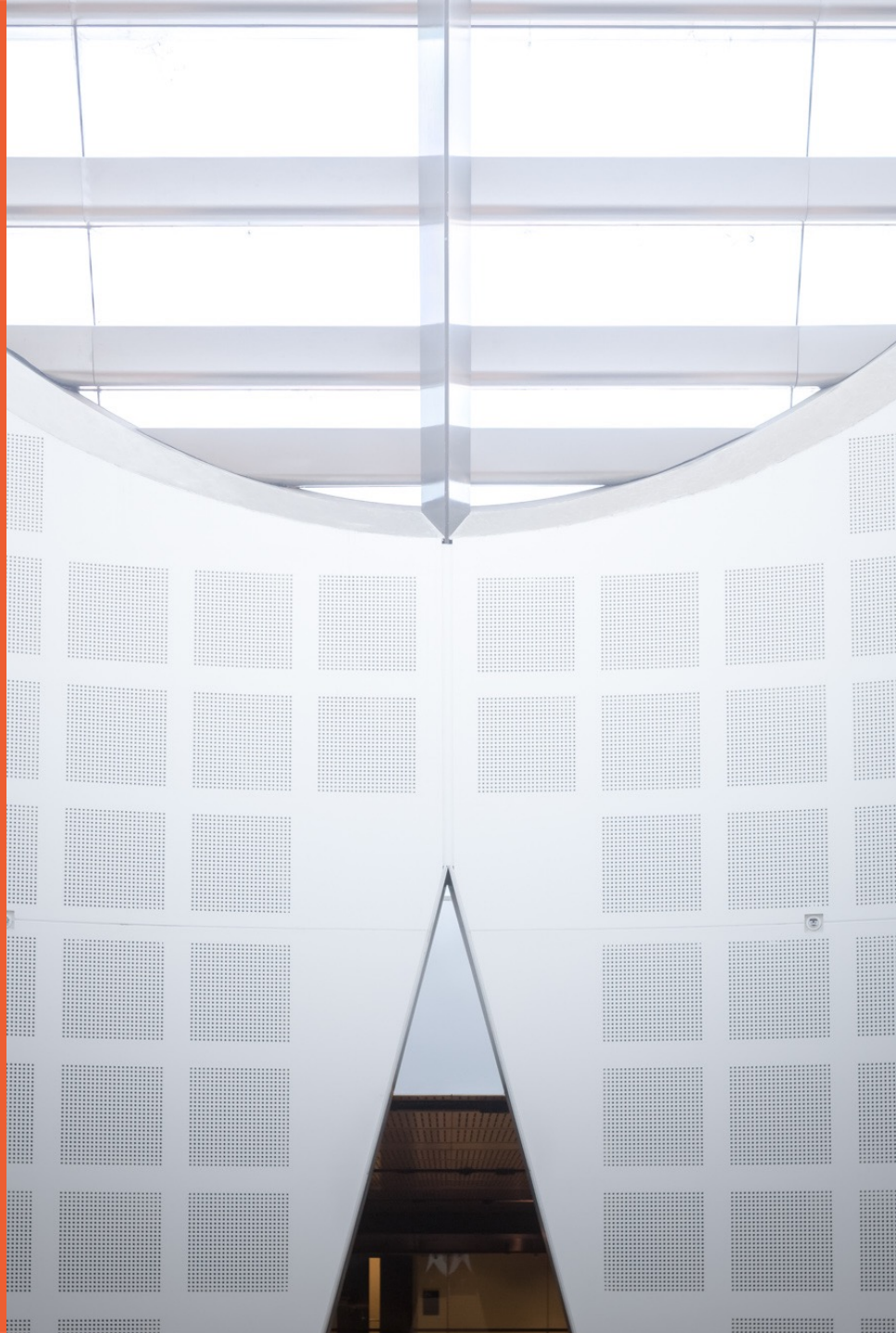
COMP9121: Design of Networks and Distributed Systems

Week 3: Link Layer 2

Wei Bao

School of Computer Science

THE UNIVERSITY OF SYDNEY



Last week

- Introduction and services
- Error detection and correction
- Multiple access protocols
  - Aloha, Slotted Aloha
  - CSMA/CD
- Link-layer Addressing

CSMA (carrier sense multiple access)

- *CSMA:* listen before transmit:

- If channel sensed idle: transmit entire frame
- If channel sensed busy, defer transmission

- Human analogy: don't interrupt others!
- *Three categories*

  - 1-persistent
  - non-persistent
  - p-persistent

CSMA (carrier sense multiple access)

- *1-persistent*
  - Sense the channel.
    - If busy, keep listening to the channel and transmit immediately when the channel becomes idle.
    - If idle, transmit a frame immediately.

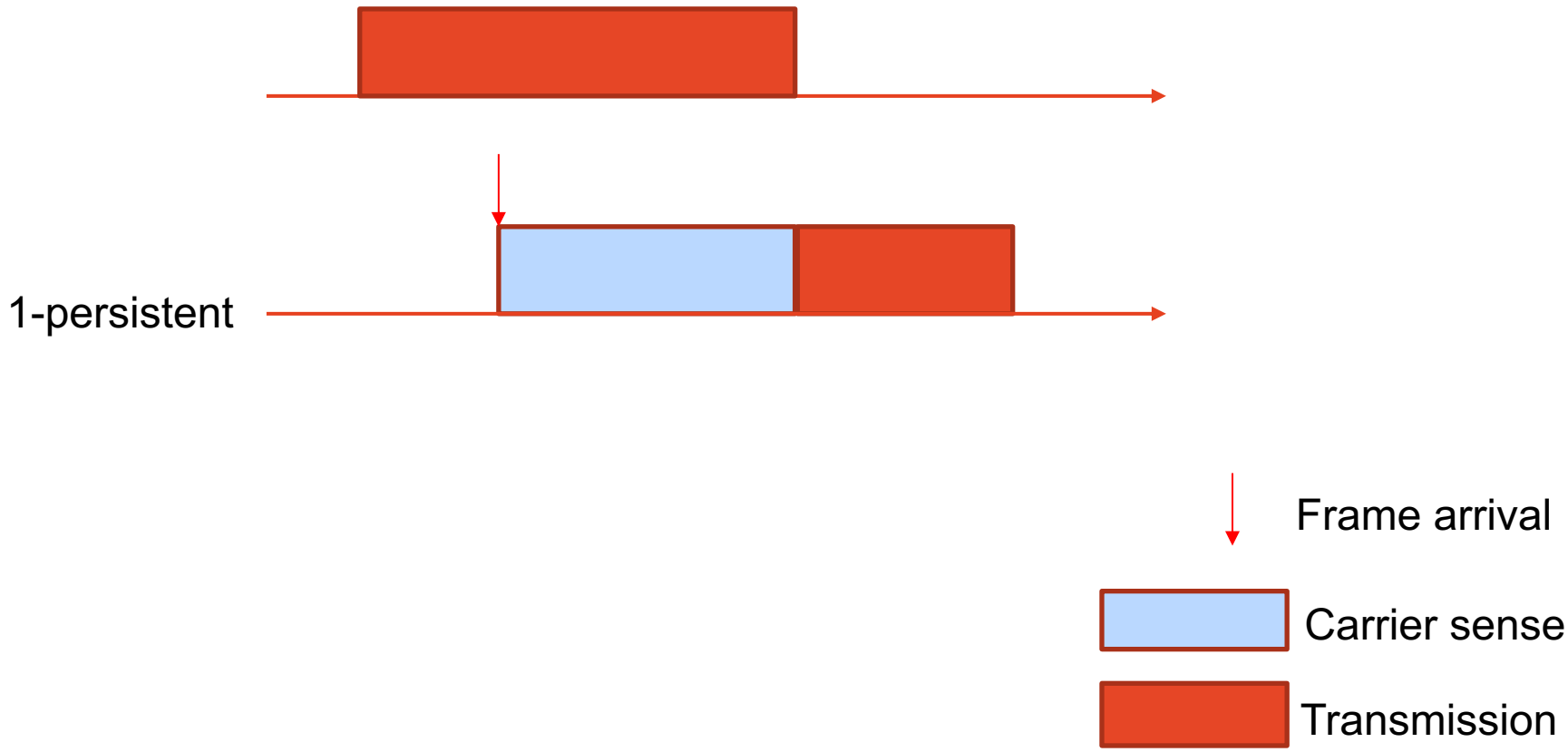
CSMA (carrier sense multiple access)

- *Non-persistent*
  - Sense the channel.
    - If busy, wait a random amount of time and sense the channel again
    - If idle, transmit a frame immediately

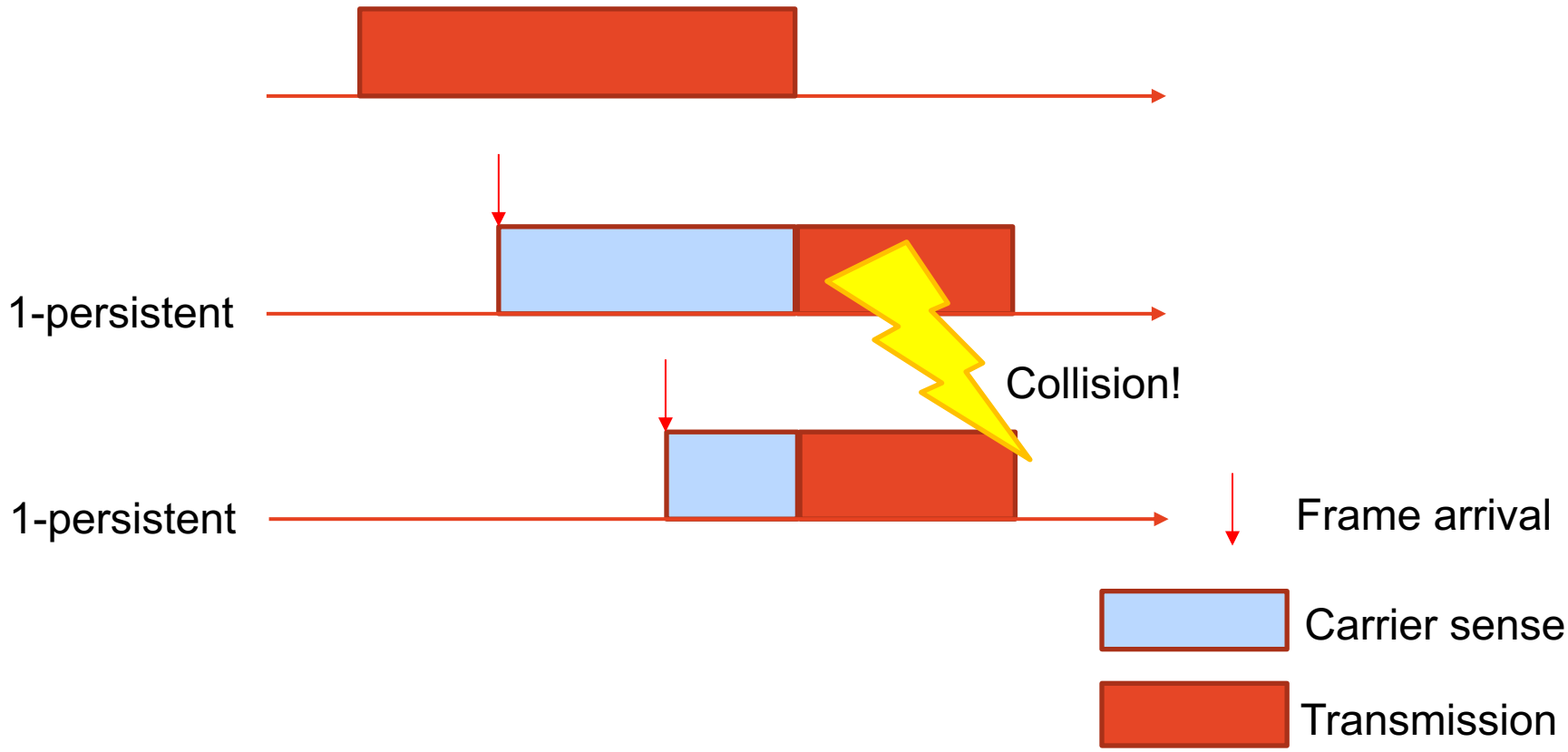
CSMA (carrier sense multiple access)

