



Principles of modern LAN design and operation

Guido Marchetto

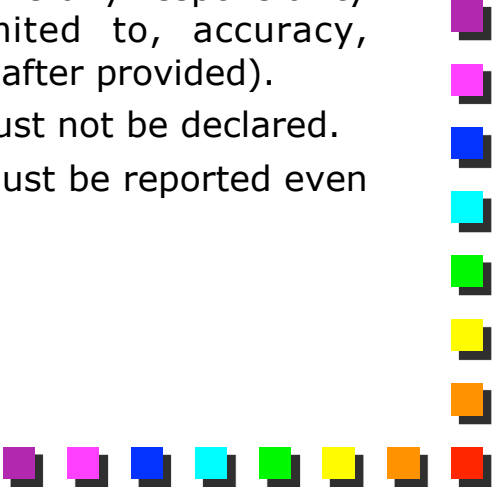
Fulvio Riso

Politecnico di Torino






Copyright notice

- This set of transparencies, hereinafter referred to as slides, is protected by copyright laws and provisions of International Treaties. The title and copyright regarding the slides (including, but not limited to, each and every image, photography, animation, video, audio, music and text) are property of the authors specified on page 1.
 - The slides may be reproduced and used freely by research institutes, schools and Universities for non-profit, institutional purposes. In such cases, no authorization is requested.
 - Any total or partial use or reproduction (including, but not limited to, reproduction on magnetic media, computer networks, and printed reproduction) is forbidden, unless explicitly authorized by the authors by means of written license.
 - Information included in these slides is deemed as accurate at the date of publication. Such information is supplied for merely educational purposes and may not be used in designing systems, products, networks, etc. In any case, these slides are subject to changes without any previous notice. The authors do not assume any responsibility for the contents of these slides (including, but not limited to, accuracy, completeness, enforceability, updated-ness of information hereinafter provided).
 - In any case, accordance with information hereinafter included must not be declared.
 - In any case, this copyright notice must never be removed and must be reported even in partial uses.
- 




A view from history (1)

- Wide Area Networks appeared first
 - '60s
 - A few mainframes; necessity to connect to them from remote
 - Partition their expensive cost between more entities
 - Local Area Networks appeared later
 - End 70's, beginning '80s
 - Minicomputers (and later PCs appeared)
 - Cost was low enough so that it was no longer needed to access a remote mainframe
 - Sharing resources between small workgroups (e.g., departments)
 - Mainframe still used, but for different purposes (e.g., scientific simulations)
- 

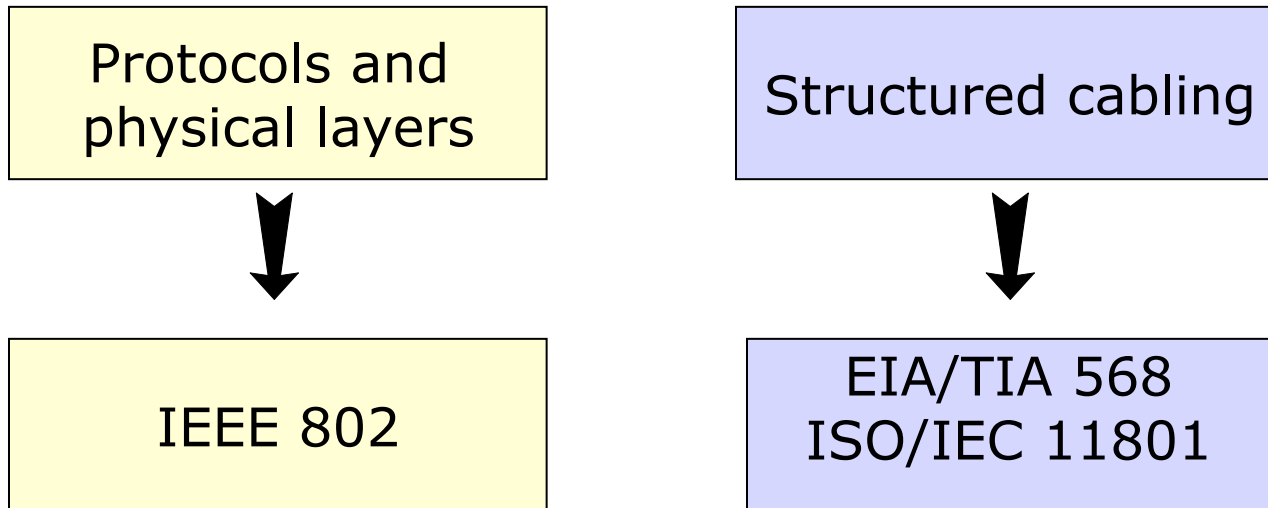


A view from history (2)

- At the beginning, WAN and LANs evolved independently
 - Different protocols, engineered by different vendors for different purposes
 - Decnet, SNA, IP
 - Novell, Banyan Vineis, NetBeui
 - Later we tried to connect LANs to WANs
 - Progressive overlapping of functions/protocols
 - One winner: IP
 - Some overlaps still remain (e.g., addressing)
- 

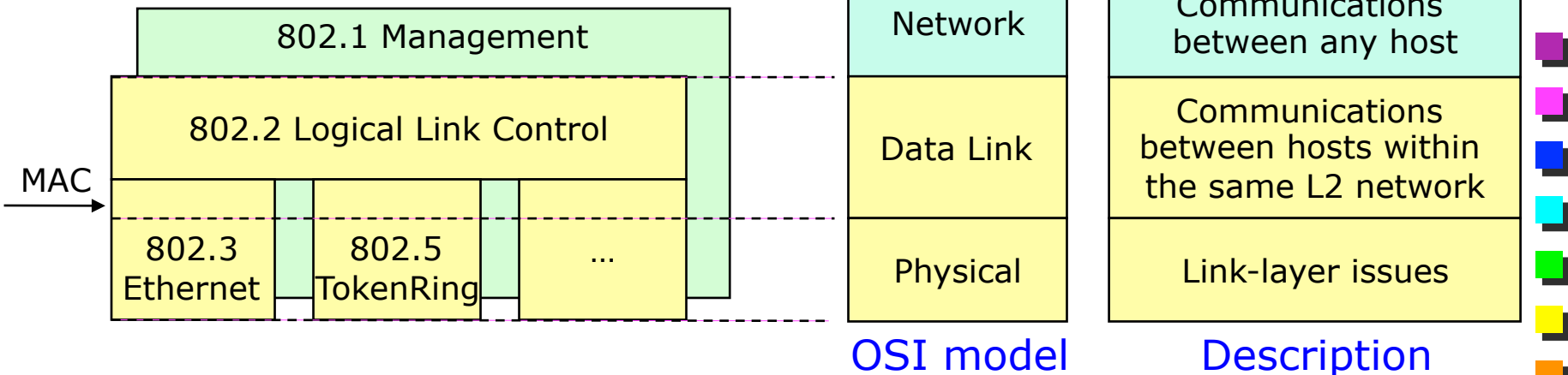


LAN important standards



LANs: IEEE and OSI models

- 802.1: Higher Layers and Management
- Logical Link Control sublayer
- Medium Access Control sublayer
- Physical layer



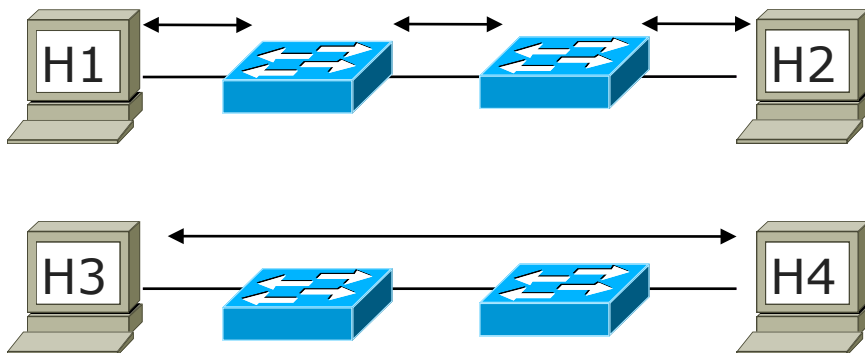
differences between Ethernet (parts of)
and IEEE 802.3 (has LLC)

MAC and LLC sublayers

- MAC sublayer
 - “Medium Access Control” solutions
 - E.g., CSMA/CD
 - Addressing
- LLC sublayer
 - L3 protocol demultiplexing
 - E.g., IPv4, IPv6, ...
 - Advanced features
 - Connection oriented communications at L2
 - Flow control at L2
 - ...

Is LLC useful in modern LANs?

- In practice, sometimes LLC is even not present...
 - E.g., Ethernet DIX avoided LLC at all, although the IEEE version supports LLC (but nobody uses it)
- ... and when is there, most of the features are disabled
 - E.g. WiFi
- No need for those features in current networks
 - E.g., flow control currently done at L4
 - Intermediate devices are simpler (hence faster and cheaper)



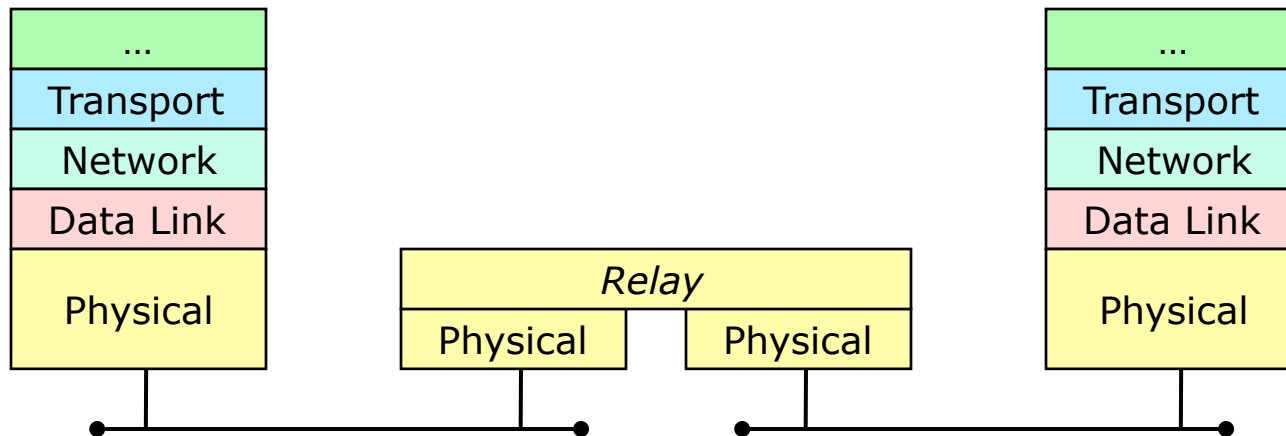


LAN devices in brief

- L1: Repeater 中继器
 - Hub 集线器 碰撞
 - Separate physical domains, same collision domain
 - L2: Bridge 网桥
 - Switch
 - Separate collision domains, same broadcast domain
 - L3: Router
 - L3 switch
 - Separate broadcast domains
 - Not really specific for LANs
 - Not covered in the current slides
- 

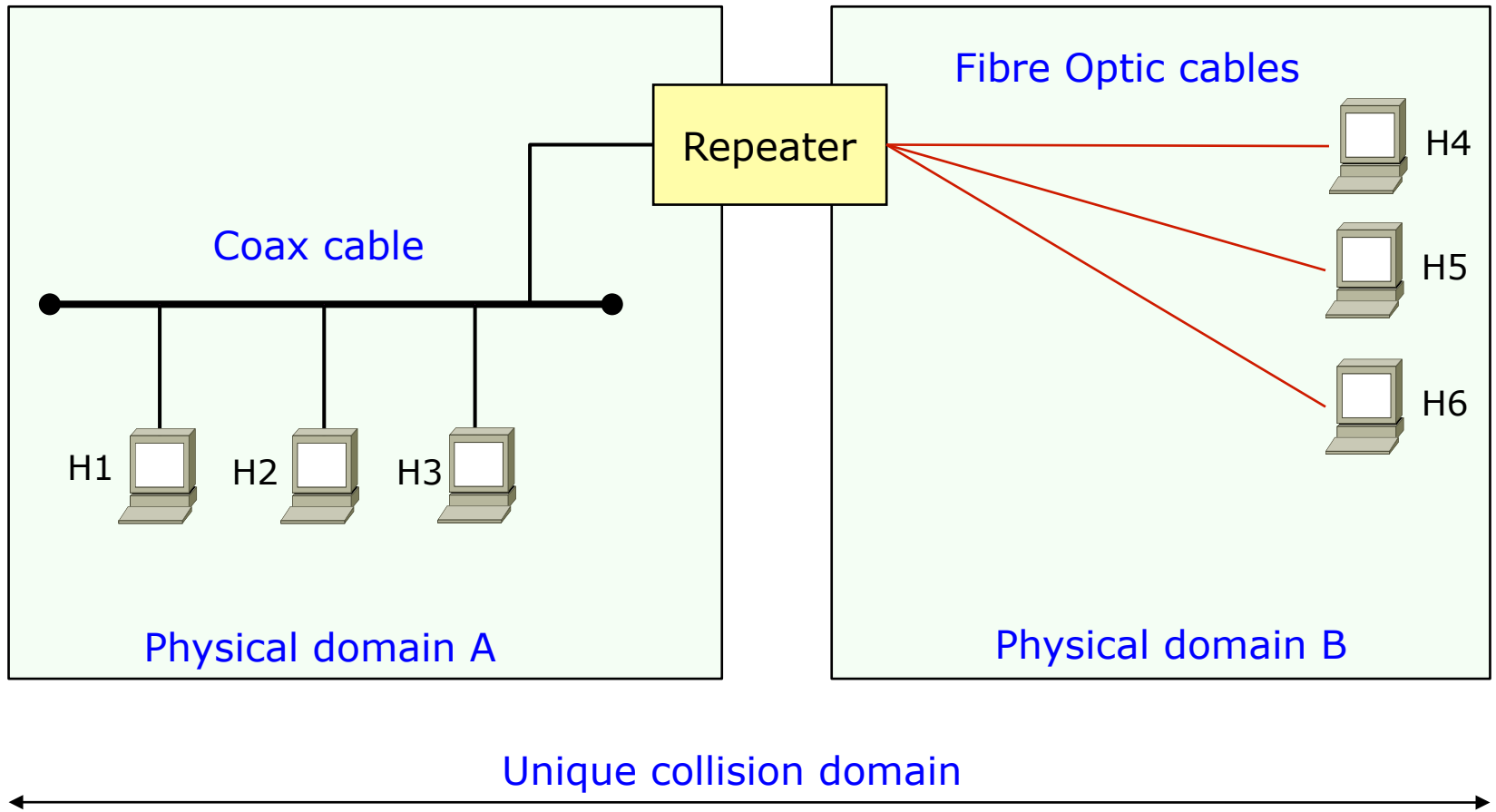
Repeater

- Interconnection at the physical layer
 - Receives and propagates a sequence of bits
- Used for **传播**
 - Interconnecting networks having the **same MAC** 集线器没有MAC地址
 - I.e., all ports must have the same speed
 - E.g., Ethernet 10Mbps fiber to copper
 - Recovering signal degradation (long cables), allowing larger distances



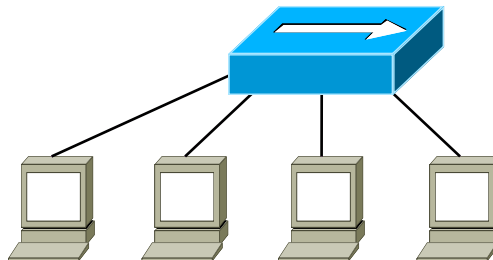
Repeater: example

同轴电缆转光纤



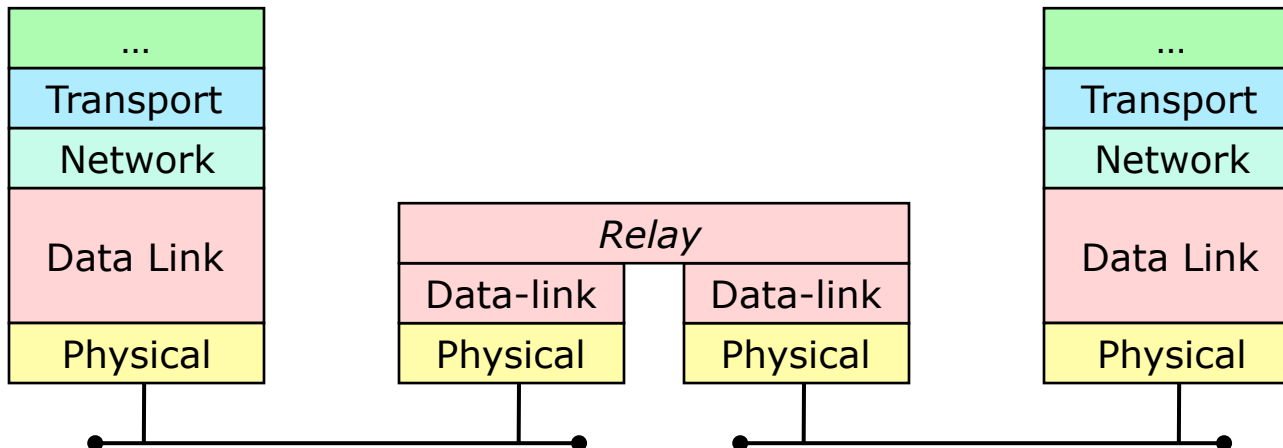
Multiport repeaters: Hubs

- Hubs are multiport repeater
 - Repeater with more than 2 ports
- Required for twisted pairs and fiber cabling (hub-and-spoke topology)
- On Ethernet, it allows reaching (almost) the theoretical collision domain
 - Overcomes limitations of physical cables (e.g., 100m on 10BaseT)



Bridge

- Introduced by DEC in 1983 (LANBridge 100)
 - Pure software
 - 2 ports (mainly for economic reasons)
- Interconnection at the **data-link layer**
 - E.g. Ethernet to WiFi, Ethernet to Fast Ethernet
 - **Different MACs** (medium access mechanism, framing)





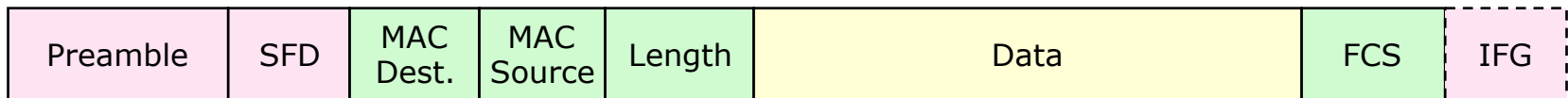
Bridge: objectives

- Interconnection between different LANs using different technologies
 - E.g., Ethernet and WiFi
 - In practice it is often impossible due to maximum frame size issues (data-link does not have fragmentation)
- LAN extension (total diameter)
 - Especially useful for FastEthernet and upper speed (200m)
 - Collision domain issues

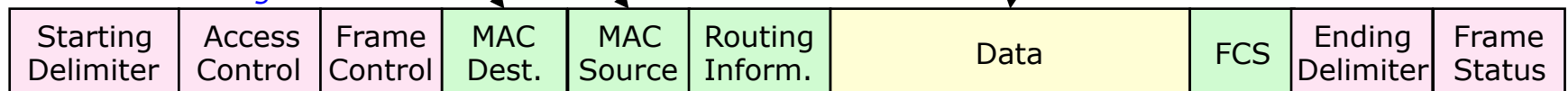
Bridge: operations

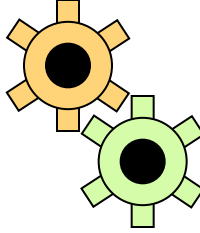
- Works by receiving and re-transmitting (later) a **frame**
 1. Store the frame (*store and forward* mode)
 2. Modify the frame (e.g. Ethernet to Token Ring)
 3. Send it out
- When a frame crosses a bridge
 - The L1 portion will be created from scratch
 - The L2 (MAC) portion will be regenerated (e.g., MAC conversion)
 - LLC and upper layers will transit unchanged

Ethernet DIX



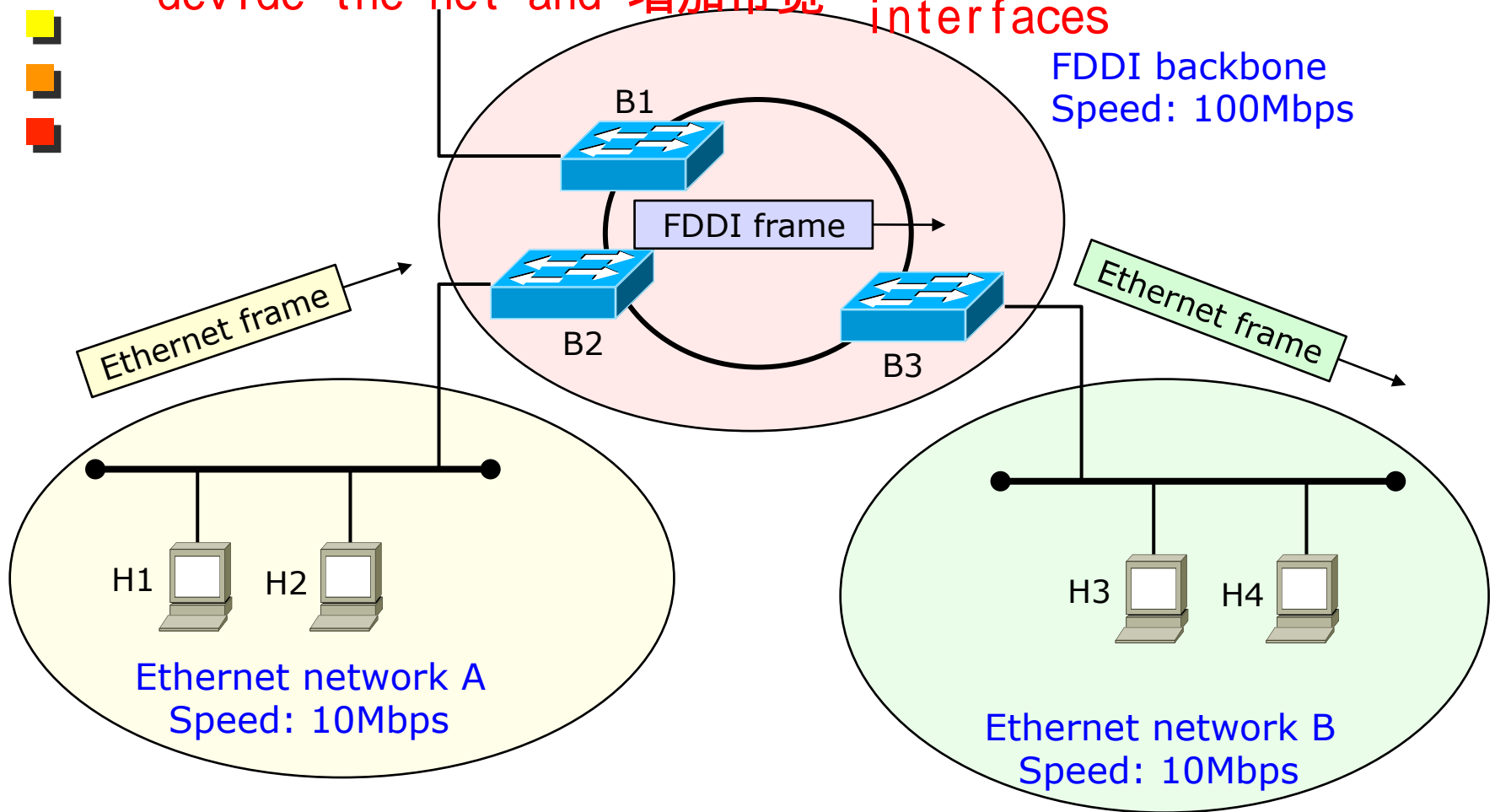
802.5 Token Ring

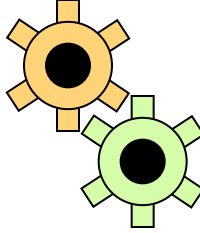




Bridge: example of interconnection

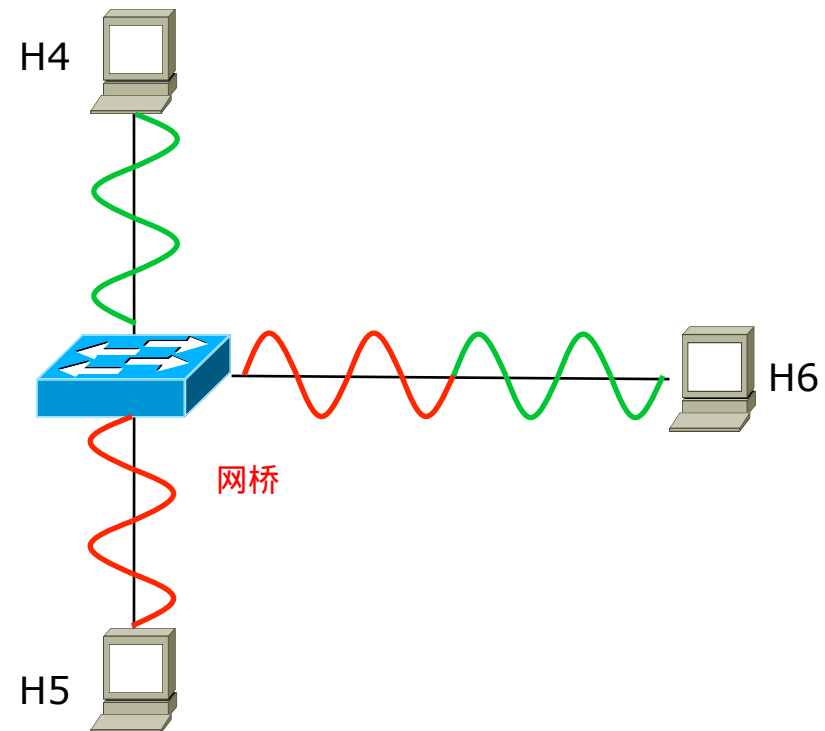
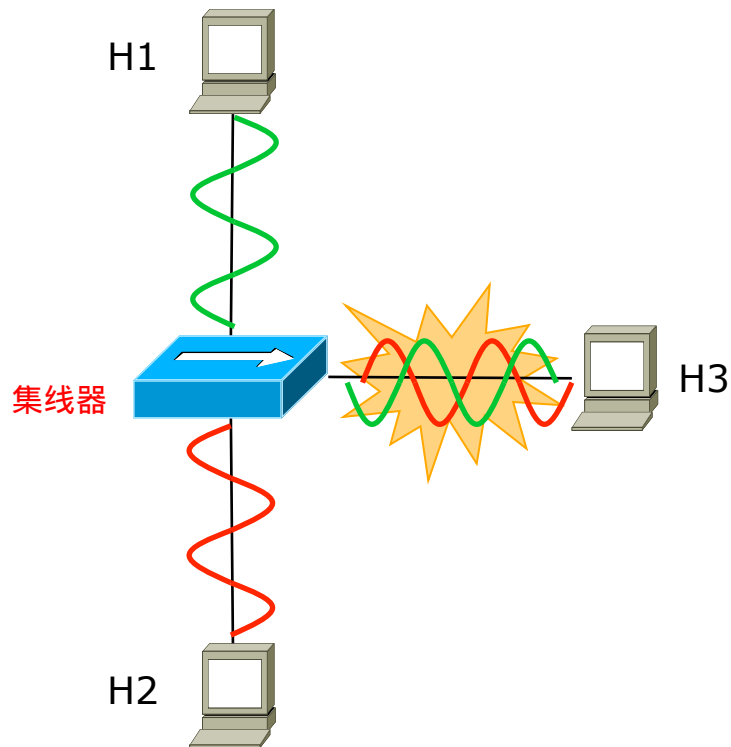
divide the net and 增加带宽 connect different interfaces





