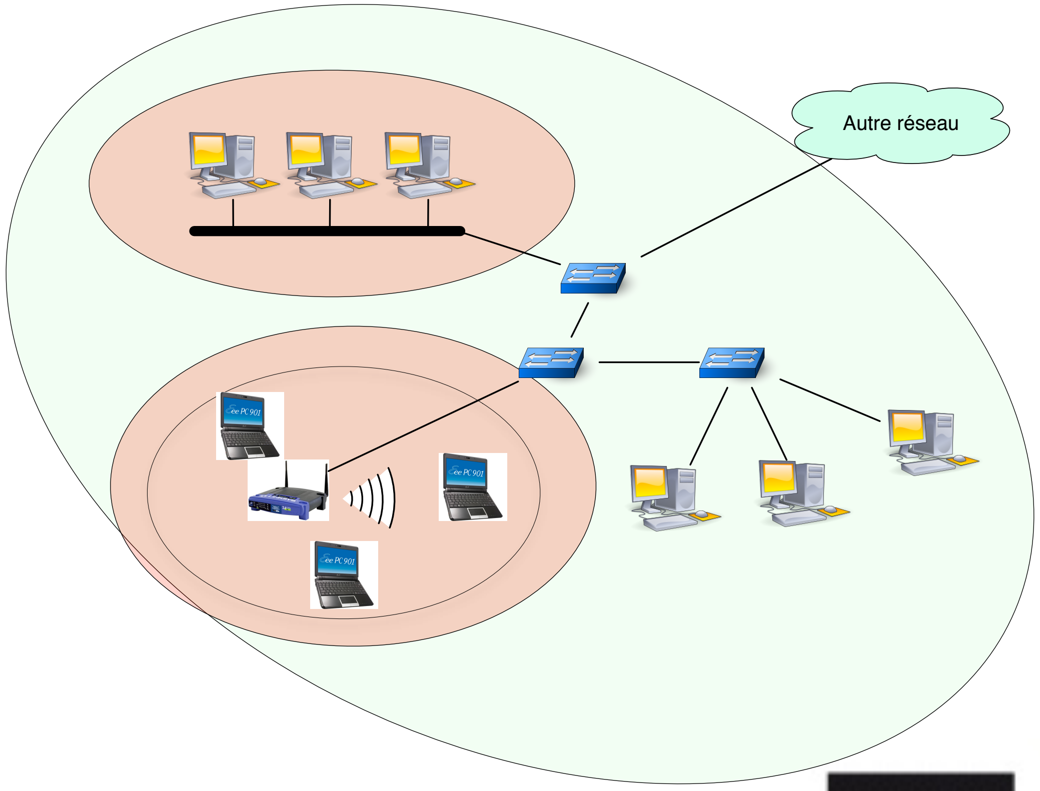
Medium Access Control

Claude Chaudet

Collision vs. broadcast domains

- **Interconnection is done by inserting active elements in the network**
  - Increase maximum distance between two hosts
  - Ease cabling
  - Separate the network in independent zones

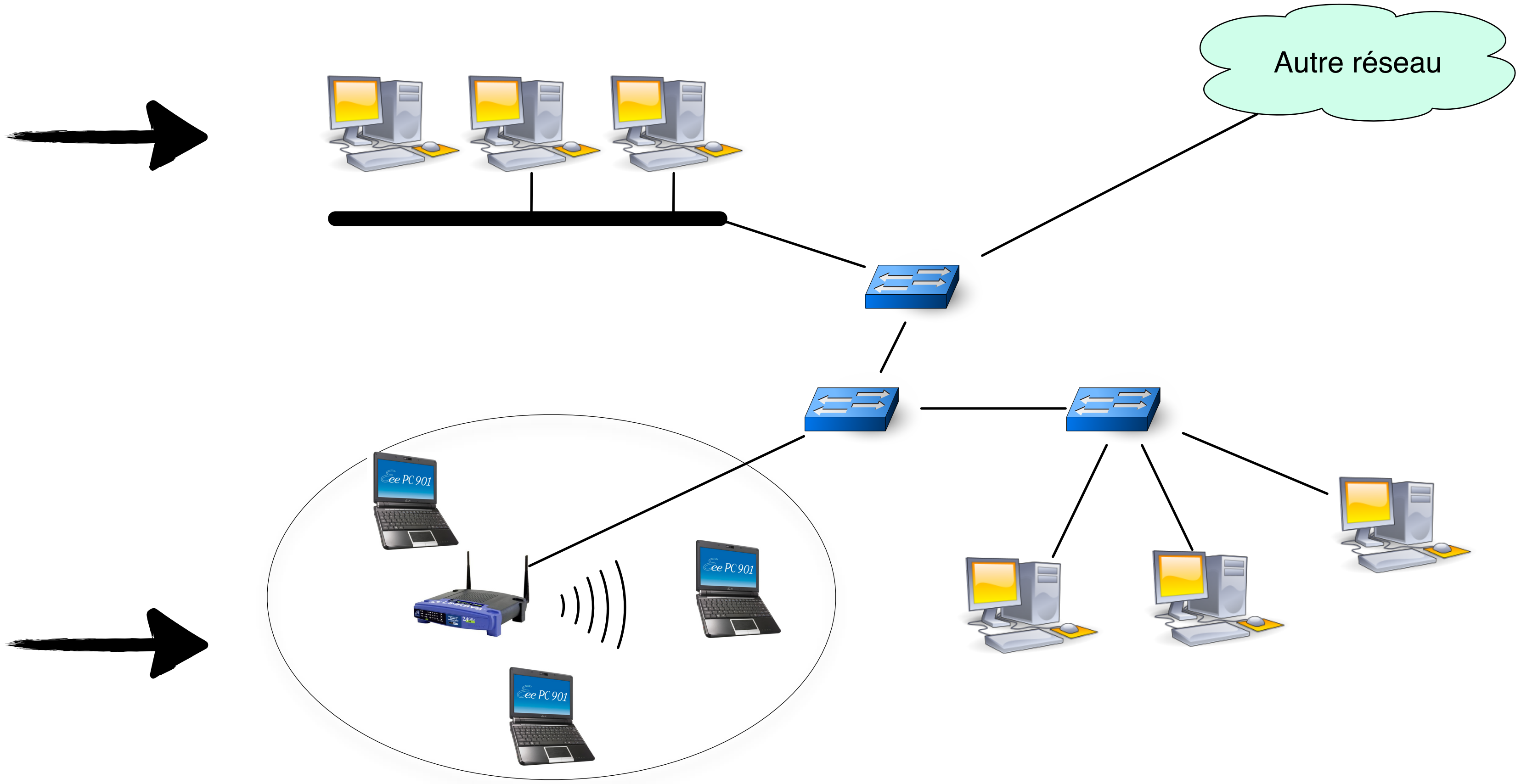
- **These devices define:**
  - Collision domains 🟥
    - Terminals need to organize to share a channel otherwise collisions will happen
  - Broadcast domains 🟩
    - Terminals belong to the same "group" and are close to each other



MAC layer role

- **Organize concurrent transmissions on a shared link**
  - Prevent collisions
  - Allow a fair access to the shared resource

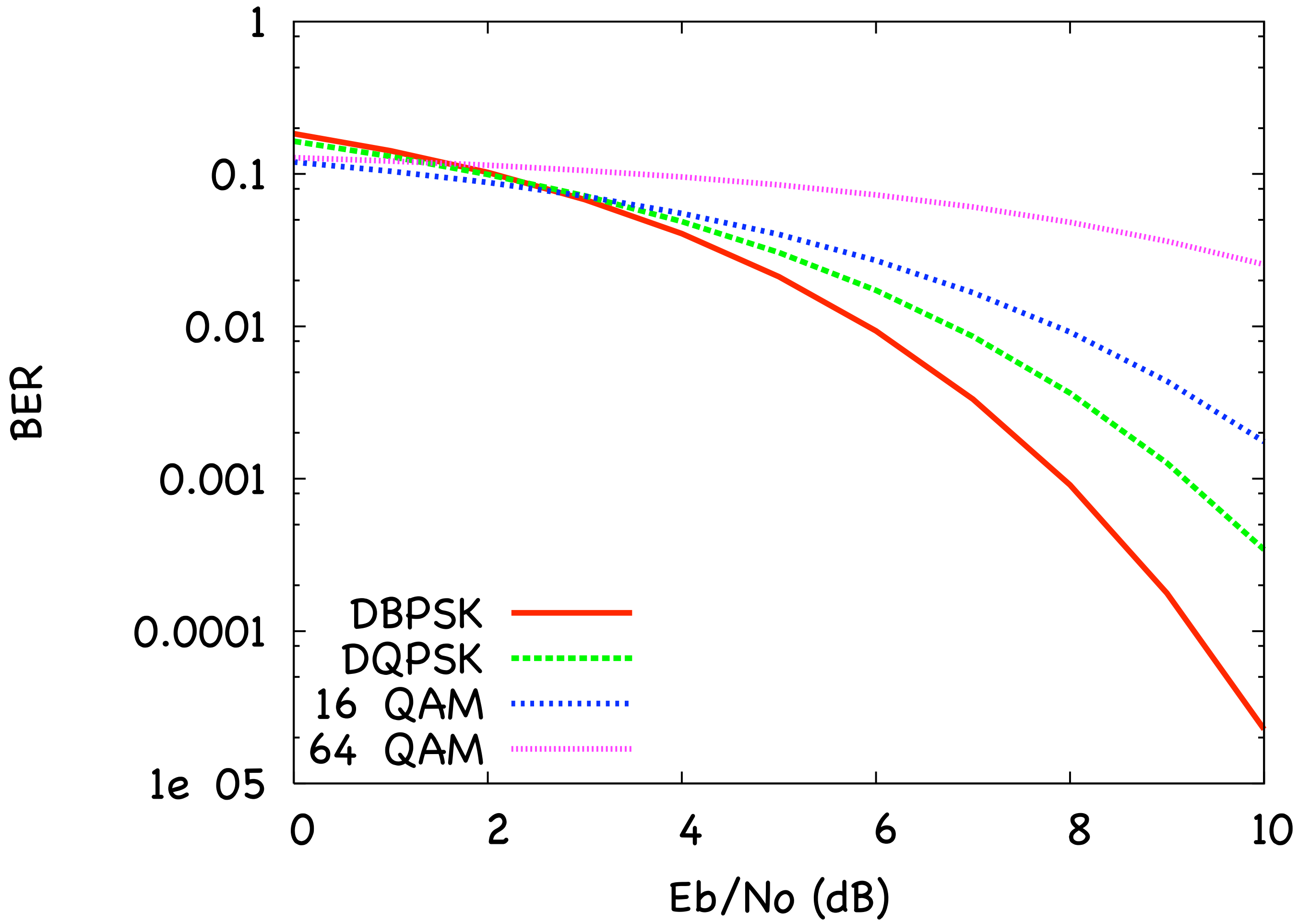
  - “Bus” type (HomePlug, “Old” Ethernet, ...)
  - Wireless (Wi-Fi, Bluetooth, ...)



Signal over noise ratio and collisions

- **The PHY layer bit-error-rate**
  - Depends on the received power (signal level)
  - Depends on the perturbation level (noise & interferences)
  - Defined by the used coding (modulation, ...)

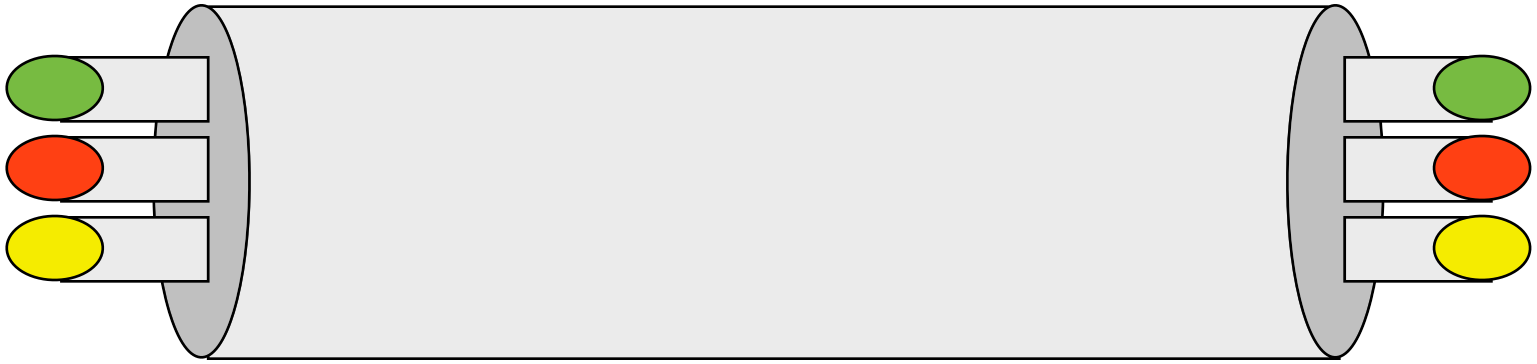
- **BER decreases when signal over noise ratio increases**



Scheduled access

FDMA (*Frequency Division Multiple Access*)

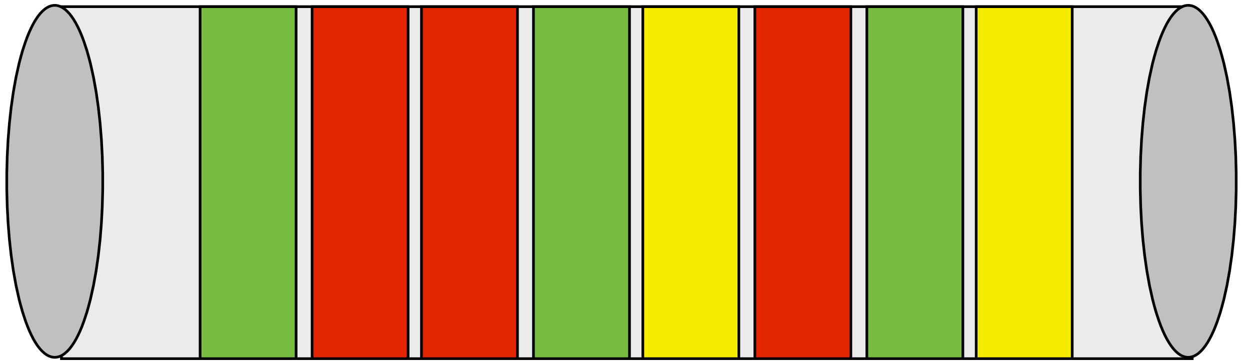
- **When multiple channels are available (e.g. frequencies), allocate one channel to each communication**



  - FM radio-like mechanism
  - Division provokes bandwidth loss: guard intervals between channels
  - Requires receivers to be able to detect channels on the whole frequency band
  - Insufficient when there are more emitters than available channels

TDMA (*Time Division Multiple Access*)

- **Each communication can be granted the whole bandwidth for a limited amount of time**



- Very suited to regular traffic (e.g. PCM sound)
- What happens when one station does not use the allocated time?
  - Performance loss in presence of irregular traffic

- Centralized mechanism - the allocation needs to be done somewhere.

Random access

Early random access

- **ALOHA: emit when a frame is ready, manage collisions**
  - 18 % efficiency; 36 % when allowing emission only at precise moments (slotted ALOHA)
- **Simple enhancement: CSMA = listen before talking**
  - Kleinrock & Tobagi; early 1970's
- **When an device wants to emit a frame:**
  - It checks the medium to verify that no other signal occupies the medium
  - If the channel is free, it transmits
  - Otherwise, it waits until the channel gets free

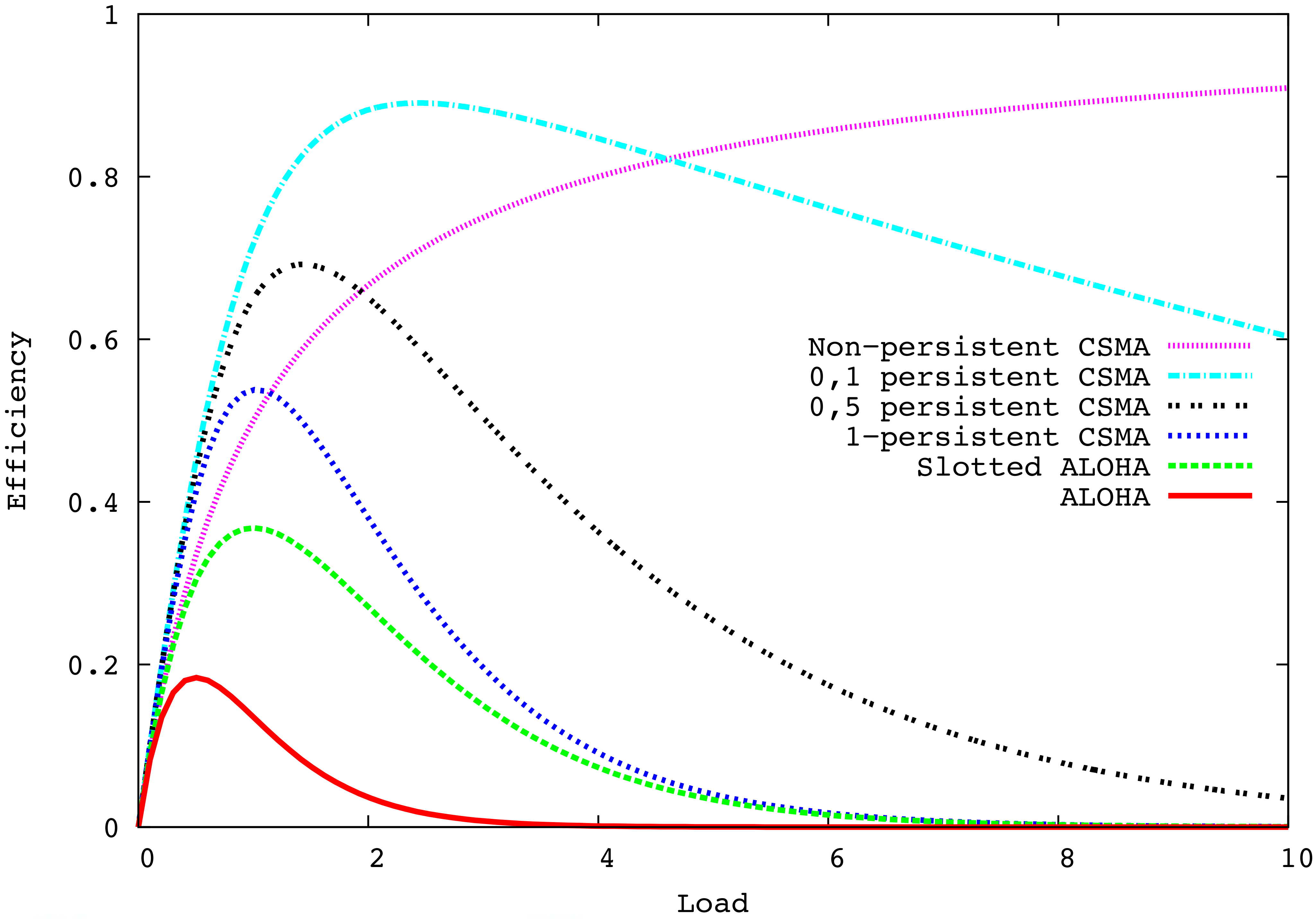
Radom access: desynchronizing emitters

- **When the channel gets free, multiple emitters can be waiting**
  - If they transmit as soon as the channel gets free => collision

- **p-persistent CSMA: every station will emit with a probability *p***

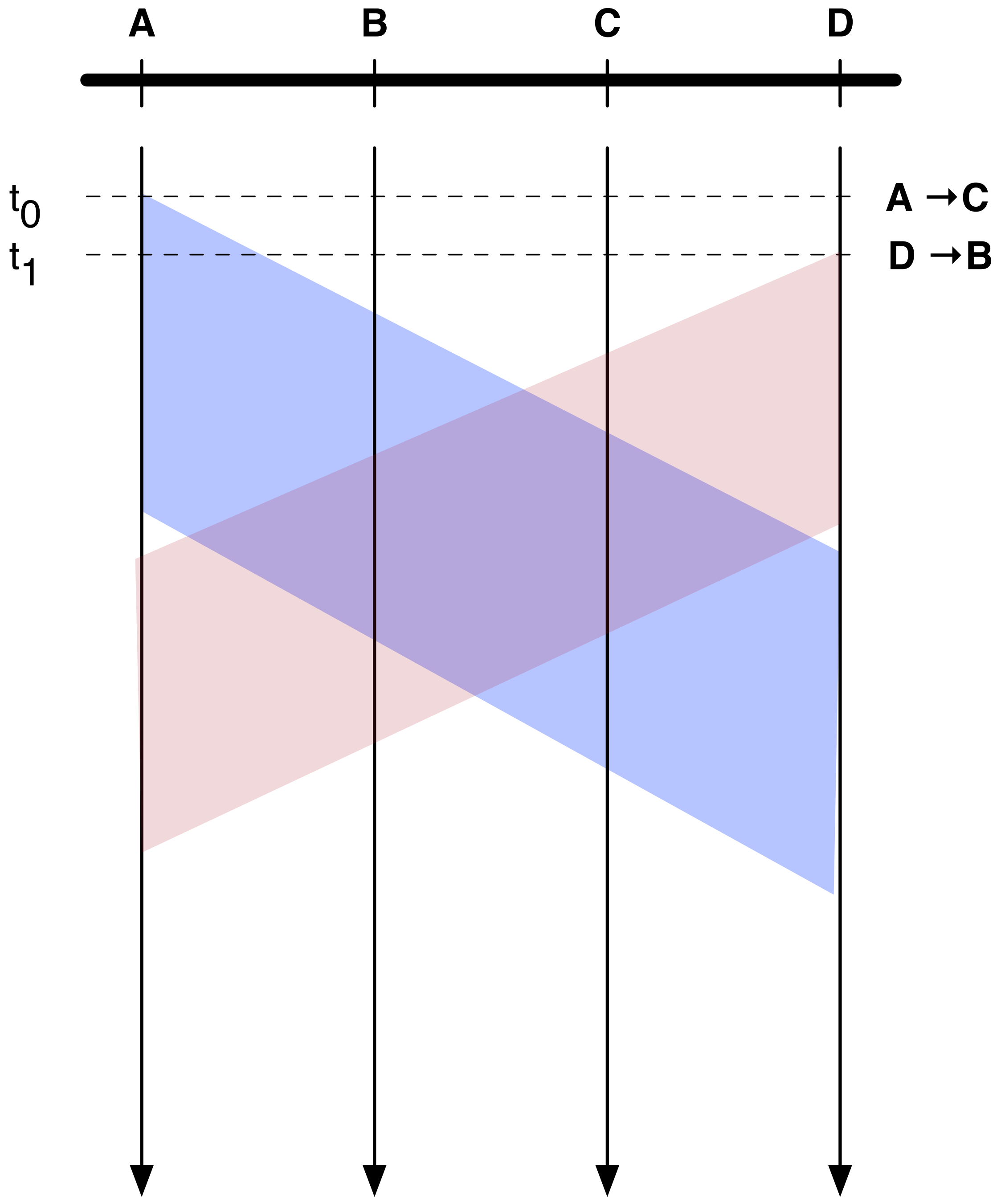
- **Slotted CSMA:**
  - Each emitter draws a random time in an interval (random backoff)
  - They all wait the corresponding time
  - The first one to acquire the channel blocks the others

