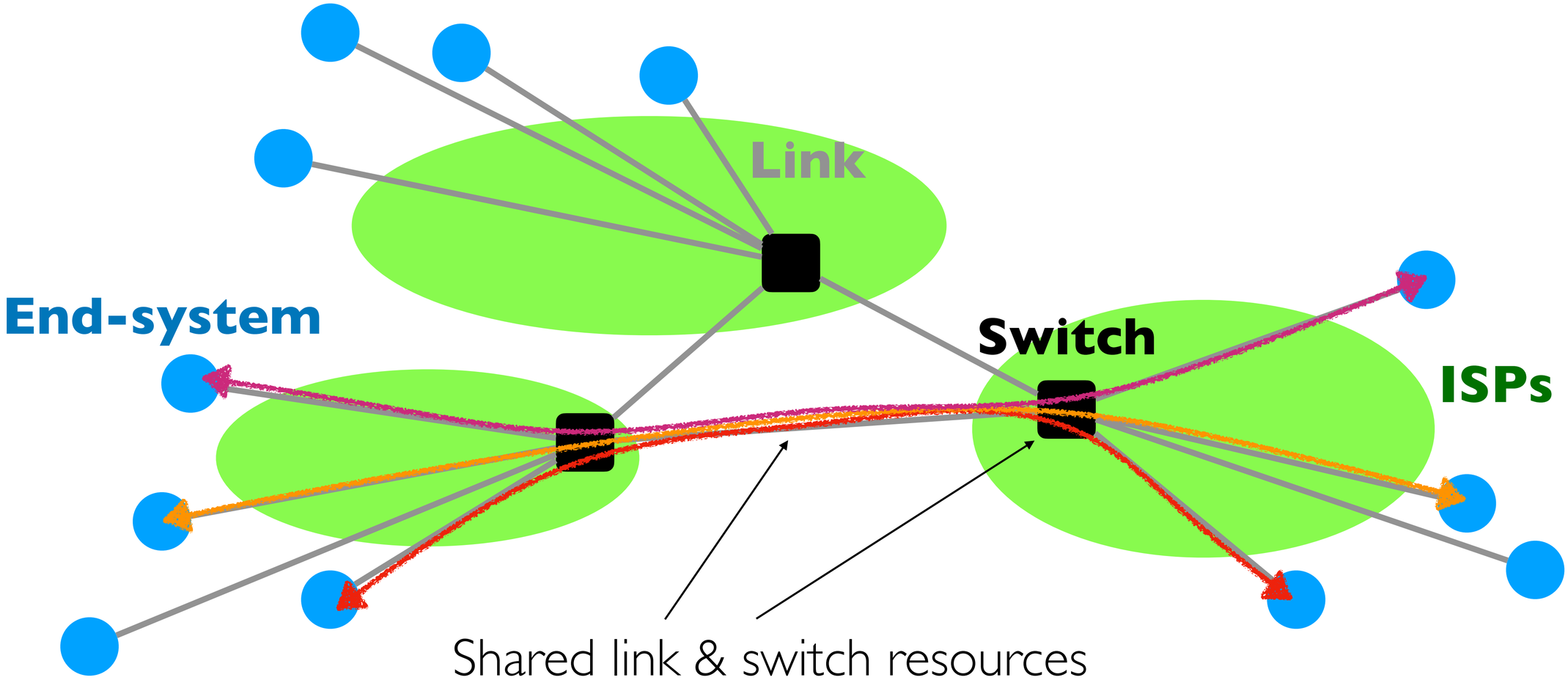
Sharing Network Resources



Sharing Network Resources



Sharing Network Resources



Sharing Network Resources

Sharing Network Resources

- What do we need to share?

Sharing Network Resources

- What do we need to share?
  - Switch processing resources, link bandwidths

Sharing Network Resources

- What do we need to share?
  - Switch processing resources, link bandwidths
  - **Link:** "Pipe" abstraction

Sharing Network Resources

- What do we need to share?
  - Switch processing resources, link bandwidths
  - **Link:** "Pipe" abstraction
  - **Bandwidth:** Number of bits sent per unit time

Sharing Network Resources

- What do we need to share?
  - Switch processing resources, link bandwidths
  - **Link:** "Pipe" abstraction
  - **Bandwidth:** Number of bits sent per unit time
  - **Network traffic:** "flows"

How can we share?

How can we share?

- Suggestions?

How can we share?

- Suggestions?
- First approach: **Reservations**

How can we share?

- **Suggestions?**
- First approach: **Reservations**
  - Reserve bandwidth needed in advance

How can we share?

- Suggestions?
- First approach: **Reservations**
  - Reserve bandwidth needed in advance
  - Must reserve at ***every*** link or switch in path

How can we share?

- Suggestions?
- First approach: **Reservations**
  - Reserve bandwidth needed in advance
  - Must reserve at ***every*** link or switch in path
  - Must reserve for ***peak*** bandwidth

How can we share?

- Suggestions?
- First approach: **Reservations**
  - Reserve bandwidth needed in advance
  - Must reserve at ***every*** link or switch in path
  - Must reserve for ***peak*** bandwidth
- **Peak** bandwidth?

How can we share?

- **Suggestions?**
- First approach: **Reservations**
  - Reserve bandwidth needed in advance
  - Must reserve at ***every*** link or switch in path
  - Must reserve for ***peak*** bandwidth
- **Peak** bandwidth?
  - An app may generate data at varying rates

How can we share?

- Suggestions?
- First approach: **Reservations**
  - Reserve bandwidth needed in advance
  - Must reserve at ***every*** link or switch in path
  - Must reserve for ***peak*** bandwidth
- **Peak** bandwidth?
  - An app may generate data at varying rates
  - 100MB in the 1st second, 10MB in the second, etc.

How can we share?

- Suggestions?
- First approach: **Reservations**
  - Reserve bandwidth needed in advance
  - Must reserve at ***every*** link or switch in path
  - Must reserve for ***peak*** bandwidth
- **Peak** bandwidth?
  - An app may generate data at varying rates
  - 100MB in the 1st second, 10MB in the second, etc.
  - Must reserve for peak (e.g., 100MB/s)

Reservation: Circuit Switching

Reservation: Circuit Switching

- One way of implementing reservations in networks

Reservation: Circuit Switching

- One way of implementing reservations in networks
- How old-school telephone networks worked

Reservation: Circuit Switching

- One way of implementing reservations in networks
- How old-school telephone networks worked
  - My grandma used to tell me stories about this

Reservation: Circuit Switching

- One way of implementing reservations in networks
- How old-school telephone networks worked
  - My grandma used to tell me stories about this

Grandma

Reservation: Circuit Switching

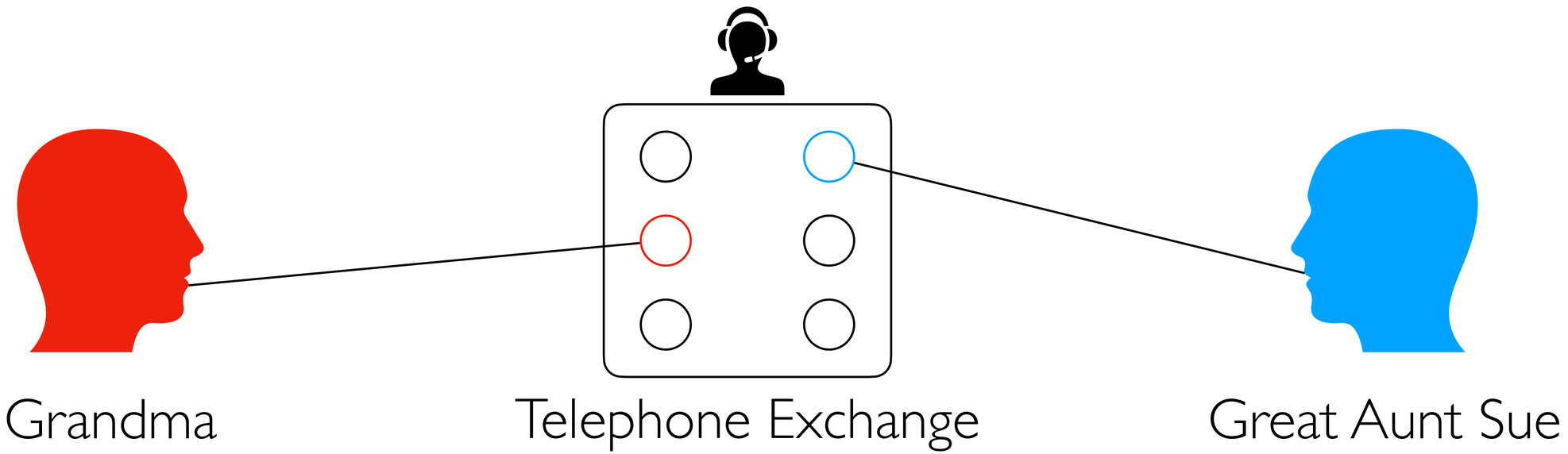
- One way of implementing reservations in networks
- How old-school telephone networks worked
  - My grandma used to tell me stories about this

Grandma

Great Aunt Sue

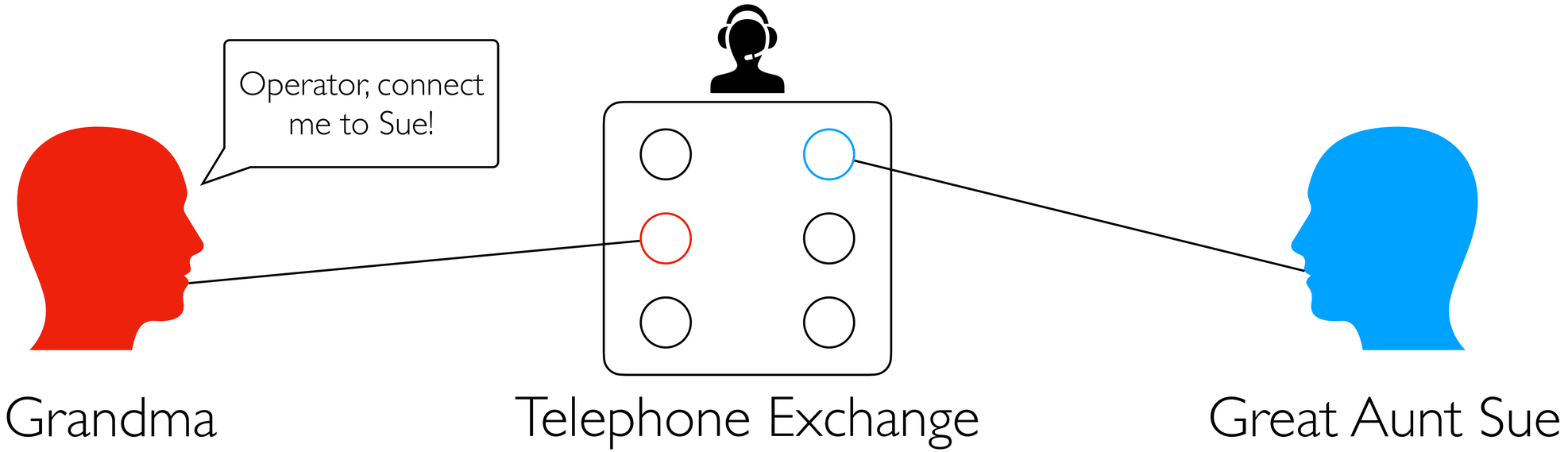
Reservation: Circuit Switching

- One way of implementing reservations in networks
- How old-school telephone networks worked
  - My grandma used to tell me stories about this



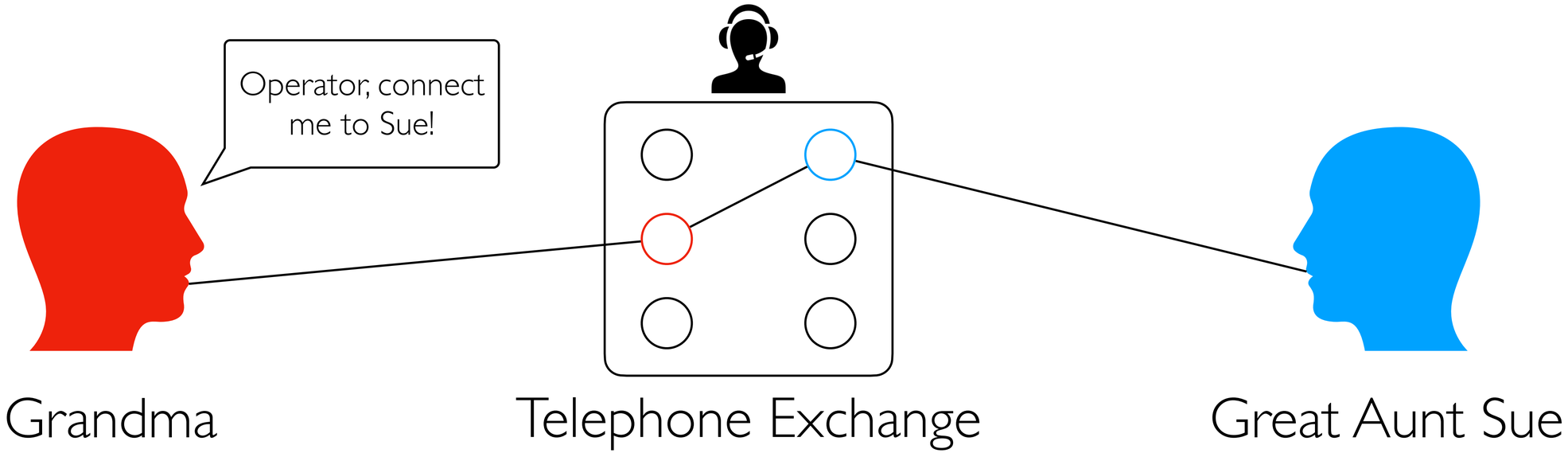
Reservation: Circuit Switching

- One way of implementing reservations in networks
- How old-school telephone networks worked
  - My grandma used to tell me stories about this



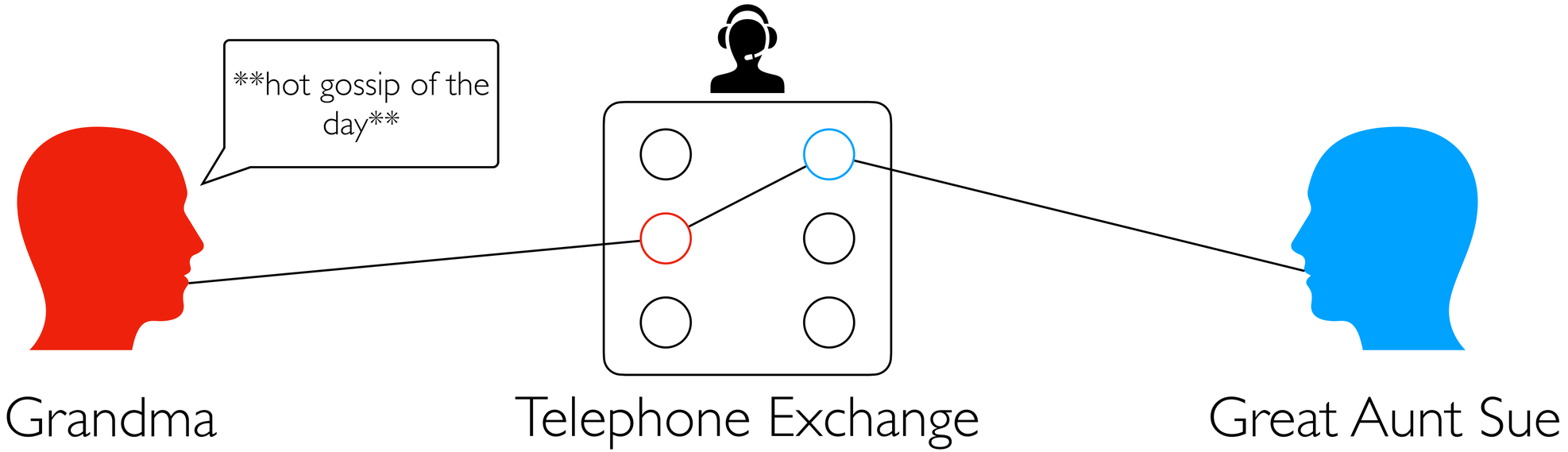
Reservation: Circuit Switching

- One way of implementing reservations in networks
- How old-school telephone networks worked
  - My grandma used to tell me stories about this



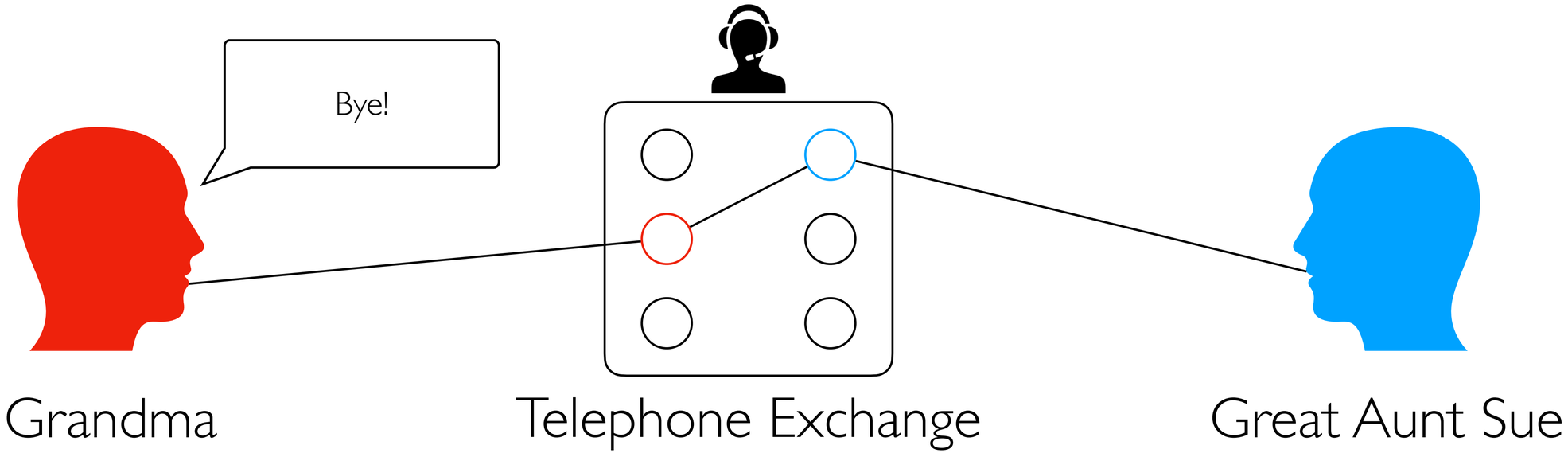
Reservation: Circuit Switching

- One way of implementing reservations in networks
- How old-school telephone networks worked
  - My grandma used to tell me stories about this



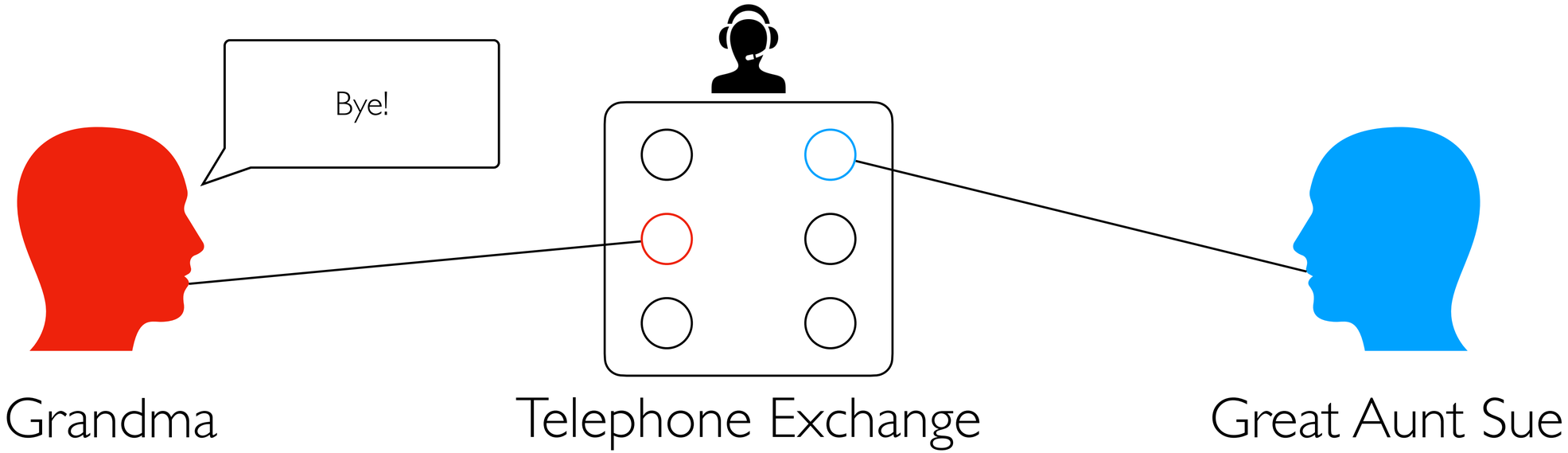
Reservation: Circuit Switching

- One way of implementing reservations in networks
- How old-school telephone networks worked
  - My grandma used to tell me stories about this

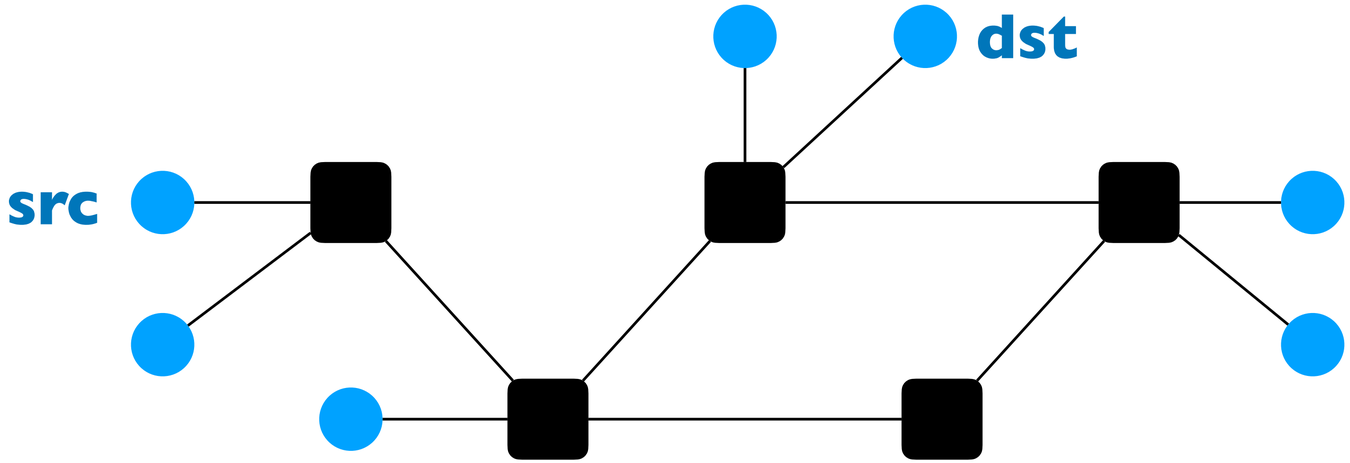


Reservation: Circuit Switching

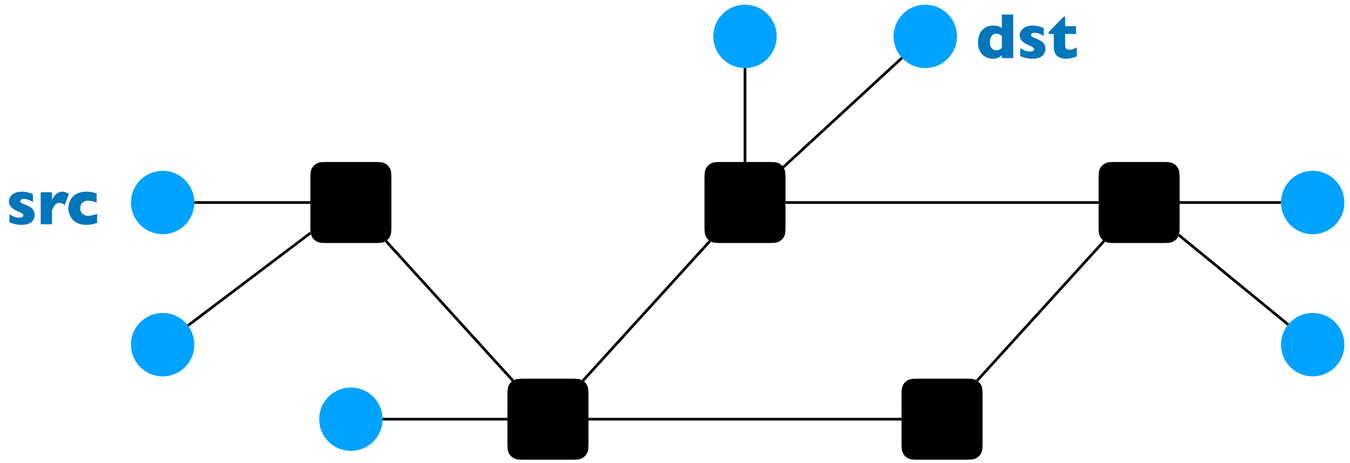
- One way of implementing reservations in networks
- How old-school telephone networks worked
  - My grandma used to tell me stories about this



Circuit Switching

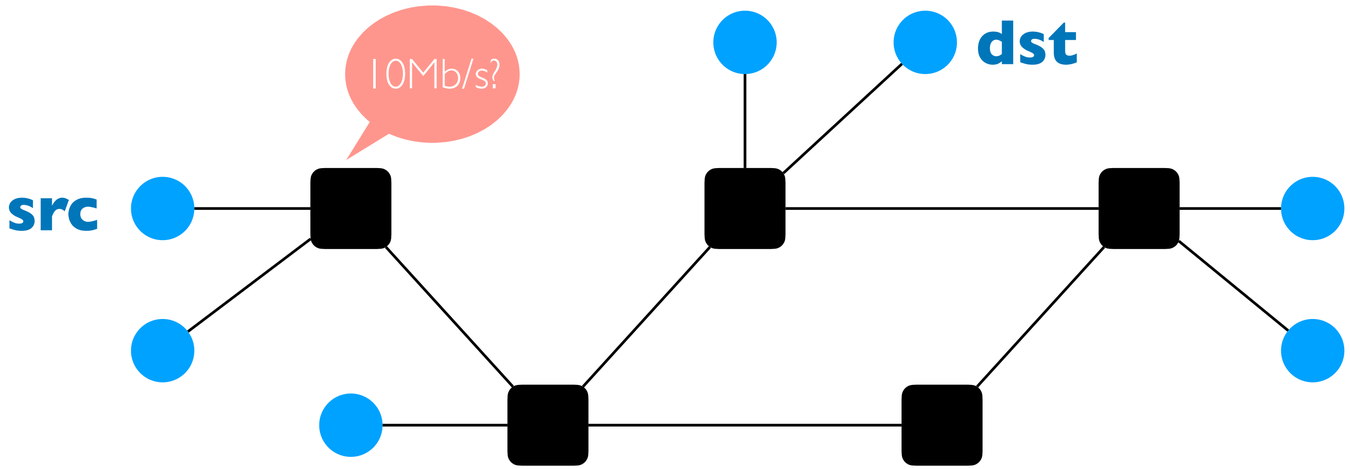


Circuit Switching



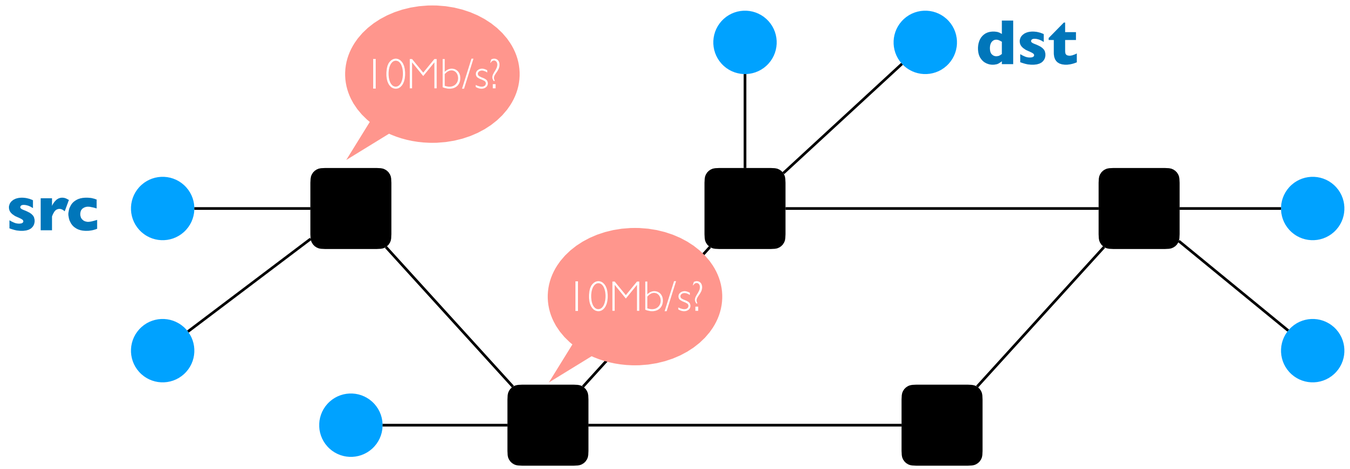
1. The **src** sends a reservation request to **dst**

Circuit Switching



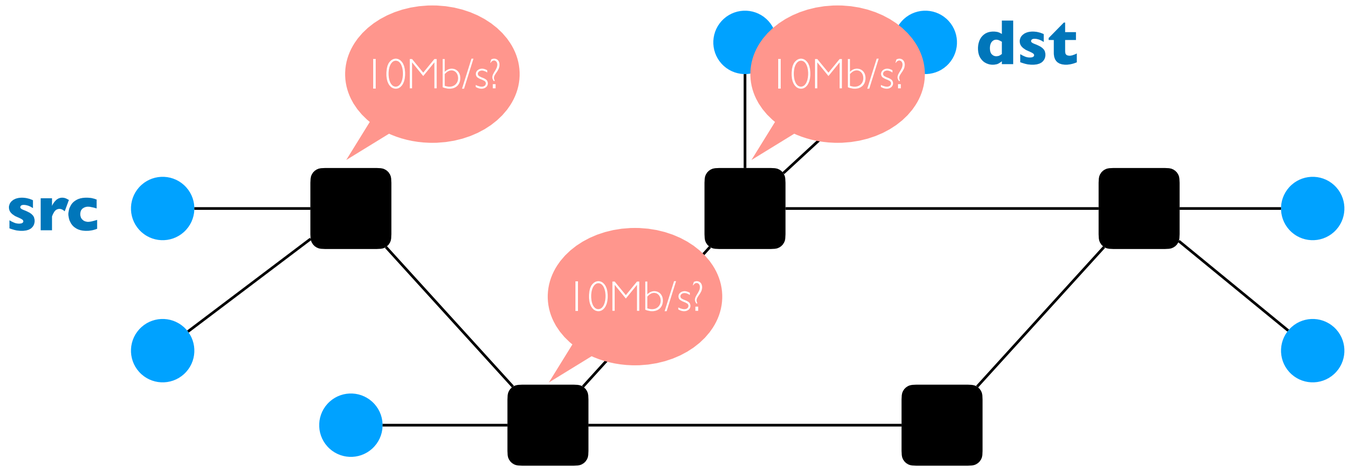
1. The **src** sends a reservation request to **dst**

Circuit Switching



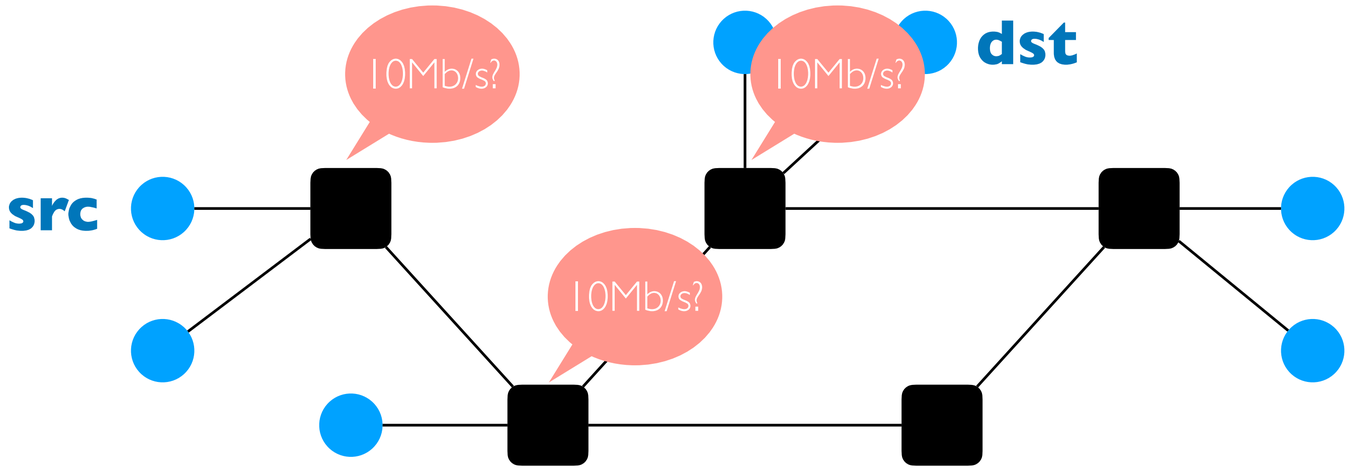
1. The **src** sends a reservation request to **dst**

Circuit Switching



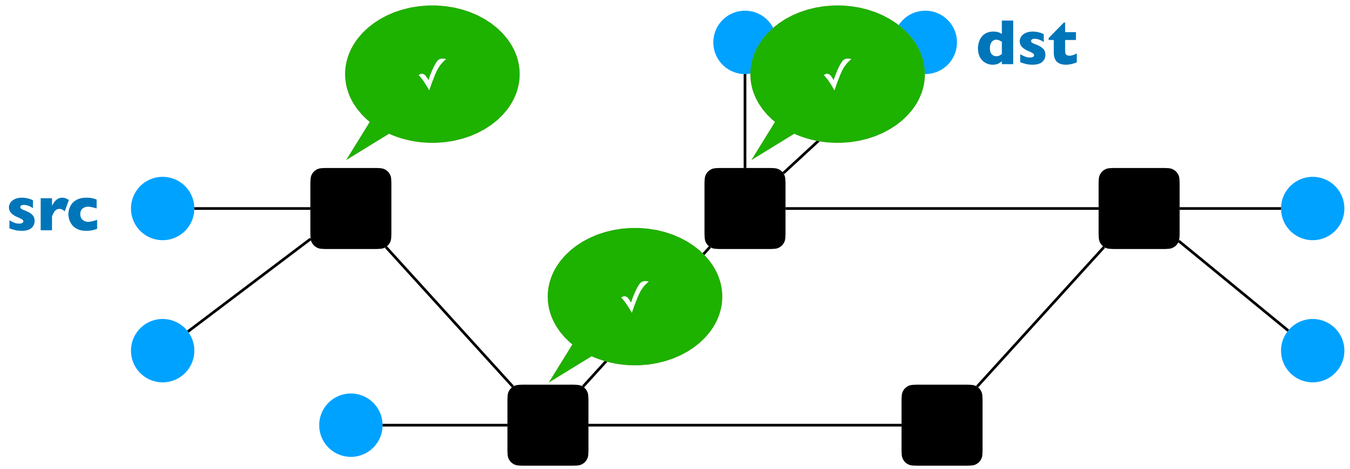
1. The **src** sends a reservation request to **dst**

