

# Lab4. VLAN Configuration

TA 許仲宇 (hsuchy)  
Credit to 陳吉遠 (ccy)

# Outline

1. 學理先備知識
2. 實驗目的
3. 實驗環境
4. 情境與指令介紹
5. 小作業 (不算分)

# Outline

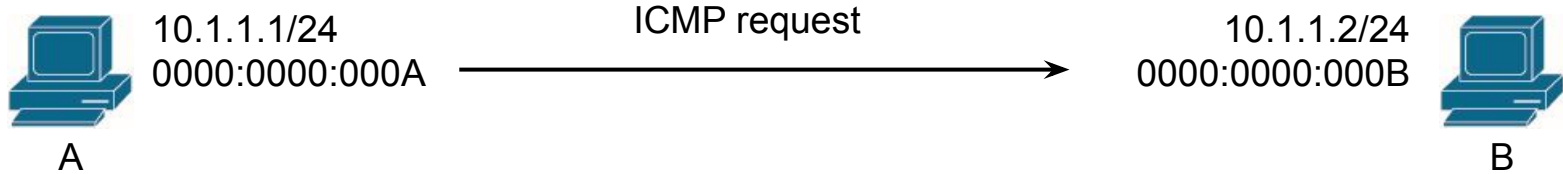
1. **學理先備知識**
2. 實驗目的
3. 實驗環境
4. 情境與指令介紹
5. 小作業 (不算分)

# 學理先備知識

- Local Area Network (LAN)
- Virtual LAN (VLAN)
- Trunk link
  - ISL (Cisco)
  - 802.1Q (IEEE)
    - Native VLAN
- Inter-VLAN Routing

# LAN

- The concept of LAN is for L2 or L3?
- Before discussion, let's deep into what happened when your host try to send a packet.



```
A# ping 10.1.1.2
```