- *p-persistent*
  - Sense the channel (*)
    - If channel is busy, wait one slot and go to step (*)
    - If channel is idle, transmit a frame with probability p
      - if a frame is not transmitted (1-p), wait one slot and go to step (*)
  - one slot: one period of time, e.g., one round-trip propagation delay

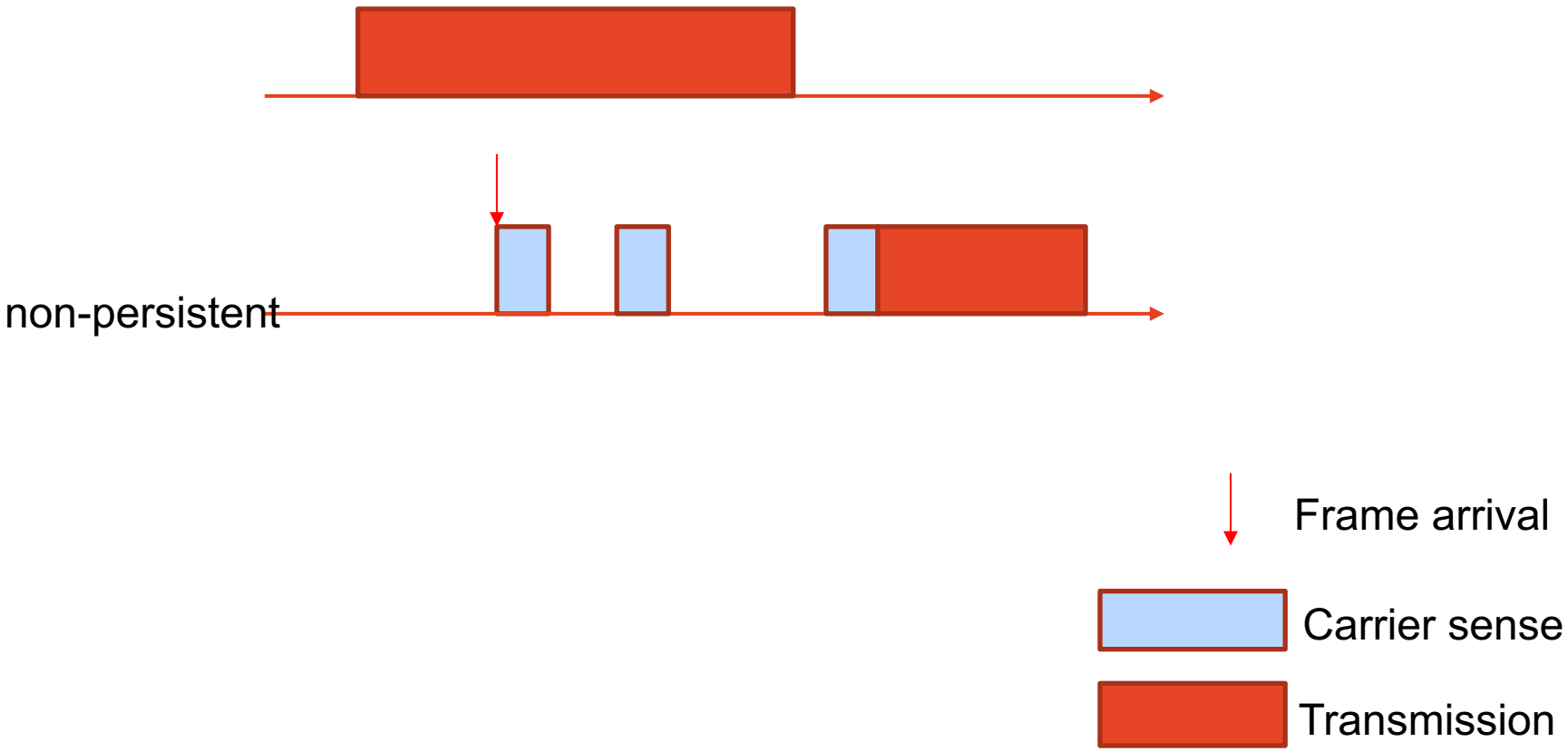
Channel sensed busy



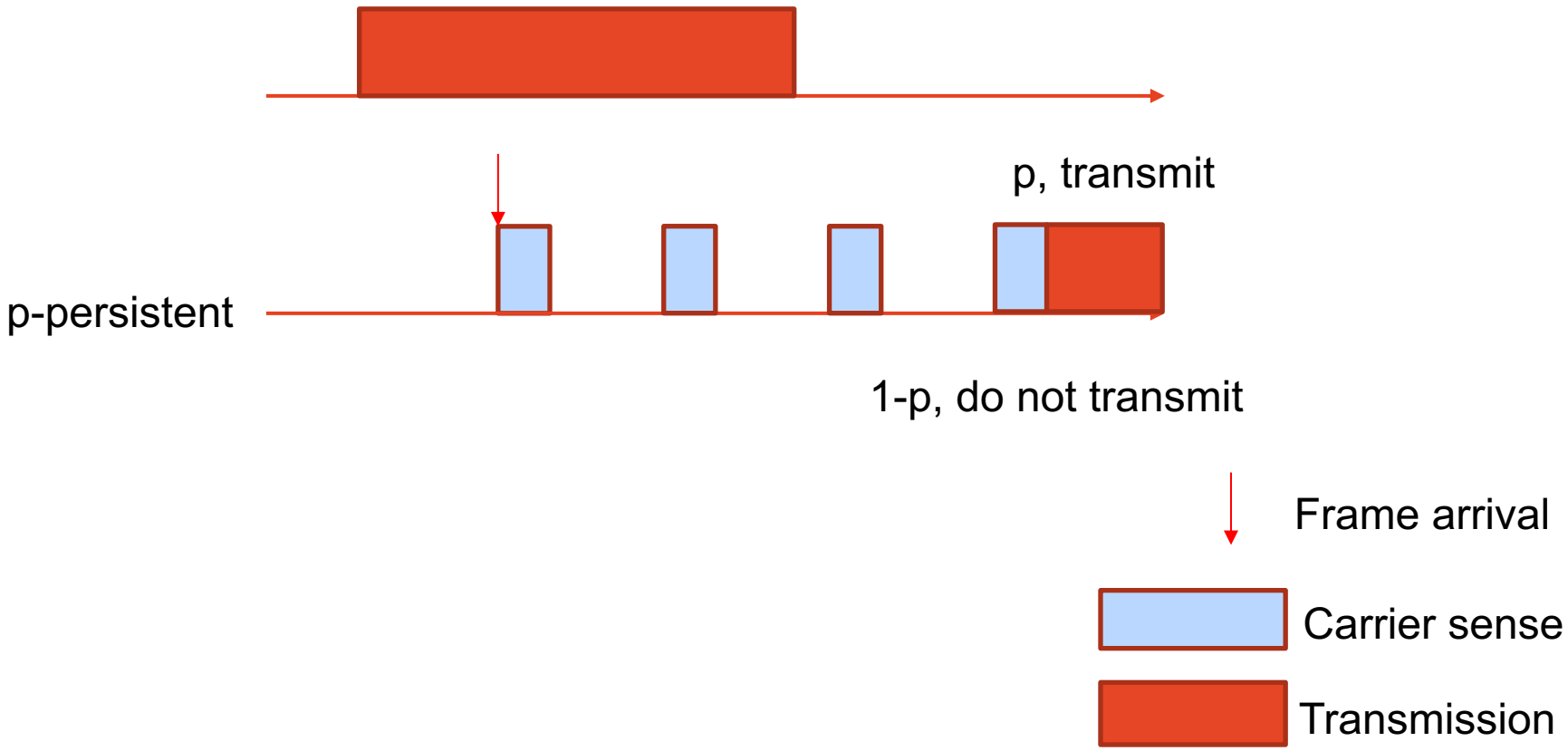
Channel sensed busy



Channel sensed busy

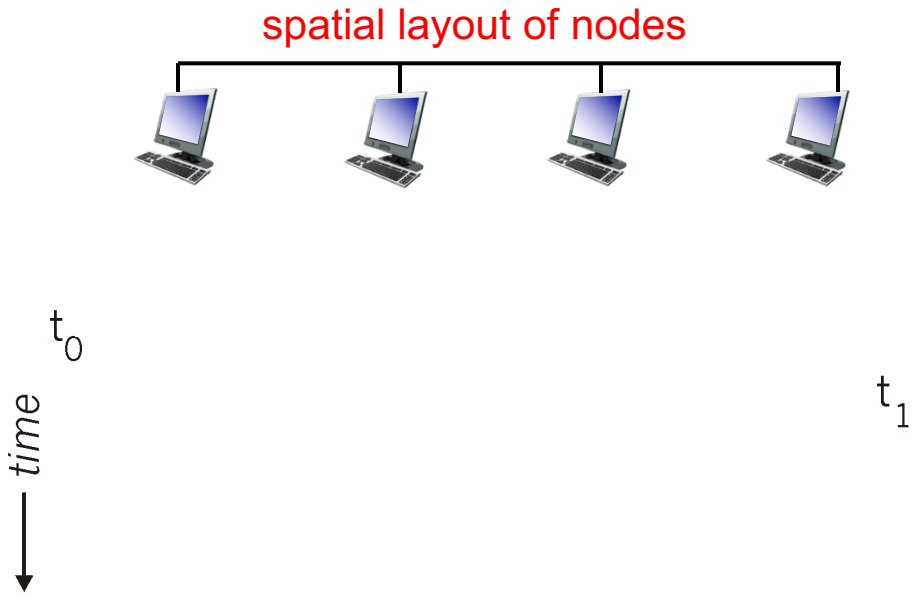


Channel sensed busy



CSMA collisions

- collisions *can* still occur: propagation delay means two nodes may not hear each other's transmission
- collision: entire frame transmission time wasted
- distance & propagation delay play role in determining collision probability
- Recovery: CSMA/CD, collision detection