Random access: performance



Propagation time influence

- **Collisions are always possible**
  - Two emitters can decide to start at the same moment
- **There exists a vulnerability interval for every frame**
  - All contenders must have noticed the signal
  - Depends on propagation time

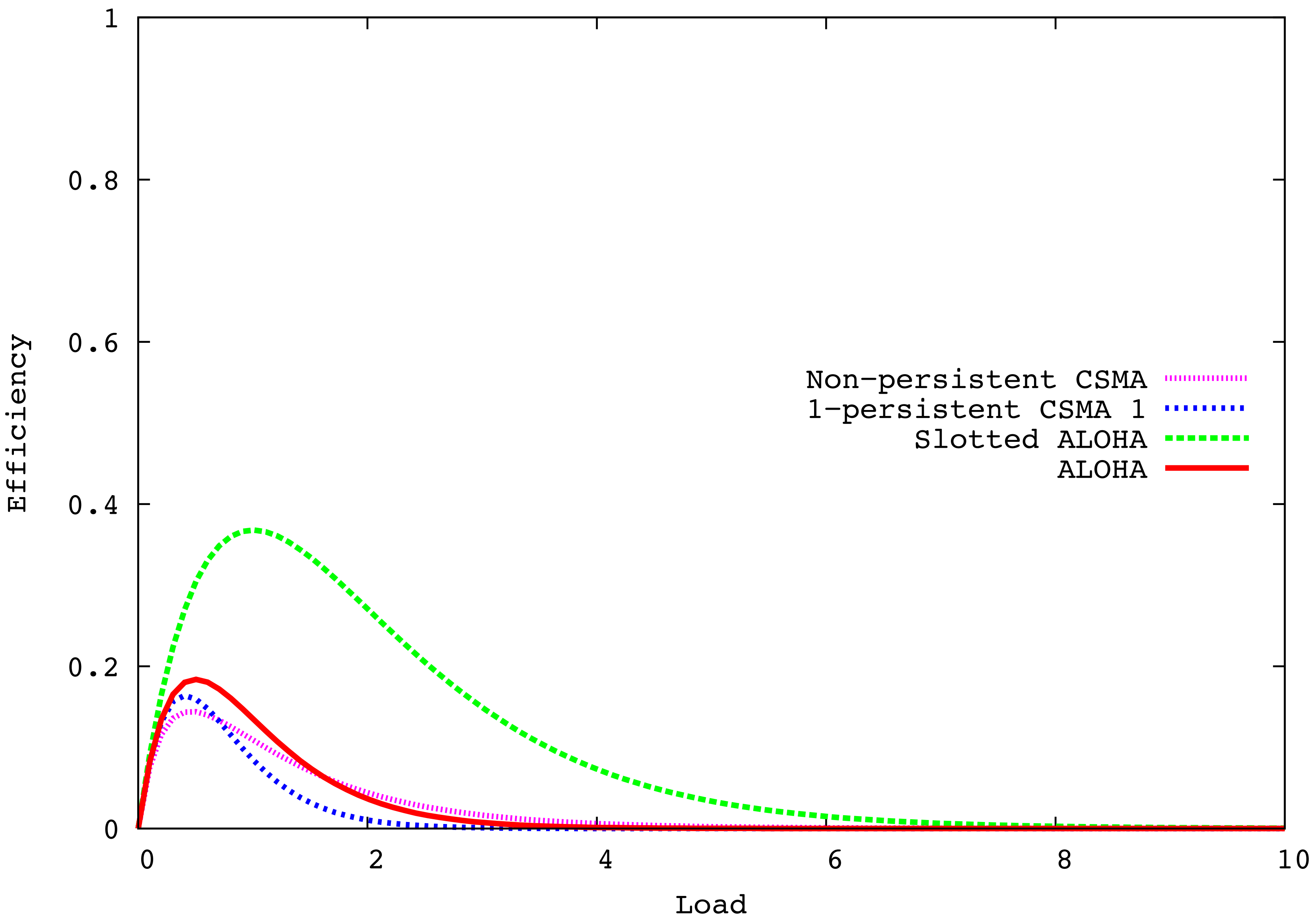


Propagation time influence (2)

- CSMA algorithms performance depends on propagation time

  - When propagation time approaches a frame transmission time, performance gets lower than ALOHA
    - Collisions
    - Waiting time

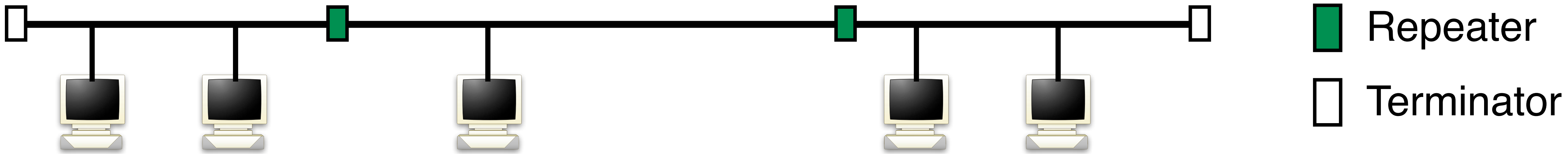
  - ⇒ Some protocols limit maximum cable length



Ethernet on a bus: CSMA/CD

Scenario

- **Bus-like network (shared medium)**
  - Repeaters can re-generate signal and allow increasing the distance limit imposed by attenuation in the cable
  - Terminators (resistors) at cable ends avoid echoes



- **Use of a CSMA-like protocol**
  - Look at the voltage on the cable before performing any transmission

Collisions management - CSMA/CD

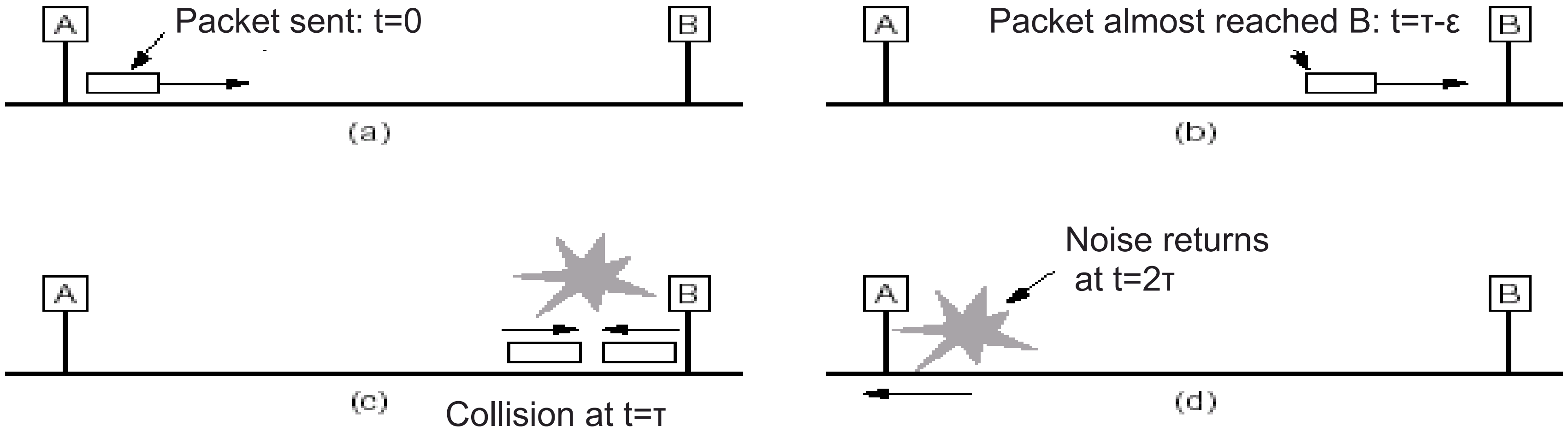
- **Emitter can detect itself when a collision happens:**
  - Compare emitted bits to the ones that pass on the medium.
  - A difference indicates a collision
- **When collisions are detected, stop transmission**
  - If the emitted cannot decode its own signal, the receiver will not be able to decode it either.
- **Exponential backoff: re-emit frame after a random time:**
  - Draw an integer in a pre-defined interval (contention window): $n \in [0; M]$
  - Wait for n times a constant time (slot time)
  - If collision repeats, double the interval every time: $M \rightarrow 2.M \rightarrow 4M$ ...

Ethernet - an emitter's point of view

- **Receive a packet from the network layer**
- **Encapsulate it in an Ethernet frame**
- **Listen to the channel**
  - If the channel is free, transmit
  - Otherwise, wait until it is free plus 96 bit-times before transmitting (9,6 μs at 10 Mb/s)
- **During transmission**
  - Listen to the medium to detect collisions
- **In case of collision**
  - Stop immediately the transmission
  - Send a 48 bits jamming signal
- **After nth collision, exponential backoff:**
  - Draw a random number *k* in [0; 2^m - 1] (m = min (n,10))
  - Wait *k* times 512 bit-times (i.e. 51,2 μs at 10 Mb/s)
  - Limit the number of retries to 15

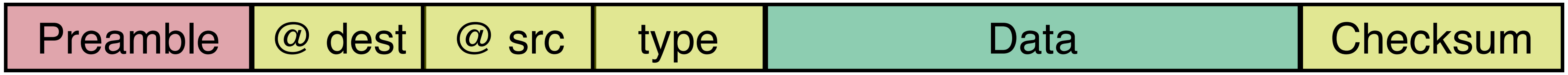
Why a 48 bits jamming signal?

- **All other terminals need to detect the collision**
  - Example: A emits, B emits just before A’s signal reaches B
  - B stops right away, after a few bits
  - A does not necessarily detect the collision (too low energy, A’s frame already finished, …)
- **Jamming signal ensures every collision is detected**



Source: A. Tannenbaum - Networks

Ethernet frame format

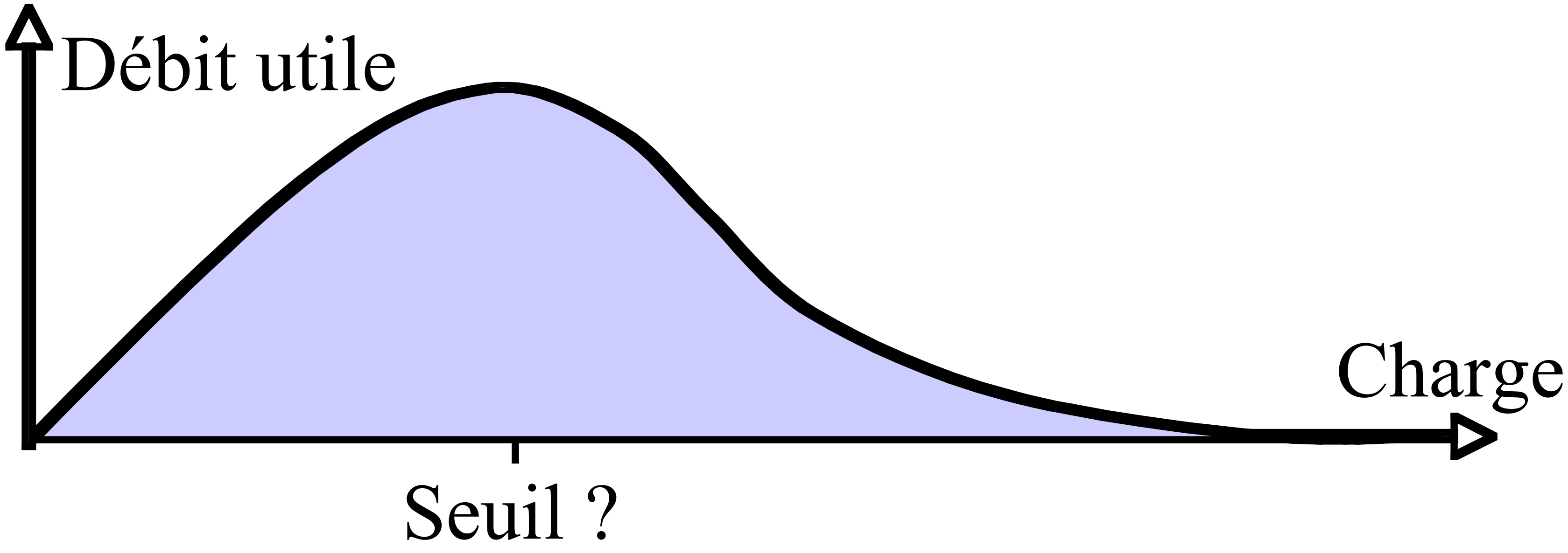


- **Preamble (8 bytes)**
  - 7x (10101010) for synchronization + 1x (10101011)
- **@ dest, @src (2 x 6 bytes)**
  - peers MAC addresses
- **Type (2 bytes)**
  - Identifies the network layer protocol (IP, IPX, ...)
- **Data (48 to 1500 bytes)**
  - If less than 48 bits, add padding
- **Checksum (4 bytes)**
  - CRC based on the @dest to data part

- **Emission order**
  - Bytes: in the left to right order (preamble sent first)
  - Bits: LSB (little endian)

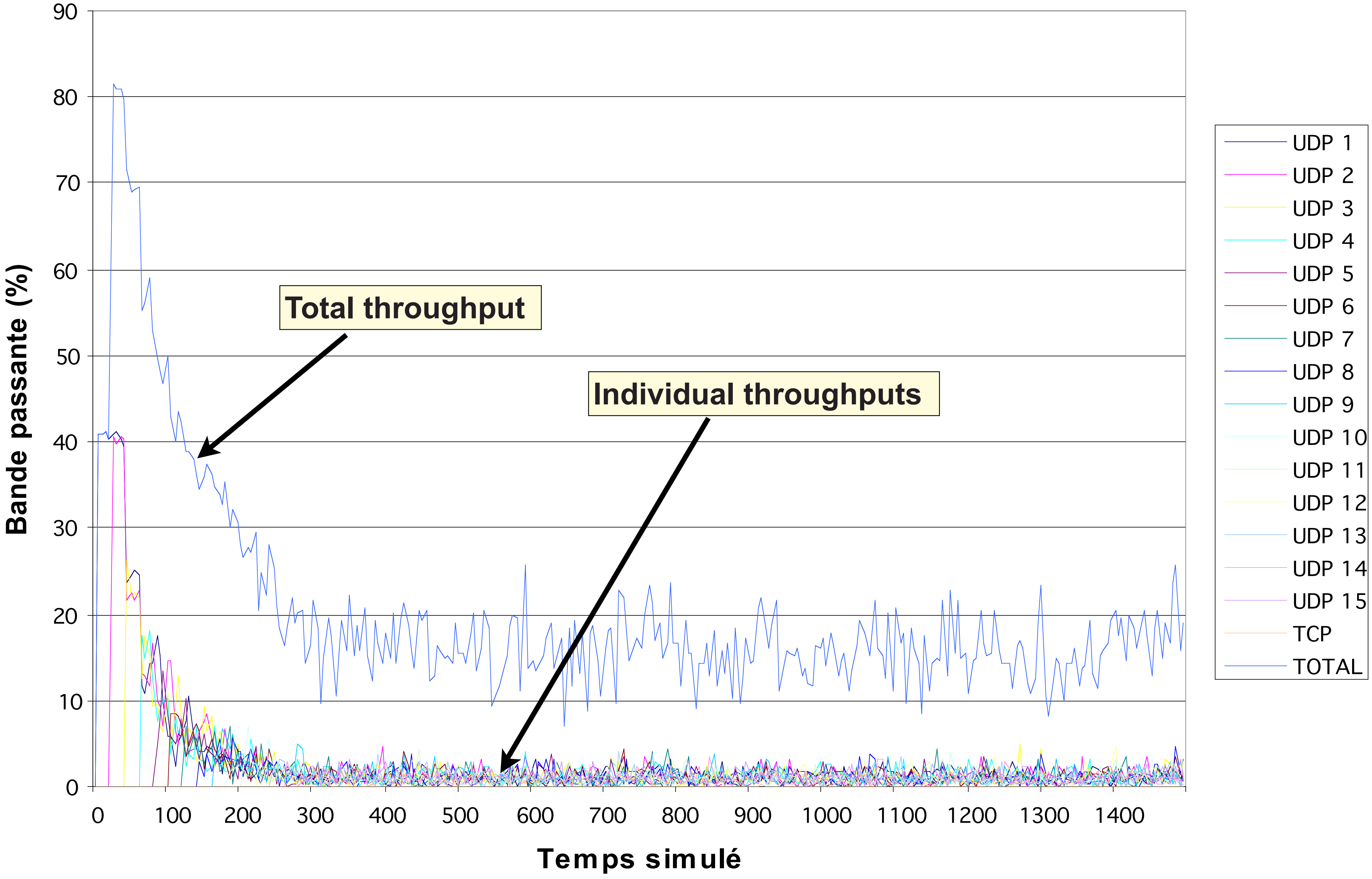
Performance

- **Depends directly on the collision probability**
  - If there are too many collisions, terminals spent their time waiting



  - Low load: optimal performance (No waiting time)
  - High load: collapse

Performance collapse simulation



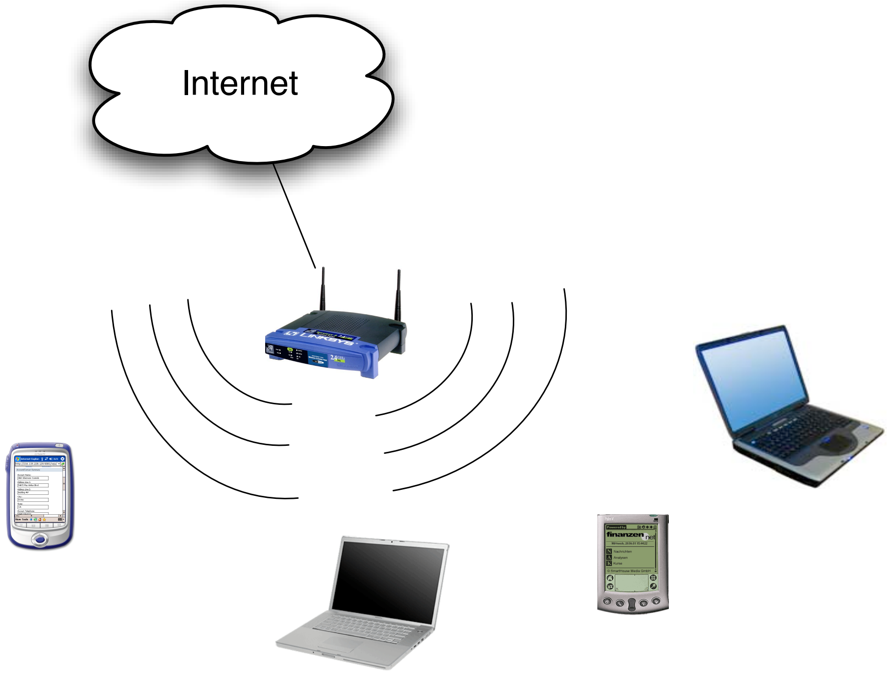
Simulation: periodical addition of emitters, each trying to occupy 40% of the medium capacity

Ethernet today

- **Topologies have changed**
  - No bus-like architectures anymore, mostly switched Ethernet
  - CSMA/CD unused today

- **Throughput increase**
  - Fast Ethernet (100 Mb/s)
  - Gigabit Ethernet
    - Standard in most of today's computers
  - 10 Gb/s Ethernet coming