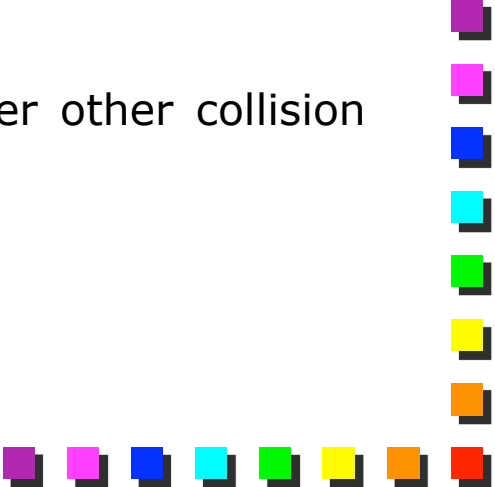
Bridges and collisions (on Ethernet)

- “Store and forward” allows smarter sending of data on output interfaces
- Bridges **decouples collision from broadcast domain**
 - Collision domain is no longer a limitation



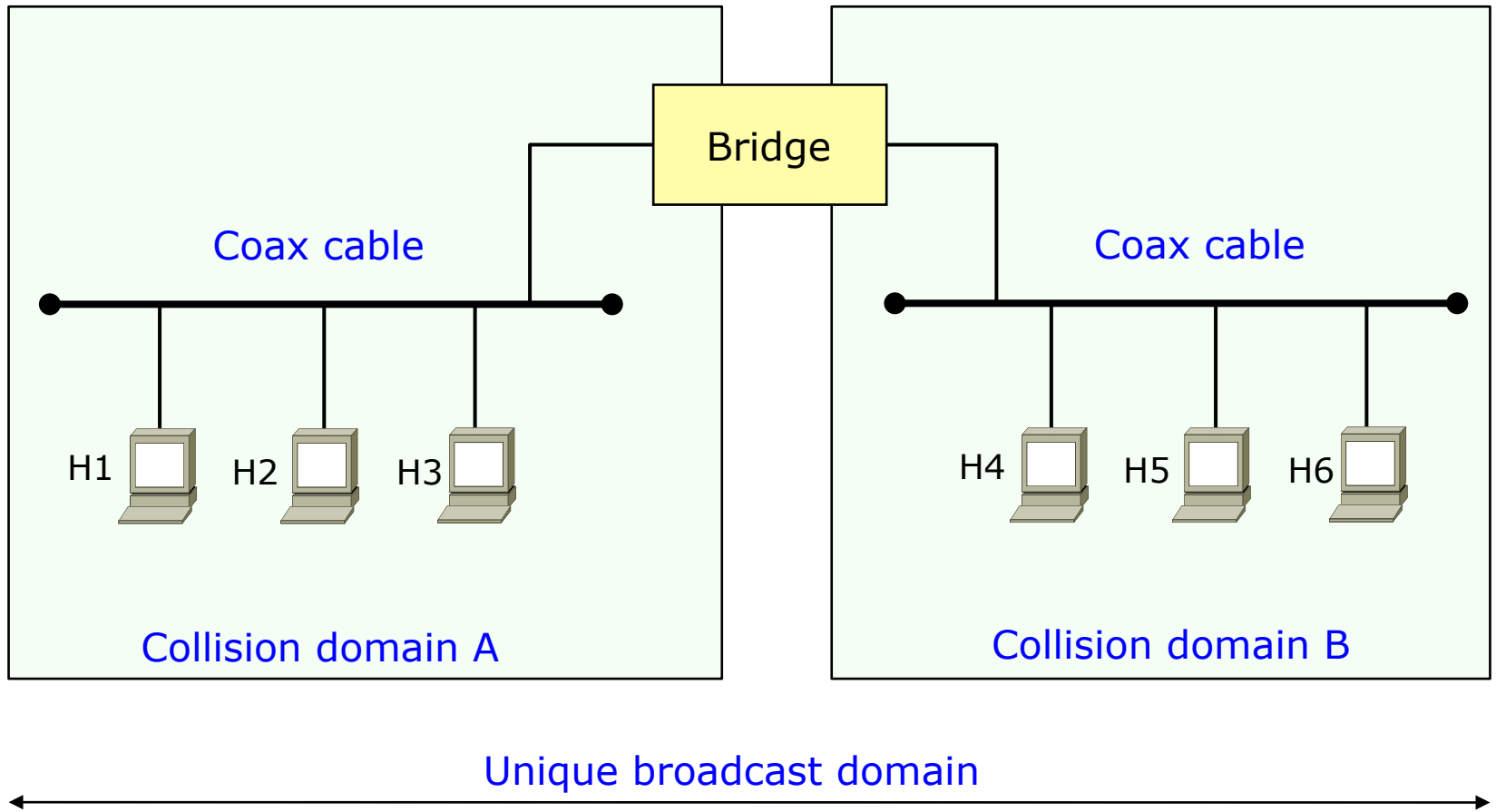


Collision and broadcast domains (1)

- **Collision domain:** area where a single instance of the access control algorithm (e.g., CSMA/CD) operates
 - I.e., the area covered by a *single "physical" link*
 - Frames are immediately propagated over all the links (possibly through repeaters)
 - Also called **network segment**
 - **Broadcast domain:** area where frames can be propagated
 - I.e., the area on which a *LAN* operates
 - Can include several collision domains
 - Frames can be stored and later propagated over other collision domains
- 


Collision and broadcast domains (2)

具有中繼器的作用





Collision and broadcast domains (3)

- Repeaters “extend” the collision domain
 - In fact, it is not actually “extended”; it allows the collision domain to reach its theoretical limits, despite cable limitations
 - Bridges **create different collision domains** and **extend** the broadcast domain
 - I.e., bridges decouple broadcast domain from collision domain
 - This is a very important feature of bridges, that comes out from their “store and forward mechanism”
- 

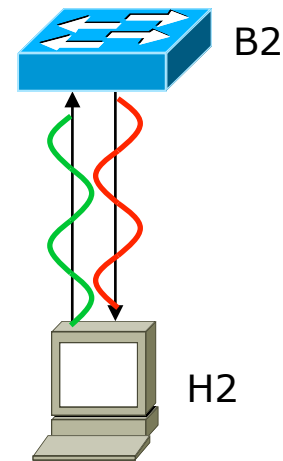
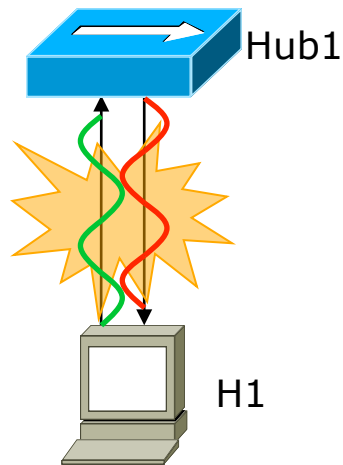
Half and Full duplex mode

■ Half Duplex mode 半雙工

- Standard operating mode of network interfaces (NICs)
- RX and TX cannot happen at the same time
- RX+TX activity is seen as collision

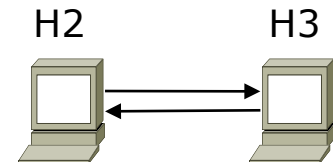
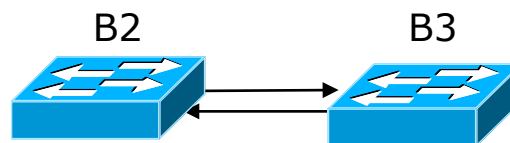
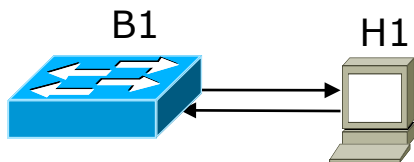
RX 接收器
TX 發射器

- But... if we have two physical links, do we really have a collision?



Full duplex

- Introduced with Fast Ethernet (part of 802.3x)
- Available whenever *the other party* can temporarily store the frame, instead of repeating (immediately) the received bits on the other ports, such as a repeater does
- Not just host \leftrightarrow bridge
 - Examples: host \leftrightarrow host, host \leftrightarrow bridge, bridge \leftrightarrow bridge





Full duplex: advantages

■ Bandwidth

- In theory, throughput x2
- In practice, limited advantage for clients and servers
 - Clients tend to saturate downlinks, servers uplinks
- May be interesting for bridges on the backbone
 - More symmetrical bandwidth

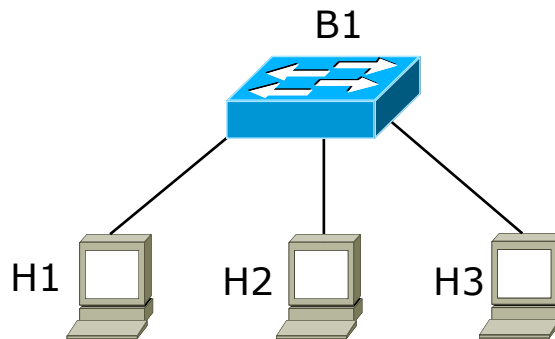
■ CSMA/CD: the real advantage

- No longer needed, since collisions are no longer possible
 - With CSMA/CD, TX and RX together are used to detect collisions
- Advantages
 - No requirement for min frame size for Ethernet
 - No limits on the network size on Ethernet (no collision domain)

carrier sense multiple access with collision detection
載波監聽多路訪問/衝突檢測協議

Full duplex and switches

- Modern LANs are heavily based on full duplex
- Hub-and-spoke topology
 - Point-to-point connections between hosts and the “bridge”
 - No collision domain
- Multiport bridges are called “switches”
 - Same functions, different internal architecture





Modern LANs: Switched Ethernet

- Modern (wired) LANs are based on full-duplex, switches, and Ethernet: *Switched Ethernet*
 - Today Gigabit Ethernet, or even more (10GE, 100GE)
- If today we say “switch”, we are referring to an Ethernet switch
- CSMA/CD no longer used
 - Available till 1GE, then no longer defined by standards
- Wireless LANs are completely different
 - Typically based on CSMA/CA (e.g., WiFi)
 - Hubs are still used (e.g., WiFi extenders)

...Here we focus on wired Ethernet-based LANs



透明网桥一般用于连接以太网段，而源路由选择网桥则一般用于连接令牌环网段（即插即用，自动配置）

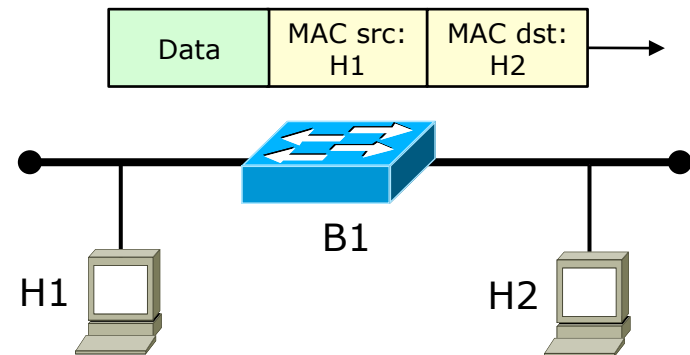
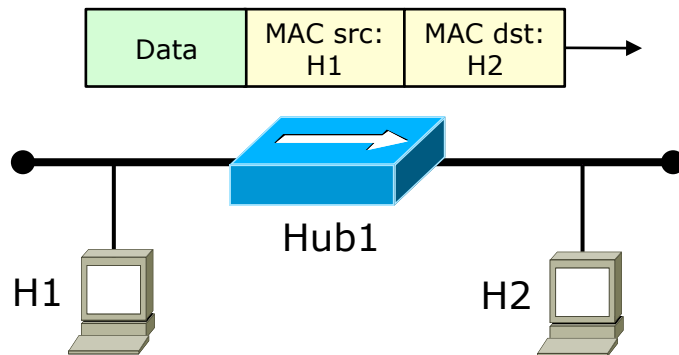
Transparent bridges

透明網橋

- Bridges (switches) used in Ethernet LANs are called **transparent bridges**
 - Other (non transparent) bridges have been proposed in the past (e.g. Token Ring networks)
 - No longer in use
- Transparent bridges standardized by IEEE in 802.1D
- Transparency
 - Bridges should be plug-and-play and must not require any change in the configuration of the end systems
 - Performance (throughput, max distances) may be different from the original network, but functionalities are the same

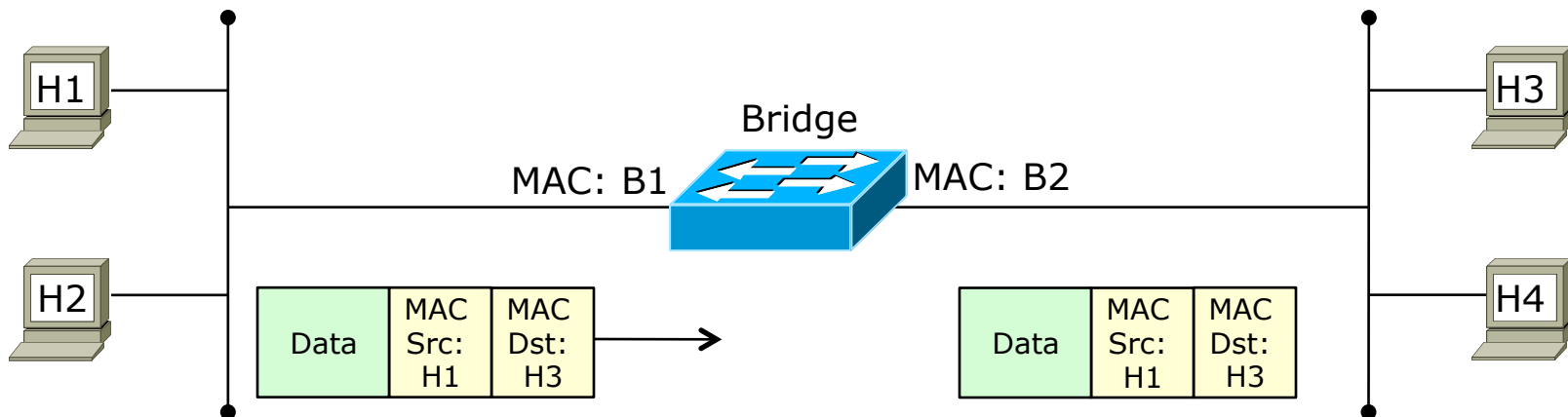
Transparent bridges and end hosts

- End systems must operate in the same way (same frames, some format, etc) with or without bridges
- In details
 - No changes at all in frames sent by end systems
 - Same frame, same src/dst MAC address, etc...
 - There may be some changes in **which** frames are received
 - No changes at all in the **format** of the received frame
 - Same source/MAC address, etc



Transparent bridges and port addresses

- Each port of a bridge has a MAC level and therefore **it has** a MAC address
 - That MAC address is never used when forwarding data frames
 - It is used when frames are generated/received by the switch itself
 - E.g. management frames





Smart forwarding process

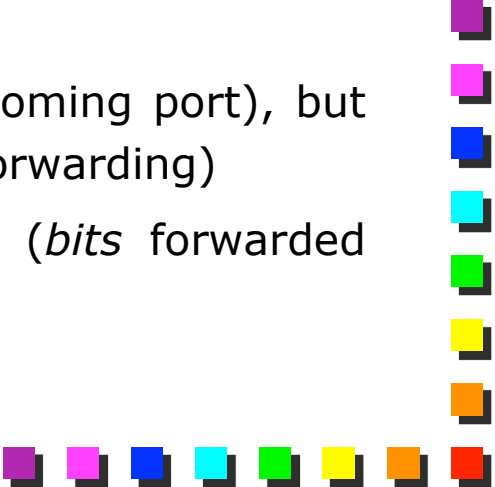
■ Smarter forwarding rules

- Unicast: only on the port toward we can reach the destination (Destination MAC-based forwarding)
- Multicast, broadcast: flooding
 - All ports except the port on which the frame has been received (*flooding*)

■ A MAC forwarding table must be available locally


- Filtering database (more details later)

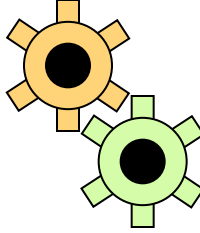
■ A note about flooding

- Frames are sent on all ports (except on the incoming port), but **may not be sent at the same time** (delayed forwarding)
 - Hubs send data in flooding at the same time (*bits* forwarded immediately)
- 



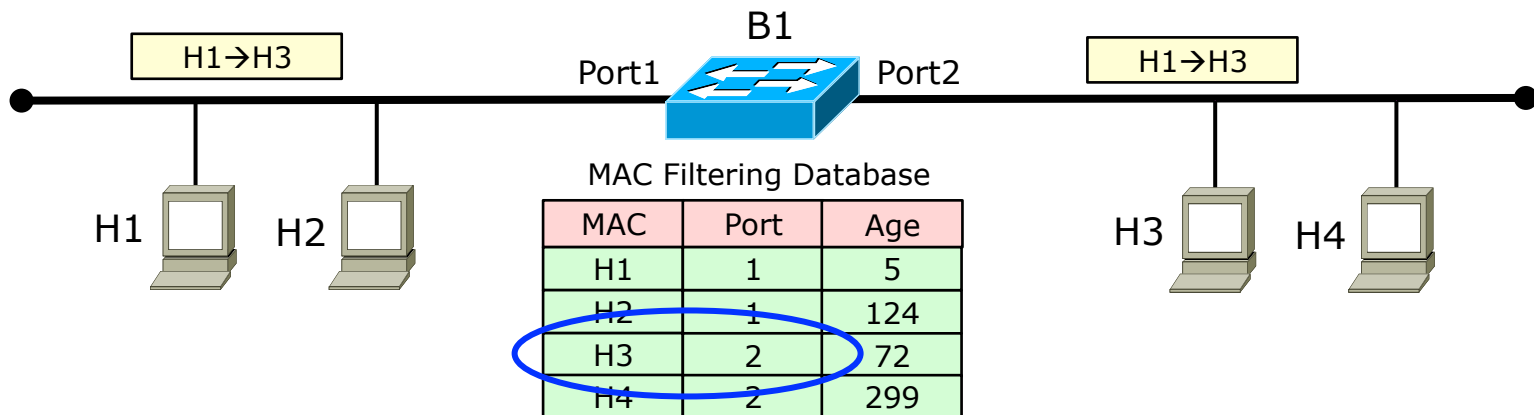
New components in “smart” bridges

- In order to operate successfully, a “smart” bridge requires three additional components:
 - A local forwarding table (*filtering database*)
 - Stations auto-learning (*backward learning*) 逆向學習法
 - Loop detection (*spanning tree algorithm*) 生成樹
 - The ultimate goal: the bridge should be able to do its job without any explicit configuration from the network admin
 - Really “plug and play”
 - By-product: stupid network admins believe they are really smart just because their networks work properly
- 




Filtering Database (1)

- Table with the “location” of any MAC address found in the network
 - MAC address
 - Destination port
 - Ageing time (default expire after 300 s)
- “Filtering” database: in the old days, the smart forwarding process was perceived as a way to “filter out” unwanted traffic from a link





Filtering Database (2)

- Entry types
 - Dynamic
 - Populated and updated by the backward learning process
 - Max entries: $2 \div 64 \text{ K}$
 - Static
 - Not updated by the learning process
 - Usually $< 1\text{K}$ entries
 - Old dynamic entries are purged out of the filtering database
 - E.g., stations that do no longer exist on the network
 - Default: 300 seconds
- 

Filtering database: real example

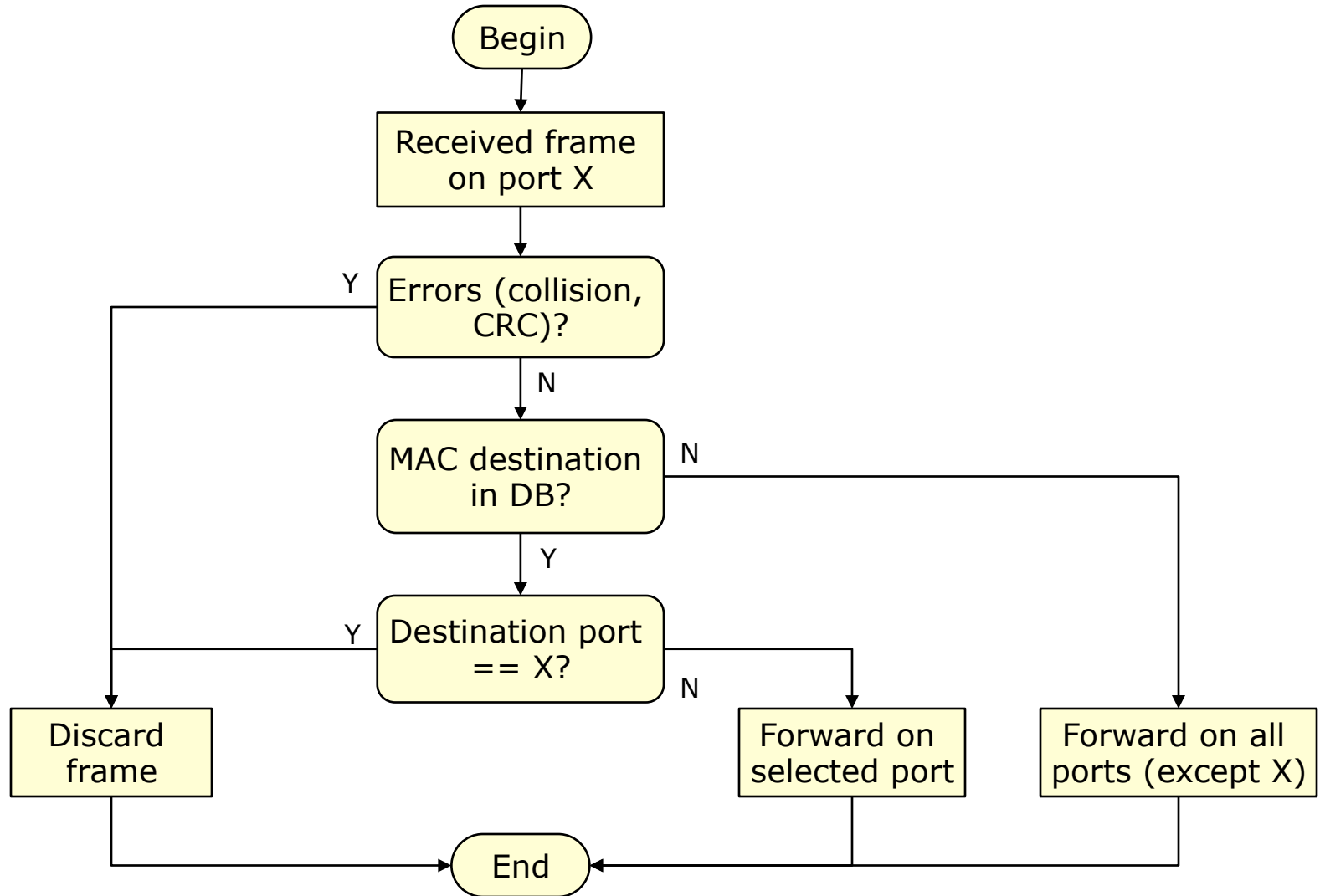
```
Cisco-switch-1> show cam dynamic
```

* = Static Entry. + = Permanent Entry.

= System Entry X = Port Security Entry


Dest MAC Address	Ports	Age
-----	-----	---
00-00-86-1a-a6-44	1/1	1
00-00-c9-10-b3-0f	1/1	0
00-00-f8-31-1c-3b	1/2	4
00-00-f8-31-f7-a0	1/1	2
00-01-e7-00-e3-80	2/2	0
00-02-a5-84-a7-a6	2/1	1
00-02-b3-1e-b4-aa	2/1	5
00-02-b3-1e-da-da	2/5	1
00-02-b3-1e-dc-fd	2/4	2

Forwarding process





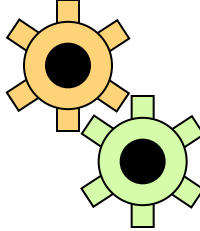
Forwarding process and transient

- What about if a MAC address is not present in the Filtering Database?
 - Bridge looks like an hub
 - Frame duplicated on all ports except the one on which it was received
 - This situation is rather common and it is called “transient”
 - Bridges are plug-and-play and have an algorithm **to learn the location of the hosts**
 - Backward learning (presented later)
 - However, at the beginning, bridges do not know where an host is located
 - In this case the “MAC Flooding” algorithm is the only way to go
- 



How do we populate the filtering database?

