# Virtual LAN

SWITCH Module 2

# Agenda

- **Introduction to VLAN Concept**
- **Basic Configuration**
- **Trunking Introduction**
- **Dynamic Trunking Protocol**
- **Virtual Trunking Protocol**
- **Etherchannel**

# Virtual LAN (VLAN)

- Creates separate, independent broadcast domains - individual VLANs are separated with each other
- Allows to virtualize physical LAN network
  - Physical network is shared among the VLANs
- *What do we get?*
  - Possibility to create several logical networks on the top of physical infrastructure
  - Separation of physical (geographical) topology from the logical topology
  - Possibility to create LAN networks based on e.g. department, project teams, applications etc.
- *How does world without VLAN look like?*
  - The one (almost) inseparable network

# Advantages of Using VLAN

- Easily move workstations on the LAN
- Easily add workstations to the LAN
- Easily change the LAN configuration
- Improved network security
    - Easily control network traffic
    - Network segmentation
    - Reduce the size of broadcast domain
- Save you $$$ because of only one physical infrastructure

# VLAN Types: Cisco Terminology

- **Default VLAN**
  - Cisco Catalyst VLAN1
  - Default VLAN is always active
  - All ports are in VLAN1 by default
  - Several management protocols using VLAN1 for communication (CDP, VTP, PaGP)
- **Native VLAN**
  - Specific for 802.1Q trunk
  - Data are forwarded without tag
- **Management VLAN**
  - `interface vlan` *`vl-id`*
  - Should not contain user’s ports
  - For remote management
- **Data VLAN**
  - User’s data communication
- **Voice VLAN**
  - Separate VLAN for VoIP
  - Sometimes referred to as „auxiliary VLAN“

# Statically Assigned Port

- Port can be assigned to a VLAN either statically or dynamically
- **Static assignment** is maintained by an administrator manually
  - Physical port on switch is assigned to a VLAN
  - Every port is member of some VLAN – there can not be unassigned ports
  - Also known as port-based, port-centric
  - *Advantages:* everything is under control, deterministic design
  - *Disadvantages:* burdensome process for administrator

# Dynamically Assigned Port

- **Dynamically assigned** port membership in a VLAN is decided by the switch dynamically
- Assignment can be according to:
    - MAC address
    - IP adrress
    - Type of protocol
    - Username/password
- Dynamic membership needs control mechanism
    - VLAN Membership Policy Server (VMPS) – Cisco proprietary
    - RADIUS + 802.1X – standardized solution
- „How to configure Catalyst switch as a VMPS“

# Internal Switch Implementation

- The basic support for VLAN can be add easily – CAM is extended with VLAN column