Circuit Switching



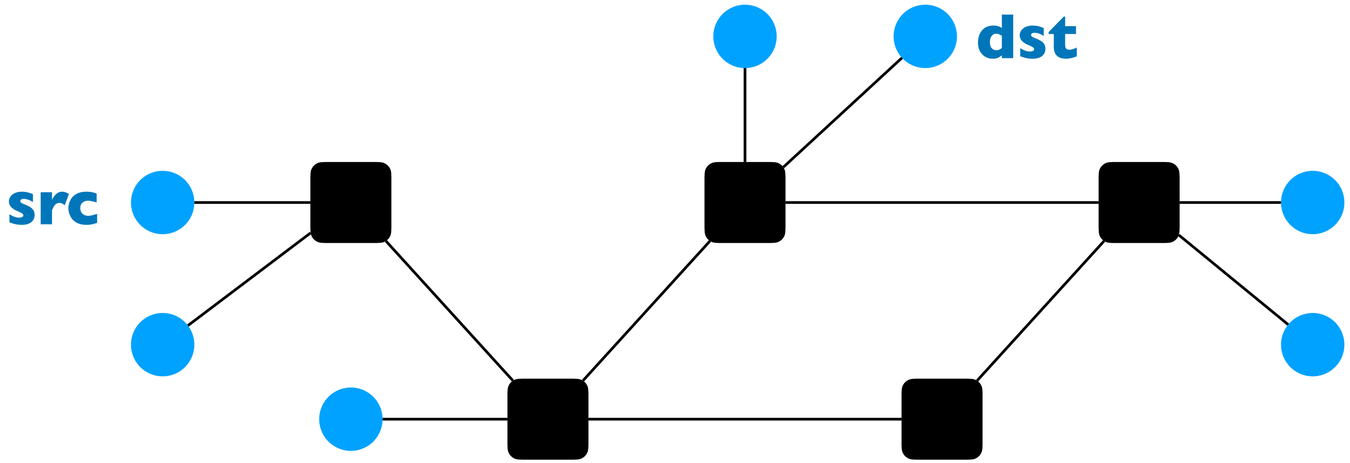
1. The **src** sends a reservation request to **dst**
2. Switches "establish a circuit"

Circuit Switching



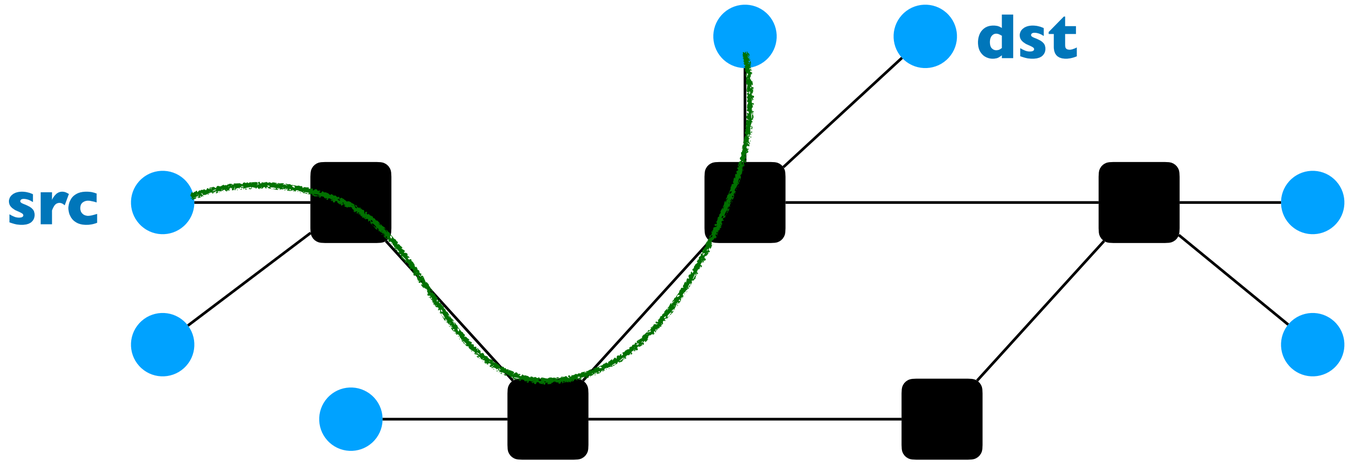
1. The **src** sends a reservation request to **dst**
2. Switches "establish a circuit"

Circuit Switching



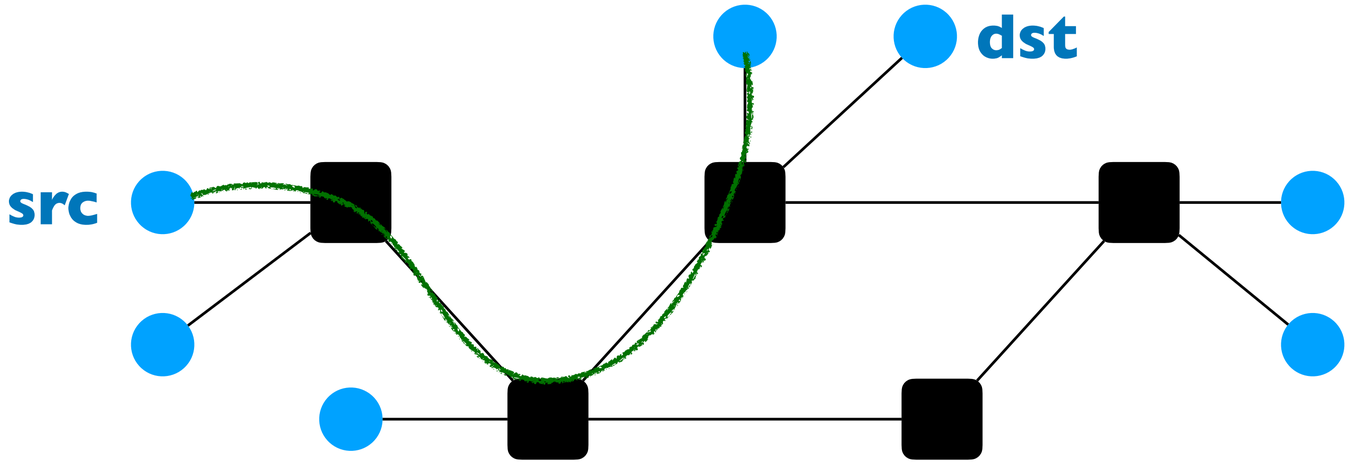
1. The **src** sends a reservation request to **dst**
2. Switches "establish a circuit"

Circuit Switching



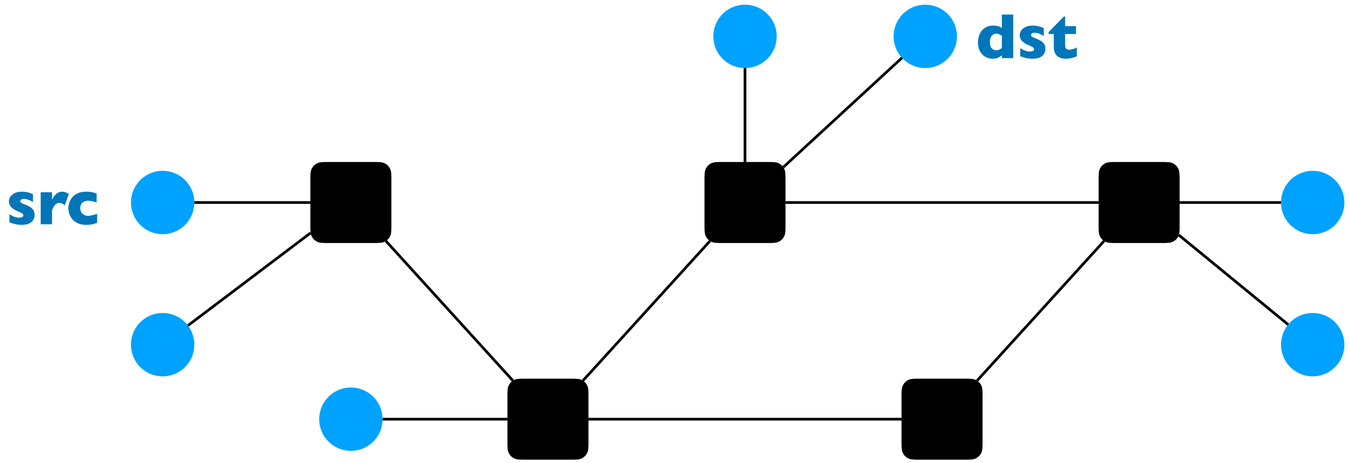
1. The **src** sends a reservation request to **dst**
2. Switches "establish a circuit"

Circuit Switching



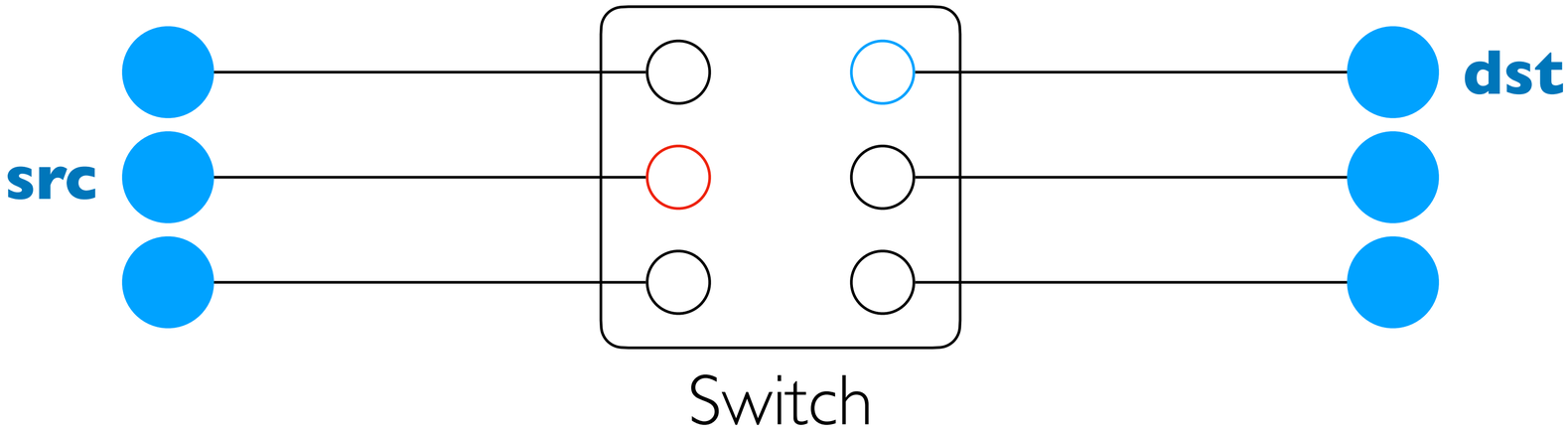
1. The **src** sends a reservation request to **dst**
2. Switches "establish a circuit"
3. The **src** starts sending data

Circuit Switching

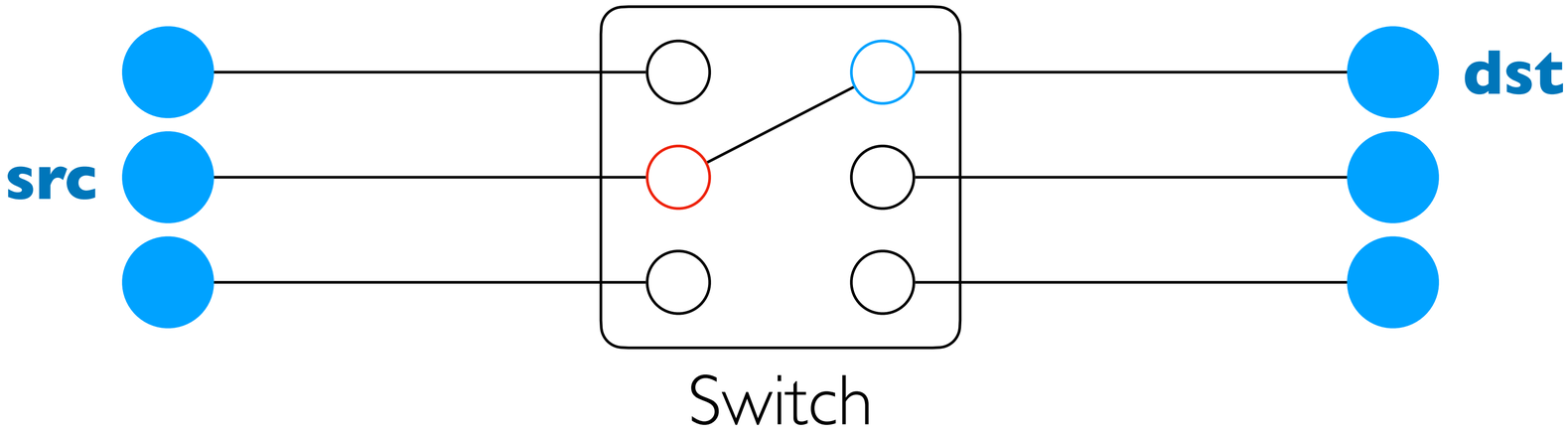


1. The **src** sends a reservation request to **dst**
2. Switches "establish a circuit"
3. The **src** starts sending data
4. The **src** sends a "teardown circuit" message