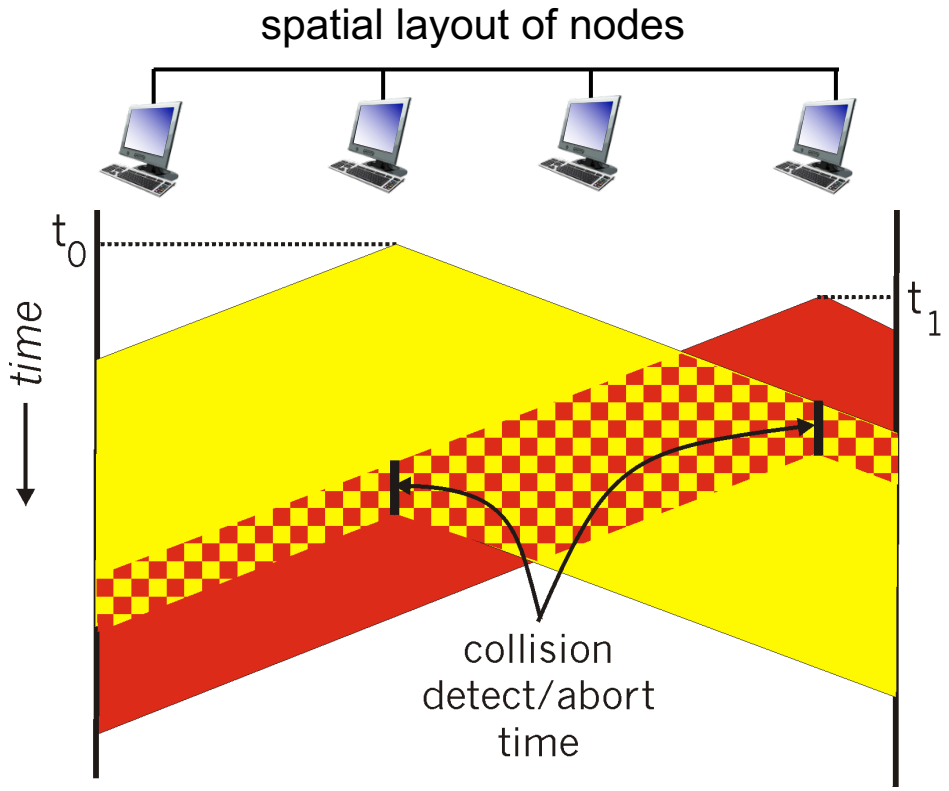
CSMA/CD (collision detection)

CSMA/CD:

- collisions *detected* within short time
- colliding transmissions aborted, reducing channel wastage

- collision detection:
  - wired LANs: measure signal strengths, compare transmitted, received signals
    - Can transmit and sense at the same time
  - wireless LANs: received signal strength overwhelmed by local transmission strength
    - CSMA-CD cannot be used in wireless LAN

CSMA/CD (collision detection)



CSMA/CD algorithm

1. NIC receives datagram from network layer, creates frame

2. If NIC senses channel idle, starts frame transmission. If NIC senses channel busy, waits until channel idle, then transmits. (1-persistent)

3. If NIC transmits entire frame without detecting another transmission, NIC is done with frame !

CSMA/CD algorithm

4. If NIC detects another transmission while transmitting, aborts and sends jam signal

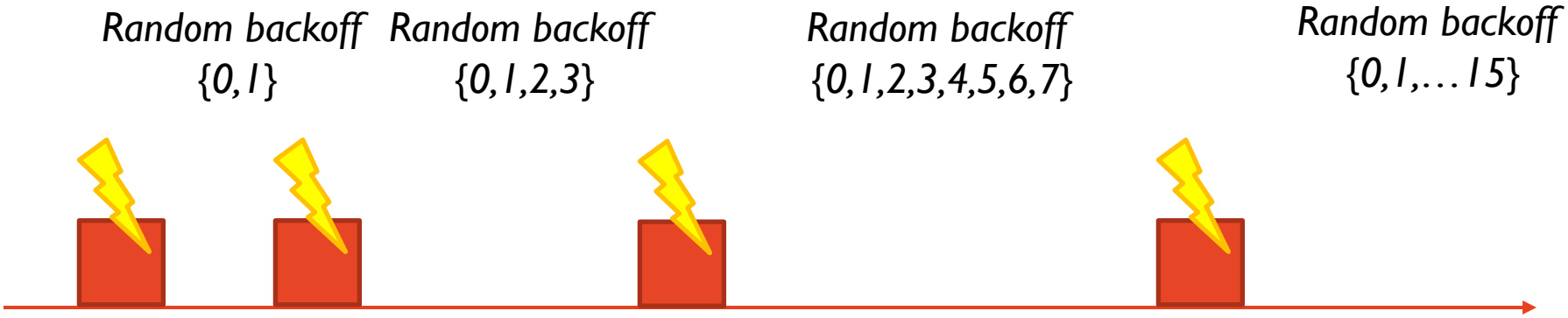
- make sure all other transmitters are aware of collision; 48 bits

5. After aborting, NIC enters *binary (exponential) backoff:*
   - after *m*th collision, NIC chooses *K* at random from $\{0,1,2,\ldots,2^m-1\}$. NIC waits K·512 bit times, returns to Step 2
   - longer backoff interval with more collisions
   - adapt retransmission attempts to estimated current load heavy load: random wait will be longer

   - bit time: time duration to push one bit in the wire
   - mth collision: number of collisions happened after last successful transmission.

Binary (exponential) backoff

- after *m*th collision, NIC chooses *K* at random from $\{0,1,2,\ldots,2^m-1\}$. NIC waits K·512 bit times



CSMA/CD efficiency

- t_{prop} = max prop delay between 2 nodes in LAN
- t_{trans} = time to transmit max-size frame

$$efficiency = \frac{1}{1 + 5t_{prop}/t_{trans}}$$

- efficiency goes to 1
  - as t_{prop} goes to 0
  - as t_{trans} goes to infinity
- better performance than ALOHA: and simple, cheap, decentralized!

Example

N computers have been connected to a shared medium (a cable) of length 2 km. The maximum channel rate of the shared medium is 10 Mbps. Each computer generates 10 frames per second with each frame being 1000 bytes. The speed of wave propagation is 2×10^8 meters/second. What is the maximum number of nodes supported in the network if CSMA-CD is used on the shared medium?

$$t_{prop} = \frac{2 \times 1000}{2 \times 10^8} = 10 \text{ microsec}$$

$$t_{trans} = \frac{1000 \times 8}{10^7} = 800 \text{ microsec}$$

$$efficiency = \frac{1}{1 + 5\frac{10}{800}} = 0.94$$

$$\text{traffic of each node} = 10 \times 1000 \times 8 = 80000 \text{bit/sec}$$

$$\text{no. nodes} = \frac{9.4 \times 10^6}{80000} = 117$$

CSMA/CD cannot be used in Wireless LAN

- wireless LANs: received signal strength overwhelmed by local transmission strength

  - Cannot detect collision at sender!

  - A and R can hear each other
  - B and R can hear each other
  - A and B can't hear each other
  - A doesn't know it collides with B

  - Solution: carrier sense multiple access with collision avoidance (CSMA/CA)

A

R

B

Important Concepts

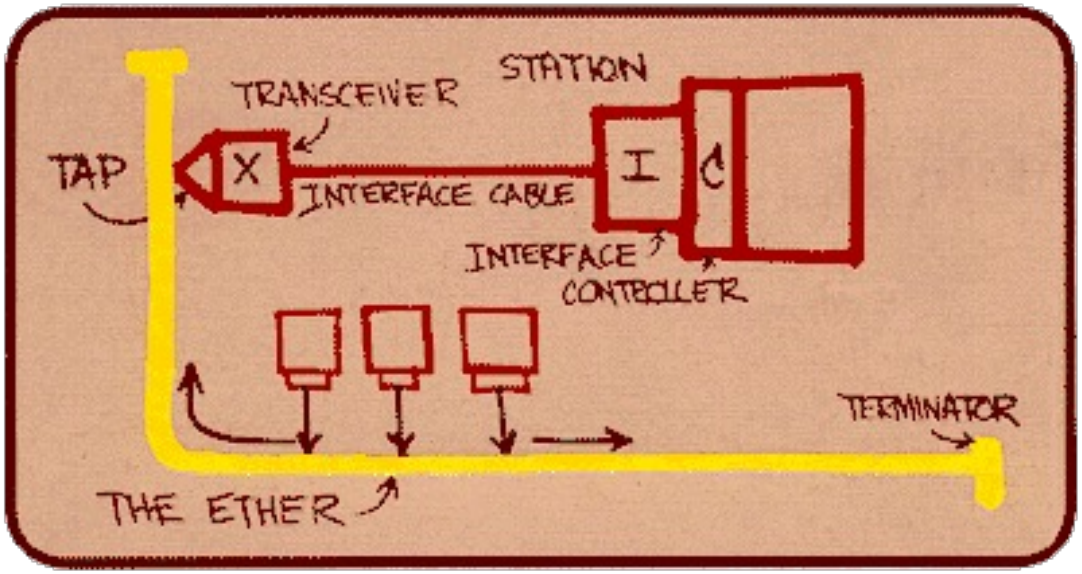
- CSMA is used to enhance efficiency
  - Sense before transmission
  - If an on-going communication, wait

- CSMA-CD is used in wired medium

Ethernet

"dominant" wired LAN technology:

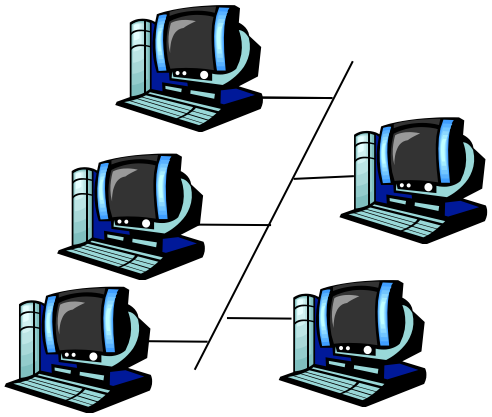
- cheap for NIC
- kept up with speed race: 10 Mbps – 10 Gbps