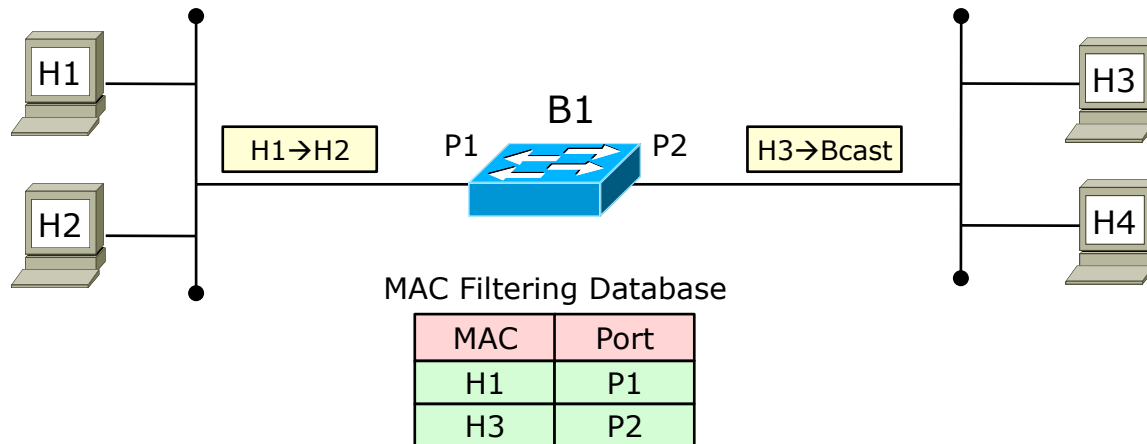
- 1) By hand
 - Possible on all modern devices, but not very handy 便利的
- 2) By means of a proper algorithm
 - Backward learning
 - The best choice, of course

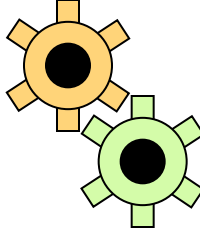


Backward learning (1)

■ The idea

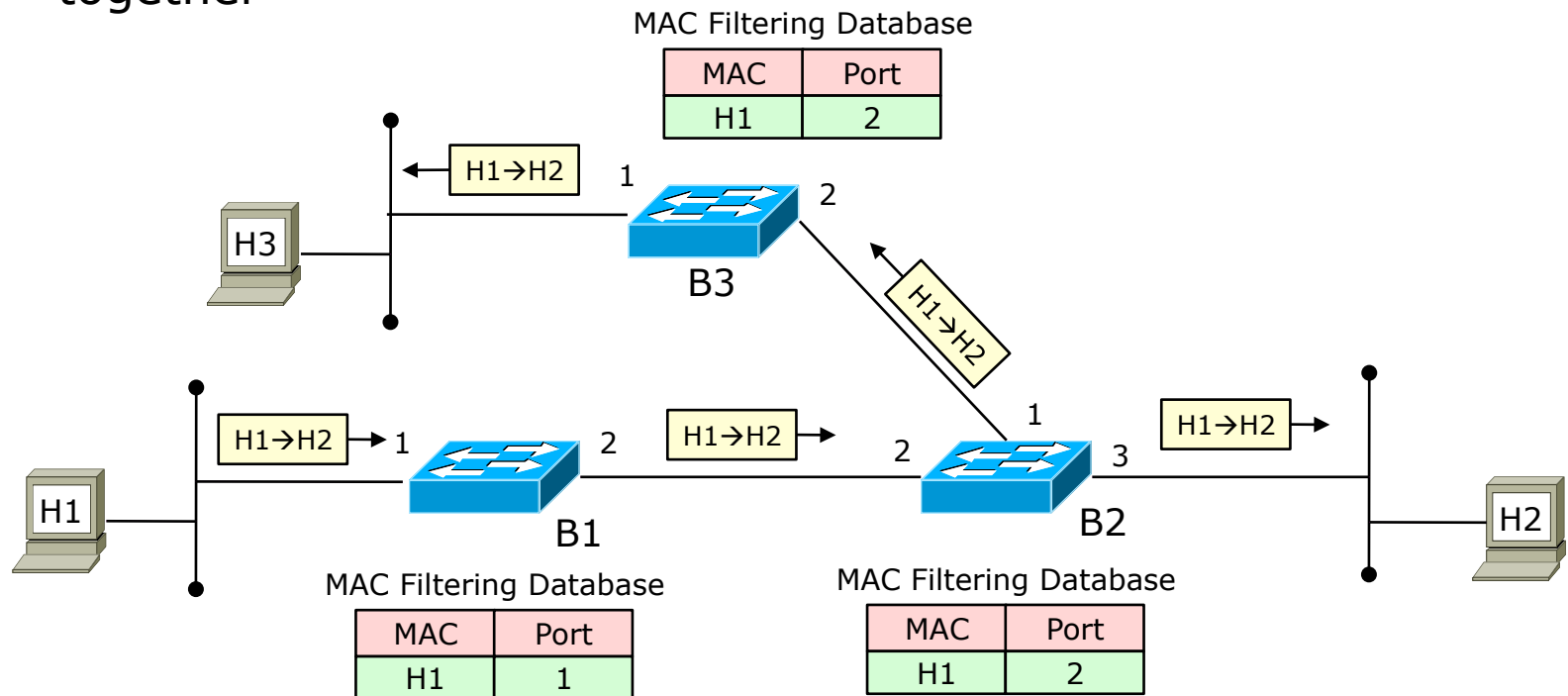
- If a bridge receives a frame whose source is host H1 from port P1, that host will be reachable through port P1
- Topology is learned by inspecting **received** frames
検査
- Analysis of MAC source address
- The destination MAC address is ignored by this algorithm



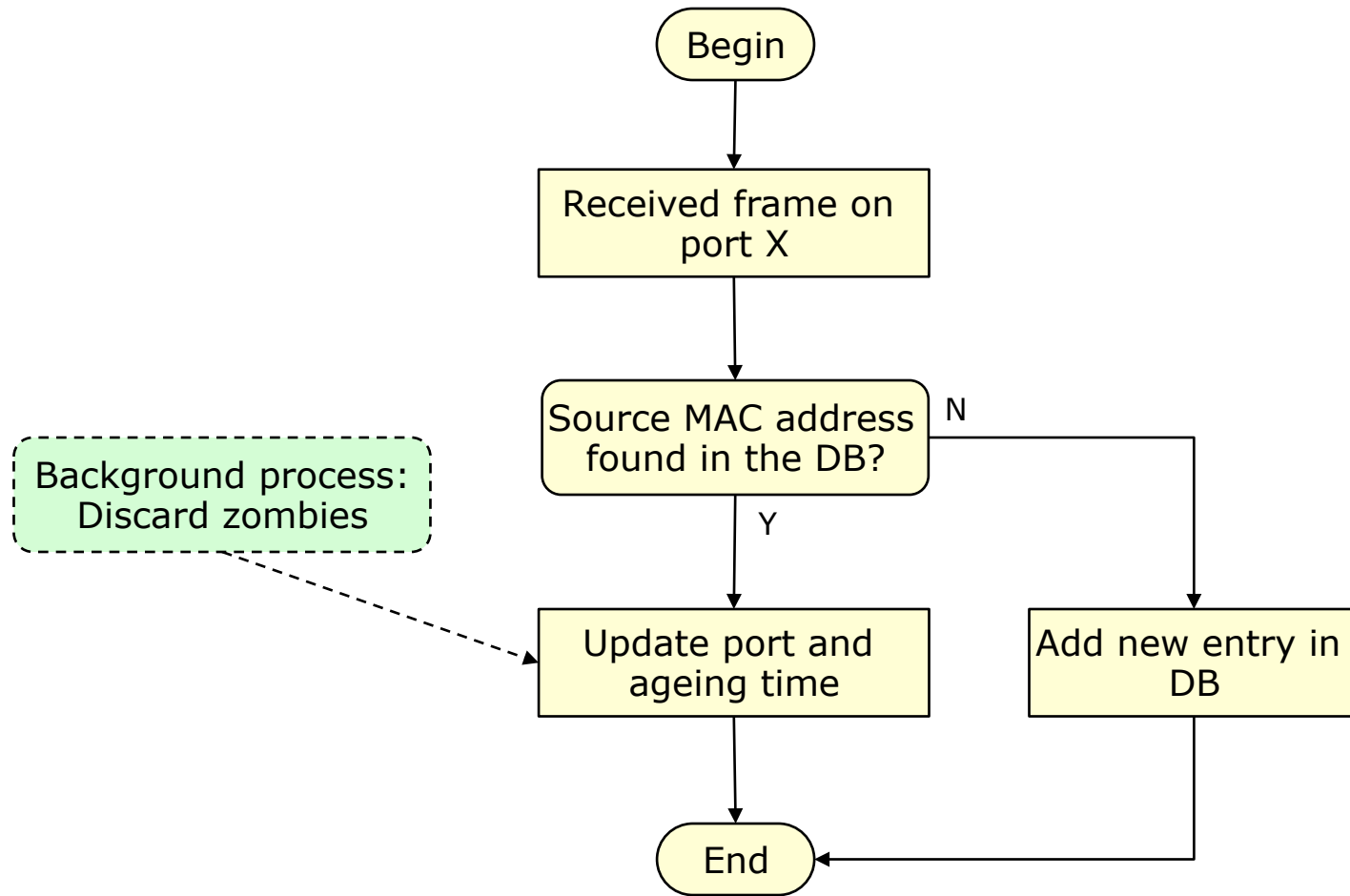


Backward learning (2)

- Works also in presence of multiple bridges
 - Remote bridges learn the position anyway, even if the end-system is not connected locally
- Example: backward learning and frame forwarding taken together

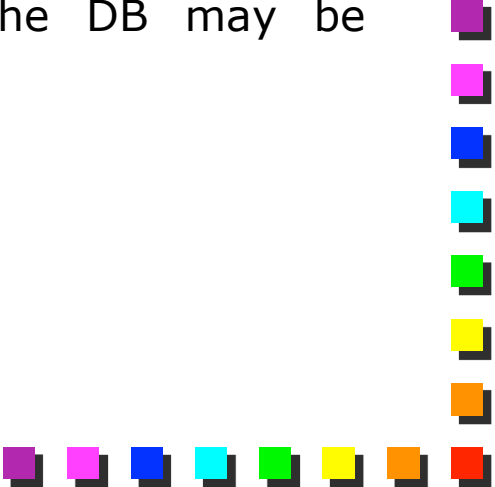


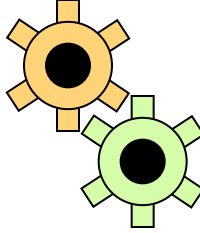
Backward learning (3)





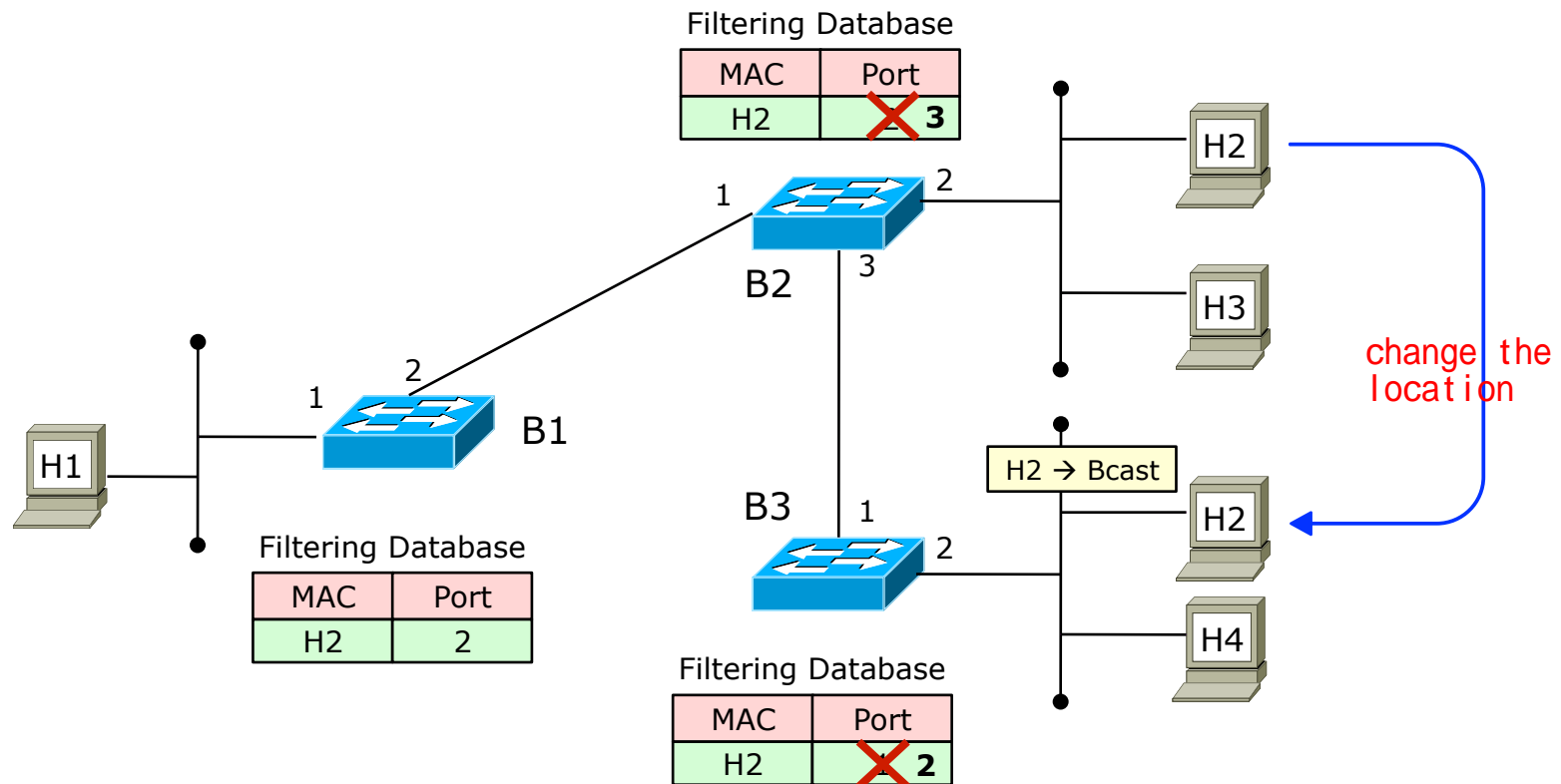
How do we keep the filtering DB up to date?

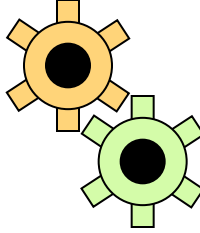
- Update the Filtering database means...
 - Refresh "Age", so that the entry keeps alive
 - Refresh "Port", so that the host is updated with the new position
 - Please note that...
 - An end-system whose MAC address is not in the DB is *always reachable*
 - Corollary: ^{推論} a frame sent to a non-existing host will always be forwarded in all the network
 - An end-system whose MAC address is in the DB may be *unreachable*
 - At most for *Aging Time*, in fact
- 



L2 networks and hosts mobility (1)

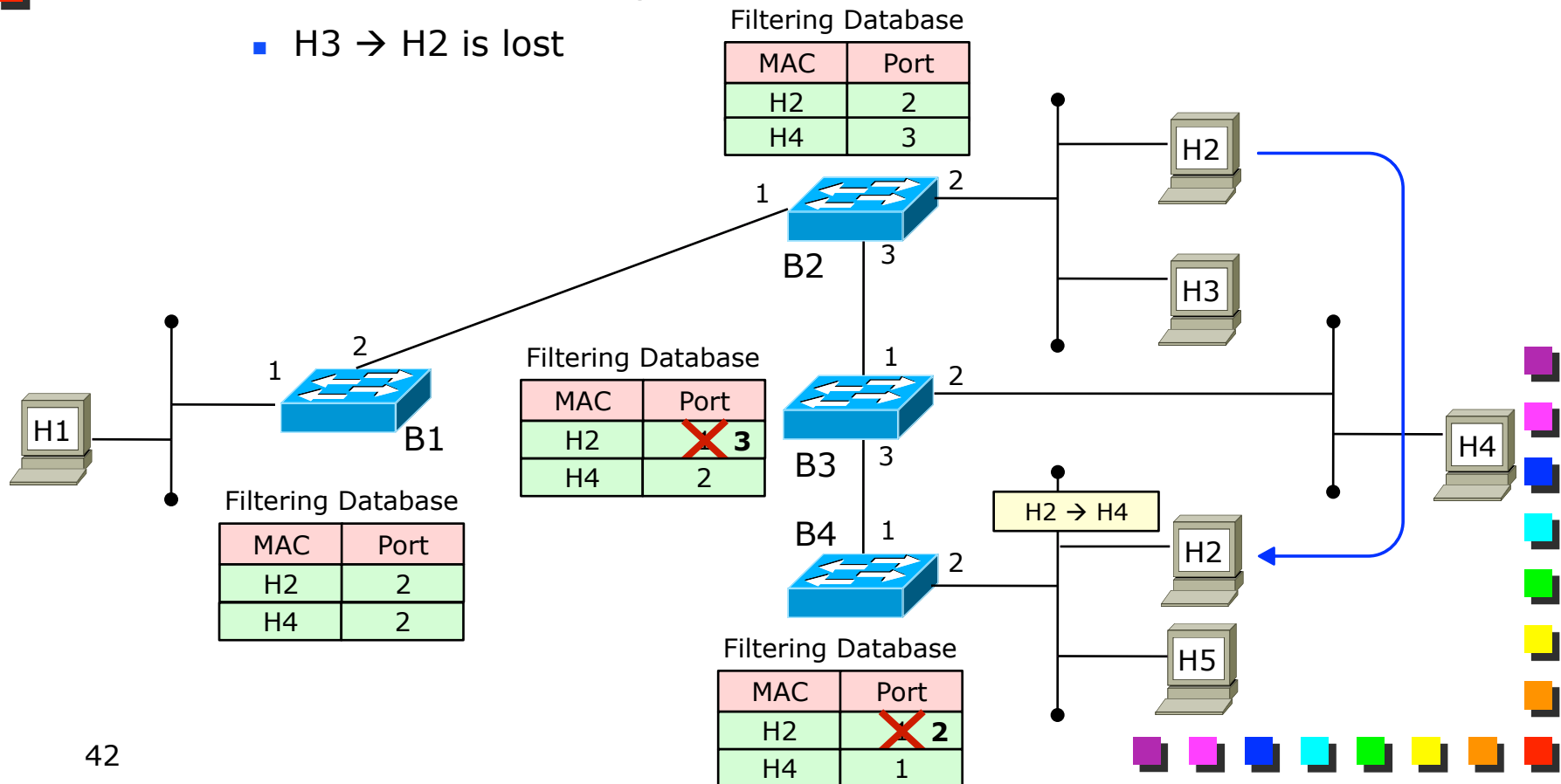
- If the end-system generates broadcast frame immediately
 - No problems

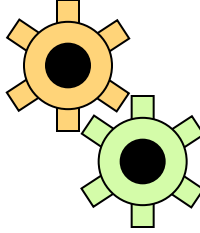




L2 networks and hosts mobility (2)

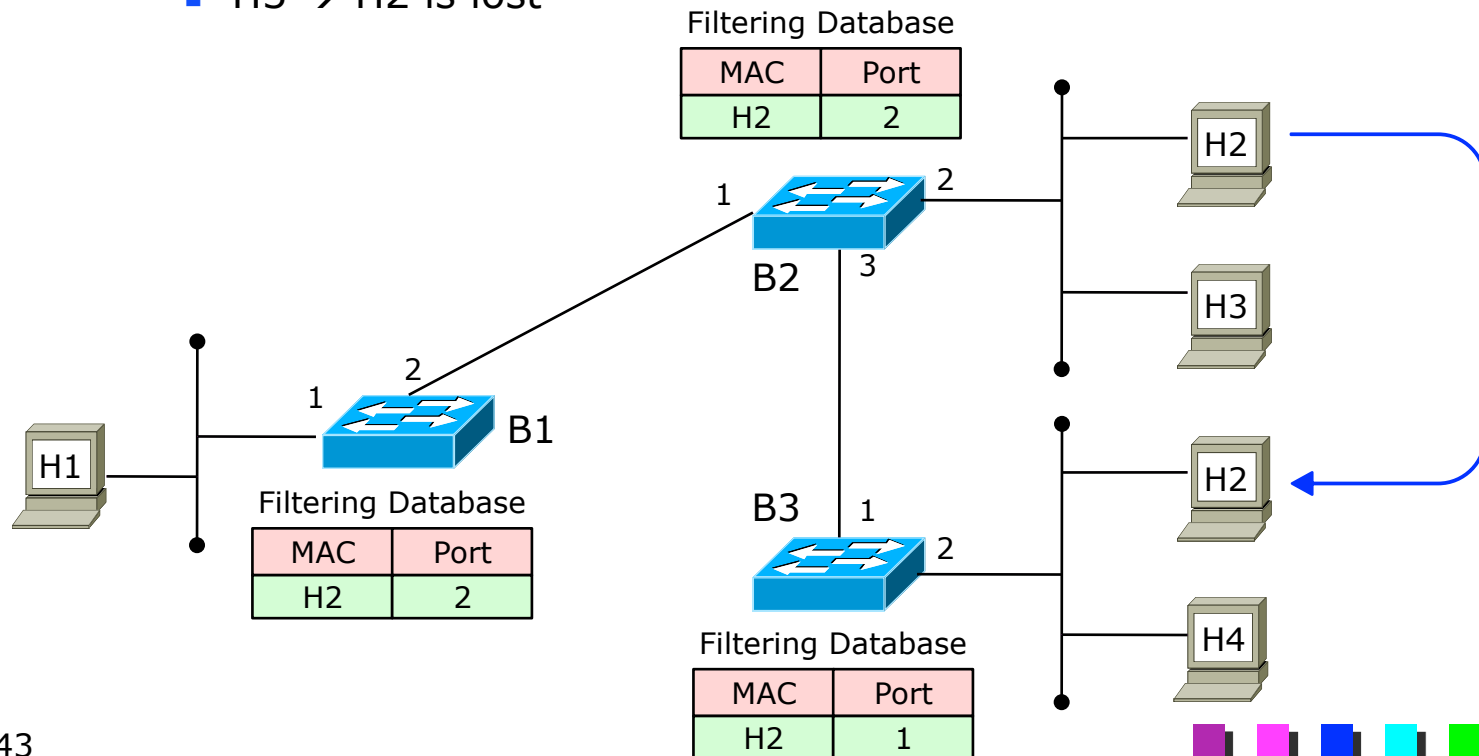
- If the end-system generates unicast traffic immediately
 - We may have forwarding errors
 - H4 → H2 is correctly delivered
 - H3 → H2 is lost





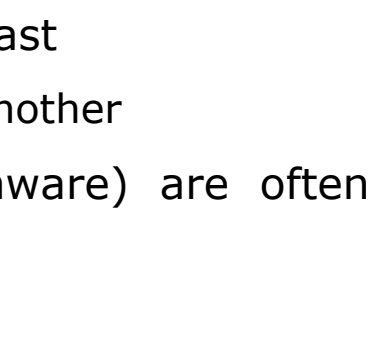
L2 networks and hosts mobility (3)

- If the end-system does not generate traffic at all
 - We may have forwarding troubles
 - H4 → H2 is correctly delivered
 - The frame is forwarded also to the original destination
 - H3 → H2 is lost



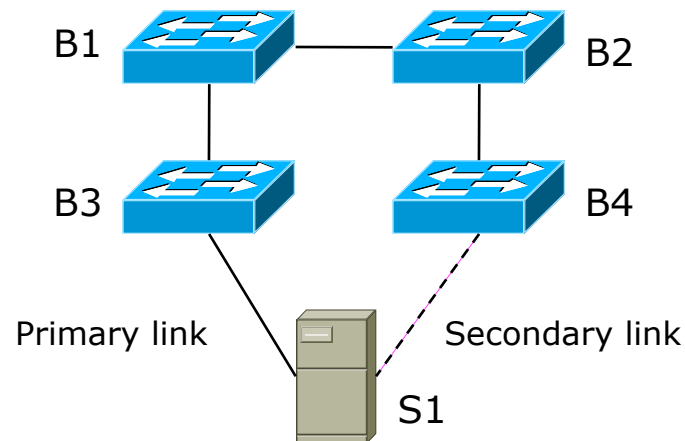


L2 networks and hosts mobility (4)

- Broadcast (multicast) frame
 - Reaches the entire network, therefore all the bridges update the location of the current station
 - Unicast frame
 - Potentially reaches only a portion of the network, hence the rest may still have the old location of the station
 - In the real world
 - Windows host typically generates a lot of broadcast
 - No problems when moving from one place to another
 - UNIX servers and virtualized hosts (e.g., Vmware) are often silent if not solicited
 - Need to wait for the aging time
- 

L2 networks and hosts mobility (5)

- The aging time
 - Usually enough in order to cope with manual movements
 - A laptop moved from office to lab
- Some problems may appear in specific environments
 - E.g. fault-tolerant NICs
 - We need to react much quickly than 5min
 - NIC driver has to generate an additional broadcast frame





Possible attacks to the filtering database

■ MAC Flooding Attack

- Generation of frames with random MAC sources
- Filtering database gets full
- Bridges will start flooding most of the frames
 - All the ones whose destination address is not present in the DB
- Objectives
 - Forces bridges to operate like hubs, so that we can intercept traffic generated by other stations
 - Slows down the network
- Some vendors give the opportunity to limit the number of MAC addresses learnt on each port



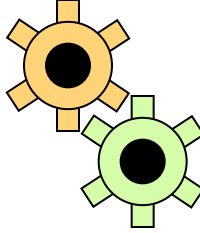
Possible attacks to the filtering database (2)

- Packet storms
 - Generation of frames to non-existing stations
 - Frames are always send to the entire network
 - Objective
 - Slows down the network

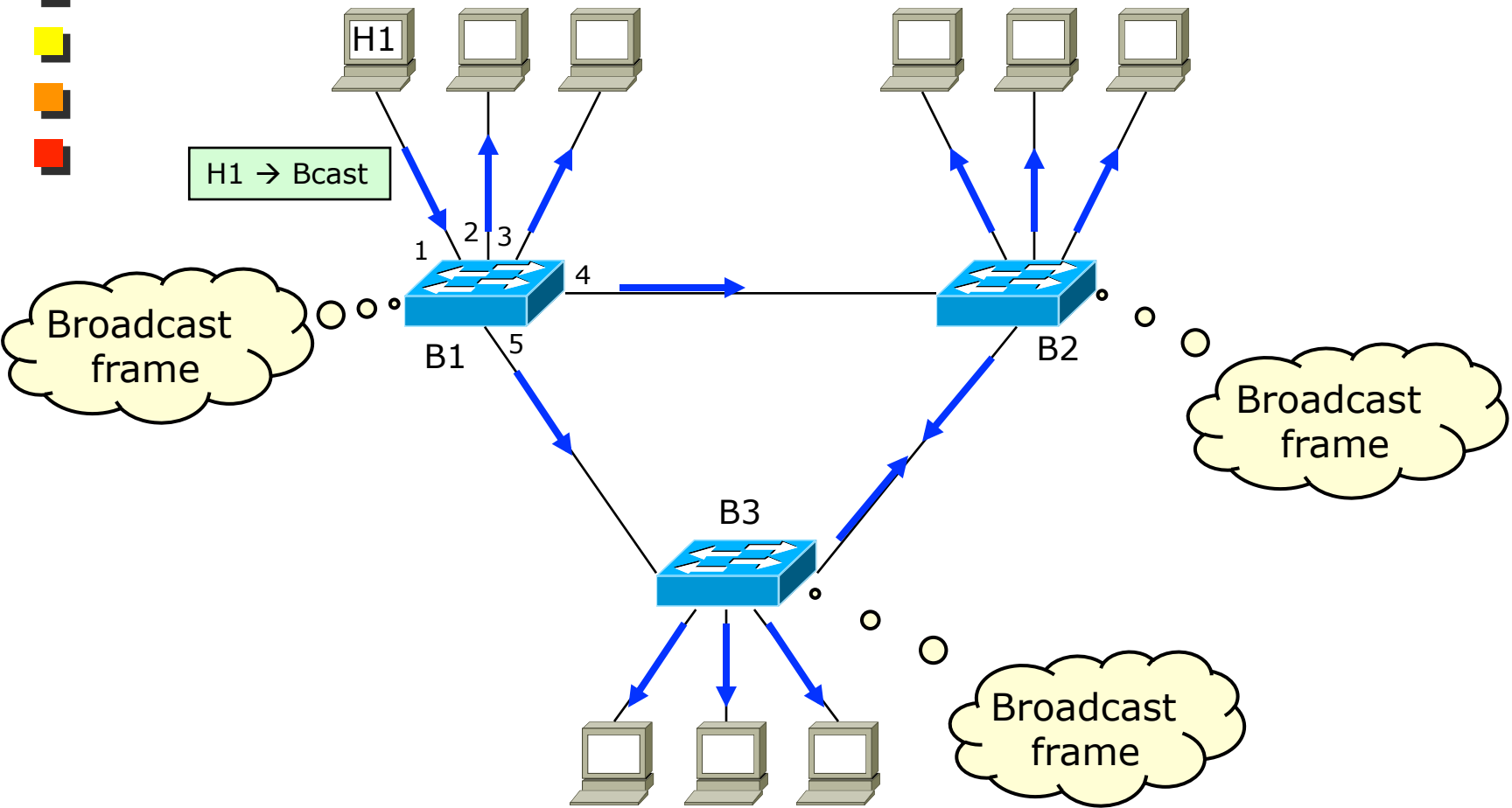


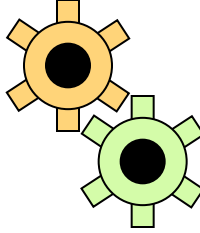
Bridges and meshes

- Two problems
 - Frames can enter in a loop
 - Backward learning no longer able to operate
- It's now the time to present the third component (i.e. "Spanning Tree") after the ones we presented earlier
 - "Filtering Database" and "Backward Learning"

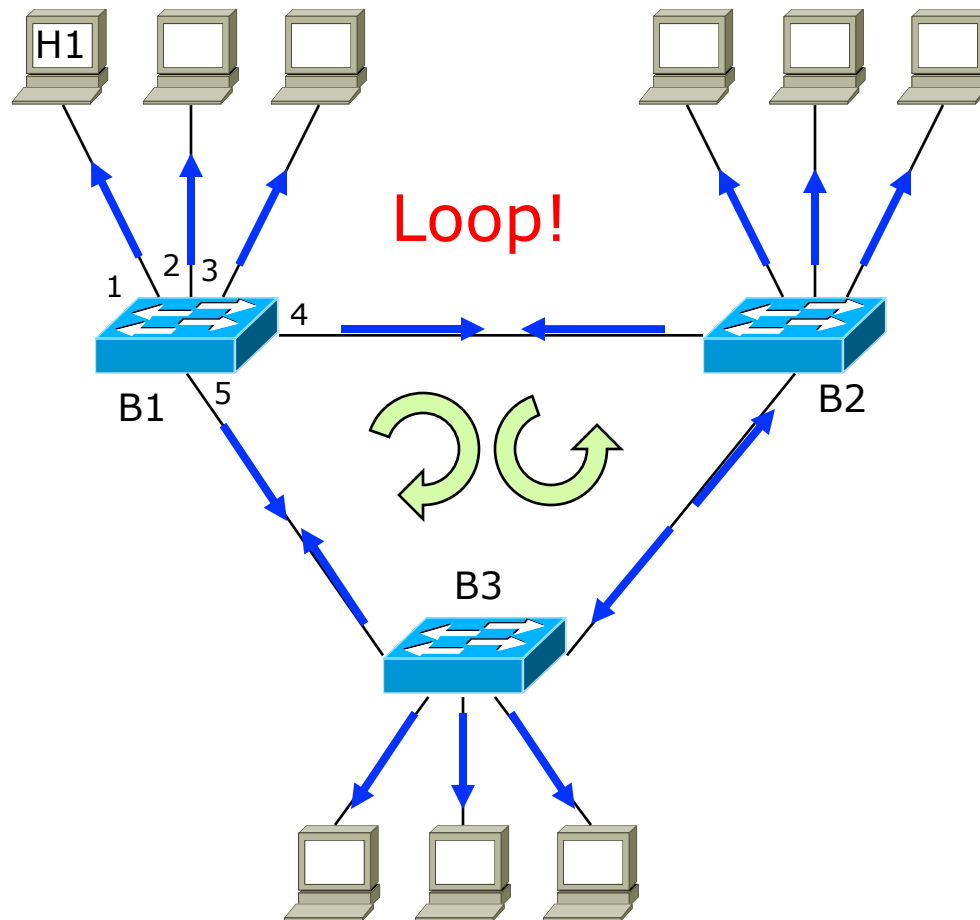


Bridges and meshes: the loop problem



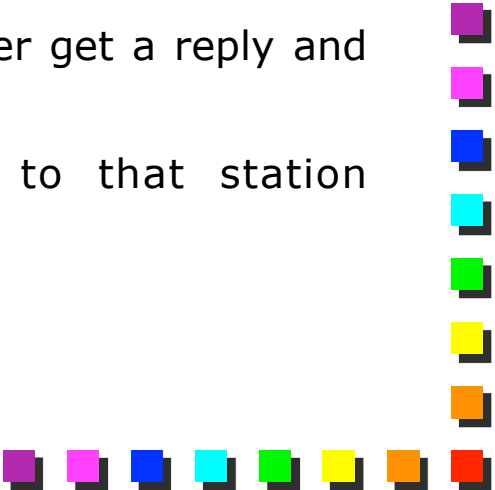


Bridges and meshes: the loop problem






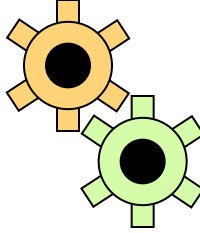
Which frames can generate a loop?