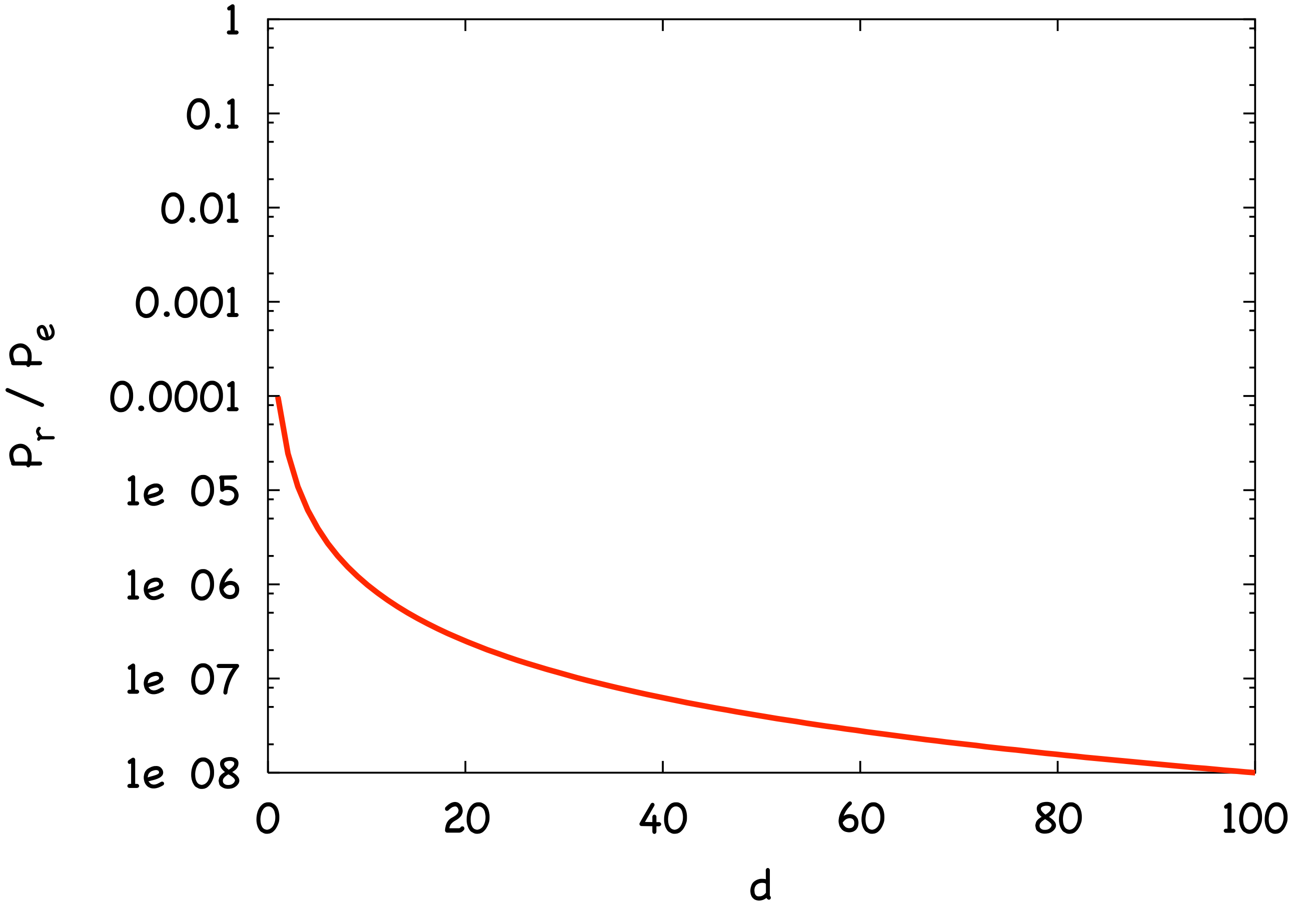
Wi-Fi (IEEE 802.11) — CSMA/CA



Wireless (radio-frequencies) attenuation

- **Free space (idealistic conditions):**

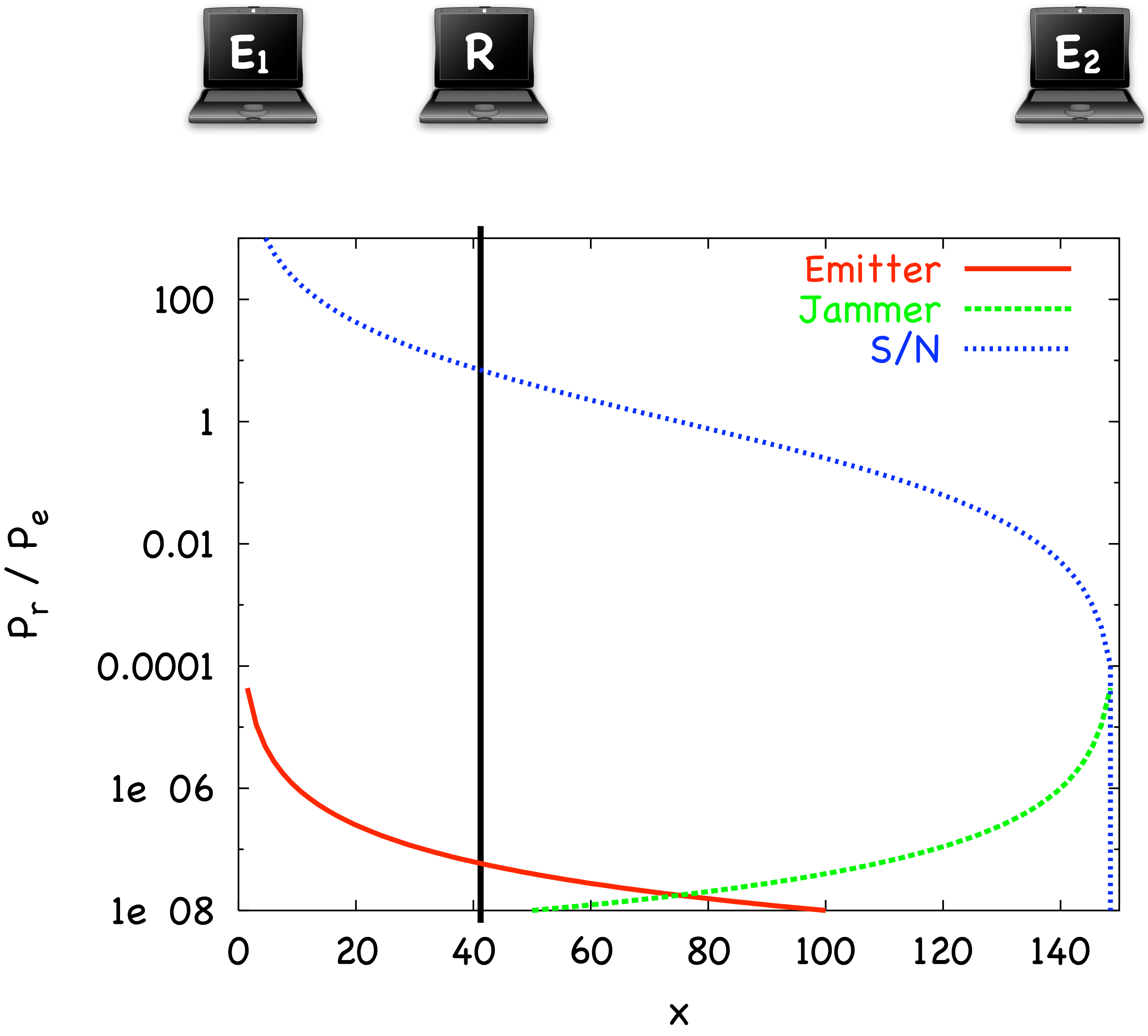
$$P_r = \left(\frac{\lambda^2}{16 \cdot \pi^2 \cdot d^2}\right) \cdot G_e \cdot G_r \cdot P_e$$



Pr: Received power
Pe: Emitted power
λ: wavelength
d: distance between emitter and receiver
Gr: receiver antenna gain
Ge: emitter antenna gain

- **Signal level decreases in function of the square of the distance (best case)**

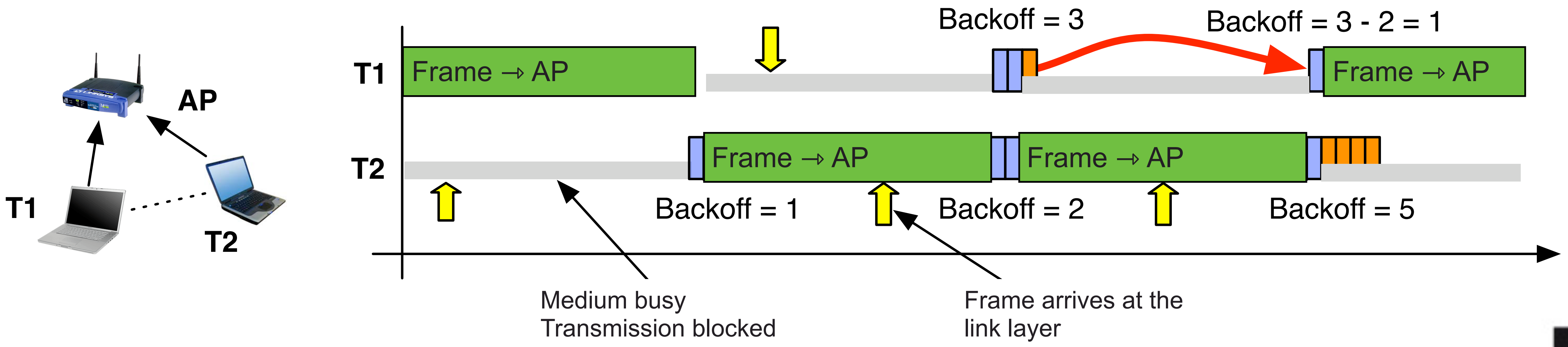
Wireless collisions



- **The emitter cannot detect collisions**
  - Impossible to stop an ongoing frame transmission
- **Explicit acknowledgments required**
  - Wi-Fi: *stop & wait*

Frame emission

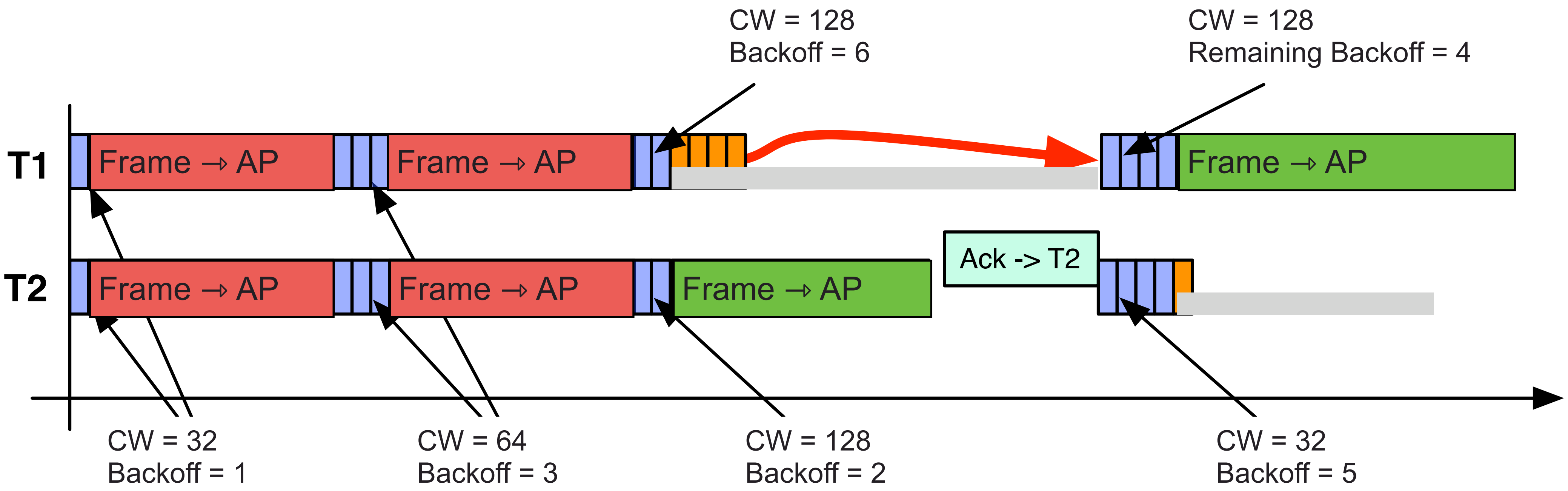
- **A frame transmission that starts will be ended (no stop)**
- **CSMA/CA: a classical, yet cautious CSMA flavor**
  - When the medium is free, emitter always waits a random backoff
    - Ethernet: one try without waiting
  - The initial contention window is larger than Ethernet’s (16 vs. 2)
    - When contention is lost, we keep the backoff value for the next try
  - There is a backoff between two successive frames
    - Allow other emitters to win the contention



Acknowledgments & retransmissions

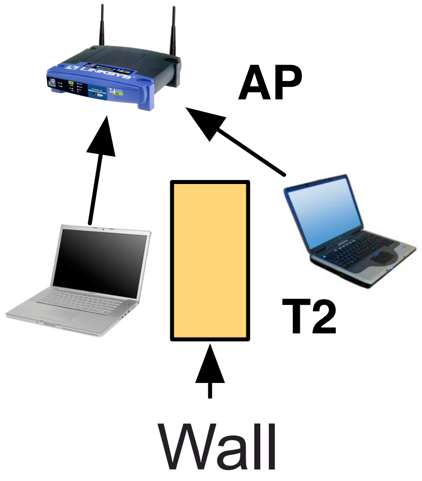
- **Every *unicast* frame must be acknowledged explicitly**
  - Stop-and-wait explicit acknowledgments

- **Retransmission when no acknowledgment arrives**
  - Contention window size is doubled (max: 1024)
  - Cancel after a limited number of tries (between 4 and 7)

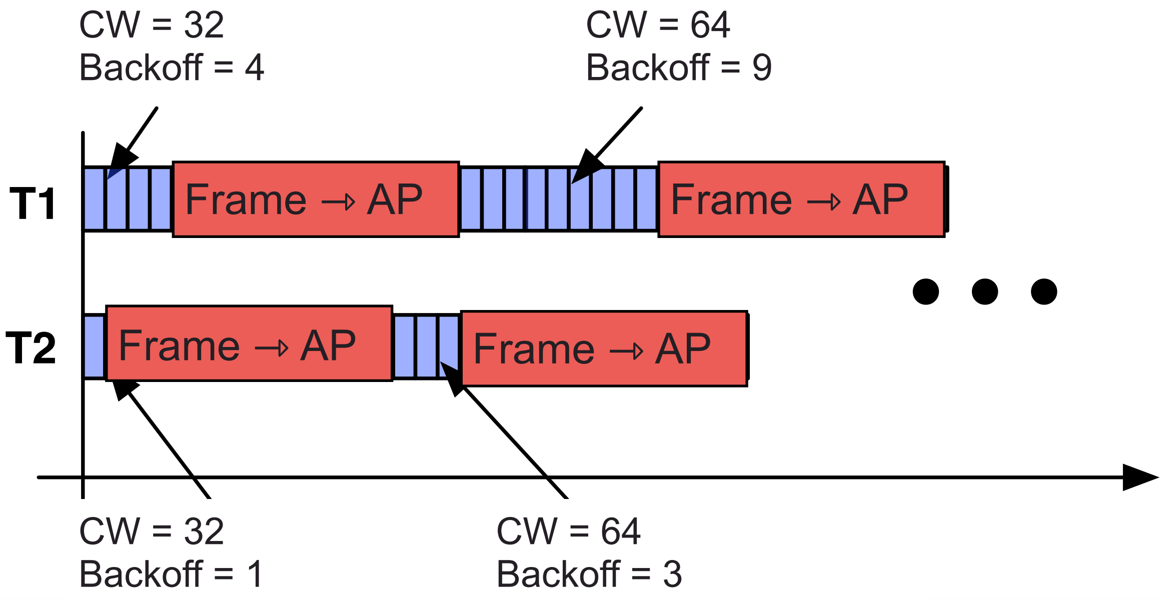


The hidden node scenario

- **Everything depends on the carrier detection**
  - What happens when two emitters do not sense their mutual emissions?

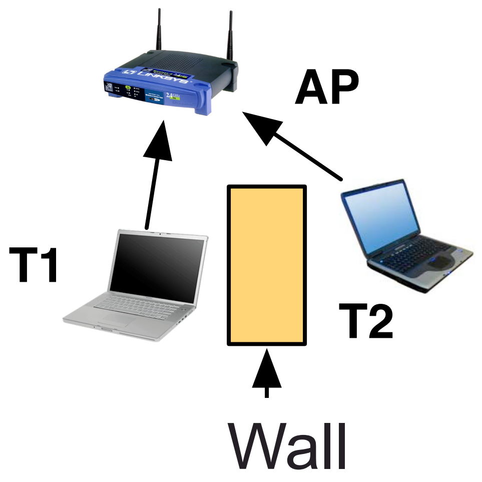


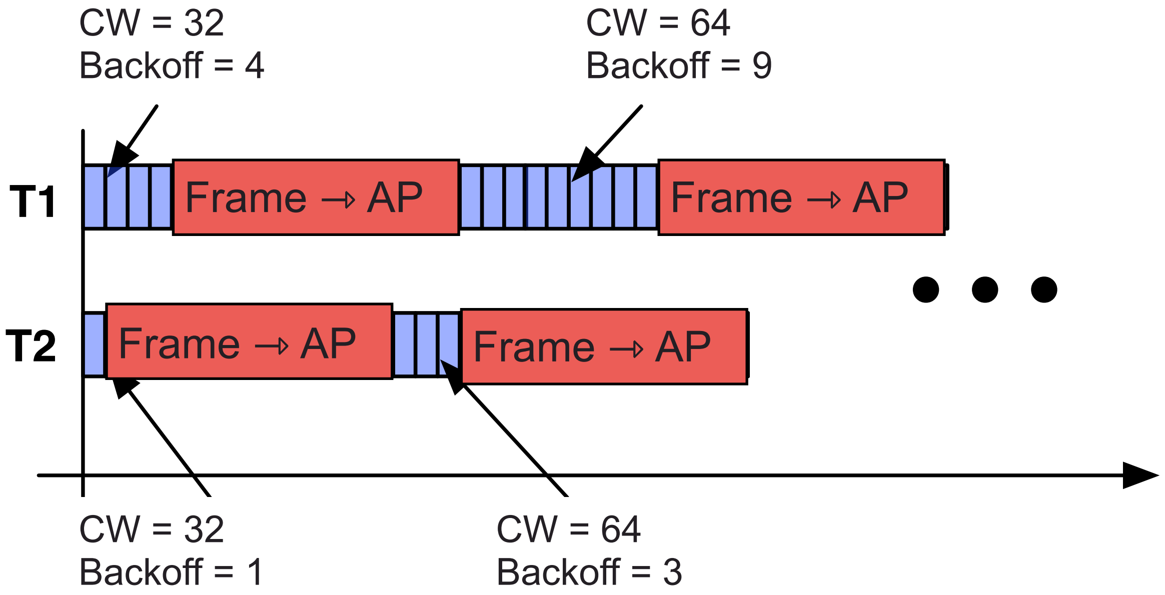
- **If frames are too long, repeated collisions:**



Origin of the issue

- The problem comes from the ratio between the average backoff and the frame transmission time



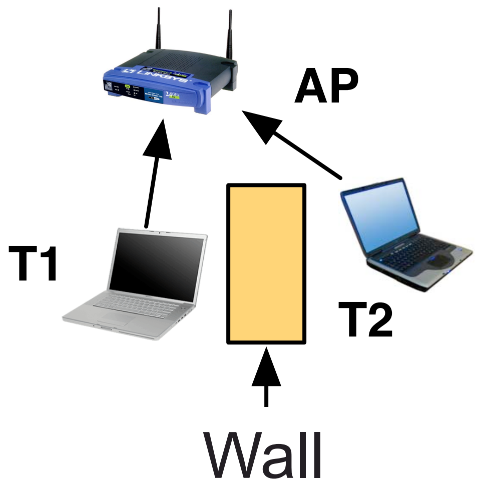


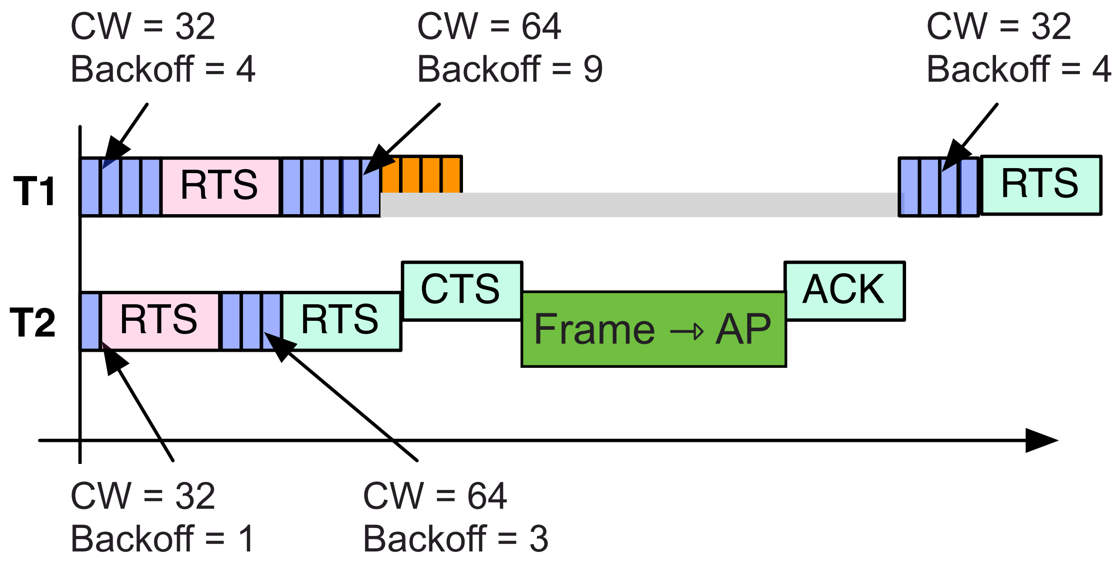
- Example: 1500 bytes at 11 Mb/s: 343 *µs*

CW	32	64	128	256	512	1024
Average backoff	310 µs	630 µs	1270 µs	2550 µs	5110 µs	10230 µs

RTS-CTS exchange

- **Solution: precede the large frames emission by a short exchange:**
  - RTS (*Request To Send*) from emitter to receiver (ask for authorization to send)
  - CTS (*Clear To Send*) from receiver to emitter (authorization granted)
    - Broadcast RF medium: the answer notifies all neighbors in range





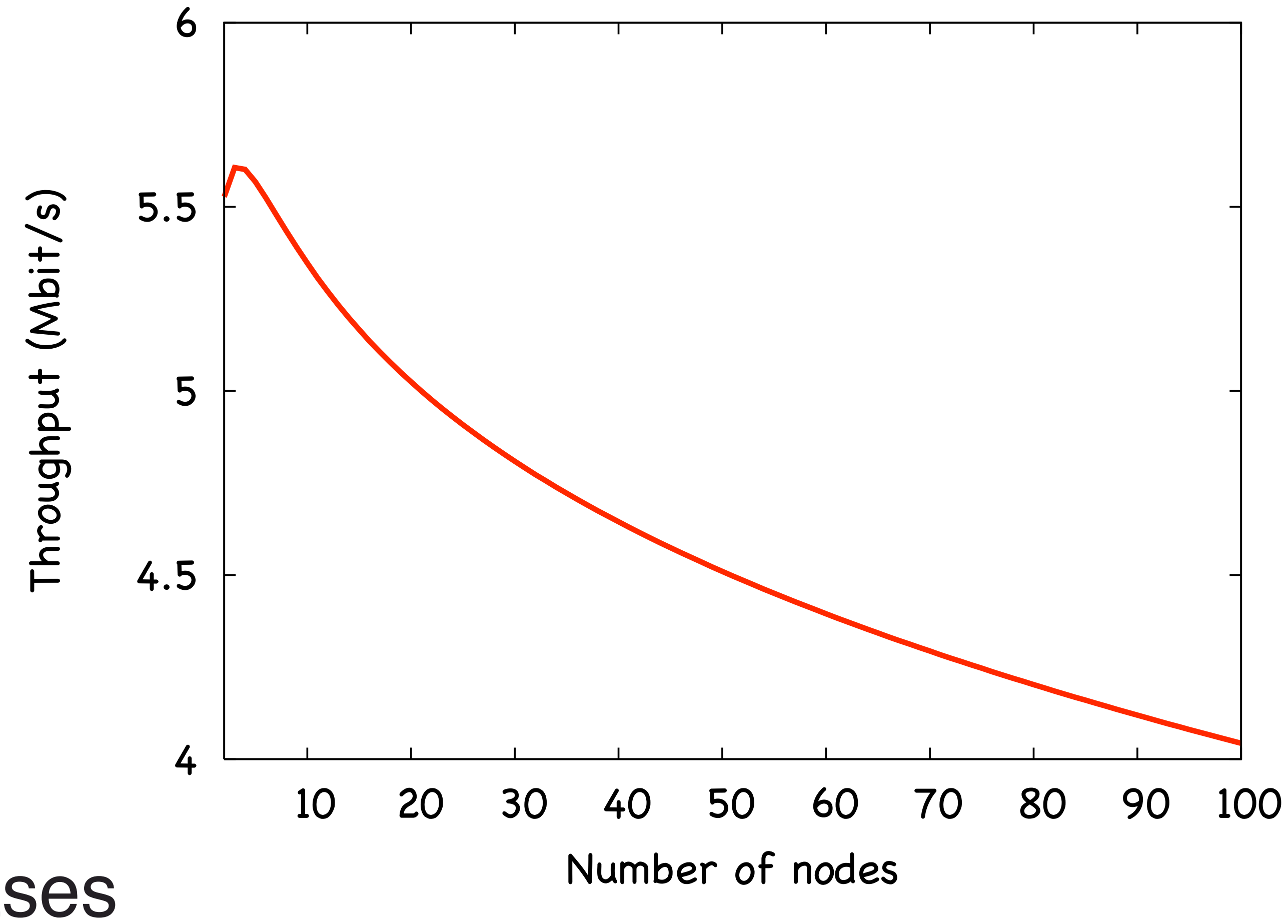
- **CTS contains the expected transmission time**
  - Other emitters are blocked during this time

MAC layer performance

- **Exact performance: probabilistic calculation (Markov chains)**

- **High overhead**
  - Backoff, RTS, CTS, acks, ...
  - Efficiency around 50% to 60%

- **When the number of emitters increases**
  - *backoffs* are spent in parallel ⇒ performance gain
  - Quickly the collision probability increases ⇒ performance decrease

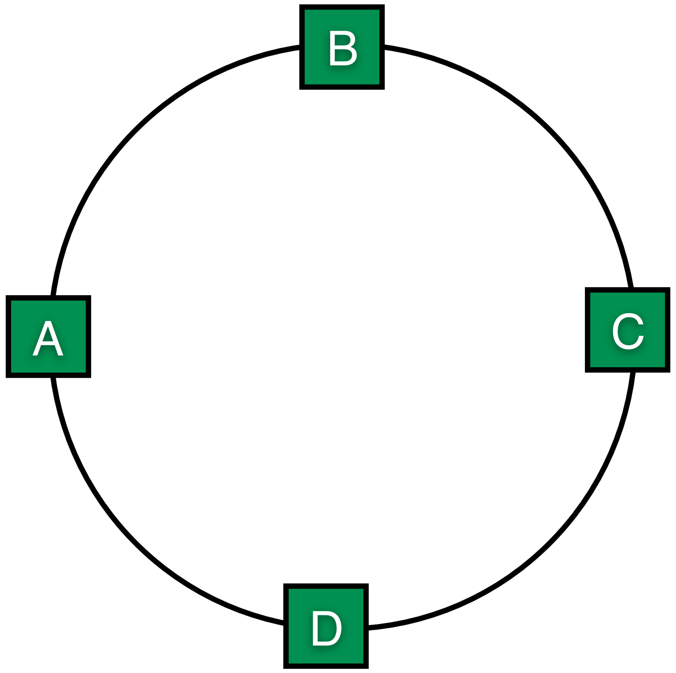


Access equity = throughput fairness?