# LAN - Network Layer

Application Layer

Transport Layer

Network Layer

Data Link Layer

Physical Layer



10.1.1.1/24  
0000:0000:000A

A



Src IP: 10.1.1.1  
Dst IP: 10.1.1.2

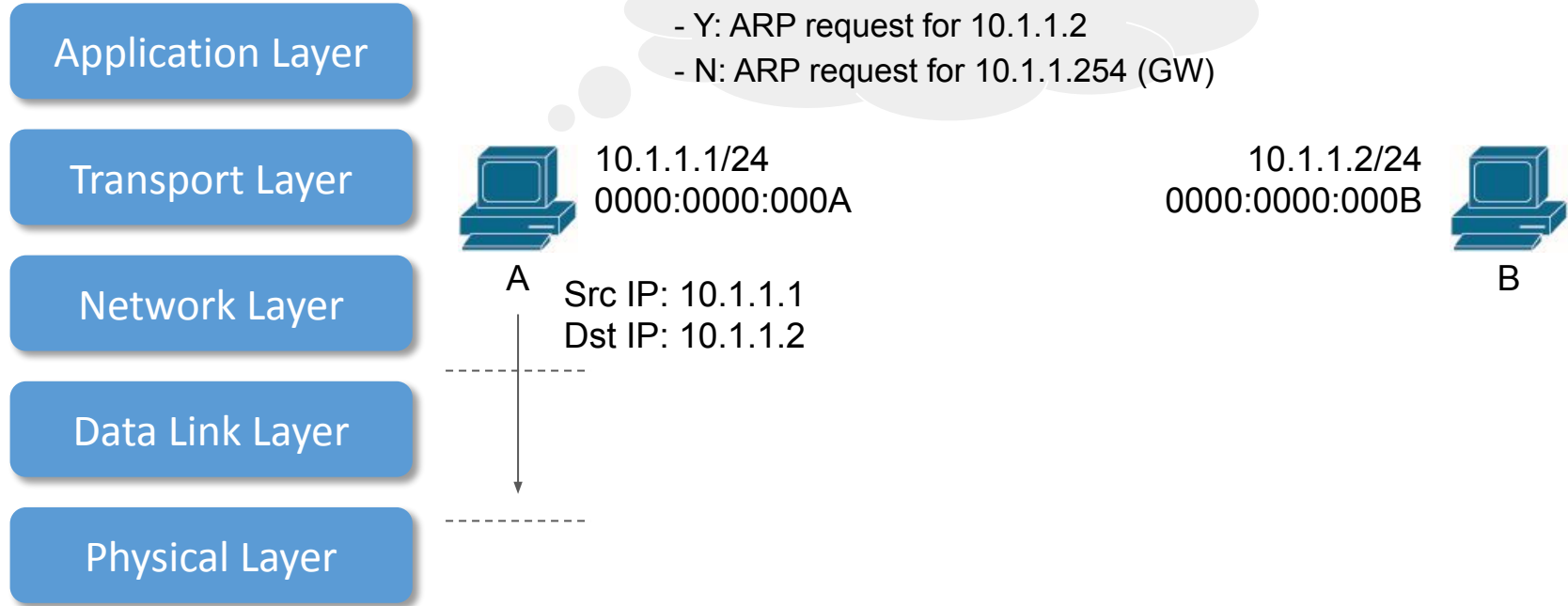


10.1.1.2/24  
0000:0000:000B

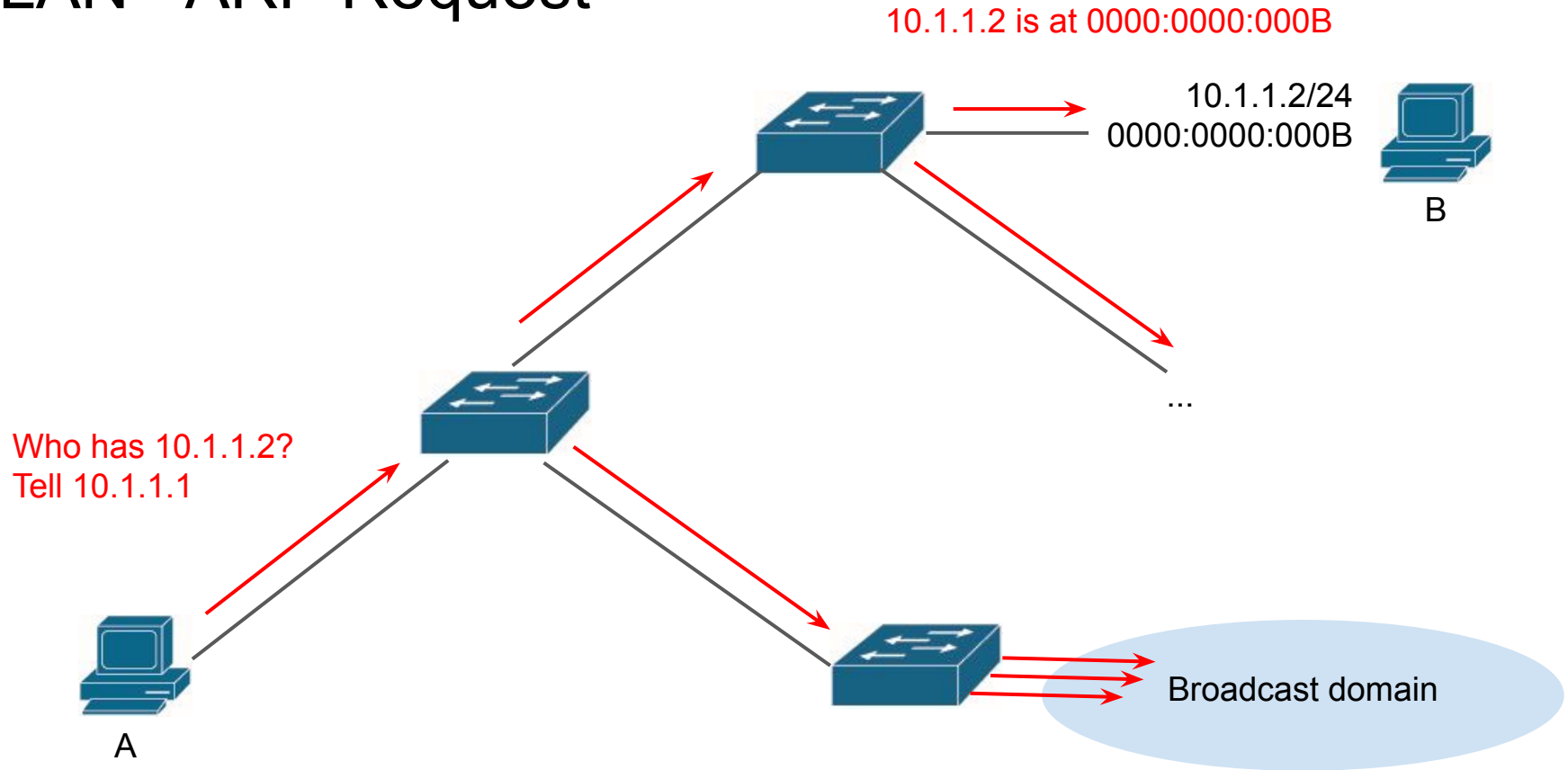


B

# LAN - Routing decision



# LAN - ARP Request

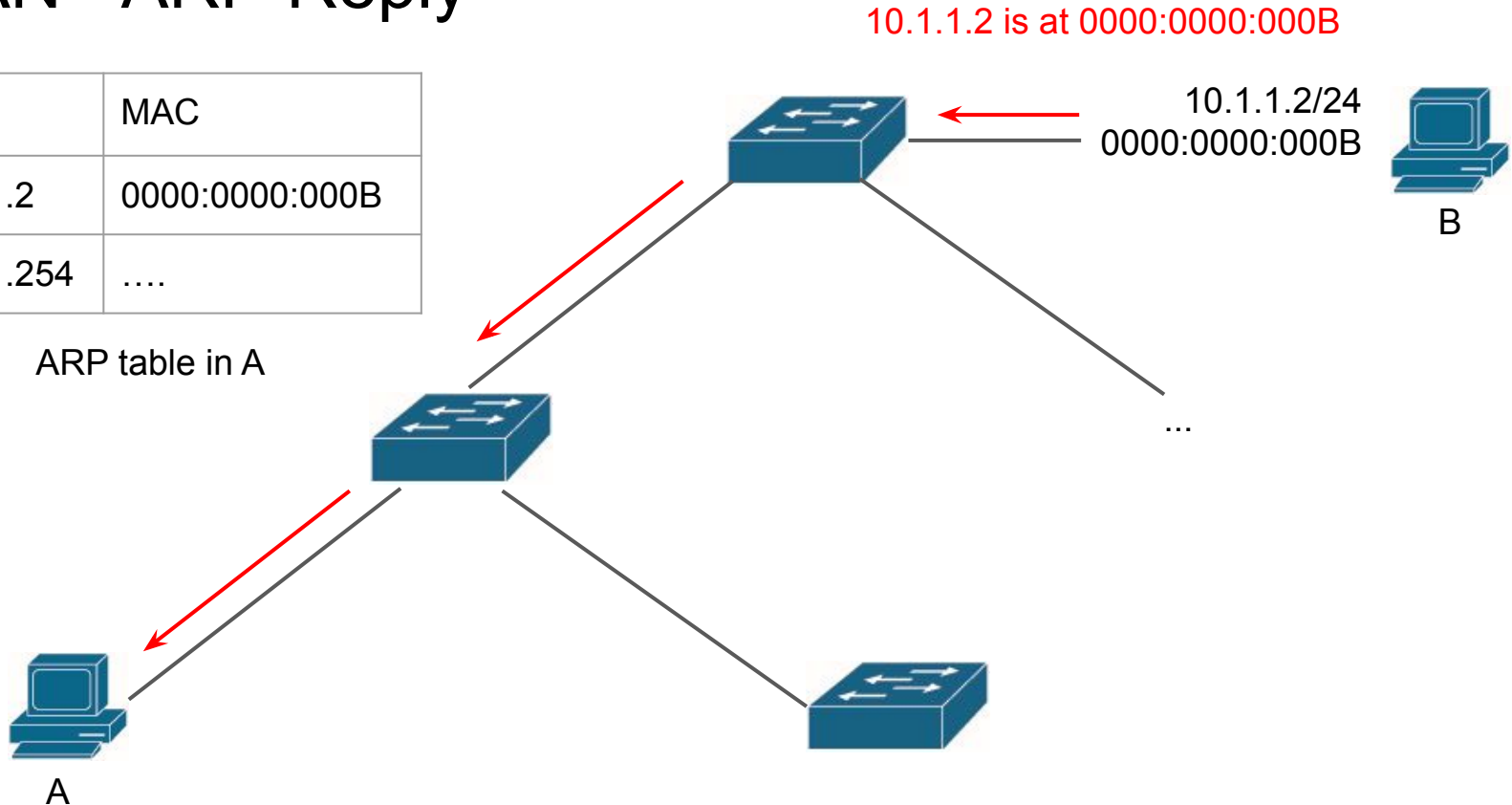




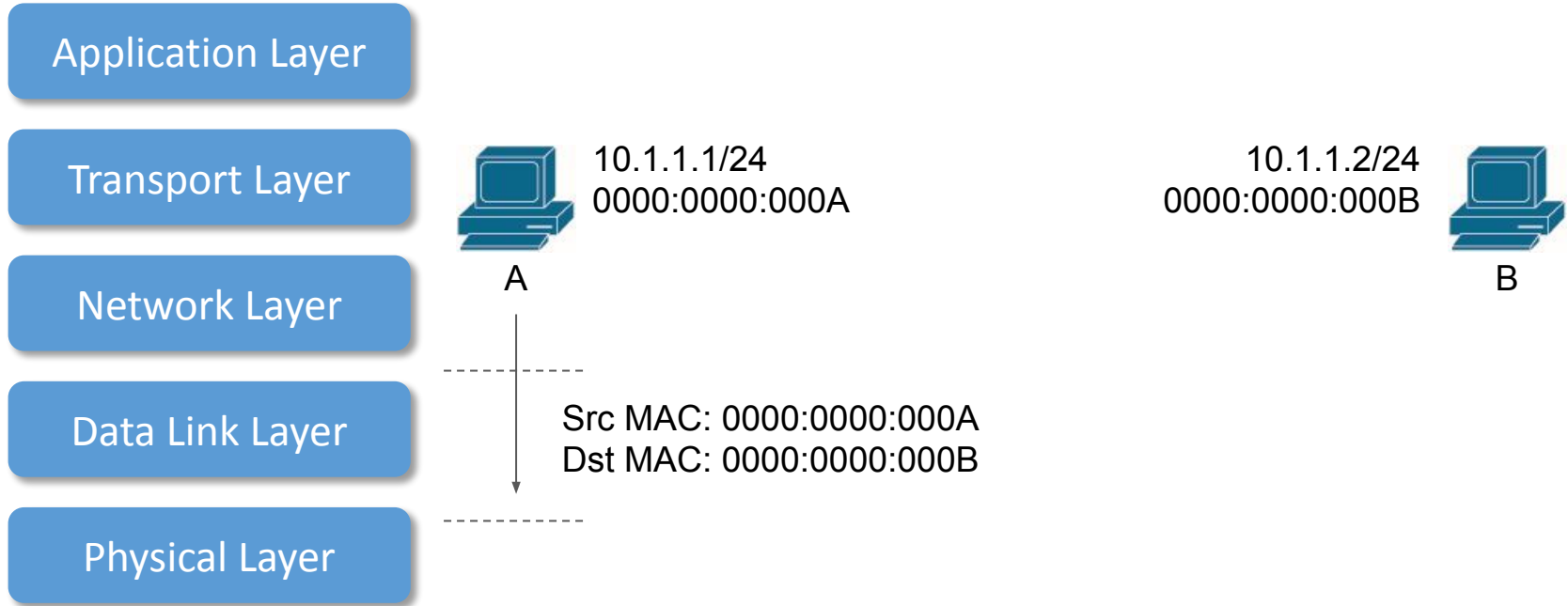
# LAN - ARP Reply

IP	MAC
10.1.1.2	0000:0000:000B
10.1.1.254	....

ARP table in A



# LAN - Data Link Layer

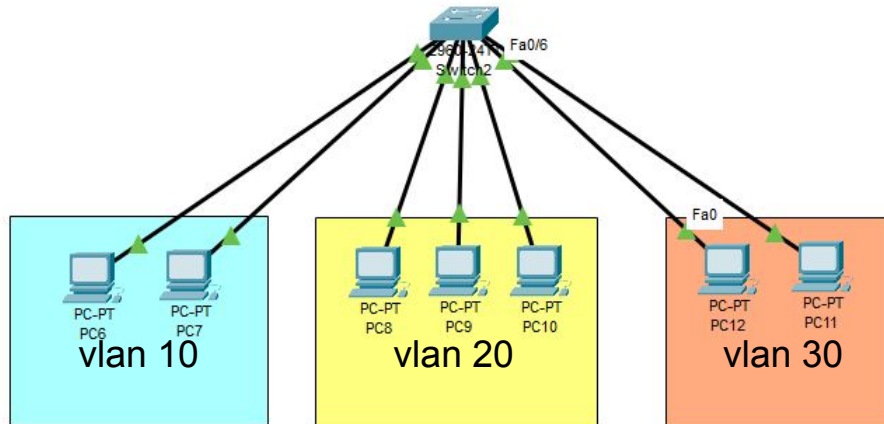


# LAN Conclusion

- So the concept of LAN is for L2 or L3?
- Both L2 and L3 have LAN concept
  - L2 LAN: Broadcast domain (Using MAC)
    - domain in which all nodes can reach each other by broadcast at the data link layer.
  - L3 LAN: Subnet (Using IP)
- But more general definition of LAN: **No need to be routed!**

# VLAN

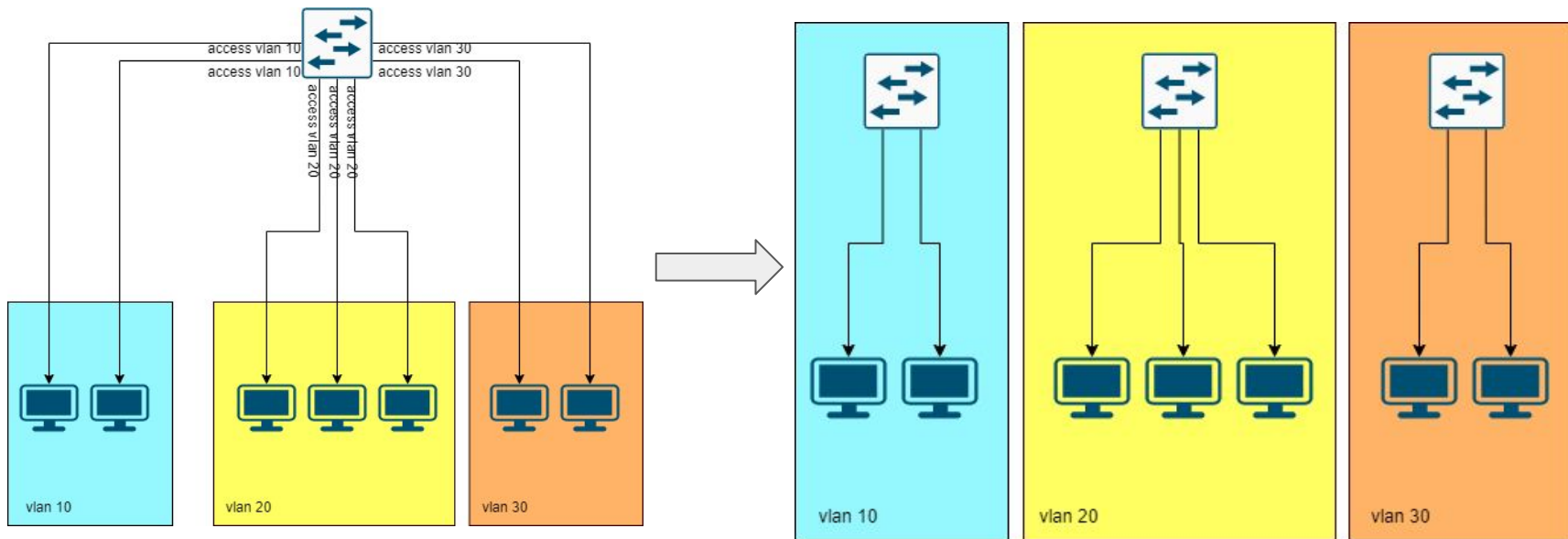
- Multiple LANs in the same physical switch
  - We want to separate the network traffic into three virtual LANs
  - Devices in vlan 20 would not received any flooded packets from the other VLAN
    - e.g. ARP request packet from other VLAN



# VLAN

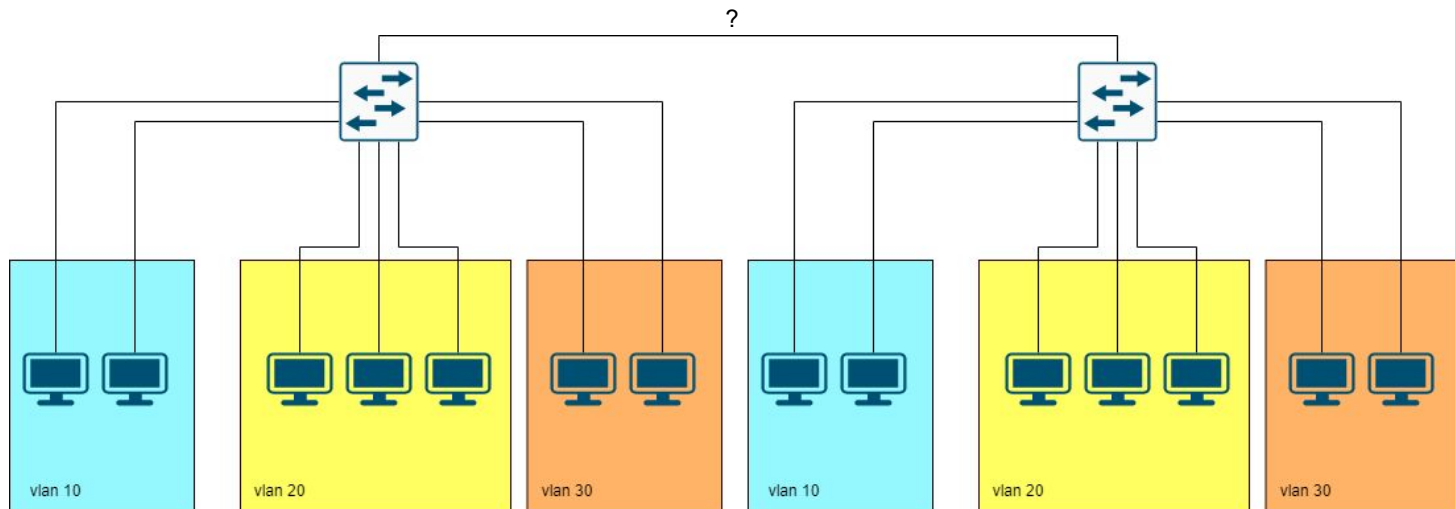
- Multiple LANs in the same physical switch

- When a switch receives packet from interfaces access to vlan 10, the switch only forwards the packet to those interfaces access to vlan 10 too.
- Hence we separate an L2 LAN into several VLANs.
  - every VLAN is a logically isolated broadcast domain



# VLAN

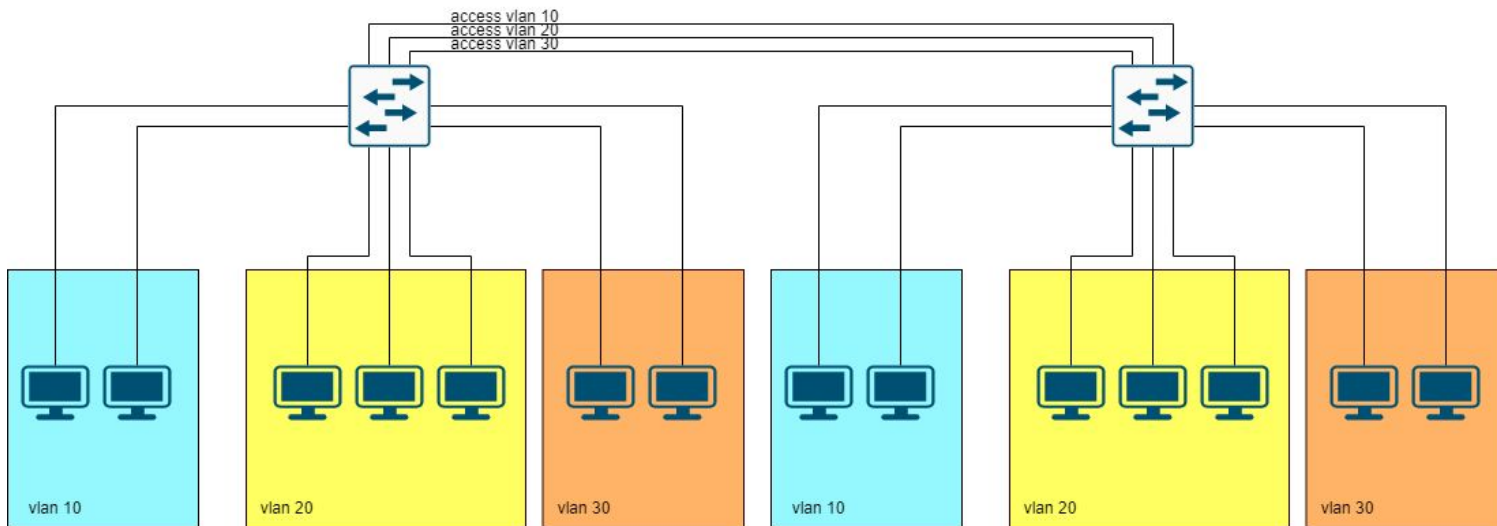
- VLANs across switches
  - vlan10 of the left switch should be connected to the vlan 10 of the right switch.
  - How should we connect the two switches?