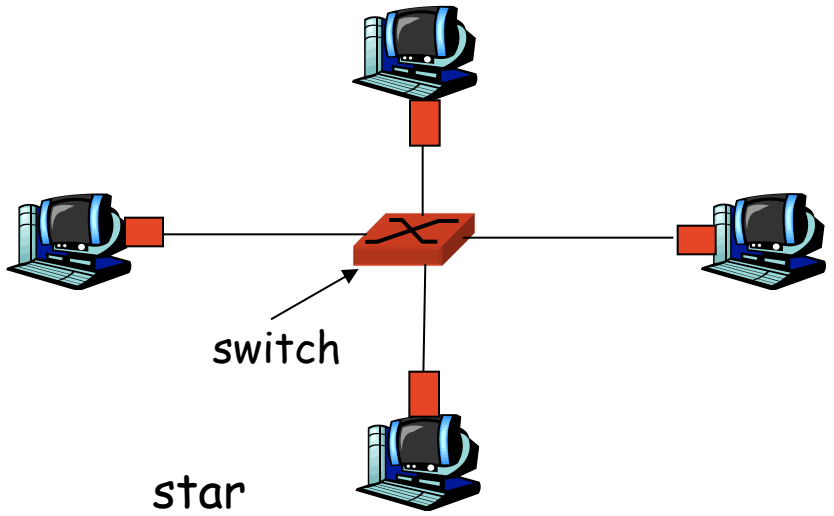
Metcalfe's Ethernet sketch

Topology

- bus topology popular through mid 90s
  - all nodes in same collision domain (can collide with each other)
- today: star topology prevails
  - active *switch* in center
  - nodes do not collide with each other – more on this shortly



bus: coaxial cable



star

Ethernet: Unreliable, connectionless

- connectionless: No handshaking between sending and receiving NICs
- unreliable: receiving NIC doesn't send acks or nacks to sending NIC
  - stream of datagrams passed to network layer can have gaps (missing datagrams)
  - gaps will be filled if app is using TCP
  - otherwise, app will see gaps
- Ethernet's MAC protocol: CSMA/CD

- Ethernet: Detect error, discard but do not recover. The gap will be filled by TCP.
- Very small bit error probability in wired network

MAC Addresses

- 32-bit IP address:
  - *network-layer* address
  - used to get datagram to destination in network layer
  - e.g. 129.56.78.123
- MAC (or LAN or physical or Ethernet) address:
  - function: *get frame from one interface to another physically-connected interface*
  - 48 bit MAC address (for most LANs)
    - burned in NIC ROM, also sometimes software settable

MAC Addresses

- MAC address name space is managed by IEEE.
- MAC-48 addresses in human-friendly form is six groups of two hexadecimal digits, separated by hyphens (-) or colons (:), in transmission order (e.g. 01-23-45-67-89-AB or 01:23:45:67:89:AB)

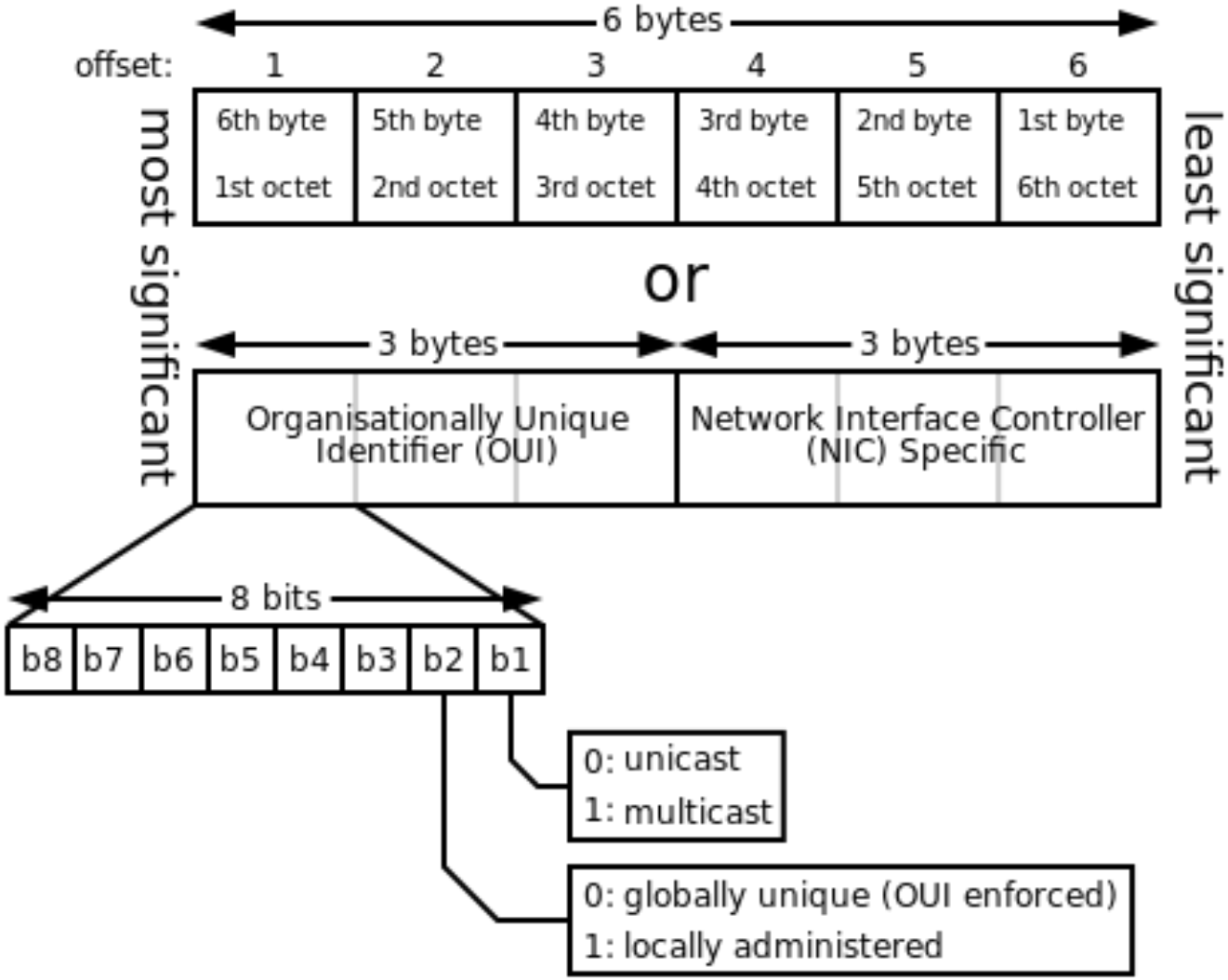
Universal and Local Addresses

- Addresses can either be **universally administered addresses** or **locally administered addresses.**
- A universally administered address is uniquely assigned to a device by its manufacturer, and burned into the device.
- A locally administered address is assigned to a device by a network administrator, overriding the burned-in address.

MAC Address Types

- Unicast
- Broadcast
  - FF:FF:FF:FF:FF
- Multicast
  - 01-00-5E-xx-xx-xx
  - 33-33-xx-xx-xx-xx
- If the addresses match, the frame is processed, otherwise it is discarded.