- **Wi-Fi proposes multiple throughputs**
  - 1, 2, 5.5, 11, 18, 27, 33, 54, ... Mbit/s
  - Different modulations ⇒ different ranges and noise reactions
  - **Each station seeks the best compromise between losses and throughput**
- **A 1 Mb/s station vs. all others at 54 Mb/s:**
  - Emission on the same shared channel
  - Wi-Fi grants packet-level fairness
  - Each station can only emit one frame per turn
  - 1 Mb/s transmission increases turn duration
  - Everybody gets a 1 Mb/s throughput

Decentralized collision-free protocols

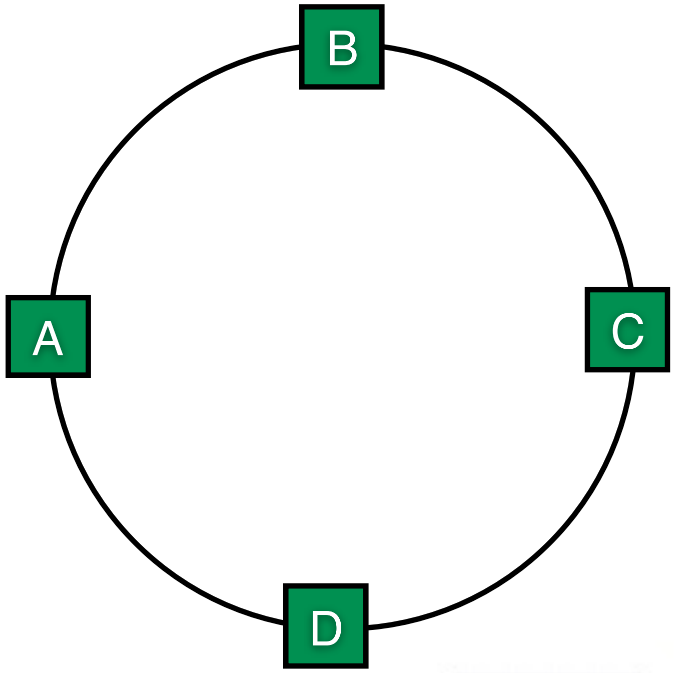
Token Ring example



Token Ring — Introduction

- **History**
  - Created in 1969
  - Introduced by IBM in the 1980’s
  - Standardized by IEEE (IEEE 802.5) in 1985
  - Not deployed anymore today

- **Ring-like topology**
  - All stations are connected to neighbors by point-to-point links

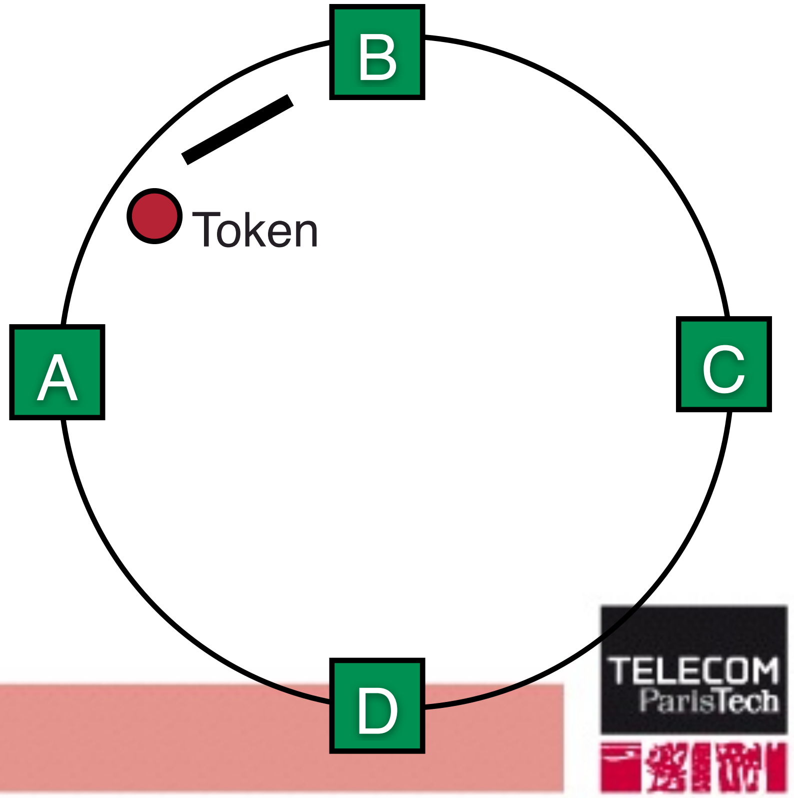


- **Typical throughputs**
  - IBM: 4 Mb/s
  - IEEE 802.5: 16 Mb/s

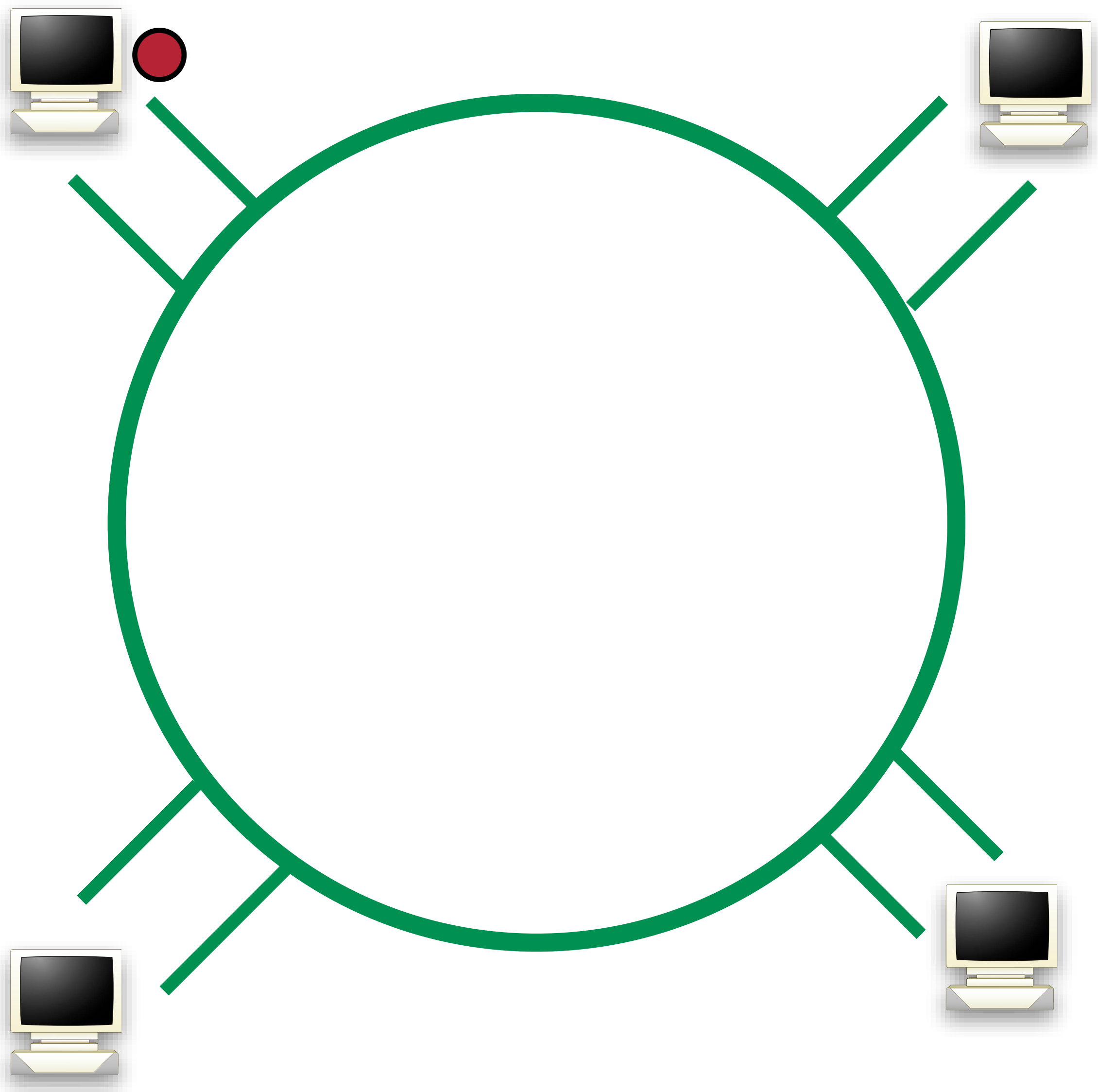
Access technique

- **One-direction ring**

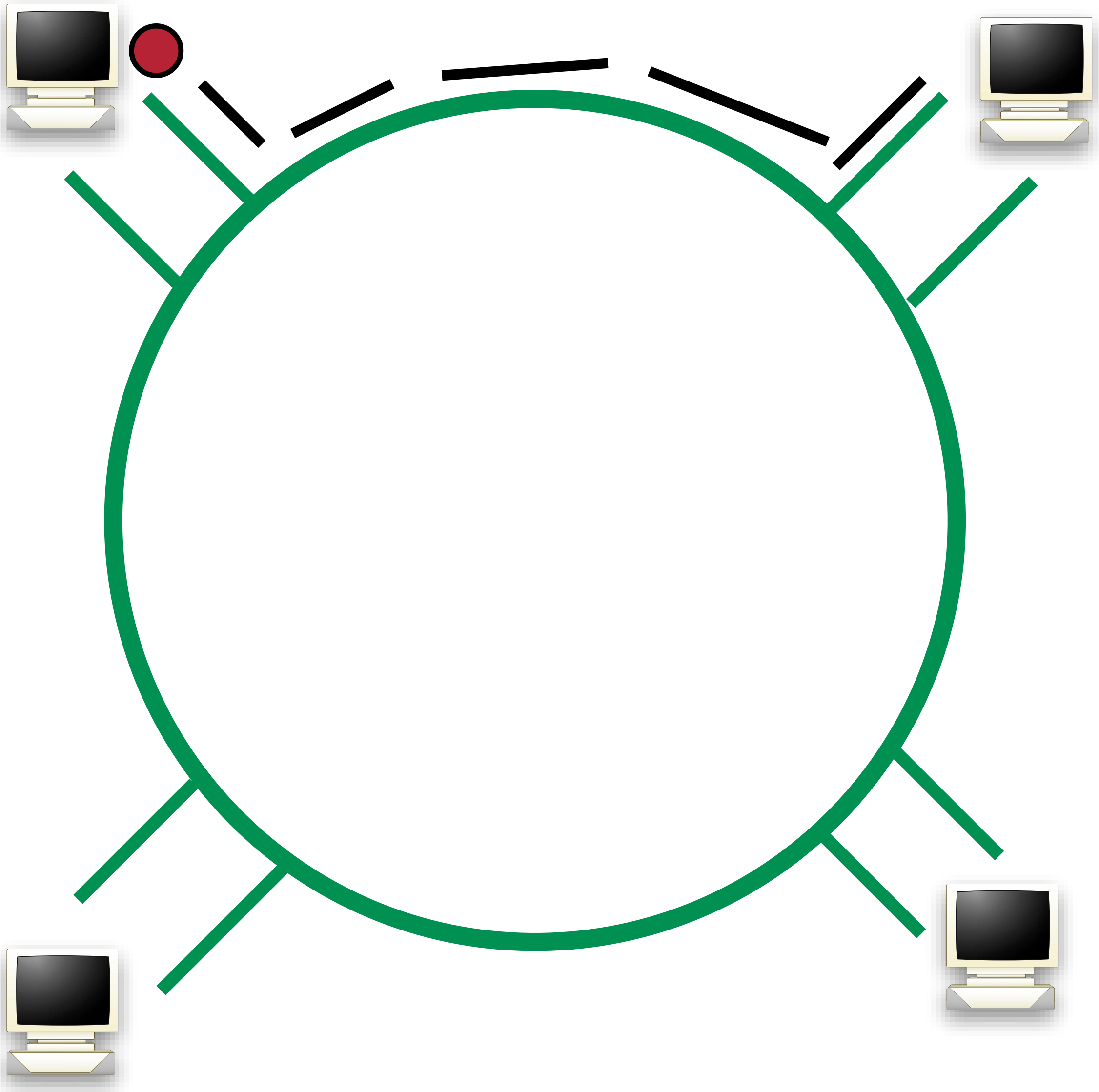
- **Use of a virtual token to organize access**
  - Only the station that possesses the token is allowed to transmit
  - No collision
  - Station passes the token to its neighbor once the transmission is finished