# VLAN

- VLANs across switches

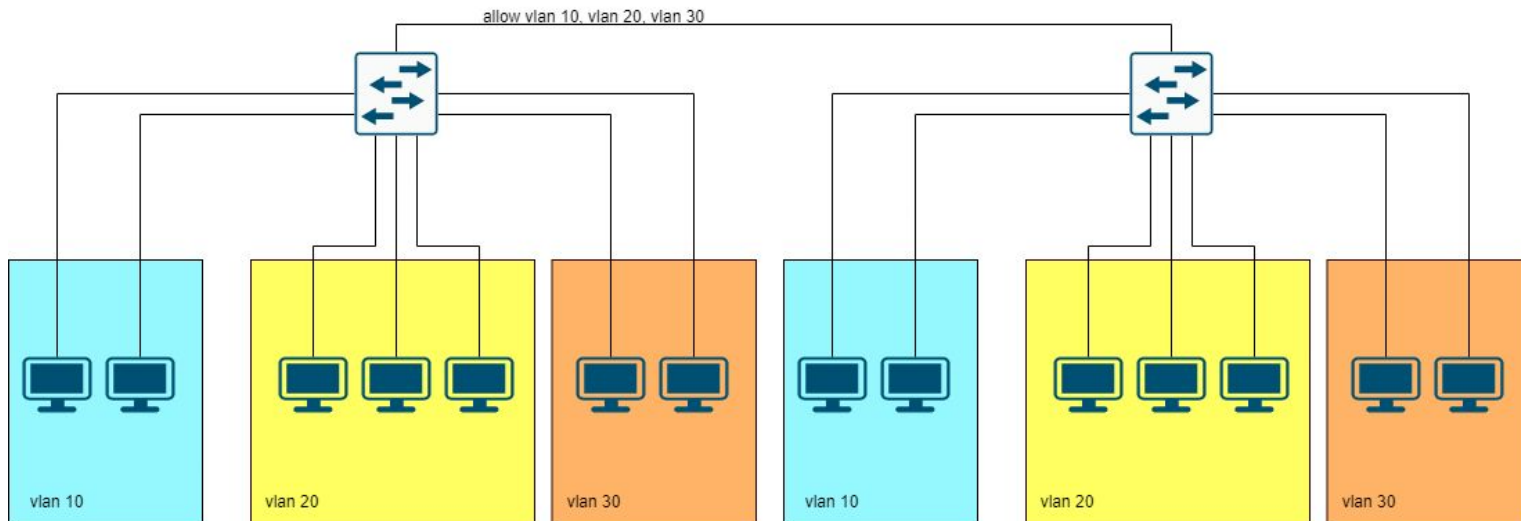
- vlan10 of the left switch should be connected to the vlan 10 of the right switch.
- One possible solution is like this:
  - It wastes not only physical interfaces but a lot of network wiring cost!



# VLAN

- VLANs across switches

- vlan10 of the left switch should be connected to the vlan 10 of the right switch.
- A better solution: connect two switches with one link allowed multiple VLANs.
  - That is what trunk does!

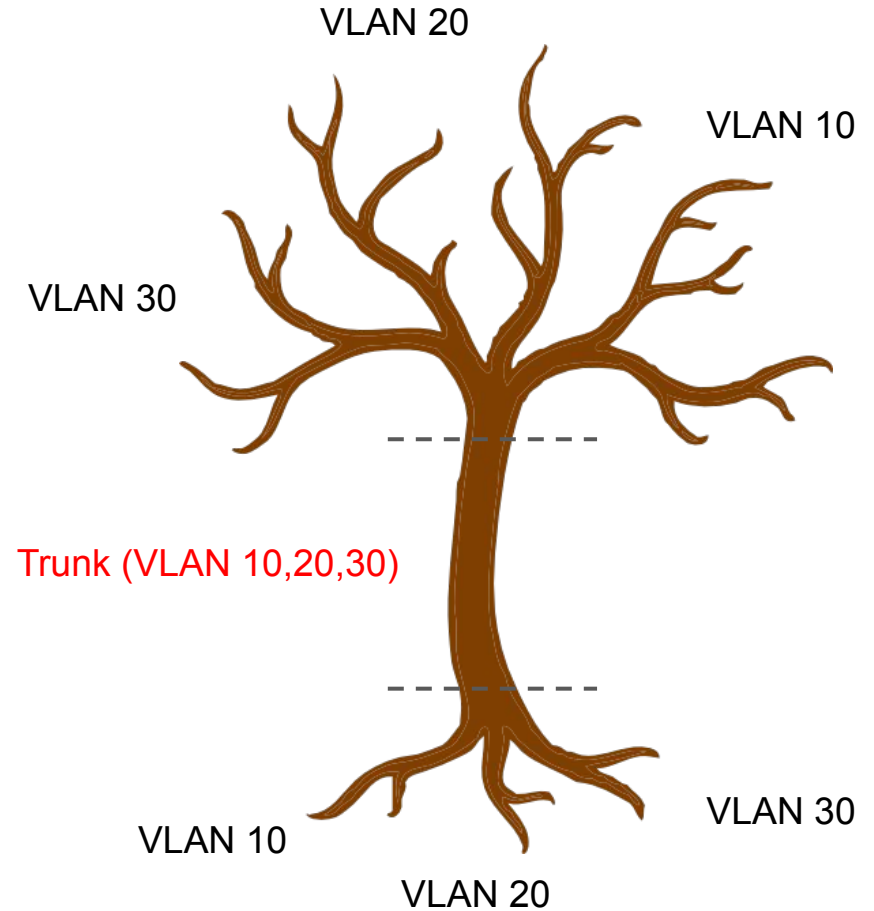
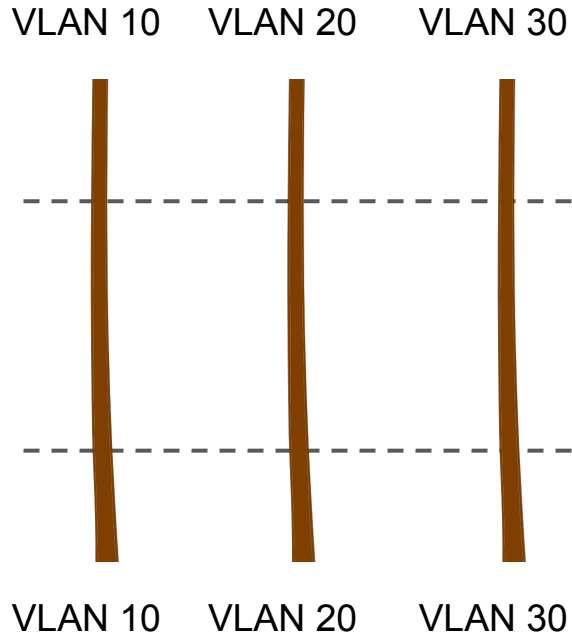




# VLAN

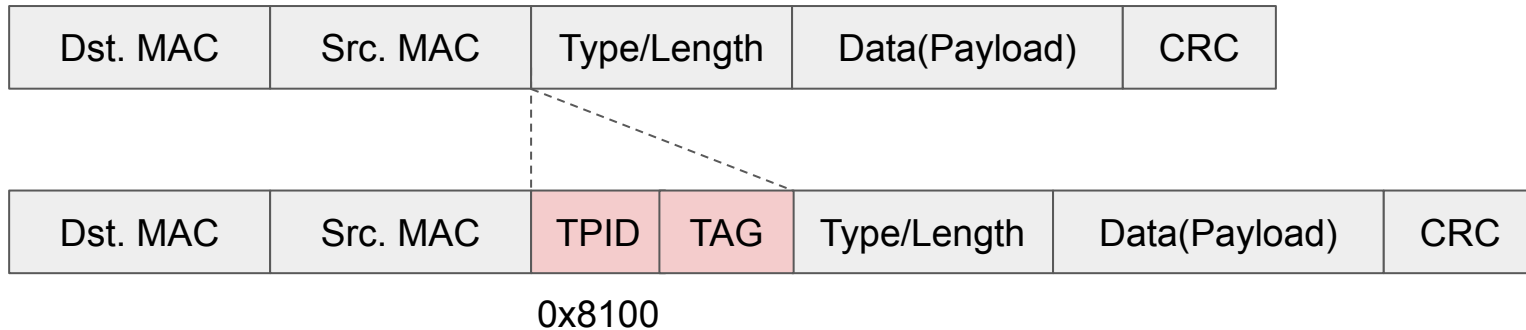
- Each interface has its VLAN mode, to be “access mode” or “trunk mode”
  - access mode
    - the interface is in a specific vlan, switch would forwards network traffic from those the same vlan to this interface
  - trunk mode
    - the interface would allows some vlans, and switch would forwards network traffic from those allowed vlans

# Trunk



# VLAN Tagging

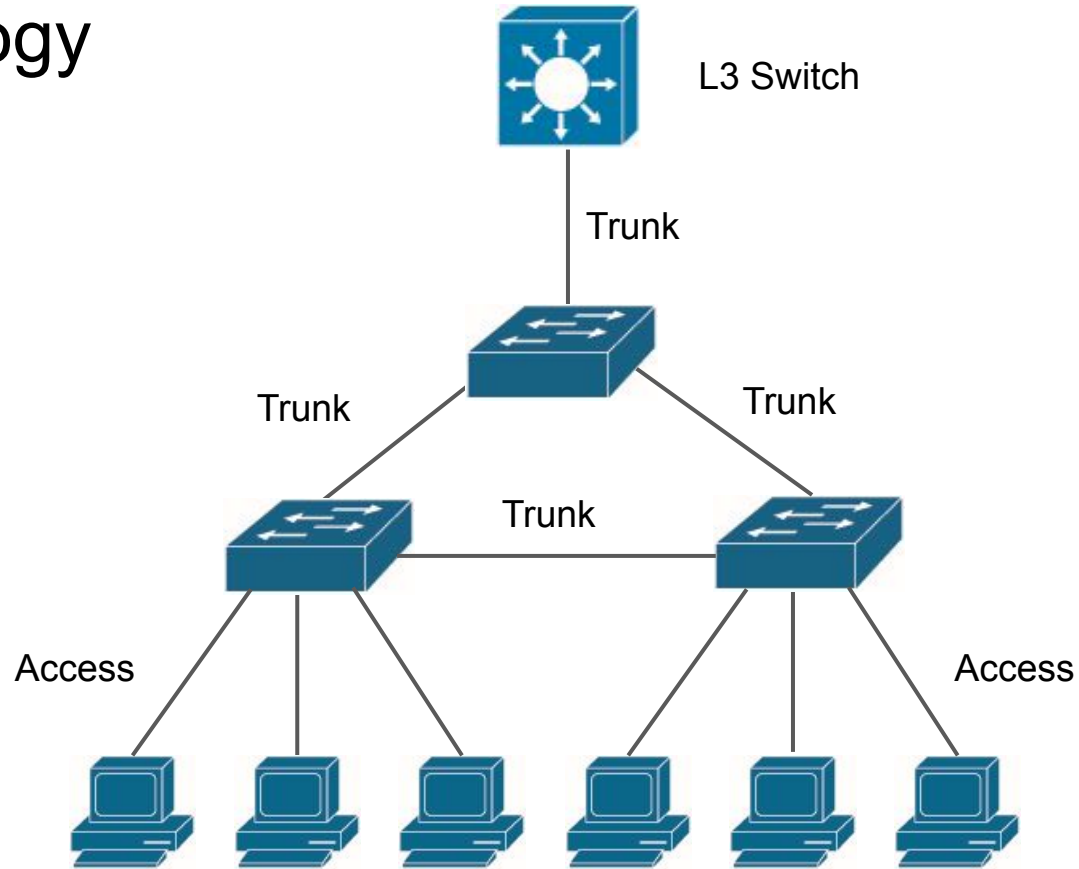
- IEEE 802.3 (Ethernet) with vlan information
  - Cisco ISL (seldom used)
  - IEEE 802.1Q



# VLAN Tagging

- Review: Tagging of access/trunk interface
- Access:
  - expect receive untagged frame
    - drop tagged frame when received
  - remove frame's tag before sending out
- Trunk
  - expect receive tagged frame.
    - if frame is untagged, consider it is inside native VLAN
  - add frame's tag before sending out