| VLAN | MAC | Interface |
|---|---|---|

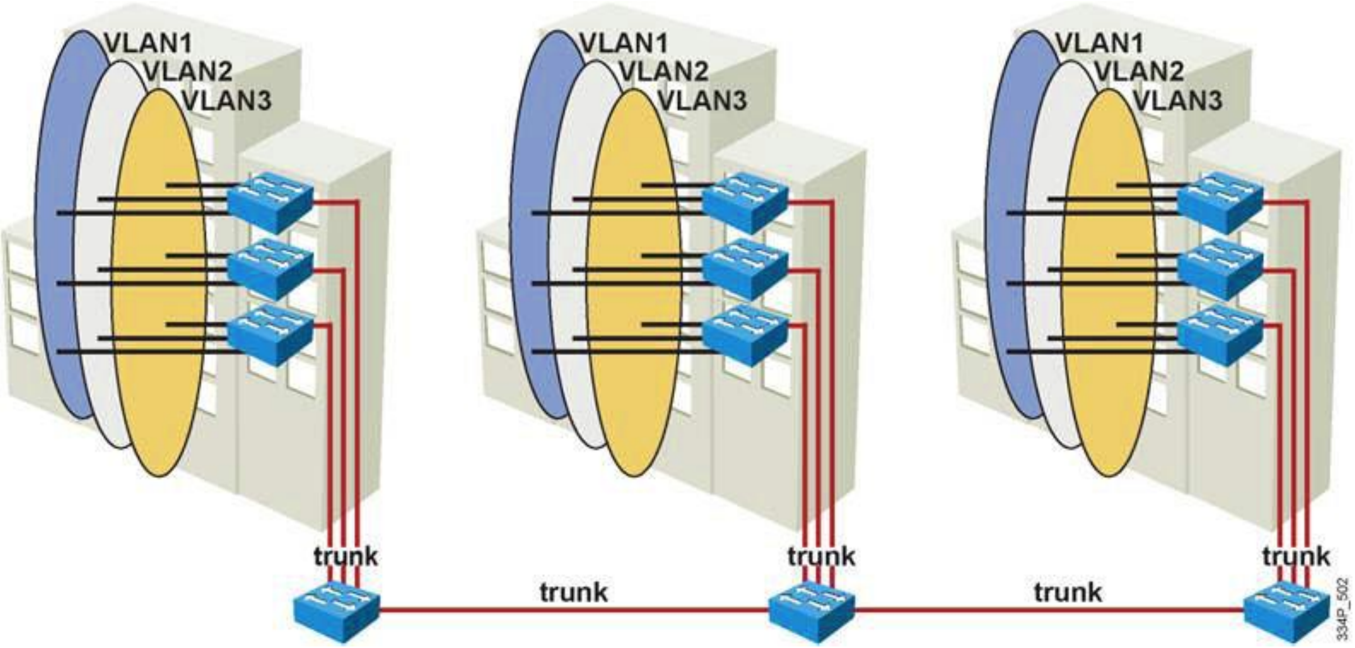
- The frame received at a physical port is processed as follows:
  - `IF` the source MAC address is unknown `THEN` a new record is created in the table together with a VLAN information of source port
  - `IF` MAC table is successfully looked up for the destination port `THEN` only ports with the same VLAN membership as the source port are taken into account

# VLAN Design

- VLAN allows flexibility
  - Users can be grouped into each VLAN
    - With regard to / regardless of physical location
    - Based on working team / job position
  - Separation of management / user network
  - Separation of voice, surveillance data etc.
- Two approaches how to design and manage VLAN
  - **End-to-End VLAN**
  - **Local VLAN**

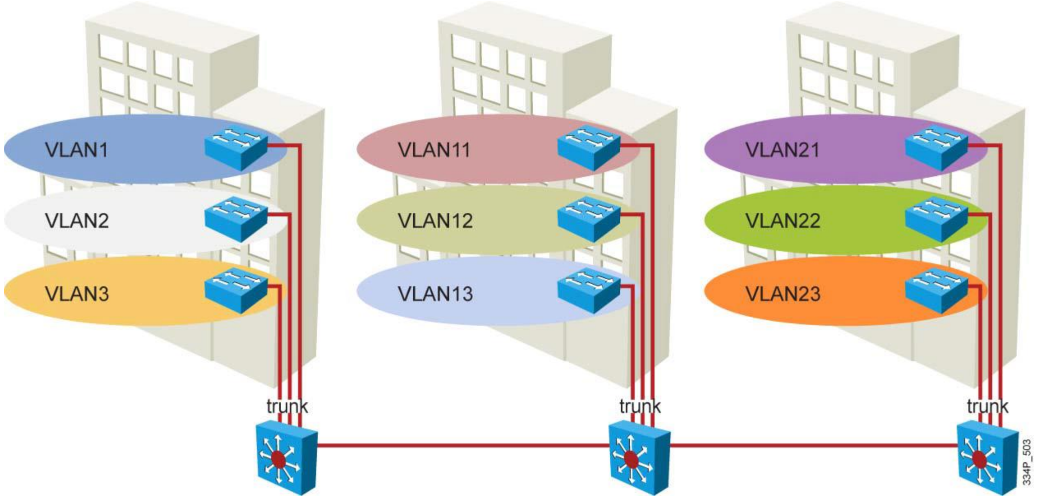
# End-to-End VLAN

- Original approach, helped facilitate the 80/20 rule and static IP configuration
- VLAN is widely dispersed throughout an enterprise network across Access, Distribution and Core layer
- The membership is according e.g. job function or department



# Local VLAN

- The VLANs are assigned based on physical location (wiring closet, building)
  - Sometimes referred to a geographic VLAN
  - Refers to 20/80 rule
  - Distribution switch allows to access to the different VLAN – L3 switching
- Local VLAN design is today the recommended approach
  - Smaller VLAN range allows to better manageability, smaller „failure domain“, simplify security and redundancy etc.