Token passing

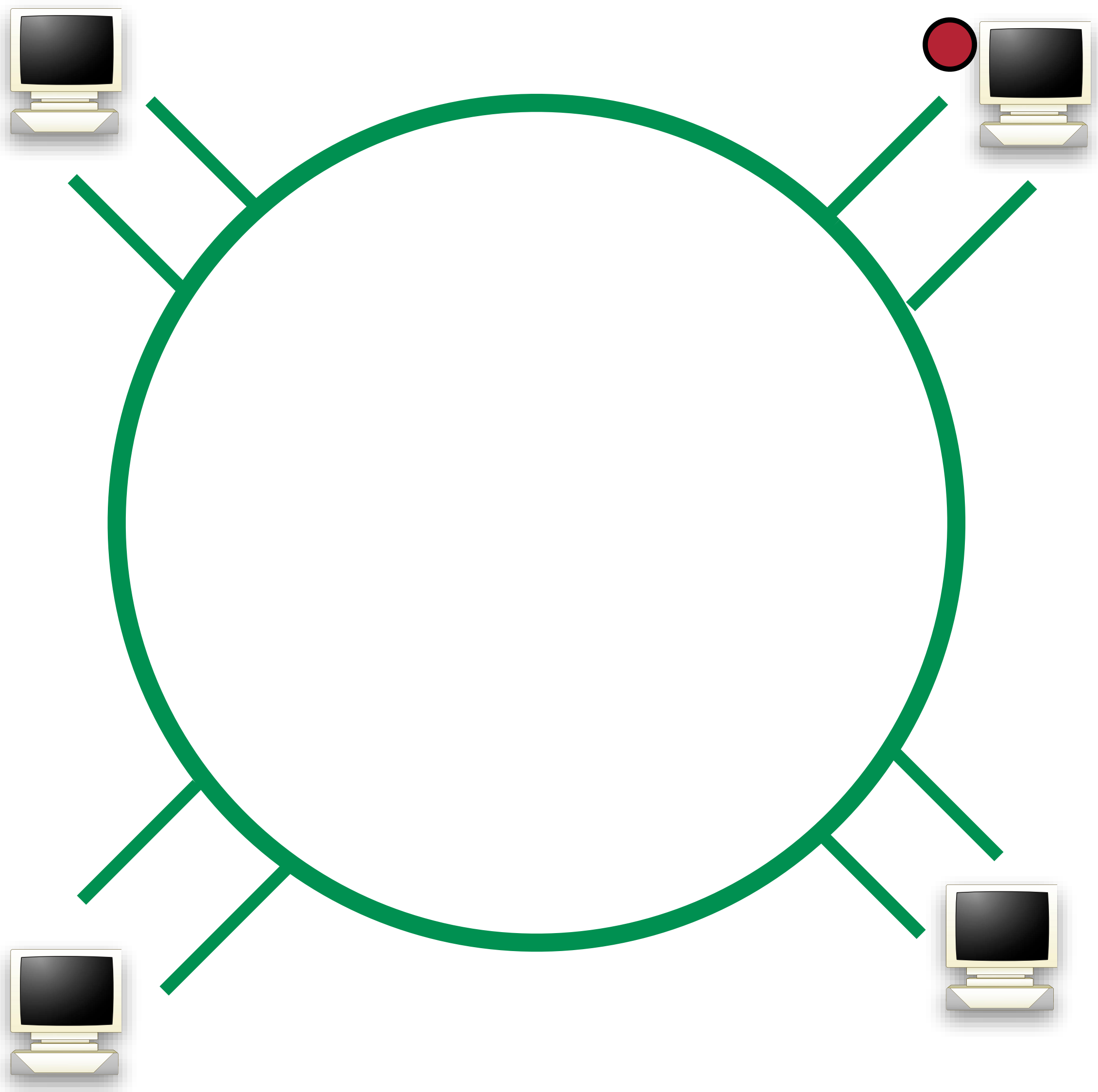


Token passing

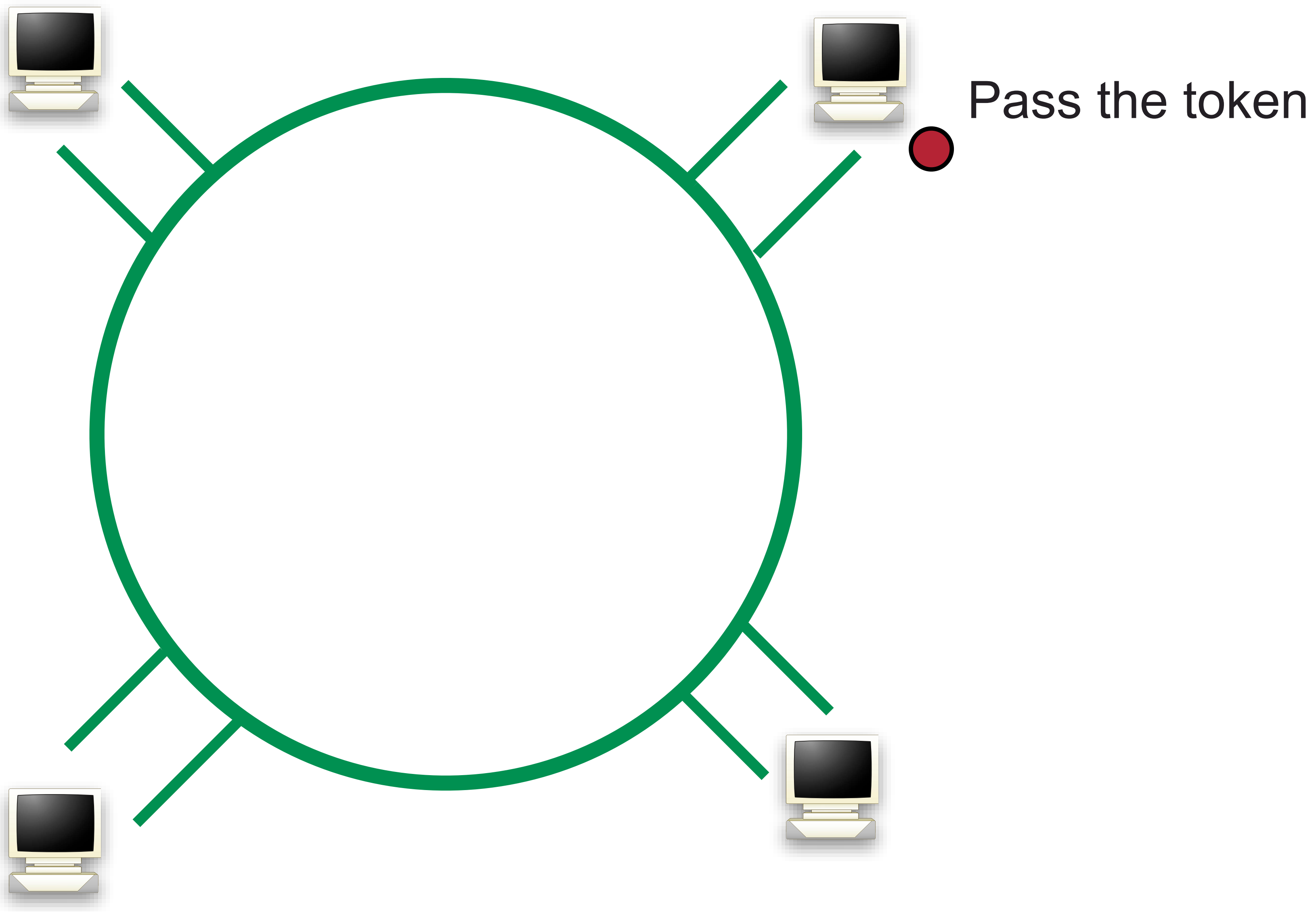


Token passing

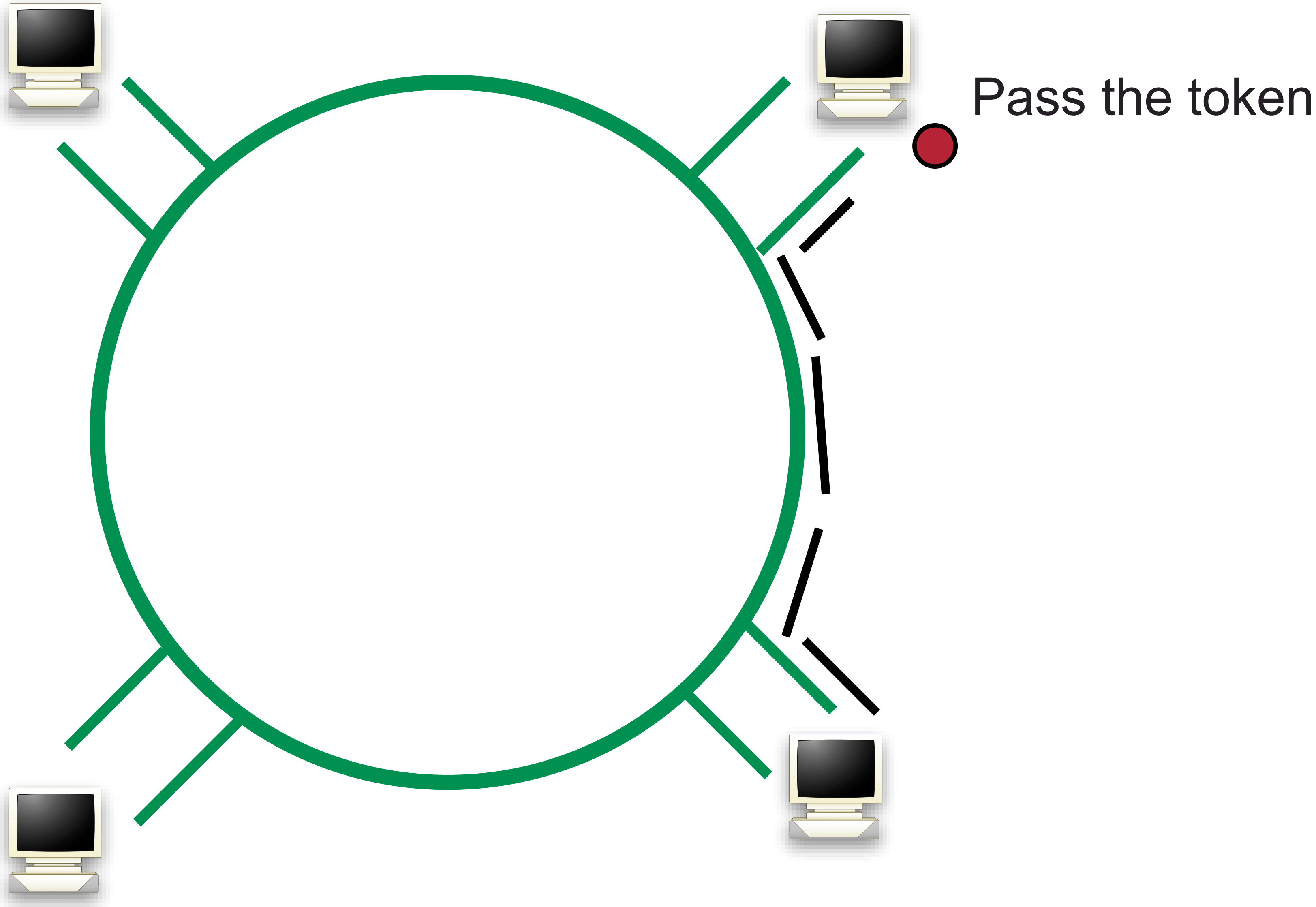
Frame awaiting?