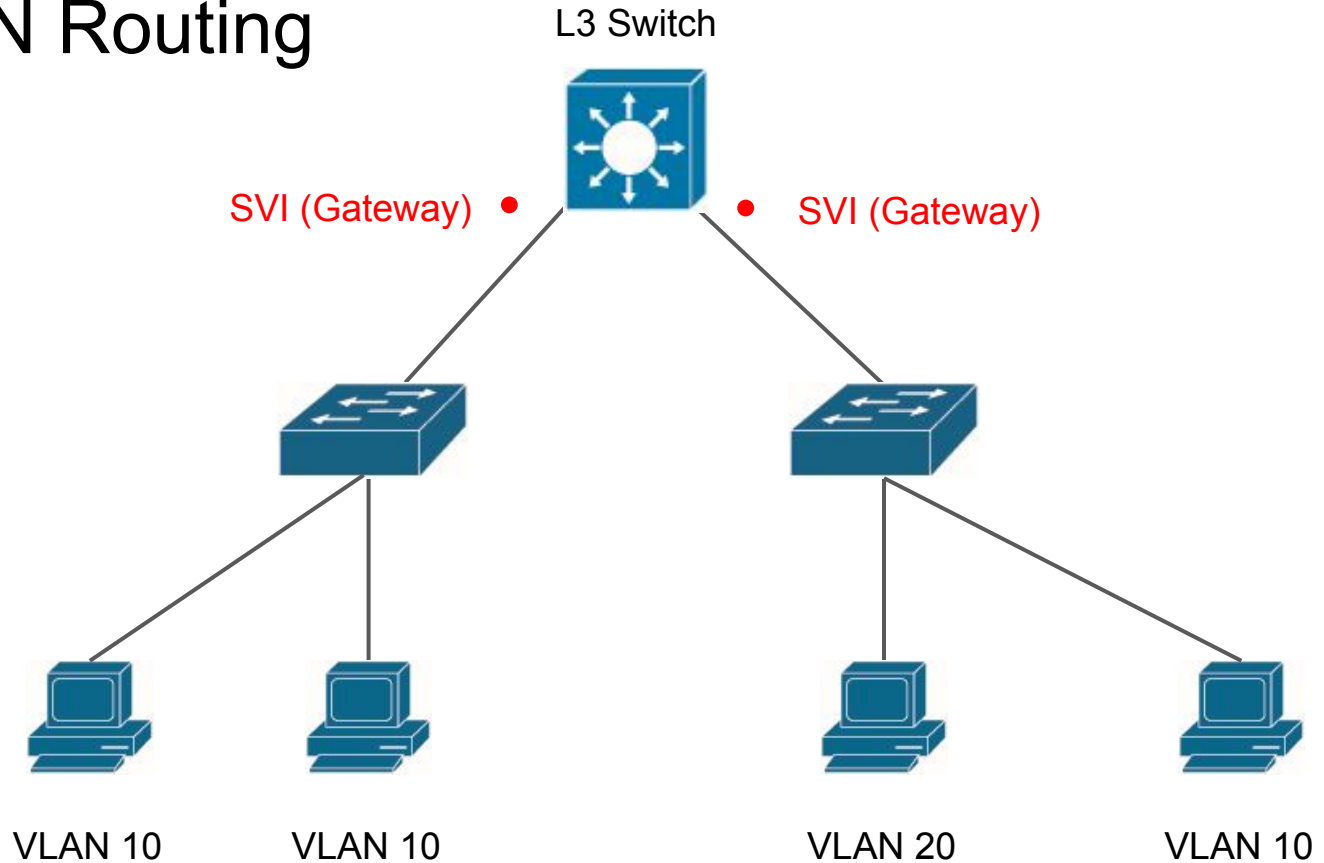
# Trunk Topology



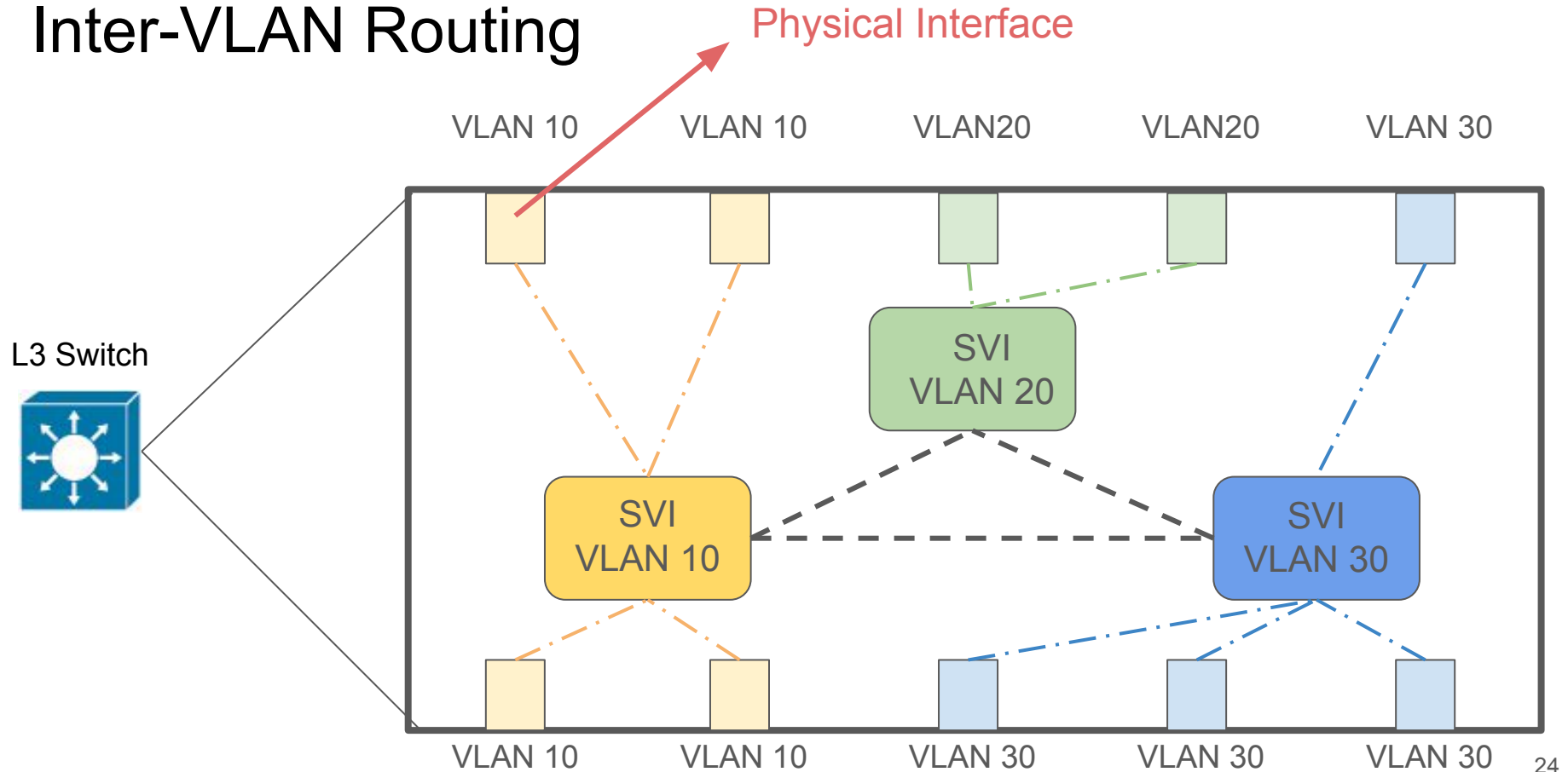
# Switched Virtual Interface (SVI)

- Switch does not have a physical interface to which an IP address can be assigned.
  - IP is configured on a virtual interface called **switched virtual interface (SVI)**.
- Switched Virtual Interface (SVI)
  - A single SVI can only be mapped to a VLAN.
  - A SVI cannot be activated unless that VLAN associated with **at least one** active physical port.
  - SVI provides the Layer 3 processing for packets from all active physical ports associated with the VLAN.
    - routing packet from/to other SVI
    - no need physical router for inter-VLAN routing

# Inter-VLAN Routing



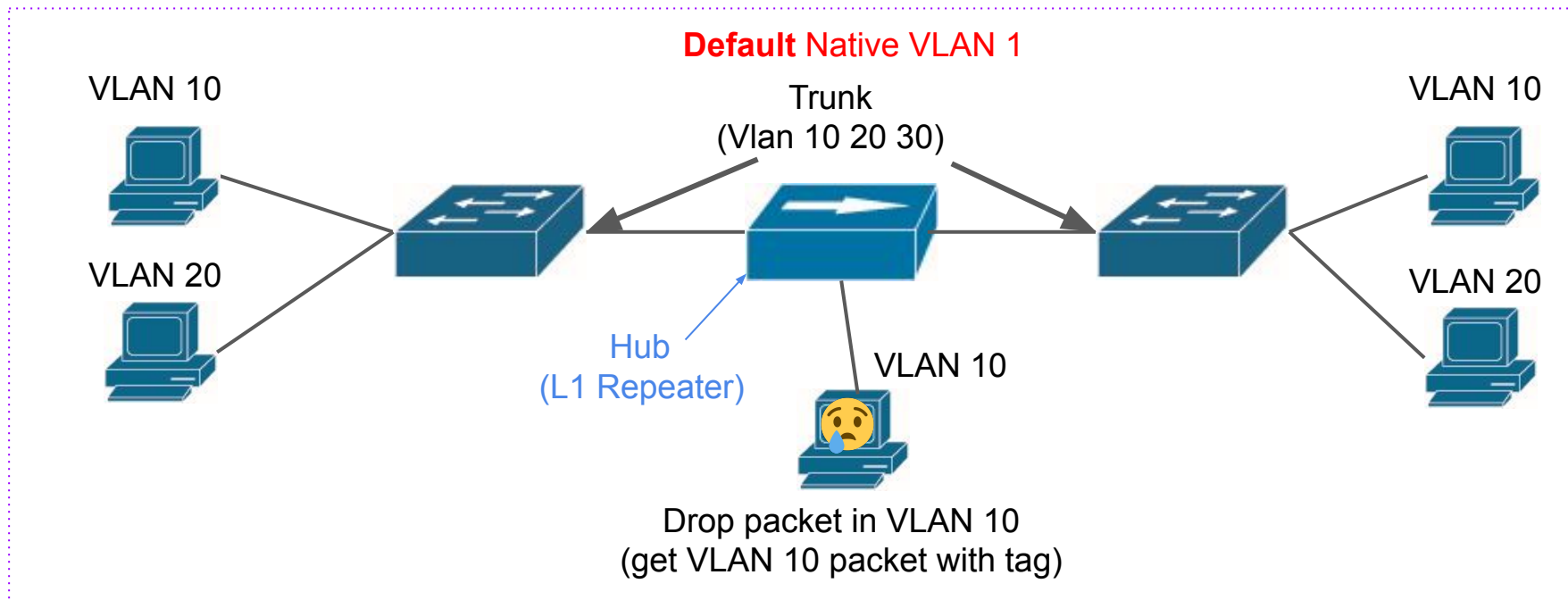
# Inter-VLAN Routing





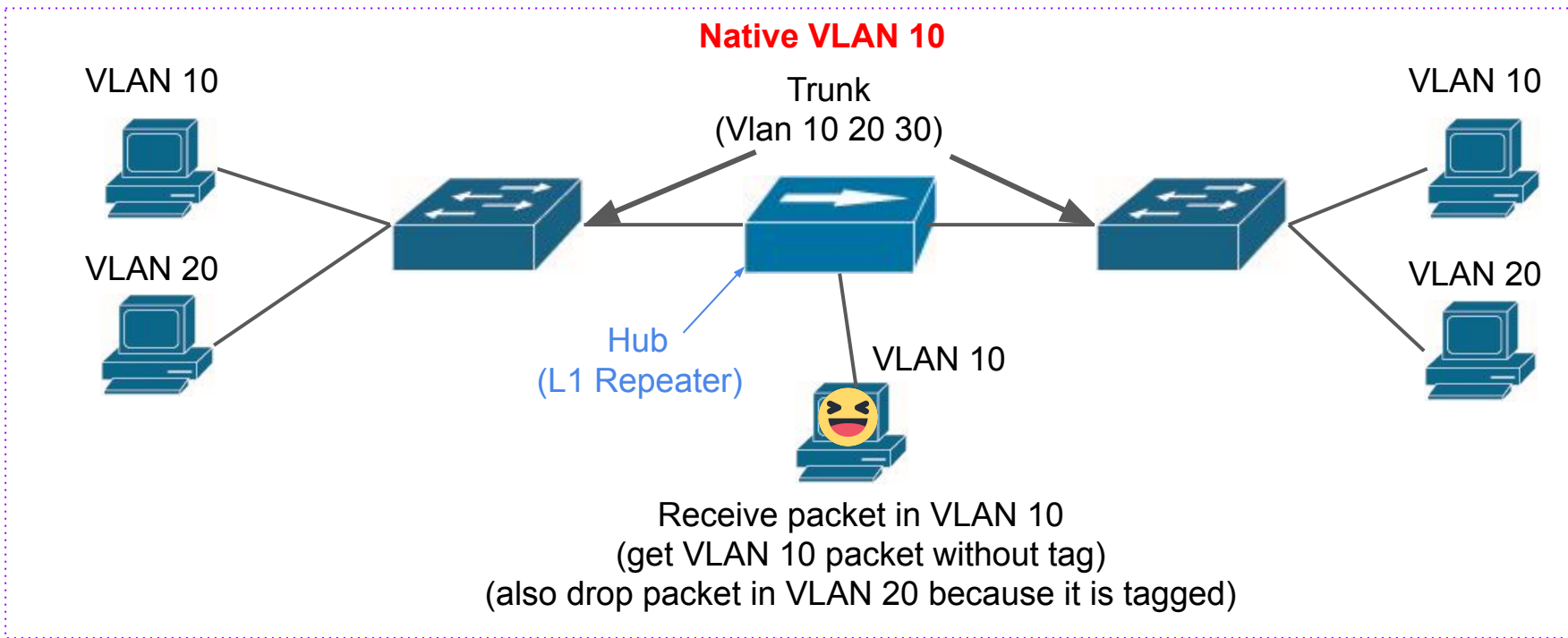
# Trunk - Native VLAN

- If your topology has device that doesn't recognize VLAN tag
  - e.g. Hub. It won't untag packet's VLAN tag. Instead, it will simply flooding the packet received.