Reservation: Circuit Switching

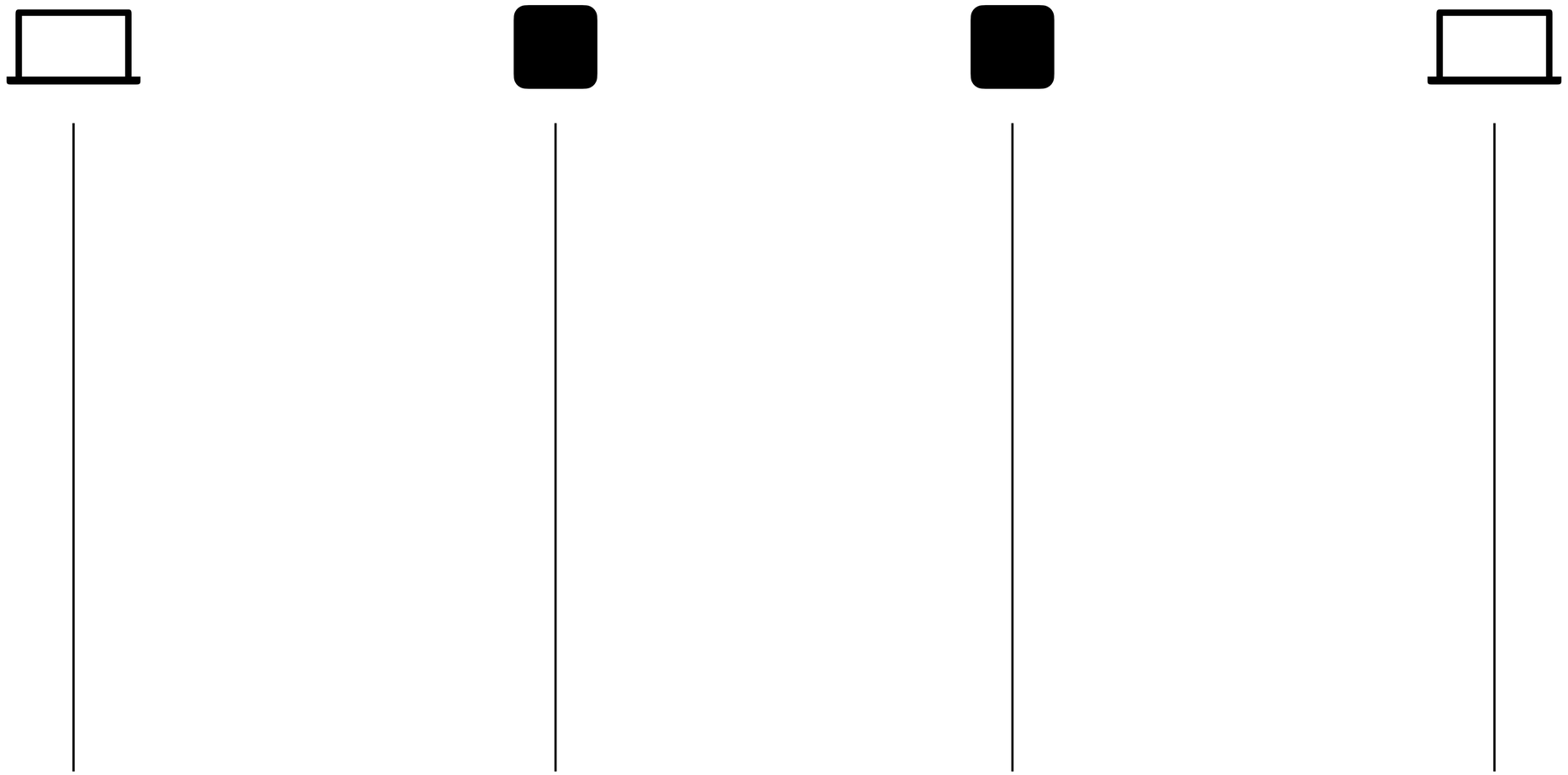


Reservation: Circuit Switching

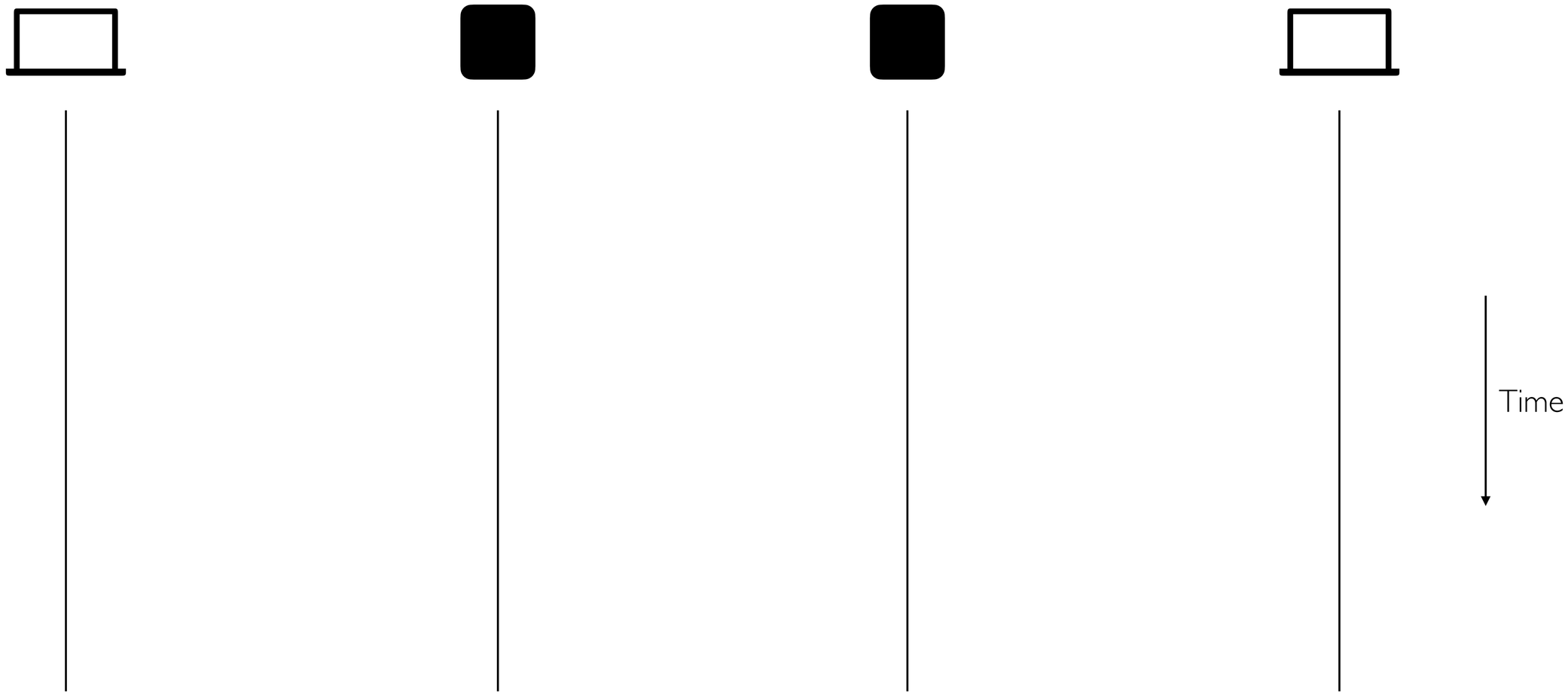


Reservation establishes a "circuit" within a switch

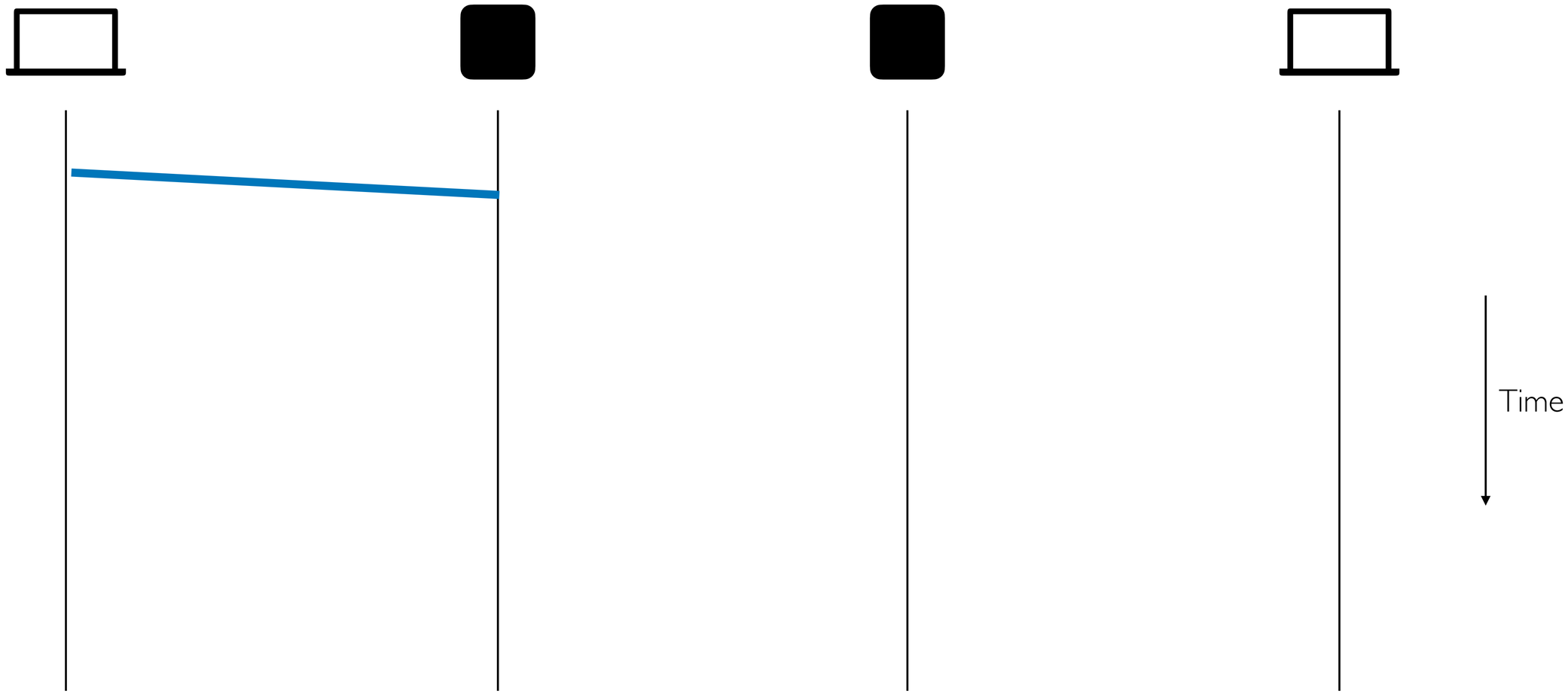
Timing in Circuit Switching



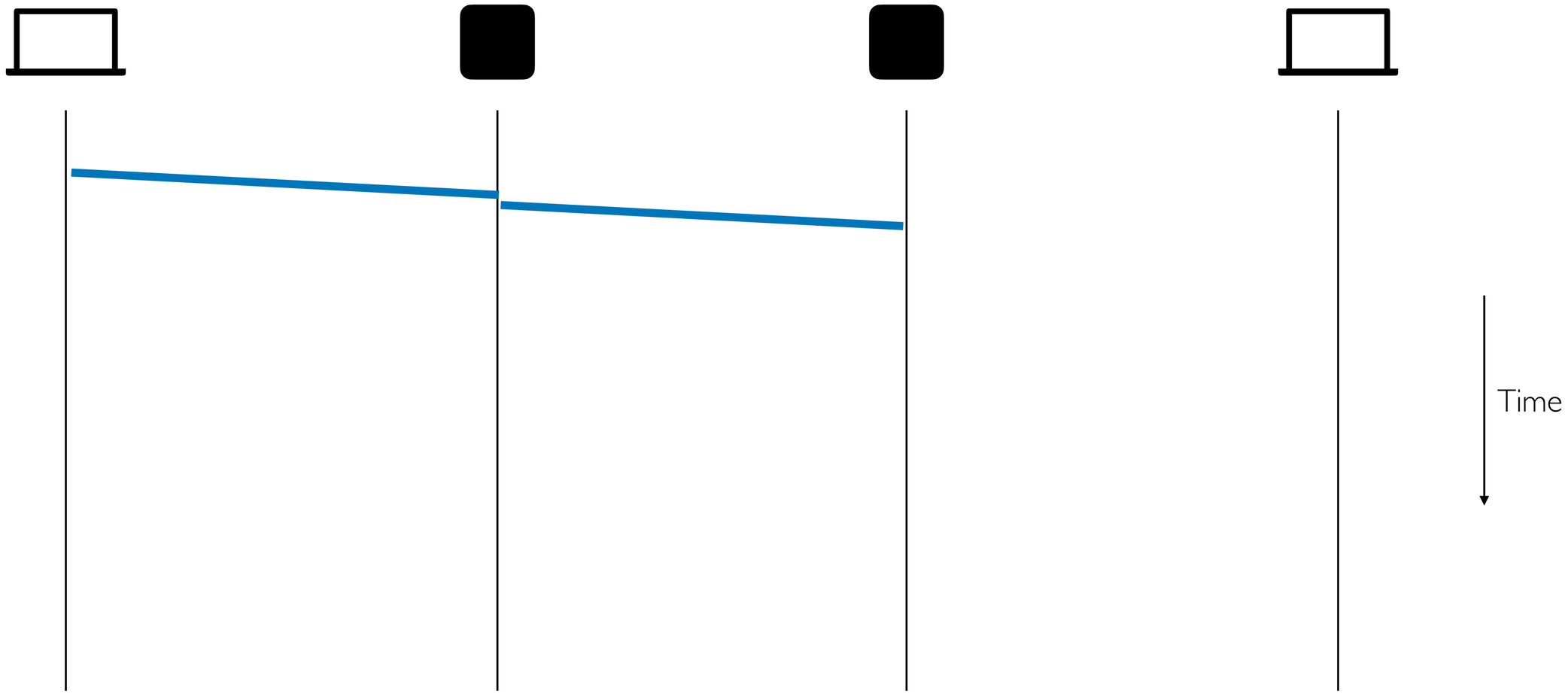
Timing in Circuit Switching



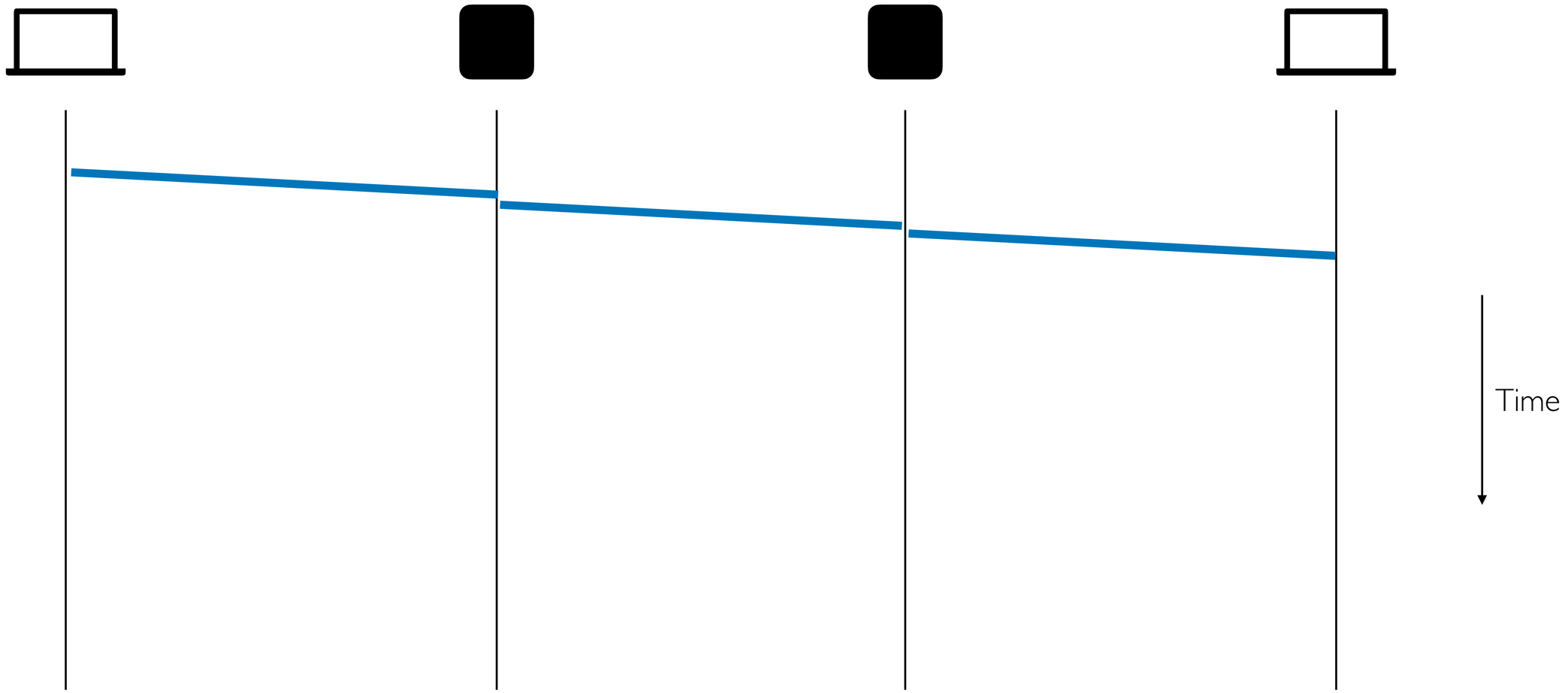
Timing in Circuit Switching



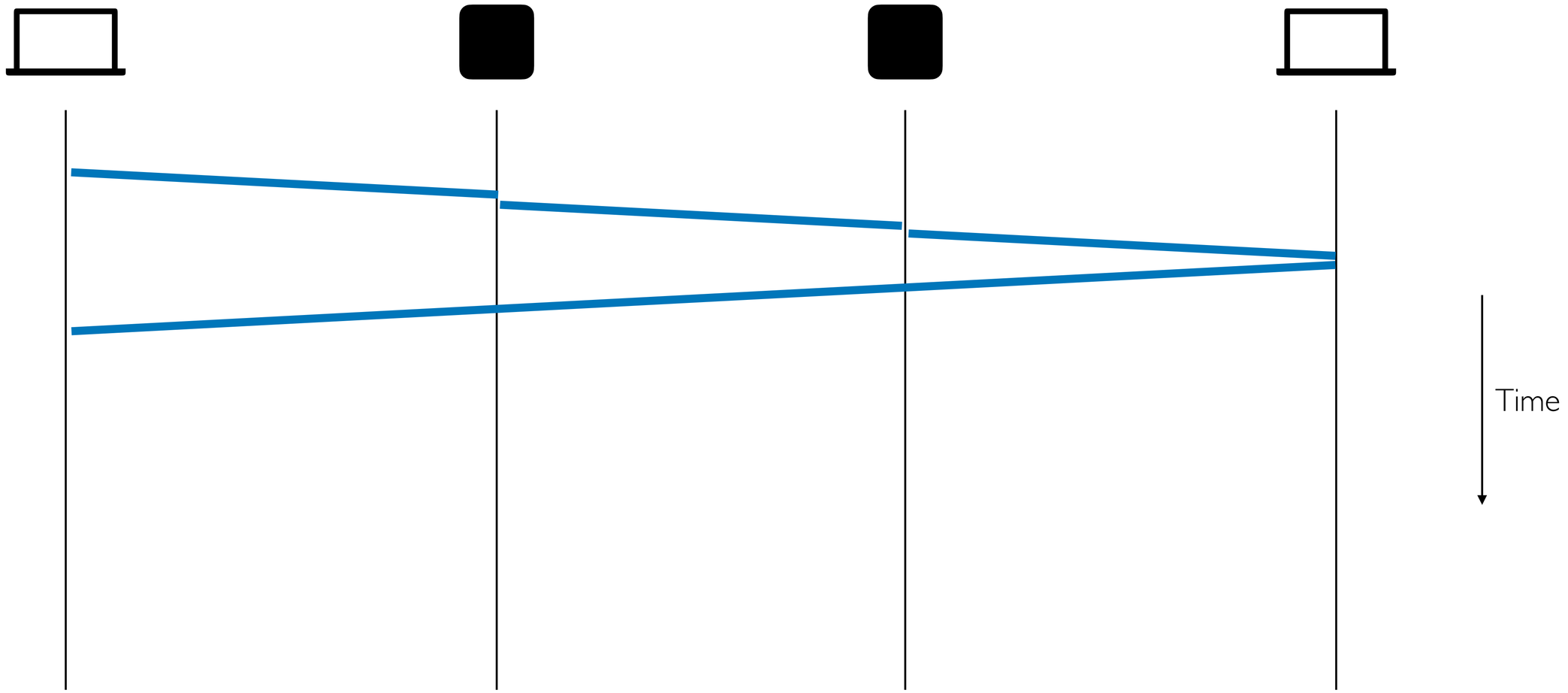
Timing in Circuit Switching



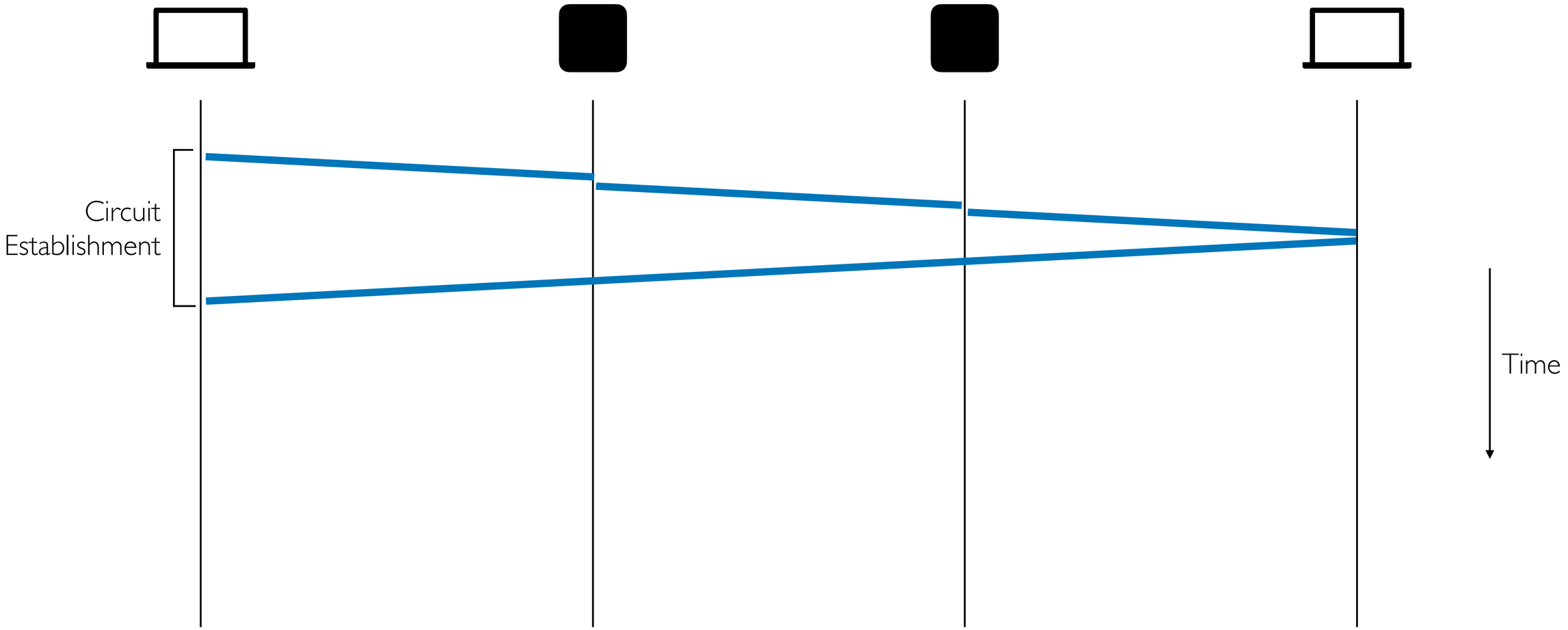
Timing in Circuit Switching



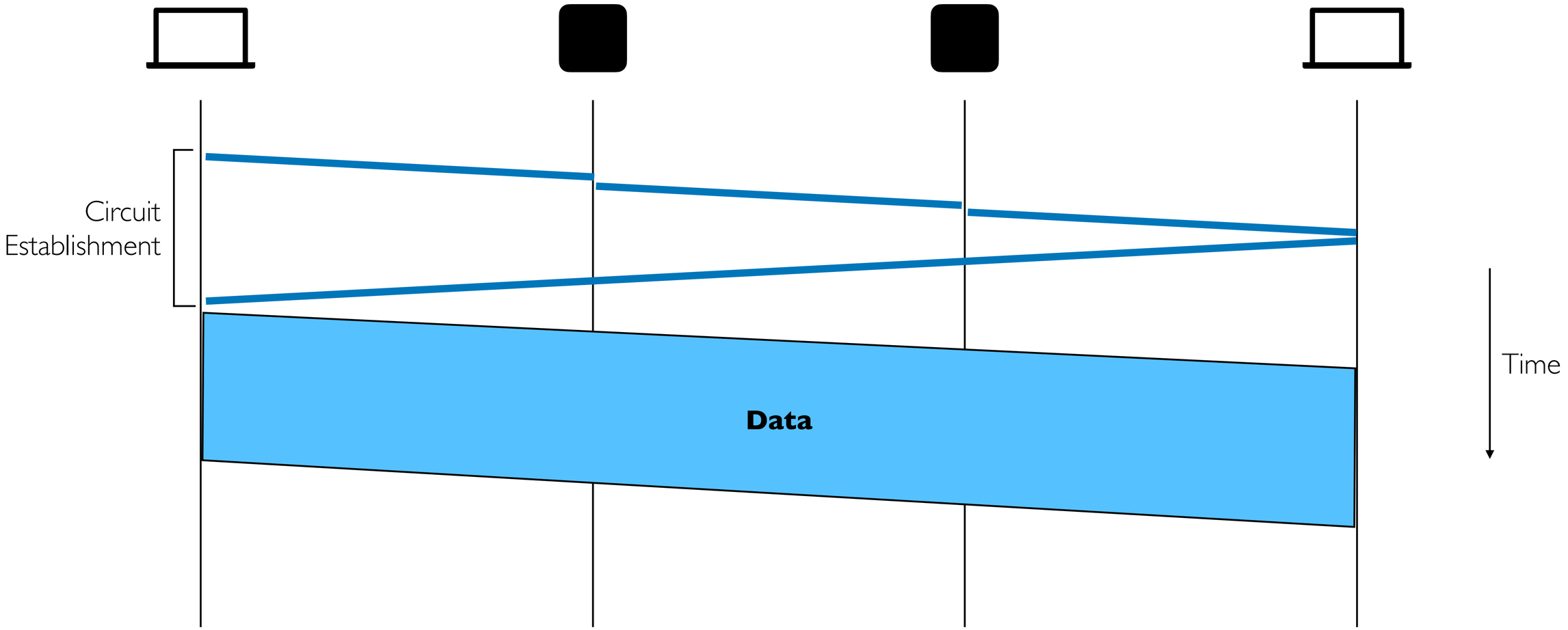
Timing in Circuit Switching



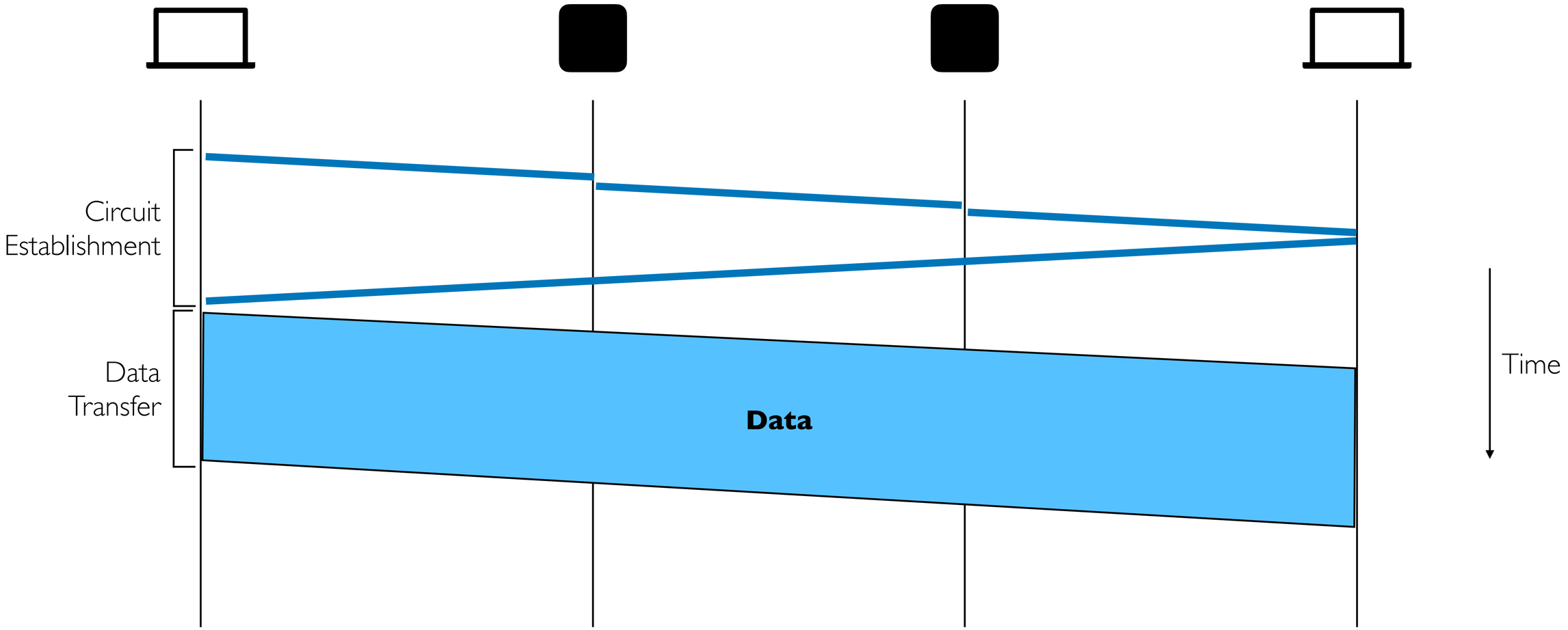
Timing in Circuit Switching



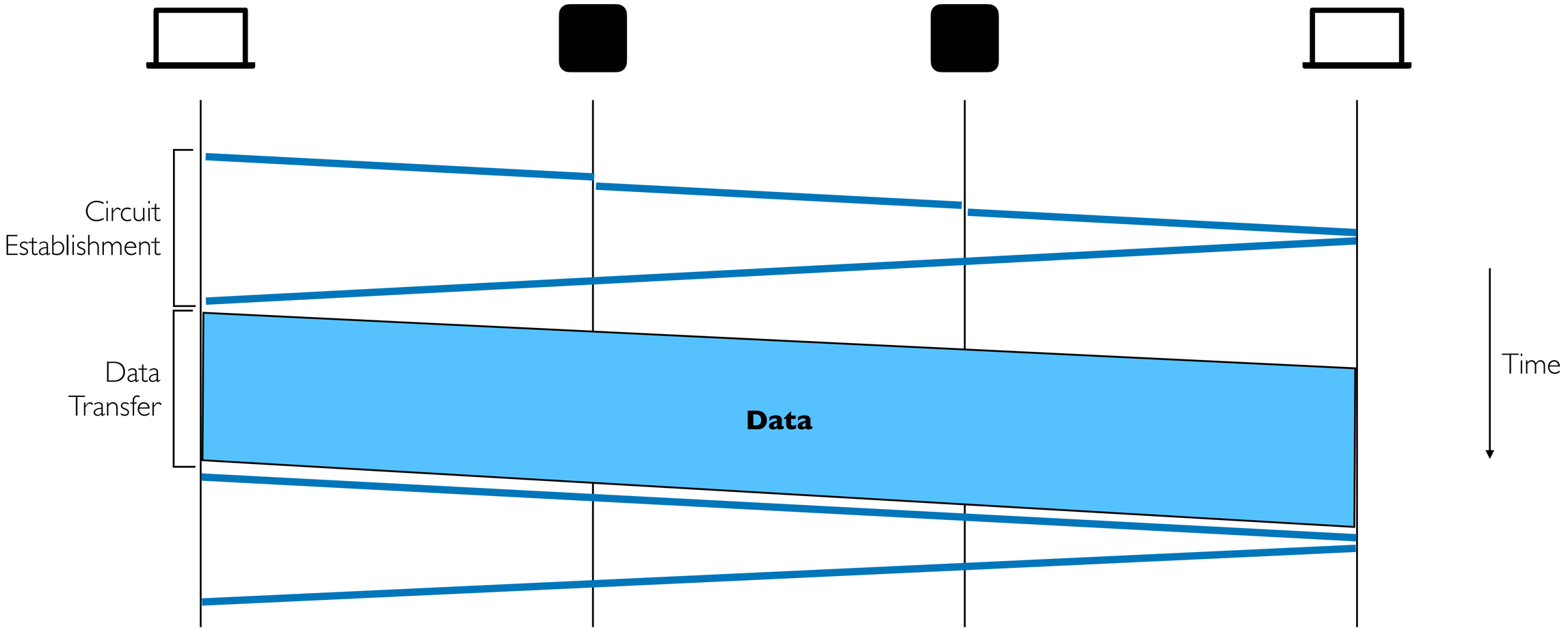
Timing in Circuit Switching



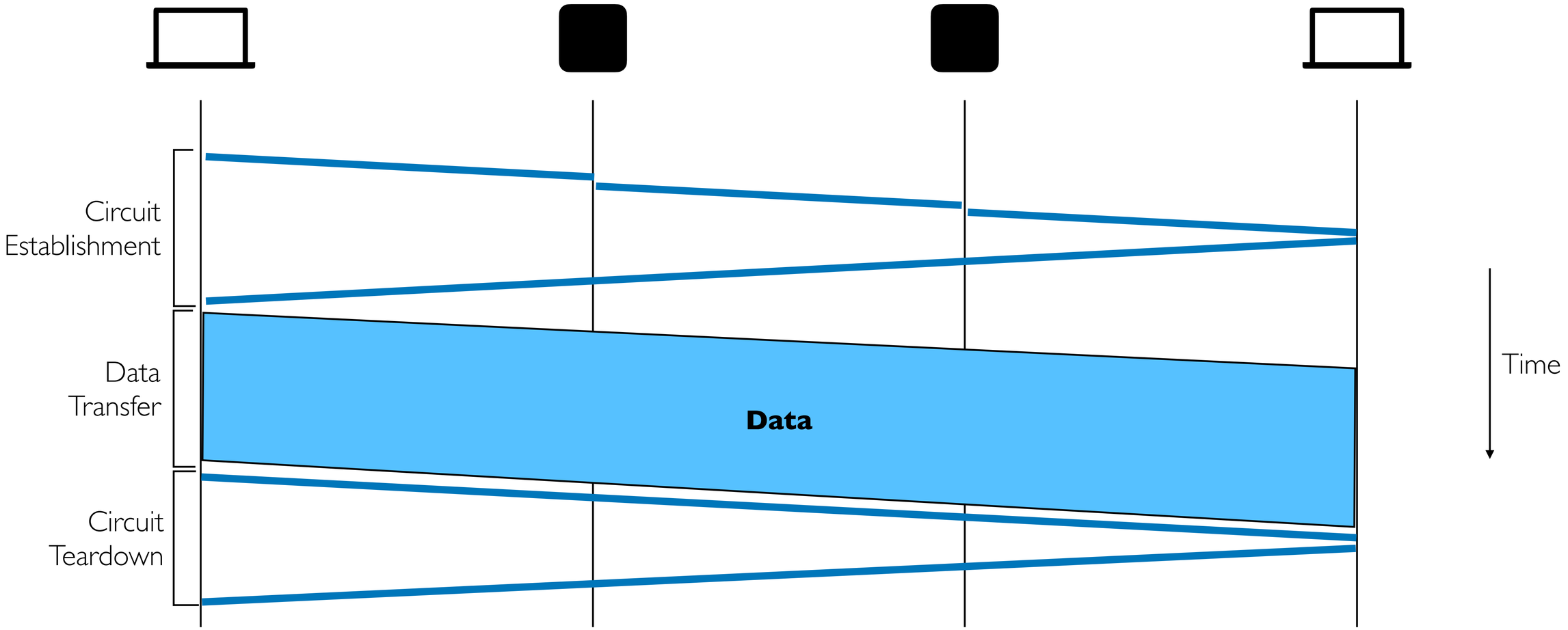
Timing in Circuit Switching



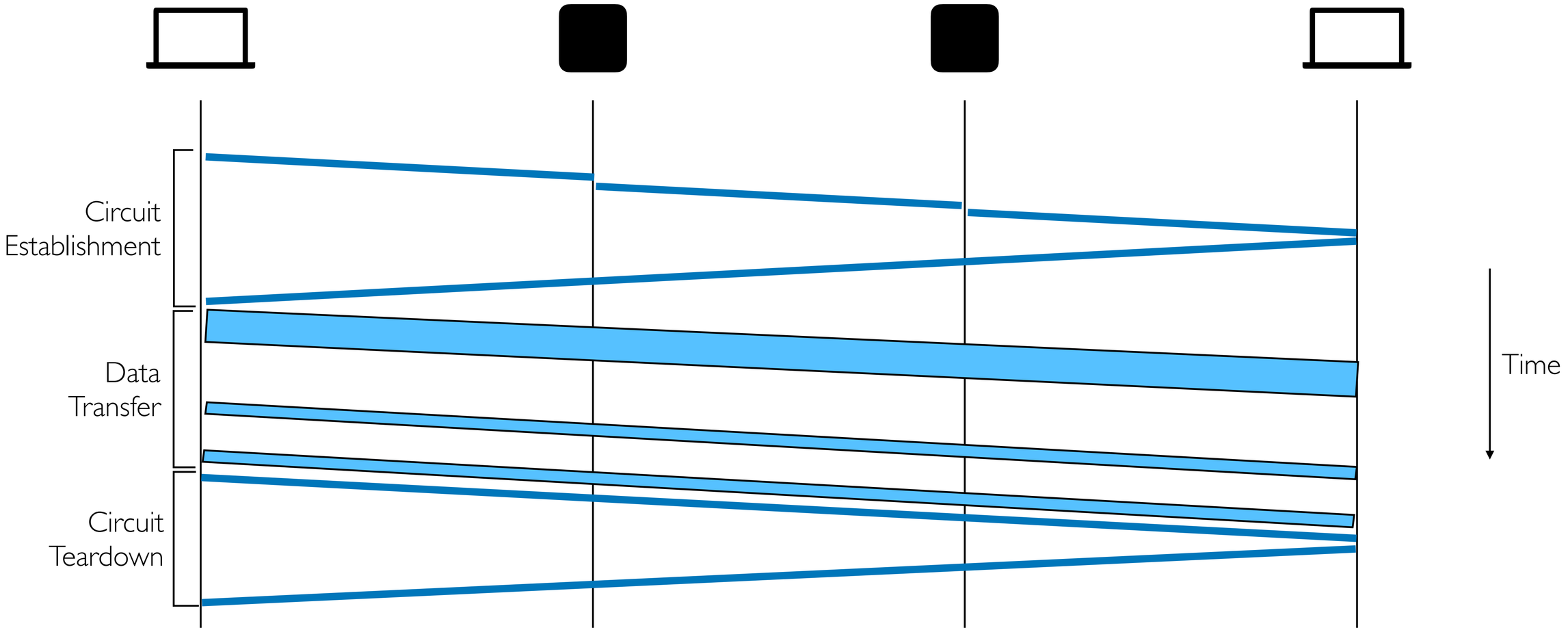
Timing in Circuit Switching



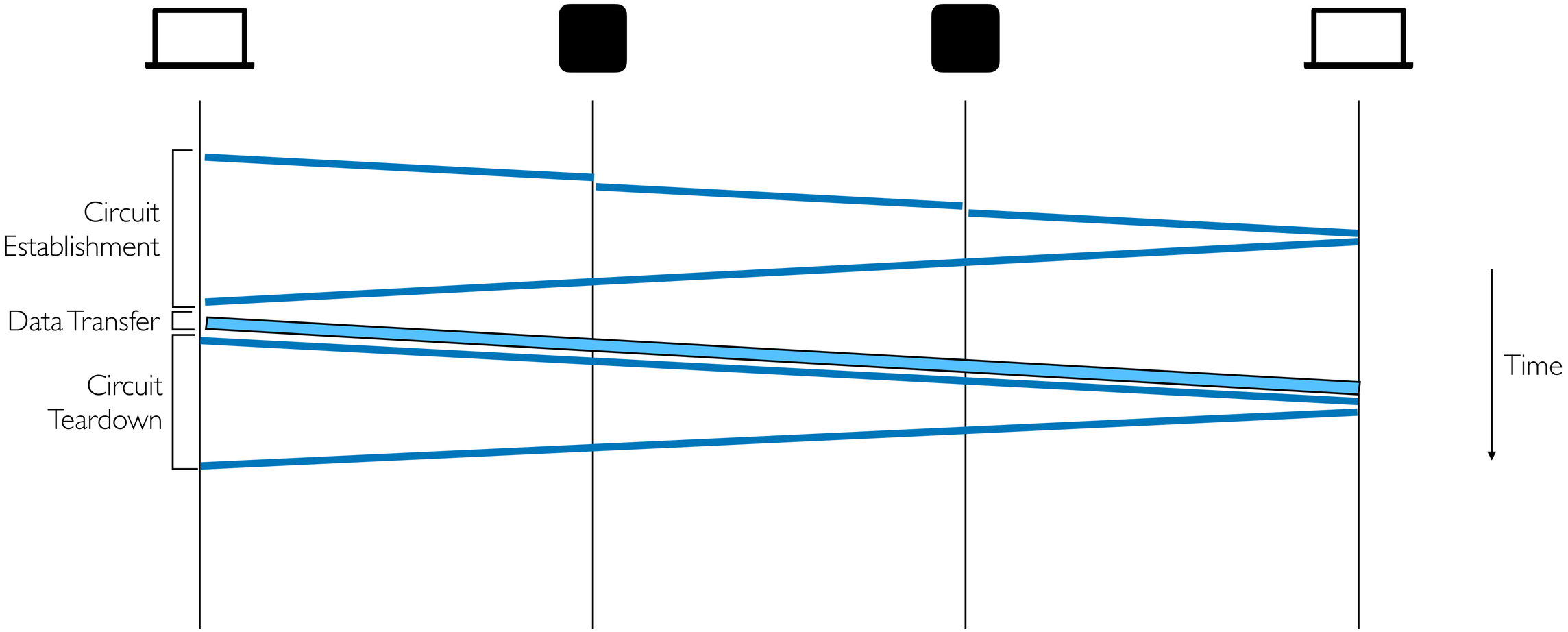
Timing in Circuit Switching



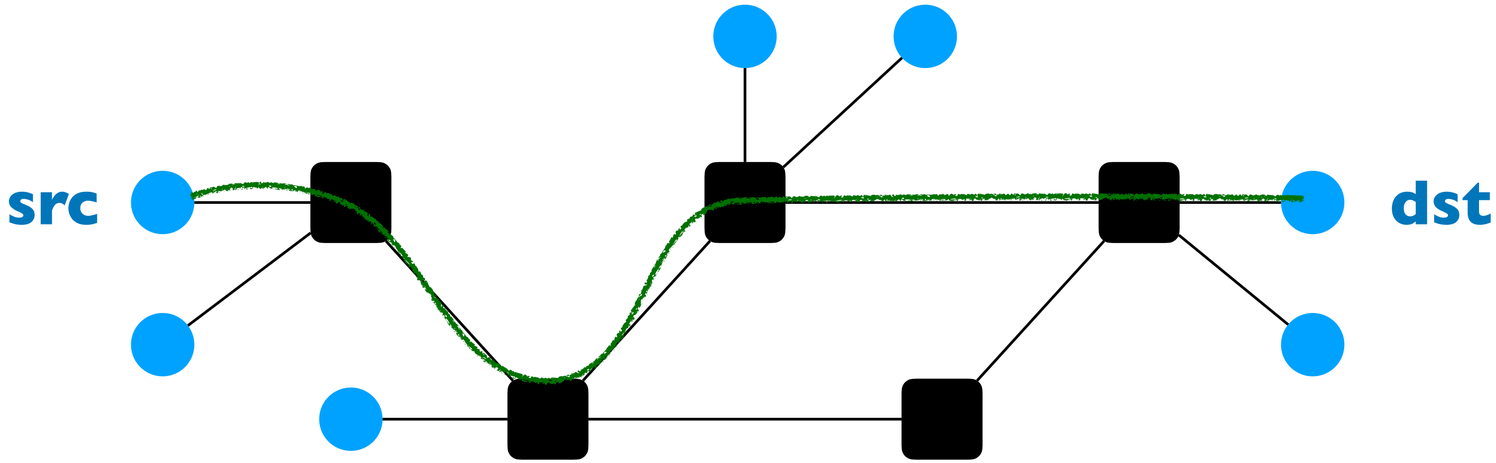
What if there are sporadic periods of data transfer?



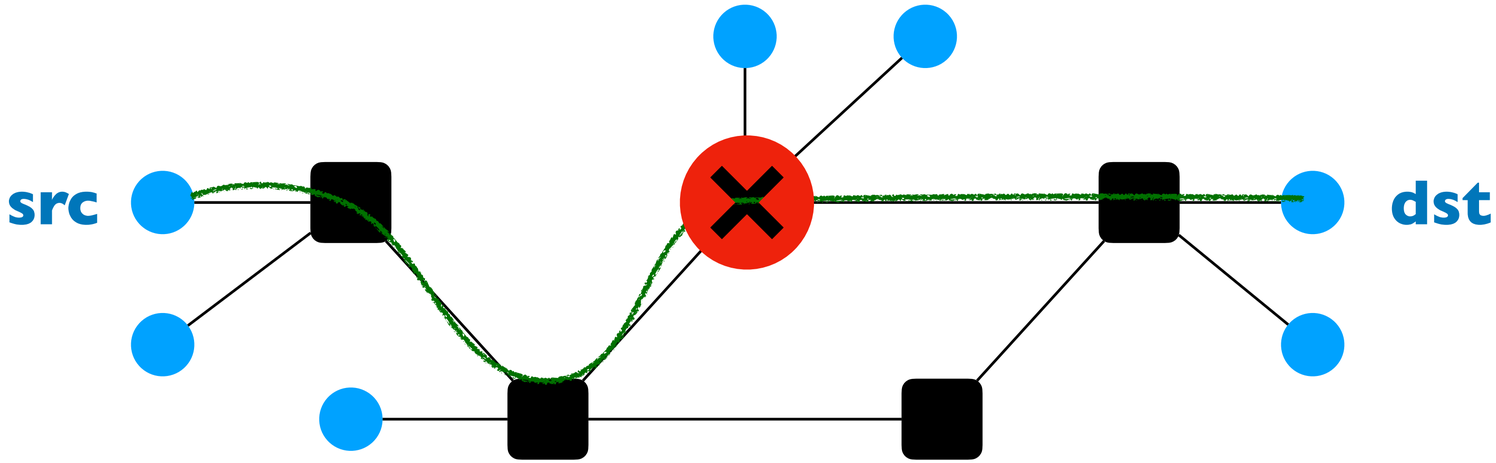
What if there is very little data to transfer?



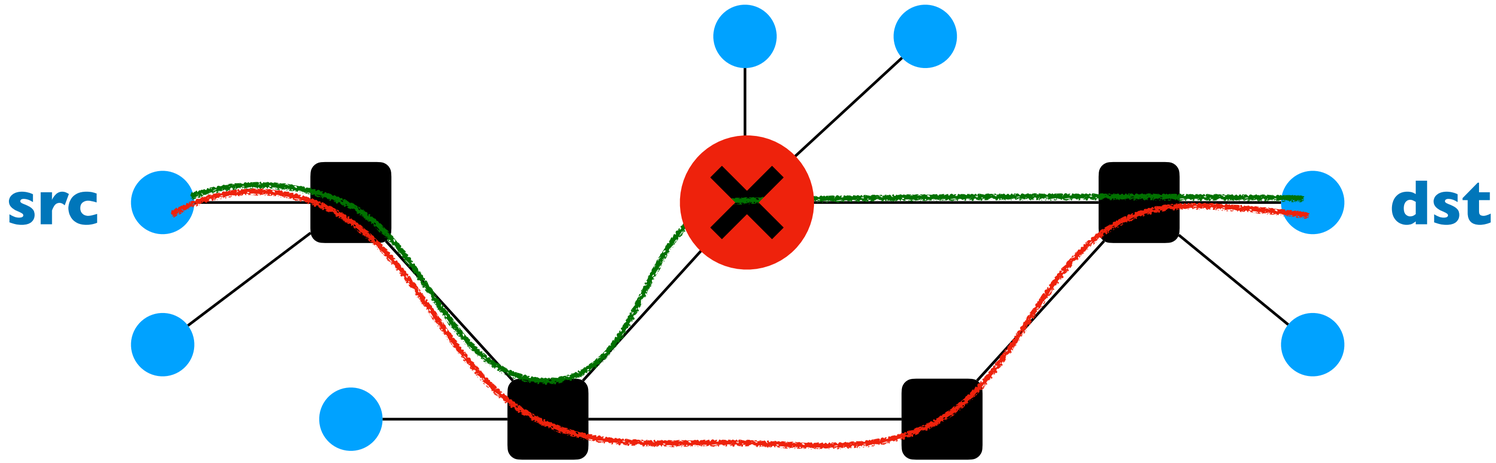
What if switches fail?



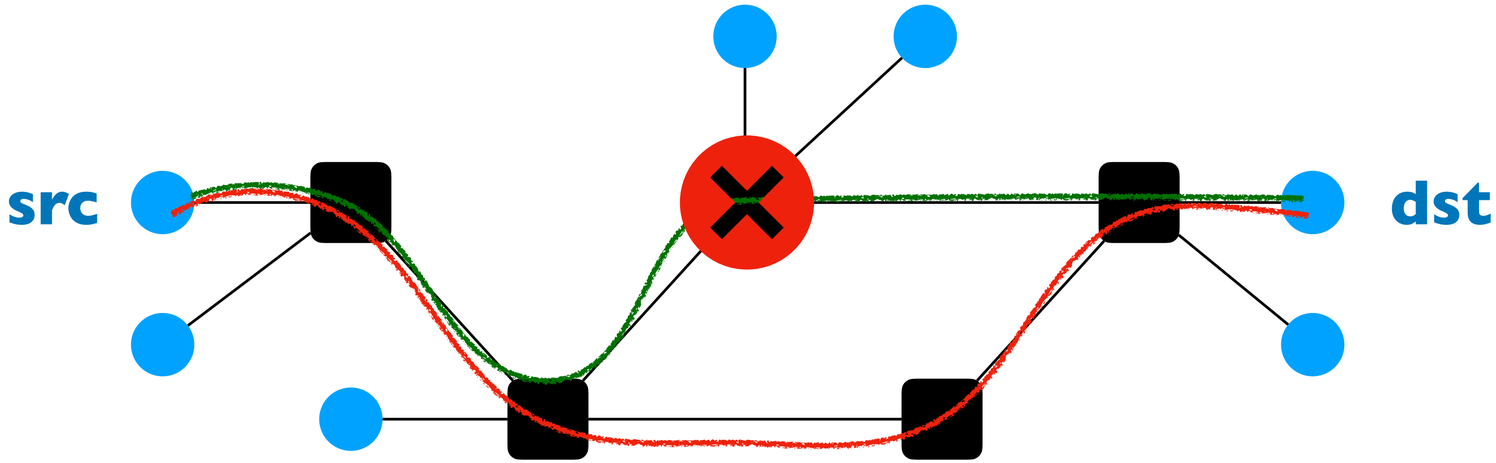
What if switches fail?



What if switches fail?



What if switches fail?



Circuit switching doesn't ***route around trouble***

Circuit Switching

Circuit Switching

- **What's good?**
  - *Predictable performance*
  - *Simple/fast switching (once circuit is established)*

Circuit Switching

- **What's good?**
  - *Predictable performance*
  - *Simple/fast switching (once circuit is established)*
- **What's not-so-good?**
  - *Complexity of circuit setup/teardown*
  - *Inefficient when traffic is bursty*
  - *Circuit setup adds delay*
  - *Switch fails -> its circuits fail*

Questions?

How else can we share?

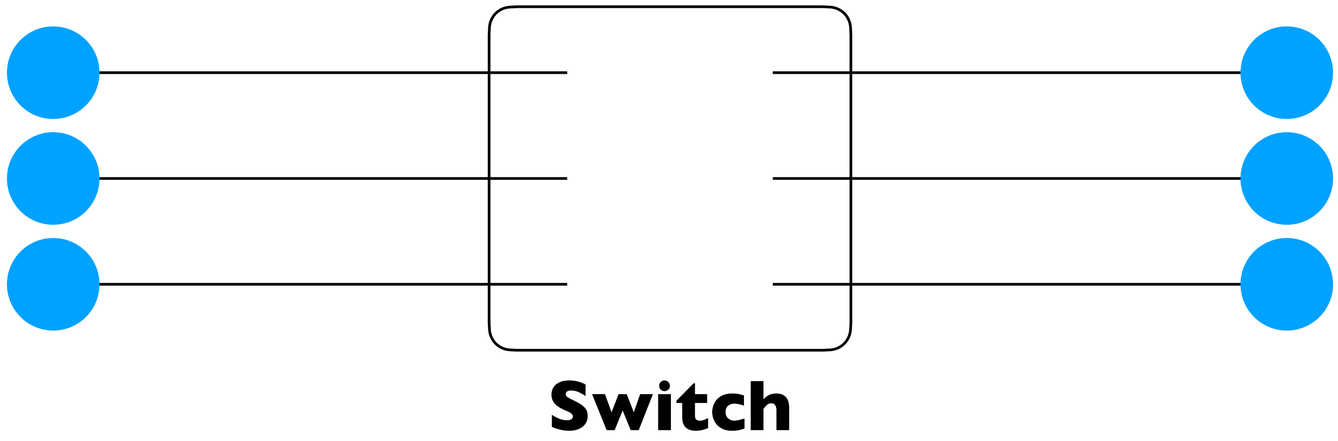
How else can we share?

- **Second approach:** *On demand* (aka "best-effort")

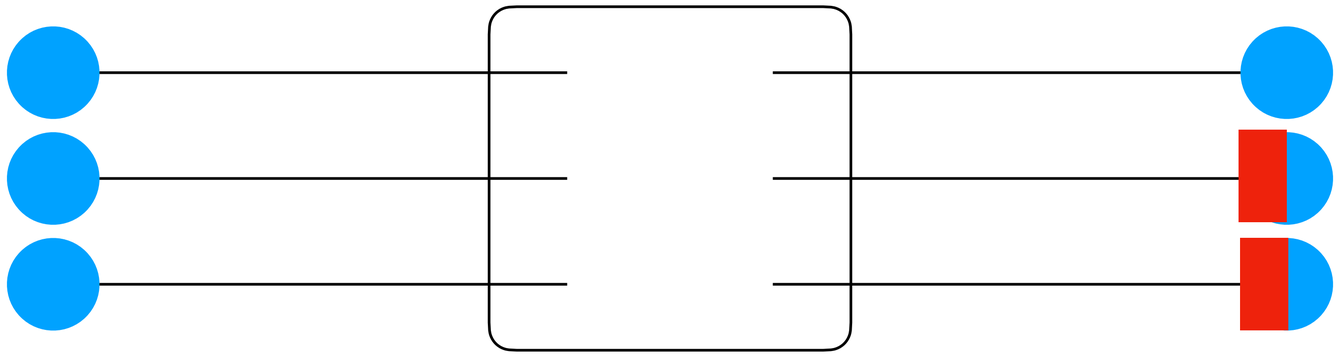
How else can we share?