# End-to-end vs. Local VLAN

## End-to-End VLANs

*Pros:*

- Geographically dispersed users appear on the same segment
- Same policy (security, QoS) can be applied to the same group of users regardless of their physical location

*Cons:*

- All switches need to know all VLANs
- Broadcast messages flood all switches
- Troubleshooting may be challenging

## Local VLANs

*Pros:*

- Design is scalable
- Easier troubleshooting and predictable traffic flow
- Redundant paths can be built easily

*Cons:*

- More routing devices are required than in end-to-end models
- Users belong to the same broadcast domain when they are at the same location

# VLAN Range: Normal vs. Extended

- **Normal Range VLANs**
  - VLAN ID is in range 1 – 1005
  - IDs from 1002 to 1005 are reserved for Token Ring and FDDI VLAN
  - VLANs1, 1002–1005 are automatically created and cannot be deleted
  - VLAN configuration is saved in `vlan.dat` file in Flash memory
  - It is possible to save the configuration into startup-config
- **Extended Range VLANs**
  - VLAN ID is in range 1006–4094, only for Ethernet
  - Saved in startup-config
  - `IF` VTPv3 is used `THEN` it is also saved in `vlan.dat`
  - Configuration limits
    - VTPv1 and VTPv2 support them in VTP Transparent mode
    - VTPv3 supports extended range VLANs in all modes

# VLAN support on Catalyst switches

- The number of supported VLANs and VLAN ID range depend on platform (IOS) support

| Type | Max. number of VLAN | VLAN ID range |
|---|---|---|
| 2940 | 4 | 1 - 1005 |
| 2950/2955 | 250 | 1 - 4094 |
| 2960 | 255 | 1 - 4094 |
| 2970/3550/3560/3750 | 1055 | 1 - 4094 |
| 2848G/2980G/4000/4500 | 4094 | 1 - 4094 |
| 6500 | 4094 | 1 - 4094 |

# Static VLAN Configuration

# Creating VLAN

- Configuration steps
  1. Create a VLAN
  2. Verify the VLAN configuration
  3. Add ports to the VLAN
  4. Verify ports configuration
  5. Verify the VLAN operation
- Best practice switch management
  - Create a management VLAN
  - Create a new native VLAN for all trunks
  - Create a parking lot VLAN (inactive) and unused ports

# Creating a VLAN: Global Config Mode

```
Switch# conf t
Enter configuration commands, one per line.  End with CNTL/Z.
Switch(config)# vlan 2
Switch(config-vlan)# name Accounting
Switch(config-vlan)# vlan 3
Switch(config-vlan)# name Marketing
Switch(config-vlan)# end
Switch#
```

# Creating a VLAN: VLAN Database Config

```
Switch# vlan database
% Warning: It is recommended to configure VLAN from config mode,
  as VLAN database mode is being deprecated. Please consult user
  documentation for configuring VTP/VLAN in config mode.

Switch(vlan)# vlan 2
VLAN 2 added:
    Name: VLAN0002
Switch(vlan)# exit
APPLY completed.
Exiting....
Switch#
```

- Changes are applied after **`exit`** or **`apply`** command
- Only normal range VLANs are supported
- *Deprecated old fashioned way, but still present in NM-16ESW or in PacketTracer*

# Display VLAN Configuration

```
Switch# show vlan

VLAN Name                             Status    Ports
---- -------------------------------- --------- -------------------------
1    default                          active   Fa0/1, Fa0/2 ,Fa0/3,
                                               Fa0/4, Fa0/5, Fa0/6,
                                               Fa0/7, Fa0/8, Fa0/9,
                                               Fa0/10, Fa0/11, Fa0/12,
                                               Fa0/13, Fa0/14, Fa0/15,
                                               Fa0/16, Fa0/17, Fa0/18,
                                               Fa0/19, Fa0/20, Fa0/21,
                                               Fa0/22, Fa0/23, Fa0/24
                                               Gi0/1, Gi0/2
2    Accounting                       active
3    Marketing                        active
1002 fddi-default                     act/unsup
1003 token-ring-default               act/unsup
1004 fddinet-default                  act/unsup
1005 trnet-default                    act/unsup
... Output omitted ...
```