- Multicast/broadcast frames
 - Very common
 - Frame to a non-existing station
 - MAC address not present in the filtering DB (e.g. non existing station)
 - Problem that may happen rarely (unless under attack)
 - IP sends an ARP before contacting an L2 station
 - If the station does not exist, the ARP will never get a reply and the destination MAC address is unknown
 - Therefore, no MAC frames will be sent to that station intentionally
- 

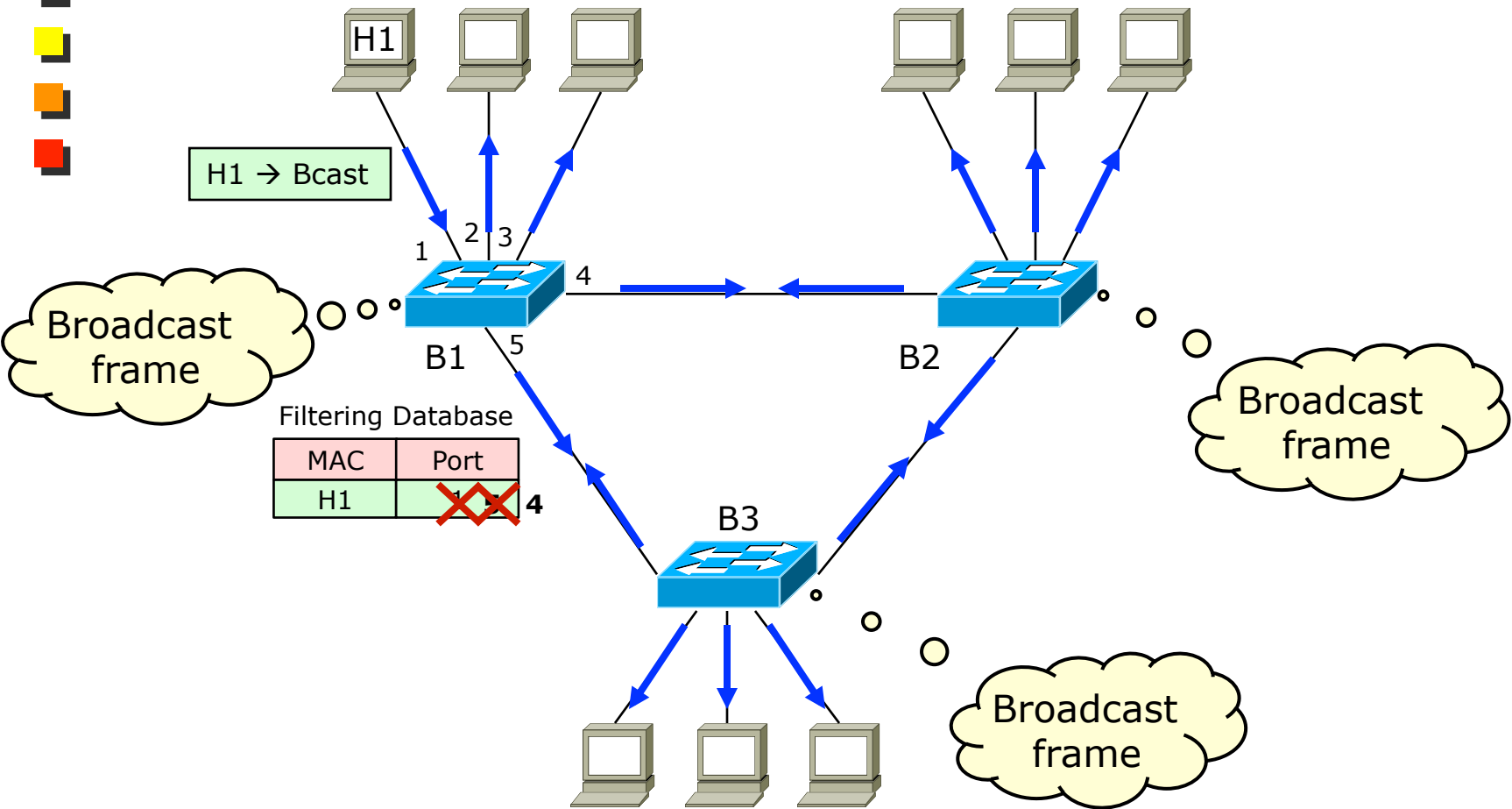


The Broadcast Storm

- Massive load due to broadcast/multicast traffic on a LAN
 - One of the most dangerous problems at data-link layer
 - No solutions, except for disabling (physically) loops
 - E.g., detach a cable from a bridge 無力的
 - Network operators are almost impotent in such this case
 - Due to the lack of a “time-to-live” field in L2 frames
 - L3 networks can tolerate transient loops
 - TTL available on L3 packets
 - Can be used to create a low-cost traffic generator sending frames at line-rate
- 




Bridges and meshes: the learning problem (1)

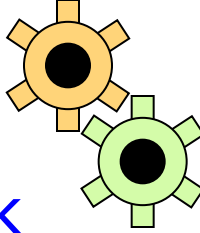




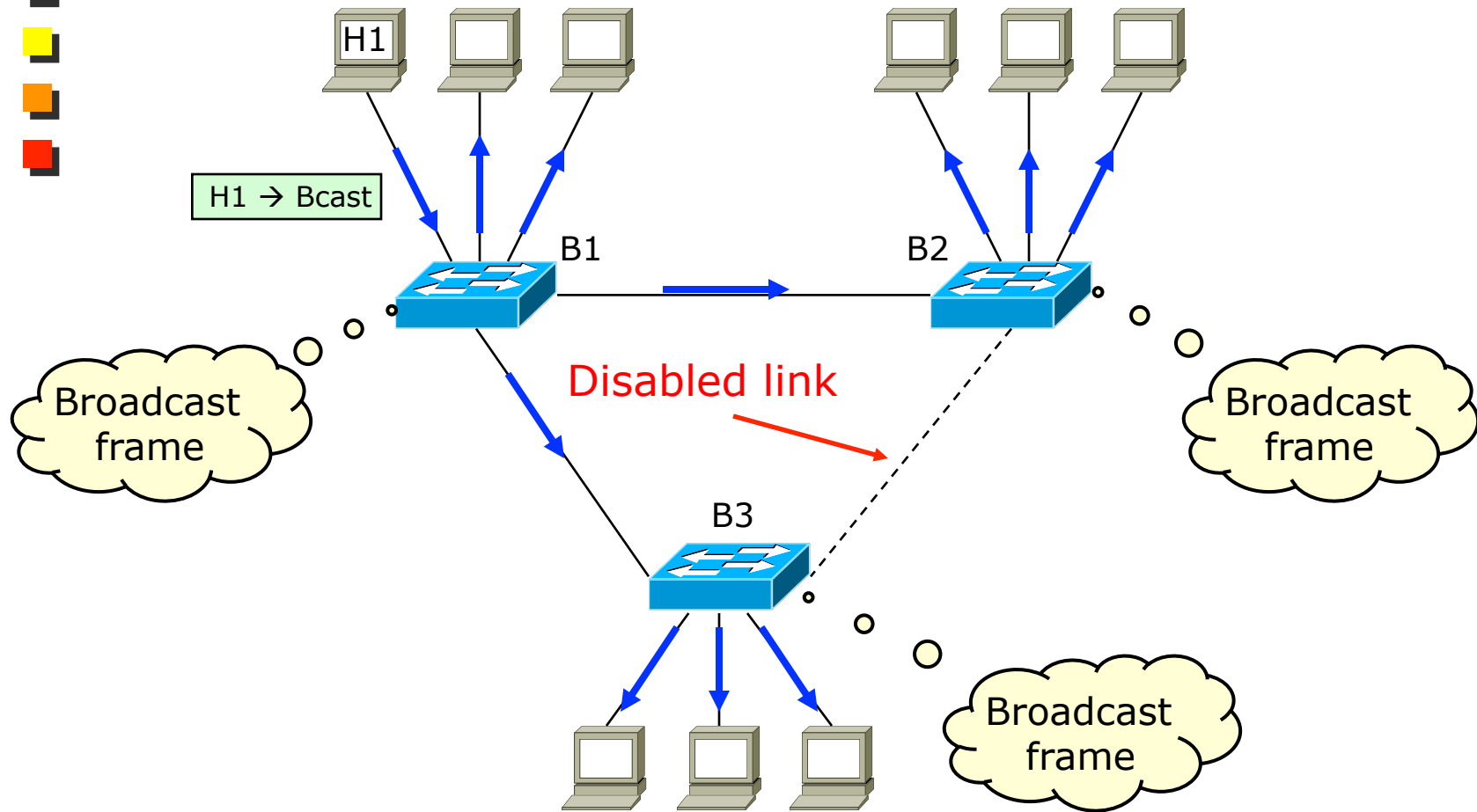
Bridges and meshes: the learning problem (2)

■ Backward learning problem

- Switches may have inconsistent filtering database
 - An entry in the filtering database may change the port indefinitely
 - An entry may not be able to reach a stable state
 - Transient loops can be created among back-to-back bridges
 - B1 forwards to B2 that forwards to B1,...
 - Larger (B1-B2-B3-B1) loops may occur as well
- 



The Spanning Tree idea: no loops in the network

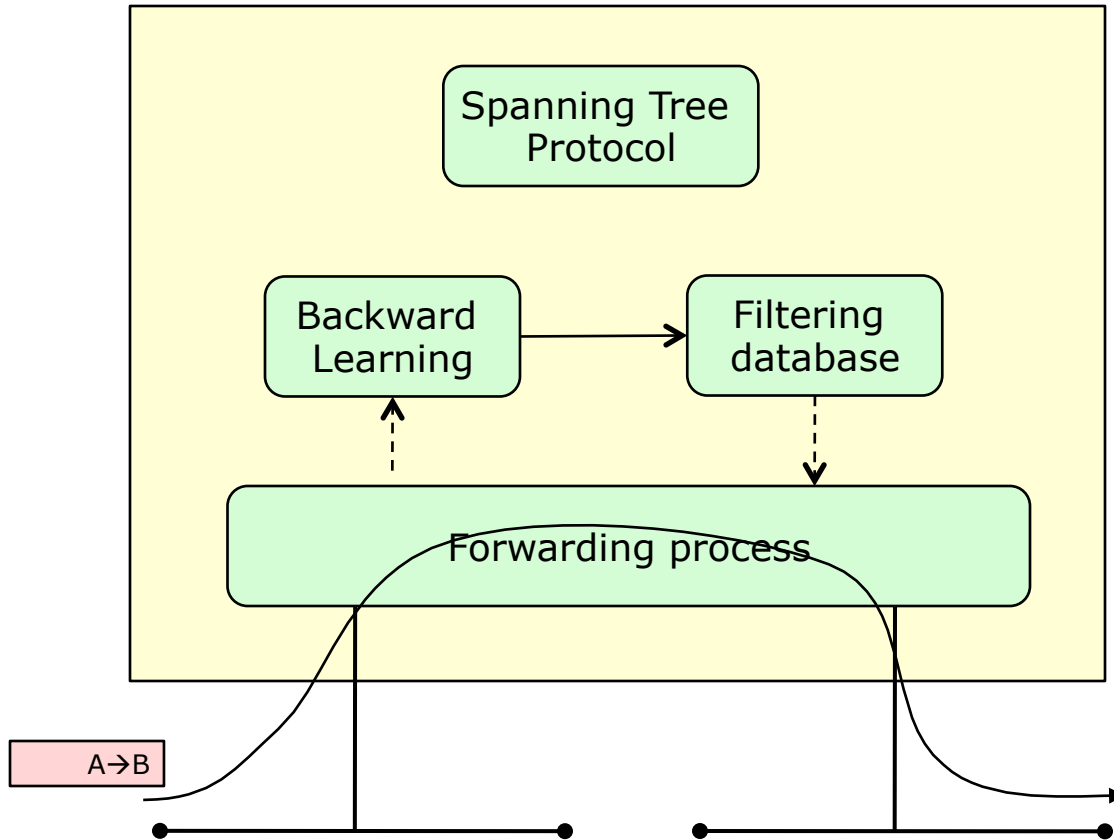




Spanning Tree

- In order to avoid troubles, you must avoid loops in the physical network
 - Either create loop-free networks
 - Discouraged; not robust
 - Or define an algorithm that disables (temporarily) loops
- 802.1D
 - Original idea from Radia Perlman, PhD @DEC
- Meshes detected and disabled; the network becomes a tree
 - 陷阱 Unique path between any source and any destination
- Operates periodically (every second)
 - Decides which port set to forwarding state and which port set to blocking state

Bridge architecture





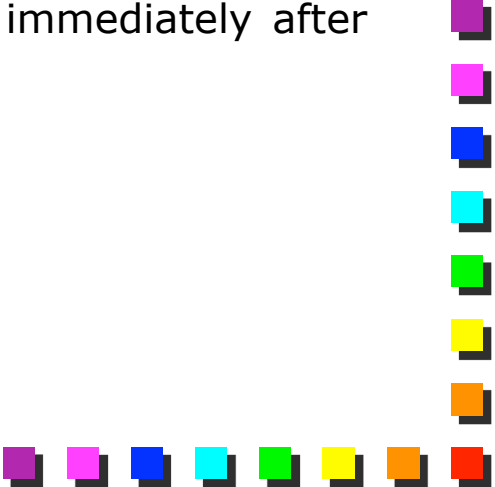
Bridges and switches (1)

- Bridge
 - Originally 2 ports, then more
 - Software-based architecture
 - No longer used in real networks
 - Still some PC-based implementations
 - For research or some special purpose
 - WiFi access points are bridges



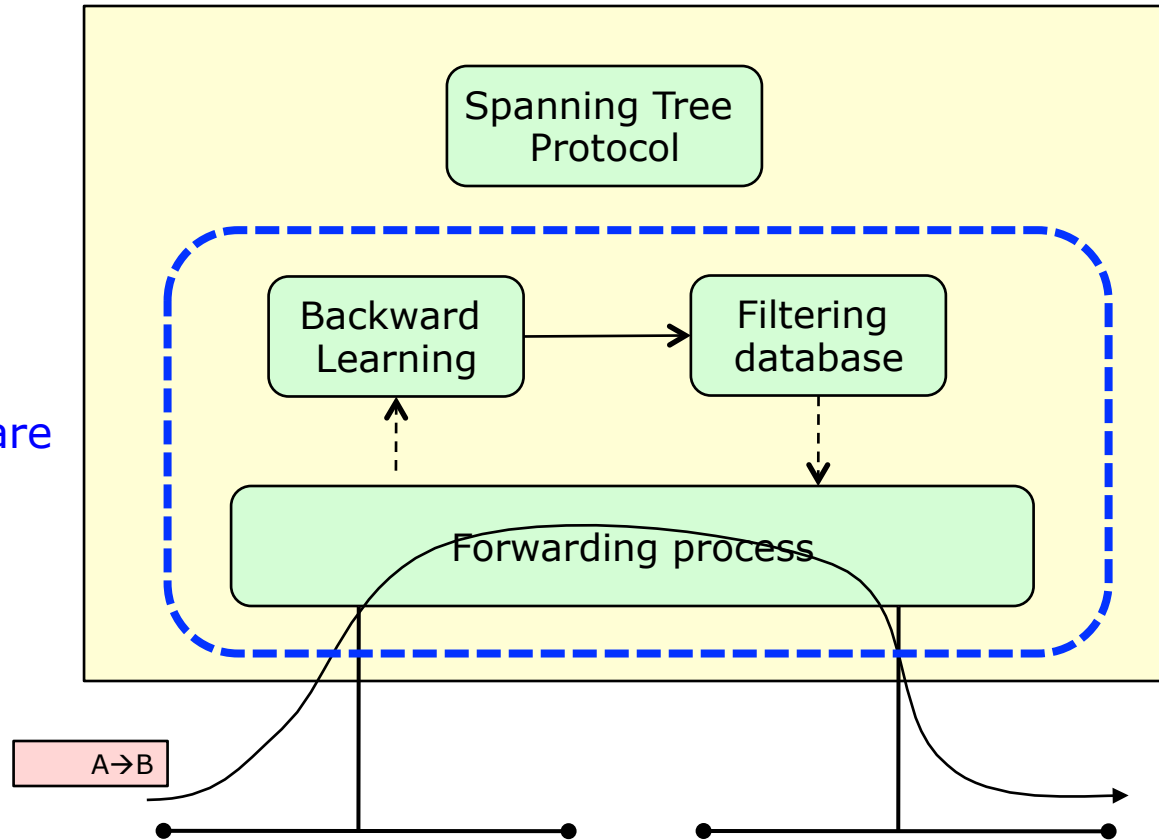
Bridges and switches (2)

■ Switch

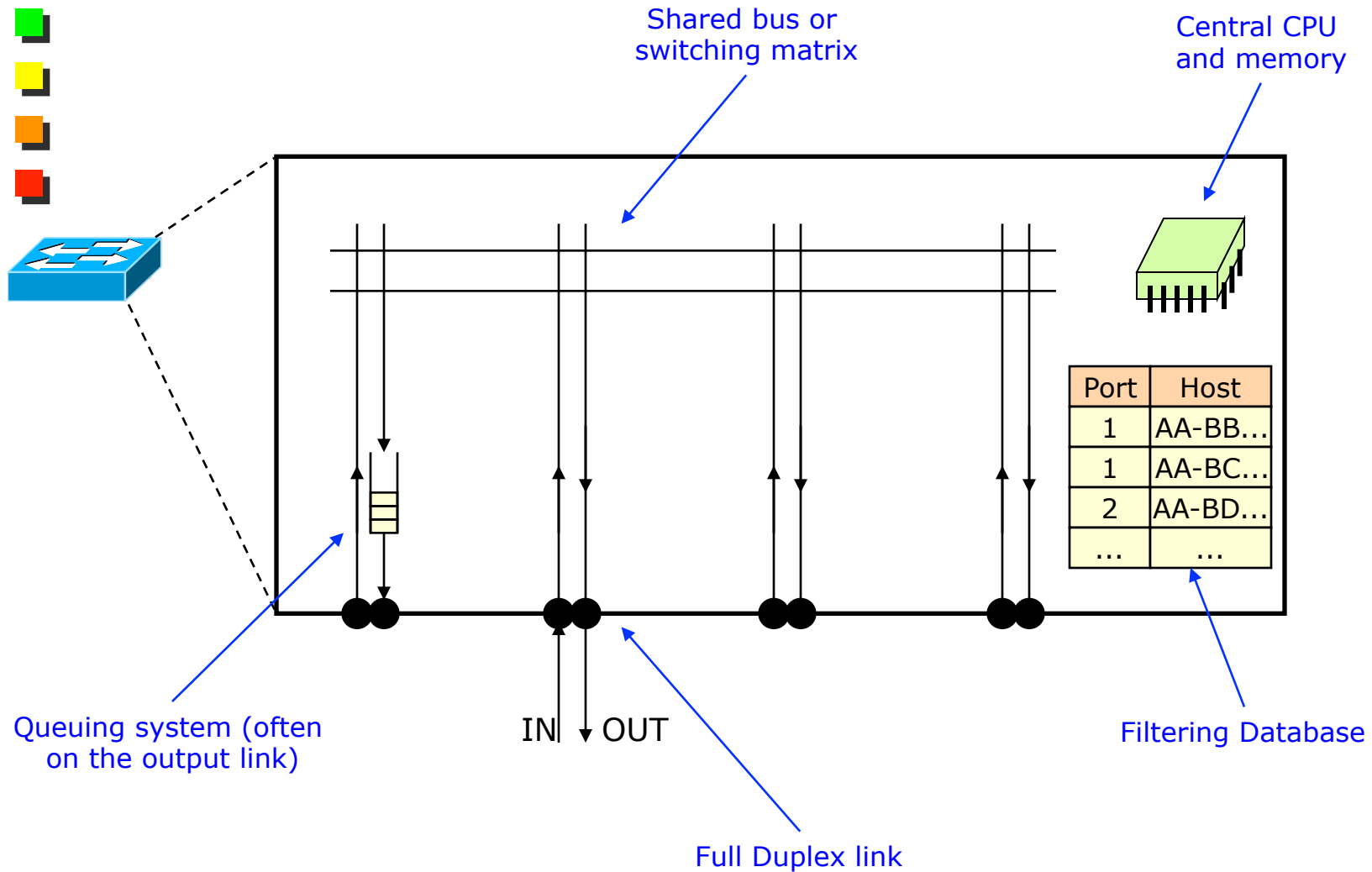
- Same device, different technology
 - Hardware based forwarding and learning
 - Lookup through CAMs (Content Addressable Memories)
 - Spanning Tree in software
 - 收斂 Convergence time in several seconds, hence hardware implementation is useless
 - Can implement a “cut-through” forwarding technology
 - A frame can be forwarded on the target port immediately after receiving the Destination MAC
 - The destination port must be free at that time
 - Faster than “store and forward”
 - Requires all ports operating at the same speed
- 