# Trunk - Native VLAN

- Use native VLAN to solve this kind of problem
  - when a trunk interface transmit packet with its native VLAN, it will send the packet **untagged**



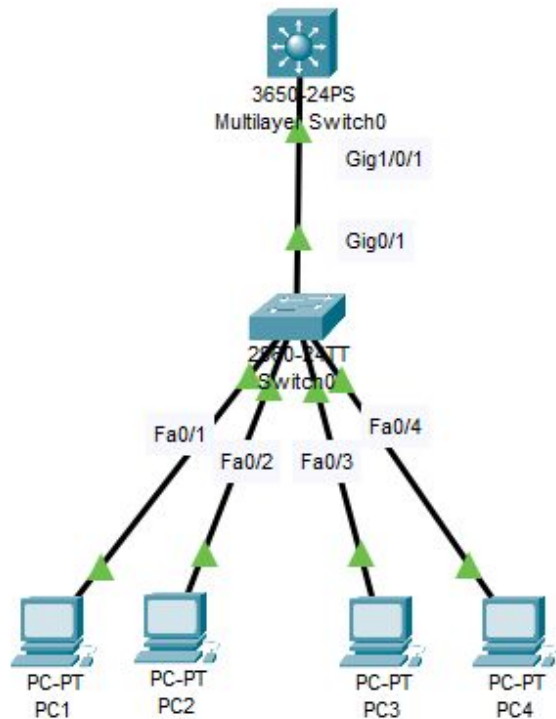
# Outline

1. 學理先備知識
2. **實驗目的**
3. **實驗環境**
4. 情境與指令介紹
5. 小作業 (不算分)

# 實驗目的

- 設定 VLAN, 檢視不同區域網路的封包能否互通
- 比較 Access link 與 Trunk link, Trunk link 有什麼好處
- 設定 Native VLAN, 學習 Native VLAN 的應用環境
- 實際操作 Inter-VLAN Routing, 了解 L2、L3 差異

# 實驗環境



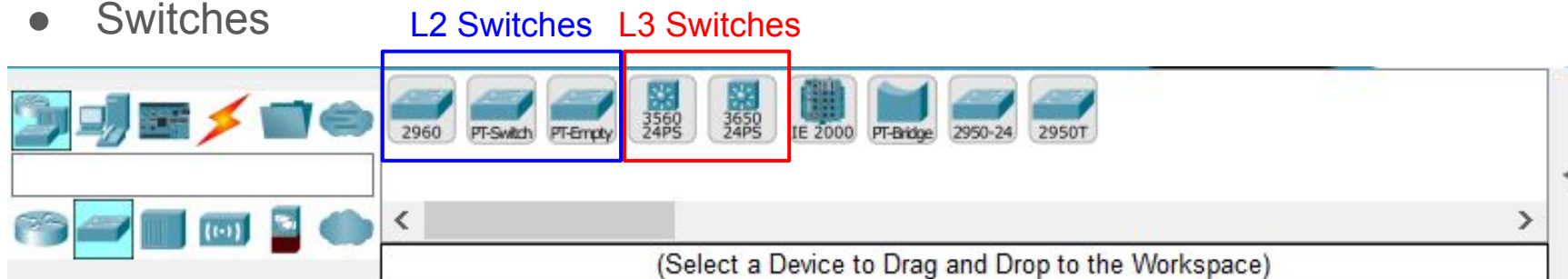
Device	IP
PC1	10.1.10.1/24
PC2	10.1.10.2/24
PC3	10.1.20.1/24
PC4	10.1.20.2/24

# 建立實驗環境 (1/4)

- Routers

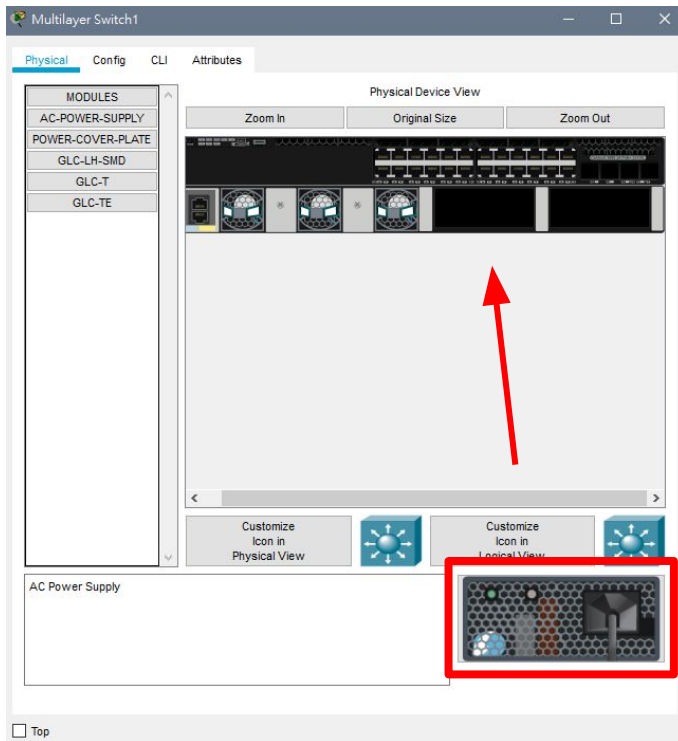


- Switches



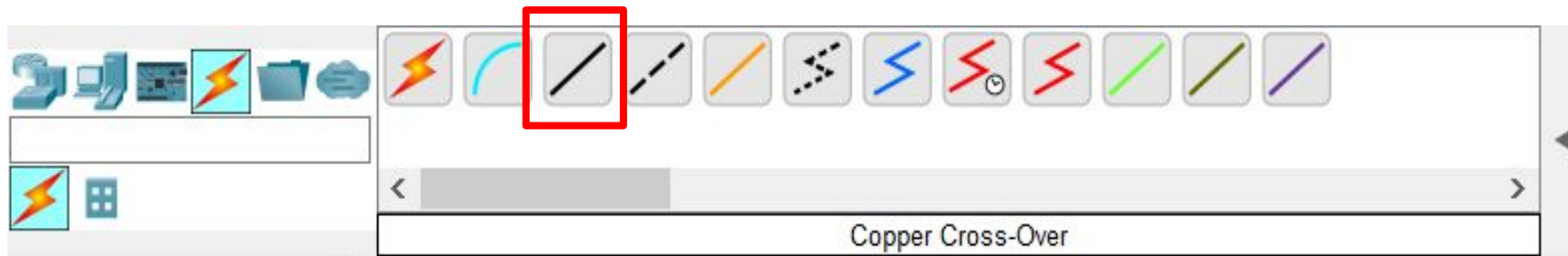
## 建立實驗環境 (2/4)

- 新增一個機器時，可能需要手動拖曳電源模組



## 建立實驗環境 (3/4)

- Connections

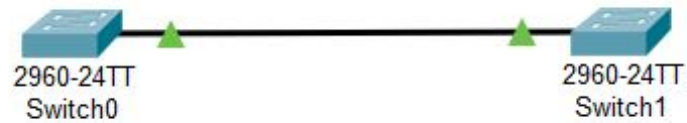
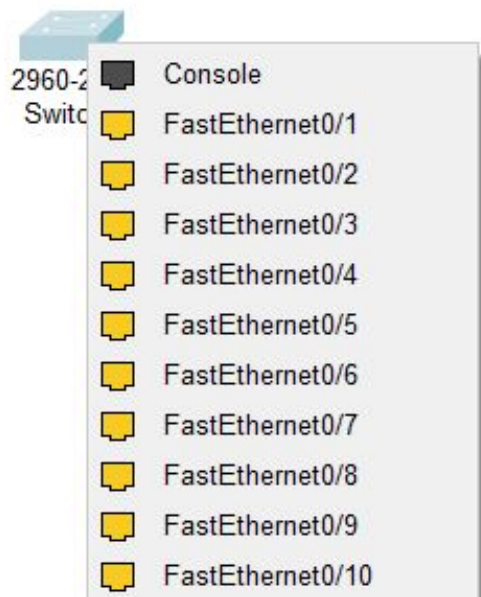


- End Devices





## 建立實驗環境 (4/4)

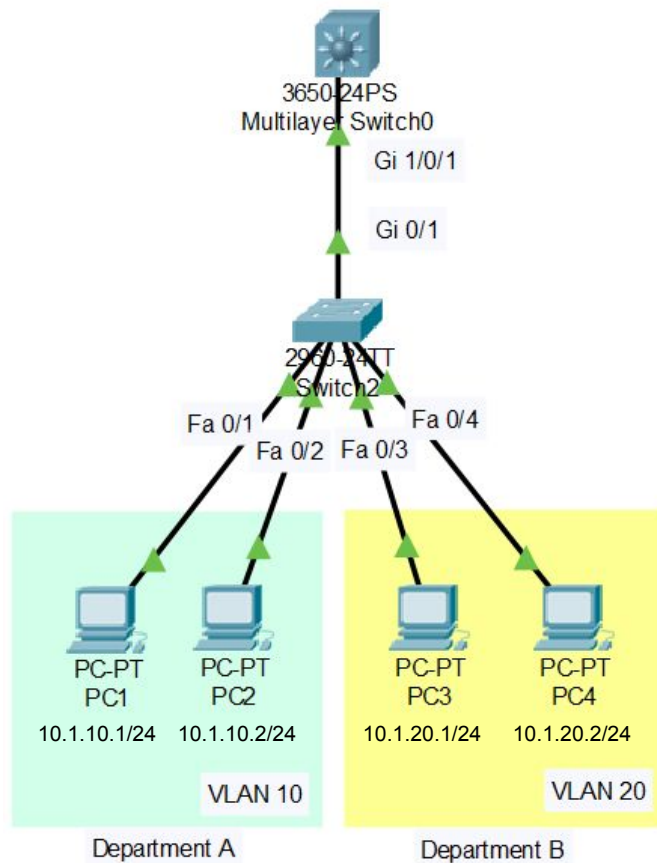


# Outline

1. 學理先備知識
2. 實驗目的
3. 實驗環境
- 4. 情境與指令介紹**
5. 小作業 (不算分)

# 情境介紹

- 公司內有兩個部門 A, B
- 部門的主管告訴你, 他希望不同部門的機器可以位在不同區域網路內
- 身為網管的你想到曾經在上課學到 VLAN 相關的知識, 於是開始著手設定



# 實驗所需指令 - VLAN

- 建立 VLAN

```
switch# configure terminal
switch(config)# vlan vlan-num —————> 可用的 vlan-num 參考後方投影片
switch(config-vlan)# name vlan-name (optional)
```

- 將 Port 指派給 VLAN

```
(config)# interface type interface_number
(config-if)# switchport mode access —————> access link
(config-if)# switchport access vlan vlan-num
```

# 實驗所需指令 - VLAN

- Cisco IOS 可用的 VLAN 範圍

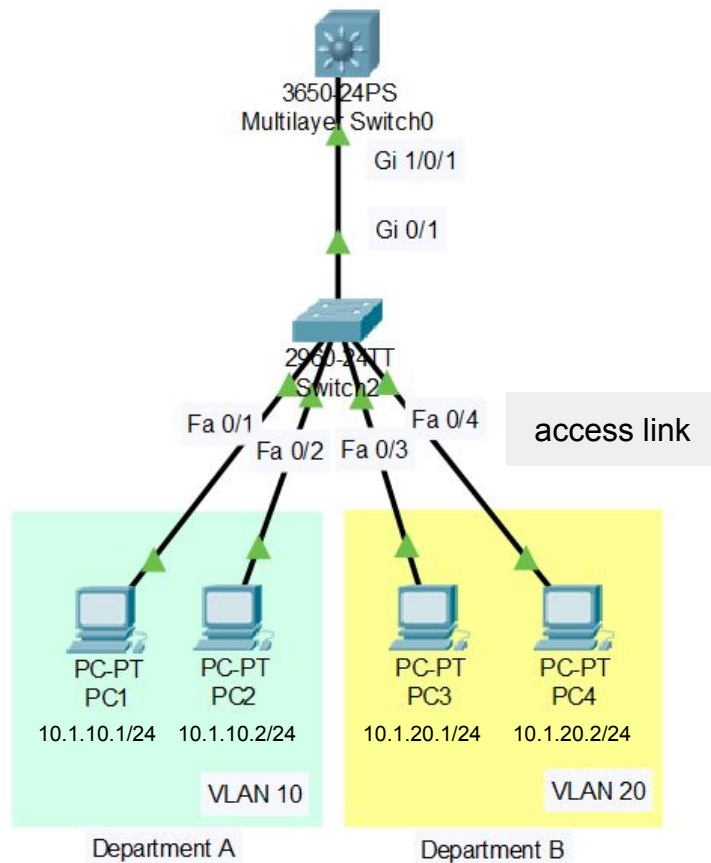
VLANs	Range	Usage
0, 4095	Reserved	For system use only. You cannot see or use these VLANs.
1	Normal	Cisco default. You can use this VLAN but you cannot delete it. Cisco will use this VLAN to send Control Plane Traffic (like CDP, BPDU)
2-1001	Normal	For Ethernet VLANs; you can create, use, and delete these VLANs.
1002-1005	Normal	You cannot delete VLANs 1002-1005. (Cisco defaults for FDDI and Token Ring)
1006-4094	Extended	For Ethernet VLANs only.