MAC Address Format



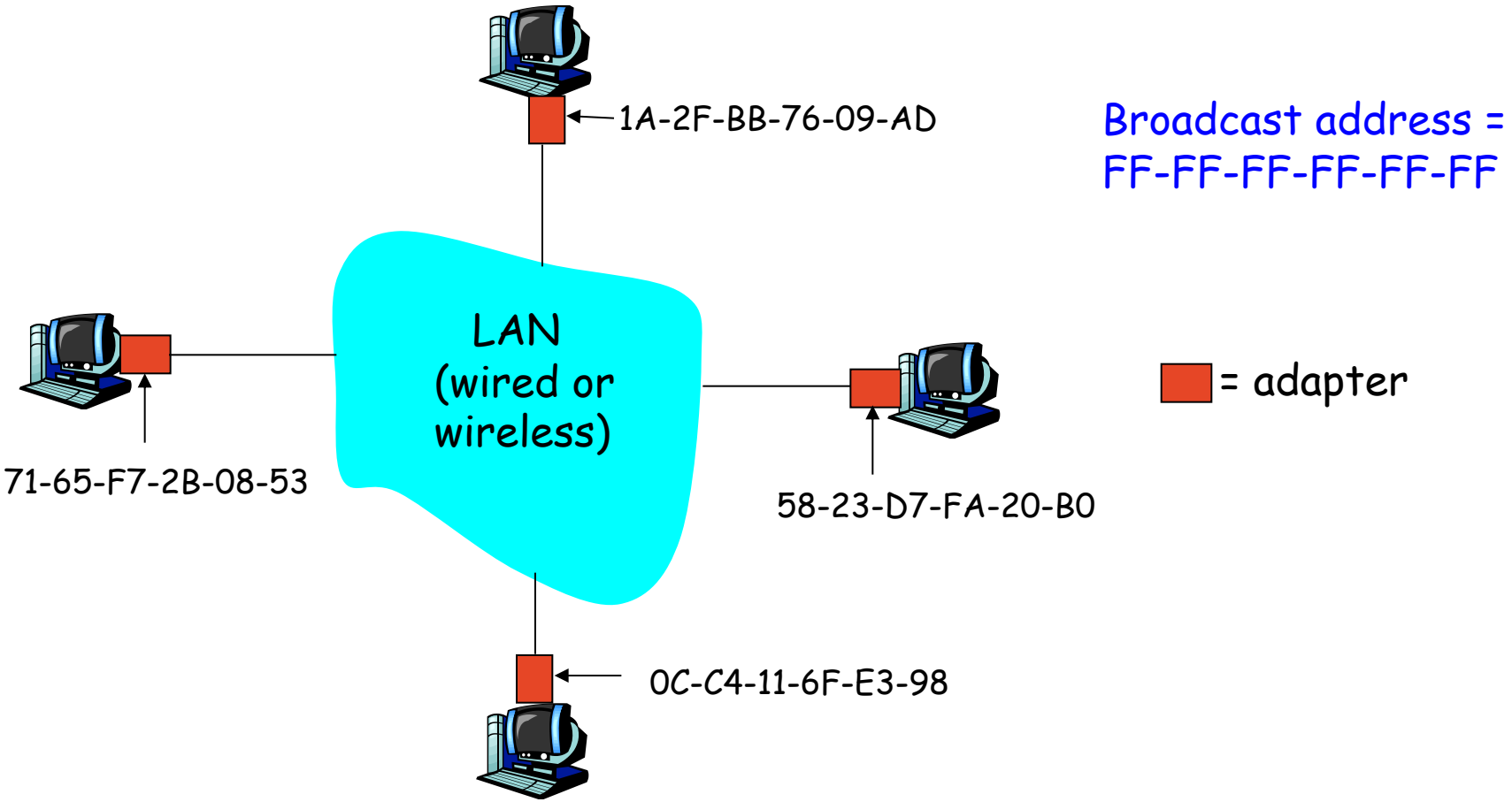
Check the manufacturer of NIC

https://dnschecker.org/mac-lookup.php

Windows: ipconfig /all
Apple: ifconfig

LAN Addresses

Each adapter on LAN has a unique LAN address



LAN Address

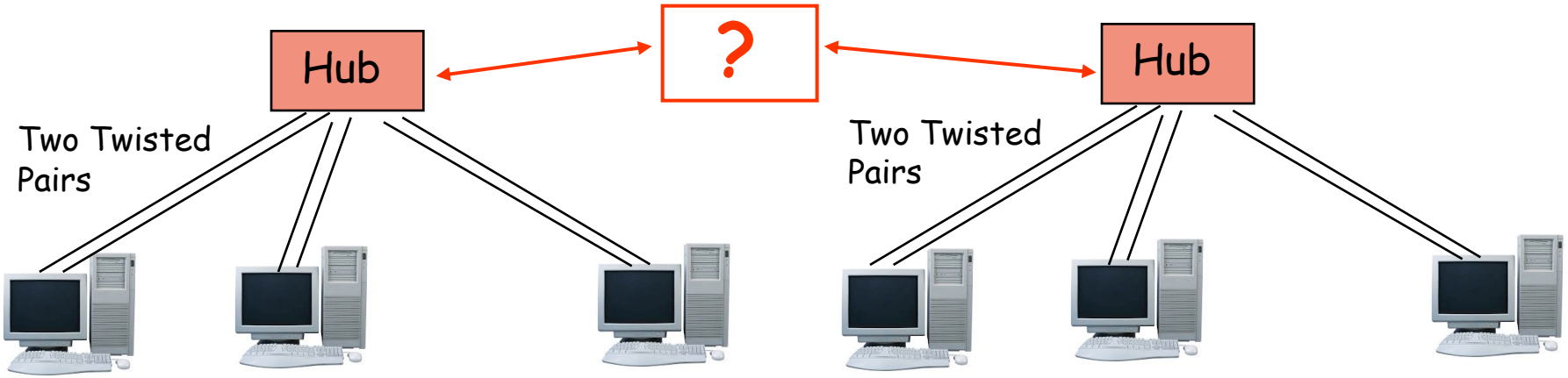
- manufacturer buys portion of MAC address space (to assure uniqueness)
- MAC address is portable
  - can move LAN card from one LAN to another
- IP address NOT portable

Hubs and Switches

Hubs

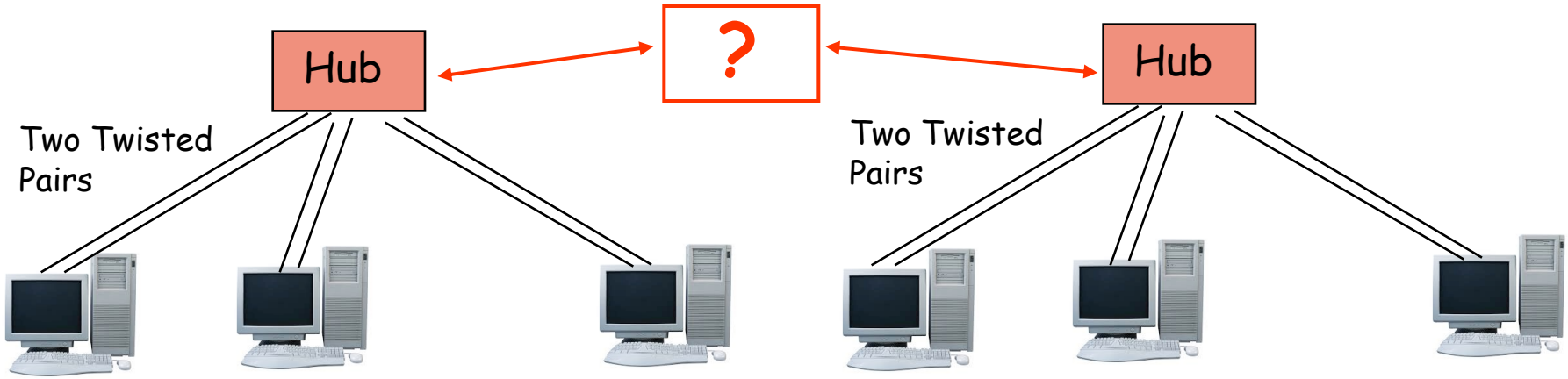
- Hub: Central element to connect computers
  - Simple repeater in Ethernet LANs
  - Equivalent to a bus

User community grows, need to interconnect hubs



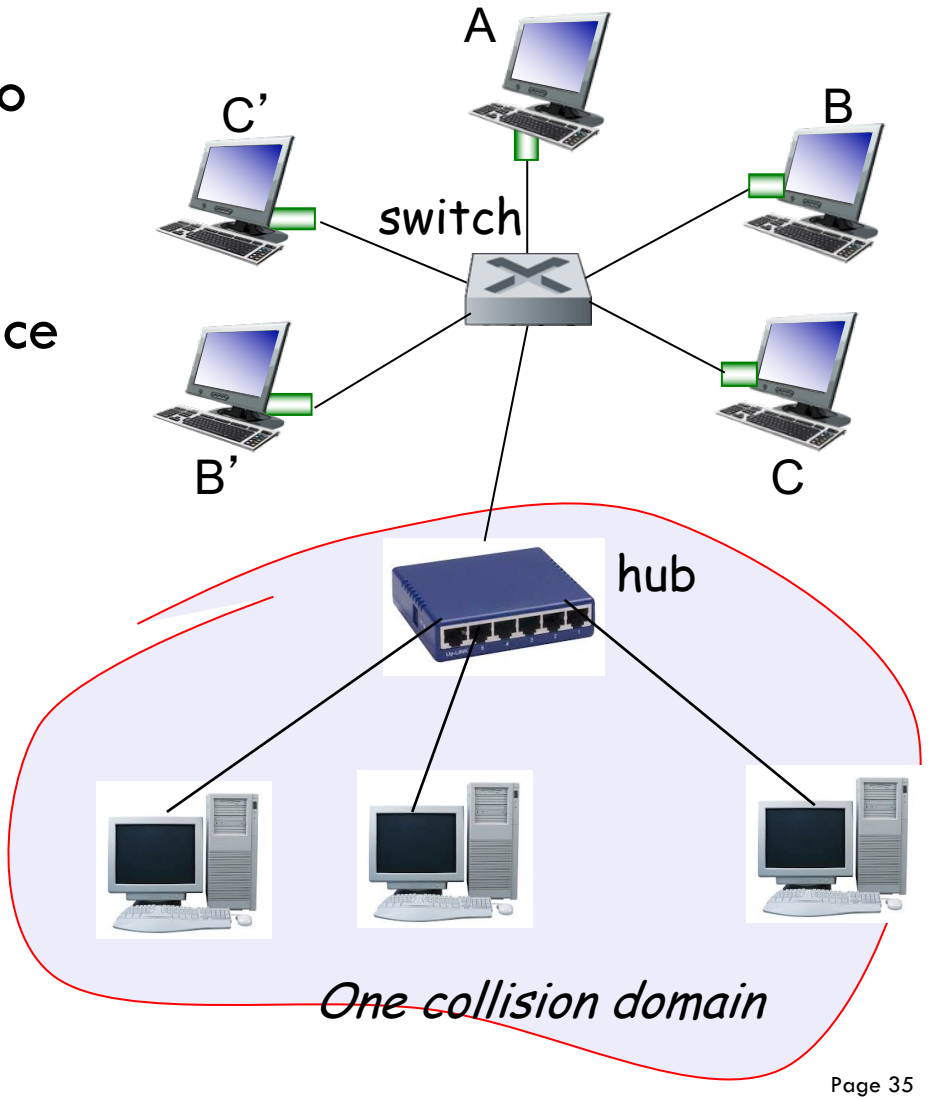
Hubs and Switches

- Interconnecting Hubs
  - **Hub** (Repeater): Signal regeneration
    - All traffic appears in both sides
  - **Switch** (Bridge): MAC address filtering
    - Local traffic stays in own side



LAN Switch (Bridge)

- Maybe connected directly to users or hubs
- Hub is a physical layer device
- Switch is a link layer device

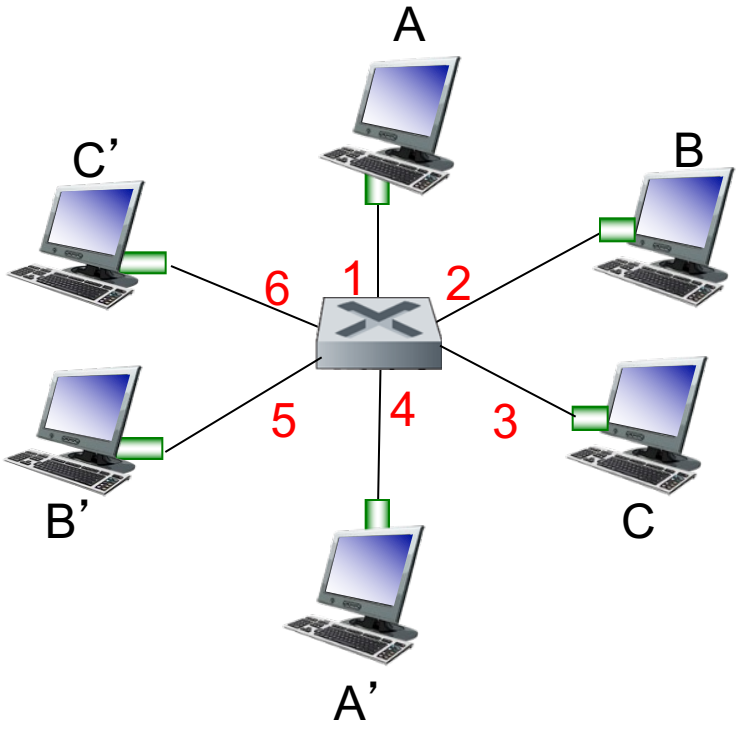


Ethernet switch

- link-layer device: takes an *active* role
  - store, forward Ethernet frames
  - examine incoming frame's MAC address, selectively forward frame to one-or-more outgoing links
  - use CSMA/CD at each interface

Switch: multiple simultaneous transmissions

- hosts have dedicated, direct connection to switch.
- switches buffer frames.
- each link has its own collision domain.
- *switching:* A-to-A' and B-to-B' can transmit simultaneously, without collisions



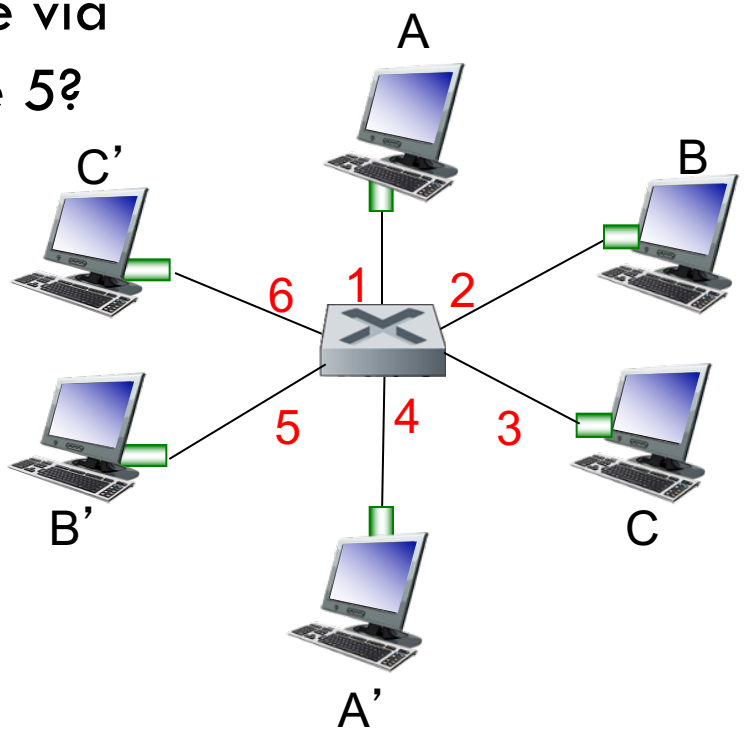
switch with six interfaces (1,2,3,4,5,6)

Switch forwarding table

Q: how does switch know A' reachable via interface 4, B' reachable via interface 5?