# Access Port

- **Access port** is a port associated with a VLAN
- Device belonging to the VLAN is in the same IP network segment with the other member of the VLAN
- Configuration options
  - Statically associated using manual configuration
  - Dynamically associated
    - Based on MAC address or login credentials
    - VMPS (VLAN Management Policy Server) or Radius + 802.1x

# Association the Port with VLAN

```
Switch# configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.
Switch(config)# int fa 0/1
Switch(config-if)# switchport mode access
Switch(config-if)# switchport access vlan 2
Switch(config-if)# exit
Switch(config)# int fa 0/2
Switch(config-if)# switchport mode access
Switch(config-if)# switchport access vlan 3
Switch(config-if)# end
Switch# show vlan
VLAN Name                             Status    Ports
---- -------------------------------- --------- -------------------------
 1 default                            active     Fa0/3, Fa0/4, Fa0/5, Fa0/6, Fa0/7,
                                                 Fa0/8, Fa0/9, Fa0/10, Fa0/11,
                                                 Fa0/12, Fa0/13, Fa0/14, Fa0/15,
                                                 Fa0/16, Fa0/17, Fa0/18, Fa0/19,
                                                 Fa0/20, Fa0/21, Fa0/22, Fa0/23,
                                                 Fa0/24, Gi0/1, Gi0/2
2 Accounting                          active     Fa0/1
3 Marketing                           active     Fa0/2
```

*Creating an access port and association the port to the VLAN*

# Useful Macro `switchport host`

```
Switch# configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.
Switch(config)# int fa 0/1
Switch(config-if)# switchport host
switchport mode will be set to access
spanning-tree portfast will be enabled
channel group will be disabled
Switch(config-if)# switchport access vlan 2
Switch(config-if)# exit
```

# Adding Several Ports to the VLAN

```
Switch# conf t
Enter configuration commands, one per line.  End with CNTL/Z.
Switch(config)# interface range fa 0/1 - 5
Switch(config-if-range)# switchport mode access
Switch(config-if-range)# switchport access vlan 2
Switch(config-if-range)# end
Switch# show vlan
VLAN Name                     Status        Ports
---- ------------------------ ---------     ----------------------------
1    default                  active        Fa0/6, Fa0/7, Fa0/8, Fa0/9,
                                            Fa0/10, Fa0/11, Fa0/12,
                                            Fa0/13, Fa0/14, Fa0/15,
                                            Fa0/16, Fa0/17, Fa0/18,
                                            Fa0/19, Fa0/20, Fa0/21,
                                            Fa0/22, Fa0/23, Fa0/24
                                            Gi0/1, Gi0/2
2    Accounting               active        Fa0/1, Fa0/2, Fa0/3, Fa0/4,
                                            Fa0/5
...
```

# Creating the VLAN Indirectly

```
Switch# conf t
Enter configuration commands, one per line.  End with CNTL/Z.
Switch(config)# int fa 0/1
Switch(config-if)# switchport mode access
Switch(config-if)# switchport access vlan 2
% Access VLAN does not exist. Creating vlan 2
Switch(config-if)# end
%SYS-5-CONFIG_I: Configured from console by console
Switch# sh vlan
VLAN Name                 Status            Ports
---- -------------------- ---------------- -------------------------
 1   default               active            Fa0/2, Fa0/3, Fa0/4, Fa0/5
                                             Fa0/6, Fa0/7, Fa0/8, Fa0/9,
                                             Fa0/10, Fa0/11, Fa0/12,
                                             Fa0/13, Fa0/14, Fa0/15,
                                             Fa0/16, Fa0/17, Fa0/18,
                                             Fa0/19, Fa0/20, Fa0/21,
                                             Fa0/22, Fa0/23, Fa0/24
                                             Gi0/1, Gi0/2
 2   VLAN0002             active            Fa0/1
```