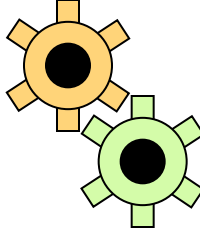
Switch architecture

Hardware

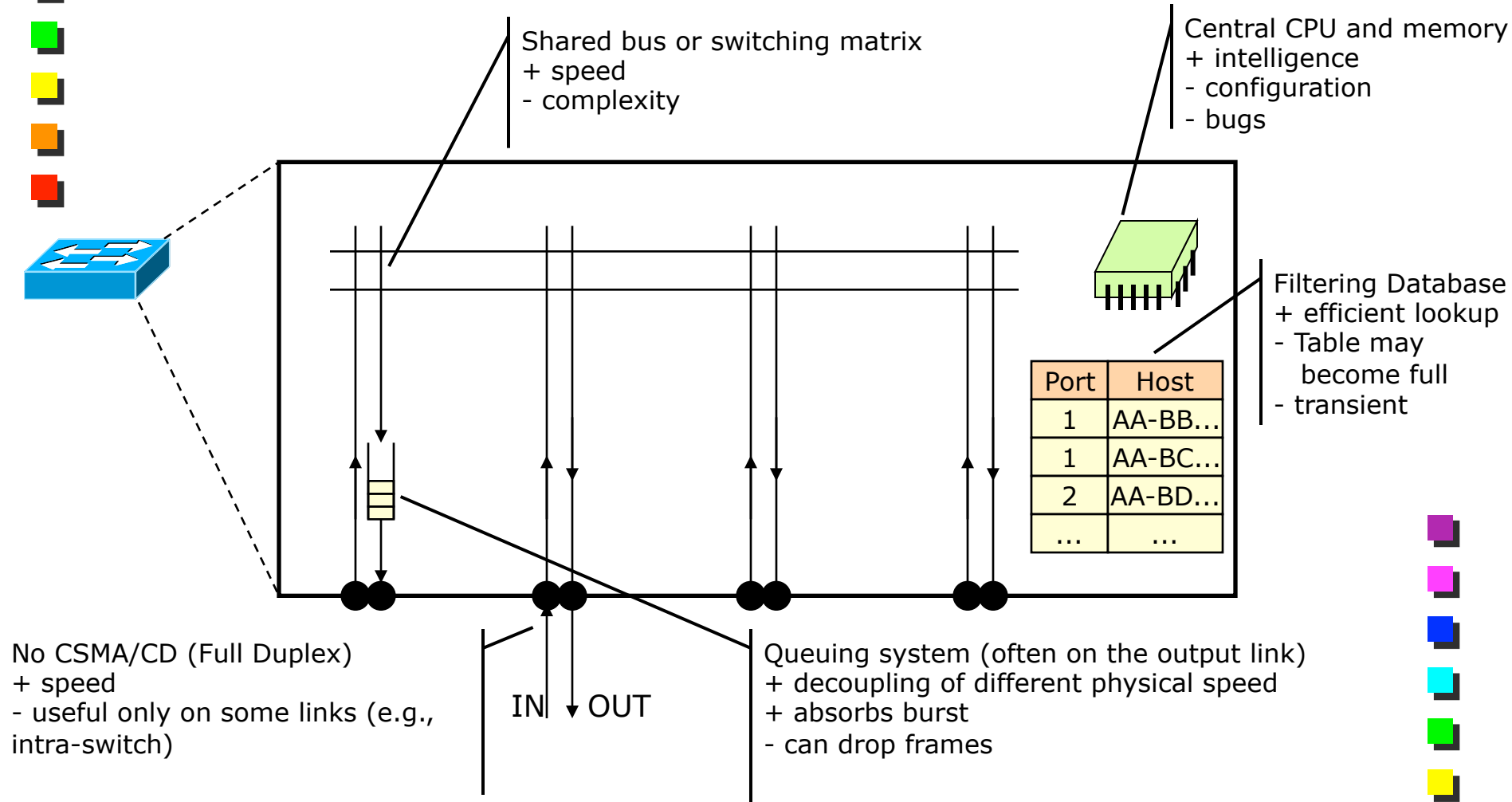


Switch internals





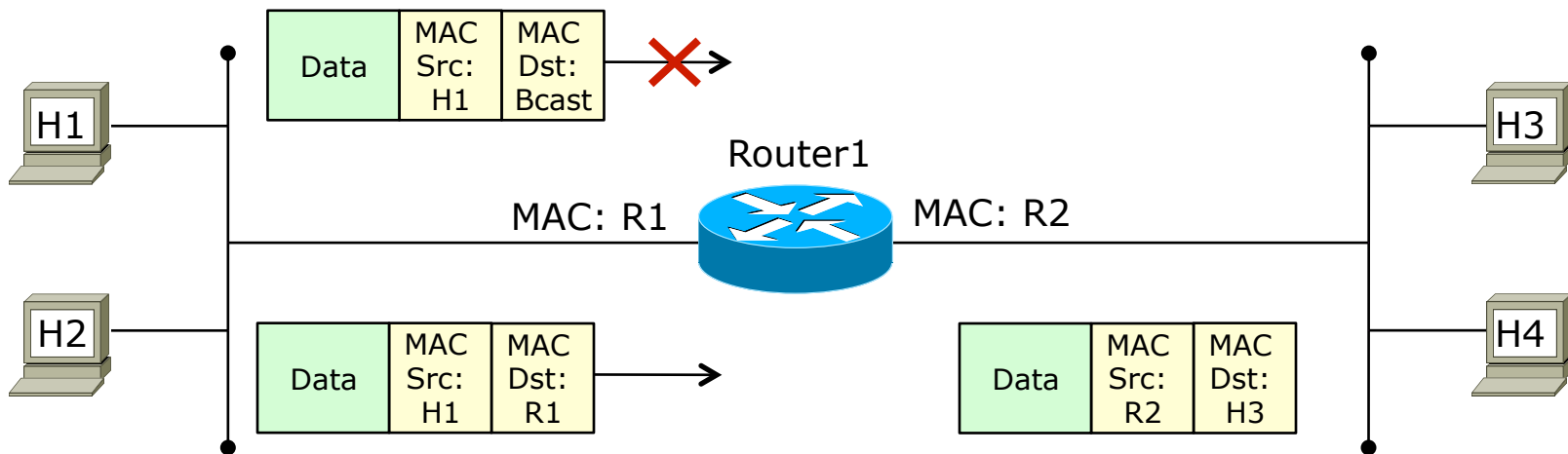
Switch internals



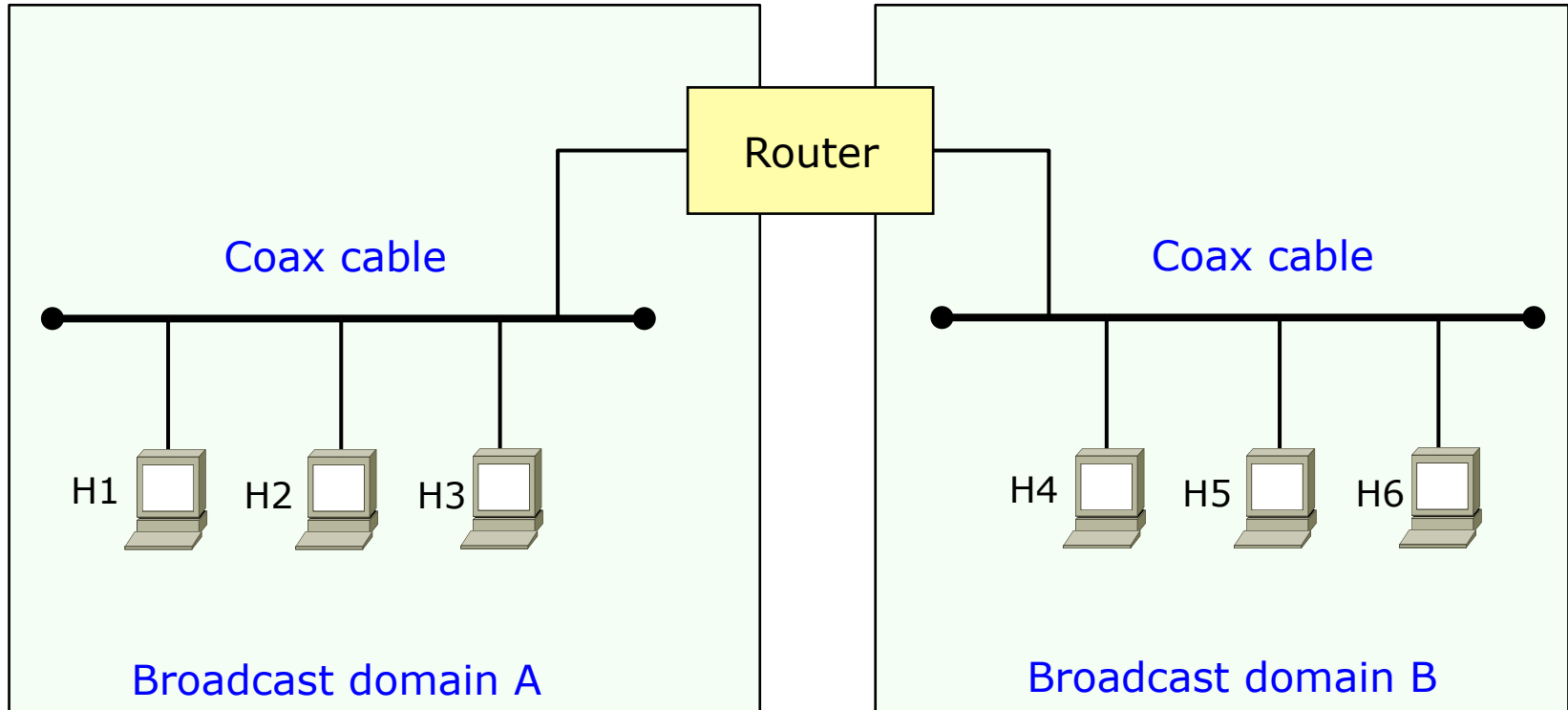
Routers

- L3 devices!

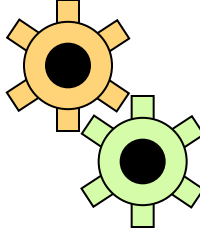
- Routers are not transparent with respect to MAC addresses
- Routers separate **broadcast domains**



Routers and broadcast domains

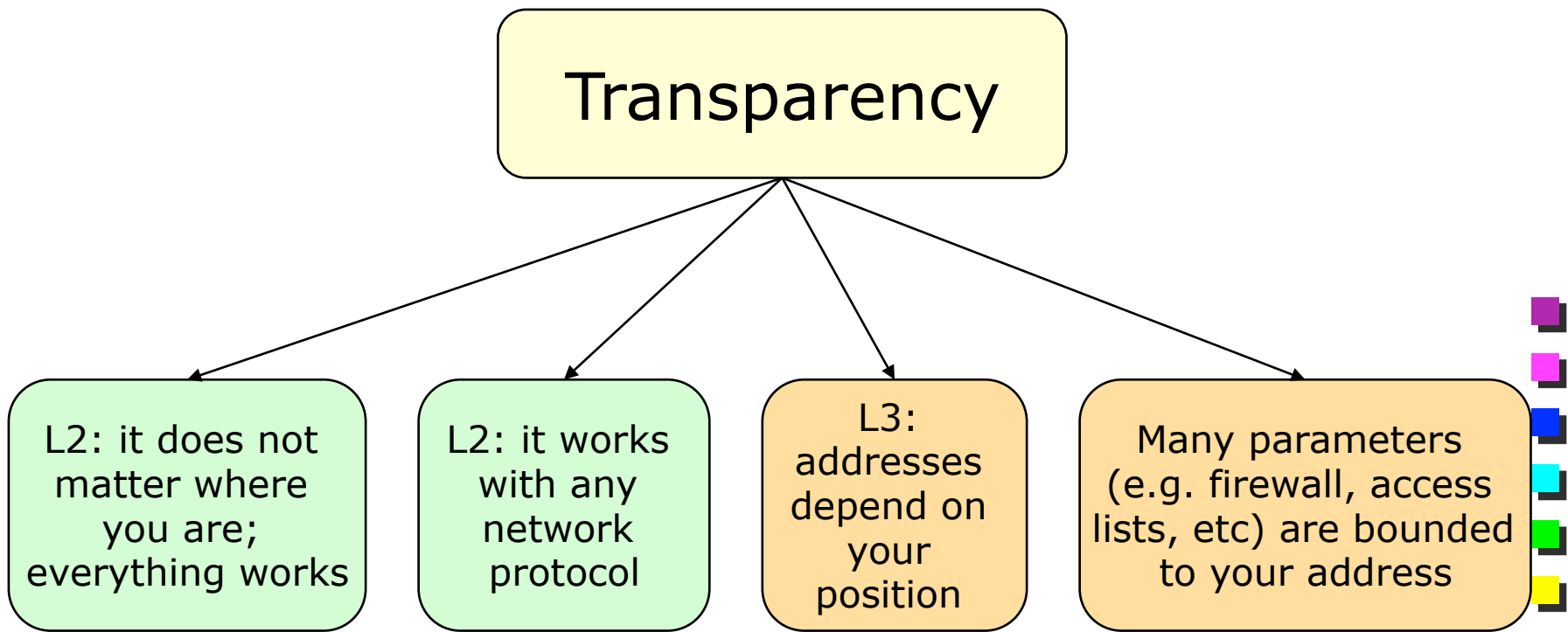


Different IP networks on the two interfaces of the router



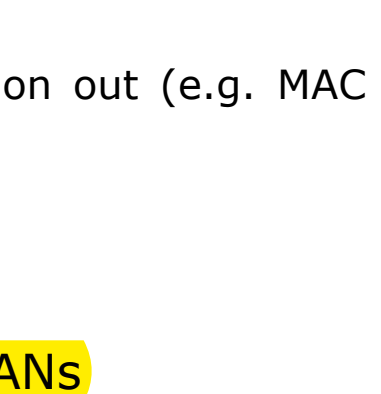
L2 or L3?

- So far, we concentrated on L2
- Shall we stay with L2 or better moving to L3?



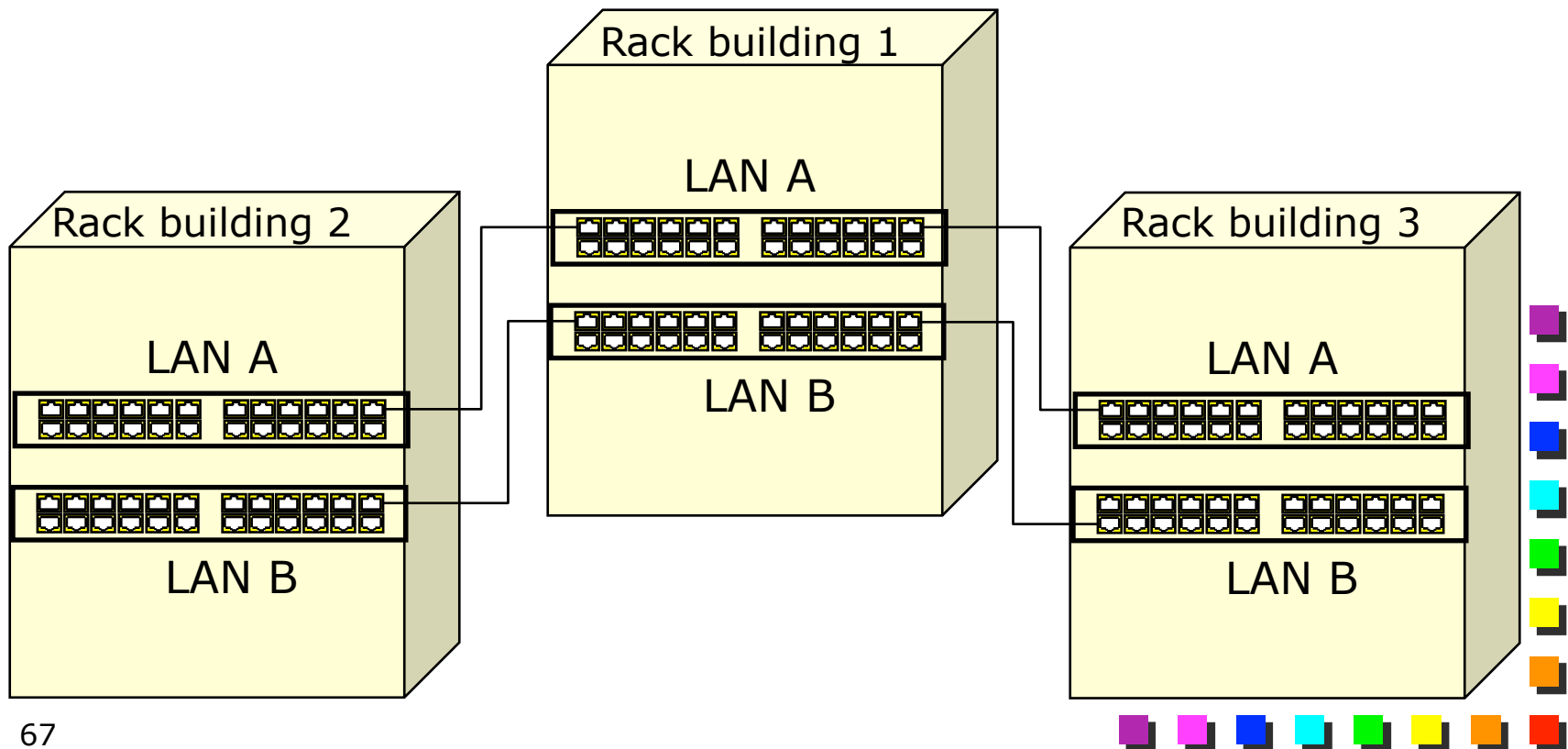


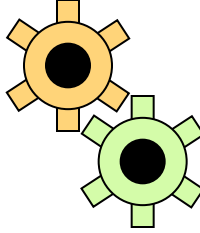
One or multiple LANs across a campus?

- Ok, so it's better to keep the L2 as long as we can
 - As far as the network is able to operate as a single L2 entity (remember scalability issues in L2 networks!)
 - But... a single, gigantic LAN, or multiple LANs?
 - Performance
 - A single LAN has too much broadcast traffic (not filtered by switches)
 - Flooded traffic (e.g. due to frequent STP reconfiguration)
 - Privacy, Security
 - Do not want a station to leak some information out (e.g. MAC Flooding attack)
 - Management
 - Smaller network, simple (and uniform) policies
 - Better to partition different users in different LANs
- 

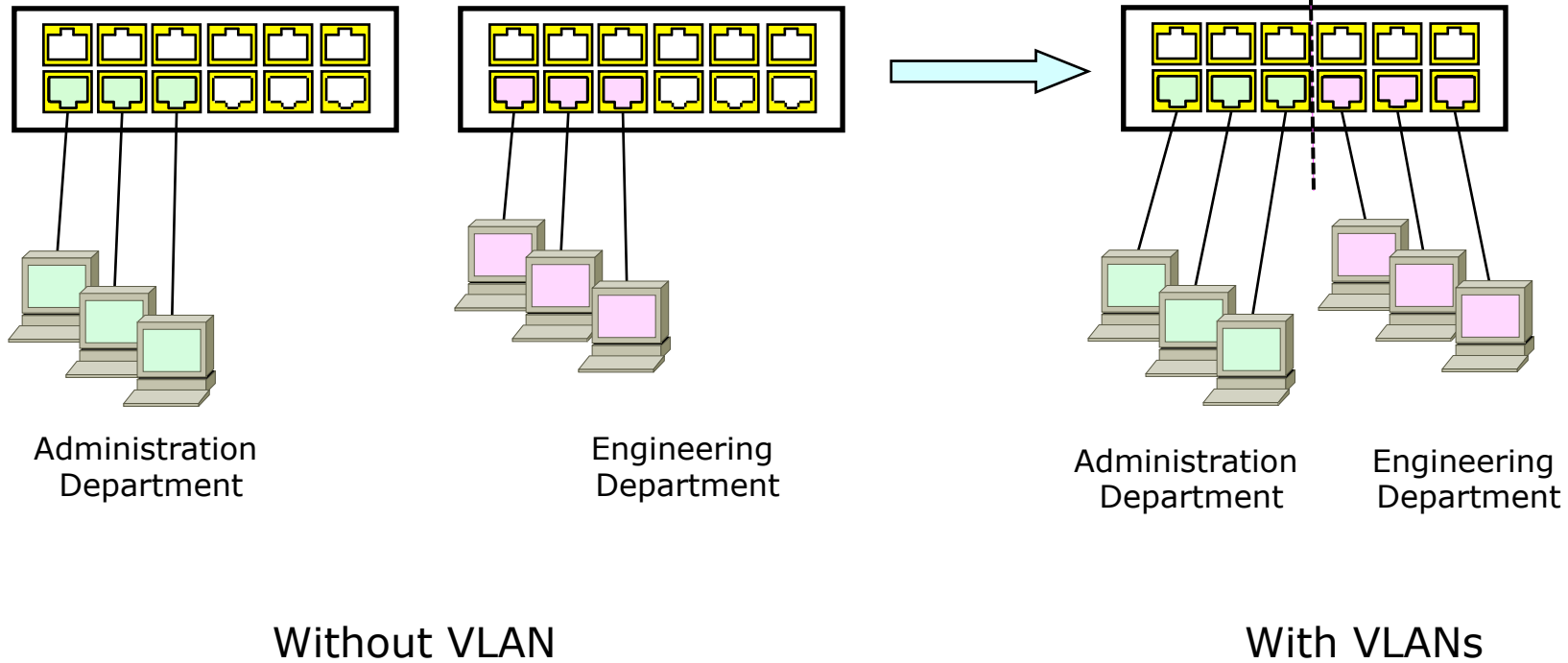
Multiple LANs across a campus: how?

- Different physical networks (full separation)
 - $N \text{ networks} = N \text{ links} + N \text{ devices}$
 - Waste of resources





Virtual LANs (1)

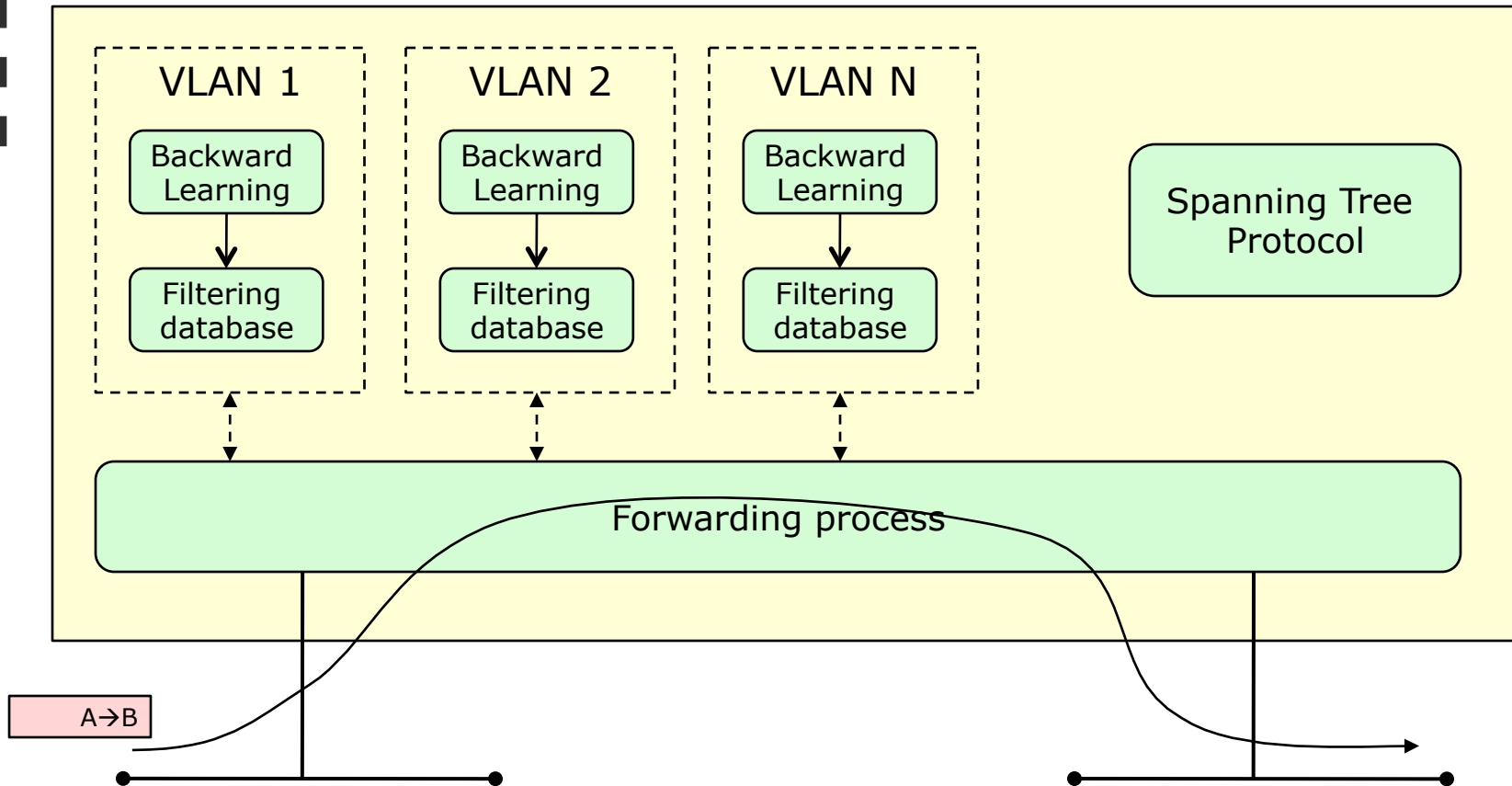




Virtual LANs (2)

- Single physical infrastructure
 - Same devices, same cabling
 - No switches in which only a few ports are used
 - No need to have multiple fibers (for different LANs) in the backbone
- Different LANs
 - Different broadcast domains
 - E.g., Ethernet frames cannot be propagated on another VLAN
 - No broadcast between LANs
 - No MAC flooding attacks
 - No ARP spoofing 電子欺騙
 - Created through a proper (logic) separation on switches
 - Intra-switch or inter-switch

VLAN: switch architecture



VLAN: forwarding database

MAC Filtering DB (VLAN1) MAC Filtering DB (VLAN2) MAC Filtering DB (VLAN3)

MAC	Port
H1	1
H4	4

MAC	Port
H2	2
H5	4

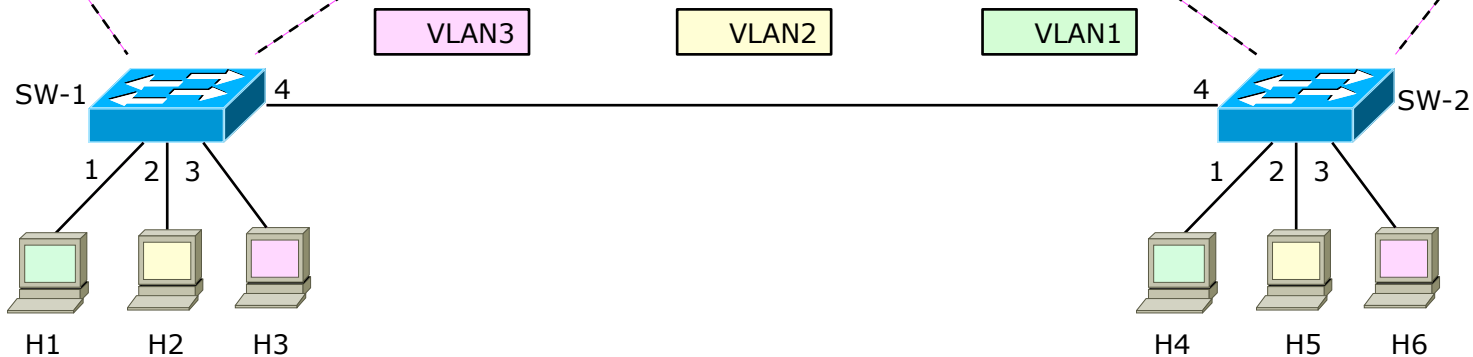
MAC	Port
H3	3
H6	4

MAC Filtering DB (VLAN1) MAC Filtering DB (VLAN2) MAC Filtering DB (VLAN3)

MAC	Port
H1	4
H4	1

MAC	Port
H2	4
H5	2

MAC	Port
H3	4
H6	3



Real implementations: unique filtering database (usually made with a TCAM, which is a single entity in the network device)



Interconnecting VLANs (1)

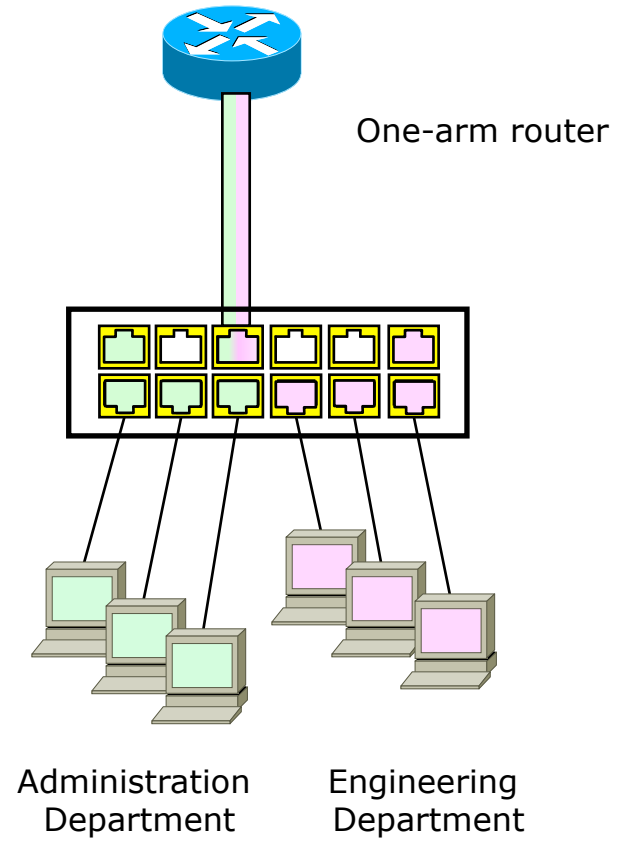
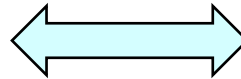
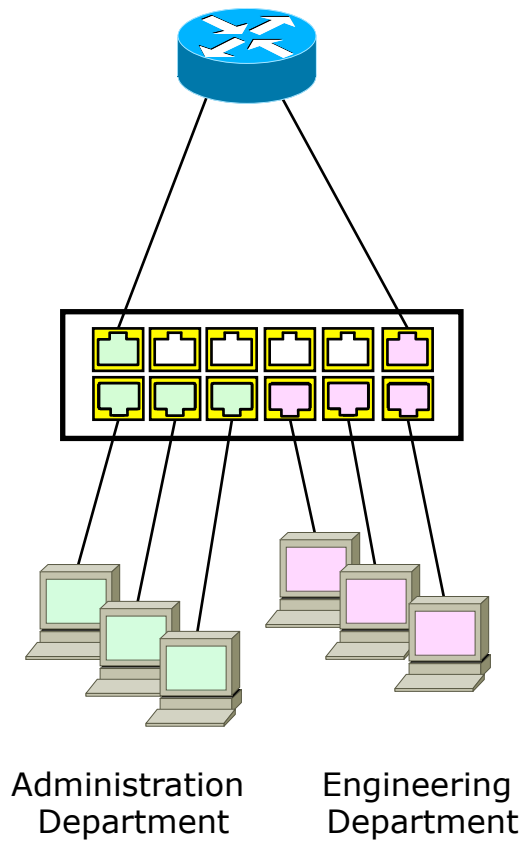
- L2 data cannot cross VLANs
 - An Ethernet station cannot send an L2 frame to another station in a different VLAN
 - VLANs are different broadcast domains

Beware:

L2 data cannot cross VLANs!