- **Second approach:** *On demand* (aka "best-effort")
- **Mechanism:**
  - Break down data into packets
  - Send packets when you have them
  - Hope for the best…

Best-effort: Packet Switching



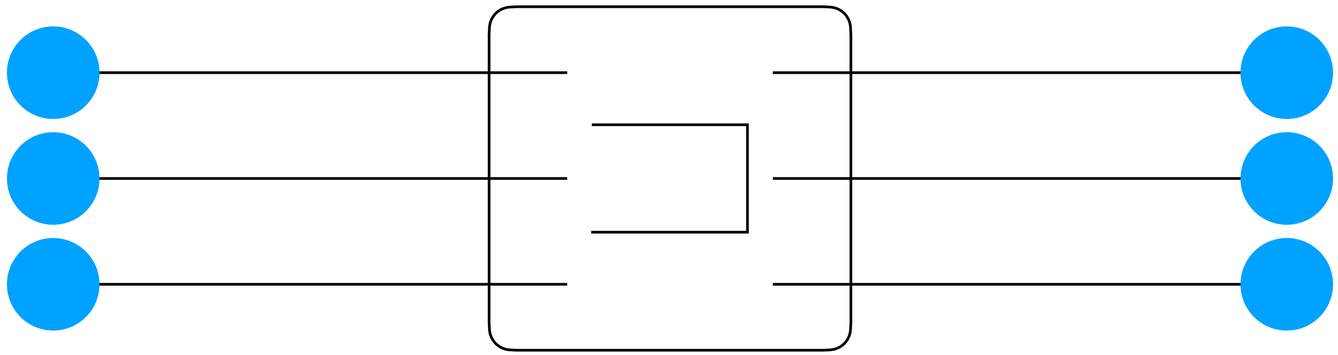
Best-effort: Packet Switching



Switch

Each packet contains destination
Each packet treated independently

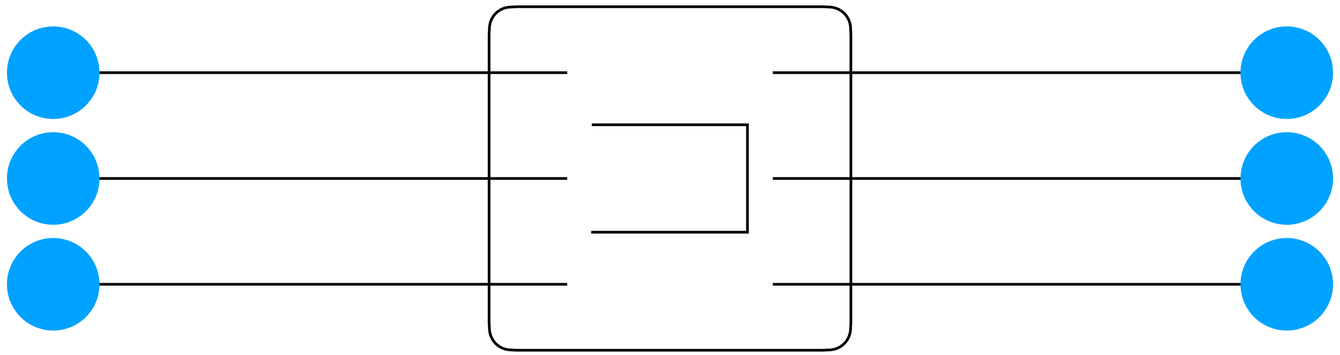
Best-effort: Packet Switching



Switch

Each packet contains destination
Each packet treated independently

Best-effort: Packet Switching

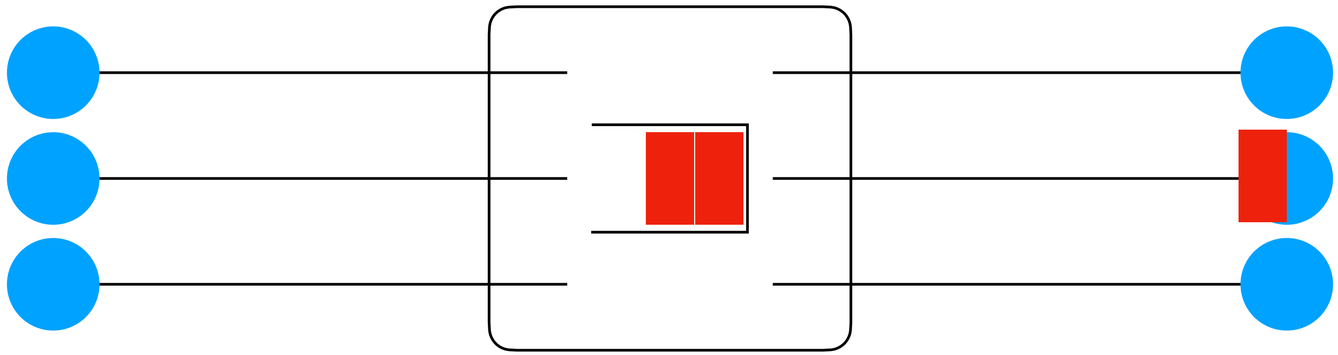


Switch

Each packet contains destination
Each packet treated independently

With buffers to absorb transient overloads

Best-effort: Packet Switching



Switch

Each packet contains destination
Each packet treated independently

With buffers to absorb transient overloads

Packet Switching

Packet Switching

- **What's good?**
  - *Efficient use of network resources*
  - *Simpler to implement*
  - *Robust: can "route around trouble"*

Packet Switching

- **What's good?**
  - *Efficient use of network resources*
  - *Simpler to implement*
  - *Robust: can "route around trouble"*
- **What's not-so-good?**
  - *Unpredictable performance*
  - *Requires buffer management and congestion control*

Circuits vs Packets

Circuits vs Packets

- **When are circuits better?**
  - *When you need dedicated performance (reserved bandwidth)*
  - *When you need predictable performance (without failures)*

Circuits vs Packets

- **When are circuits better?**
  - *When you need dedicated performance (reserved bandwidth)*
  - *When you need predictable performance (without failures)*
- **When are packets better?**
  - *When you care about overall efficiency of the network*
  - *If you want simpler switches*
  - *If you want easier recovery from failures*
  - *If you want to start sending data ASAP*

Circuits vs Packets

- **When are circuits better?**
  - *When you need dedicated performance (reserved bandwidth)*
  - *When you need predictable performance (without failures)*
- **When are packets better?**
  - *When you care about overall efficiency of the network*
  - *If you want simpler switches*
  - *If you want easier recovery from failures*
  - *If you want to start sending data ASAP*
- Internet chose to go for packets: ***why?***

The principle behind sharing: **Statistical Multiplexing**

The principle behind sharing: **Statistical Multiplexing**

- **Statistical Multiplexing:** Combining **demands** to **share resources** efficiently

The principle behind sharing: **Statistical Multiplexing**

- **Statistical Multiplexing:** Combining **demands** to **share resources** efficiently
- **Long history in computer science:** Processes in OSes, Cloud Computing, …

The principle behind sharing: **Statistical Multiplexing**

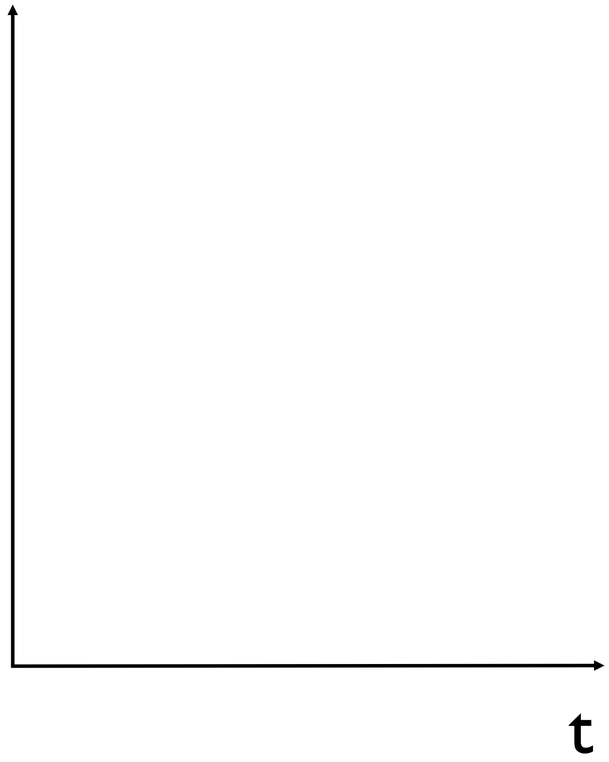
- **Statistical Multiplexing:** Combining **demands** to **share resources** efficiently
- **Long history in computer science:** Processes in OSes, Cloud Computing, …
- **Intuition:** peak of aggregate load is << aggregate of peak loads

Statistical Multiplexing: Intuition

- **Peak of aggregate load is << aggregate of peak loads**

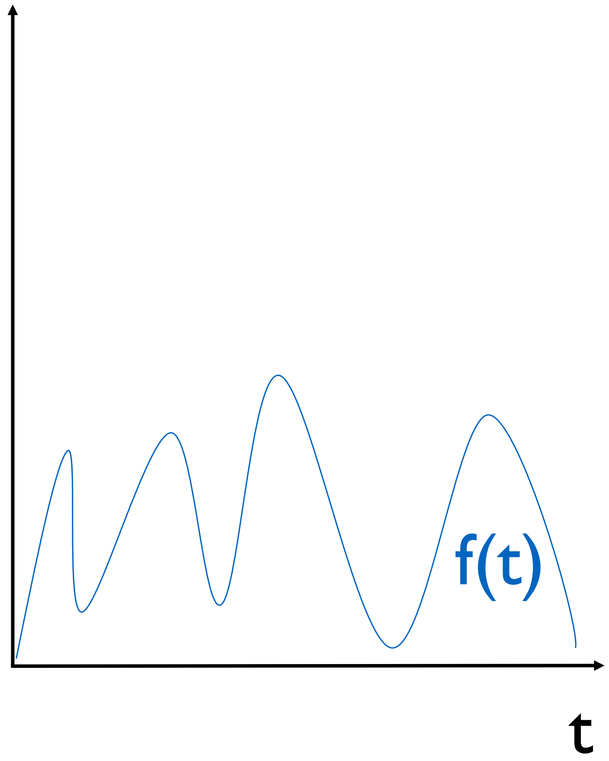
Statistical Multiplexing: Intuition

- **Peak of aggregate load is << aggregate of peak loads**



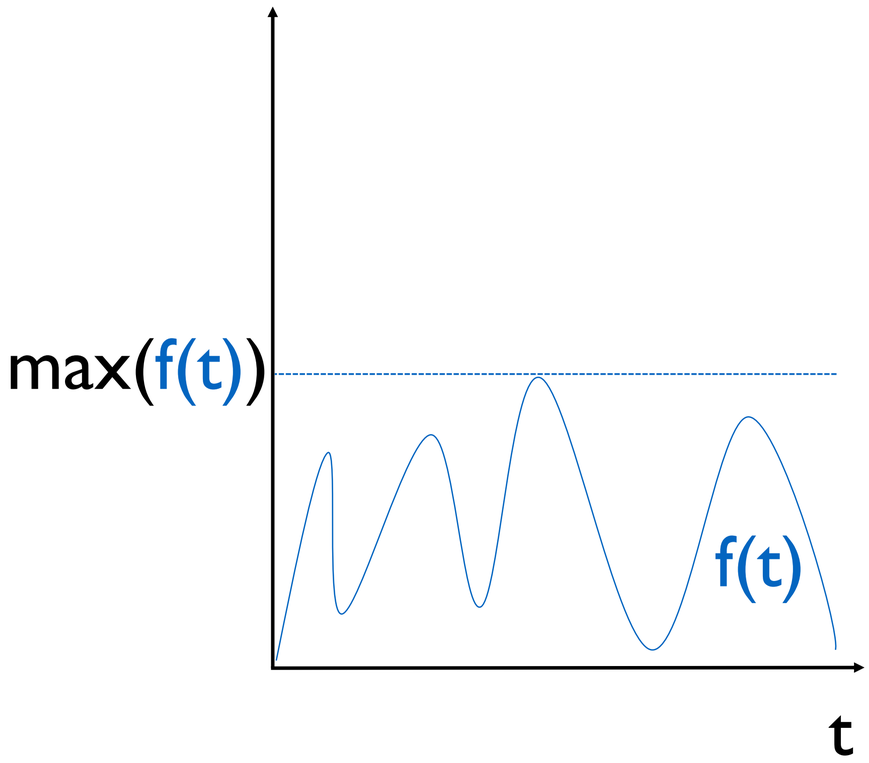
Statistical Multiplexing: Intuition

- **Peak of aggregate load is << aggregate of peak loads**



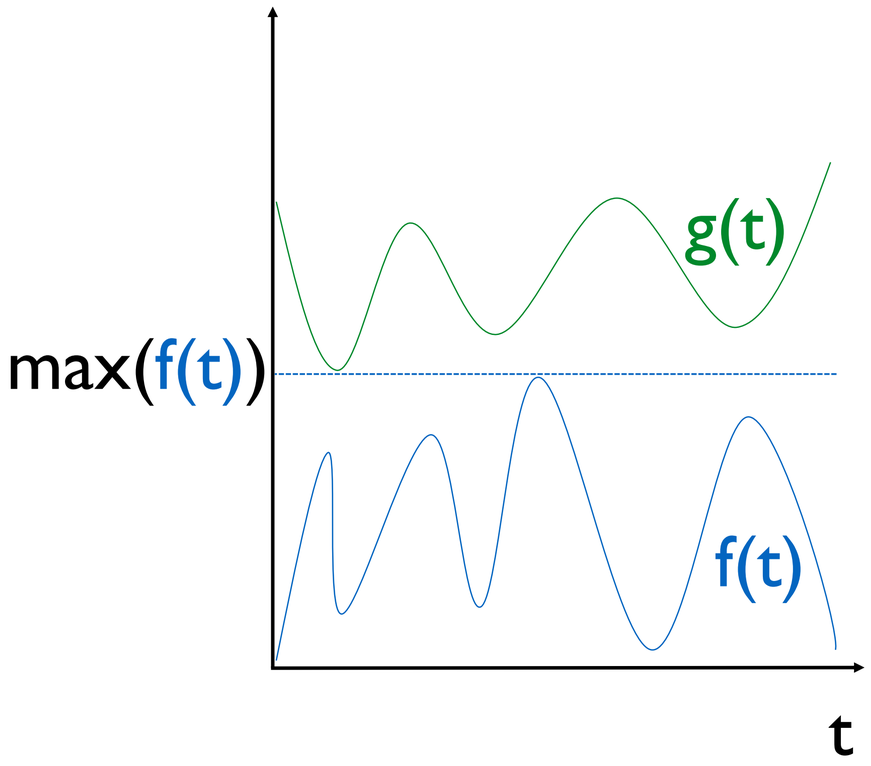
Statistical Multiplexing: Intuition

- **Peak of aggregate load is << aggregate of peak loads**



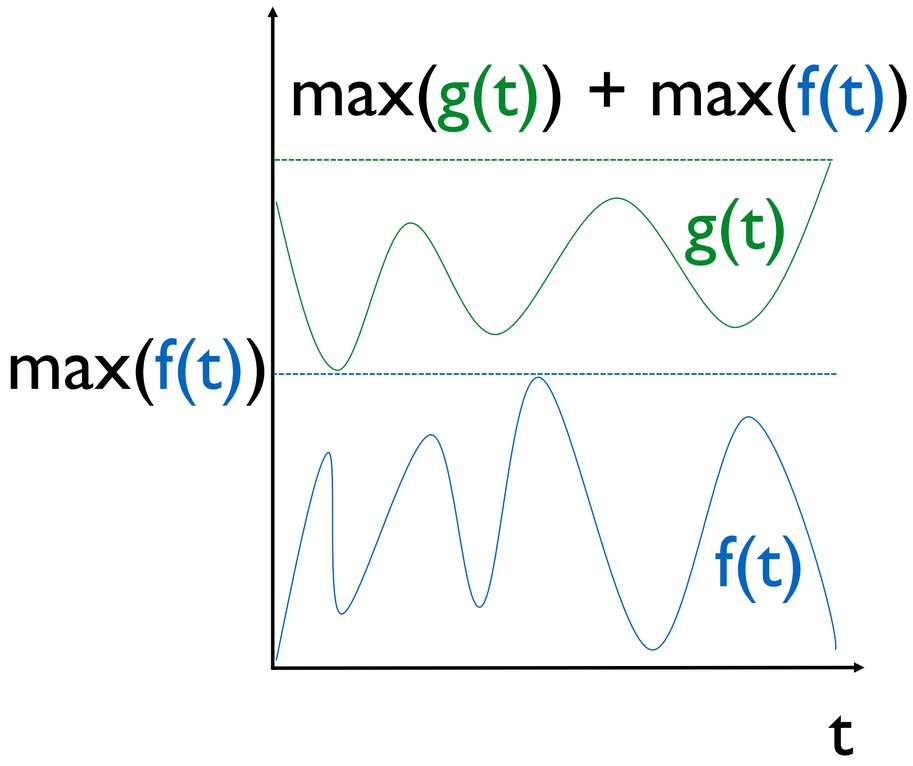
Statistical Multiplexing: Intuition

- **Peak of aggregate load is << aggregate of peak loads**



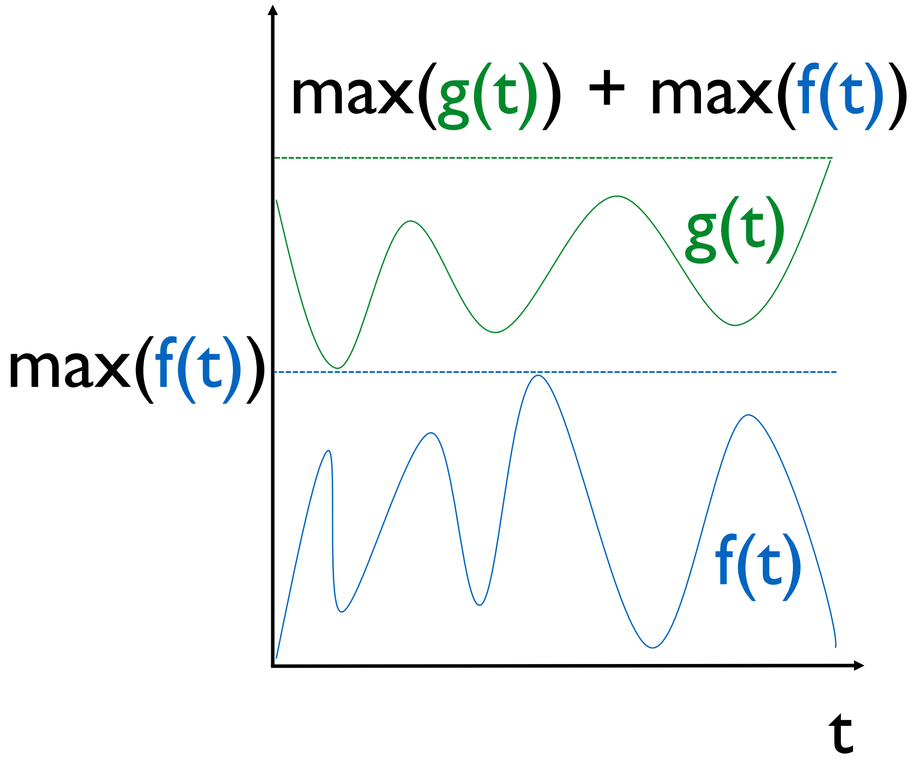
Statistical Multiplexing: Intuition

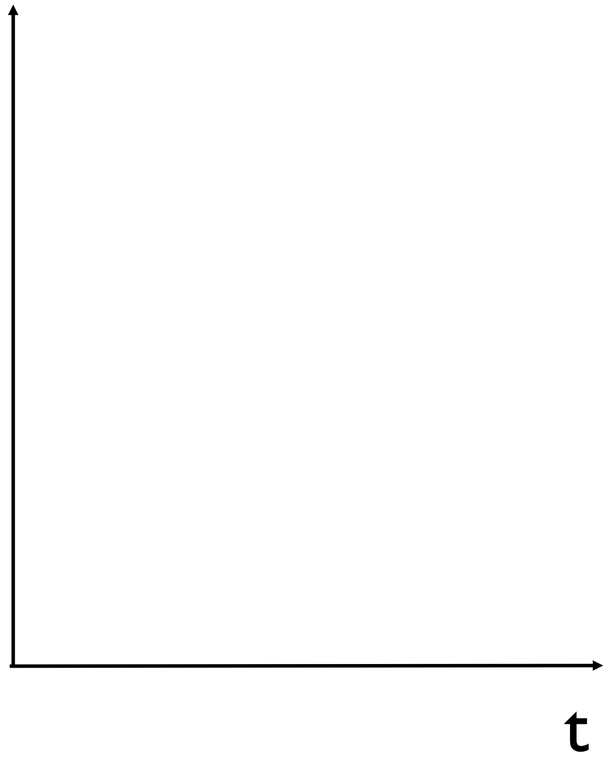
- **Peak of aggregate load is << aggregate of peak loads**



Statistical Multiplexing: Intuition

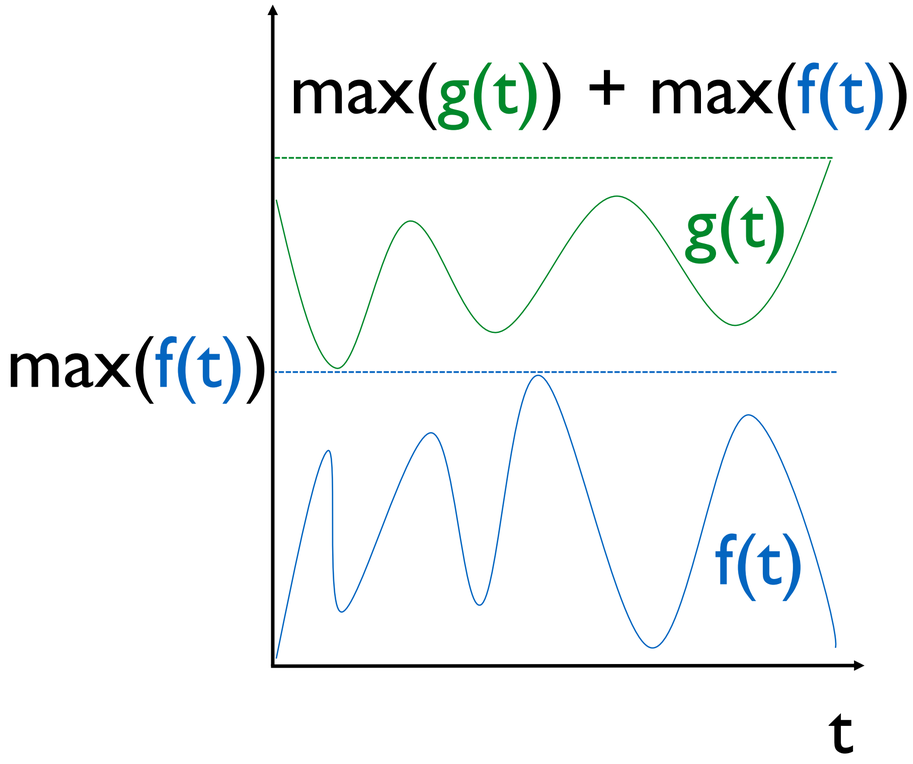
- **Peak of aggregate load is << aggregate of peak loads**

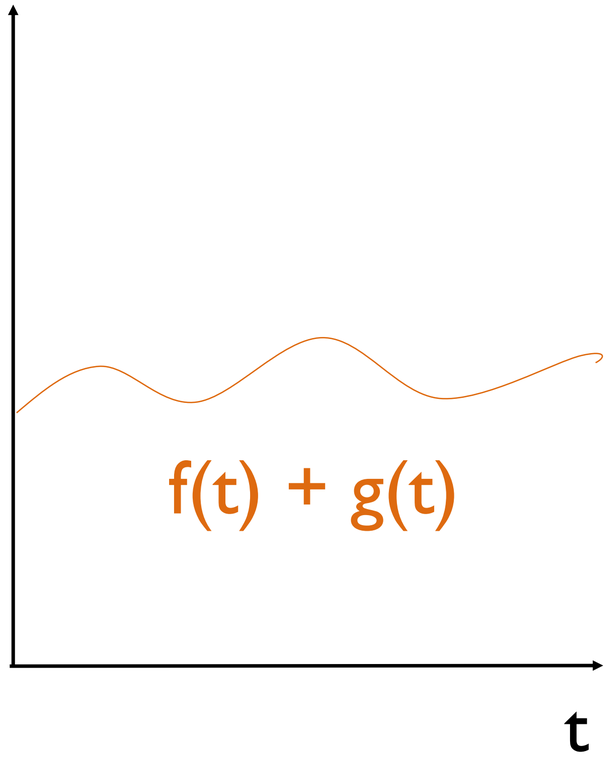




Statistical Multiplexing: Intuition

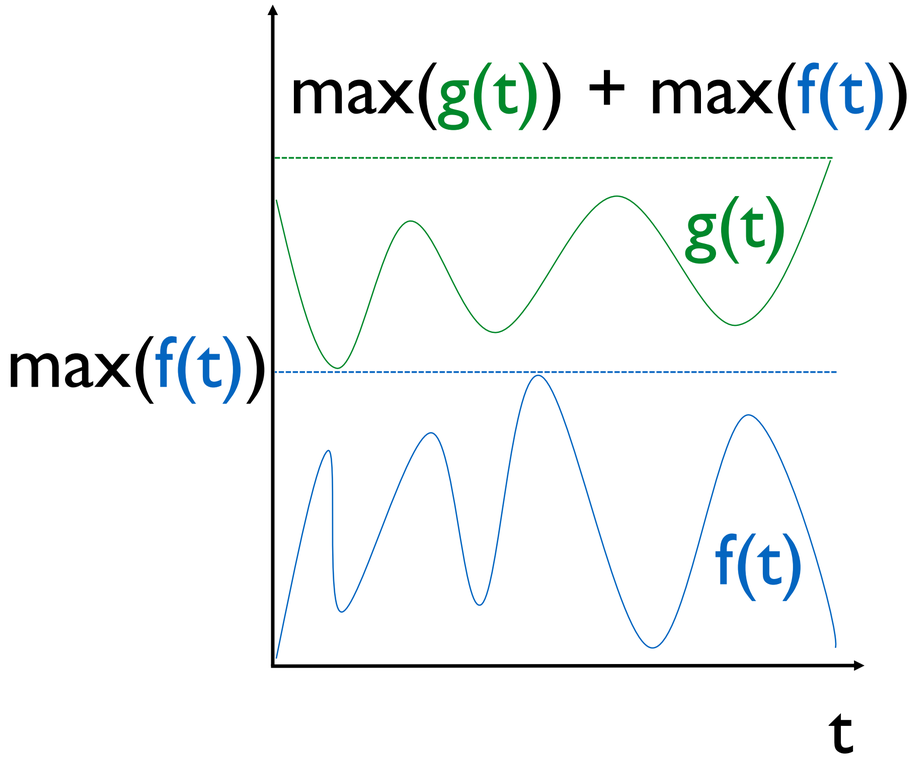
- **Peak of aggregate load is << aggregate of peak loads**

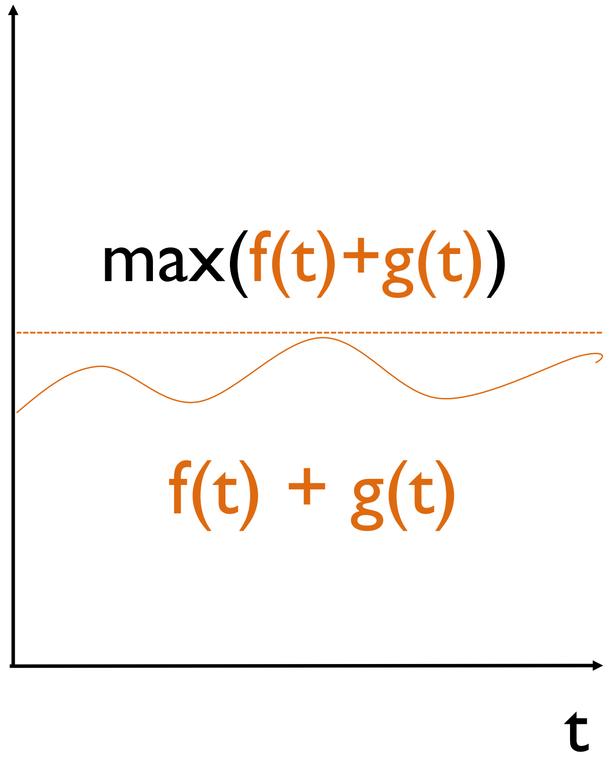




Statistical Multiplexing: Intuition

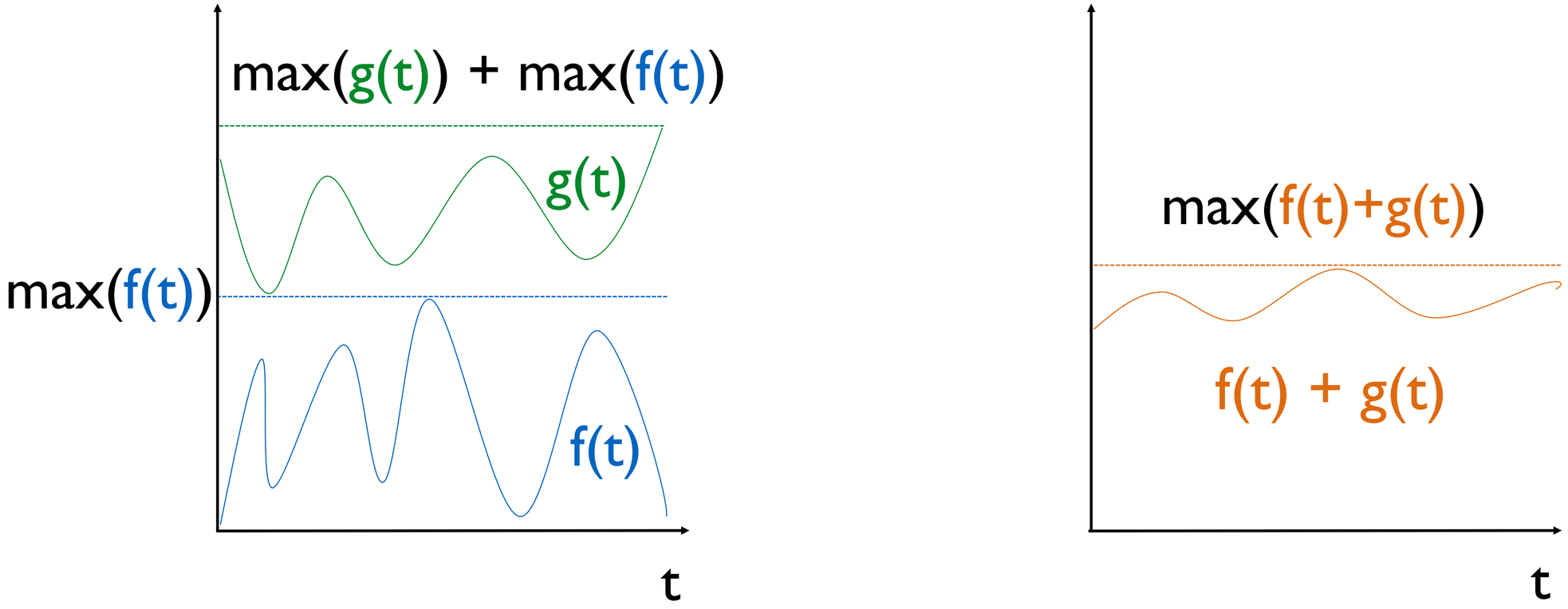
- **Peak of aggregate load is << aggregate of peak loads**





Statistical Multiplexing: Intuition

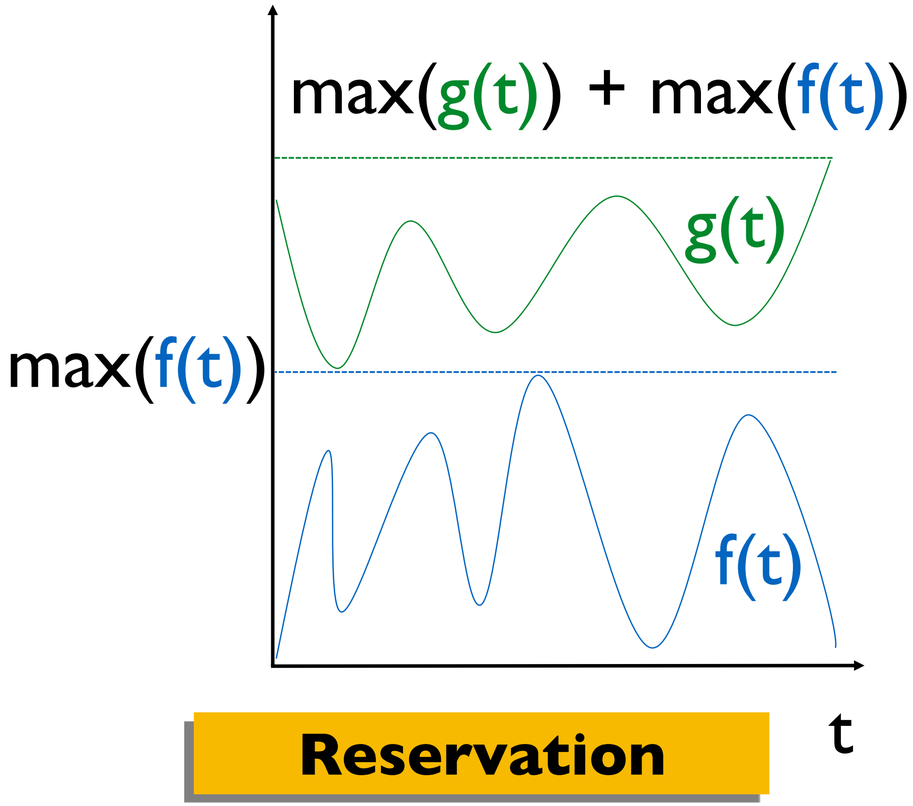
- **Peak of aggregate load is << aggregate of peak loads**



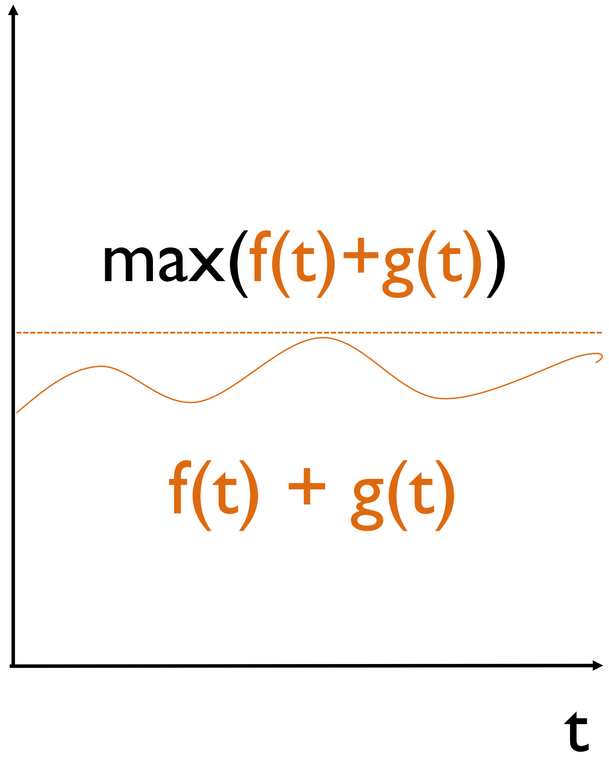
$\max_t(f(t)+g(t)) << \max_t(f(t)) + \max_t(g(t))$

Statistical Multiplexing: Intuition

- **Peak of aggregate load is << aggregate of peak loads**



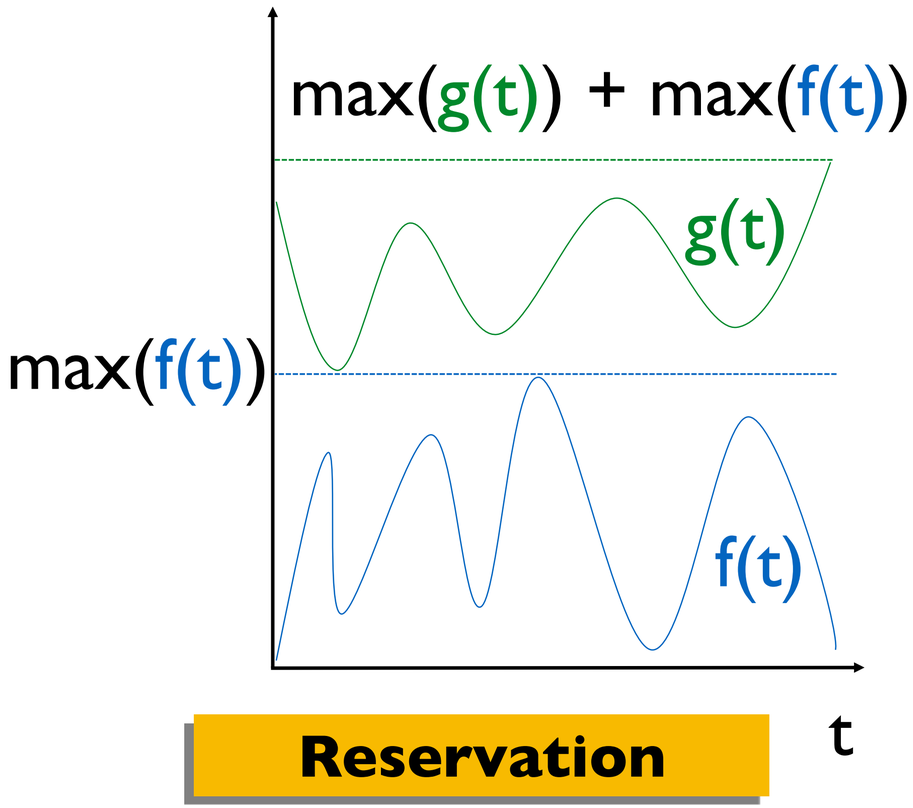
Reservation



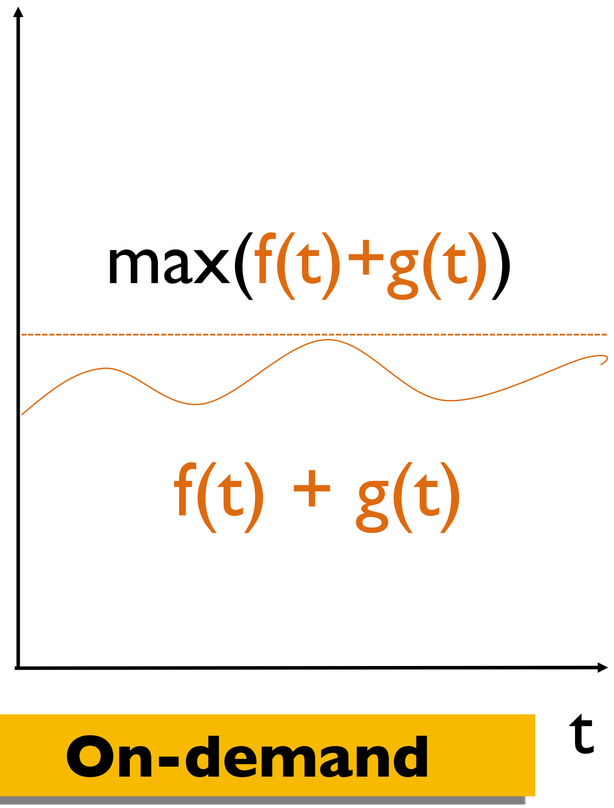
$$\max_t(f(t)+g(t)) << \max_t(f(t)) + \max_t(g(t))$$

Statistical Multiplexing: Intuition

- **Peak of aggregate load is << aggregate of peak loads**



Reservation



On-demand

$\max_t(f(t)+g(t)) << \max_t(f(t)) + \max_t(g(t))$

Questions?