# Useful Verifying Commands

```
show vlan

show vlan brief

show vlan id ID_VLAN

show vlan name VLAN_NAME

show vlan summary

show interfaces switchport

show run vlan
```

# Verifying VLAN port configuration

```
Switch# show int fa 0/1 switchport
Name: Fa0/1
Switchport: Enabled
Administrative Mode: static access
Operational Mode: static access
Administrative Trunking Encapsulation: dot1q
Operational Trunking Encapsulation: native
Negotiation of Trunking: Off
Access Mode VLAN: 2 (VLAN0002)
Trunking Native Mode VLAN: 1 (default)
Voice VLAN: none
Administrative private-vlan host-association: none
Administrative private-vlan mapping: none
Administrative private-vlan trunk native VLAN: none
Administrative private-vlan trunk encapsulation: dot1q
Administrative private-vlan trunk normal VLANs: none
Administrative private-vlan trunk private VLANs: none
Operational private-vlan: none
Trunking VLANs Enabled: ALL
Pruning VLANs Enabled: 2-1001
Capture Mode Disabled
Capture VLANs Allowed: ALL
Protected: false
Appliance trust: none
```

# Delete the VLAN Configuration

```
! Delete vlan.dat (restart needed)
Switch# delete flash:vlan.dat

! Remove VLAN 5
Switch(config)# no vlan 5

! VLAN database mode
Switch# vlan database
Switch(vlan)# no vlan 5
Switch(vlan)# exit

! Remove port from the VLAN(port will return to the VLAN 1)
Switch(config)# interface fastethernet 0/5
Switch(config-if)# no switchport access vlan
```

# Suspending VLAN

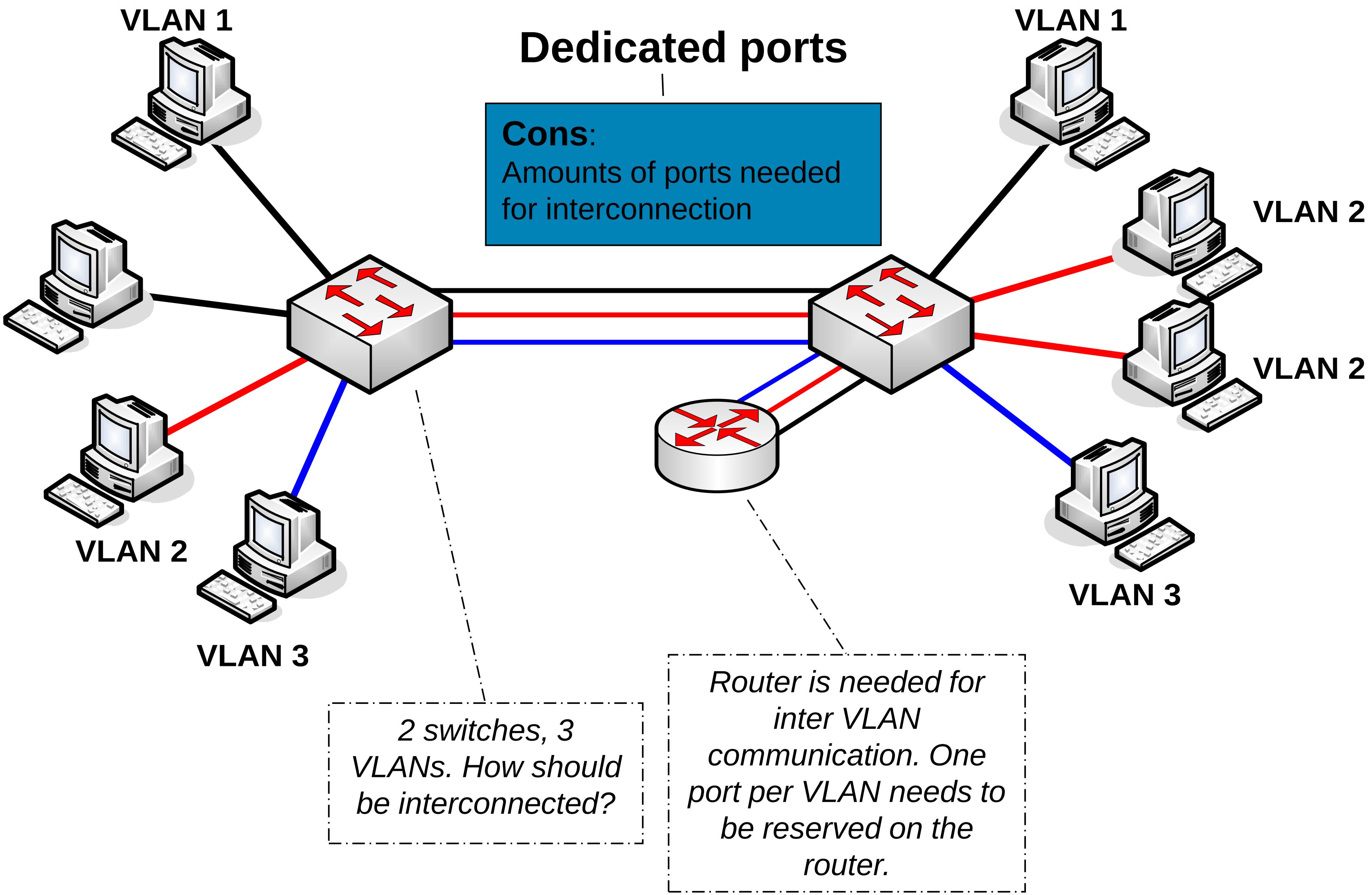
- VLAN can be administratively deactivated
  - VLAN still exists in the database but the traffic is discarded (user traffic, STP, …)
  - VLAN can be stopped either locally or globally via VTP

```
Switch(config)# vlan 99
Switch(config-vlan)# state suspend ! Globally using VTP
Switch(config-vlan)# shutdown ! Locally
```