Interconnecting VLANs (2)



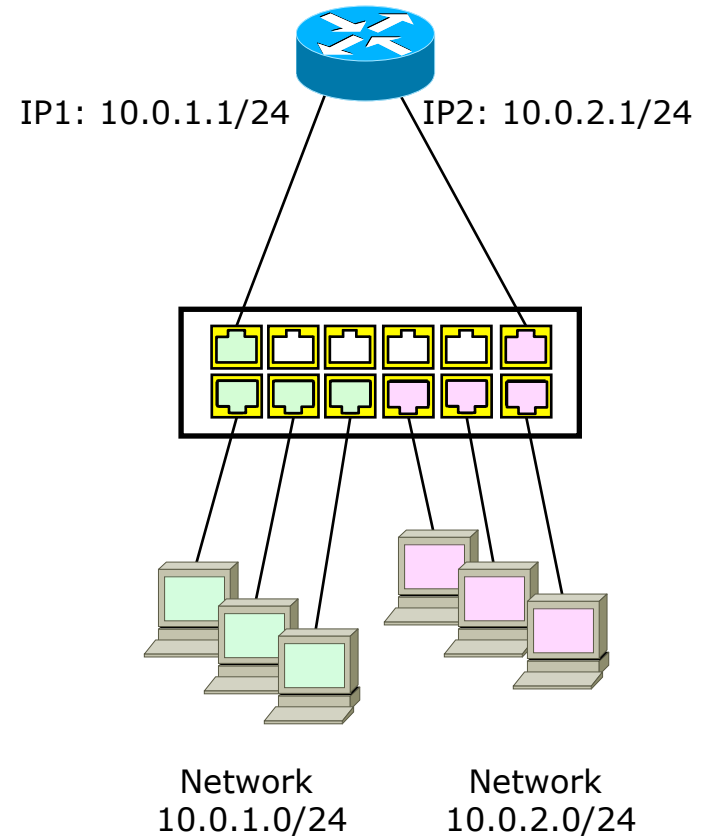
A vertical column of eight colored squares (purple, pink, blue, cyan, green, yellow, orange, red) on the left side of the slide.

Interconnecting VLANs (3)

- A router (i.e., device operating at layer 3) is needed
 - Lookup at layer 3 (e.g., IP destination address)
 - A router is often used to enforce L3 (or even L4/7) layer protection (e.g. firewall)
 - The original L2 header is thrown away and a new one is created with other MAC addresses (src/dst)

VLANs and IP addresses

- Broadcast cannot cross the VLAN boundaries
 - Cannot use ARP to resolve the MAC address in another VLAN
- Hosts in different VLANs must belong to different IP networks



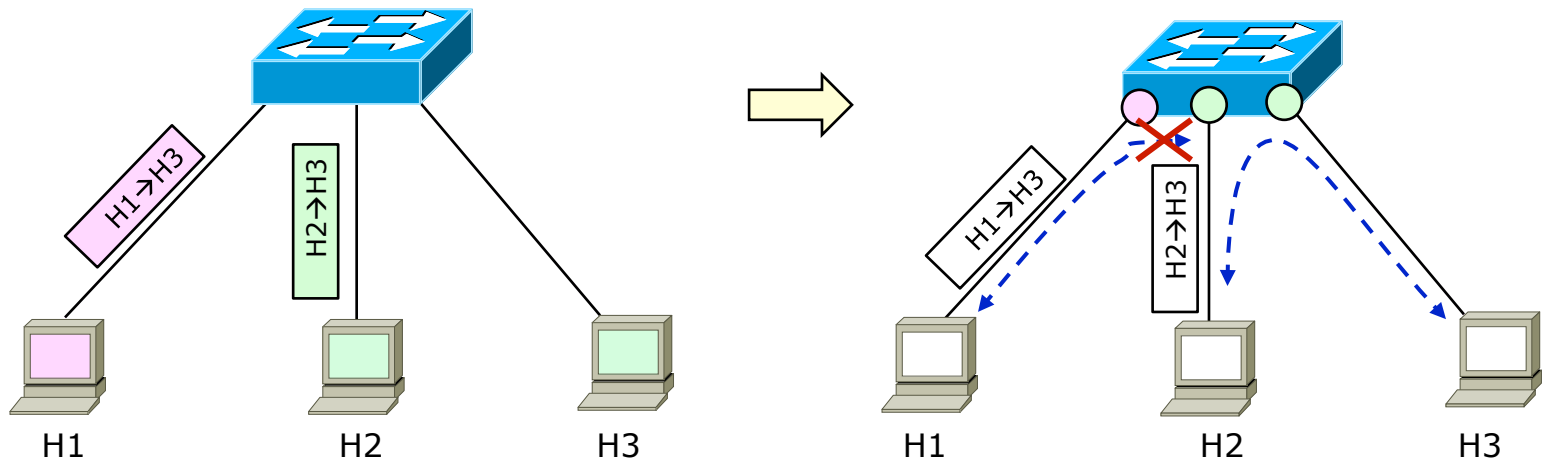
Associate frames to VLANs (1)

■ Problem

- How can we associate frames to VLANs?

■ VLANs on a single switch

- Simplest method: we can mark the ports on the switch
 - The received frame is associated to the VLAN the port belongs to
- Other methods exist
 - Presented later



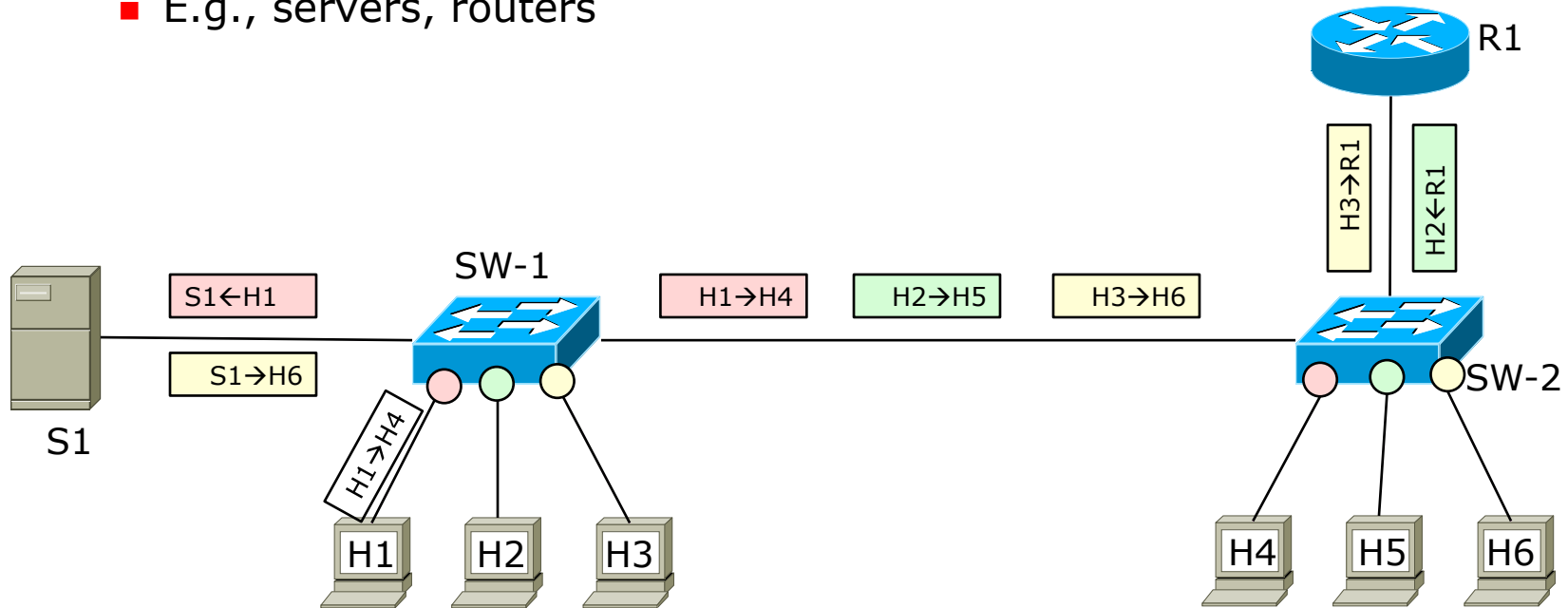
Associate frames to VLANs (2)

- VLAN on different switches

- Problem: how to distinguish which VLAN a frame belongs to, as there is a single link between switches?

- Same problem for devices that belong to different VLANs


- E.g., servers, routers



Note: the IDs in the frames are the MAC addresses of the involved stations

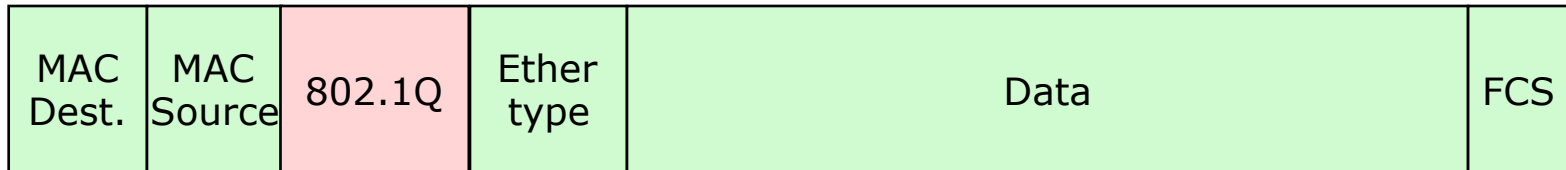


Associate frames to VLANs: tagging

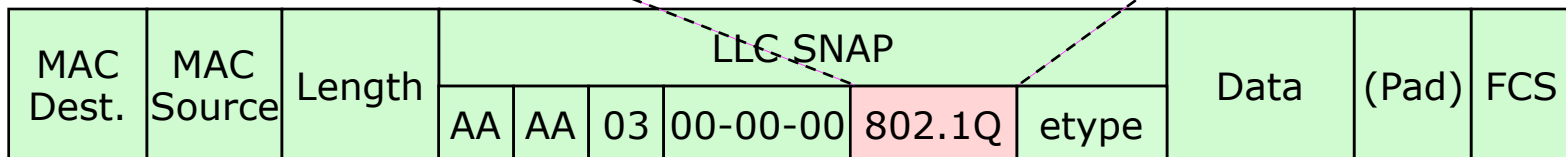
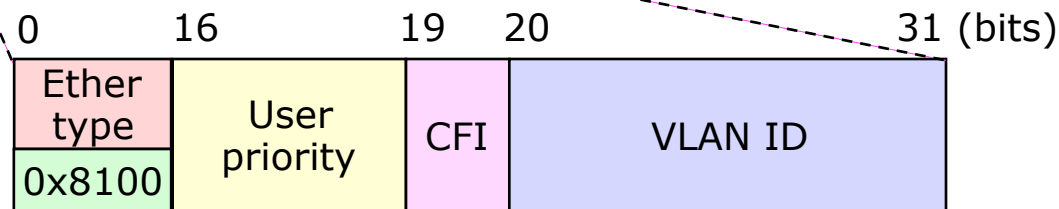
- Required only on links that transport traffic of different VLANs
 - Old method: Tunneling
 - An Ethernet (Token Ring or FDDI) frame is encapsulated into another Ethernet frame
 - Proprietary solutions
 - E.g., ISL (Inter-Switch Link) by Cisco
 - Frame Tagging
 - An additional header is added to the MAC header
 - Standardized by IEEE 802.1Q
 - 4 additional bytes added to the frame
 - Basically, VLAN-ID plus a bunch of other info
- 

IEEE 802.1Q Tag Encoding (1)

VLAN in Ethernet encapsulation (default)



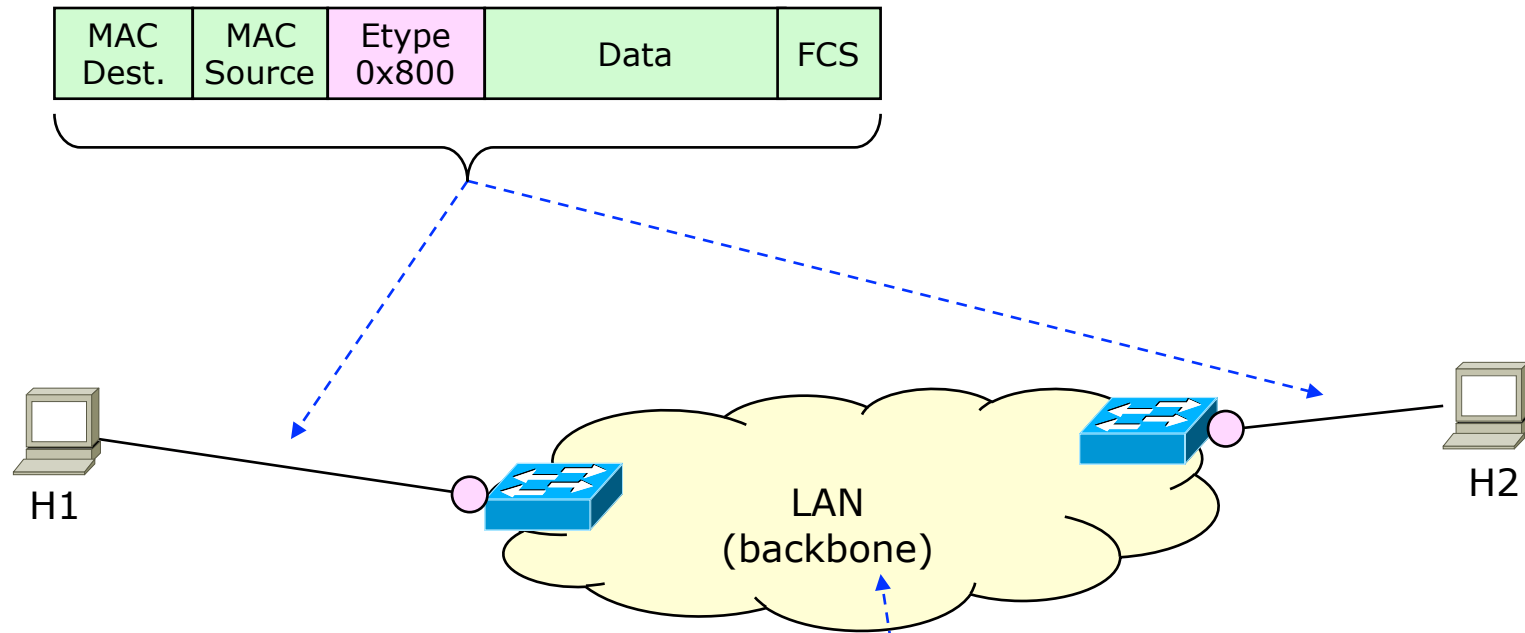
Ethertype for VLAN tagging



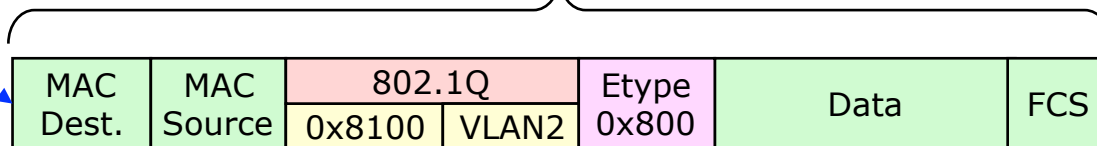
VLAN in IEEE 802.3 with LLC SNAP

Ethertype for VLAN tagging

IEEE 802.1Q Tag Encoding (2)

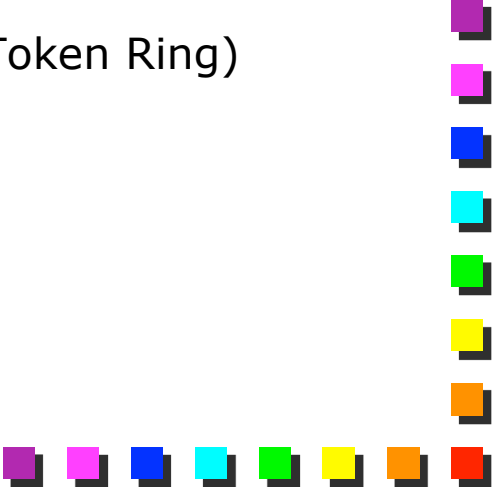


Frame is 4 bytes longer than the one generated by H1





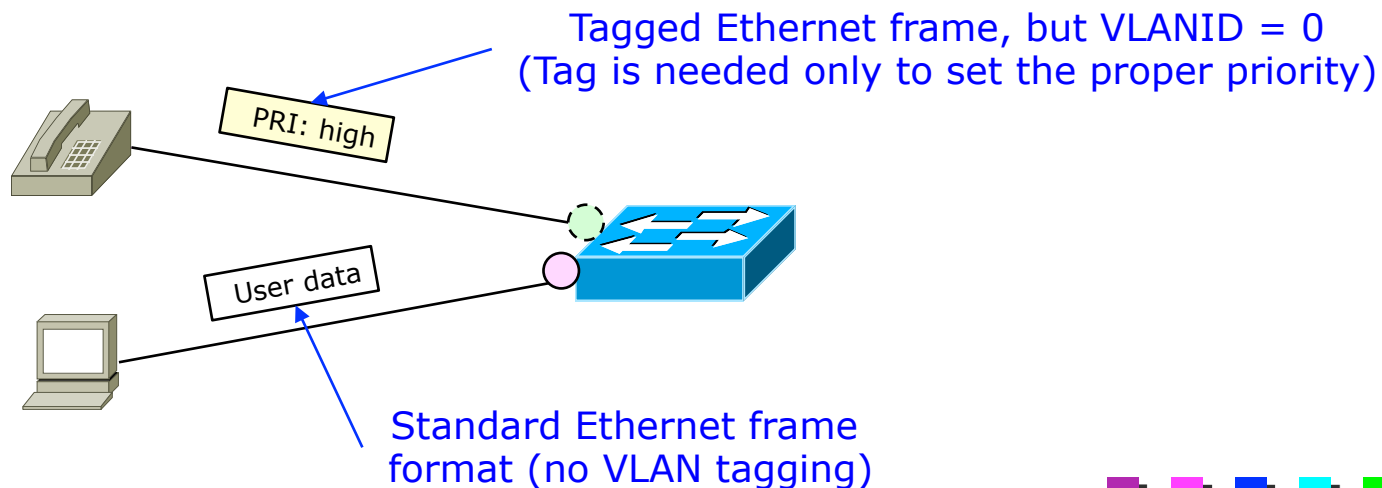
IEEE 802.1Q Tag Encoding (3)

- It can be encapsulated in either Ethernet (DIX) or any link layer using LLC SNAP
 - In both cases, it uses the Ethertype 0x8100
 - The frame has IEEE 802.1Q tag
 - Called TPID (Tag Protocol Identifier)
 - PCP (Priority Code Point)
 - Refers to IEEE 802.1p priority
 - CFI (Canonical Format Indicator)
 - "1": MAC address in non-canonical format (e.g. Token Ring)
 - Usually set to "0" (e.g., Ethernet)
- 

IEEE 802.1Q Tag Encoding (4)

■ VID (VLAN Identifier)

- Values 1- 4094
- Usually, "1" refers to the default VLAN
- 0xFFF: reserved
- 0: the frame does not belong to any VLAN (or I don't know which VLAN this frame belongs to)
 - Used in case the user just wants to set the priority for her traffic





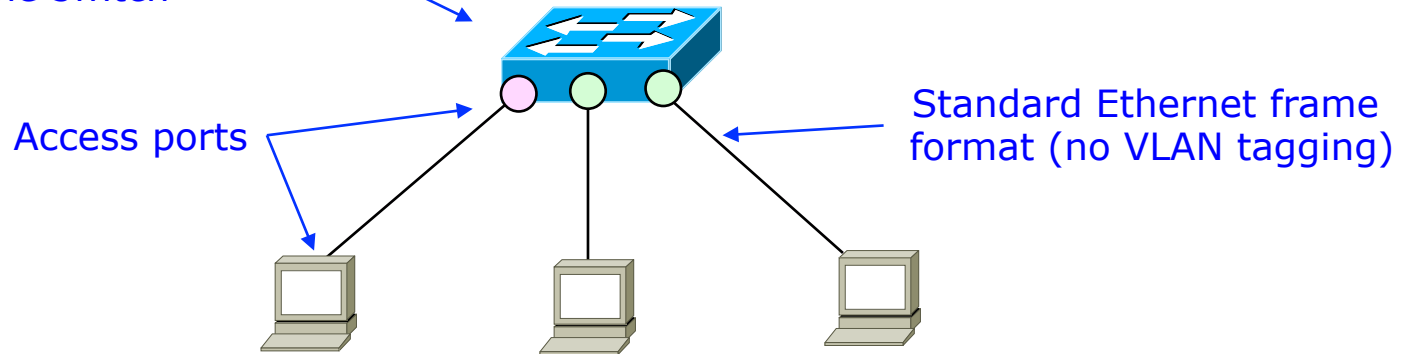
Modification to existing MACs

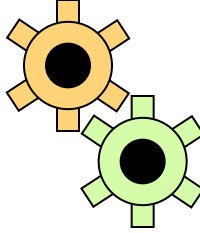
- Minor modifications
- New framing (for tagging) specified in 802.1Q
 - Independent from the technology of the Medium Access Control
- Maximum length of the frame has to be extended 4 bytes
 - E.g., Ethernet reaches 1522 bytes (from 1518)
 - Minimum length unchanged (still 64 bytes)

Link types: Access (1)

- Access Links receive and transmit *Untagged* frames
- Default configuration (on hosts, switches, servers, routers, etc)
- Usually used to connect end-stations to the network
 - Hosts do not need to change their frame format

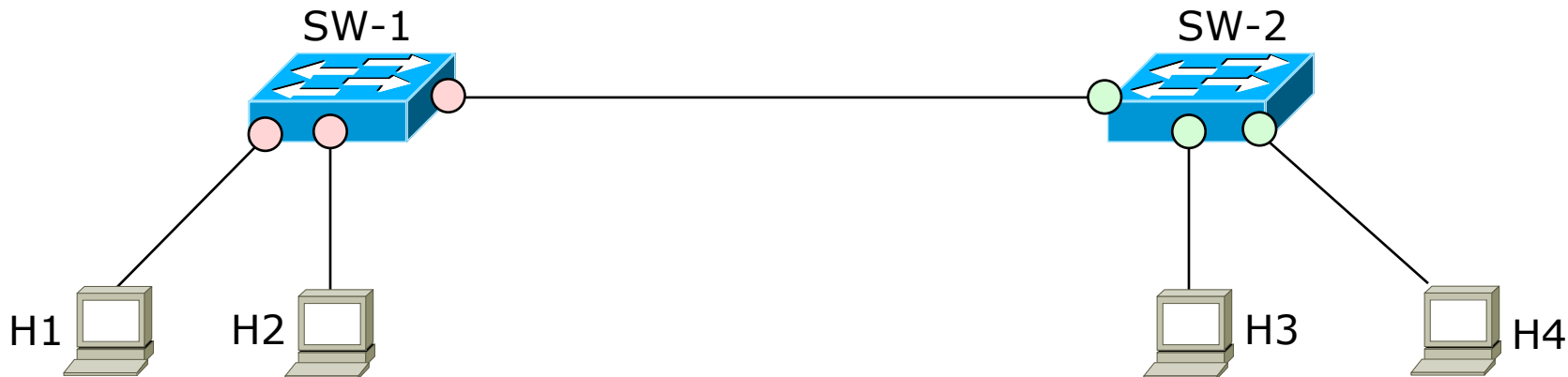
Incoming traffic is associated to the VLAN configured on the port of the switch





Link types: Access (2)

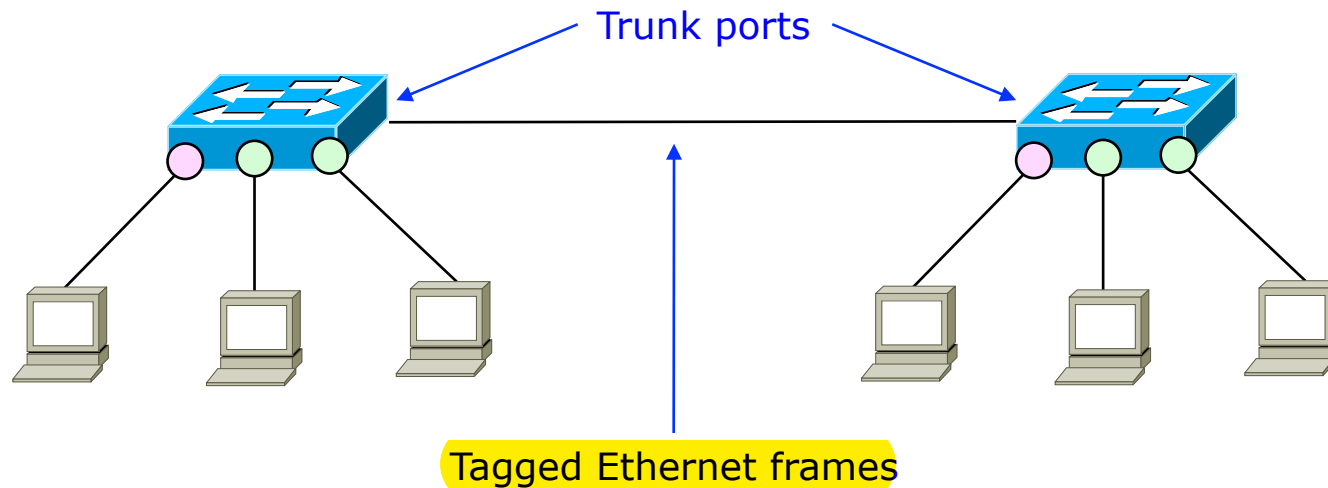
- Given the following network
 - All ports are configured in "access mode"
 - SW-1 is configured with the RED VLAN on all its ports
 - SW-2 is configured with the GREEN VLAN on all its ports
- Can host H1 communicate with host H4?



Yes, because values configured on access ports are not propagated outside the switch!