Today's Agenda

- More details on What & How of the Internet
- Sharing a network
- Evaluating a network

Performance Metrics

Performance Metrics

- **Delay:** How long does it take to send a packet from its source to destination?

Performance Metrics

- **Delay:** How long does it take to send a packet from its source to destination?
- **Loss:** What fraction of the packets sent to the destination are dropped?

Performance Metrics

- **Delay:** How long does it take to send a packet from its source to destination?
- **Loss:** What fraction of the packets sent to the destination are dropped?
- **Throughput:** At what rate is the destination receiving data from the source?

Delay

Delay

- Consists of four components

Delay

- Consists of four components
  - **Transmission Delay**

Delay

- Consists of four components
  - Transmission Delay
  - Propagation Delay

Delay

- Consists of four components
  - Transmission Delay
  - Propagation Delay
  - Queueing Delay

Delay

- Consists of four components
  - Transmission Delay
  - Propagation Delay
  - Queueing Delay
  - Processing Delay

Delay

- Consists of four components
  - Transmission Delay
  - Propagation Delay
  - Queueing Delay
  - Processing Delay

Due to link properties

Due to traffic matrix and switch internals

A Network Link

A Network Link

- **Link bandwidth**
  - Number of bits sent/received per unit time

A Network Link

- **Link bandwidth**
  - Number of bits sent/received per unit time
- **Propagation delay**
  - Time for one bit to move through the link (seconds)

A Network Link

- **Link bandwidth**
  - Number of bits sent/received per unit time
- **Propagation delay**
  - Time for one bit to move through the link (seconds)
- **Bandwidth-Delay Product (BDP)**
  - Number of bits "in flight" at any time
  - BDP = Bandwidth x Propagation delay

Examples

Examples

- **Same city over a slow link:**
  - **Bandwidth:** ~100Mbps
  - **Propagation delay:** ~0.1ms
  - **BDP:** 10,000 bits (1.25KB)

Examples

- **Same city over a slow link:**
  - **Bandwidth:** ~100Mbps
  - **Propagation delay:** ~0.1ms
  - **BDP:** 10,000 bits (1.25KB)
- **Cross country over a fast link:**
  - **Bandwidth:** ~10Gbps
  - **Propagation delay:** ~10ms
  - **BDP:** 10^8 bits (12.5MB)

Transmission Delay

Transmission Delay

- How long does it take to put all the bits of a packet into a link?

Transmission Delay

- How long does it take to put all the bits of a packet into a link?
- Packet size / Transmission rate of link

Transmission Delay

- How long does it take to put all the bits of a packet into a link?
- Packet size / Transmission rate of link
- **Example:**
  - **Packet size:** 1000 bits
  - **Rate:** 100 Mbits/s
  - **Delay:** 1000 bits / (100 * 10^6 Mbits/s) = 10^{-5} s

Propagation Delay

Propagation Delay

- How long does it take to one bit from one end of the link to the other?

Propagation Delay

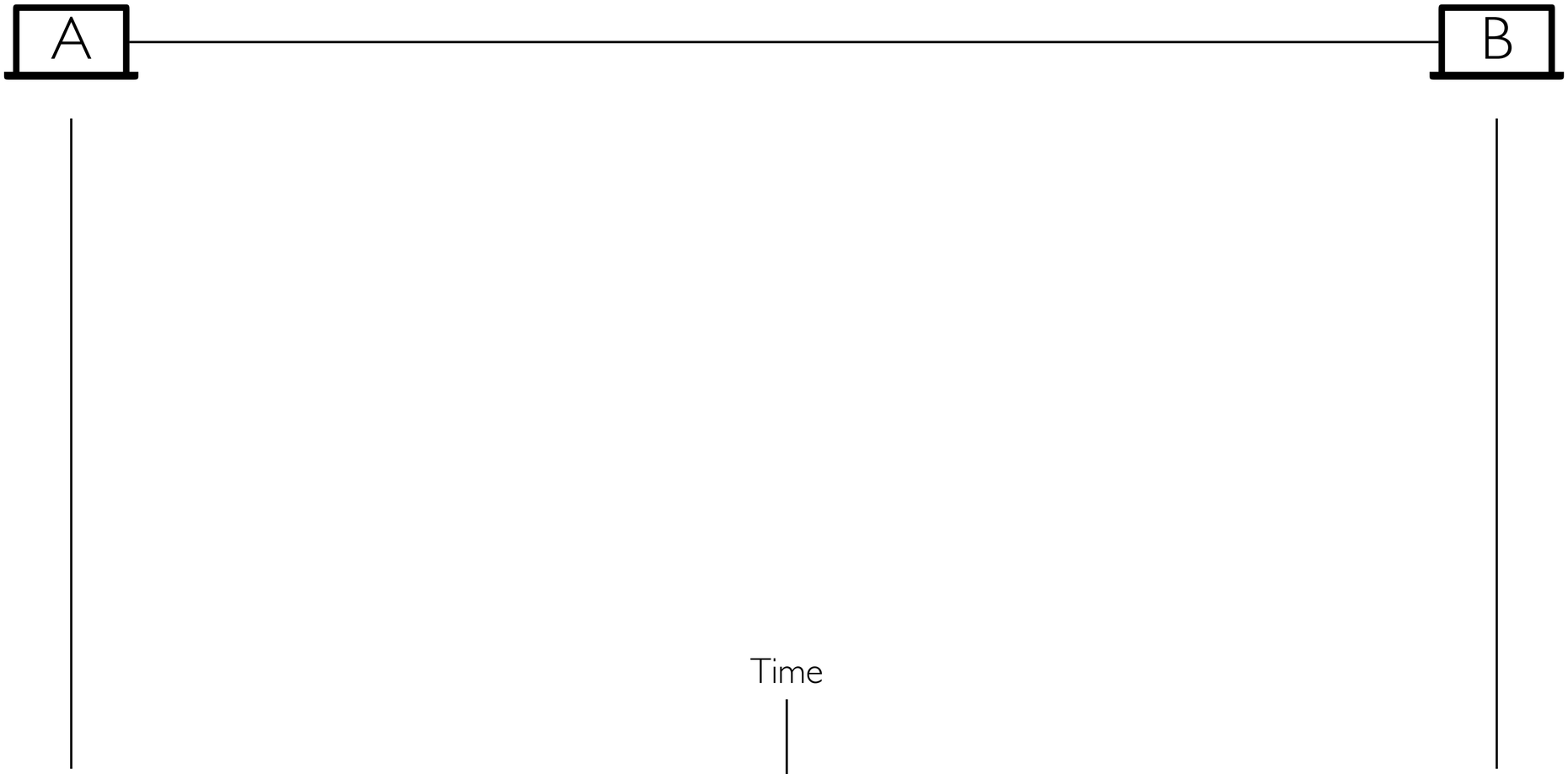
- How long does it take to one bit from one end of the link to the other?
- Link length / Propagation speed of link
  - Propagation speed ~ (close to) speed of light

Propagation Delay

- How long does it take to one bit from one end of the link to the other?
- Link length / Propagation speed of link
  - Propagation speed ~ (close to) speed of light
- **Example:**
  - **Length:** 30 km = 30,000 m
  - **Delay:** 30,000 m / ($3 * 10^8$ m/s) = 10^{-4} s

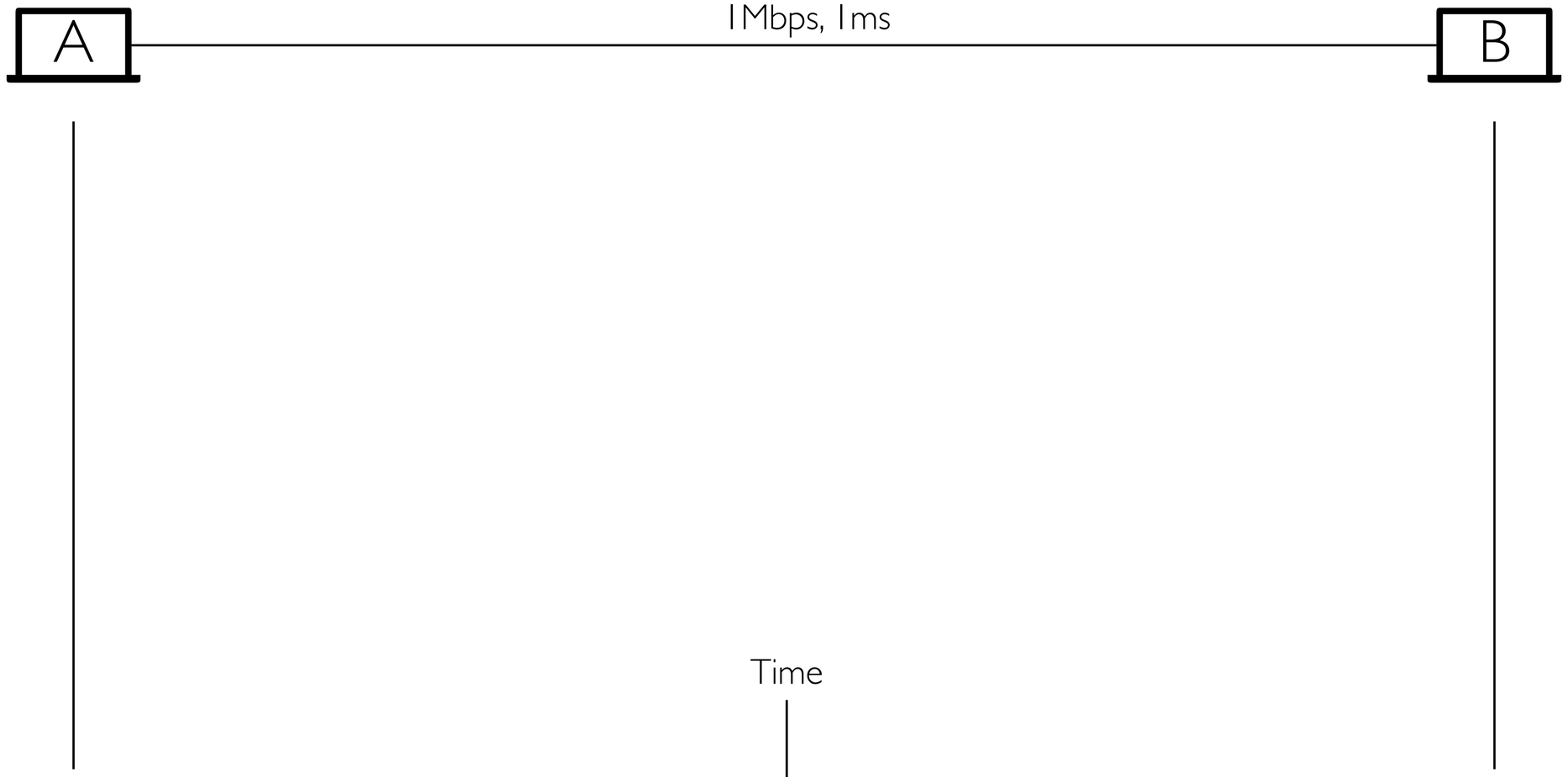
Packet Delay

Sending 100B packets from A to B?



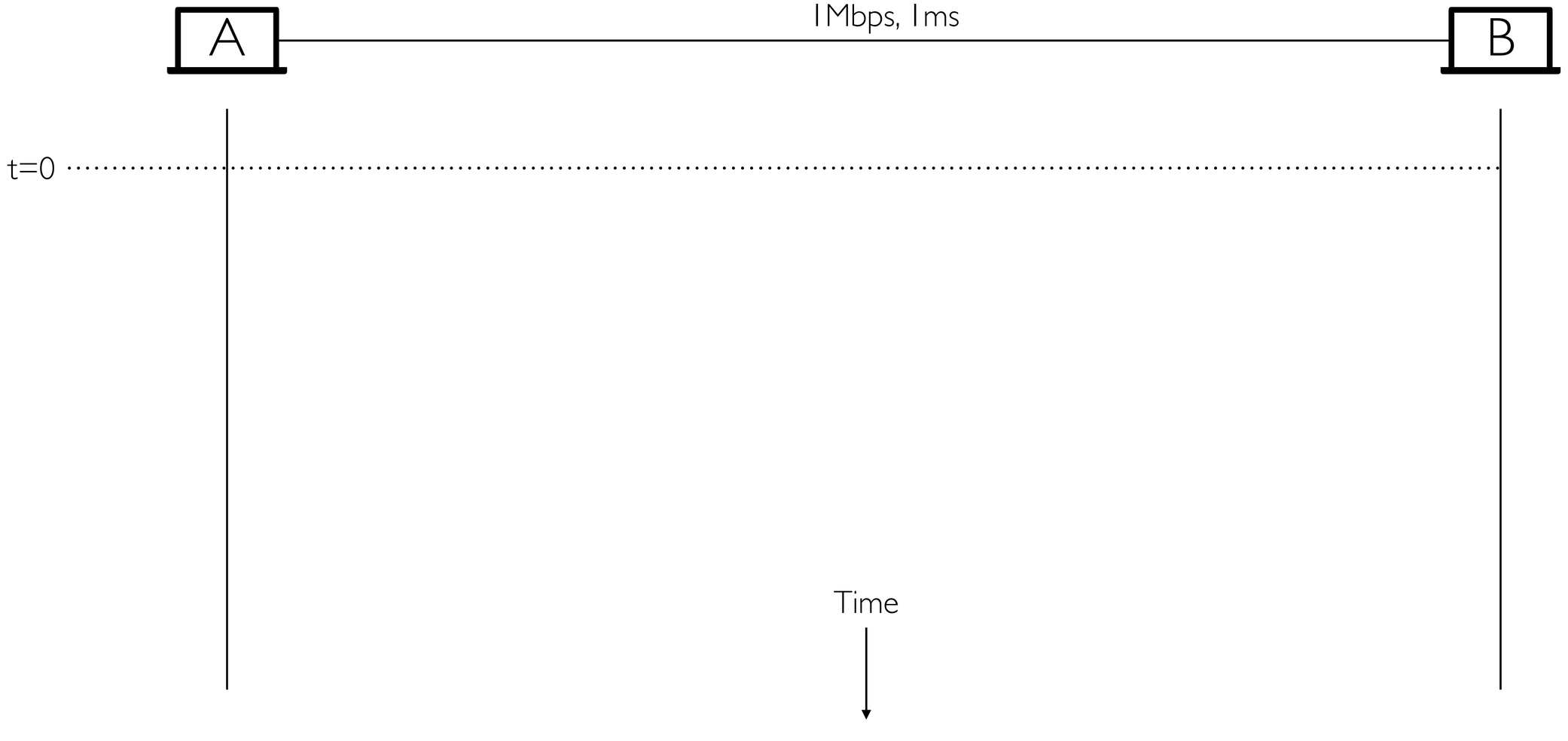
Packet Delay

Sending 100B packets from A to B?



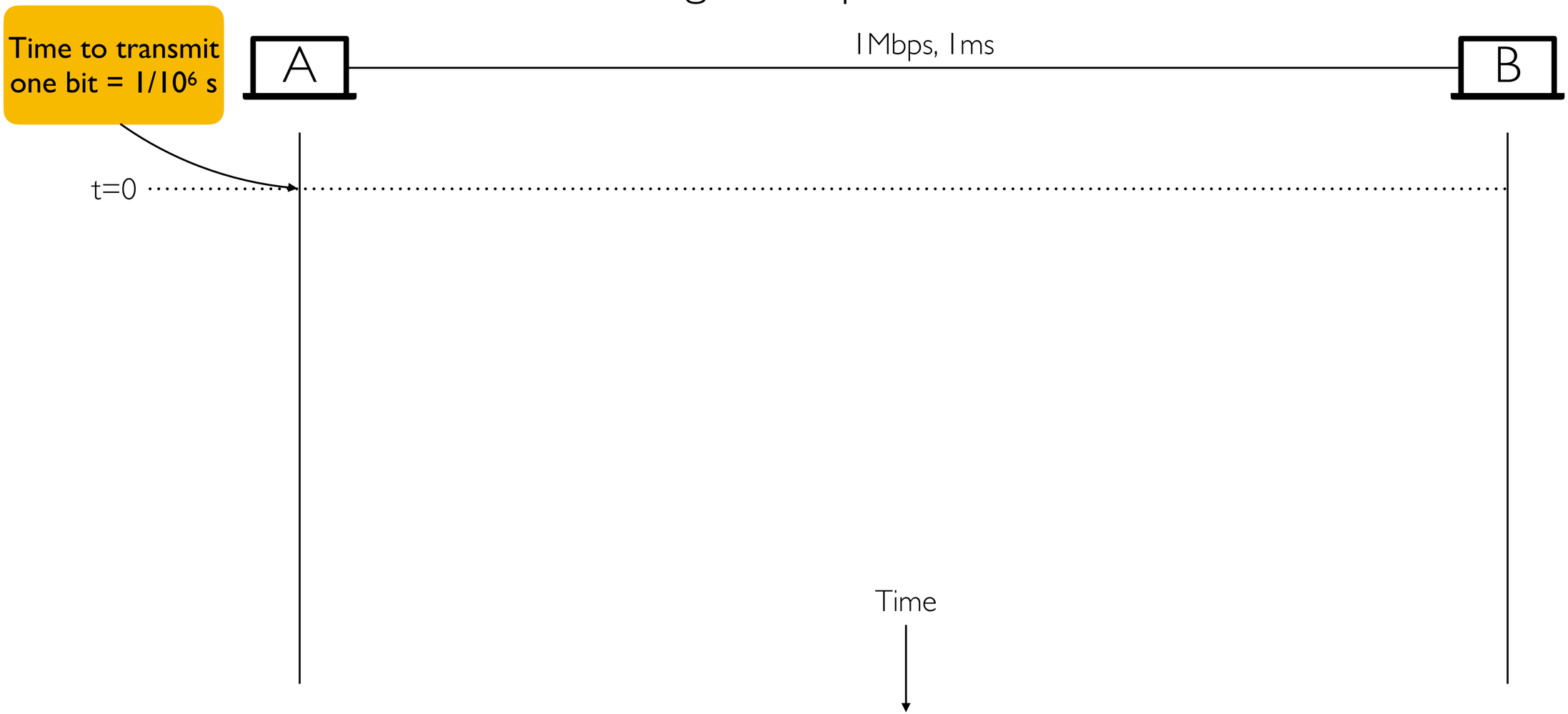
Packet Delay

Sending 100B packets from A to B?



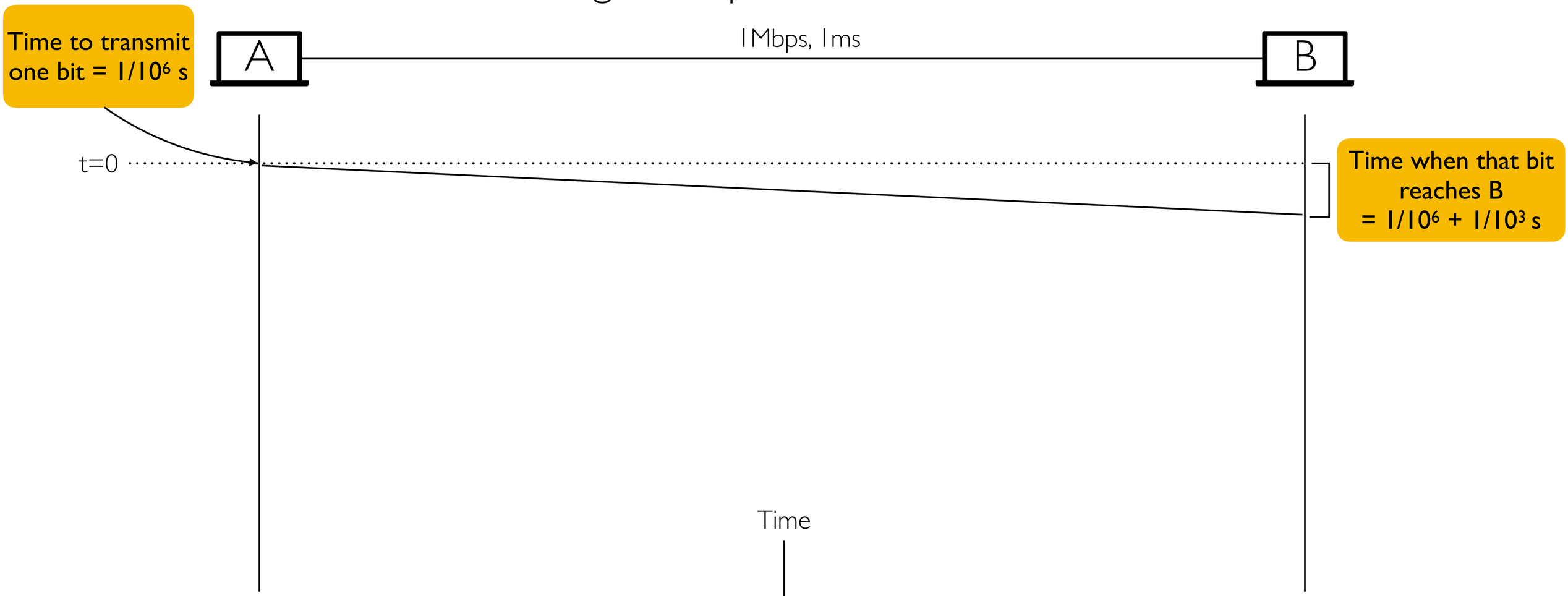
Packet Delay

Sending 100B packets from A to B?



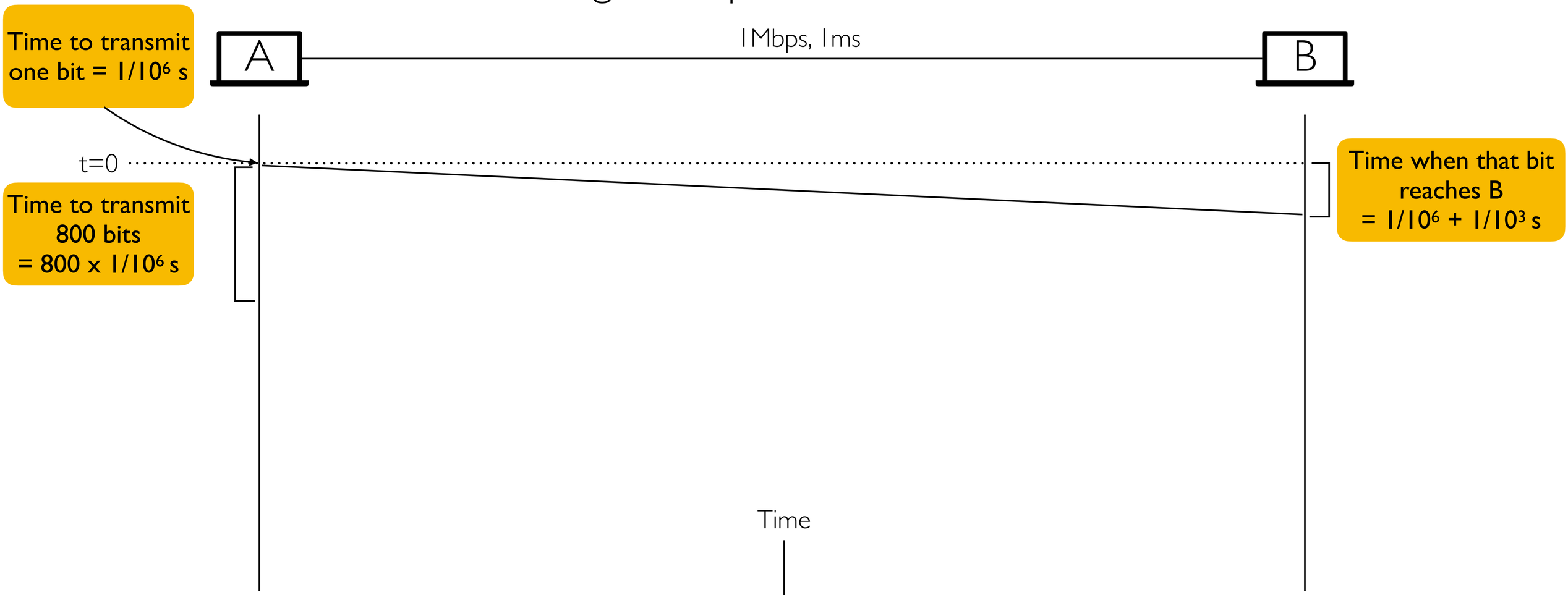
Packet Delay

Sending 100B packets from A to B?



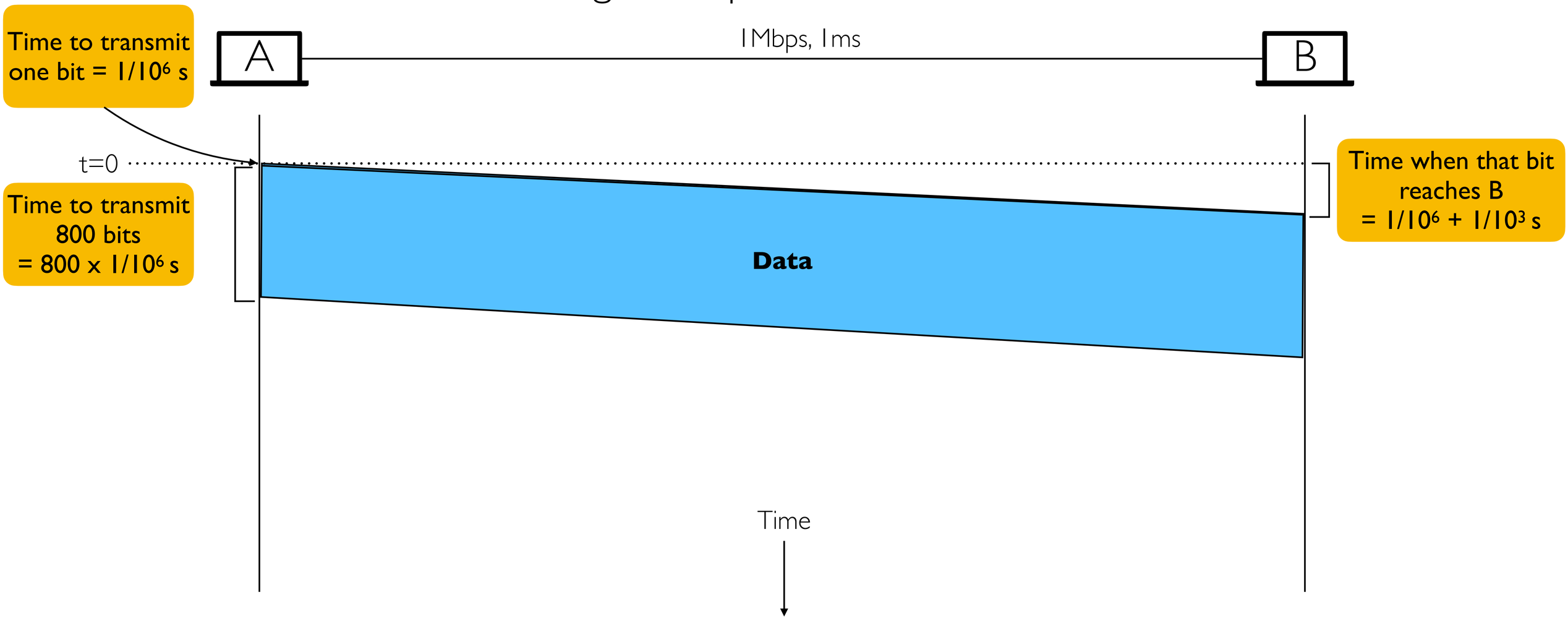
Packet Delay

Sending 100B packets from A to B?



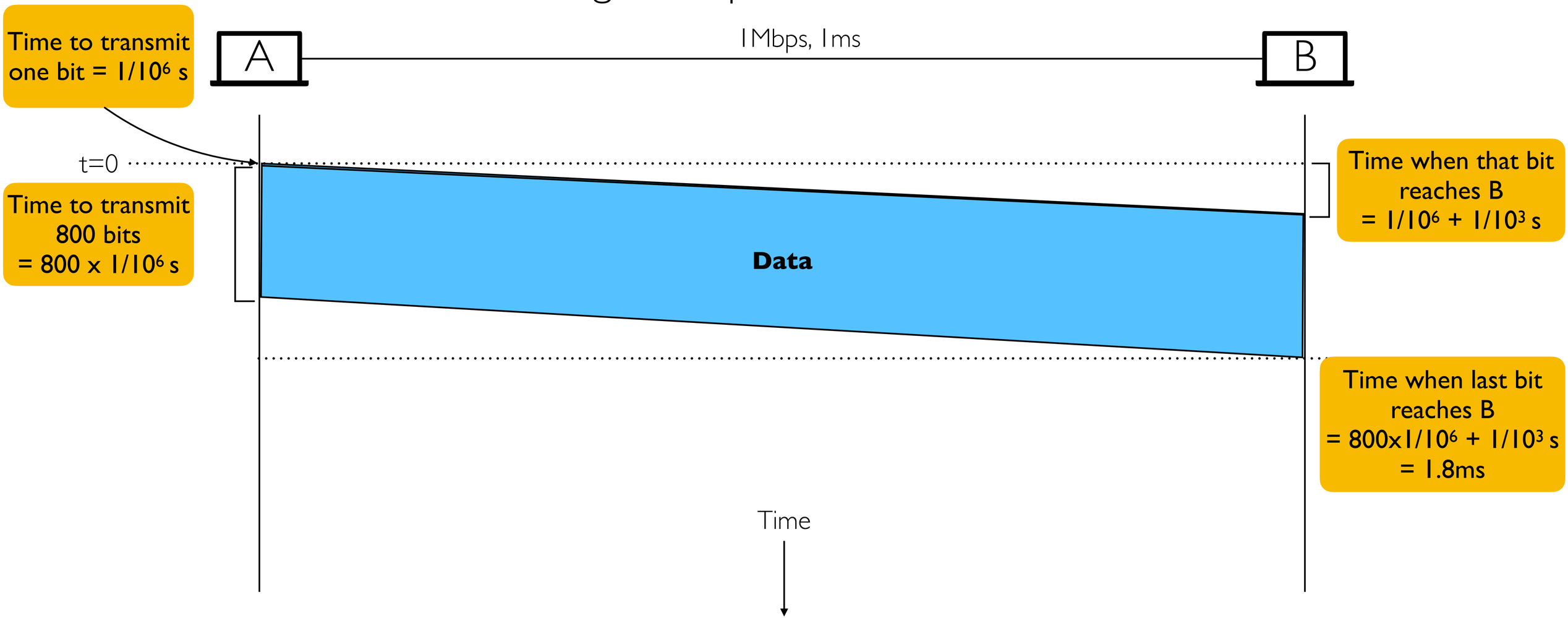
Packet Delay

Sending 100B packets from A to B?