A: each switch has a switch table, each entry:

- (MAC address of host, interface to reach host, time stamp)



switch with six interfaces (1,2,3,4,5,6)

Q: how are entries created, maintained in switch table?

A: Self-learn and flood

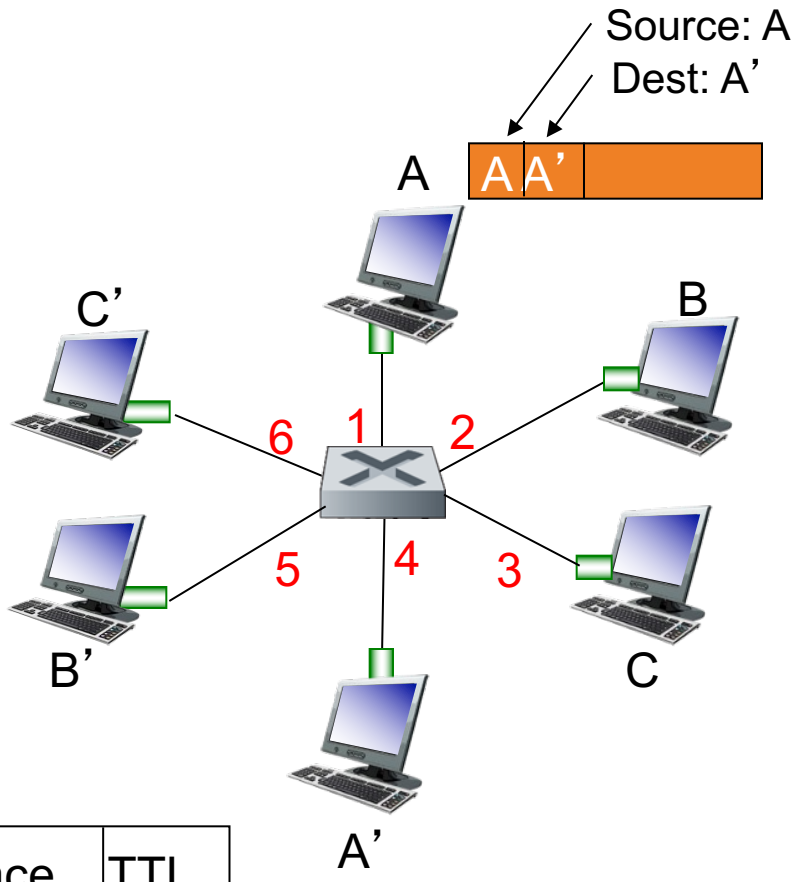
Transparent Switches

- *Transparent*
  - hosts are unaware of presence of switches

- *plug-and-play, self-learning*
  - switches do not need to be configured

Switch: self-learning

- switch *learns* which hosts can be reached through which interfaces
  - when frame received, switch "learns" location of sender: incoming LAN frame
  - records sender/location pair in switch table



MAC addr	interface	TTL
A	*1*	*60*

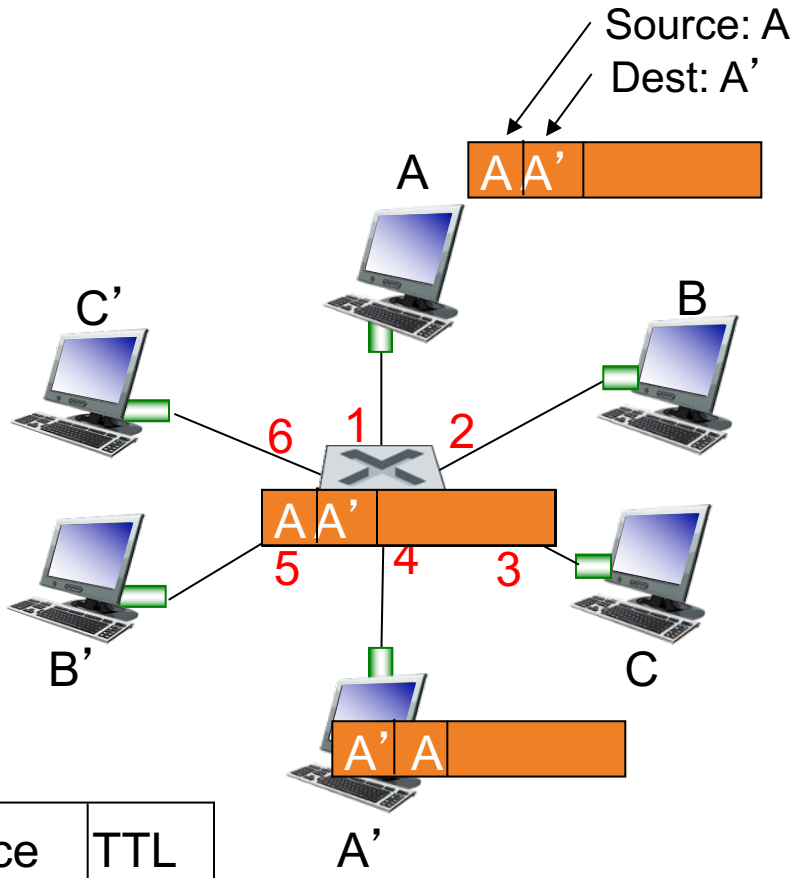
Switch table (initially empty)

Self-learning, forwarding: example

- frame destination, A', location unknown:

flood

- destination A location known: selectively send on just one link

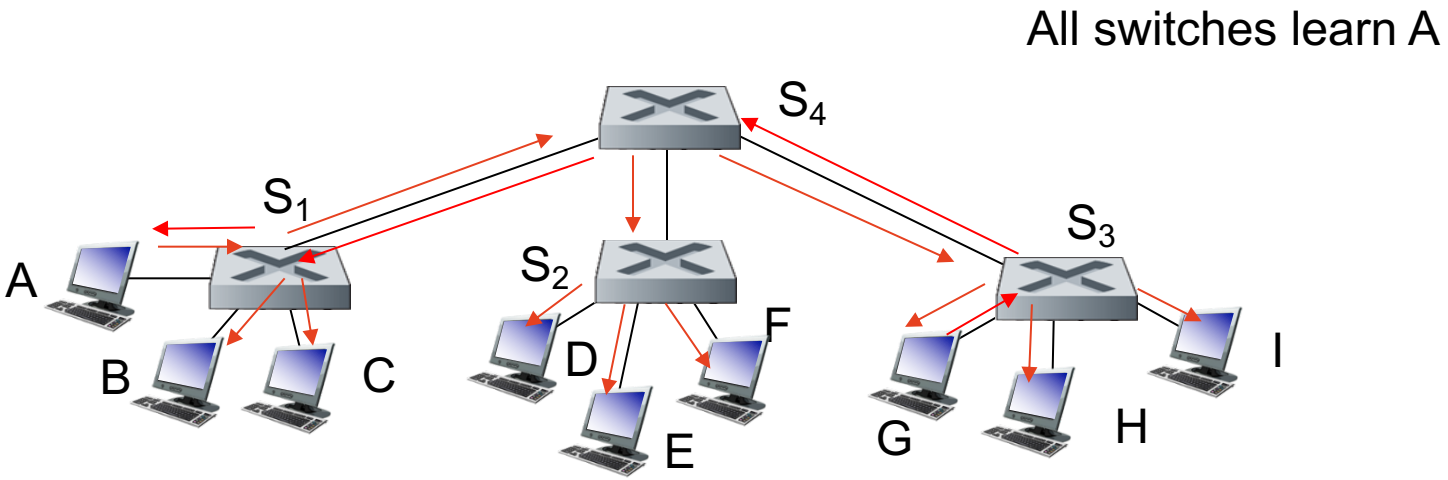


MAC addr	interface	TTL
A	*1*	*60*
A'	*4*	*60*

switch table (initially empty)

Interconnecting switches

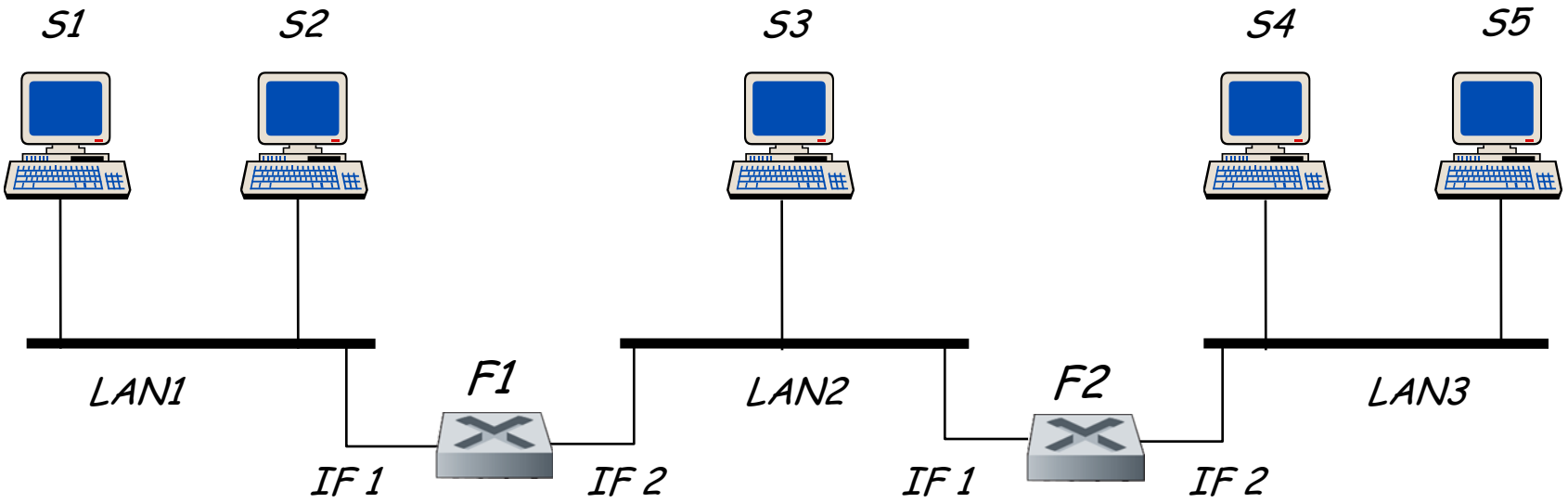
- switches can be connected together



Q: sending from A to G - how does S_1 know to forward frame destined to G via S_4 and S_3?