Link types: Trunk (1)

- Trunk links receive and transmit Tagged frames
- Must be configured explicitly 必须显式配置
 - Often used in switch-to-switch connections and to connect servers/routers





Link types: Trunk (2)

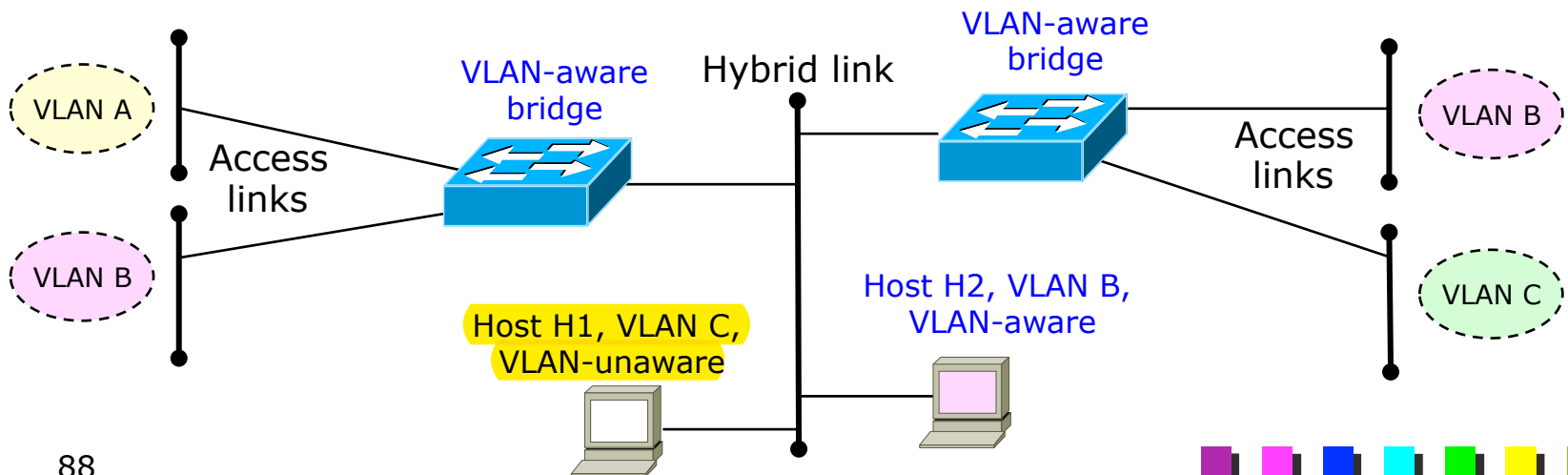
■ Tagging on trunk ports

- Different possibilities
 - Some switches tag the traffic belonging to all VLANs
 - Other leave the traffic belonging to VLAN 1 untagged
- A possible reason of incompatibility between network devices of different vendors

供应商

Link types: Hybrid

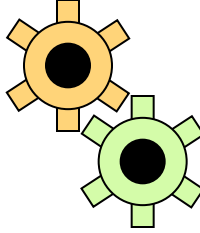
- Hybrid links accepts both tagged and untagged frames
 - Differentiates frame according to the "type" field (0x8100 or not)
 - Some hosts may not be fully operational (e.g. Station A cannot understand tagged traffic directed to it)
- Trunk links are usually also Hybrid links
- May be used on ports on which both hosts and servers / routers / switches are connected
- In any case, very uncommon nowadays





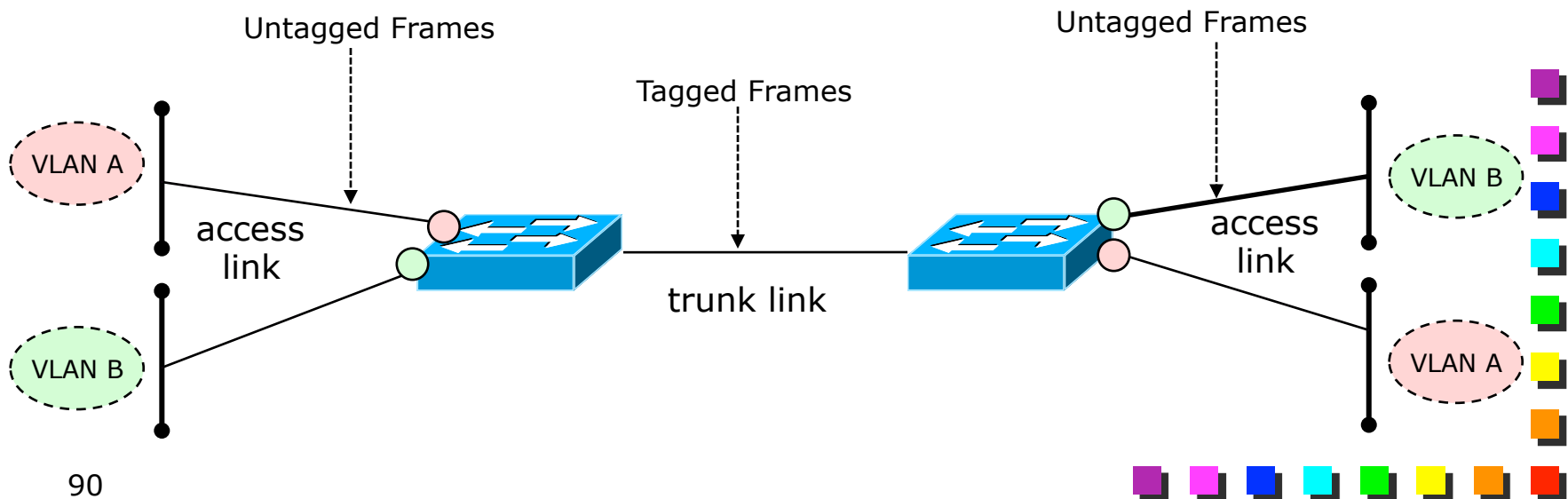
Assigning hosts to VLANs

- Different methods to associate devices to the proper VLAN
 - Port-based VLANs most use
 - Transparent assignment
 - Per-user assignment (802.1x)
 - Cooperative assignment don't trust users
- Note: a station can be associated also to multiple VLANs
 - E.g., required in case of servers, routers
 - In this case, trunk links are required on the device
 - Frames are tagged directly by the device
 - Fourth assignment method: Configuration of Trunk Interfaces
 - Can be seen as an extension of the Cooperative Assignment



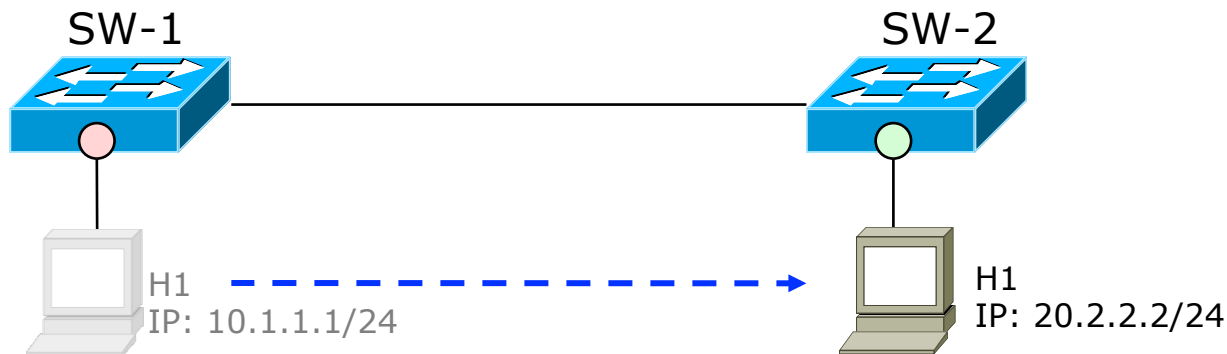
Port-based VLANs (1)

- Most common choice in current networks
 - Each port can be configured as either access port or trunk port
 - Each access port is associated to a single VLAN
 - Each trunk port is associated to a group of allowed VLANs
- Default: all ports in Access mode, associated to VLAN 1



Port-based VLANs (2)

- Completely transparent to the user
 - Association is done on the switch
 - Maximum compatibility, since there is no need to configure hosts
- Different VLANs (e.g., privileges) depending on the actual physical network socket we connect to
- No seamless mobility at L3
 - Host will change the IP address when moved into another VLAN





Transparent assignment

- New criteria in transparent assignment
 - Per L3 protocol (802.1v; no longer useful)
 - Per MAC address
 - Configuration problems
 - Keep MAC database aligned (new host, host with new NIC card, ...)
 - Network administrator has full control on association user-VLAN
 - Allows seamless mobility
- Mainly historical


Per user-assignment (802.1x)

- 802.1x is a standard that enables the network port on the switch only if the user authenticates successfully
- Since the switch knows who is attached to the port, it can assign the proper VLAN to the user
 - E.g., if the switch detects that user U1 connects to the switch, it enables VLAN1
 - Assignment is *per-user*, not *per-host*
 - It looks similar to the per-port assignment, but the coloring is done based on the UserID





Cooperative assignment (1)

- Also known as “anarchic” VLAN assignment
 - Users keep control of the VLAN assignment
 - User sets the VLAN on the network card
 - Allows seamless mobility
 - User will attach always to the same VLAN anywhere in the campus
 - What about a user joining the wrong VLAN?
 - Negligence or bad will
 - Used mostly on devices than must be part of different VLANs
 - E.g. routers, servers
- 

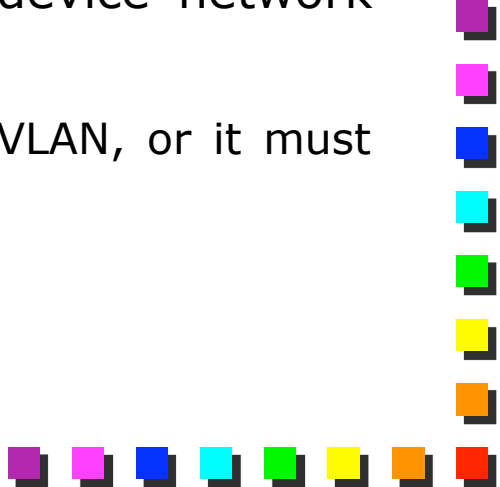


Cooperative assignment (2)

■ Requires

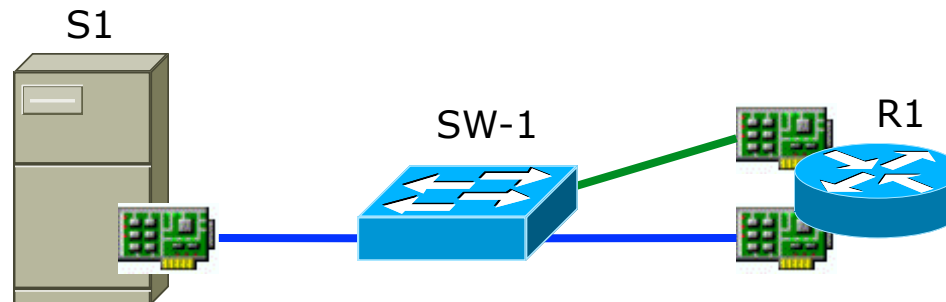
- The (manual?) configuration on all the PCs
- The usage of trunk interfaces
 - Frames are tagged by the user, which sets the right VLAN-ID in outgoing frames
 - In any case, the port on the switch has to be configured anyway with the list of allowed VLANs
 - Often we use "VLAN allow all"

■ Two way of configuring this feature on the device network card

- Depends if the device has to support *a single* VLAN, or it must belong to *multiple* VLANs
- 

Cooperative assignment: single VLAN per NIC

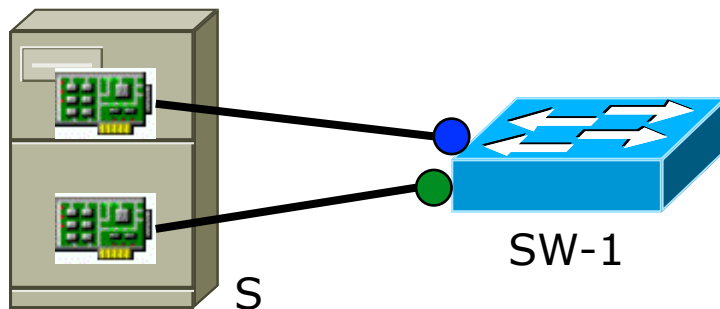
- Simple association of VLAN tagging to the incoming/outgoing traffic
 - Incoming/outgoing traffic is generated with 802.1Q tagging
 - Only one VLAN-ID per NIC interface is allowed (and specified by configuration)
 - Allowed on almost all network cards (e.g., the ones we have in our PCs)
 - We may have multiple cards in case multiple VLANs are required
 - Barely used



Coop. assignment: multiple VLANs per NIC (1)

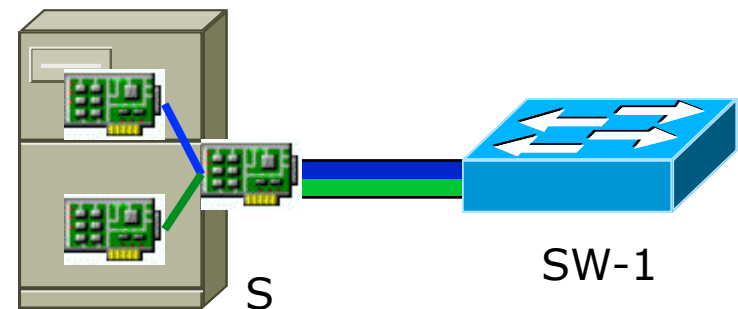
■ Without VLANs in the host

- Two network interfaces
- Each one with its own IP configuration
- Each one belongs to a different LAN
 - E.g., receives only the broadcast associated with that VLAN



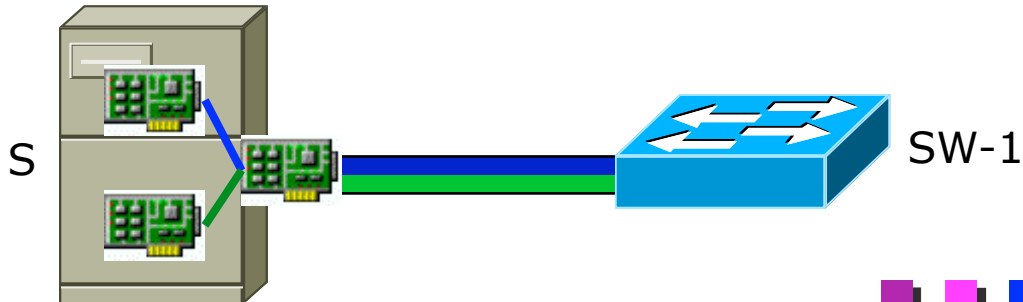
■ With VLANs in the host

- We need to create exactly the same environment that was available before VLANs
- We had two NICs before, we need two NICs now as well

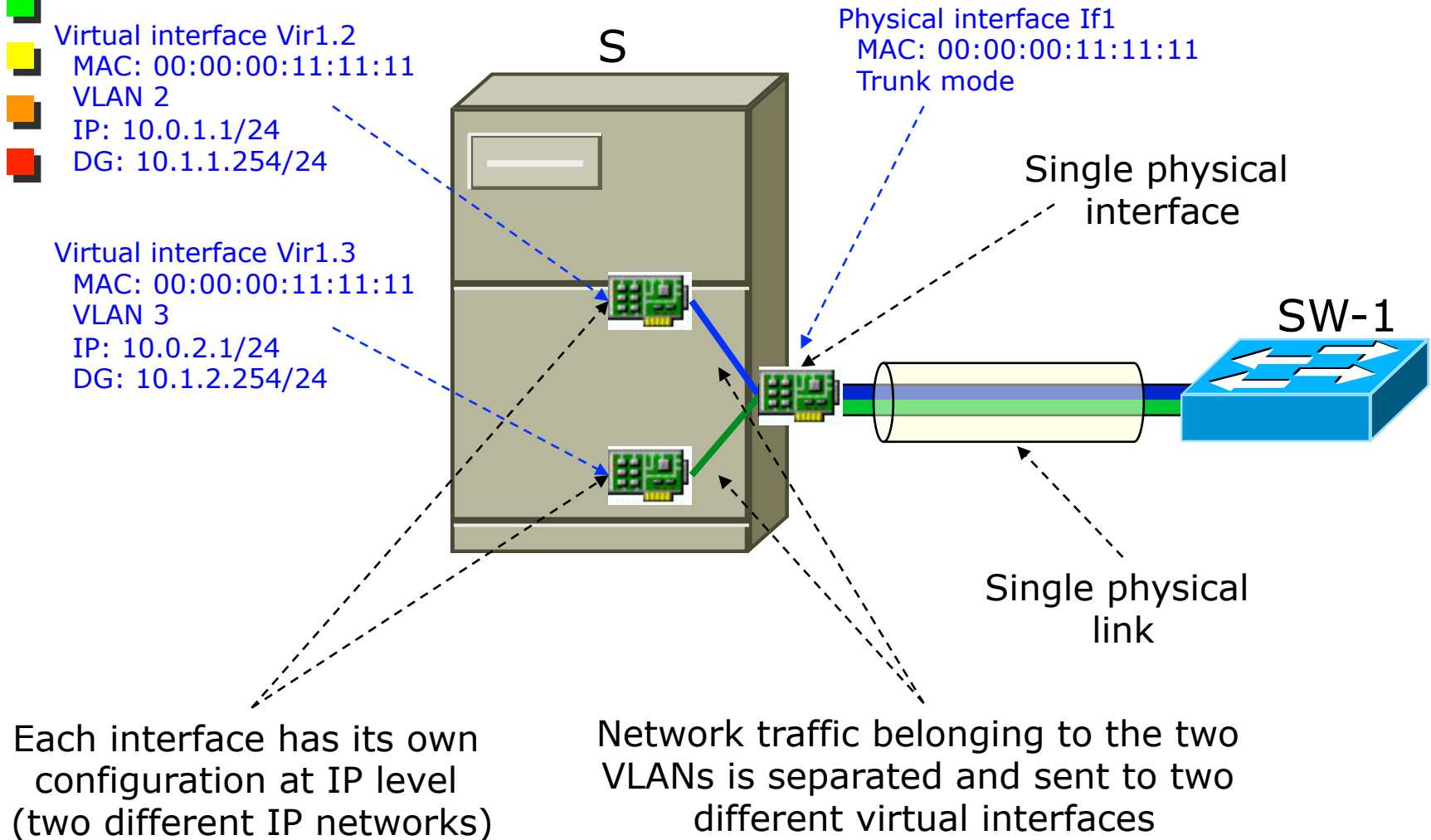


Coop. assignment: multiple VLANs per NIC (2)

- Requires the usage of virtual NICs
 - Multiple virtual network interfaces are created
 - Each one with its L3 configuration (e.g. IP address) and VLAN-ID
 - Only one VLAN-ID is allowed per virtual card
 - A maximum of N VLANs are allowed (N = number of V-NICs)
 - Widely used; mostly on servers and routers
 - Explicit support required from the NIC driver *and/or* the Operating System
- Important: IP addresses associated to the interfaces (either real or virtual) **must** belong to different IP networks

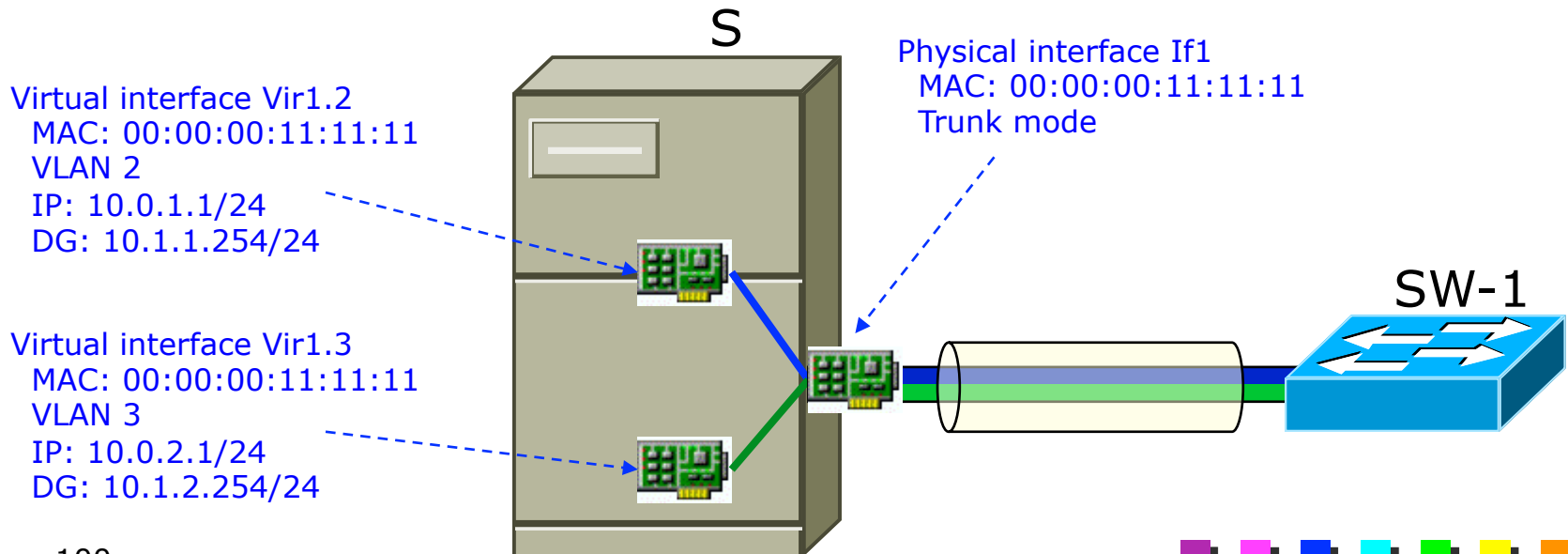


Trunk Interfaces and IP configuration



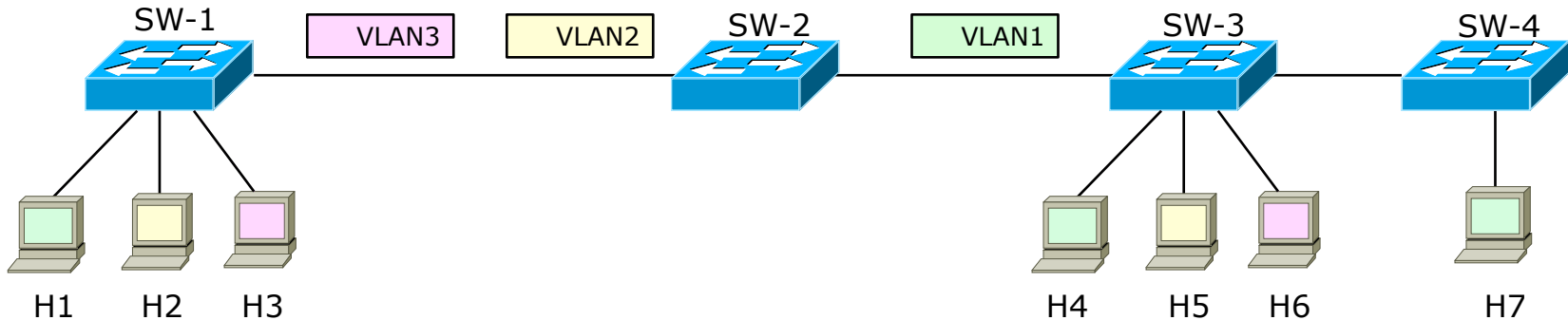
Note: duplicated MAC addresses

- Please note that duplicate MAC addresses are
 - Very common in modern LANs
 - Another common situation is host virtualization (e.g. virtual machines)
 - Do not cause troubles as soon as they belong to different VLANs
 - Switches MUST handle the filtering databases of different VLANs as distinct entities



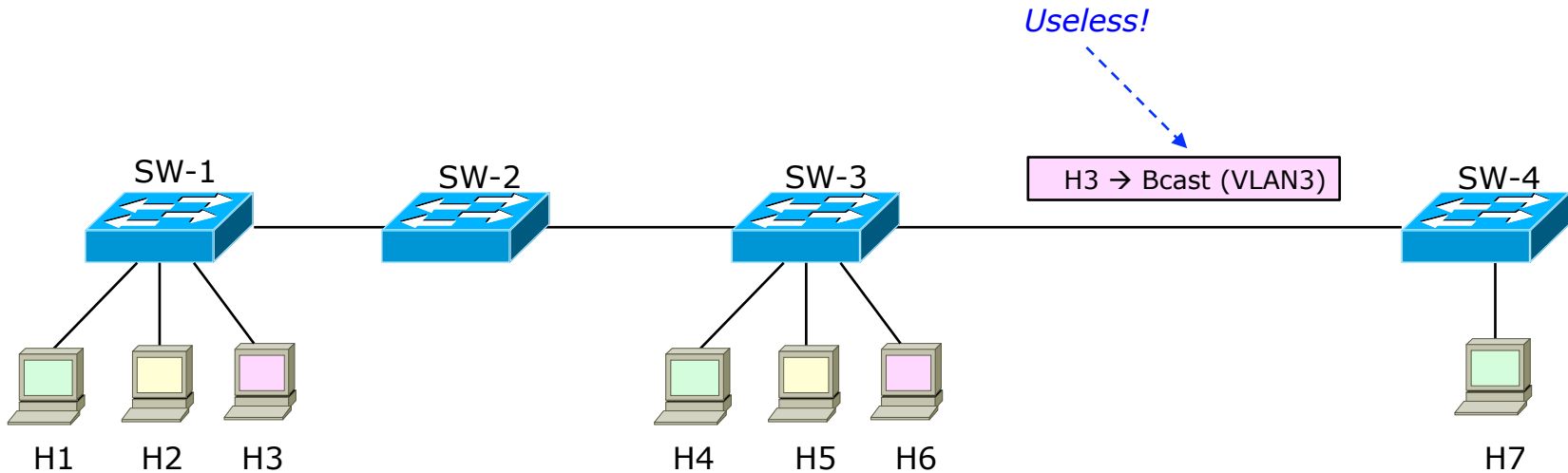
Assigning VLANs to trunk links (1)

- Necessity to know which VLANs are handled on a given trunk link / switch
 - The switch needs to create the proper number of filtering DB
 - How can SW-2 know that it will have to forward VLANs 1-3?
- Possibility to optimize the number of filtering DB on the switch
 - E.g., FilteringDB for VLANs 2,3 are not needed on SW4
 - Useful to reduce the number of MAC entries on the switches



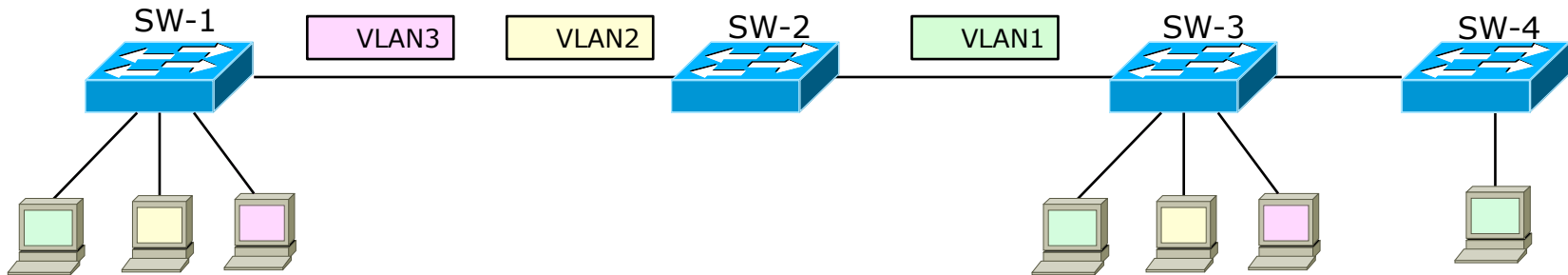
Assigning VLANs to trunk links (2)

- Possibility to optimize broadcast traffic
 - Avoiding to send *broadcast/flooded* traffic belonging to a VLAN on a switch where no such VLANs are present
 - Unicast (not flooded) traffic is always optimized by the filtering database