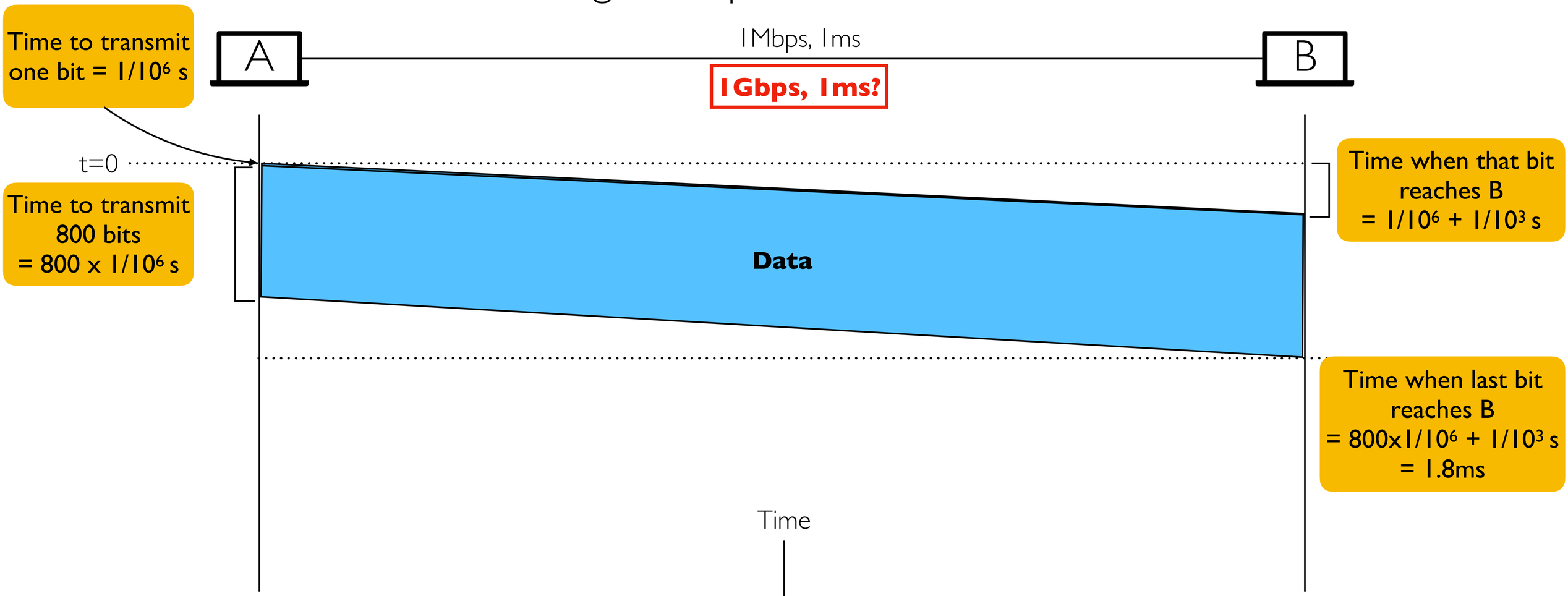
Packet Delay

Sending 100B packets from A to B?



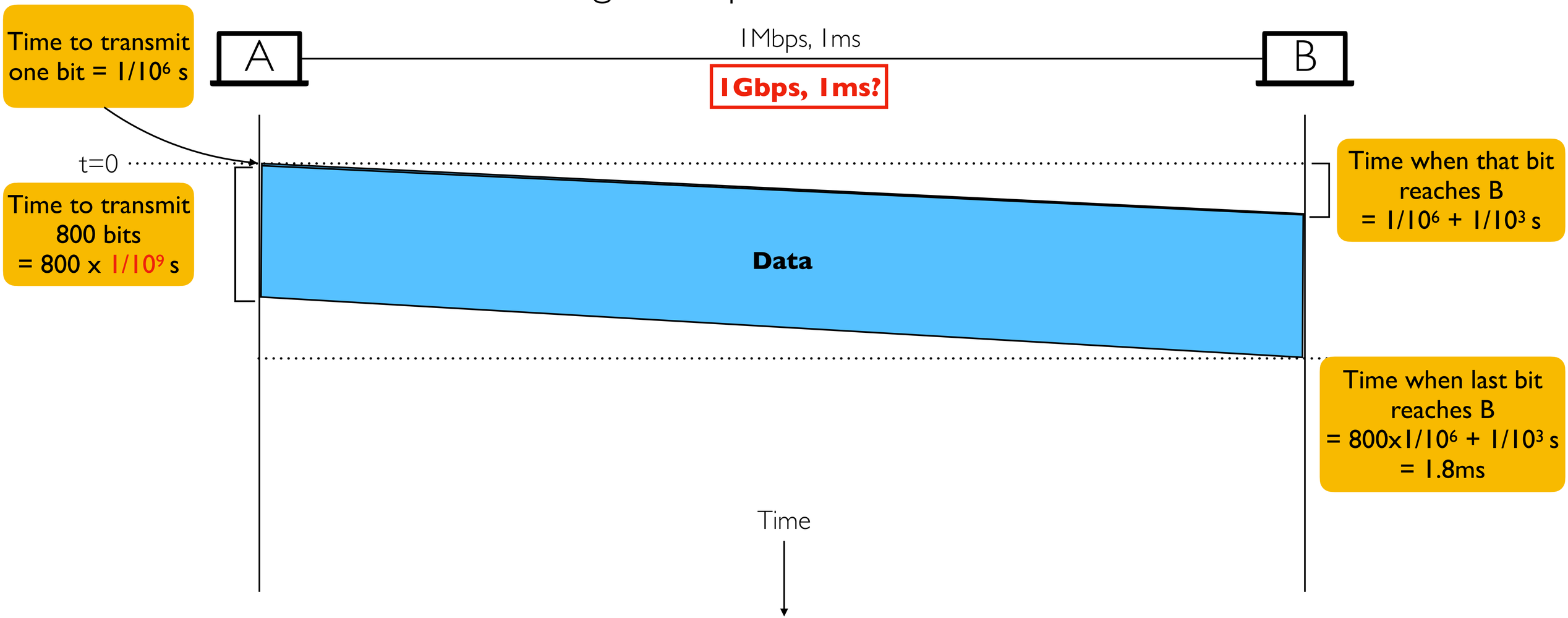
Packet Delay

Sending 100B packets from A to B?



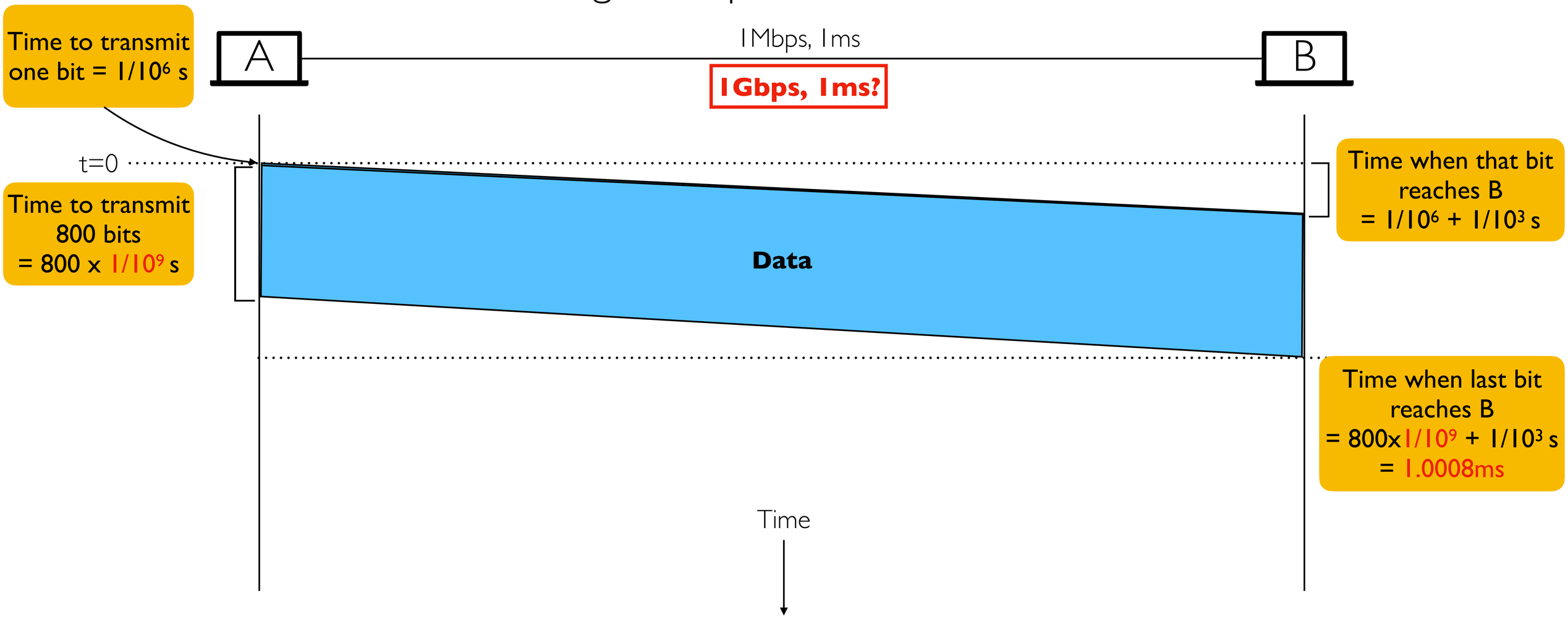
Packet Delay

Sending 100B packets from A to B?



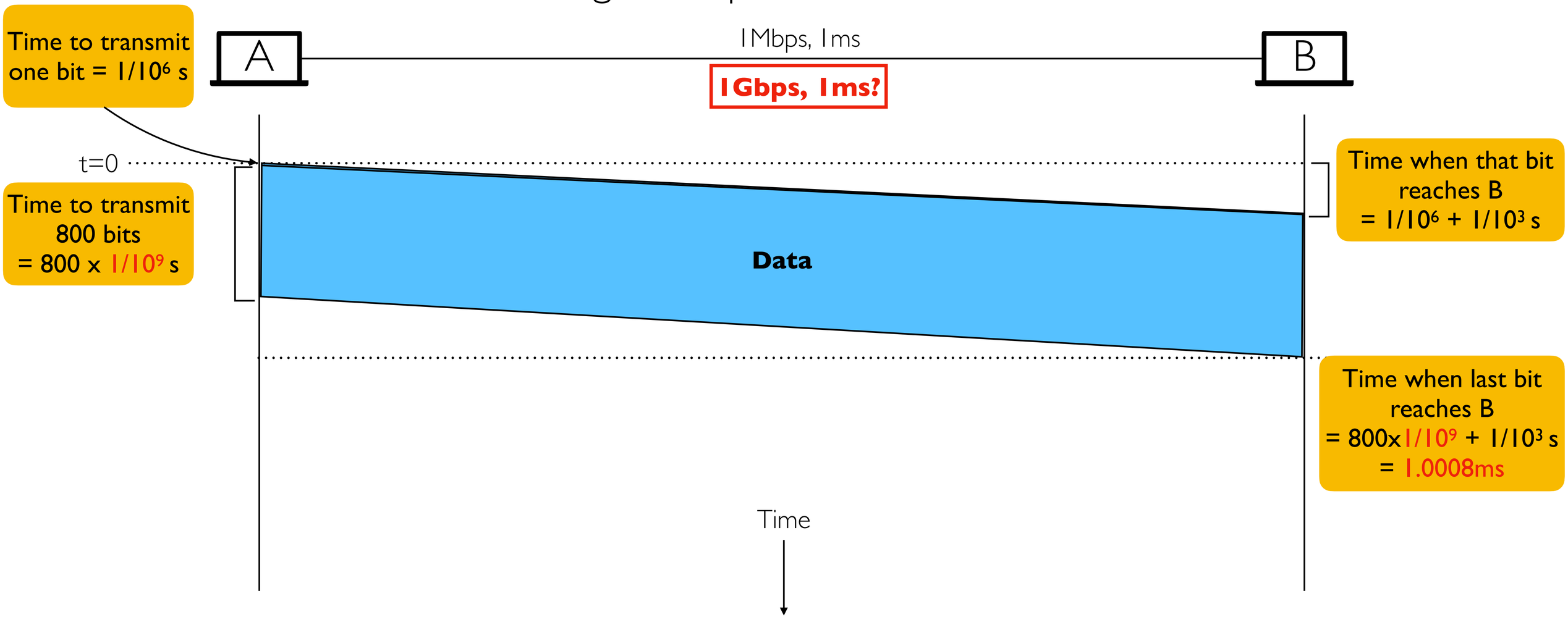
Packet Delay

Sending 100B packets from A to B?



Packet Delay

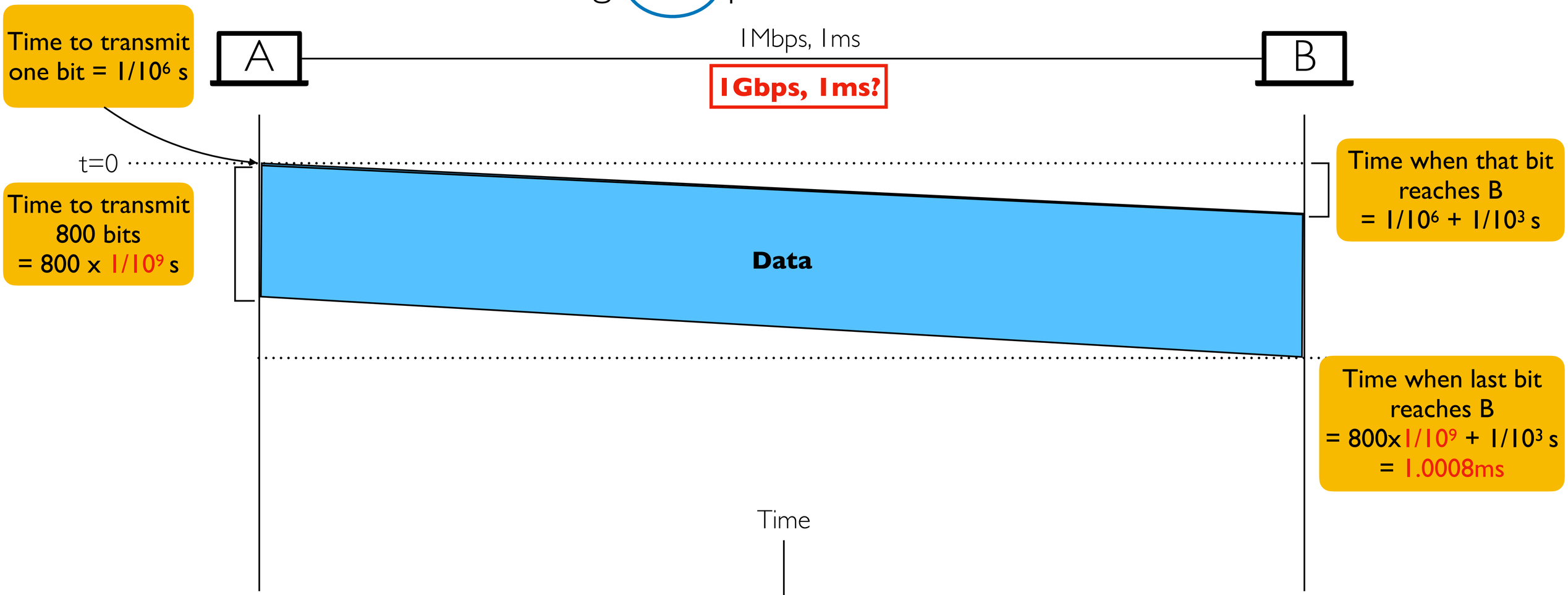
Sending 100B packets from A to B?



Packet Delay

1GB file in 100B packets

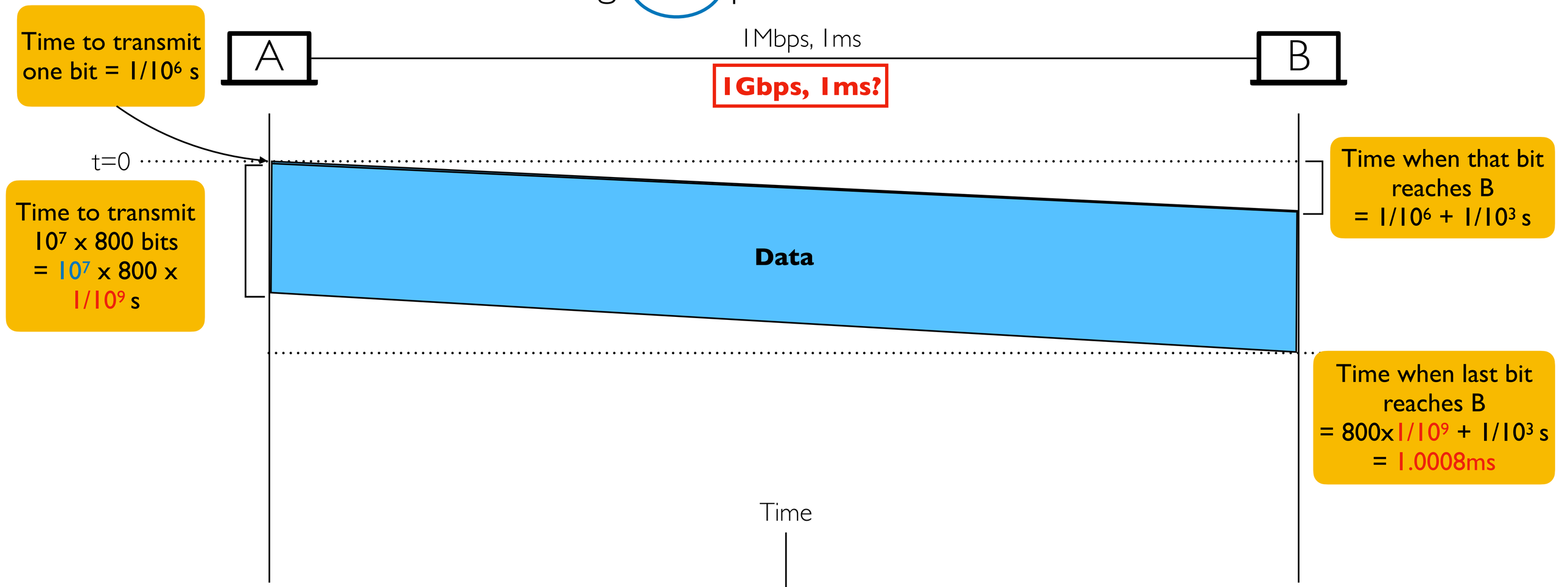
Sending 100B packets from A to B?



Packet Delay

1GB file in 100B packets

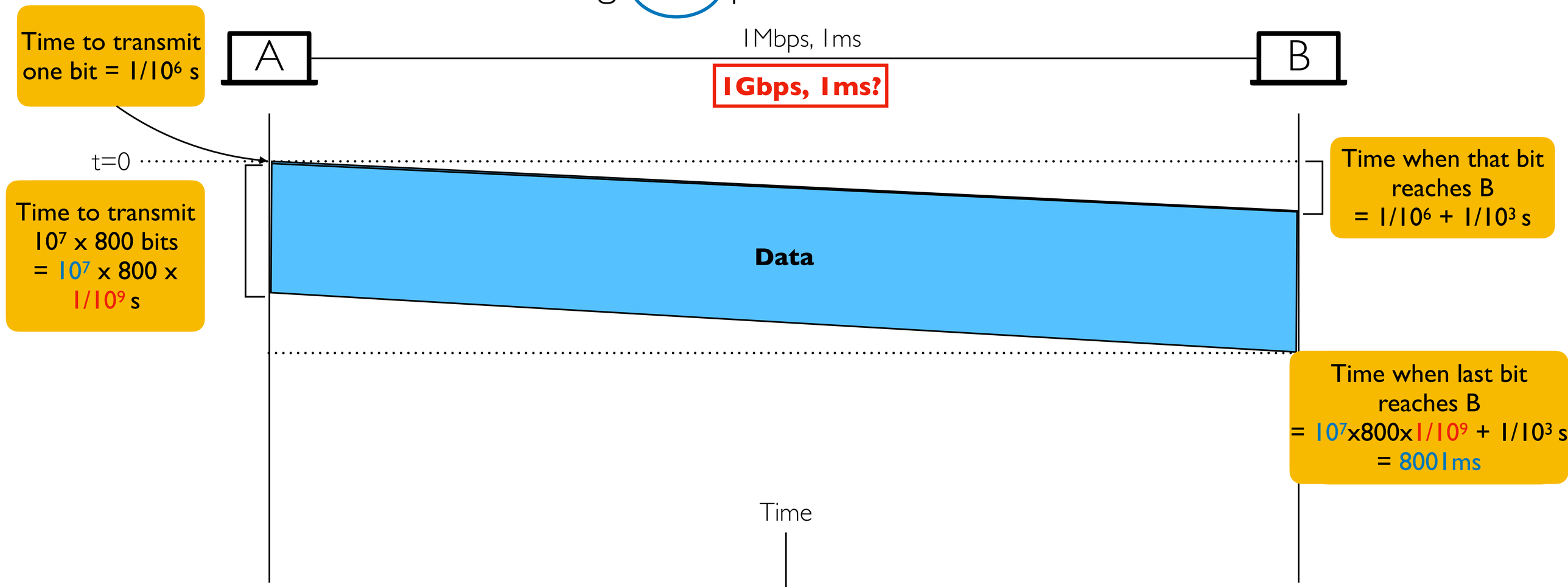
Sending 100B packets from A to B?



Packet Delay

1GB file in 100B packets

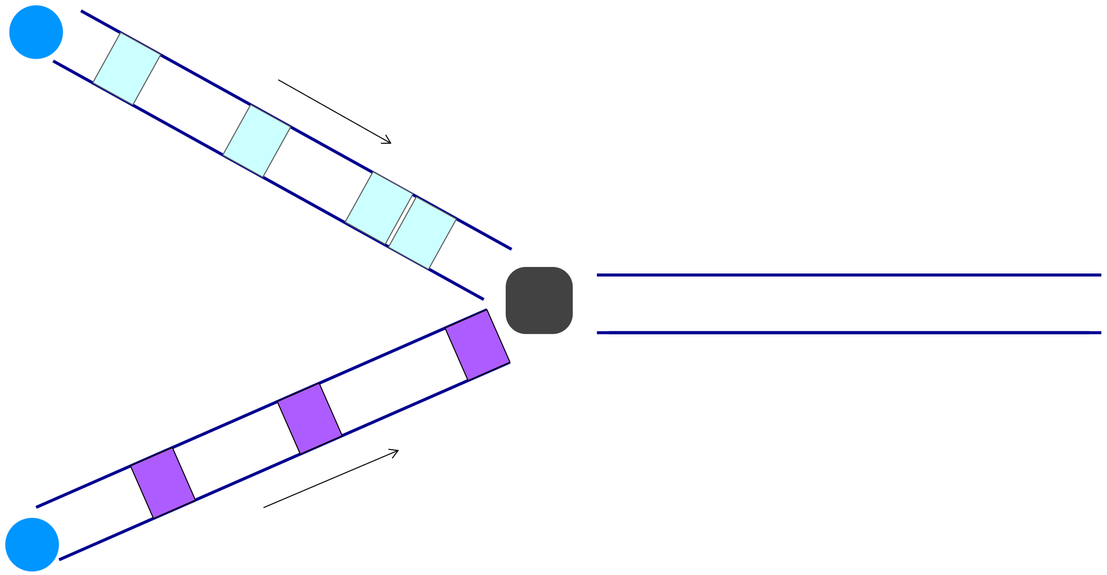
Sending 100B packets from A to B?



Questions?

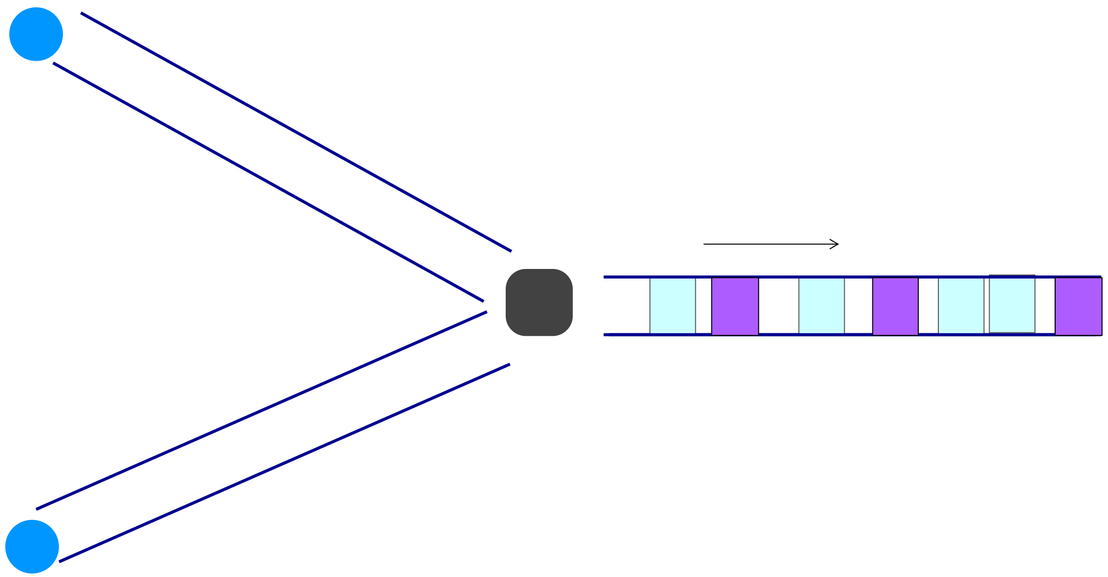
Queueing Delay: "Pipe" View

- How long does a packet have to sit in a buffer before it is processed



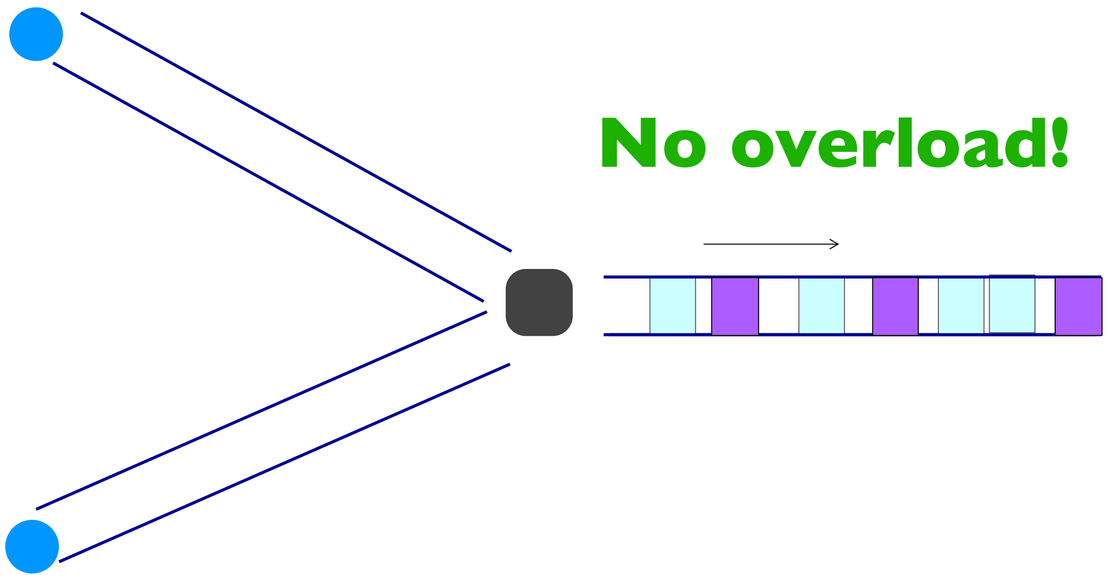
Queueing Delay: "Pipe" View

- How long does a packet have to sit in a buffer before it is processed



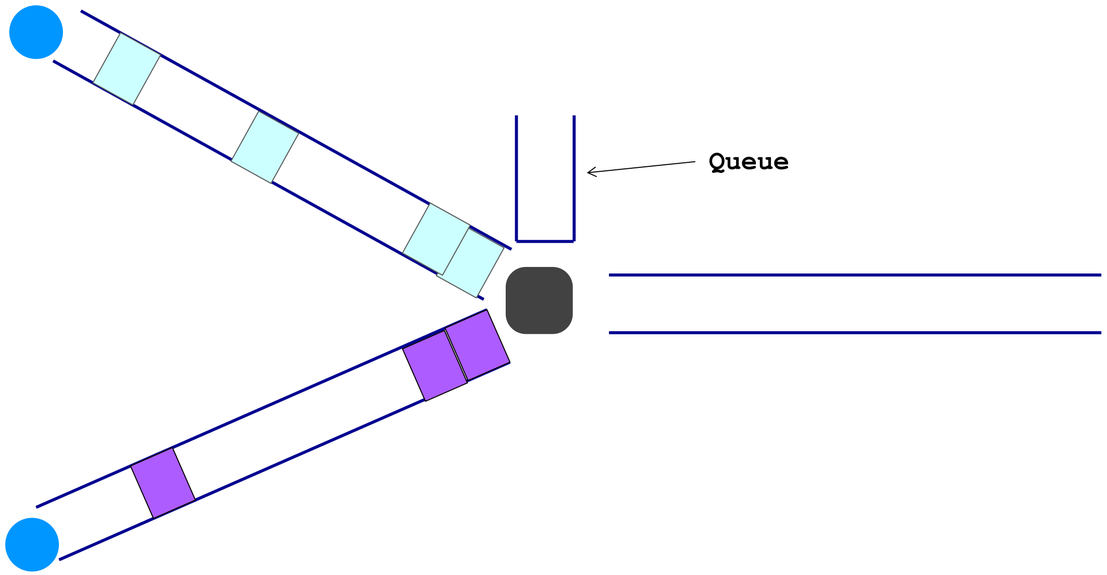
Queueing Delay: "Pipe" View

- How long does a packet have to sit in a buffer before it is processed



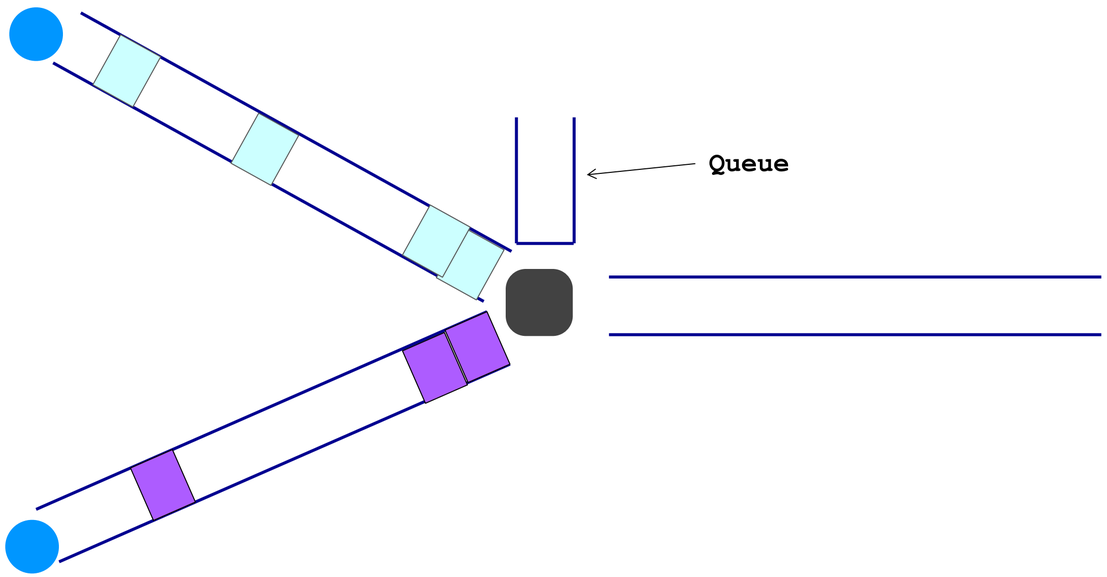
Queueing Delay: "Pipe" View

- How long does a packet have to sit in a buffer before it is processed



Queueing Delay: "Pipe" View

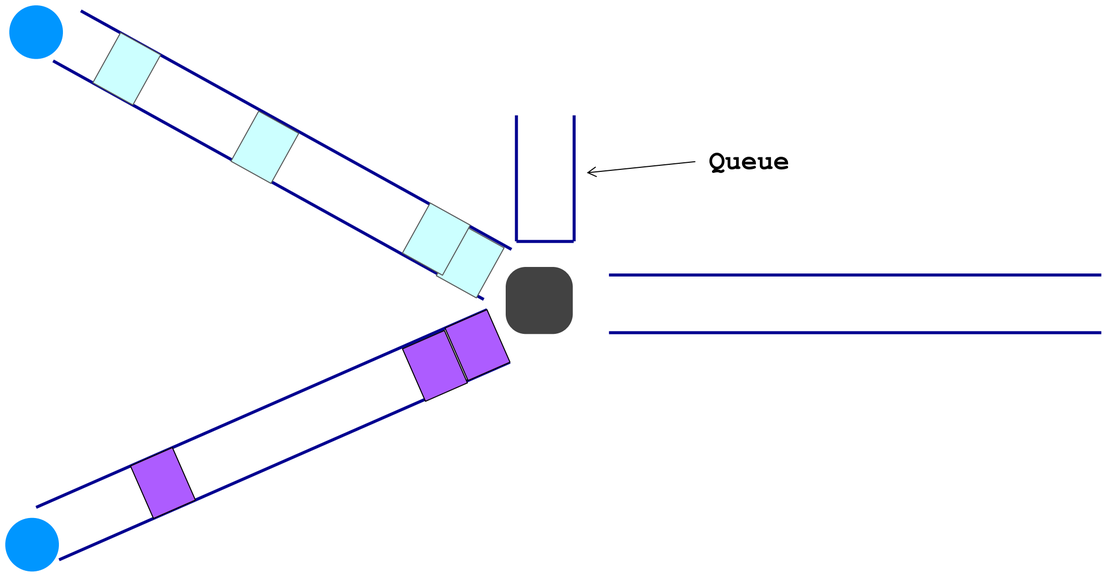
- How long does a packet have to sit in a buffer before it is processed



Transient overload!

Queueing Delay: "Pipe" View

- How long does a packet have to sit in a buffer before it is processed

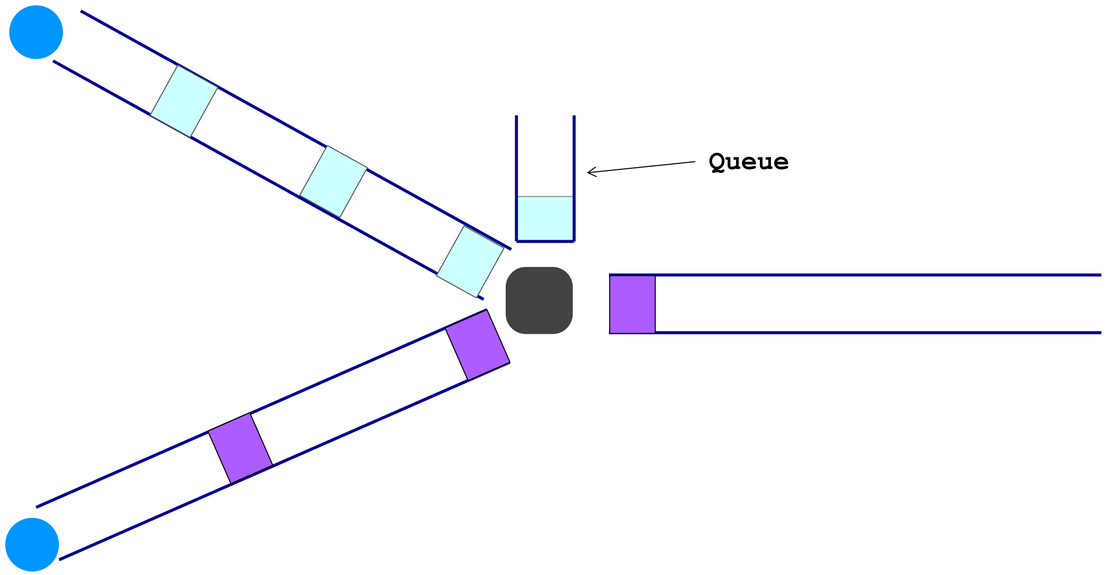


Transient overload!