## 實驗所需指令 - 情境示範

```
switch(config)# vlan 10
switch(config-vlan)# name department-A
...
switch(config)# vlan 20
switch(config-vlan)# name department-B
...
switch(config)# interface fastEthernet 0/1
switch(config-if)# switchport mode access
switch(config-if)# switchport access vlan 10
...
switch(config)# interface fastEthernet 0/4
switch(config-if)# switchport mode access
switch(config-if)# switchport access vlan 20
...
```


# 實驗所需指令 - 情境示範

- 現在兩個部門的 L2 流量切開了
  - 部門 A : VLAN 10
  - 部門 B : VLAN 20
- endpoint 透過 access link 連到 L2 Switch
- 但為了讓不同 VLAN 能互通, 我們需要 L3 Switch 來幫我們做路由
- 怎麼讓 L2 Switch 與 L3 Switch 透過一條實體線路就能讓不同 VLAN 的流量能過通過?
  - 透過 Trunk



# 實驗所需指令 - Trunk

- 設定 Trunk link

```
(config)# interface type interface_number
(config-if)# switchport trunk encapsulation {isl|dot1q|negotiate}
! 預設為 negotiate, 會視另一端協商使用 ISL 或是 802.1Q
! 有些較新的 Switch 不支援 ISL 會沒有這個指令, 或是只有 dot1q 的選項
(config-if)# switchport trunk allowed vlan {vlan-list|except vlan-list|all}
(config-if)# switchport mode trunk  trunk link
```

- 修改已經設定完的 Trunk link

```
(config-if)# switchport trunk allowed vlan {add|remove} vlan-list
```

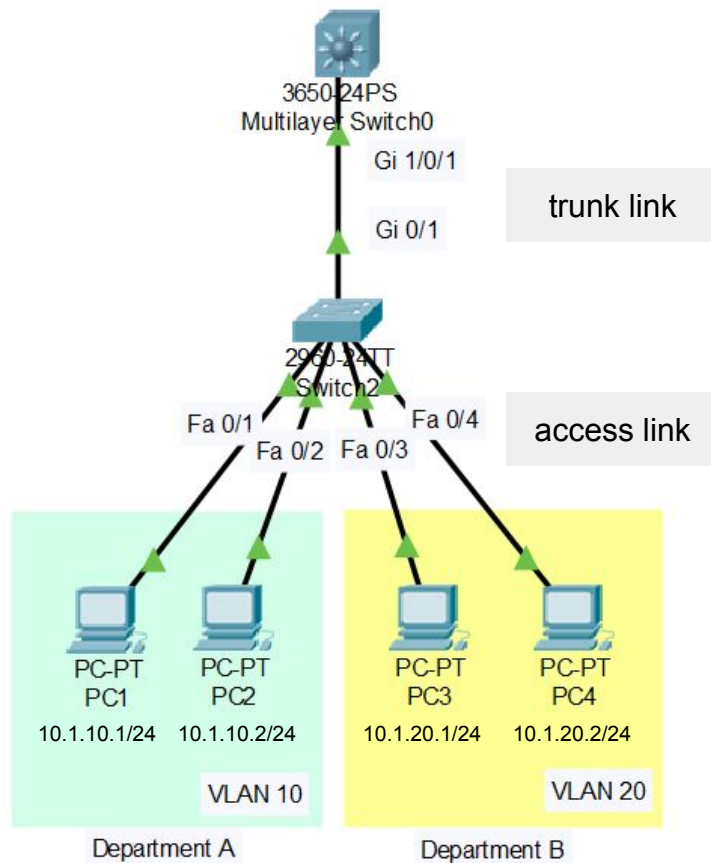


## 實驗所需指令 - 情境示範

```
switch(config)# interface gigabitEthernet 0/1
switch(config-if)# switchport trunk allowed vlan 10,20
switch(config-if)# switchport mode trunk
! switch 2960 上沒有 encapsulation 的指令
...
L3switch(config)# interface gigabitEthernet 1/0/1
L3switch(config-if)# switchport trunk encapsulation dot1q
L3switch(config-if)# switchport trunk allowed vlan 10,20
L3switch(config-if)# switchport mode trunk
! L3 switch 3650 上有 encapsulation 的指令, 但只有 dot1q 的選項
...
```

# 實驗所需指令 - 情境示範

- L3 Switch 能夠拿到 PC 傳上來的流量了
- 最後, 要在 L3 Switch 上的 **SVI** 設定 VLAN 的 Gateway IP, 並開啟 Routing
- 將 PC 的 Gateway 設定好, 讓不同 VLAN 能透過 Routing 互通



# 實驗所需指令 - Inter-VLAN Routing

- 在 Router/L3 Switch 建立 VLAN

```
(config)# vlan vlan-num  
(config-if)# name vlan-name (optional)
```

- 設定 VLAN Interface (= SVI), 並設定 VLAN Gateway 的 IP

```
(config)# interface vlan vlan-num  
(config-if)# ip address ip netmask
```

- 在 L3 Switch 上啟用 Routing

```
(config)# ip routing
```

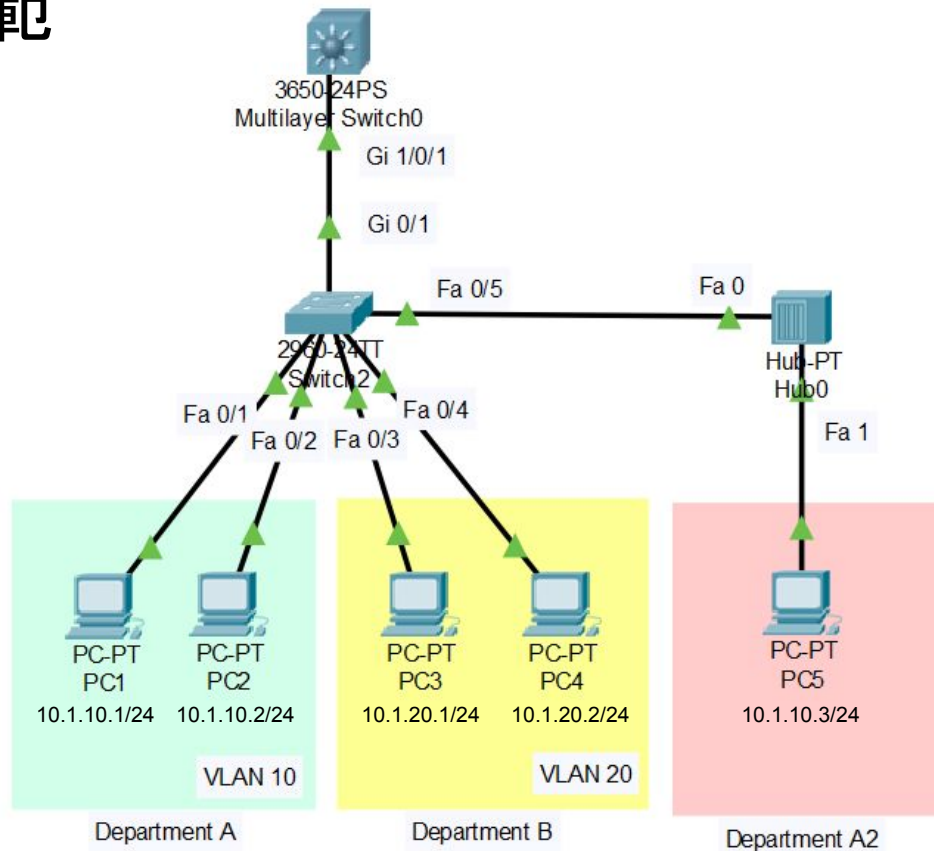
## 實驗所需指令 - 情境示範

```
L3switch(config)# vlan 10
...
L3switch(config)# vlan 20
...
L3switch(config)# interface vlan 10
L3switch(config-if)# ip address 10.1.10.254 255.255.255.0
...
L3switch(config)# interface vlan 20
L3switch(config-if)# ip address 10.1.20.254 255.255.255.0
...
L3switch(config)# ip routing
```

- 最後再將 PC 上的 Gateway 設為那個 VLAN 下對應的 Gateway

# 實驗所需指令 - 情境示範

- 終於設定完了, VLAN 10 和 20 現在能夠上到 L3 Switch 做路由後互通了
- 準備下班的你, 看到老闆拿著一台 Hub (不支援VLAN), 問你能不能為部門 A 的子部門 A2 設定, 讓他們也能夠使用公司的網路



# 實驗所需指令 - Native VLAN

- 設定 Native VLAN, Cisco Native VLAN 預設為 VLAN 1

```
(config)# interface type interface_number  
(config-if)# switchport trunk encapsulation dot1q  
(config-if)# switchport trunk native vlan vlan-num
```

## 實驗所需指令 - 情境示範

```
switch(config)# interface fastEthernet 0/5  
switch(config-if)# switchport trunk native vlan 10  
switch(config-if)# switchport mode trunk
```

- 就算 Hub 不支援 VLAN, 現在部門 A2 與部門 A 可以直接透過 L2 溝通了

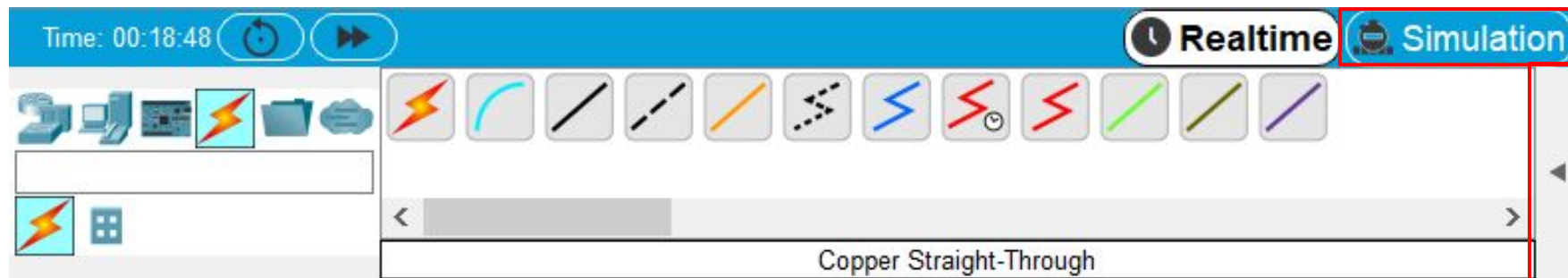
# 實驗所需指令(補充) - Dynamic trunking

```
(config) # ... 前略  
(config-if) # switchport mode dynamic {desirable|auto}
```

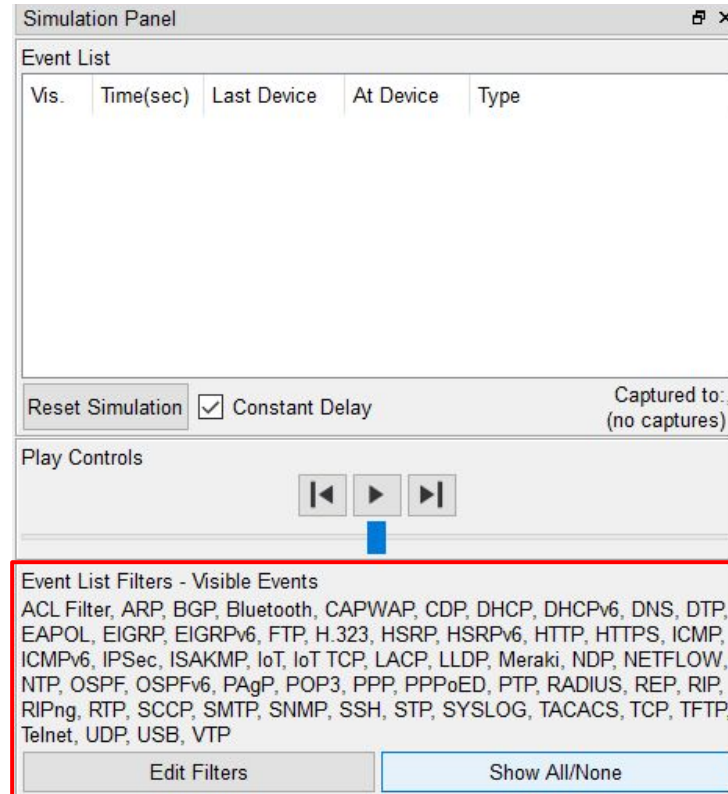
- 除了前面提到的 trunk mode 與 access mode, 還有 dynamic mode 可以動態協商是要使用 access 或是 trunk
  - dynamic desirable, port 會**主動**嘗試將鏈路轉為 trunking mode
  - 預設為 dynamic auto, **被動**協商, 僅在另一端交換器主動要求時才會轉為 trunking mode
- 有興趣可以自己查關鍵字: "cisco dtp"、"dynamic trunking protocol"



# Simulation (1/5)



# Simulation (2/5)



Event Filter

# Simulation (3/5)

File Edit Options View Tools Extensions Help

Logical Physical x: 812, y: 0 Add Simple PDU (P)

File Edit Options View Tools Extensions Help

Logical Physical x: 915, y: 0 Add Complex PDU (C)

2960-24TT Switch0 2960-24TT Switch1

Create Complex PDU

Source Settings

Source Device: Switch0

Outgoing Port:  ☒ Auto Select Port

PDU Settings

Select Application: PING

Destination IP Address:

TTL:

TOS:

Sequence Number:

Size:

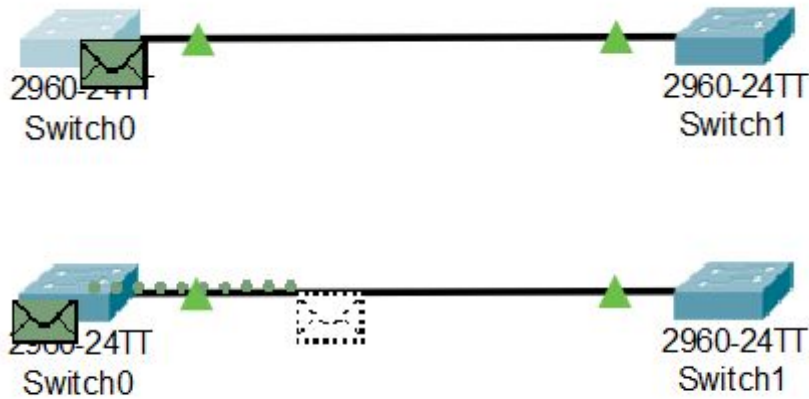
Simulation Settings

☒ One Shot Time:  Seconds

☐ Periodic Interval:  Seconds

Create PDU

# Simulation (4/5)



Scenario 0

New Delete

Toggle PDU List Window

Fire	Last Status	Source	Destination	Type	Color	Time(sec)	Periodic	Num	Edit	Delete
	Successful	Switch0	Switch1	ICMP		0.000	N	0	(edit)	(delete)

Simulation Panel

Event List

Vis.	Time(sec)	Last Device	At Device	Type
	0.000	--	Switch0	ICMP

Reset Simulation ☒ Constant Delay Captured to: 0.000 s

Play Controls

Event List Filters - Visible Events

ACL Filter, ARP, BGP, Bluetooth, CAPWAP, CDP, DHCP, DHCPv6, DNS, DTP, EAPOL, EIGRP, EIGRPv6, FTP, H.323, HSRP, HSRPv6, HTTP, HTTPS, ICMP, ICMPv6, IPsec, ISAKMP, IoT, IoT TCP, LACP, LLDP, Meraki, NDP, NETFLOW, NTP, OSPF, OSPFv6, PAgP, POP3, PPP, PPPoE, PTP, RADIUS, REP, RIP, RIPng, RTP, SCCP, SMTP, SNMP, SSH, STP, SYSLOG, TACACS, TCP, TFTP, Telnet, UDP, USB, VTP

Edit Filters Show All/None

# Simulation (5/5)

Simulation Panel				
Event List				
Vis.	Time(sec)	Last Device	At Device	Type
	0.000	--	PC1	ICMP
	0.000	--	PC1	ARP

PDU Information at Device: PC1

OSI Model **Outbound PDU Details**

At Device: PC1  
Source: PC1  
Destination: PC5

## In Layers

Layer7
Layer6
Layer5
Layer4
Layer3
Layer2
Layer1

## Out Layers

Layer7
Layer6
Layer5
Layer4
Layer 3: IP Header Src. IP: 192.168.1.1, Dest. IP: 192.168.1.3 ICMP Message Type: 8
Layer 2:
Layer1

1. The Ping process starts the next ping request.
2. The Ping process creates an ICMP Echo Request message and sends it to the lower process.
3. The source IP address is not specified. The device sets it to the port's IP address.
4. The device sets TTL in the packet header.
5. The destination IP address is in the same subnet. The device sets the next-hop to destination.