- **`state suspend`** – suspend the VLAN globally, the command is propagated through whole VTP domain
- **`shutdown`** – suspend the VLAN on the selected switch
- Best practice: *It is useful to have one suspended VLAN for unused ports* – **parking lot VLAN**

# VLAN Trunking

# Intra VLAN Communication: Dedicated Ports



**Dedicated ports**

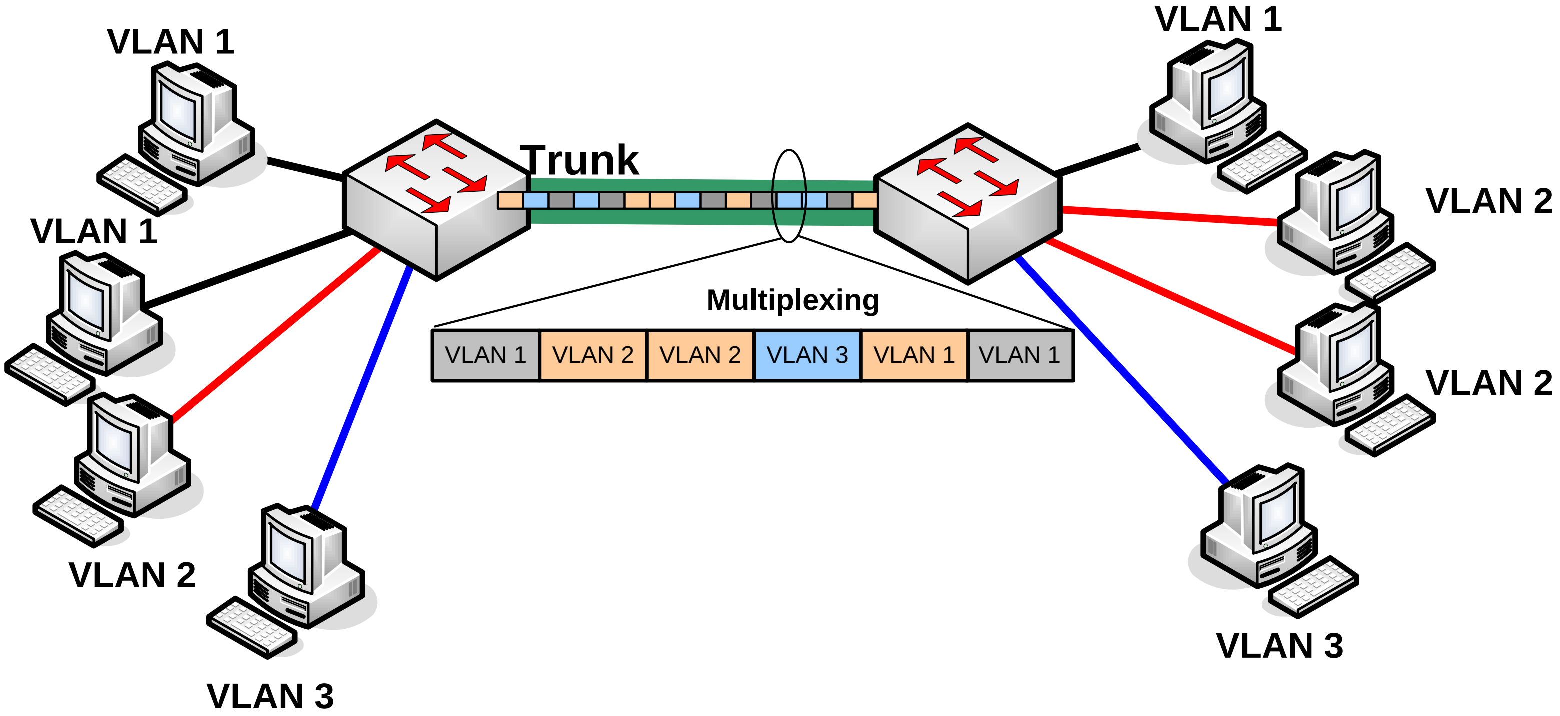
**Cons**:
Amounts of ports needed for interconnection