- works exactly the same as in single-switch case!

Example

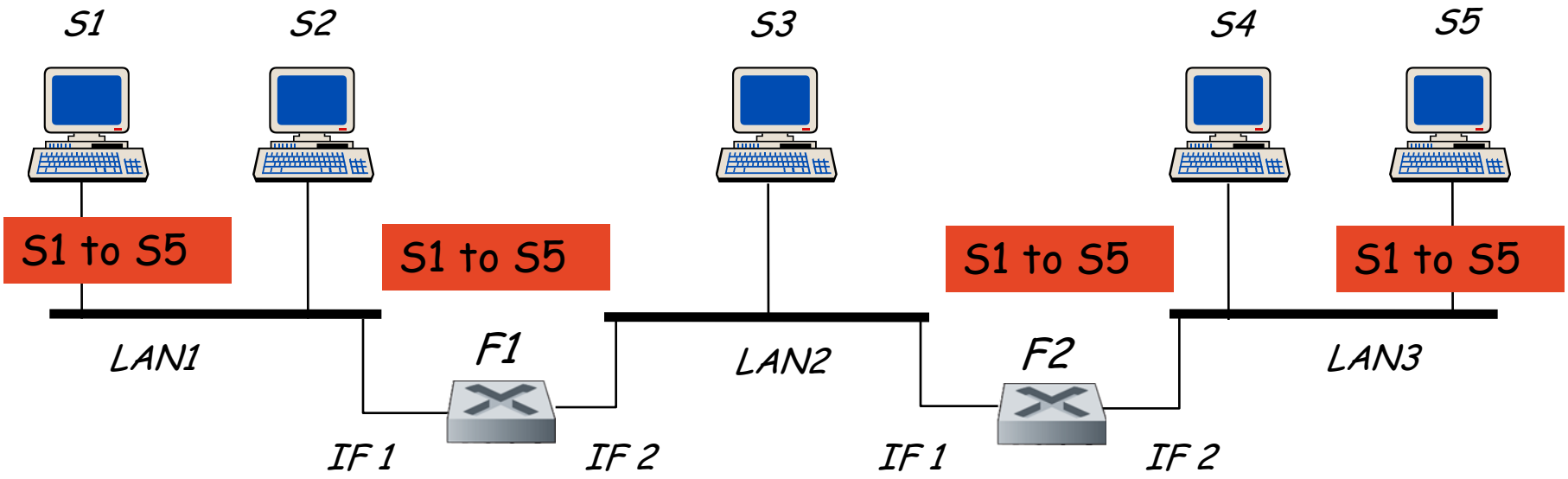


Address	IF

Address	IF

Source: Alberto Leon-Garcia, Indra Widjaja Communication Networks, 2ed. McGraw-Hill, 2003