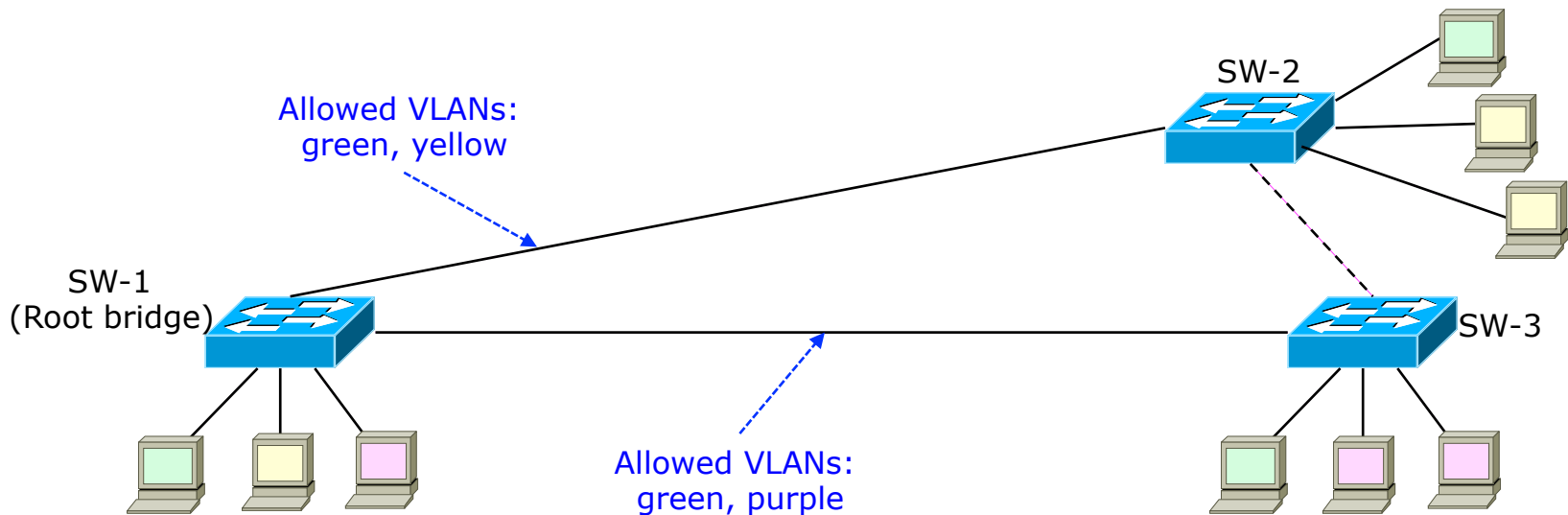
Assigning VLANs to trunk links (3)

- The idea: let each switch to know which VLANs are active on its ports
- Three solutions
 - Manual configuration
 - Proprietary mechanisms
 - GVRP




VLANs in the backbone: manual configuration

- Used in most networks
- Usually, VLANs are configured explicitly on each switch
 - Possible problems (related to STP) in case you want to optimize trunk ports and filter useless VLANs out
 - What about if the link between SW-1 and SW-2 is turned off?
 - Better to allow all VLANs on all links and avoid optimizations



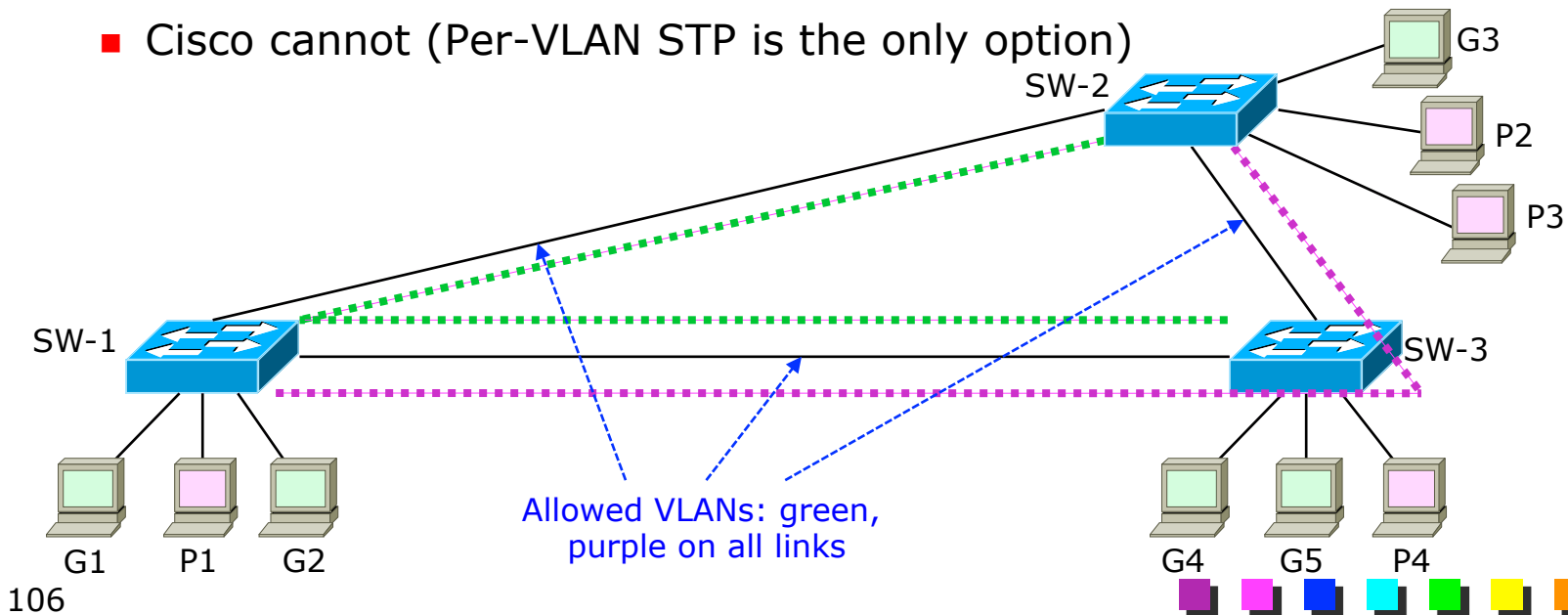


VLANs in the backbone: GVRP

- It propagates info about required VLANs on all the switches
 - Prunes switches that are not interested by some VLANs from the tree of that VLAN
 - Can filter the broadcast traffic of some VLANs on some switches
 - Handy (because automatic), but not widely used
 - It inserts a new level of intelligence in switches
 - Configuration required
 - New software (i.e. bugs)
 - Is it really needed (especially if you want to have a robust network)?
- 

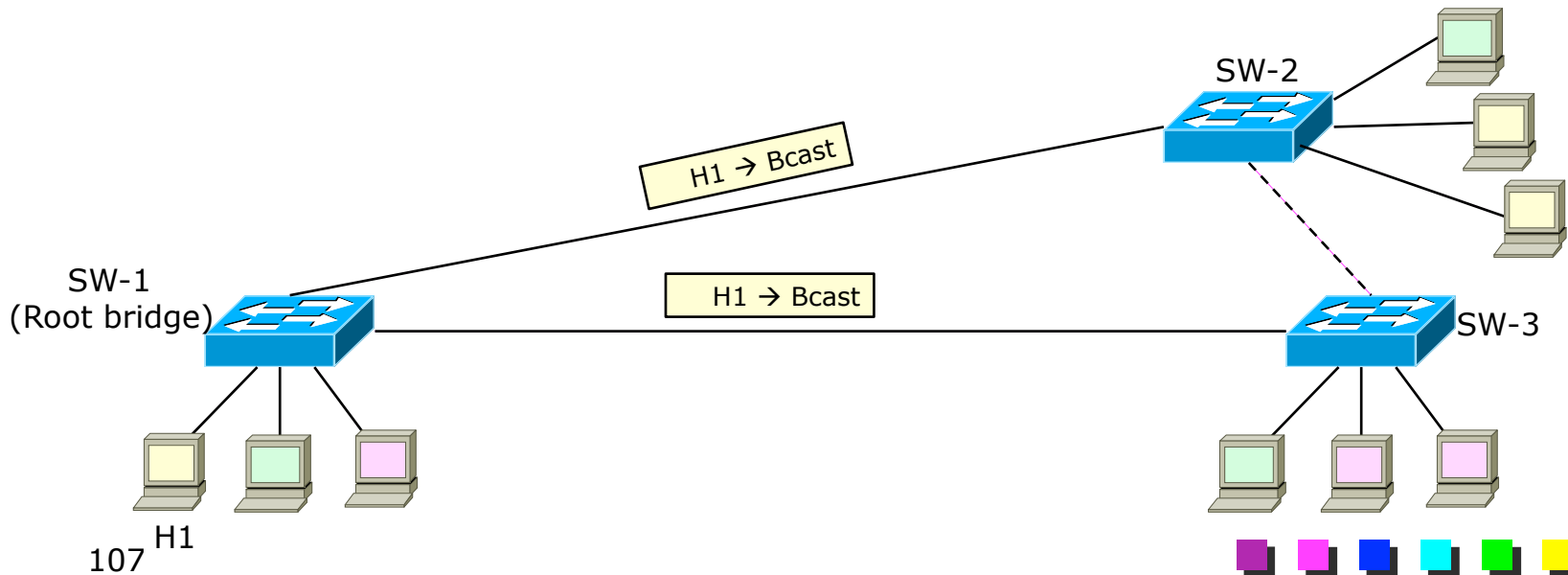
VLANs and Spanning Tree

- In theory, they are completely independent
 - First, Spanning Tree is computed in order to disable loops
 - Then, VLANs are used on the resulting topology
 - Unique forwarding tree for all the VLANs
- Almost all vendors offer Per-VLAN Spanning Tree
 - Most vendors can turn back to an unique STP via configuration
 - Cisco cannot (Per-VLAN STP is the only option)




VLANs and network isolation (1)

- Network isolation is not complete, even with VLANs
 - Although frames cannot cross the border of a VLAN, links are shared, hence a problem on a *link*, caused by the traffic of one VLAN, may affect other VLANs



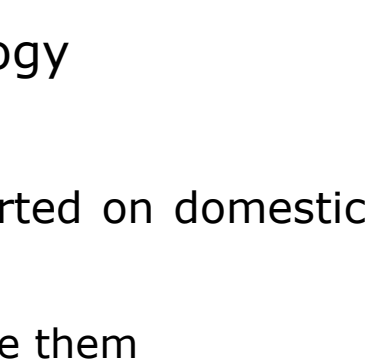


VLANs and network isolation (2)

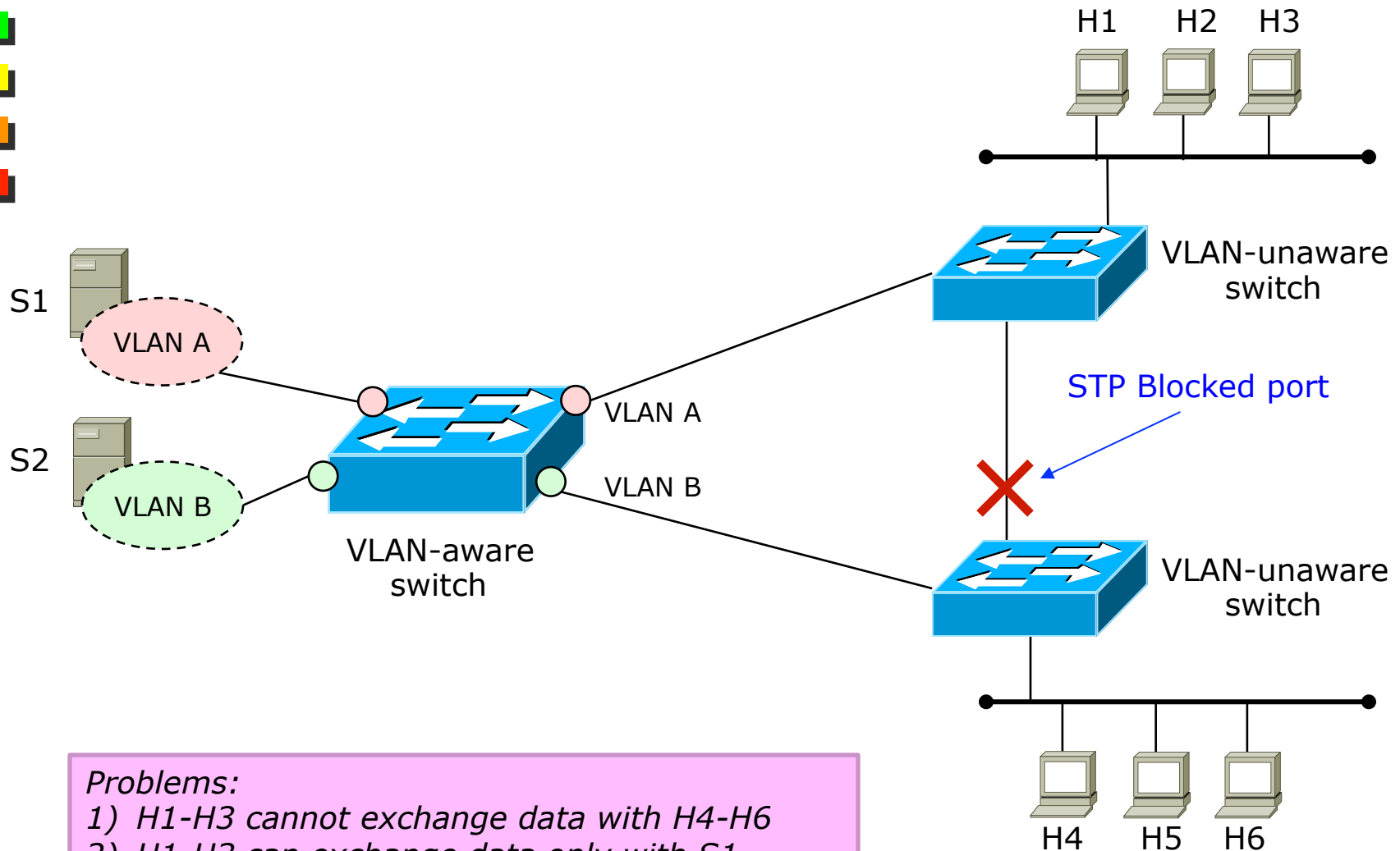
- For example, VLANs do not protect from broadcast storms
 - In fact, *broadcast traffic* is sent on the entire network
 - Except on the edge ports, since those are assigned to a specific VLAN
 - A trunk link may be saturated by a broadcast storm on a VLAN
 - Other VLANs do not receive that broadcast but...
 - ... the trunk link is congested and it may be unable to transport the traffic of other VLANs
 - Per-VLAN QoS may be required
 - E.g., “Round-robin” service model based on VLAN ID, which guarantees a minimum amount of bandwidth to each VLAN
- 



VLANs and network switches

- Two types of switches
 - VLAN-Aware: handle tagged and untagged frames
 - VLAN-Unaware: do not accept tagged frames
 - May discard frames (if too big)
 - Low-end devices
 - Availability on the market
 - Almost all professional products can handle VLAN tagging
 - Almost all domestic products do not have VLAN support
 - VLANs are no longer a “plug and play” technology
 - STP is (with some limitations)
 - This is one of the reasons VLANs are not supported on domestic switches
 - Typical users are not skilled enough to configure them
- 

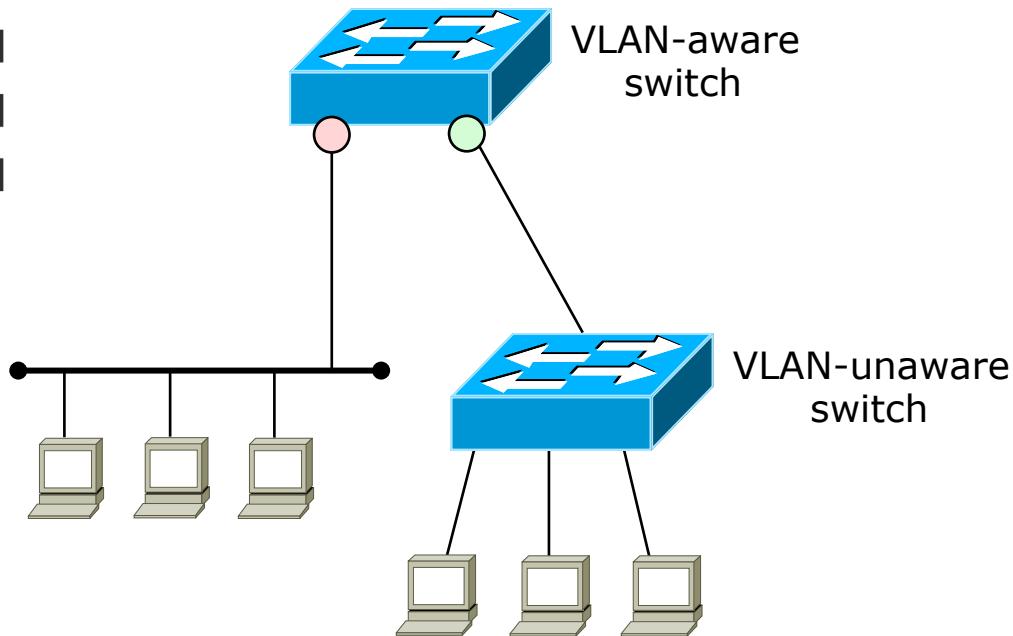
Mixing VLAN-aware/unaware switches (1)



Problems:

- 1) H1-H3 cannot exchange data with H4-H6
- 2) H1-H3 can exchange data only with S1
- 3) H4-H4 exchange data only with S2

Mixing VLAN-aware/unaware switches (2)



VLAN-unaware switches may be OK in the access side (e.g., in order to add new ports), provided that all clients belong to the same VLAN

Corollary

It is pretty common to have VLAN-unaware switches in corporate networks.

Network managers typically deploy only professional switches (with VLAN support) but often end users have some limitations (e.g., necessity to attach multiple hosts on a single network socket) and tend to sort those problems out by themselves, which usually means they buy the cheapest switch on the market, which does not have VLAN support.

Therefore, it is important that the network manager takes into account those situations (even if he does not know exactly where those switches may be installed) in order to prevent possible misbehaviour of the network.



Configuring VLANs on Cisco switches (1)

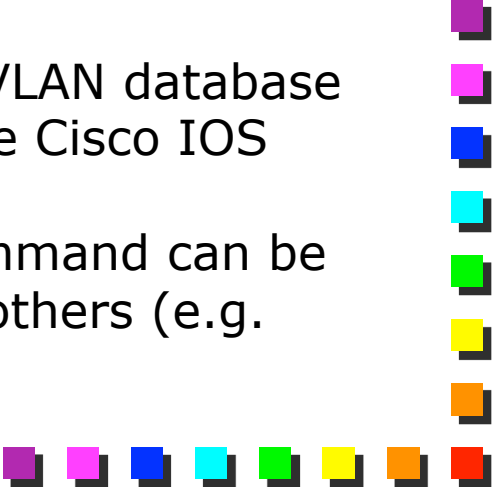
■ VLAN creation

```
Switch# vlan database
Switch(vlan)#vlan 2 name Administration
VLAN 2 added:
      Name: Administration

Switch(vlan)#exit
APPLY completed.
Exiting....
switch#
```

Note: the command for adding an entry in the VLAN database changes according to the different version of the Cisco IOS and given device in use.

In more modern devices, the *vlan database* command can be issued also in standard configuration mode. In others (e.g. Cisco 6500) the command is even different.






Configuring VLANs on Cisco switches (2)

■ VLAN port association

- Default behavior: a port is considered Access and associated to a default VLAN
- The switch has a VLAN-unaware behavior

```
Switch# configure terminal
Switch(config)#interface FastEthernet 0/1
Switch(config-if)#switchport access vlan 2
Switch(config-if)#exit
Switch# show vlan brief
```

VLAN	Name	Status	Ports
1	default	active	Fa0/2, Fa0/3, Fa0/4
2	Administration	active	Fa0/1





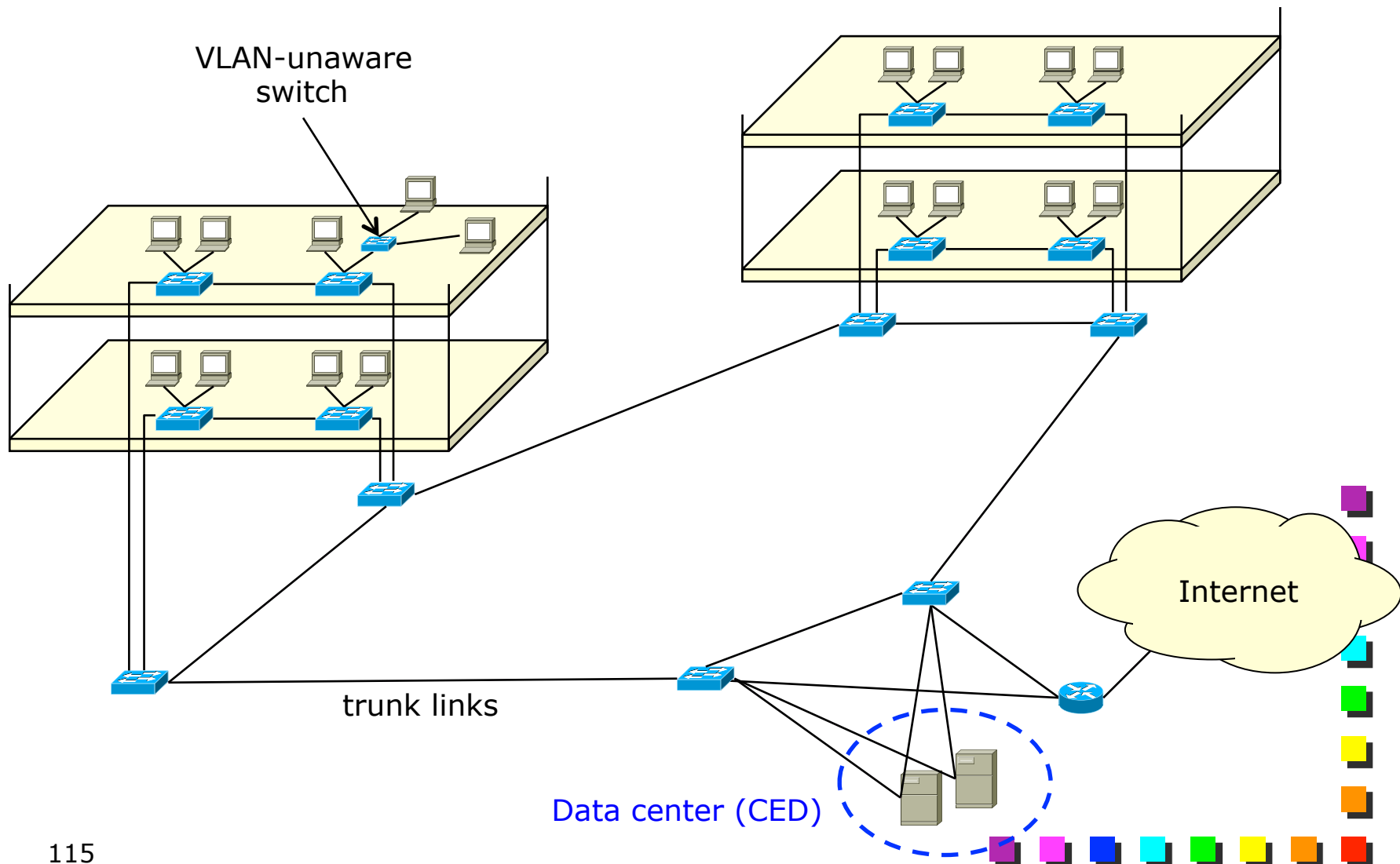
Configuring VLANs on Cisco switches (3)

- Configuration of the trunk port

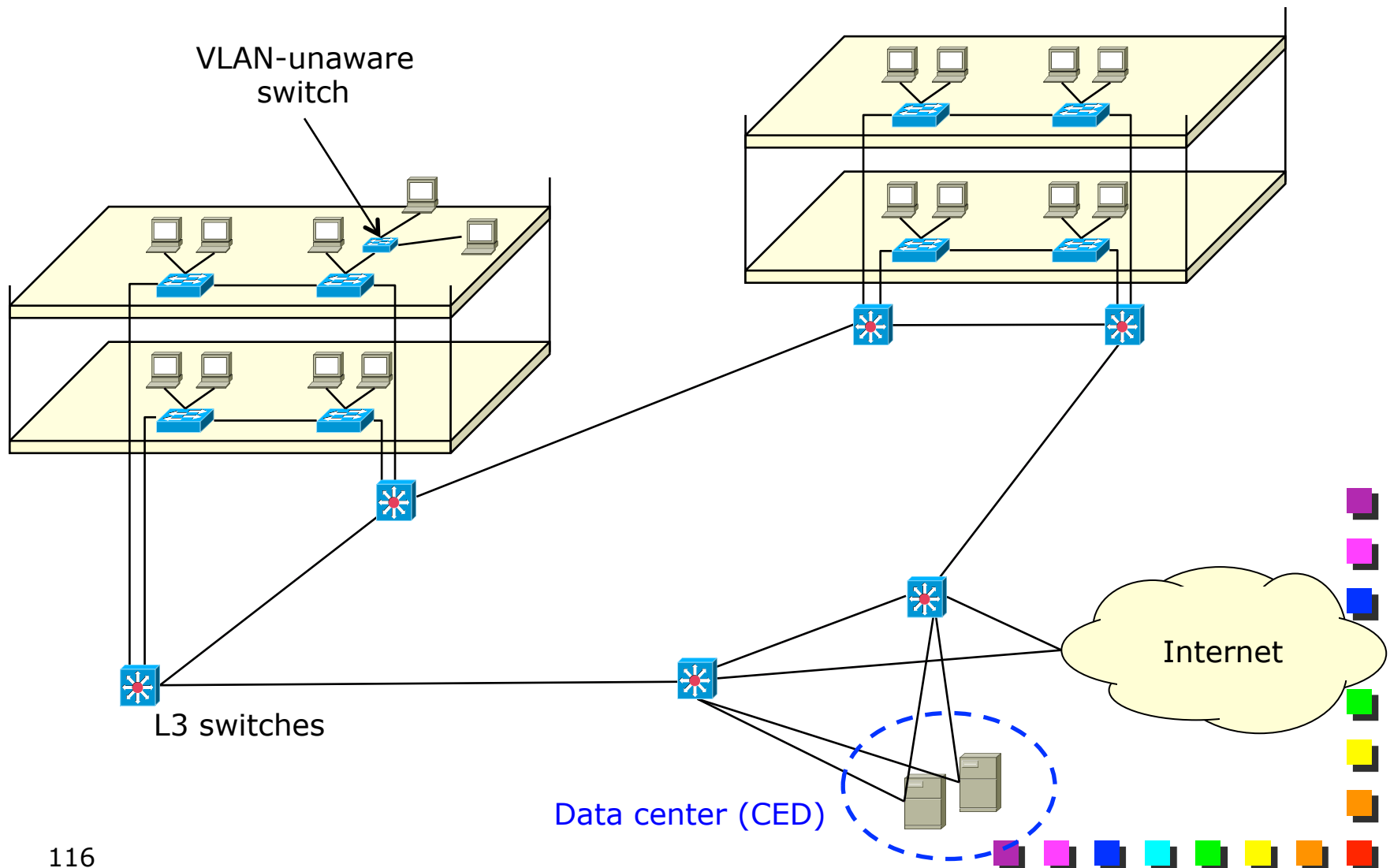
```
Switch# configure terminal
Switch(config)# interface FastEthernet 0/2
Switch(config-if)# switchport mode trunk
Switch(config-if)# switchport trunk allowed vlan
                        add 1,2 [or "all"]

Switch(config-if)# exit
Switch#
```

Ethernet LAN design (1)



Ethernet LAN design (2)





Conclusions

- Modern wired LANs are based on Switched Ethernet
 - Star topology with full-duplex links
 - Collision domain no longer exists
 - No need for CSMA/CD
 - Fault tolerance given by redundancy + Spanning Tree Protocol
 - Wide adoption of VLANs
 - Traffic isolation
 - Broadcast domain size reduction
 - Routers required for both internal and external communications
 - L3 closer to the users
 - L3 switches
- 