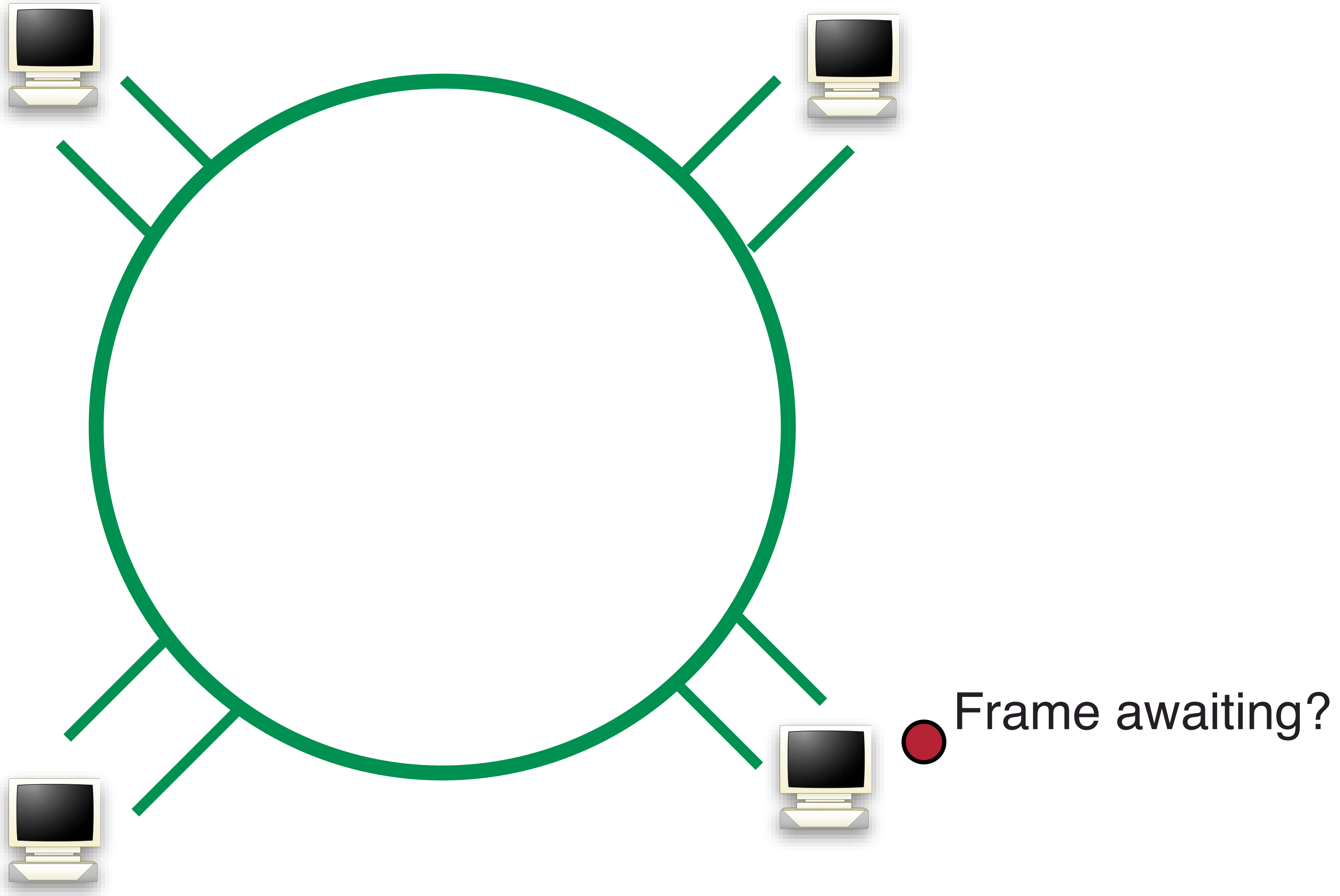
Token passing



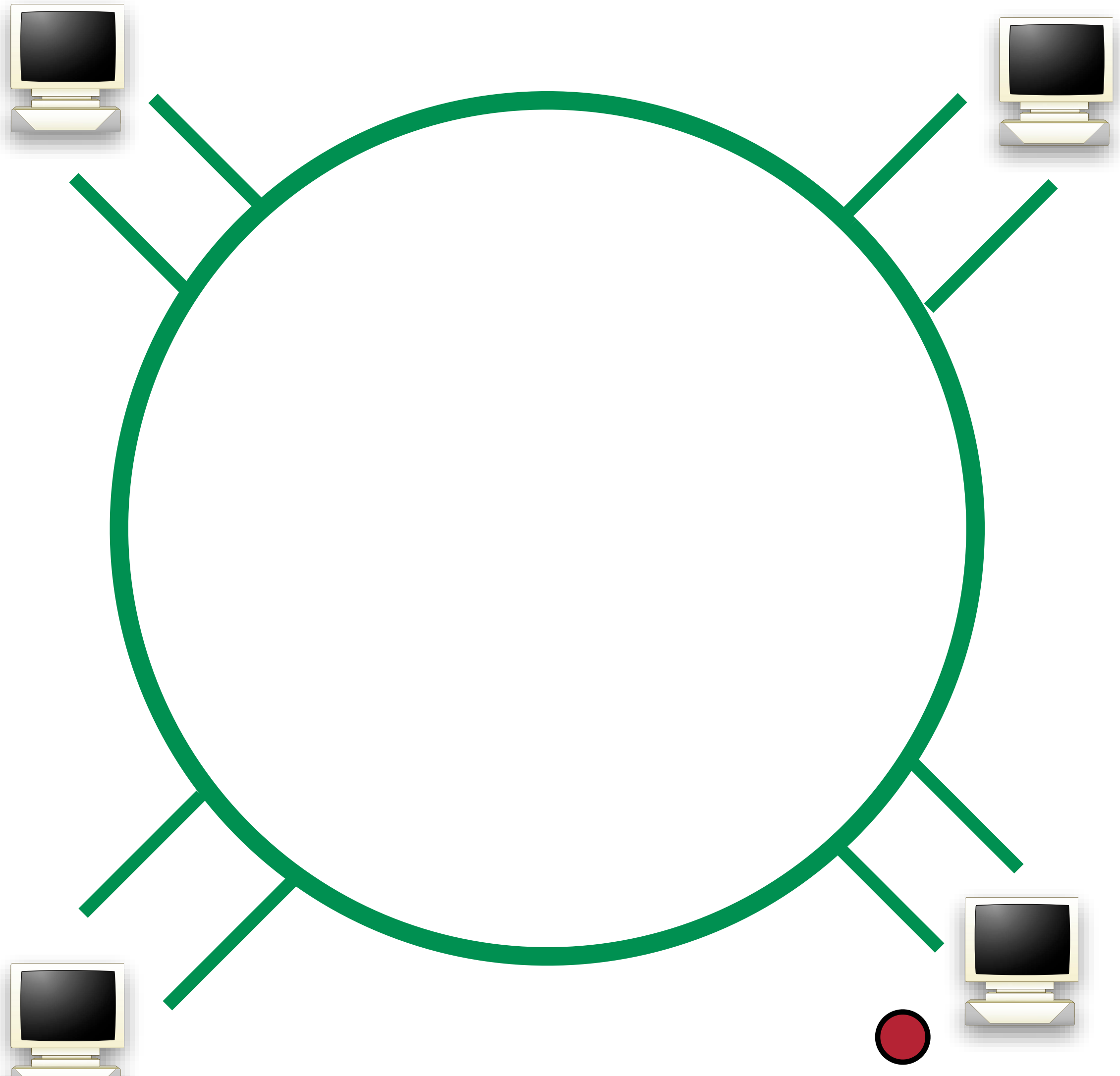
Token passing



Token passing

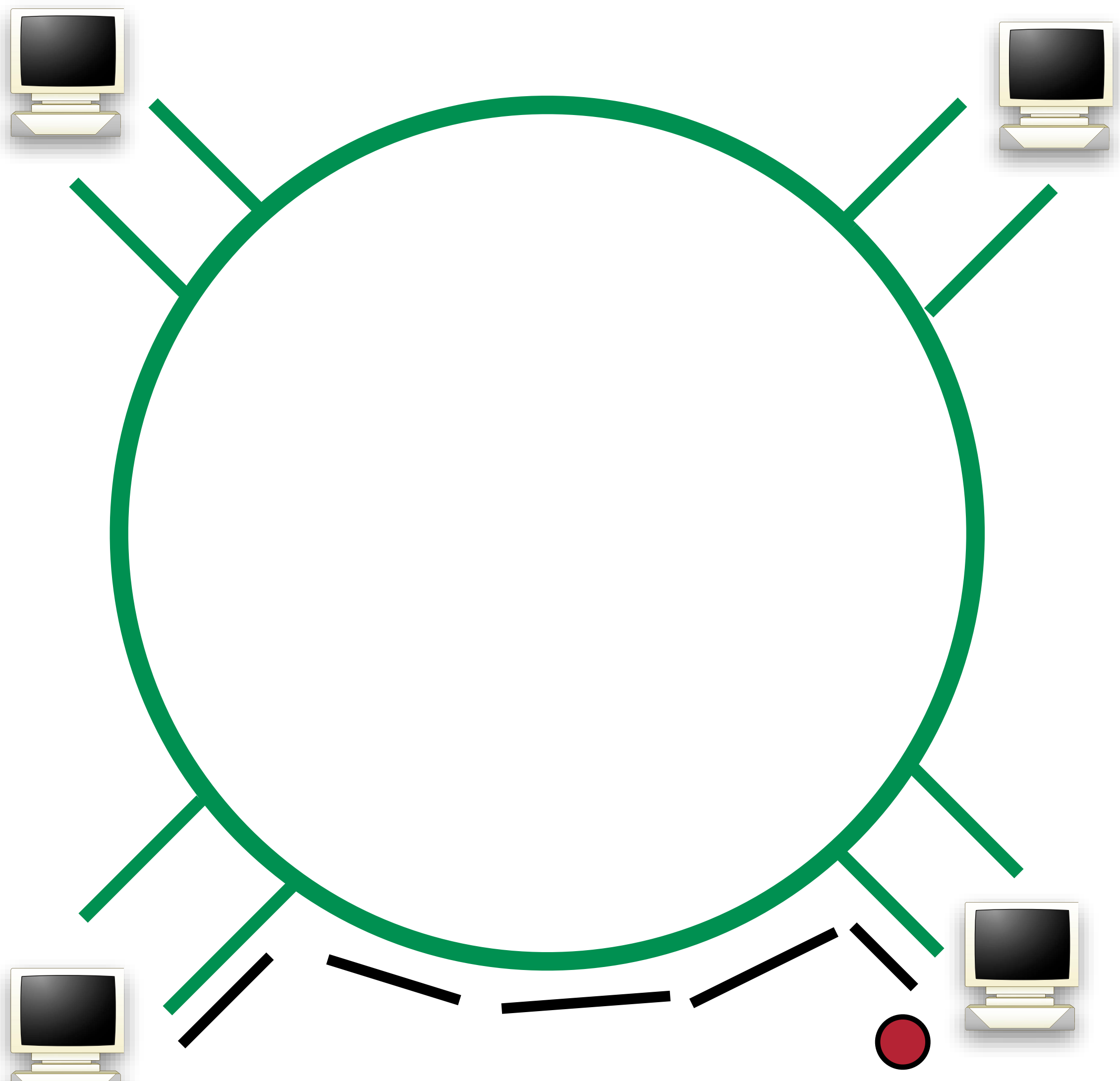


Token passing



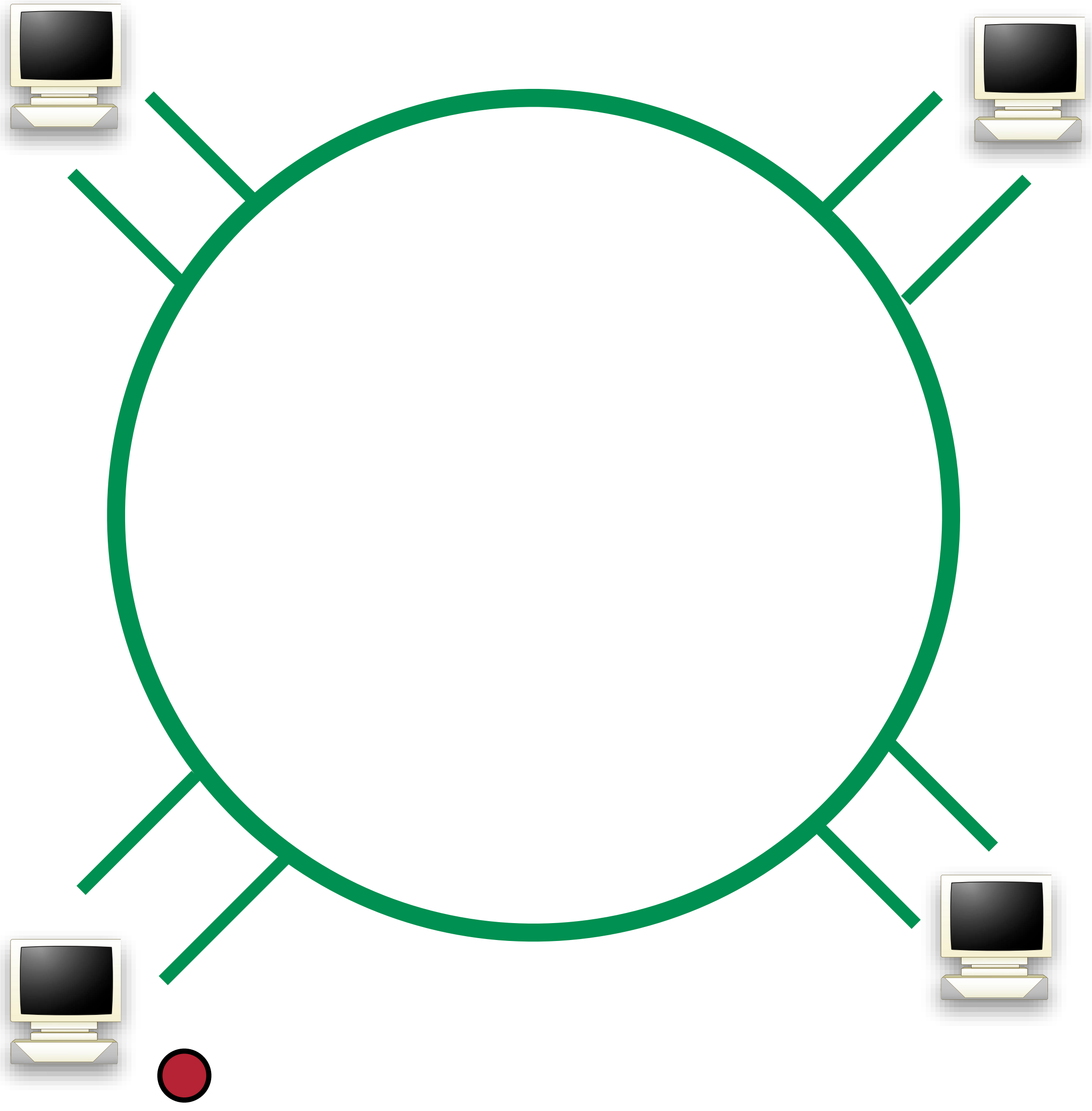
Pass the token

Token passing



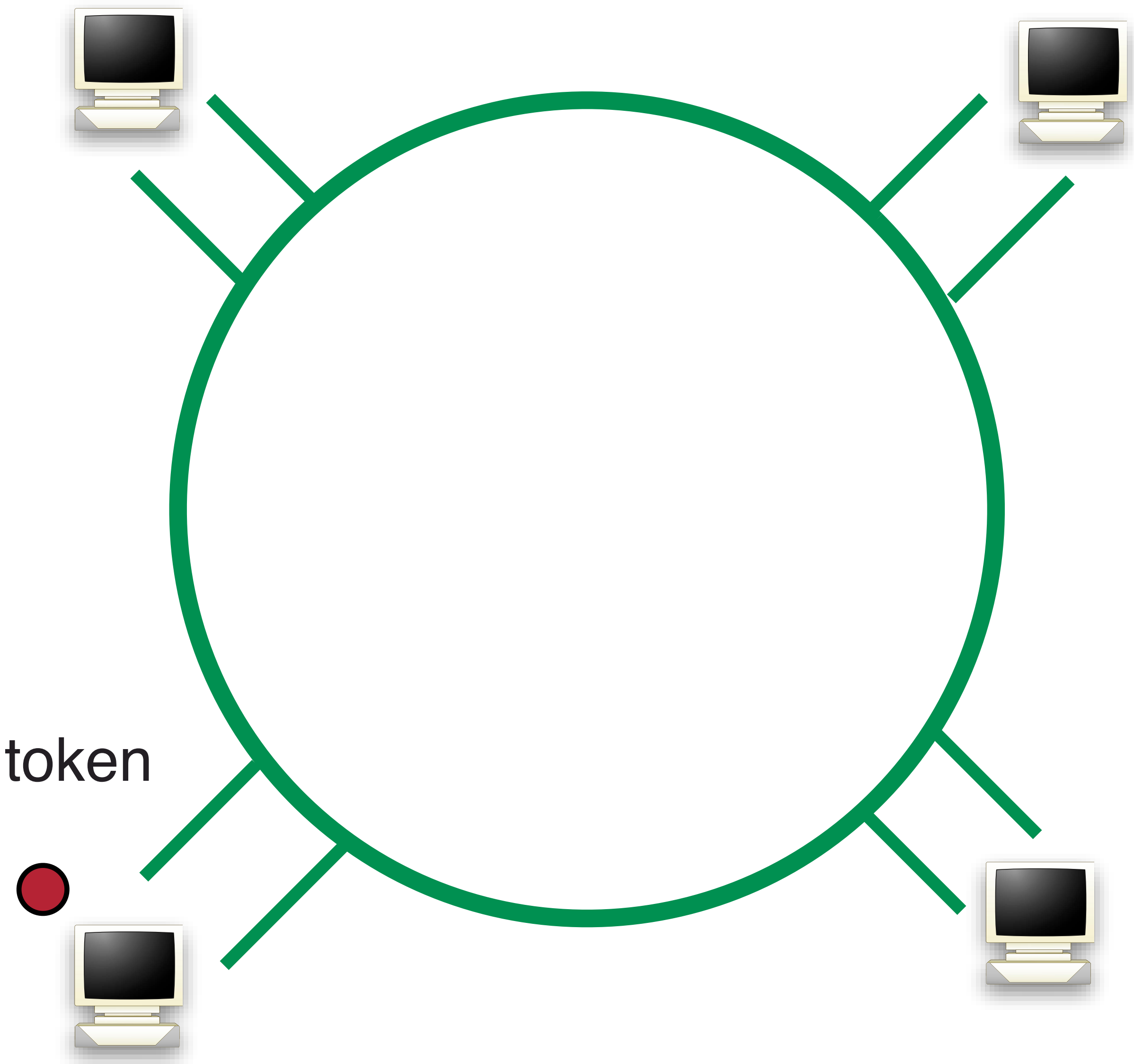
Pass the token

Token passing



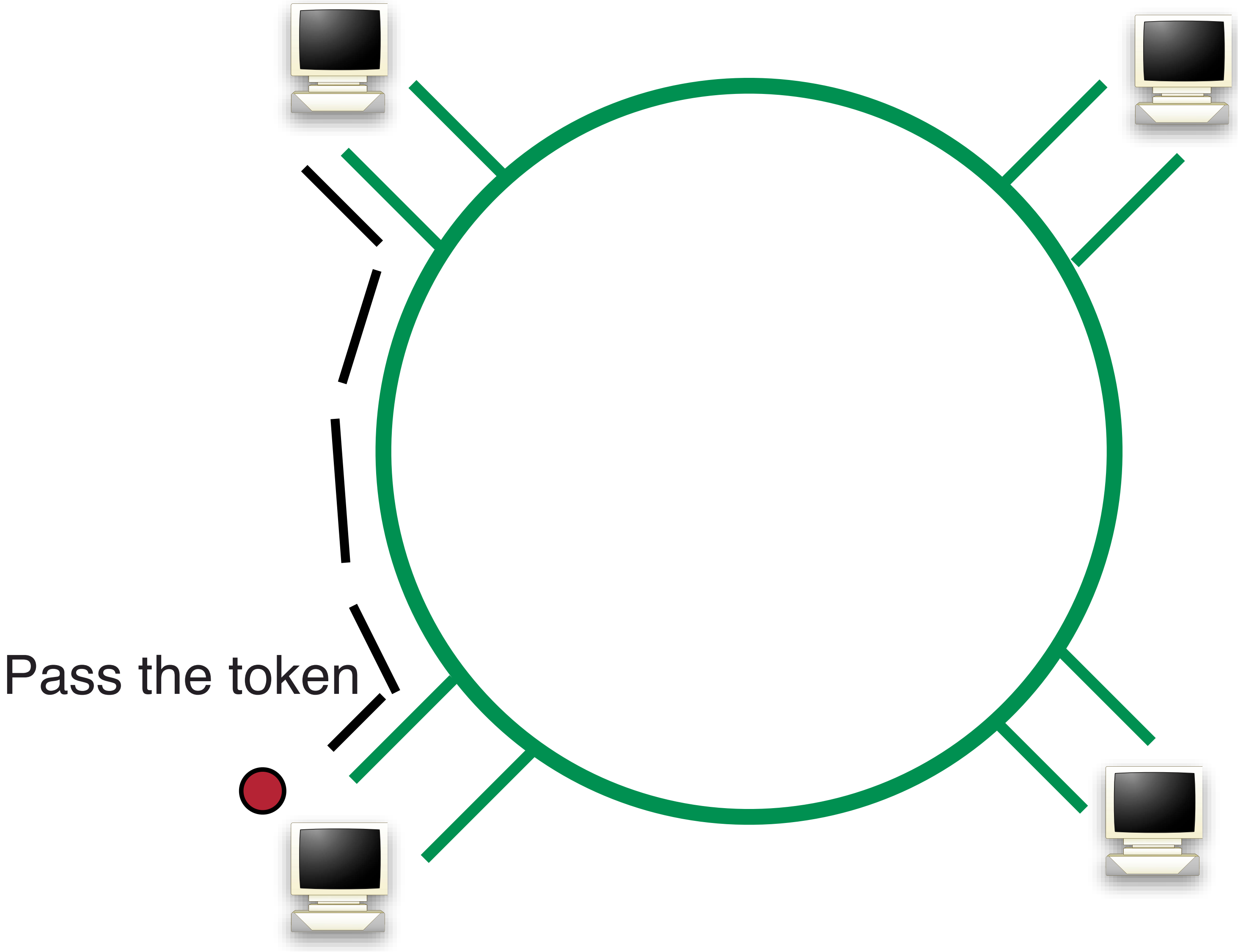
Frame awaiting?