Not a rare event.

Queueing Delay: "Pipe" View

- How long does a packet have to sit in a buffer before it is processed

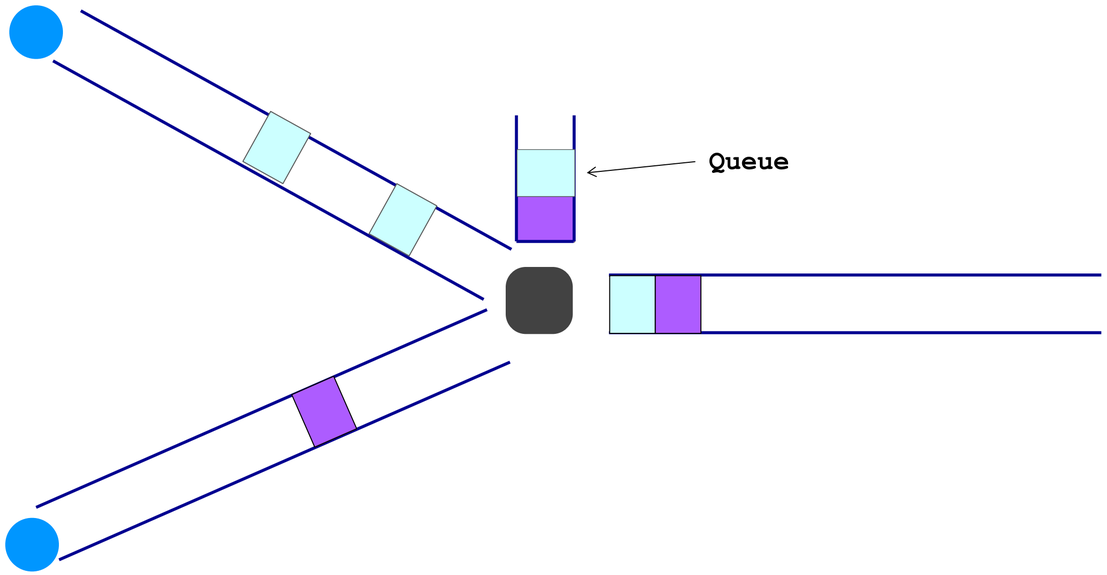


Transient overload!

Not a rare event.

Queueing Delay: "Pipe" View

- How long does a packet have to sit in a buffer before it is processed

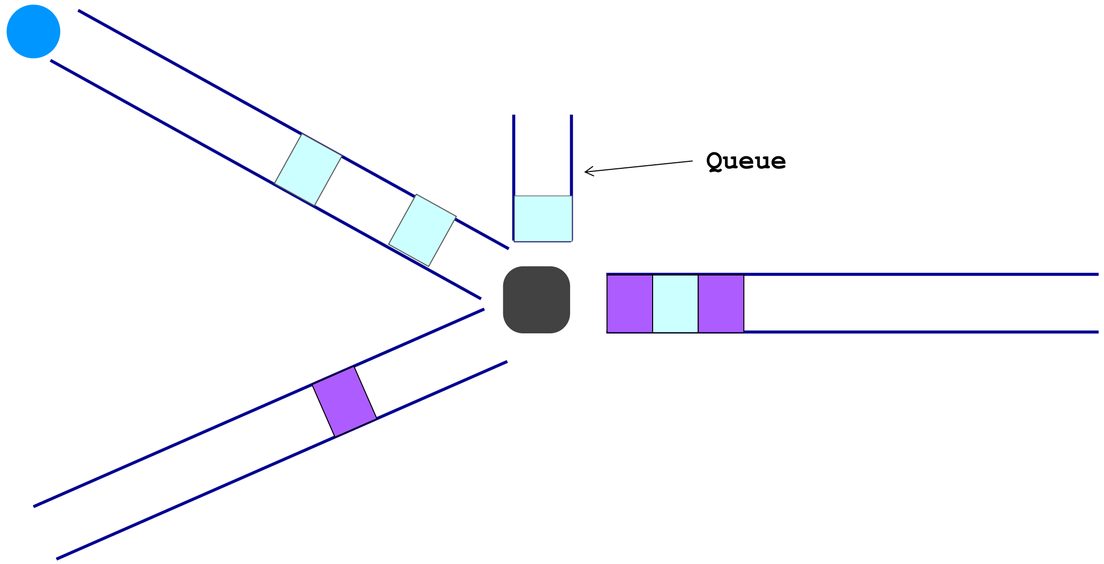


Transient overload!

Not a rare event.

Queueing Delay: "Pipe" View

- How long does a packet have to sit in a buffer before it is processed

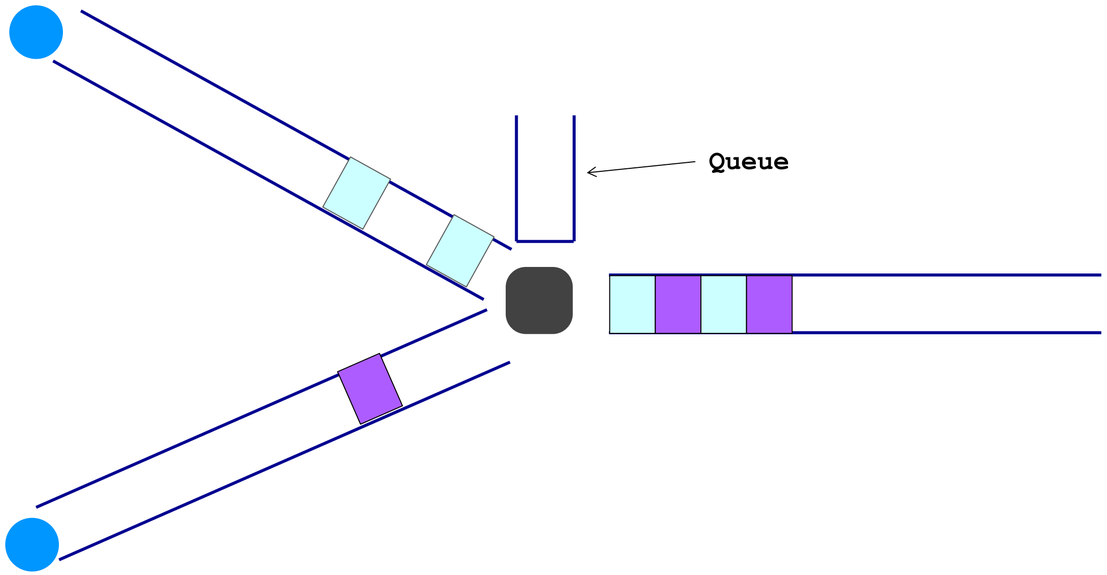


Transient overload!

Not a rare event.

Queueing Delay: "Pipe" View

- How long does a packet have to sit in a buffer before it is processed

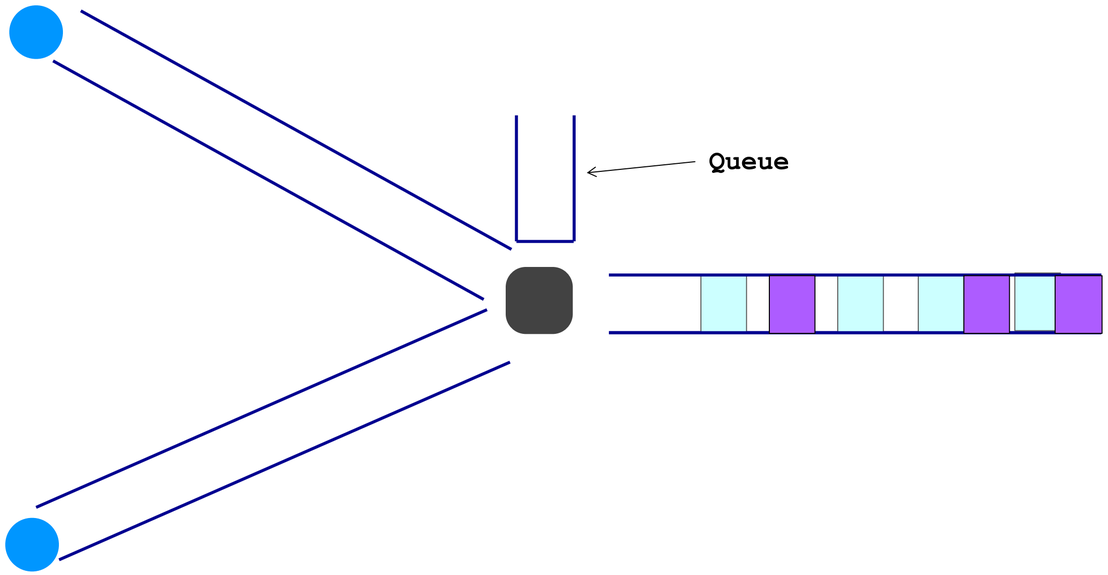


Transient overload!

Not a rare event.

Queueing Delay: "Pipe" View

- How long does a packet have to sit in a buffer before it is processed

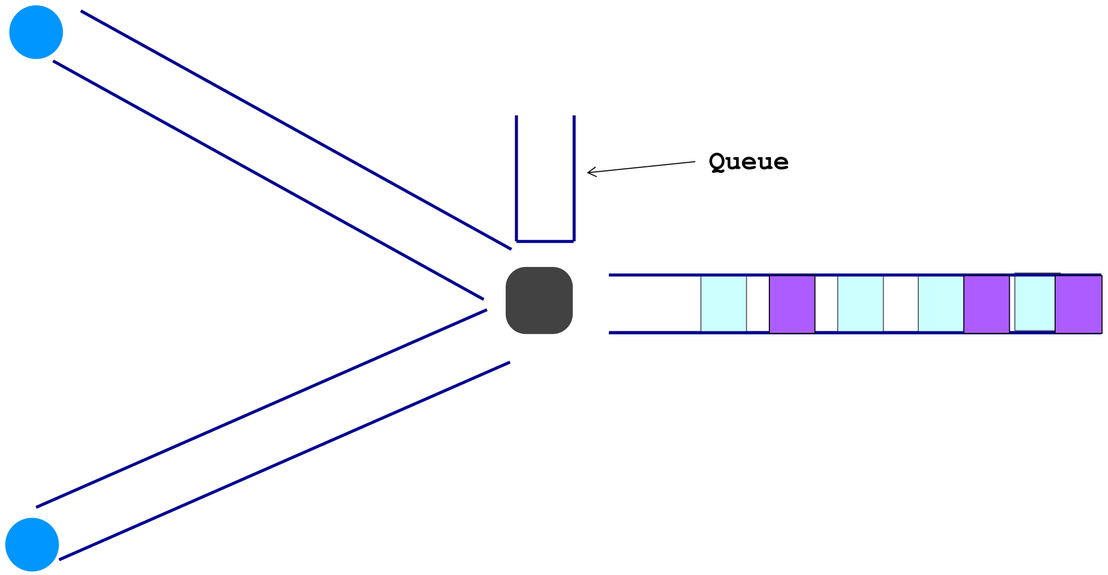


Transient overload!

Not a rare event.

Queueing Delay: "Pipe" View

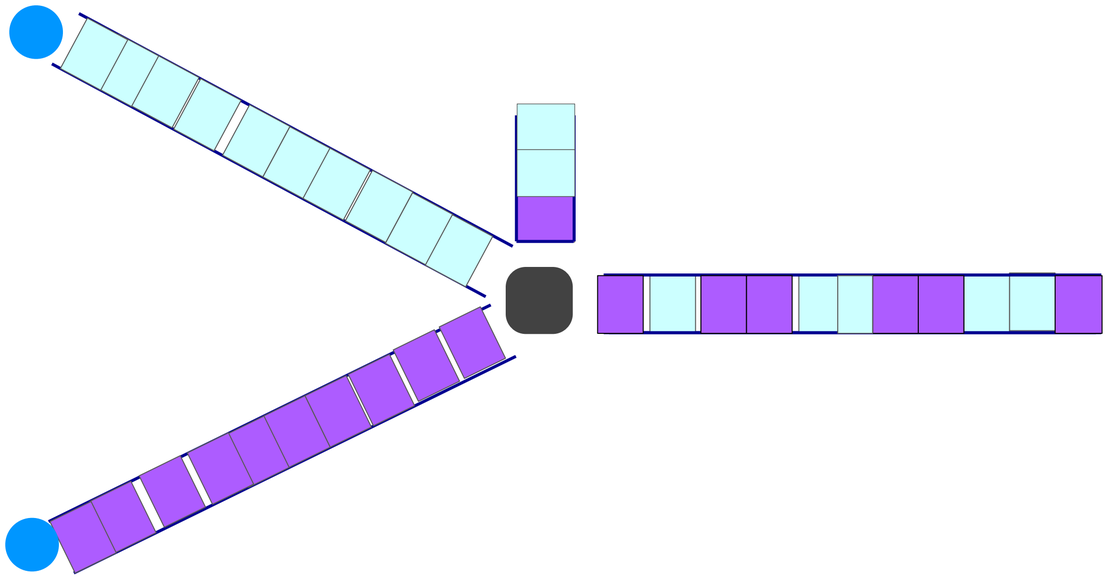
- How long does a packet have to sit in a buffer before it is processed



Queues absorb transient bursts but introduce queueing delay

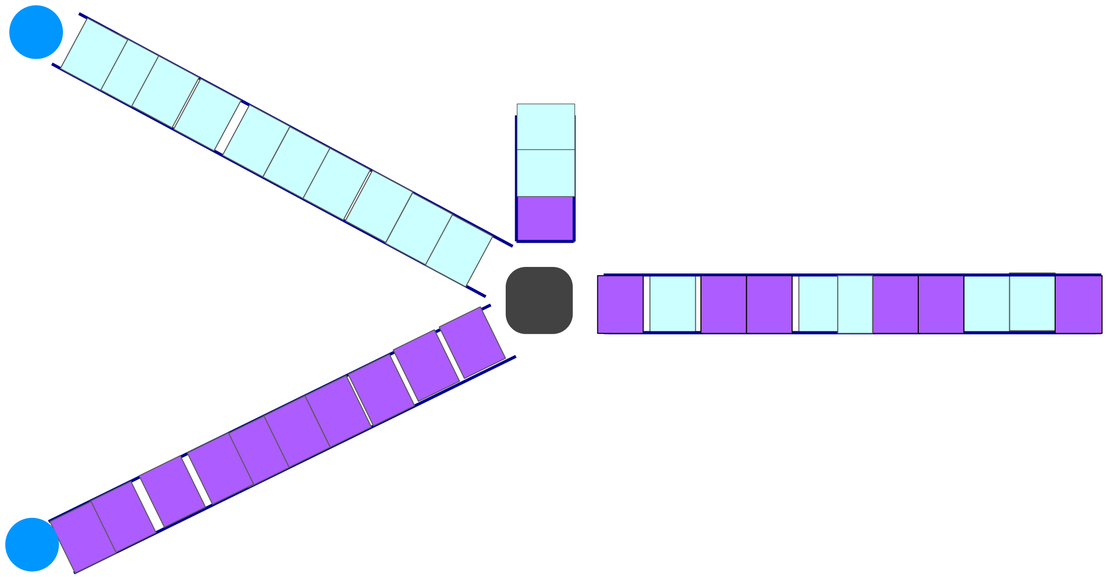
Queueing Delay: "Pipe" View

- How long does a packet have to sit in a buffer before it is processed



Queueing Delay: "Pipe" View

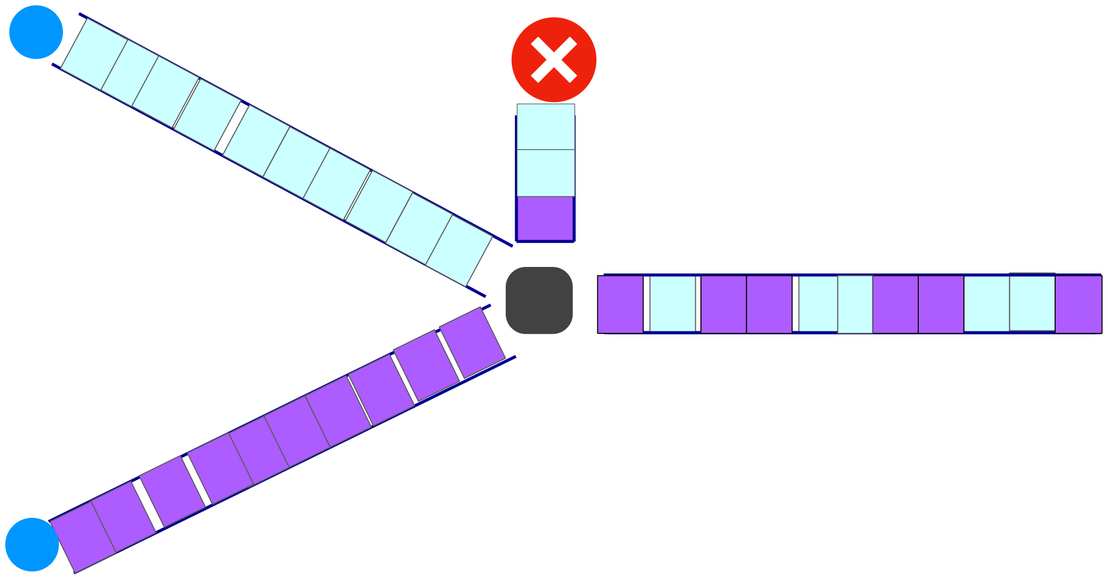
- How long does a packet have to sit in a buffer before it is processed



What about persistent overload?

Queueing Delay: "Pipe" View

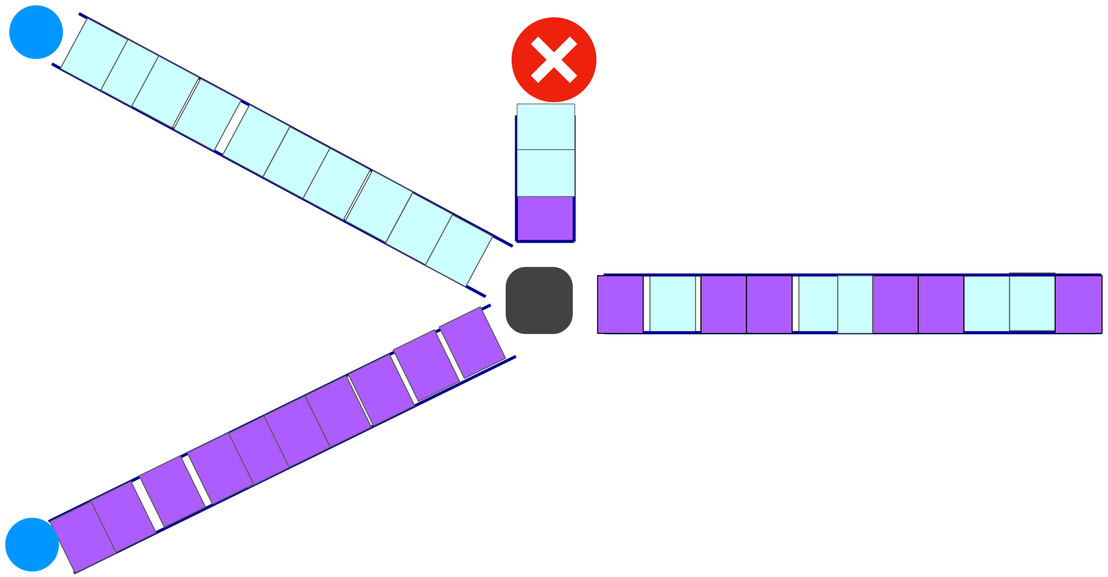
- How long does a packet have to sit in a buffer before it is processed



What about persistent overload?

Queueing Delay: "Pipe" View

- How long does a packet have to sit in a buffer before it is processed



What about persistent overload?

Leads to packet loss

Queueing Delay

Queueing Delay

- **If arrival rate > departure rate**
  - Approaches infinity (assuming an infinite buffer)
  - In practice, finite buffer = loss

Queueing Delay

- **If arrival rate > departure rate**
  - Approaches infinity (assuming an infinite buffer)
  - In practice, finite buffer = loss
- **If arrival rate < departure rate**
  - Depends on burst rate

Queueing Delay

- How long does a packet have to sit in a buffer before it is processed

Queueing Delay

- How long does a packet have to sit in a buffer before it is processed
- Depends on traffic pattern

Queueing Delay

- How long does a packet have to sit in a buffer before it is processed
- Depends on traffic pattern
- Characterized by statistical measures
  - Average queueing delay
  - Average arrival rate
  - Average departure rate

Basic Queueing Theory Terminology

Basic Queueing Theory Terminology

- **Arrival process:** how packets arrive
  - Characterized by average arrival rate **A**

Basic Queueing Theory Terminology

- **Arrival process:** how packets arrive
  - Characterized by average arrival rate **A**
- **W:** average time packets wait in the queue
  - W for "waiting time"

Basic Queueing Theory Terminology

- **Arrival process:** how packets arrive
  - Characterized by average arrival rate **A**
- **W:** average time packets wait in the queue
  - W for "waiting time"
- **L**: average number of packets waiting in the queue
  - L for "length of the queue"

Little's Law (1969)

A: avg. packet arrival rate (/s)

L: avg. # of packets waiting in queue

W: avg. time packets wait in queue

Little's Law (1969)

A: avg. packet arrival rate (/s)

L: avg. # of packets waiting in queue

W: avg. time packets wait in queue

L = A x W

Little's Law (1969)

A: avg. packet arrival rate (/s)

L: avg. # of packets waiting in queue

W: avg. time packets wait in queue

L = A x W

or,

W = L / A

Processing Delay

Processing Delay

- How long does a switch take to process this packet?
  - Typically assume this is negligible

Delay

- Consists of four components
  - Transmission Delay
  - Propagation Delay
  - Queueing Delay
  - Processing Delay