Challenge Me

<< Previous Layer

Next Layer >>

PDU Information at Device: PC1

OSI Model **Outbound PDU Details**

PDU Formats

IP									
0	4	8	16	20	24				Bits
VER:4		IHL		DSCP:0x00		TL:28			
ID:0x0002				FLAGS:0x0		FRAG OFFSET:0x000			
TTL:255		PRO:0x01		CHKSUM					
SRC IP:192.168.1.1									
DST IP:192.168.1.3									
OPT:0x00000000						PADDING:0x00			
DATA (VARIABLE LENGTH)									

ICMP

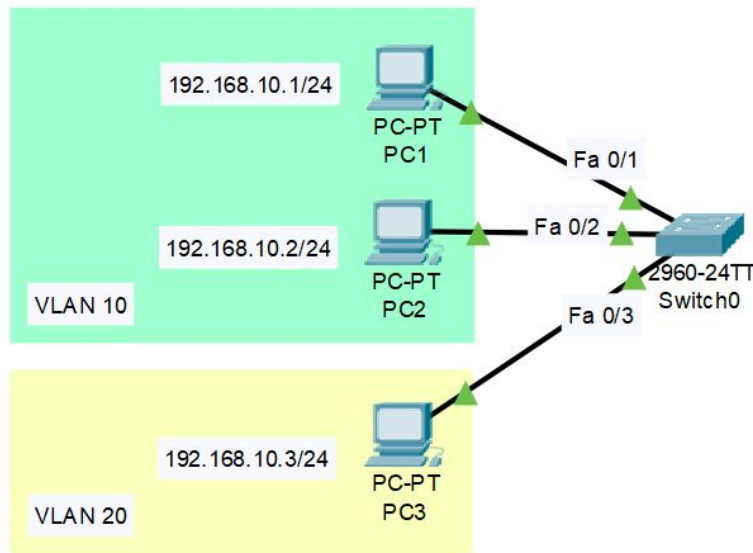
# Outline

1. 學理先備知識
2. 實驗目的
3. 實驗環境
4. 情境與指令介紹
5. **小作業 (不算分)**

# 作業要求(1) VLAN

設定：

1. 使用 Cisco 2960 當作 Switch
2. 建立三個 end point (PC)
  - PC1: 192.168.10.1/24
  - PC2: 192.168.10.2/24
  - PC3: 192.168.10.3/24
3. 在 2960 上為各個 PC 設定 VLAN 10, 20



# 作業要求(1) VLAN

回答以下問題：

1. 監聽封包，觀察在 Interface 上 VLAN 前後，PC1 是否可以碰到 PC2，為什麼？  
請附上在加上 VLAN 之後封包的截圖
2. 監聽封包，觀察在 Interface 上 VLAN 前後，PC1 是否可以碰到 PC3，為什麼？  
請附上在加上 VLAN 之後封包的截圖
3. 根據截圖，說明 boardcast domain 的變化 (hint: ARP)



## 作業要求(2) Trunk link

建出下方拓模



想想如果不透過 Trunk link, 該如何實作 ? 中的鏈路, 讓 PC1、PC2、PC4 可以互通, PC3、PC5 可以互通

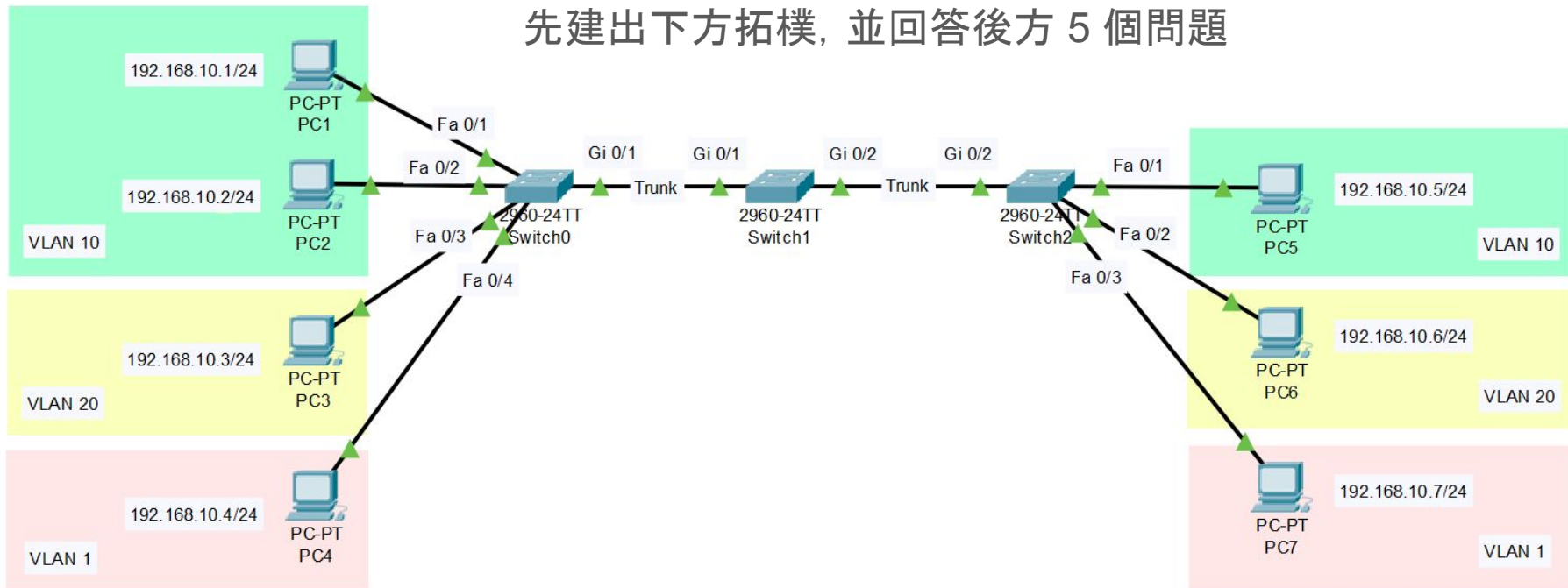
## 作業要求(2) Trunk link

回答以下問題：

1. 附上拓樸的截圖
2. 解釋如何實作？部分的線路
3. 說明 trunk link 的優點, trunk link 與？之間的線路比較

# 作業要求(3) Native VLAN

先建出下方拓模, 並回答後方 5 個問題

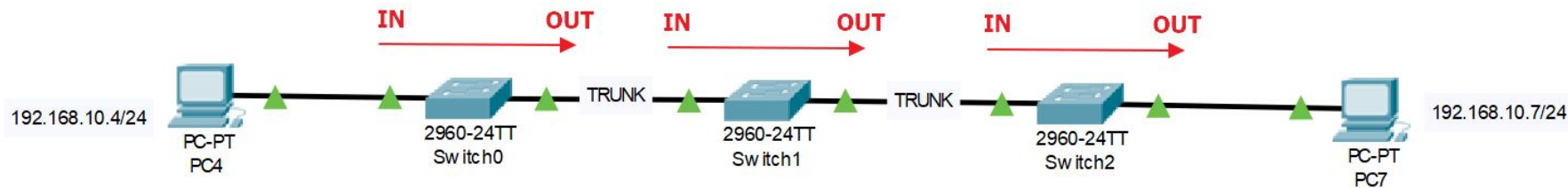


## 問題 3-1

- 在沒有設定 Native VLAN 的情況，觀察 PC4 到 PC7 的封包，並依標記規則(詳見後方投影片)直接在圖上標記答案

## 作業要求(3) 標記規則

請複製下圖，並以**文字**標示在圖上來回答題組，箭頭方向為封包傳送方向

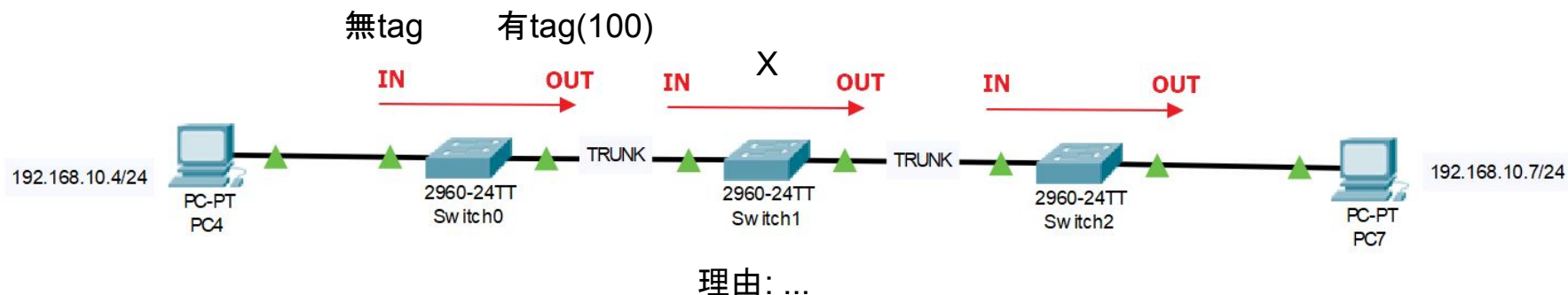


### 標記規則

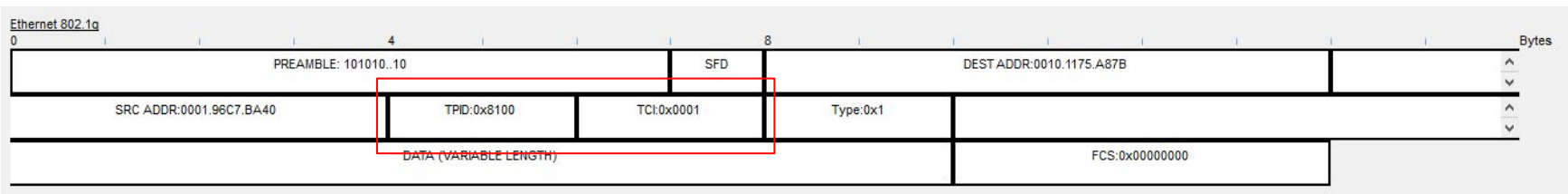
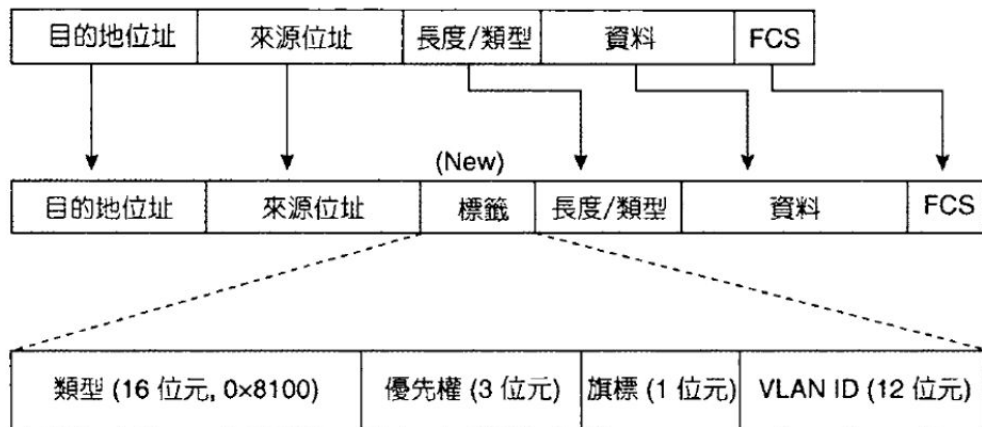
- 在圖上標示 IN/OUT 的附近標示該位置是否帶有 VLAN tag (IN, OUT 都要標)
- 有 VLAN tag: **有tag(vlan-id)**
- 無 VLAN tag: **無tag**
- 如果 PC4 傳送不到 PC7, 請直接以 **X** 標示在錯誤發生**起始點**, 並**說明理由**