Token passing

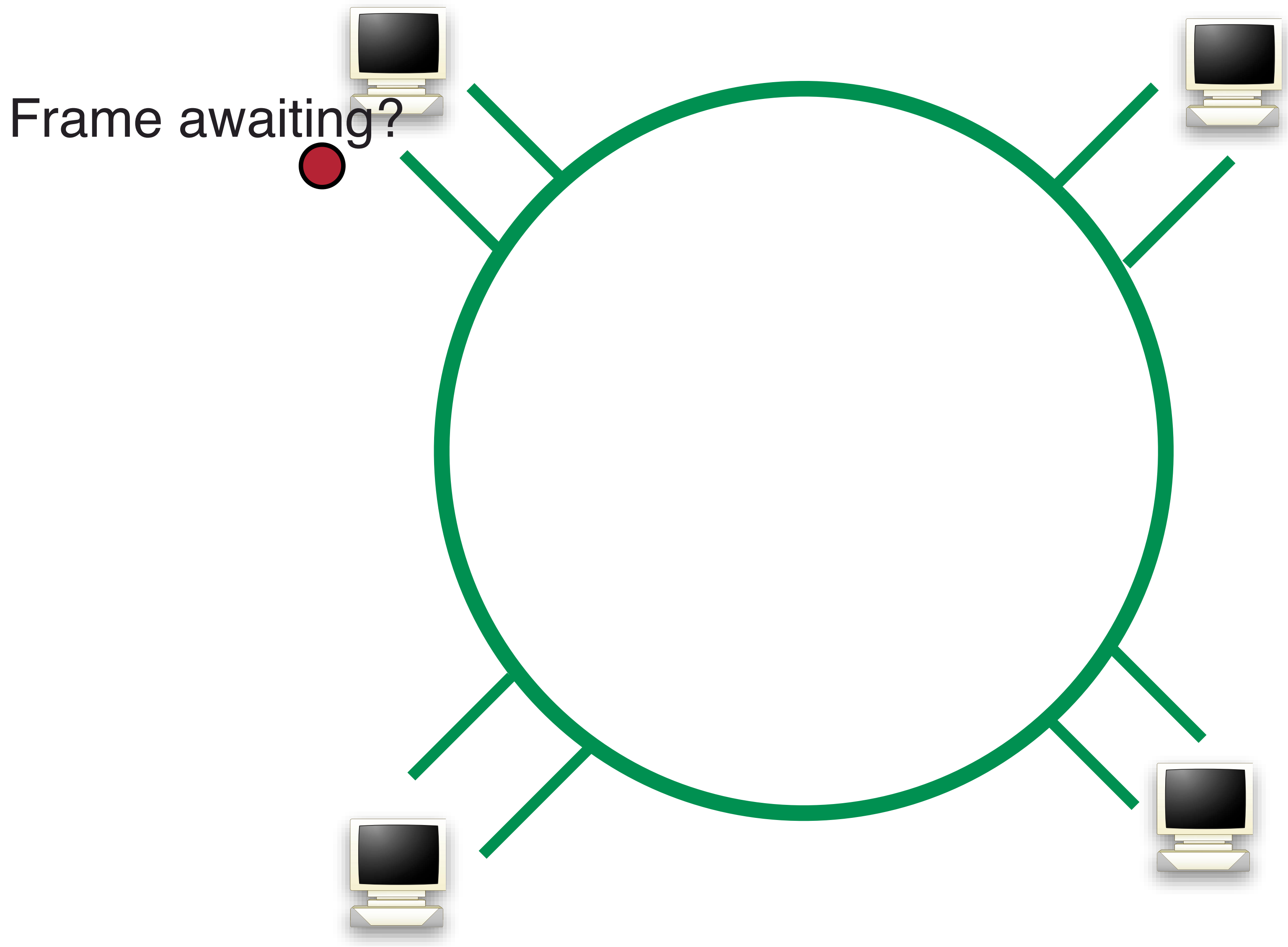


Pass the token

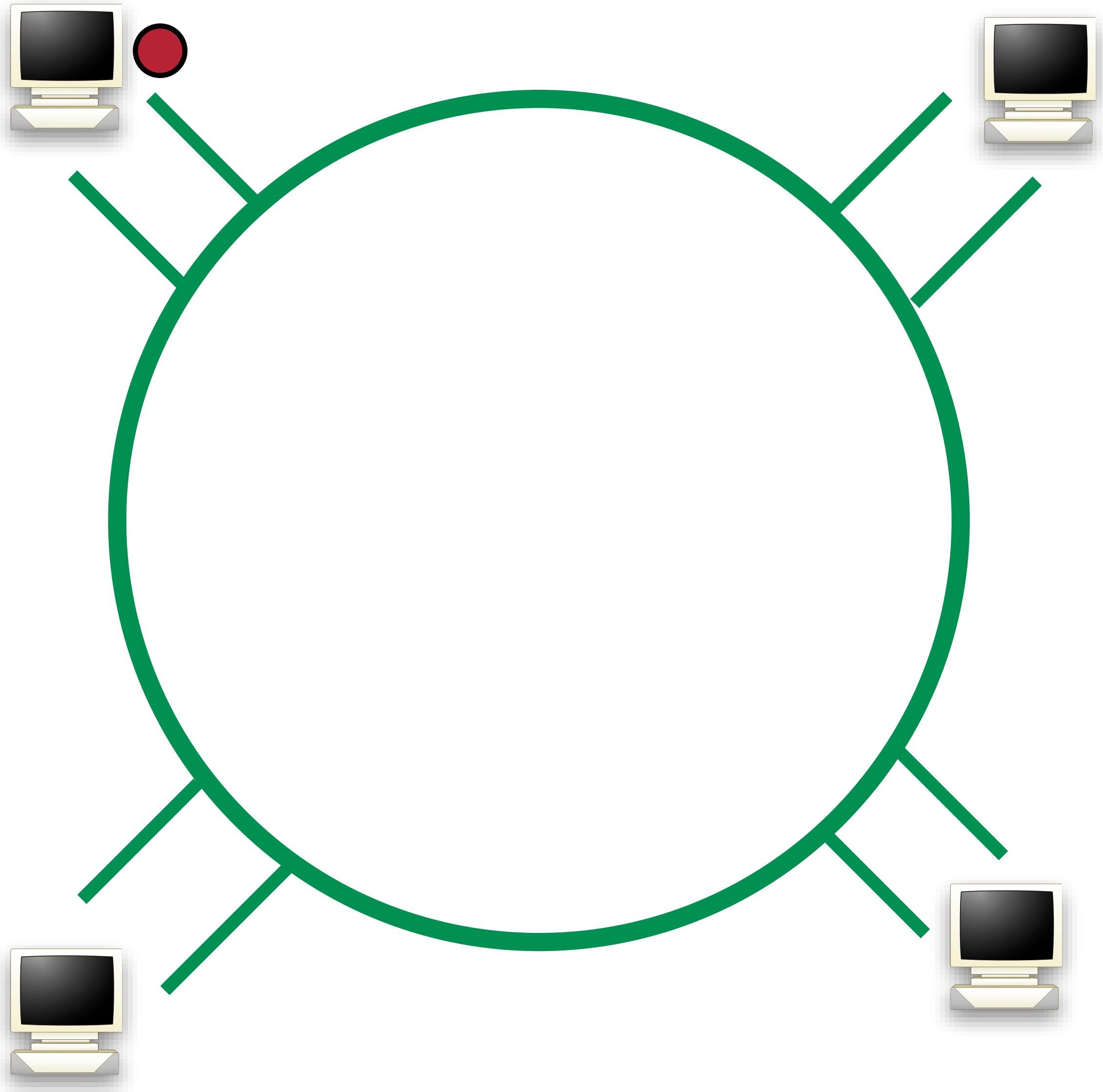
Token passing



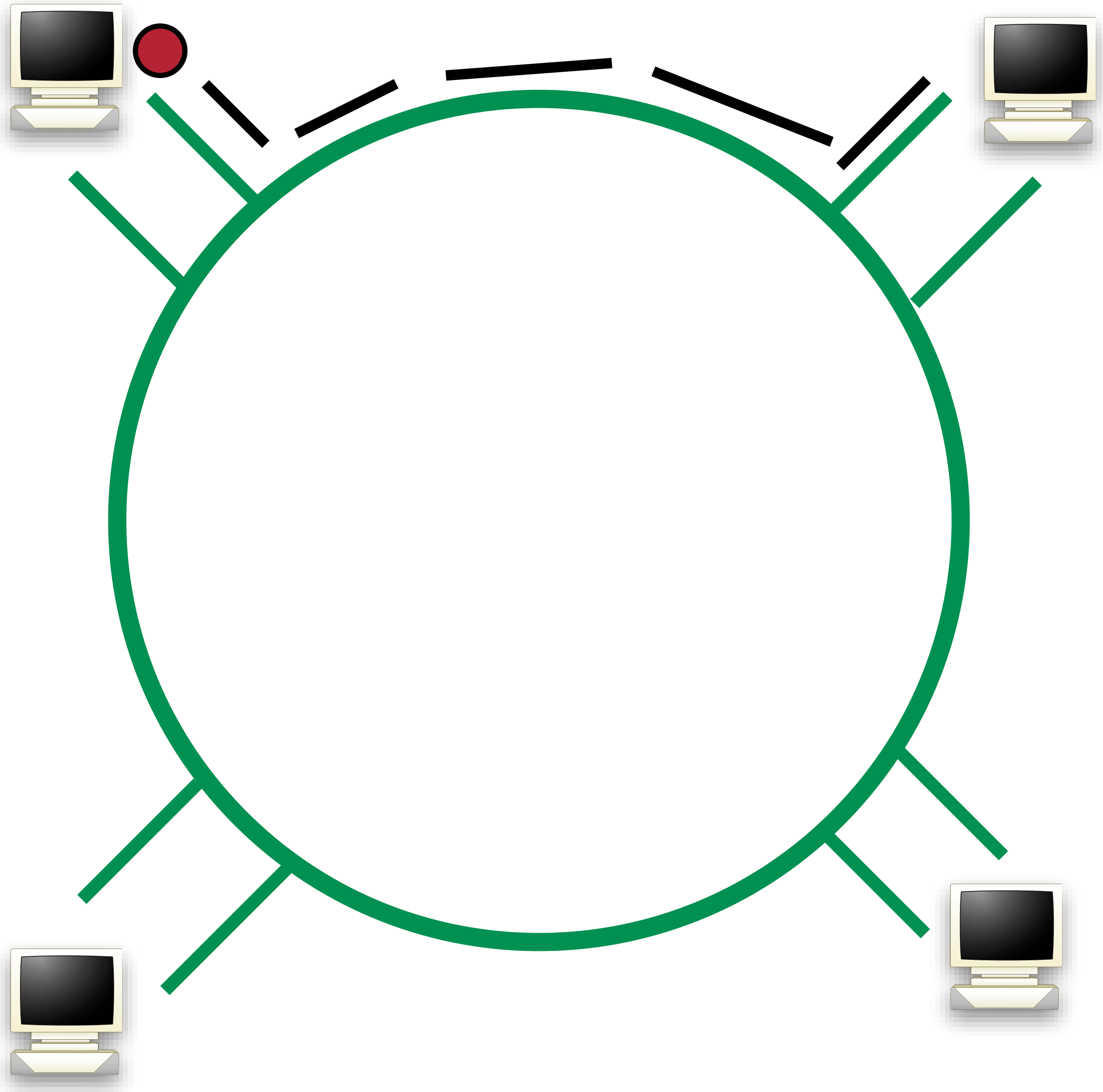
Token passing



Frames circulation

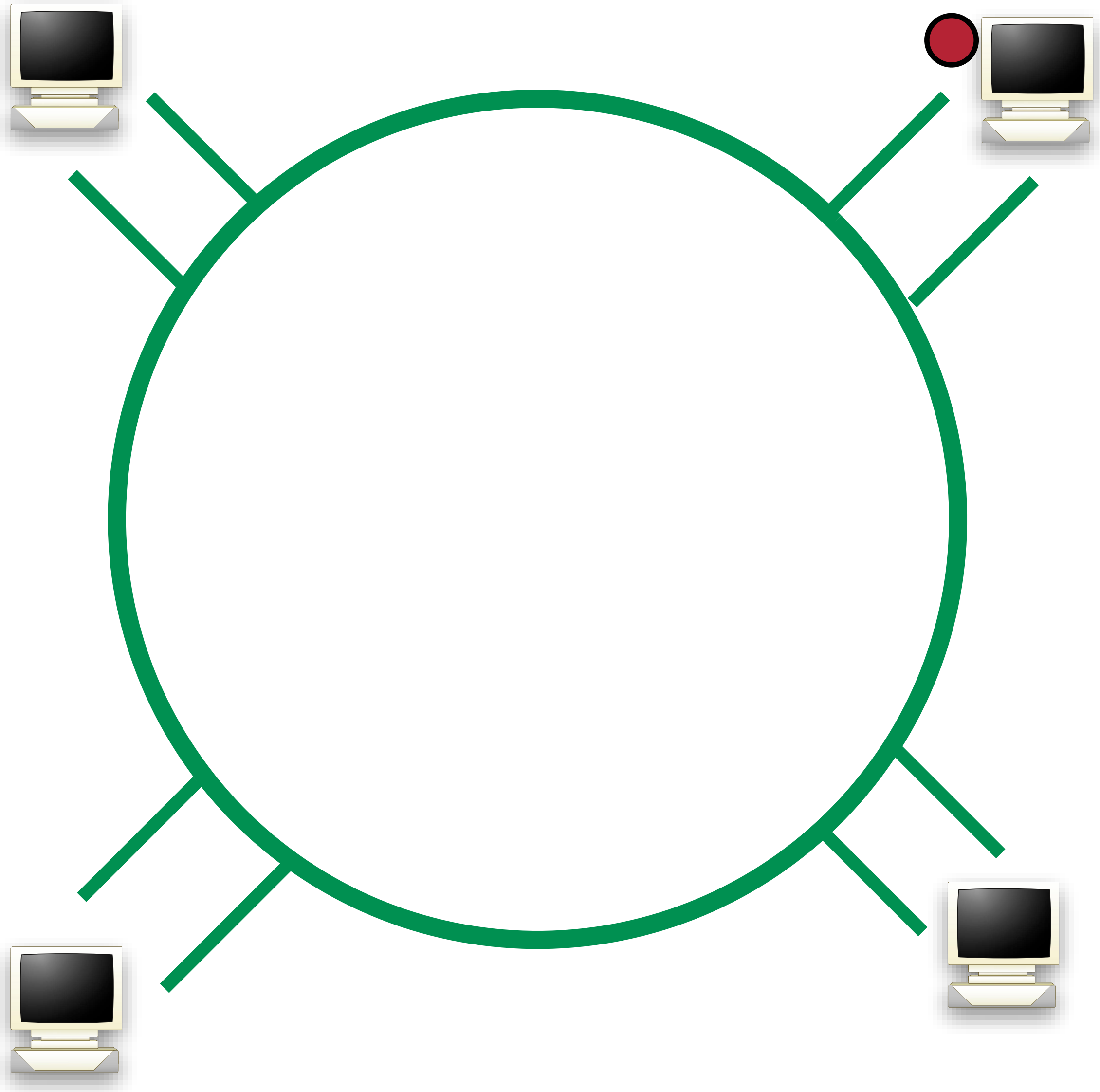


Frames circulation

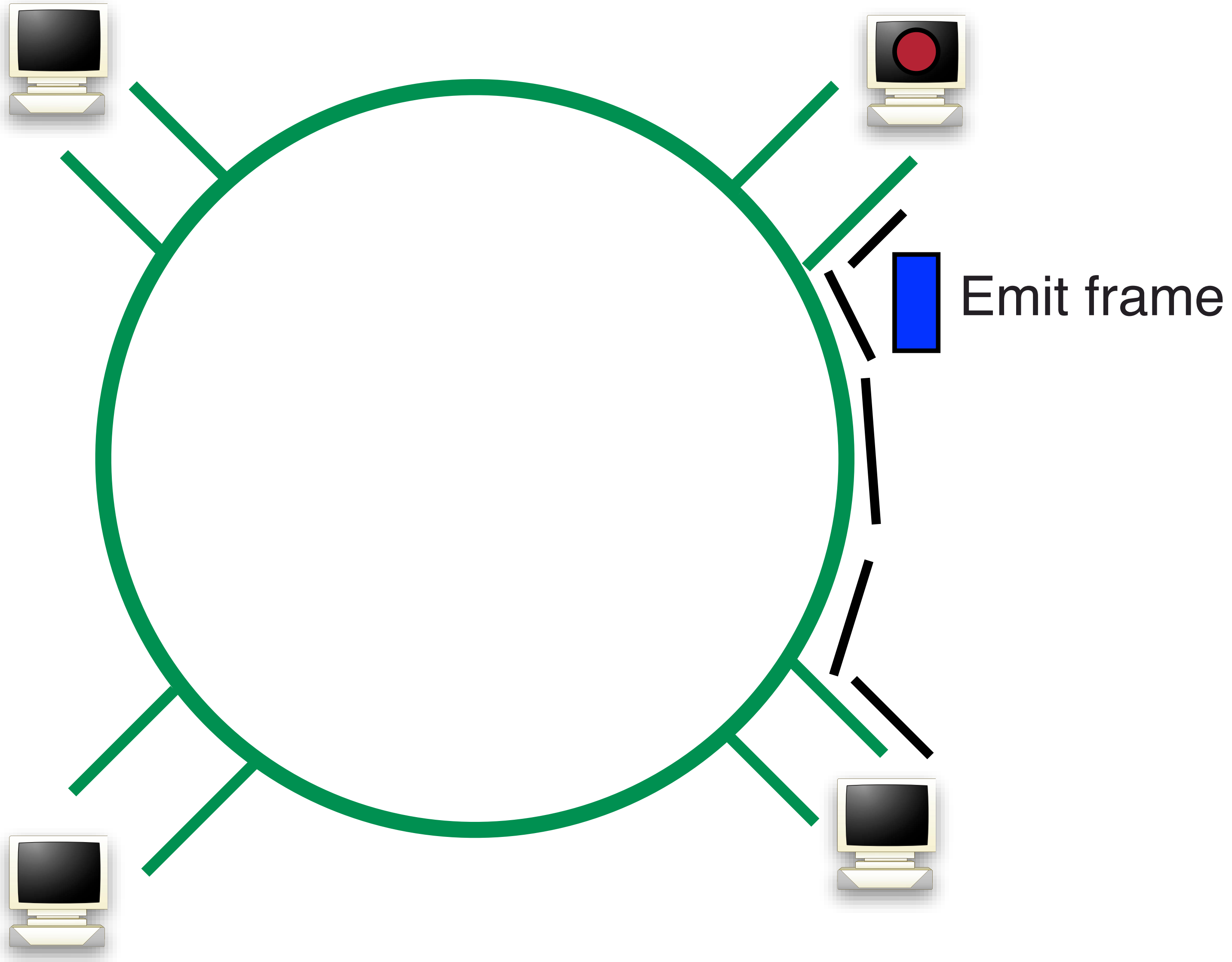


Frames circulation

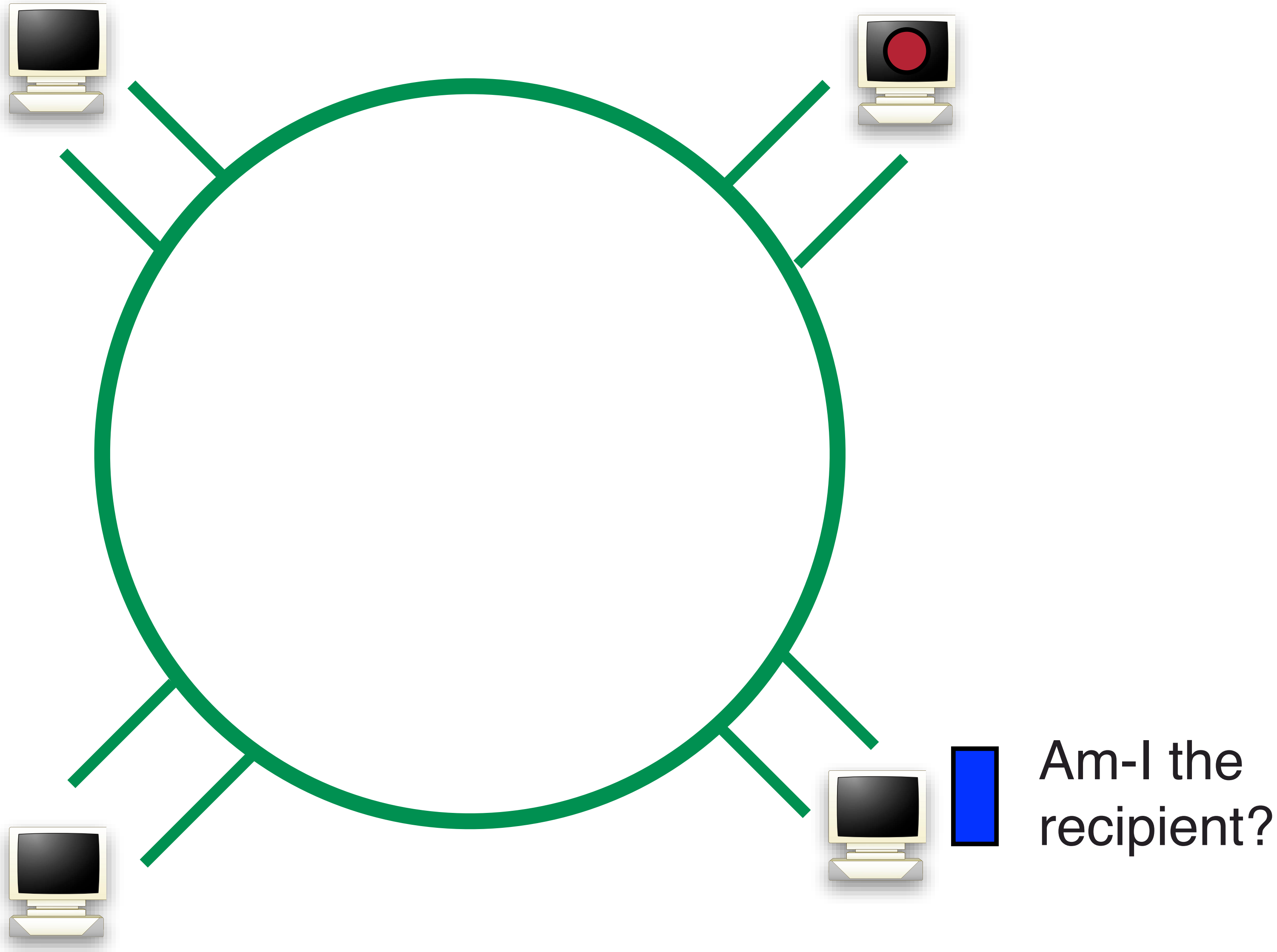
Frame awaiting?



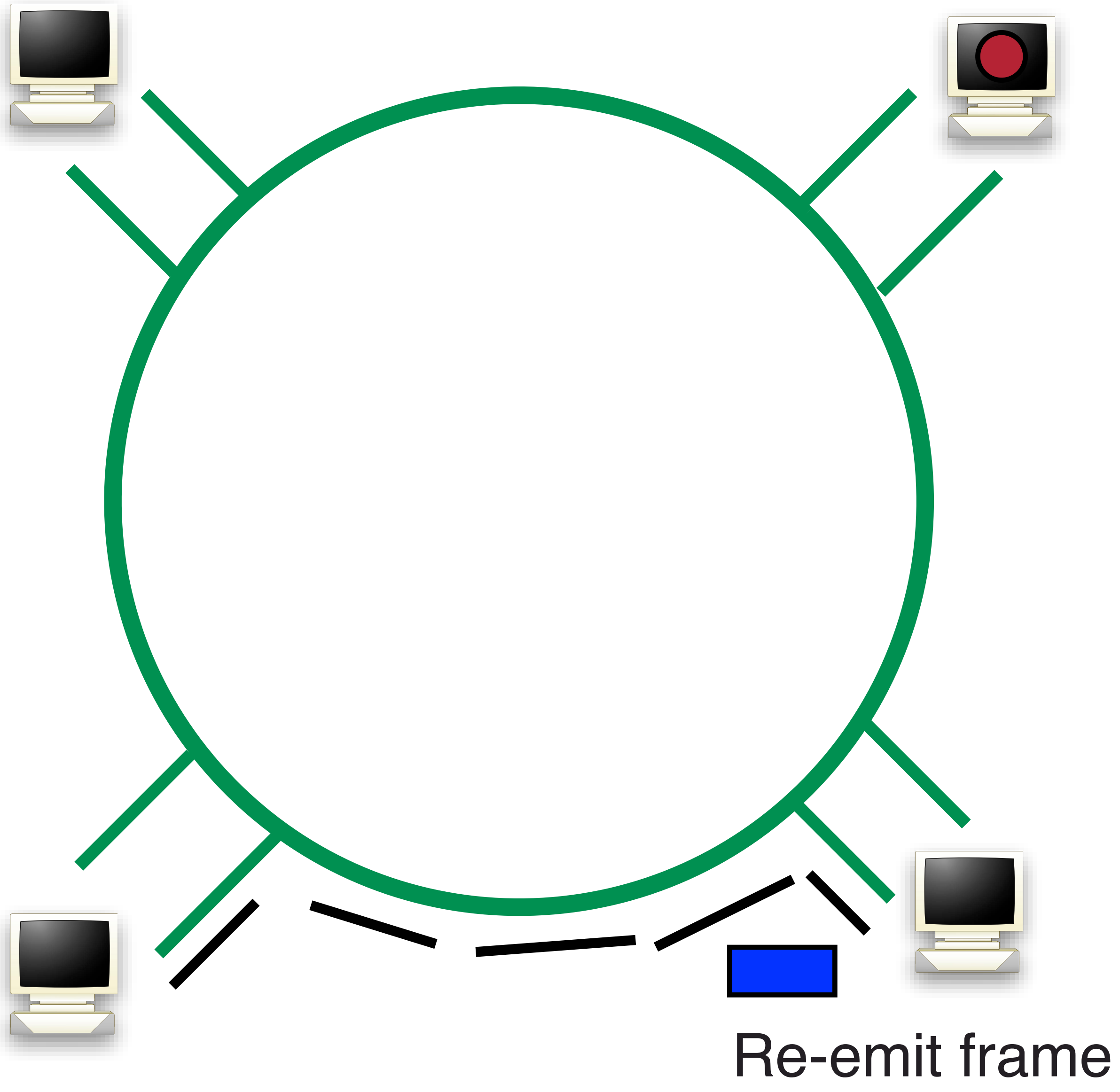
Frames circulation



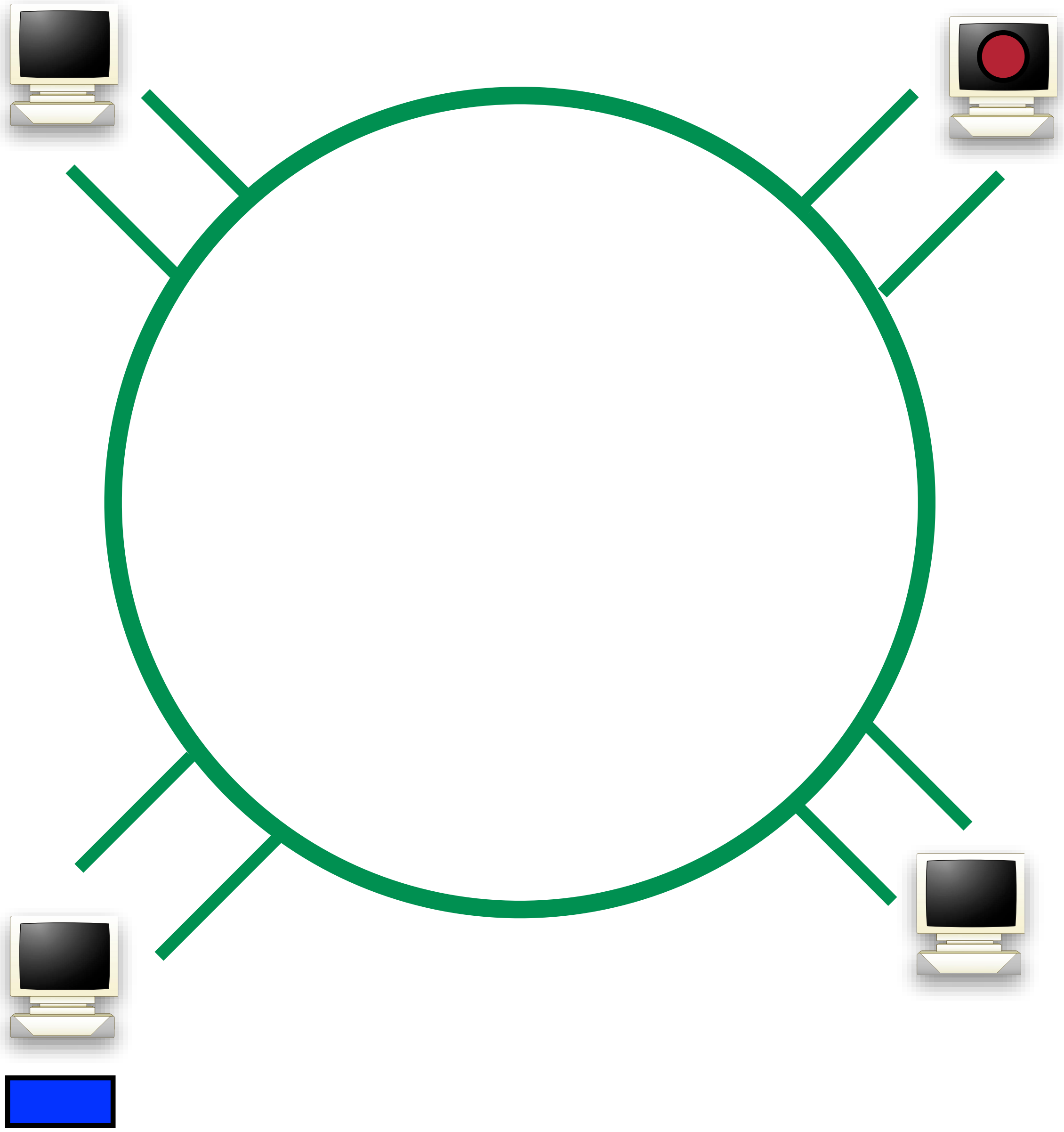
Frames circulation



Frames circulation

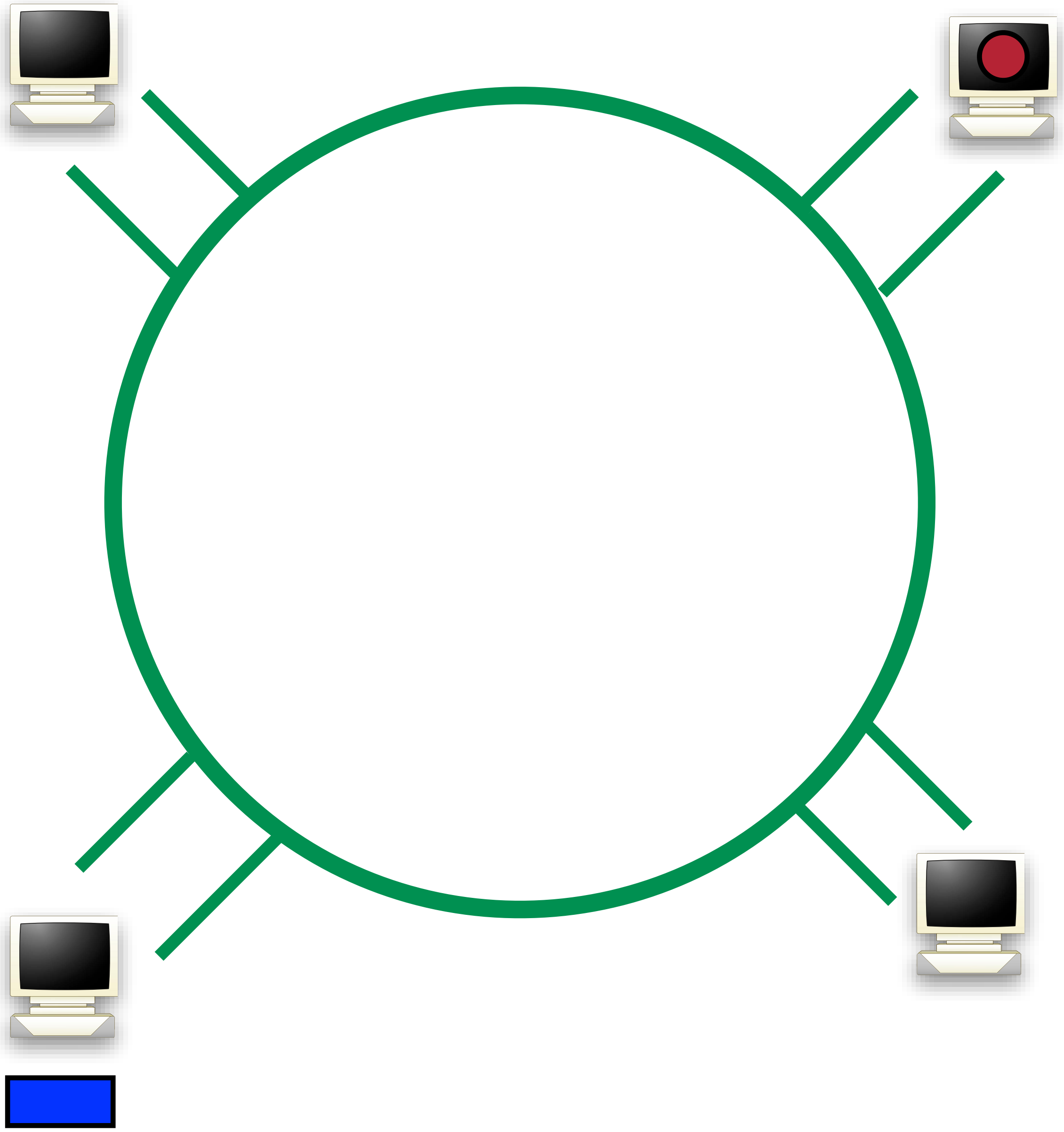


Frames circulation



Frame reception

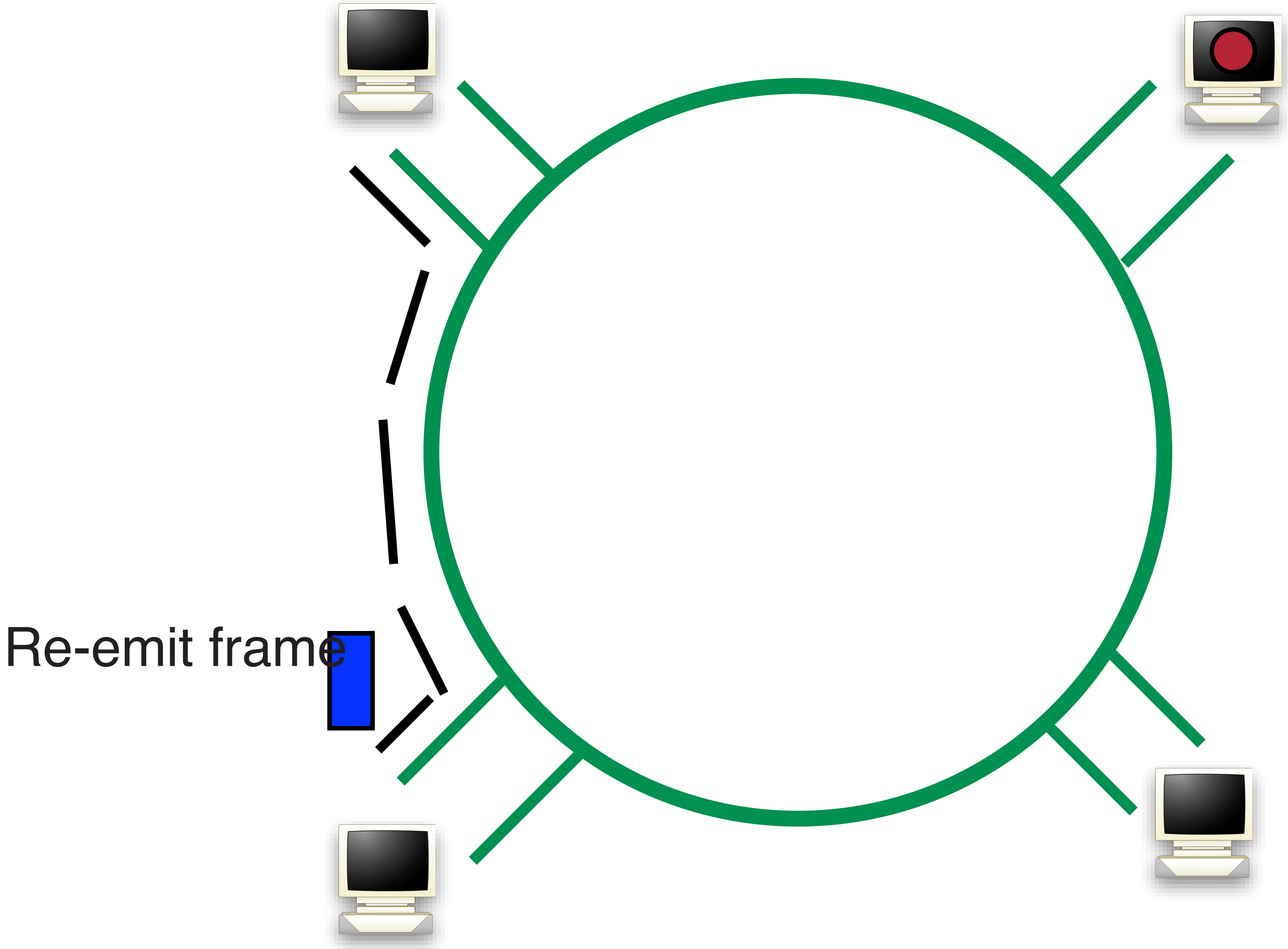
Frames circulation



Should we destroy the frame?