Transmission Delay, Propagation Delay: Due to link properties

Queueing Delay, Processing Delay: Due to traffic matrix and switch internals

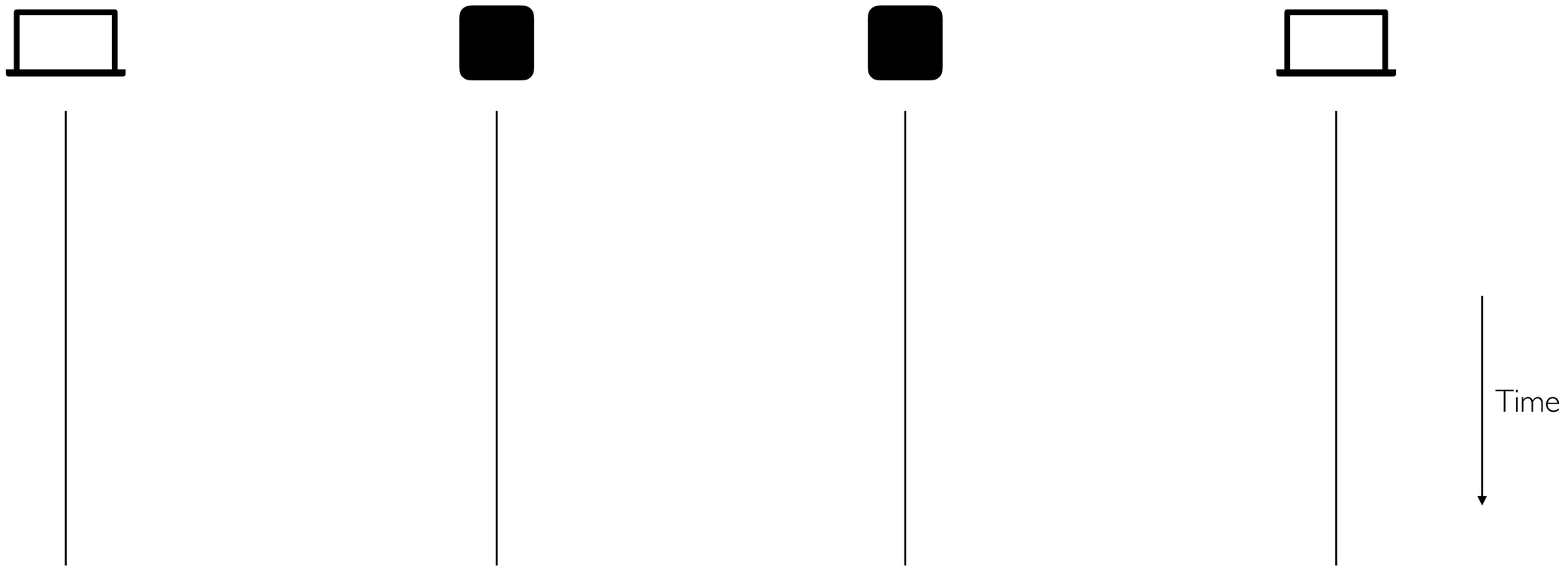
End-to-end Delay

T Transmission Delay

Q Queuing Delay

P Propagation Delay

Pr Processing Delay



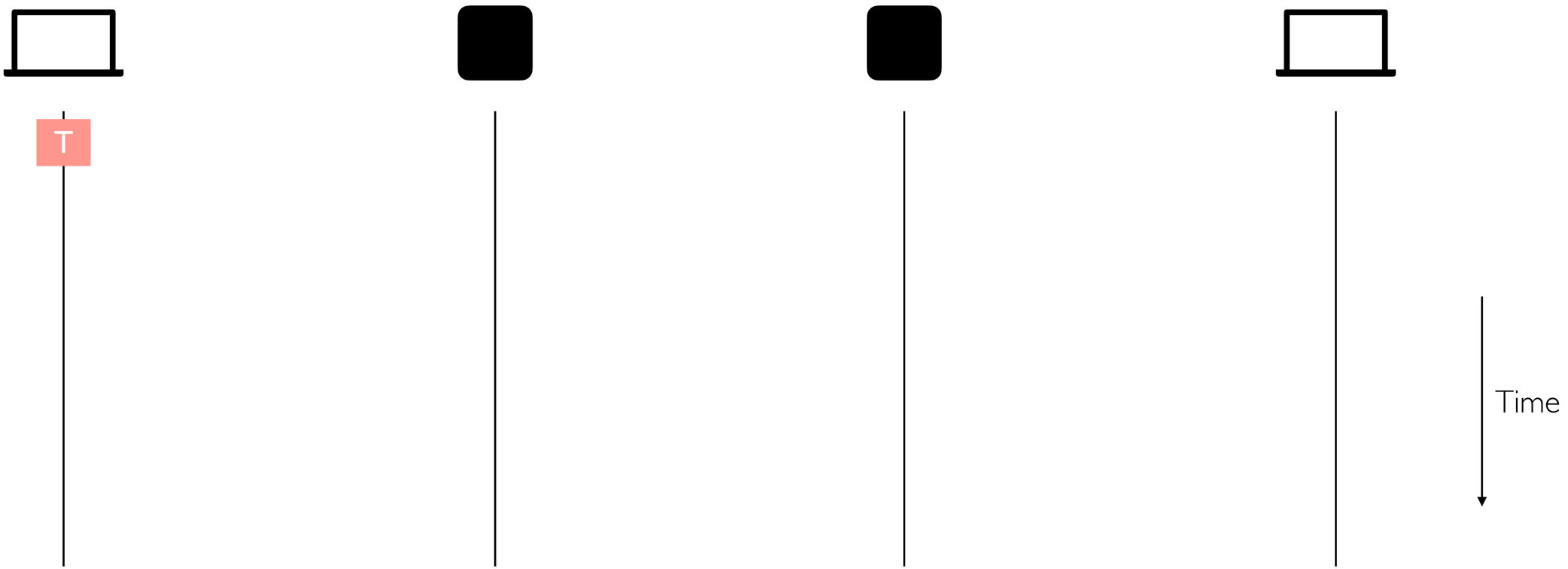
End-to-end Delay

T Transmission Delay

Q Queuing Delay

P Propagation Delay

Pr Processing Delay



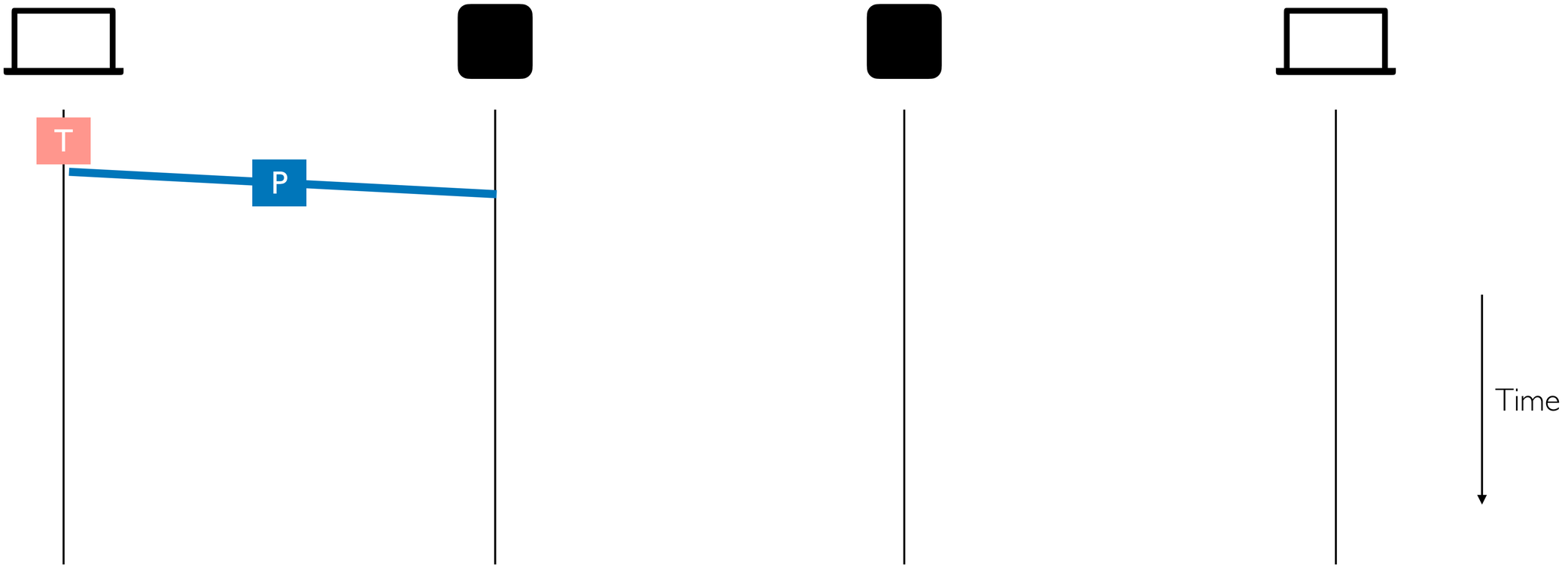
End-to-end Delay

T Transmission Delay

Q Queuing Delay

P Propagation Delay

Pr Processing Delay



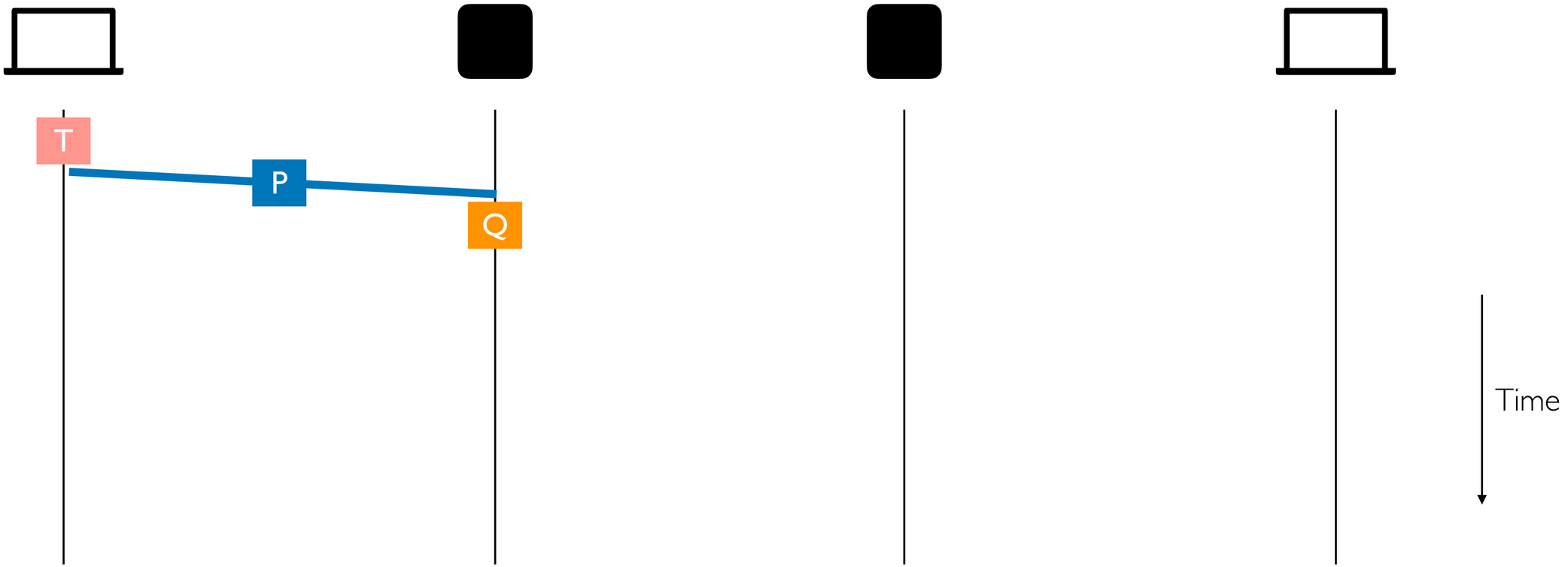
End-to-end Delay

T Transmission Delay

Q Queuing Delay

P Propagation Delay

Pr Processing Delay



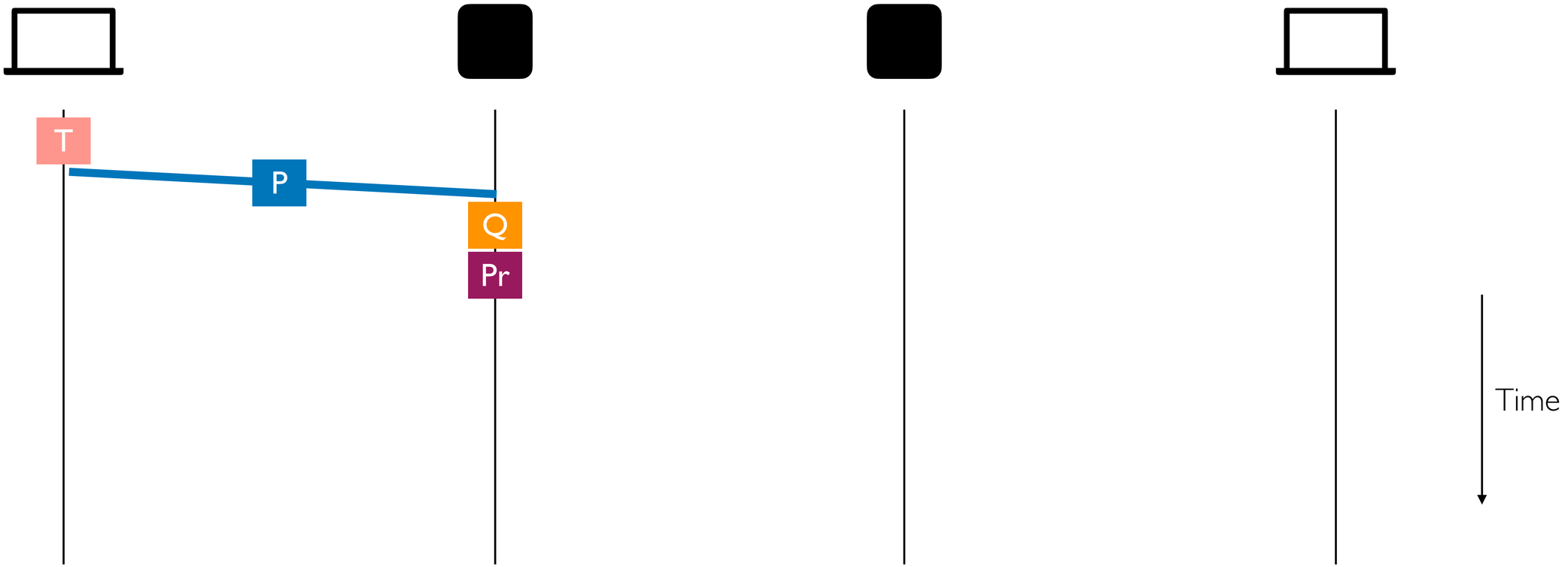
End-to-end Delay

T Transmission Delay

Q Queuing Delay

P Propagation Delay

Pr Processing Delay



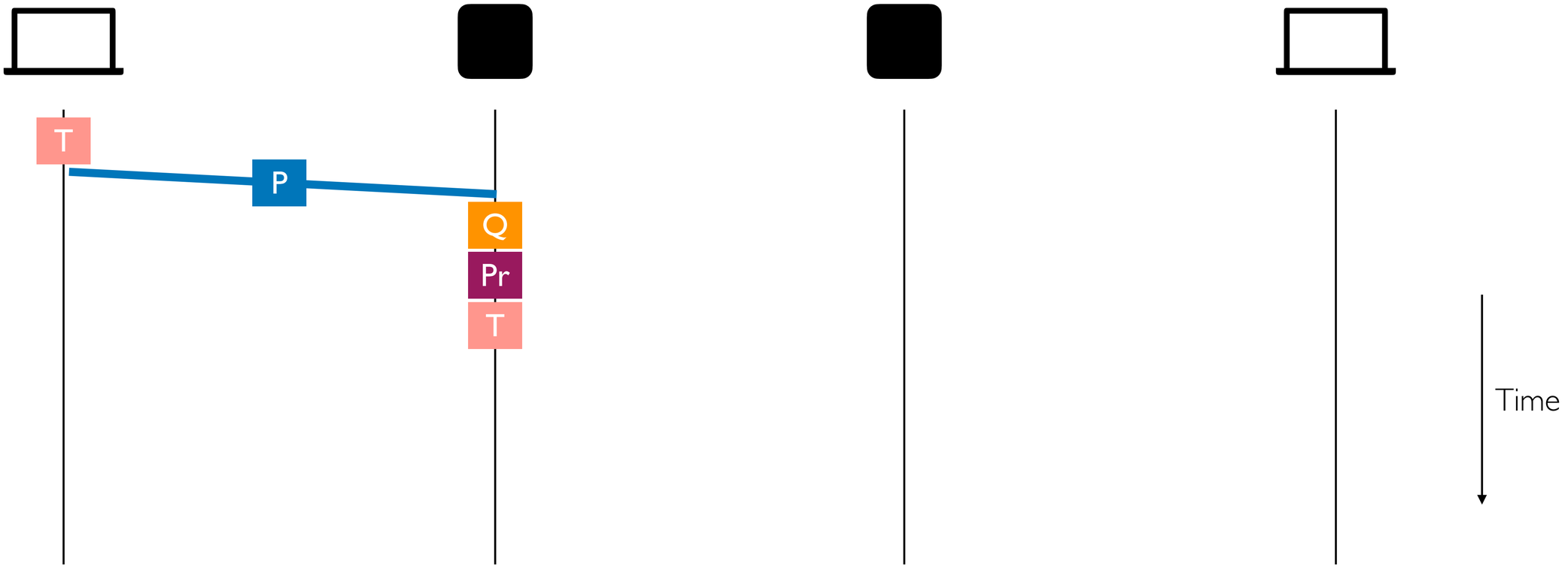
End-to-end Delay

T Transmission Delay

Q Queuing Delay

P Propagation Delay

Pr Processing Delay



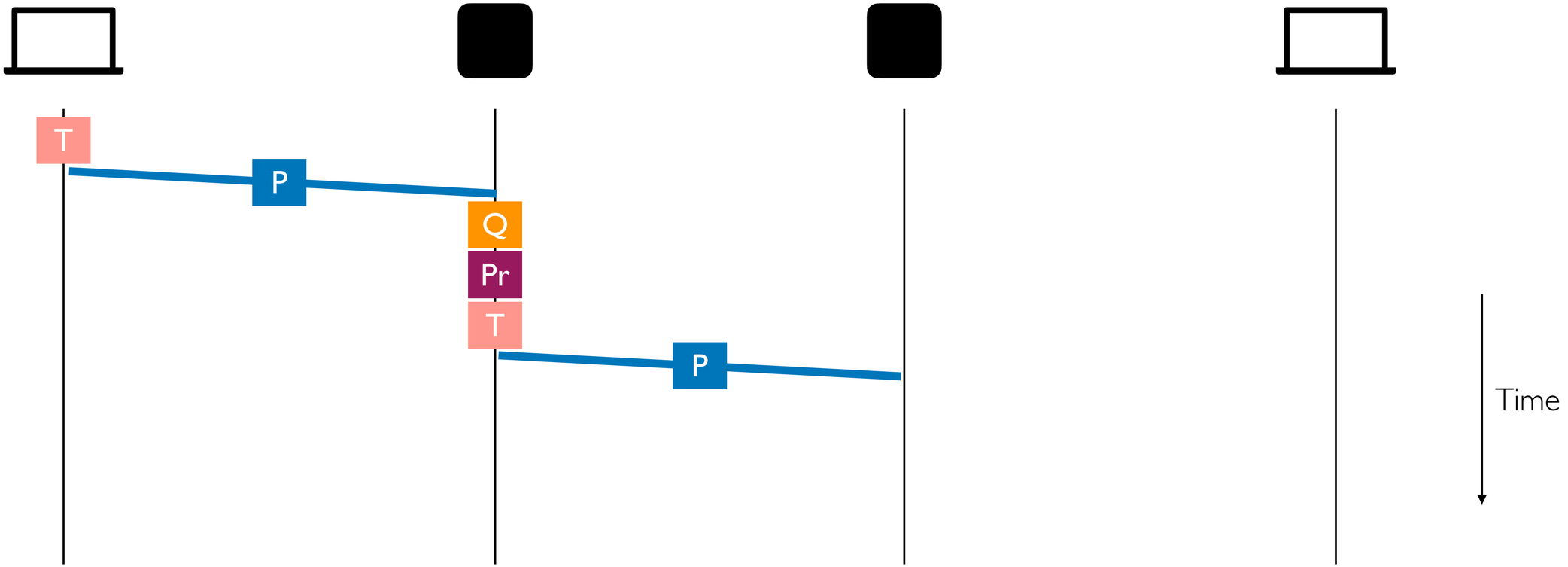
End-to-end Delay

T Transmission Delay

Q Queuing Delay

P Propagation Delay

Pr Processing Delay



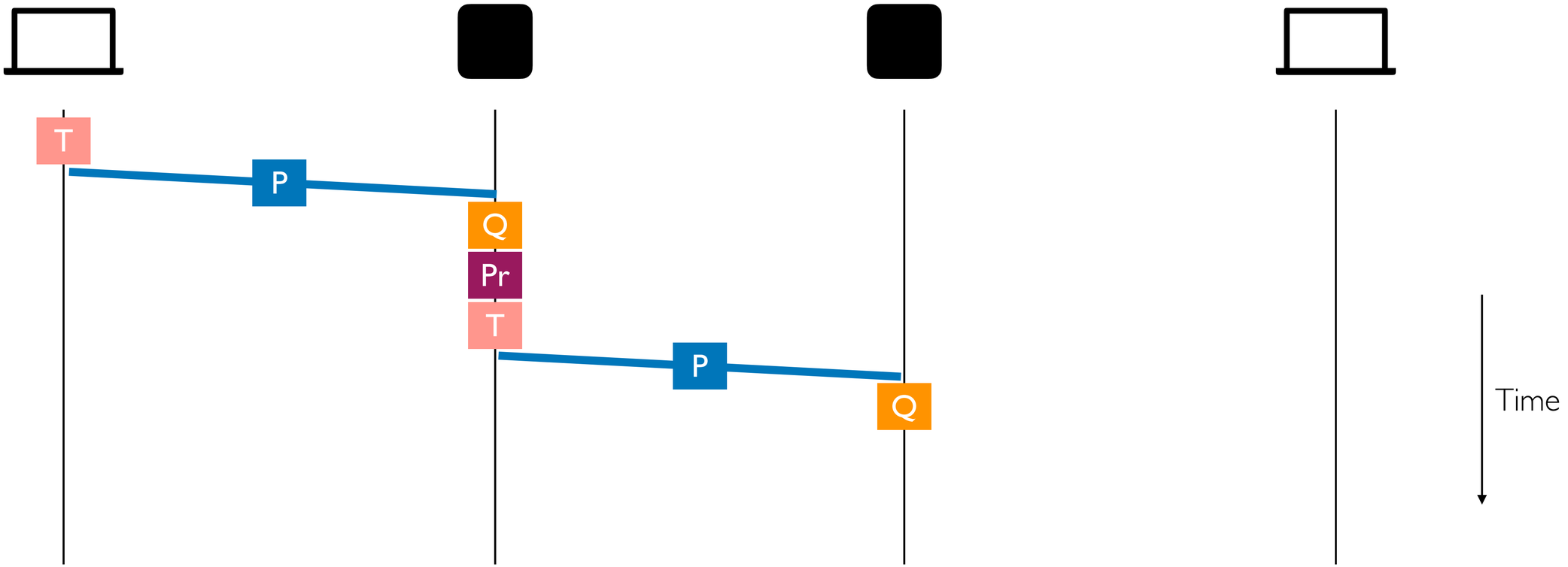
End-to-end Delay

T Transmission Delay

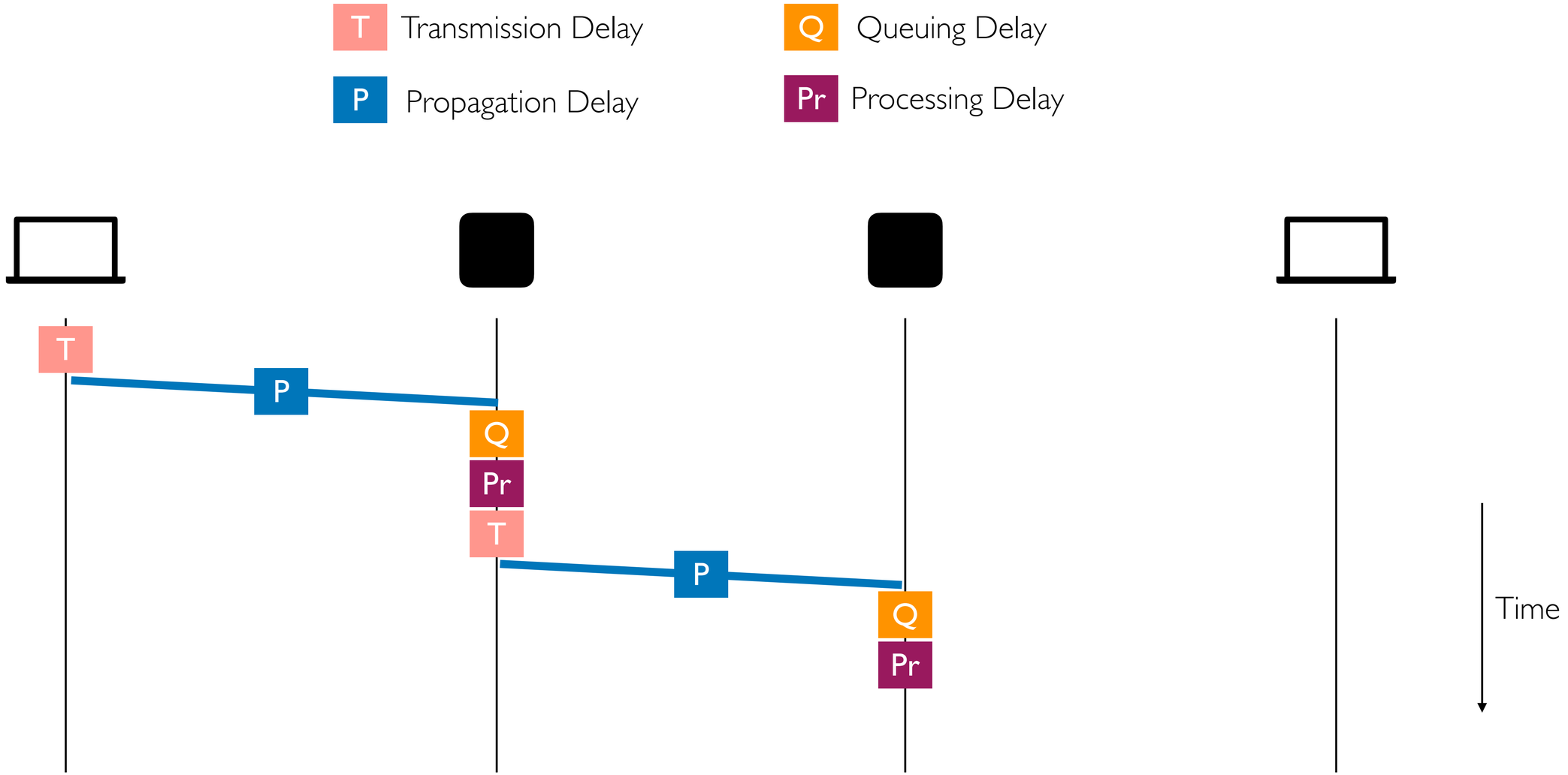
Q Queuing Delay

P Propagation Delay

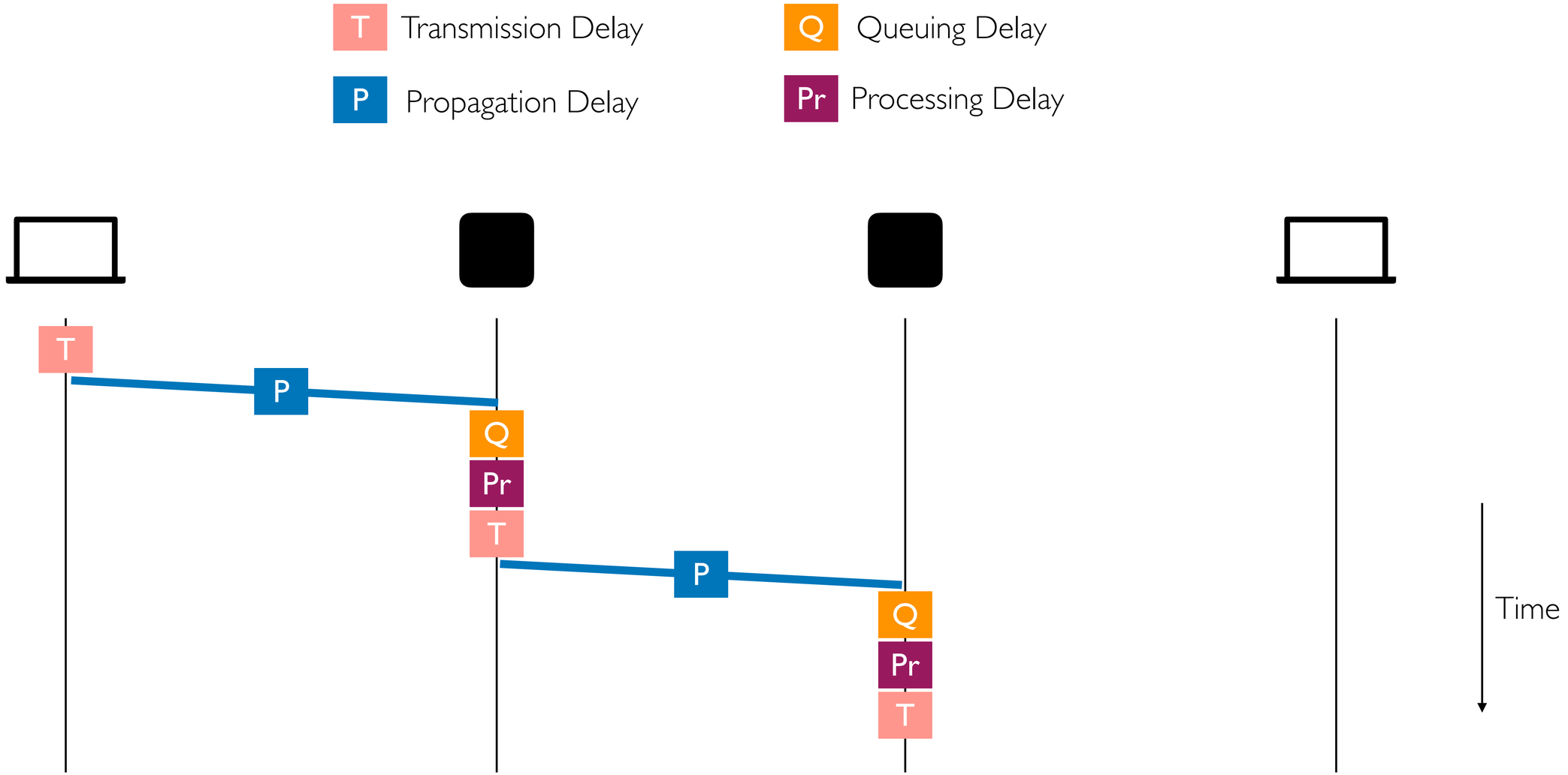
Pr Processing Delay



End-to-end Delay



End-to-end Delay



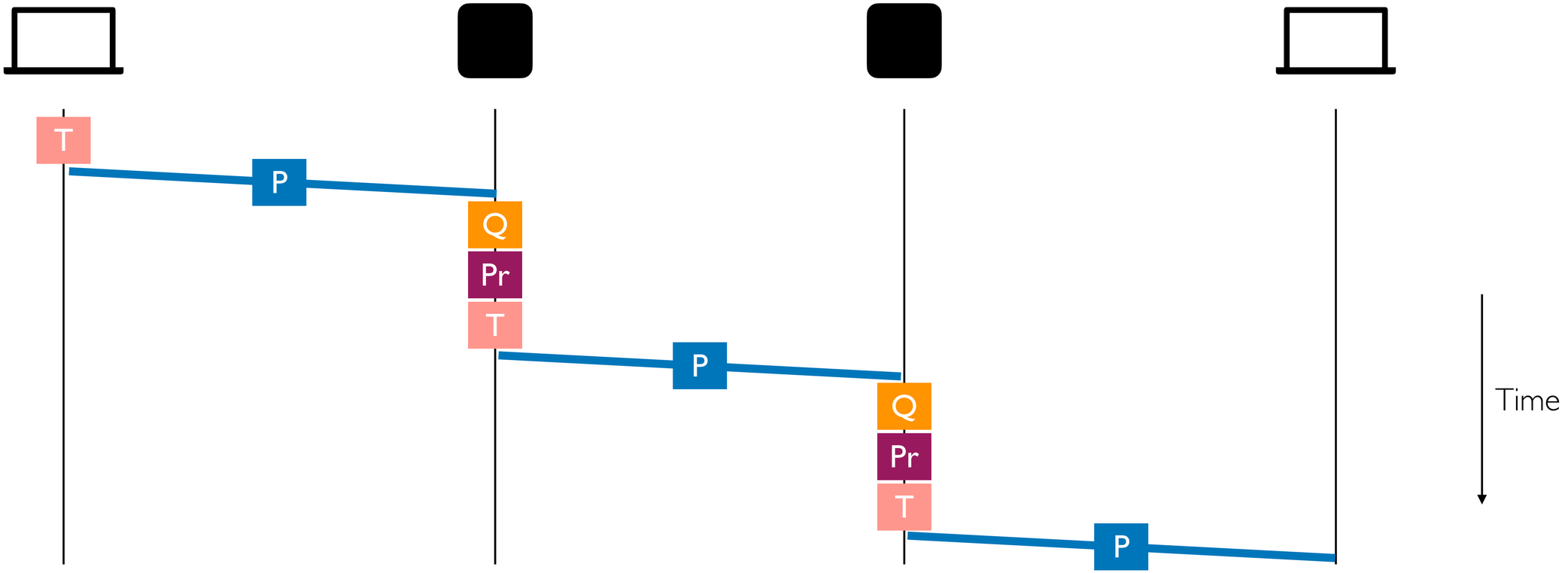
End-to-end Delay

T Transmission Delay

Q Queuing Delay

P Propagation Delay

Pr Processing Delay



Questions?

Loss

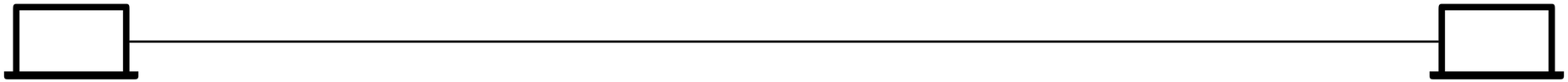
Loss

- What fraction of the packets sent to a destination are dropped?

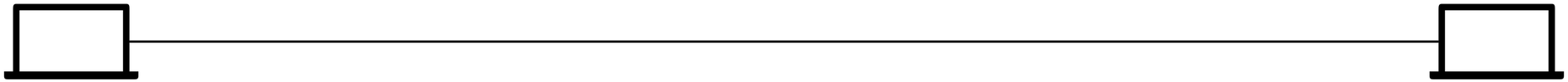
Throughput

- At what rate is the destination receiving data from the source?
  - **Data size / transfer time**

Throughput



Throughput



File of size **F** bits
Packets of size **L** bits

Throughput

File of size **F** bits
Packets of size **L** bits

Throughput

File of size **F** bits
Packets of size **L** bits

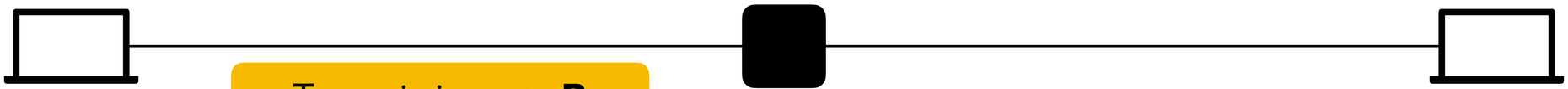
- Transfer time = **F**/**R** + propagation delay

Throughput

File of size **F** bits
Packets of size **L** bits

- Transfer time = **F**/**R** + propagation delay
- Average throughput = **R**

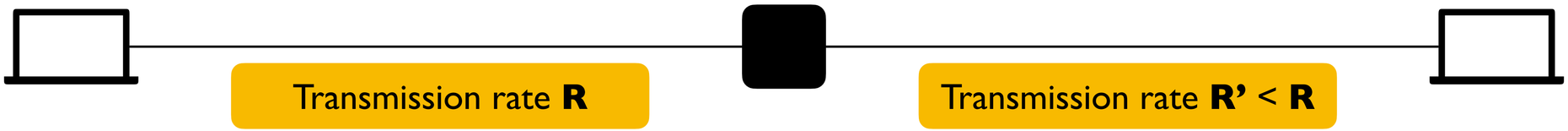
Throughput



Transmission rate **R**

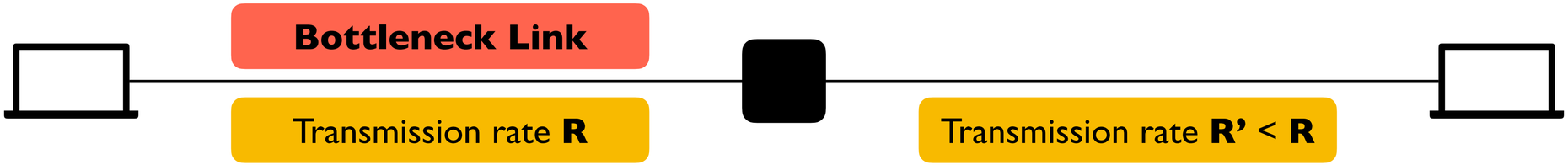
File of size **F** bits
Packets of size **L** bits

Throughput



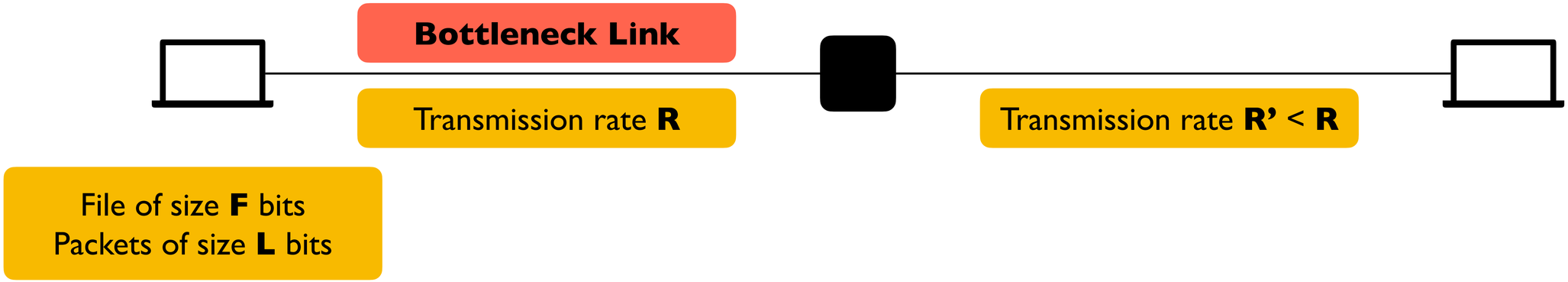
File of size **F** bits
Packets of size **L** bits

Throughput



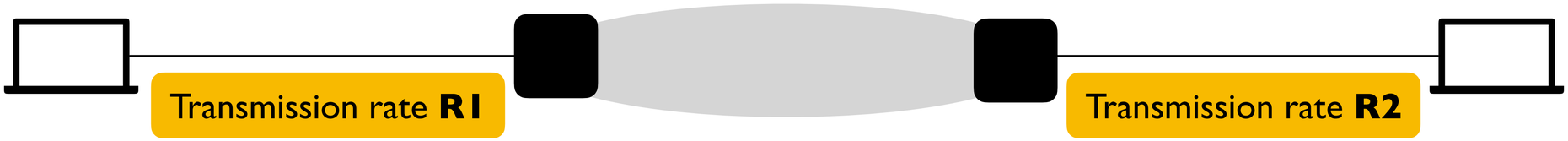
File of size **F** bits
Packets of size **L** bits

Throughput

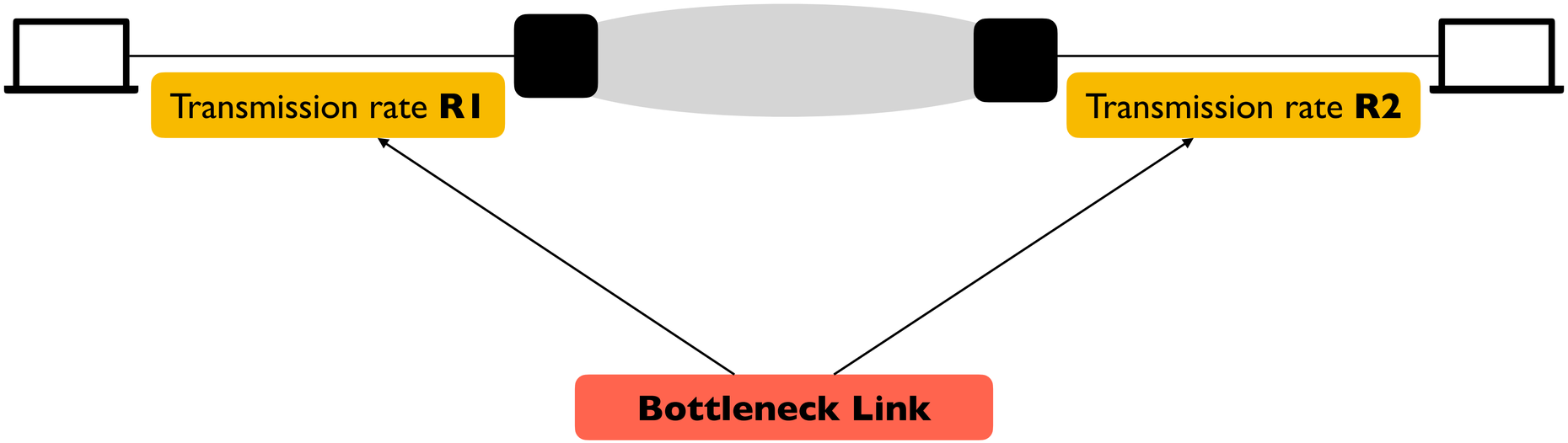


- Average throughput = min {**R, R'**} = **R**

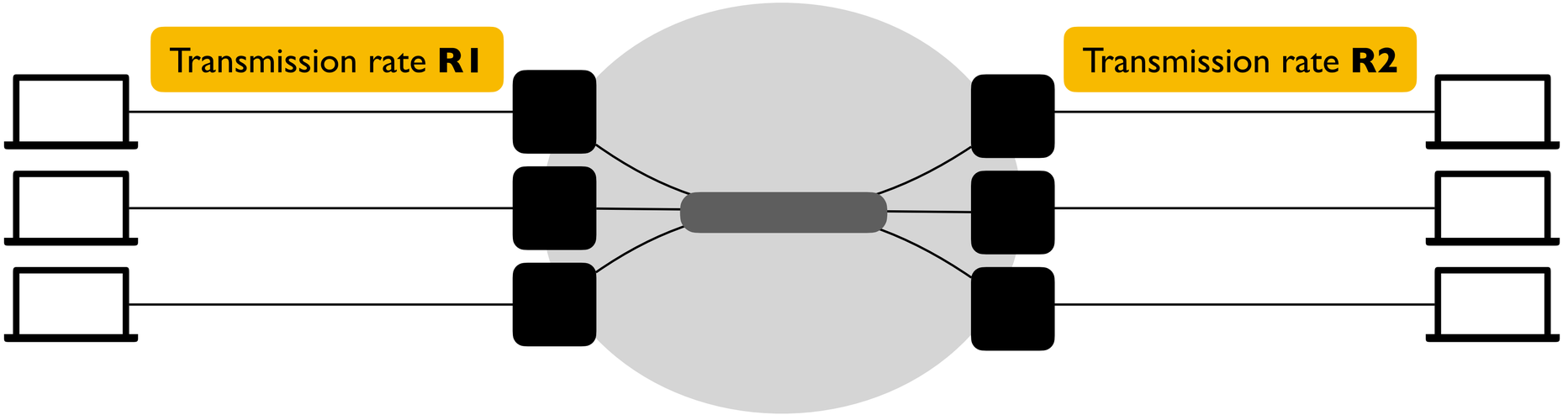
Throughput



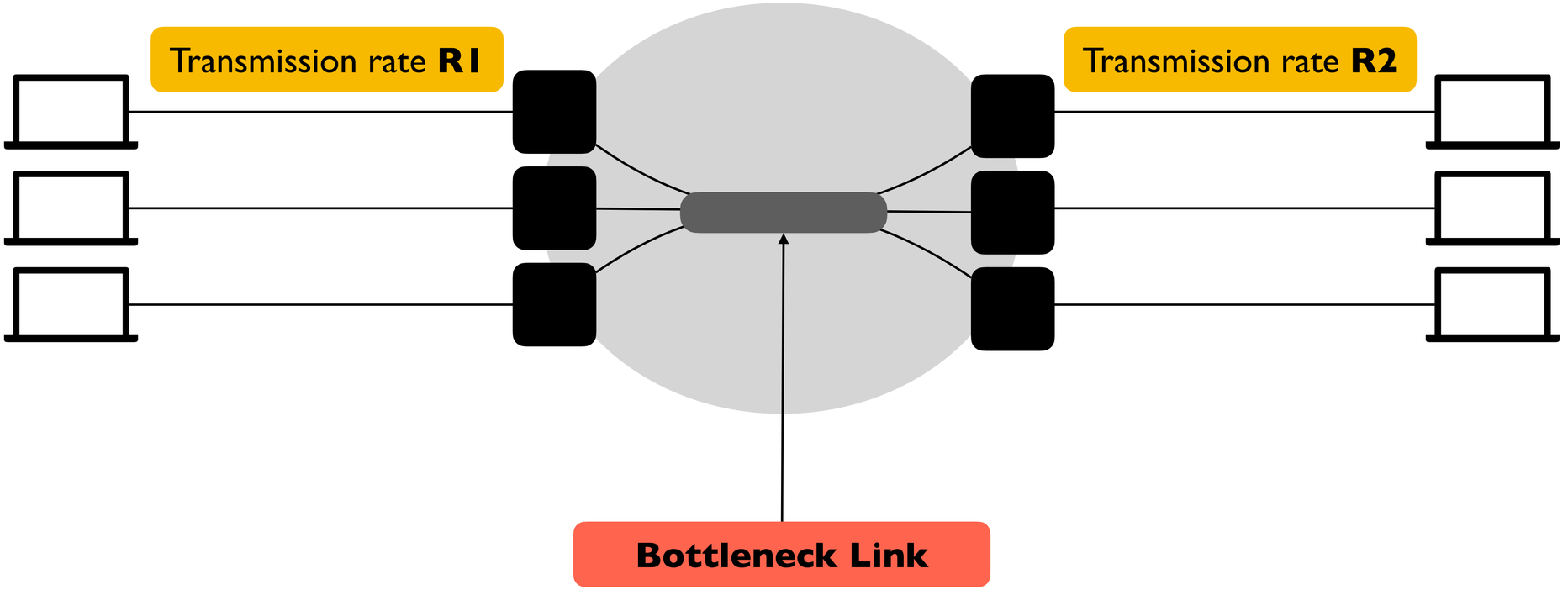
Throughput


Transmission rate **R1**
Transmission rate **R2**
Bottleneck Link

Throughput



Throughput


Transmission rate **R1**
Transmission rate **R2**
Bottleneck Link

Questions?

Taking Stock

Taking Stock

- **What & how of the Internet**
  - A slightly detailed look at physical infrastructure

Taking Stock

- **What & how of the Internet**
  - A slightly detailed look at physical infrastructure
- **Sharing networks**
  - Reserve or on-demand?

Taking Stock

- **What & how of the Internet**
  - A slightly detailed look at physical infrastructure
- **Sharing networks**
  - Reserve or on-demand?
- **Evaluating networks**
  - Delay (transmission, propagation, queueing, processing), loss and throughput