S1→S5

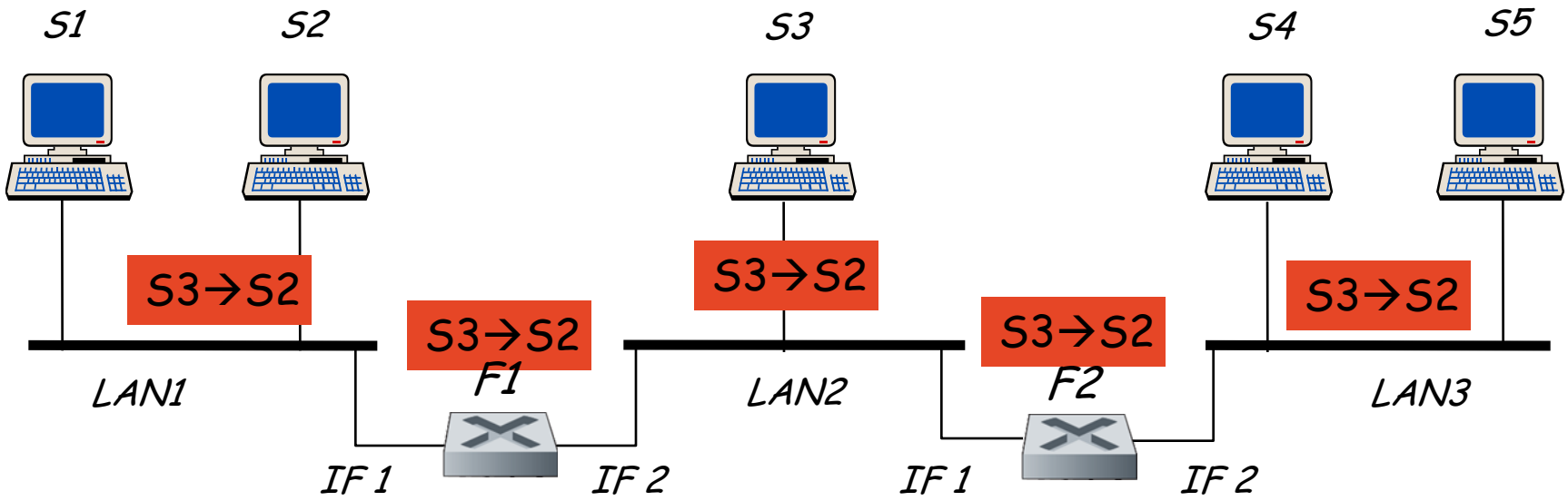


Address	IF
S1	1

Address	IF
S1	1

Source: LGW

S3→S2

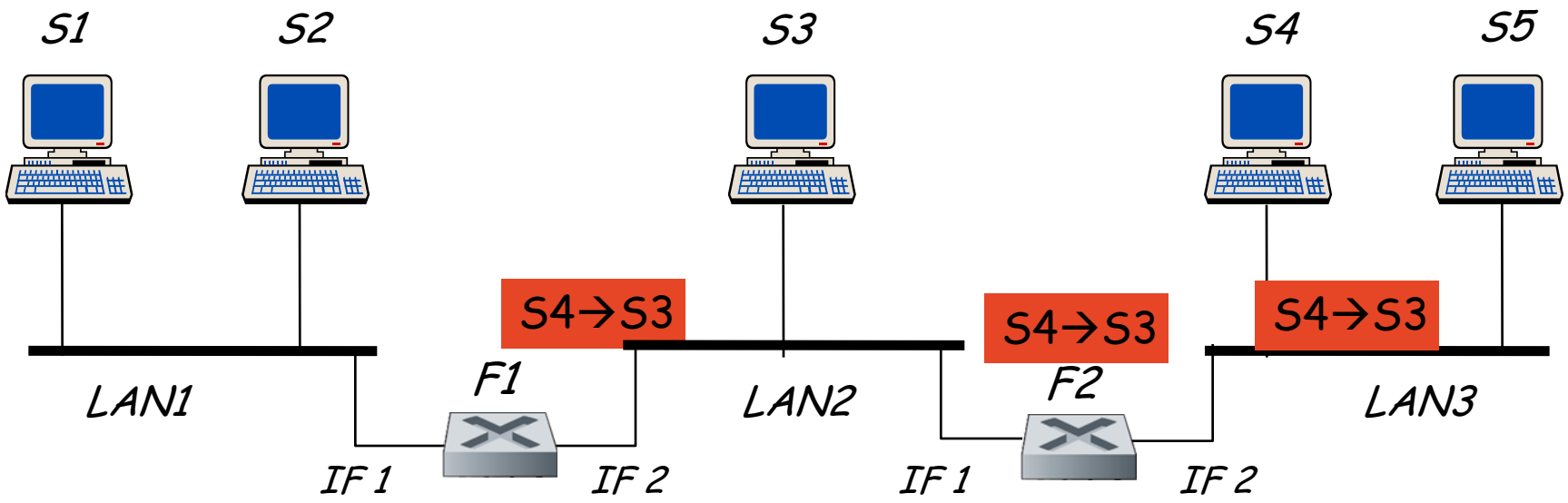


Address	IF
S1	1
S3	2

Address	IF
S1	1
S3	1

Source: LGW

S4→S3

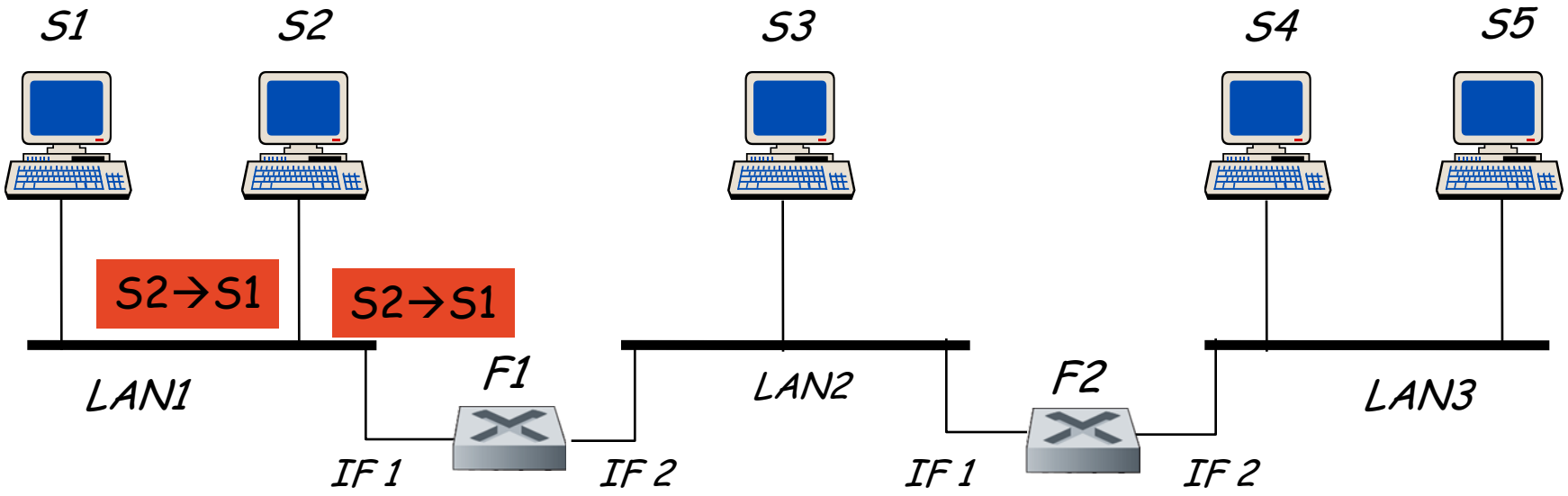


Address	IF
S1	1
S3	2
S4	2

Address	IF
S1	1
S3	1
S4	2

Source: LGW

S2→S1

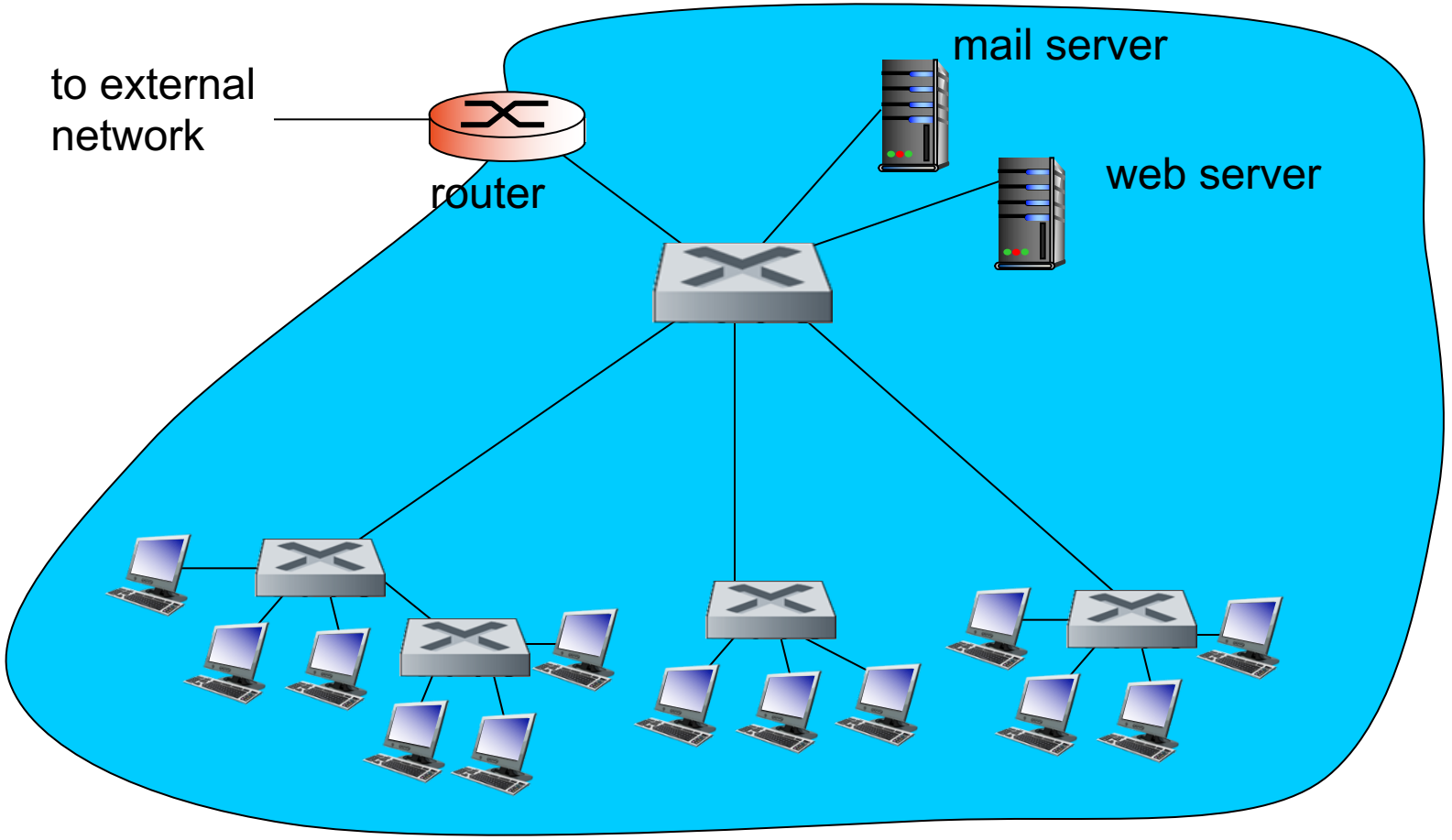


Address	IF
S1	1
S3	2
S4	2
S2	1

Address	IF
S1	1
S3	1
S4	2

Source: LGW

Institutional network



Python

Python

```
x = 34 - 23                                  # A comment
y = "Hello"                                  # Another comment
z =3.45
if z== 3.45 or y =="Hello":
    x = x + 1
    y = y + " World"                         # String concat.

print (x)
print (y)
```

Basic Datatypes

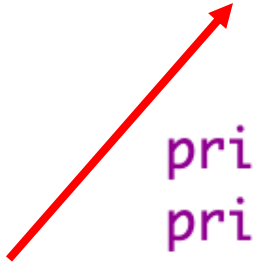
- Integers (default for numbers)
- z = 5
- Floats
- x = 3.456
- Strings
- Can use “” or ‘’ to specify.
- “abc” ‘abc’ (Same thing.)

Indentation

- Indentation is meaningful in Python.
- The first line with less indentation is outside of the block.
- The first line with more indentation starts a nested block
- Often a colon appears at the start of a new block. (E.g. for function and class definitions.)

Indentation

```
x = 34 - 23                                  # A comment
y = "Hello"                                  # Another comment
z =3.45
if z== 3.45 or y =="Hello":
    x = x + 1
    y = y + " World"                         # String concat.

print (x)
print (y)
```



Indentation

colon

Comments

- Start comments with #–the rest of line is ignored.

```
x = 34 - 23                                  # A comment
y = "Hello"                                  # Another comment
z =3.45
if z== 3.45 or y =="Hello":
    x = x + 1
    y = y + " World"                         # String concat.

print (x)
print (y)
```

Assignment

- A variable is created the first time it appears on the left side of an assignment expression:
- x= 3
- x, y=2,3

List

- List
- Ordered sequence of items of mixed types
- Lists are defined using square brackets (and commas).
- >>>li= [“abc”, 34, 4.34, 23]

- li[0]=“abc”
- li[1]=34
- li[-1]=23
- len(li)=4

List

- https://www.tutorialspoint.com/python3/python_lists.htm

- More operations, e.g., append, delete, insert,

List

- List copy
- >>>list2 = list1
- #wrong
- # 2 names refer to 1 list
- # Changing one affects both
- >>>list2 = list1[:]
- # Two independent copies, two lists
- >>> import copy
- >>> list2=copy.copy(list1)

Boolean Expressions

- True and False
- Comparison operators: ==, !=, <, <=, etc.
- X == Y
- X and Y have same value (like Java equals method)

Control Flow

- If statement

```
if x == 3:
    print ("X equals 3.")
elif x == 2:
    print ("X equals 2.")
else:
    print ("X equals something else.")
print ("This is outside the 'if'.")
```

Control Flow

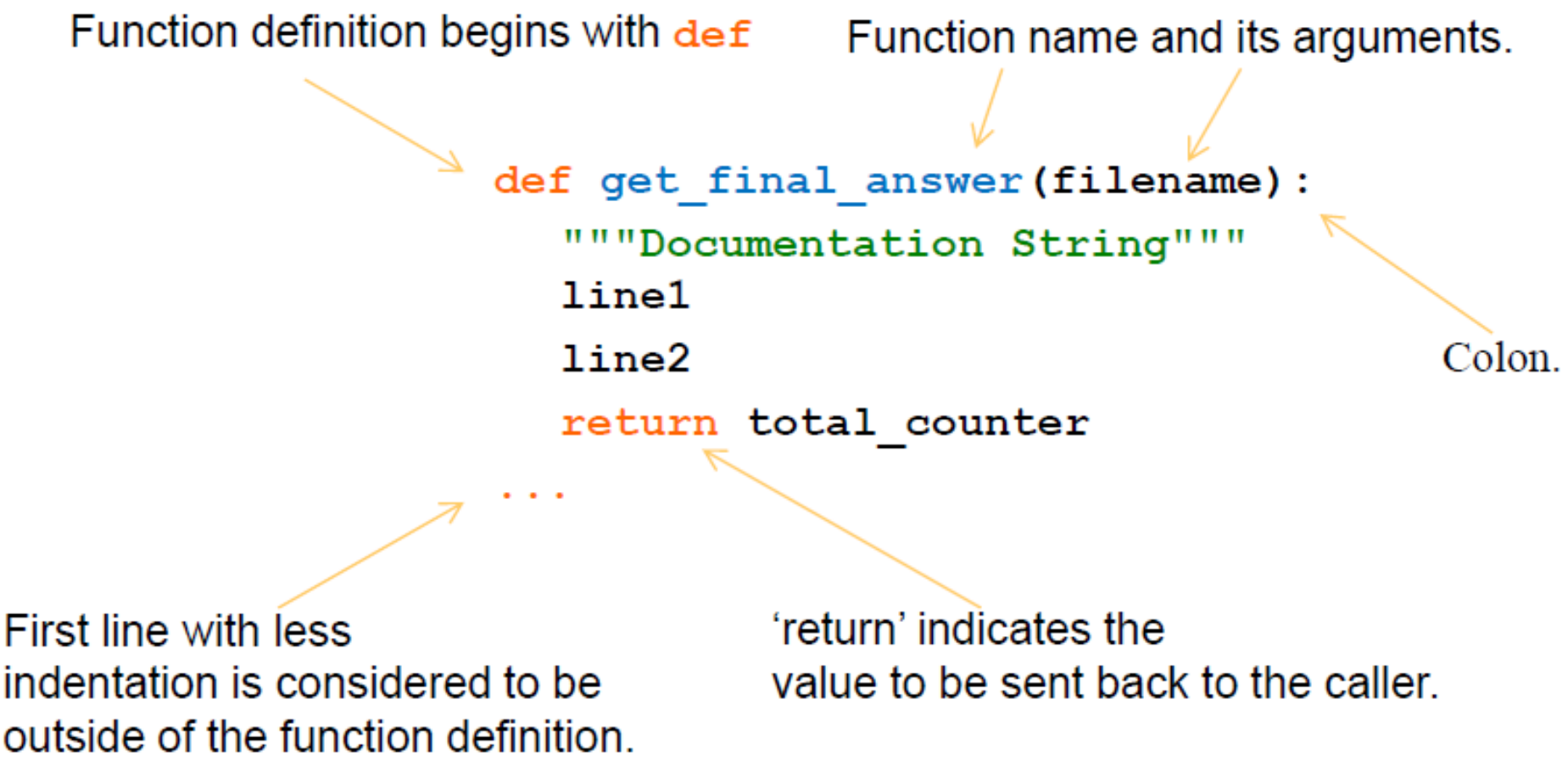
- while loop

```
>>> x = 3
>>> while x < 5:
    print (x, "still in the loop")
    x = x + 1


3 still in the loop
4 still in the loop
>>> x = 6
>>> while x < 5:
    print (x, "still in the loop")
    x = x + 1


>>>
```

Function

```
>>> def myfun(x, y):
        return x * y
>>> myfun(3, 4)
12
```

Function


Function definition begins with def
Function name and its arguments.
def get_final_answer(filename):
"""Documentation String"""
line1
line2
return total_counter
...
Colon.
First line with less indentation is considered to be outside of the function definition.
'return' indicates the value to be sent back to the caller.

More

https://docs.python.org/3.12/tutorial/