# 作業要求(3) Native VLAN

舉例來說：

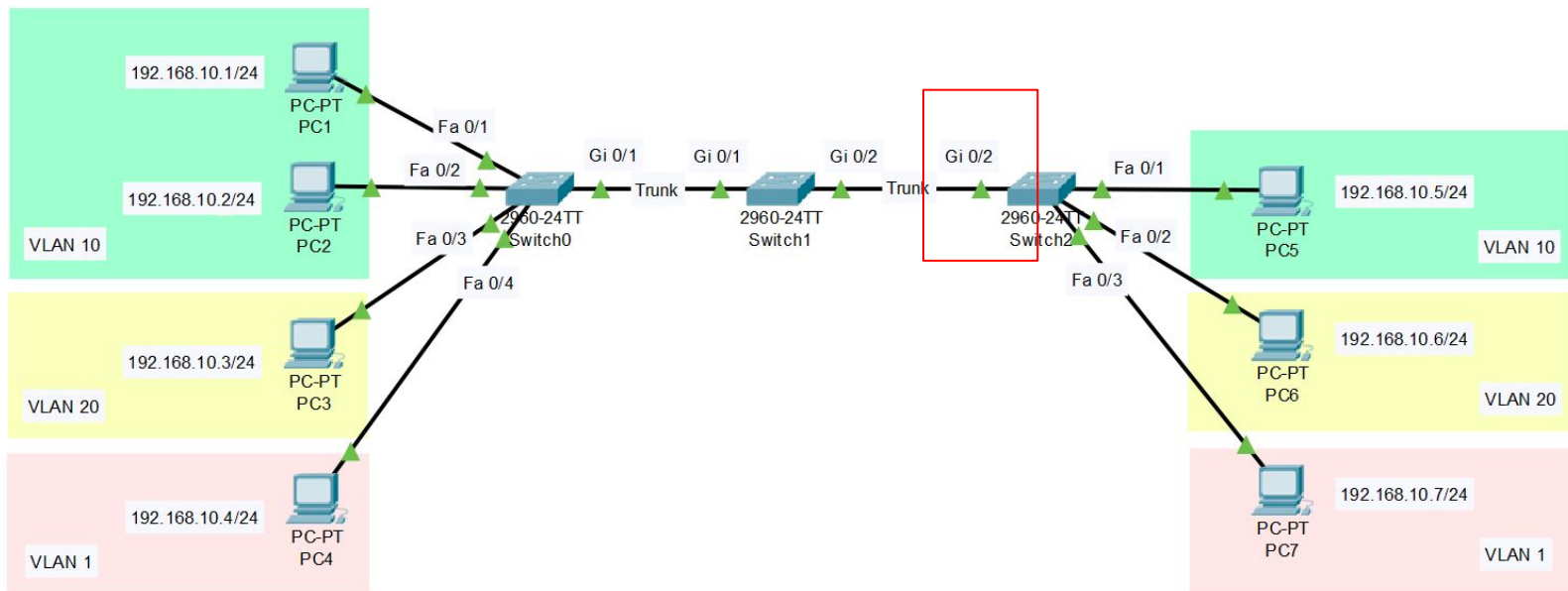


# 補充: 怎麼看 VLAN tag



## 問題 3-2

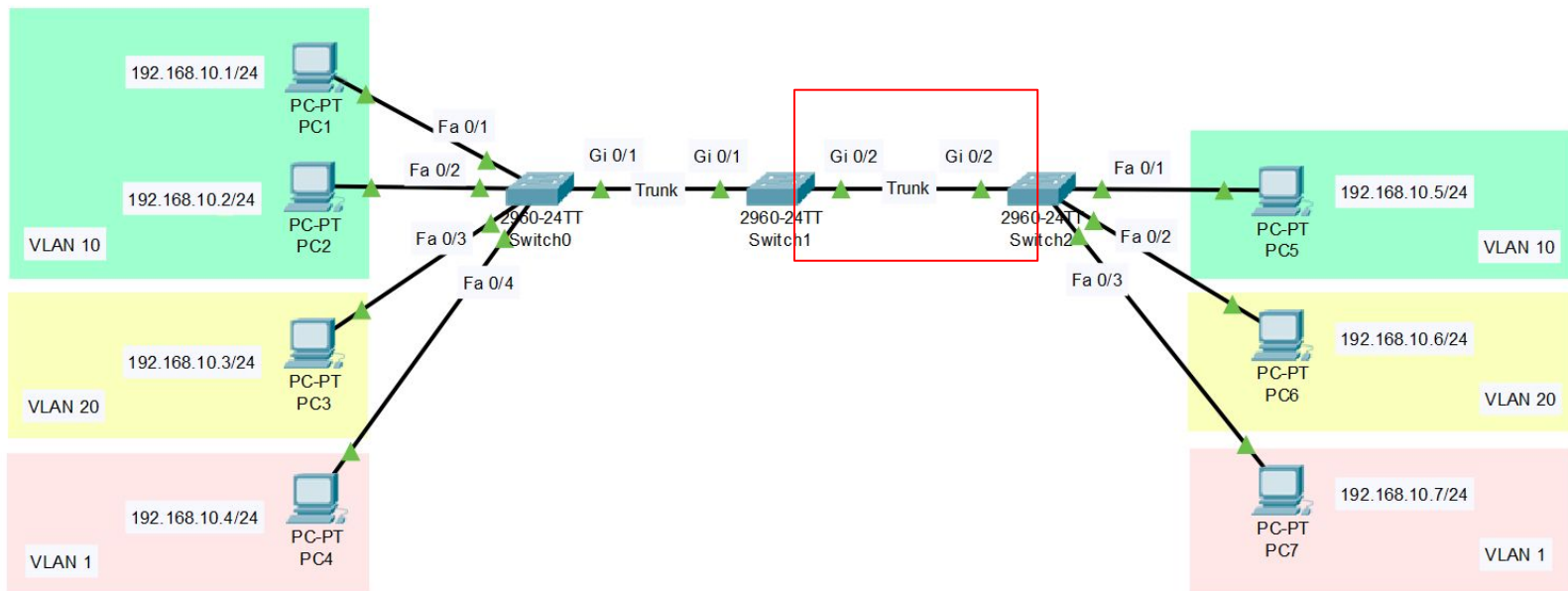
- 將 Switch2 Gi0/2 的 Native VLAN 設為 20，觀察 PC4 到 PC7 的封包，並依標記規則直接在圖上標記答案





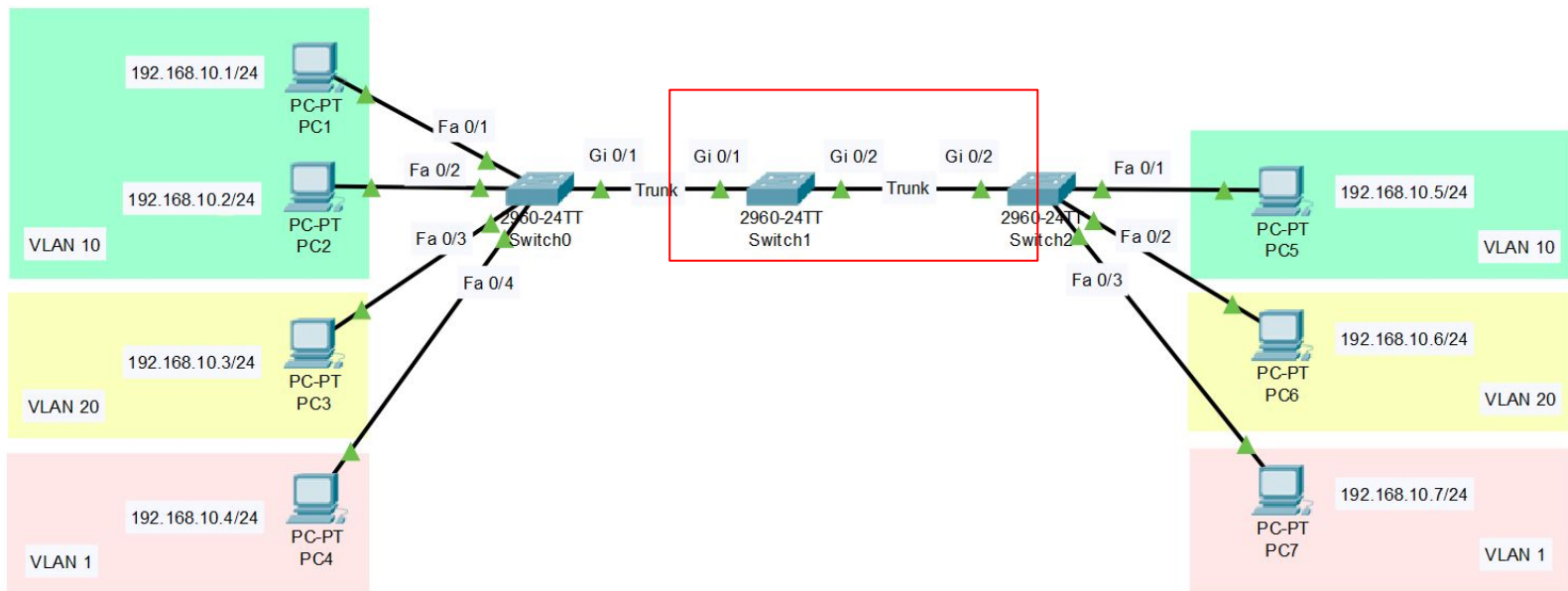
## 問題 3-3

- 接續問題 3-2, 將 Switch1 Gi0/2, Switch2 Gi 0/2 的 Native VLAN 都設為 20, 觀察 PC4 到 PC7 的封包, 並依標記規則直接在圖上標記答案



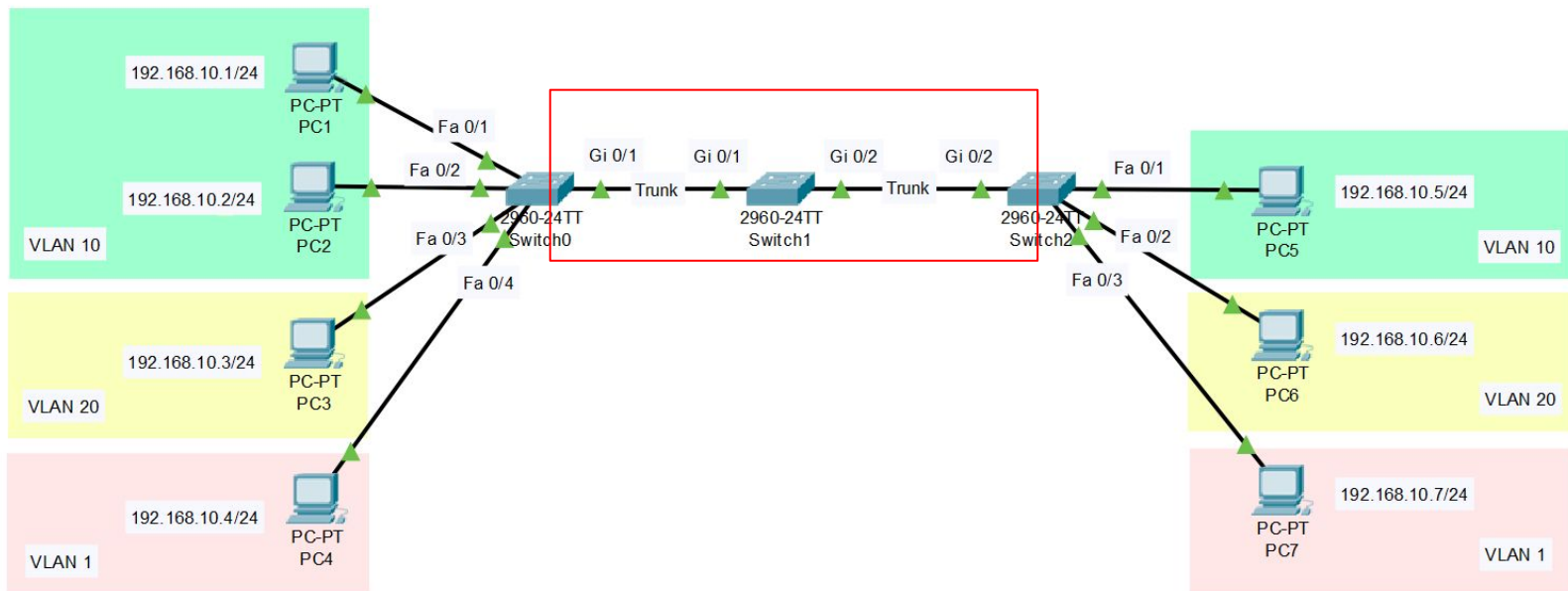
## 問題 3-4

- 接續問題 3-3, 再將 Switch1 Gi0/1 的 Native VLAN 設為 20, 觀察 PC4 到 PC7 的封包, 並依標記規則直接在圖上標記答案



## 問題 3-5

- 接續問題 3-4，將全部的 Native VLAN 設為 20，觀察 PC4 到 PC7 的封包，並依標記規則直接在圖上標記答案



## 附錄: 檢驗指令

- **show running-config**
- **show interfaces [status | trunk]**
- **show vlan id *vlan-id***
- **show vlan [brief]**
- **show interface *type member/module/name* switchport**
- **show ip route**