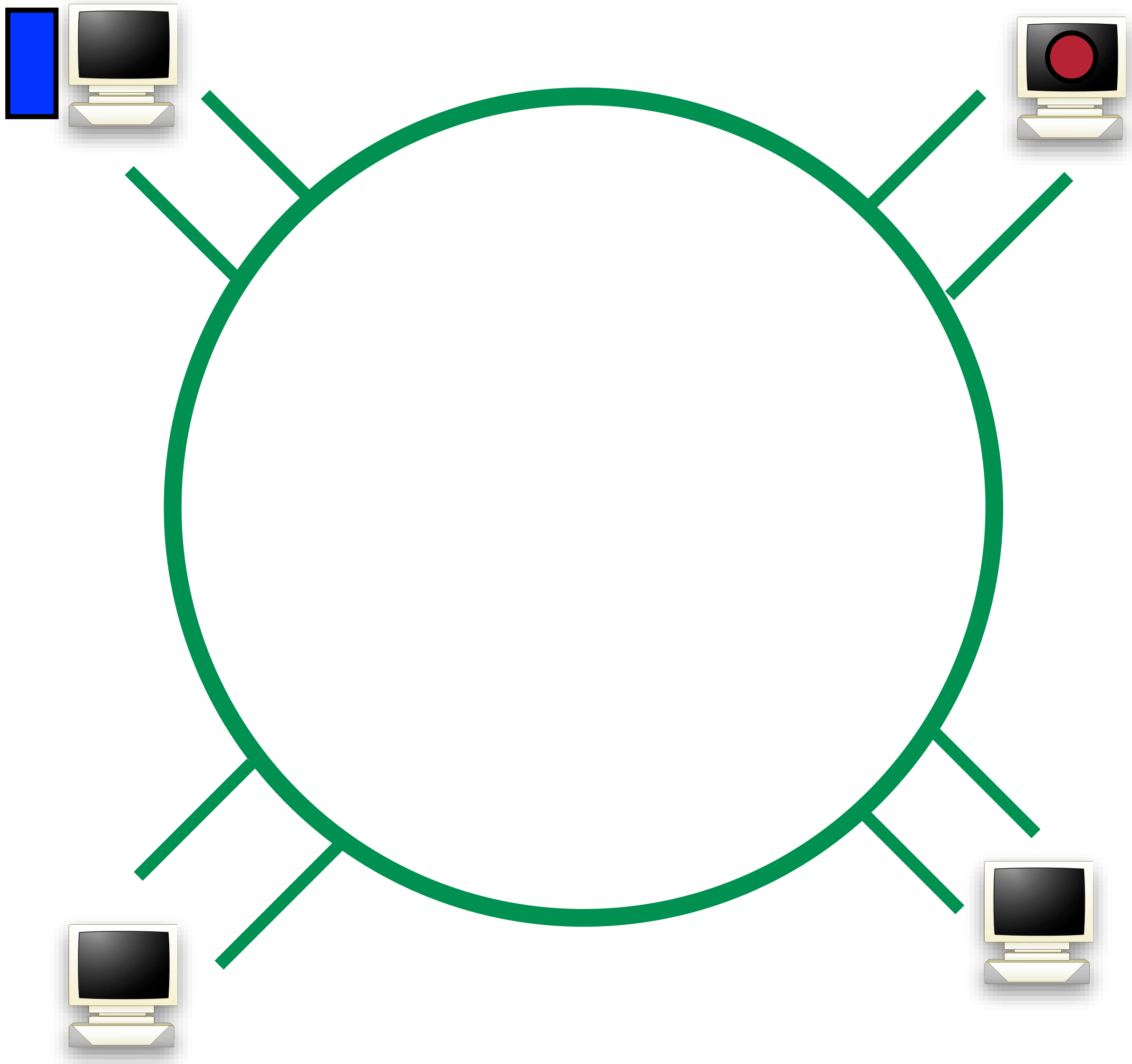
Frame reception

Frames circulation



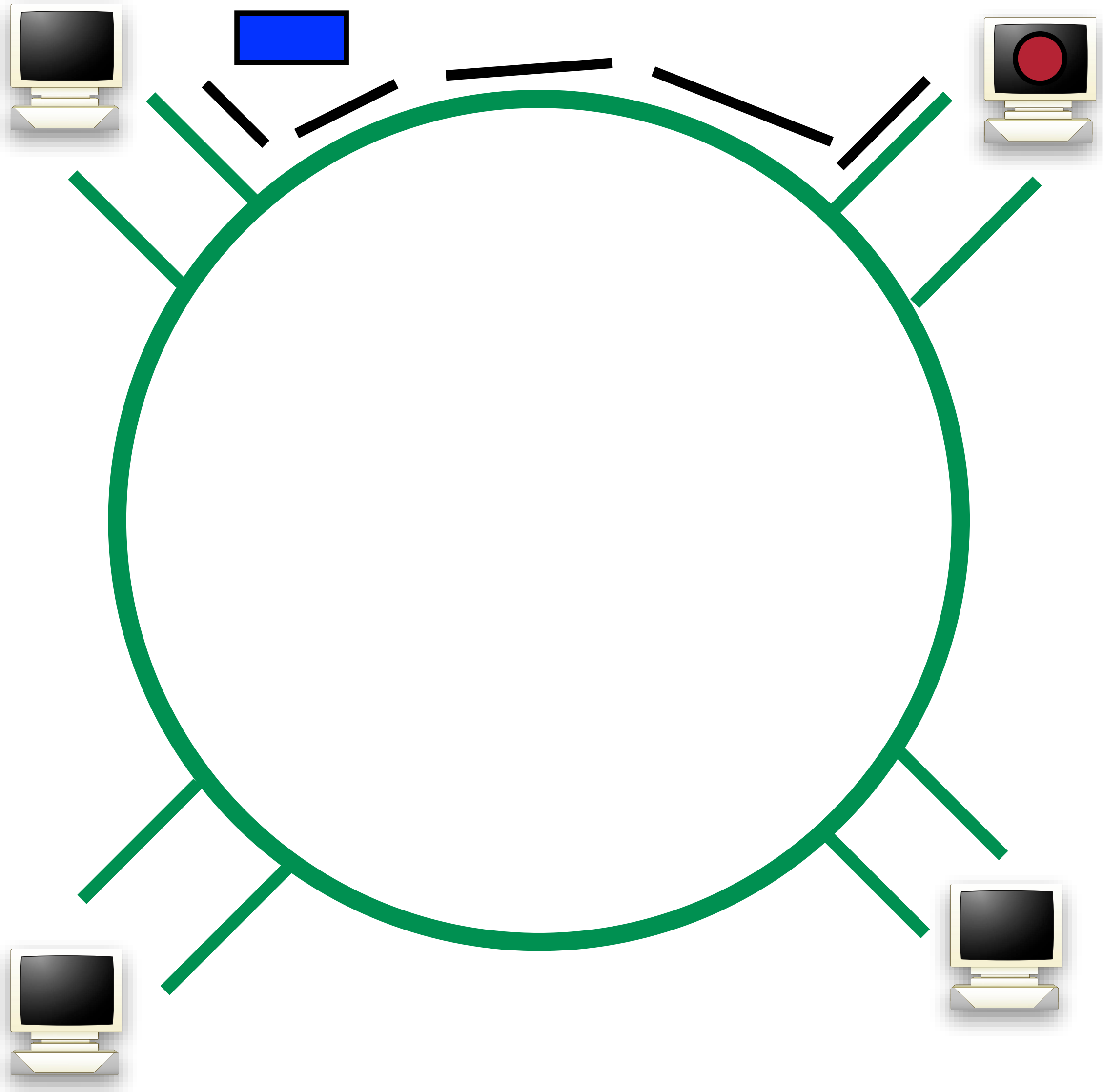
Frames circulation

Am-I the recipient?

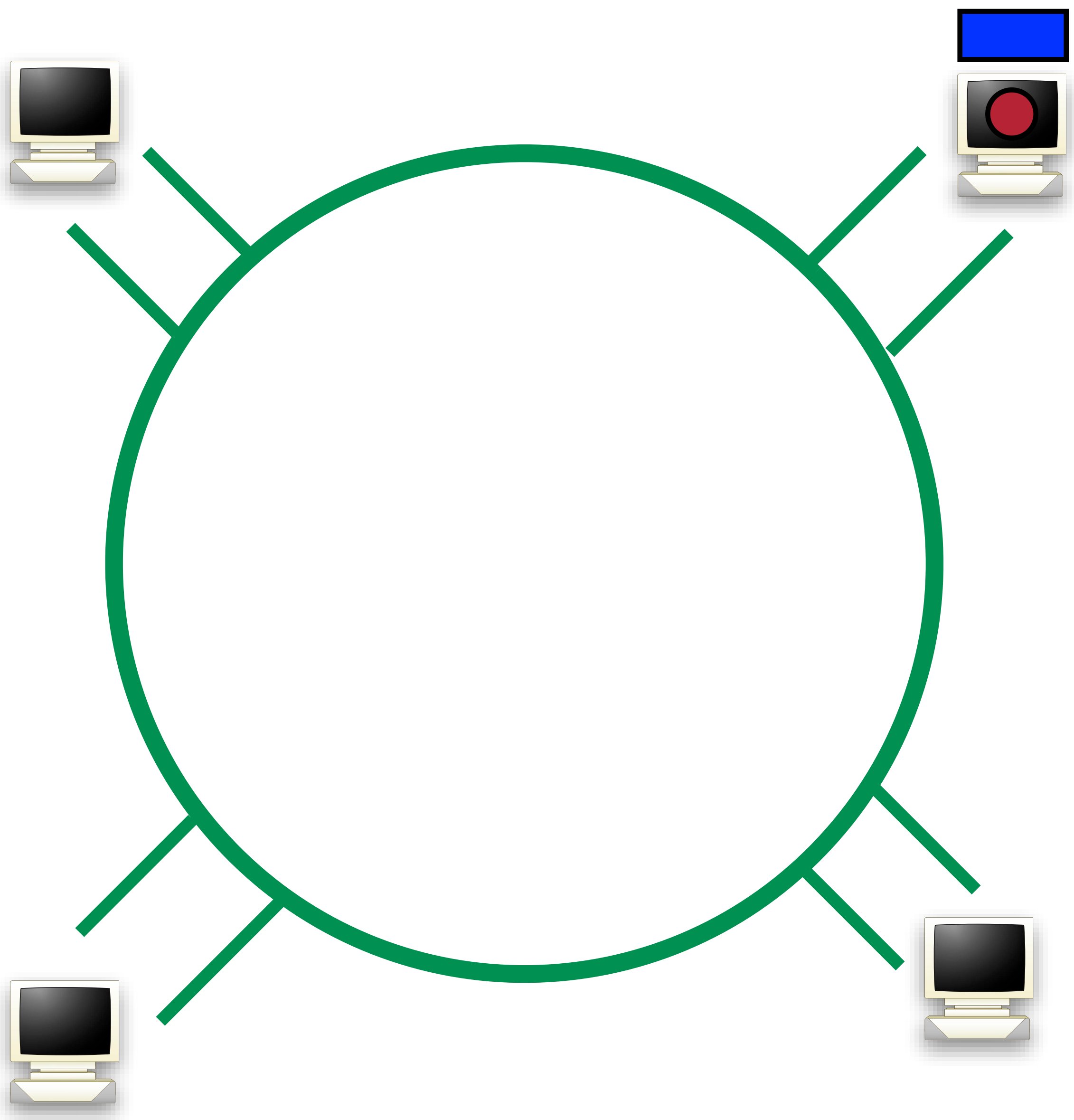


Frames circulation

Re-emit frame

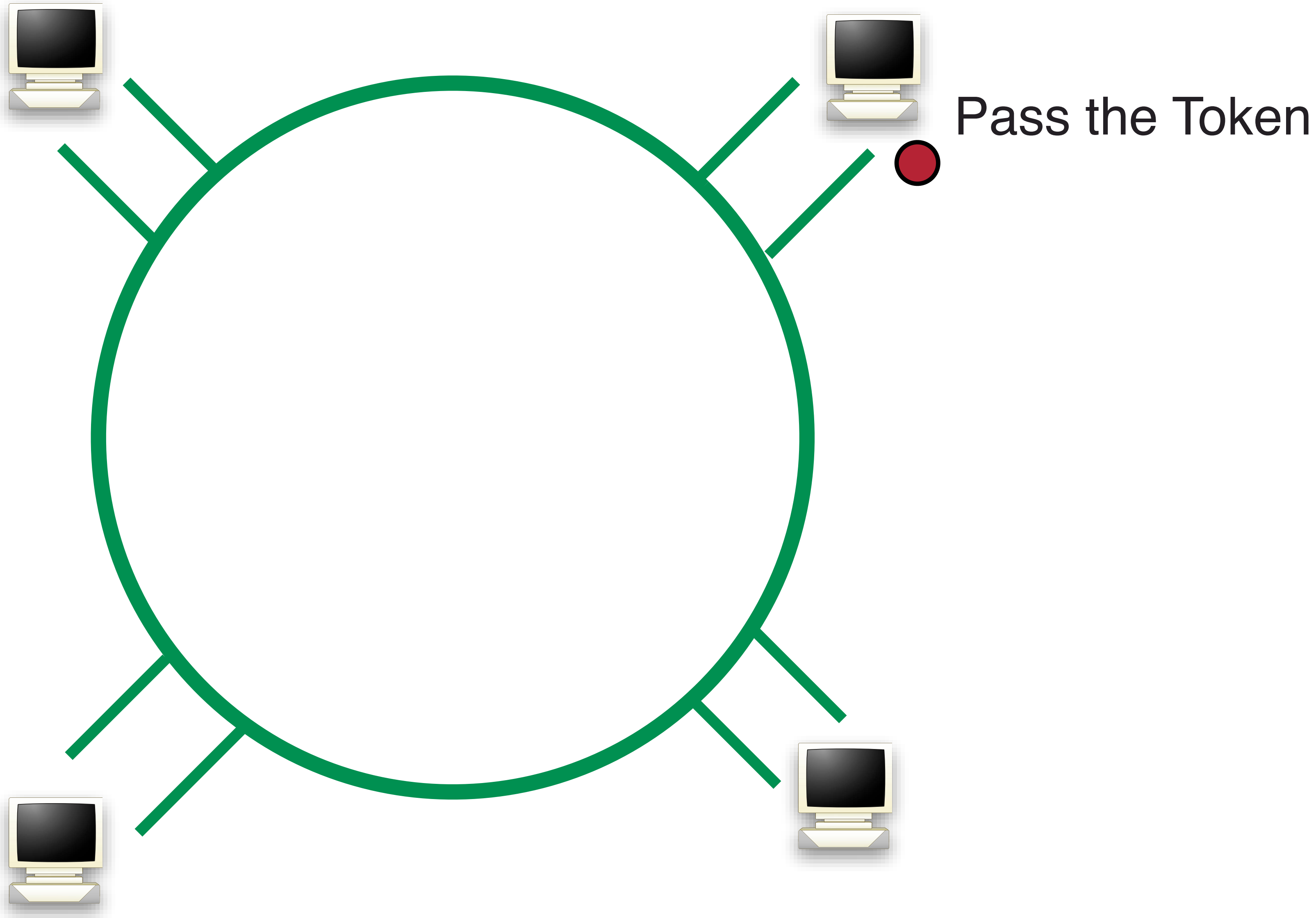


Frames circulation

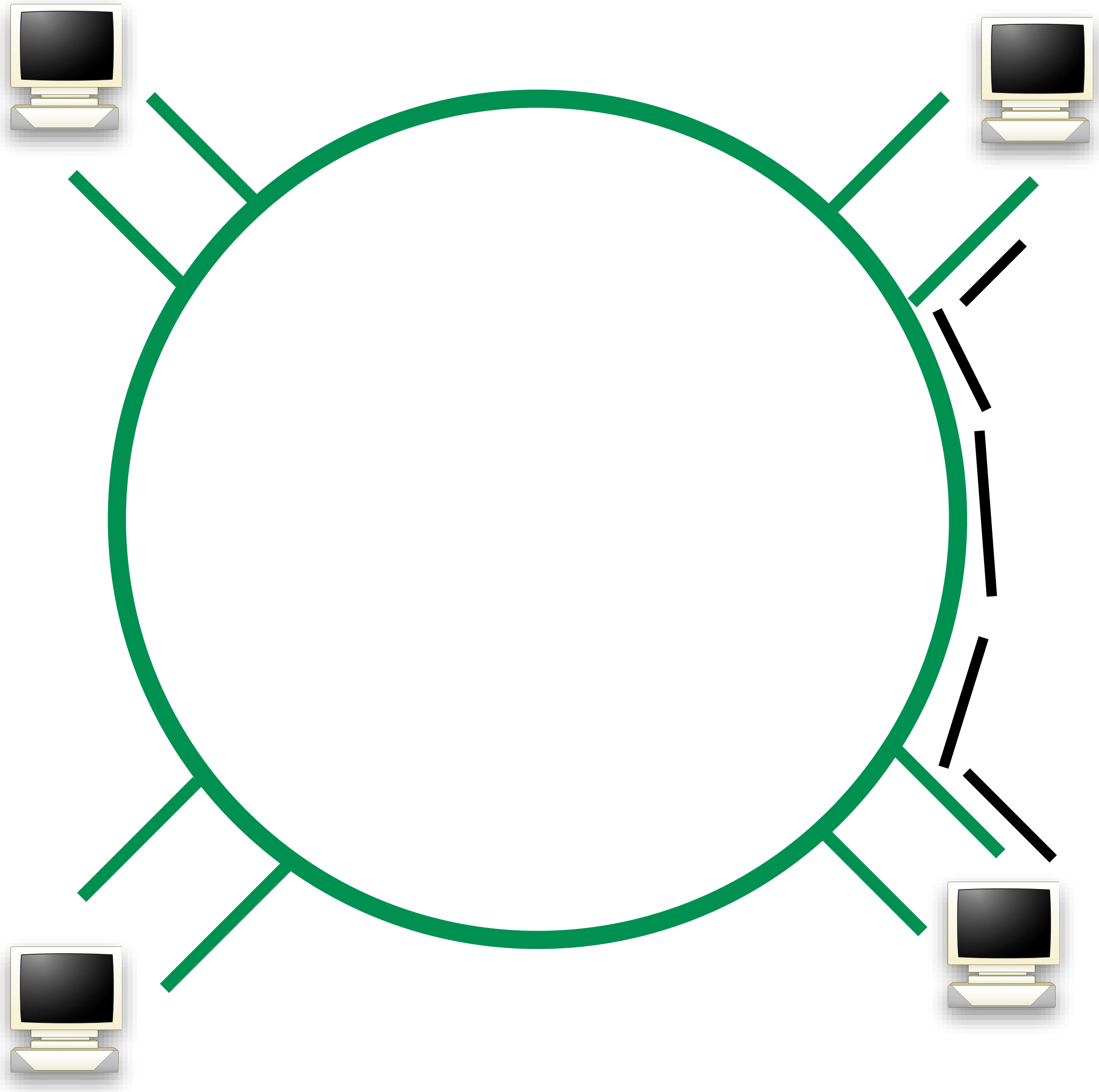


Frame acknowledged ⇒ destruction

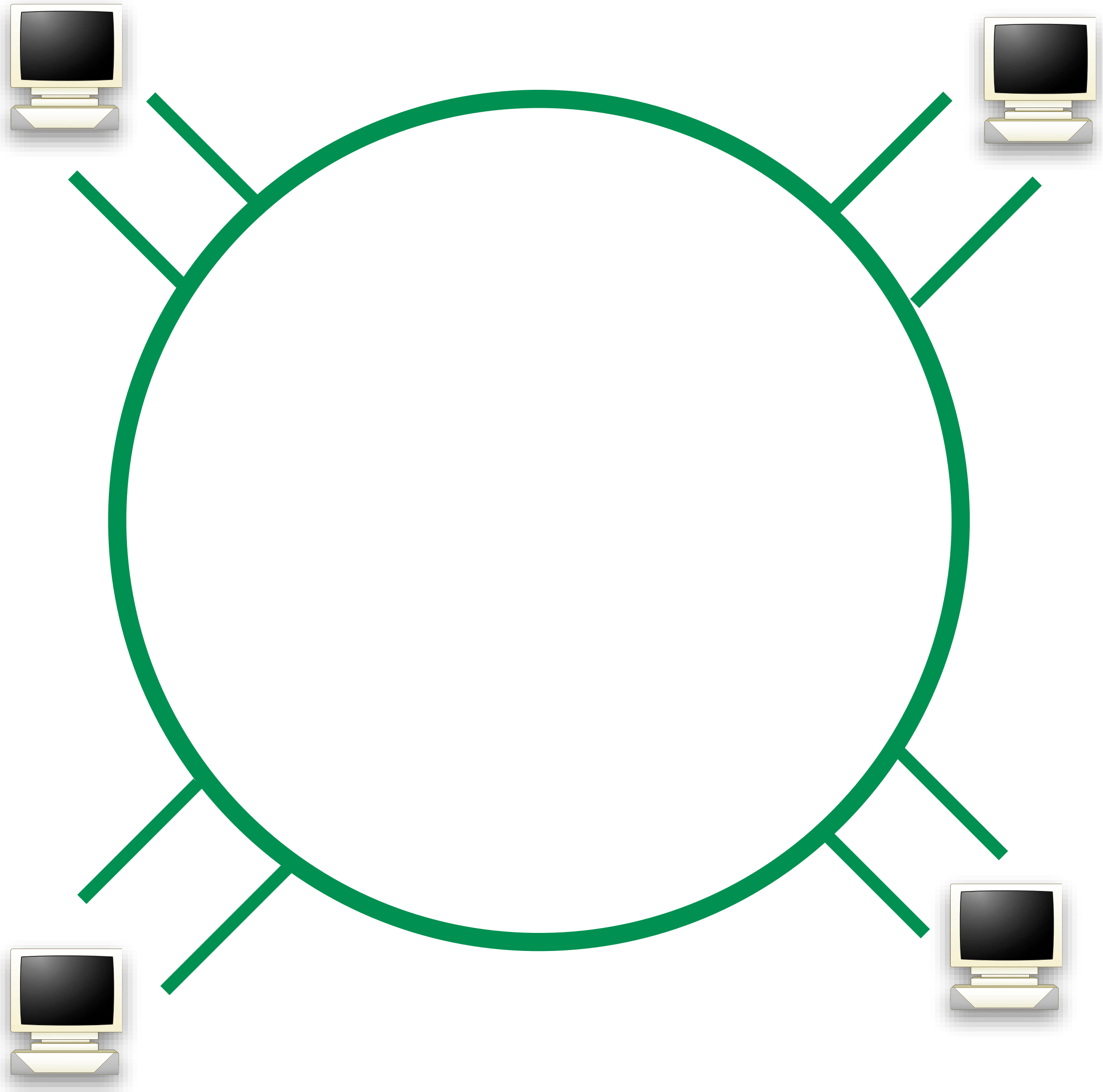
Frames circulation



Frames circulation



Frames circulation



● Frame awaiting?

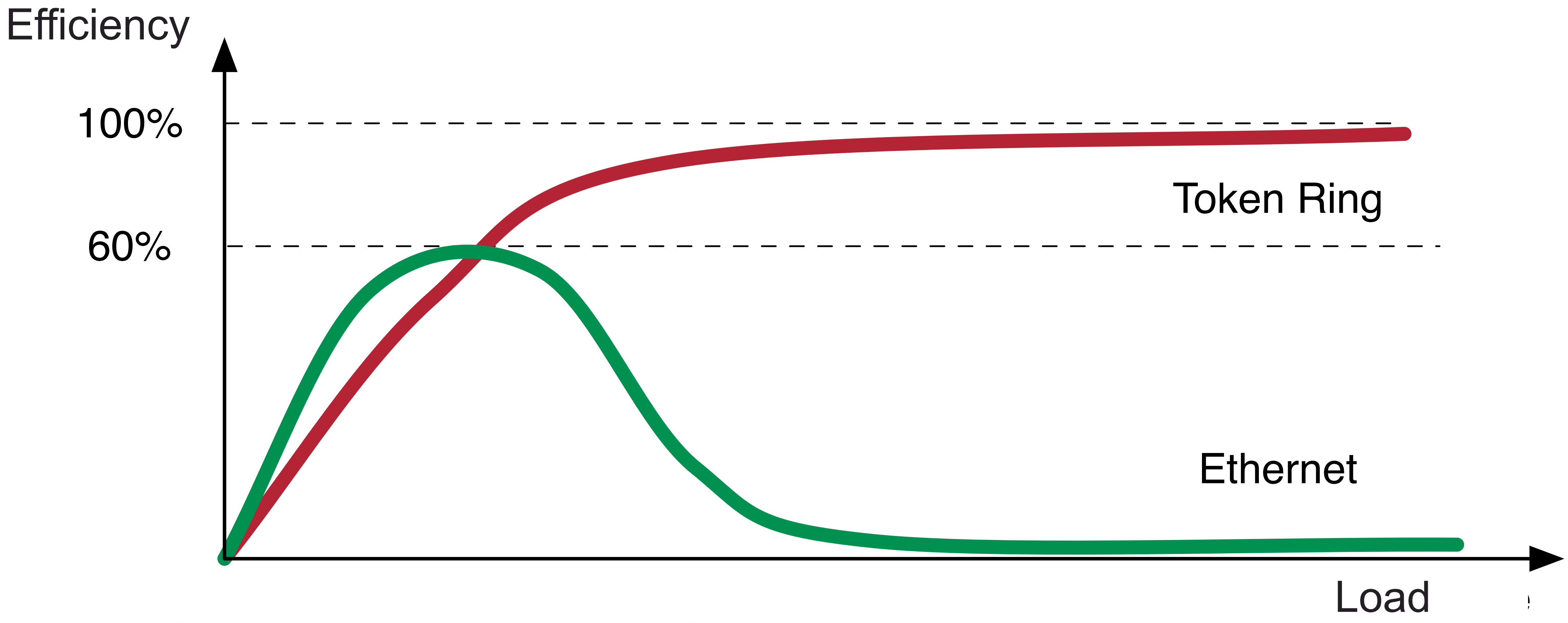
Ring management

- **Initialization**
  - A token is necessary to start the transmissions, who generates it?
    - Use of a particular node: the monitor

- **Device failure**
  - The ring is cut if one station fails
    - How to detect stations failures?
    - Previous and next stations send the traffic in the opposite direction
  - If the station possessed the token when it failed?
    - The monitor is in charge of re-generating the token
  - What if the monitor fails?
    - Election / re-election of a monitor among the stations

Ethernet-Token Ring comparison

- Performance calculation: see exercises session
- Low-load performance reduced because of token passing
- Scalability much better than Ethernet
  - Constant overhead, no collisions


Efficiency
Token Ring
Ethernet
Load