*2 switches, 3 VLANs. How should be interconnected?*

*Router is needed for inter VLAN communication. One port per VLAN needs to be reserved on the router.*

# Intra VLAN Communication: Trunking

- *How to distinguish to which VLAN each frame belongs?*

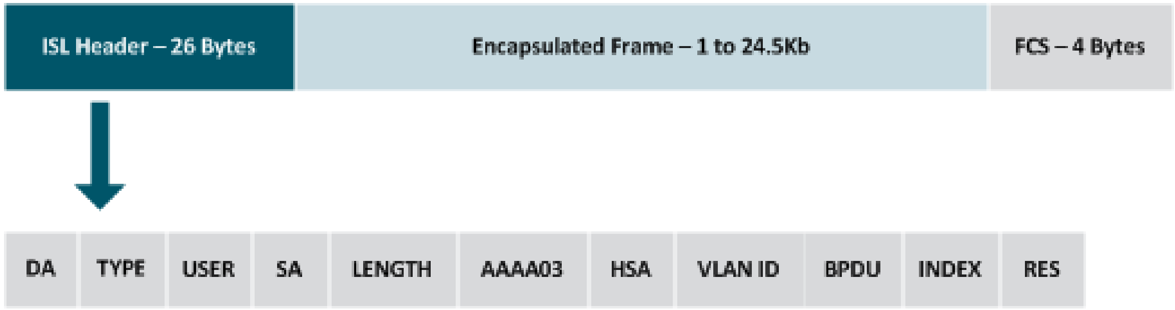


## Trunk

- Used to carry traffic that belongs to multiple VLANs between devices over the same link
- Frames are multiplexed

# Trunk protocols: ISL and 802.1Q

- Trunk protocols determine how the frames will be (de)multiplexed over the single trunk link
  - A trunking protocol marks the frame to identify its associated VLAN
- **Inter-Switch Link Protocol (ISL)**
  - Cisco proprietary
  - Original frame is encapsulated with the ISL header
  - ISL header 26B + original frame + 4B CRC
  - Cisco Document ID: 17056, „Inter-Switch Link and IEEE 802.1Q Frame Format“



| ISL Header – 26 Bytes | Encapsulated Frame – 1 to 24.5Kb | FCS – 4 Bytes |
|---|---|---|

| DA | TYPE | USER | SA | LENGTH | AAAA03 | HSA | VLAN ID | BPDU | INDEX | RES |
|---|---|---|---|---|---|---|---|---|---|---|

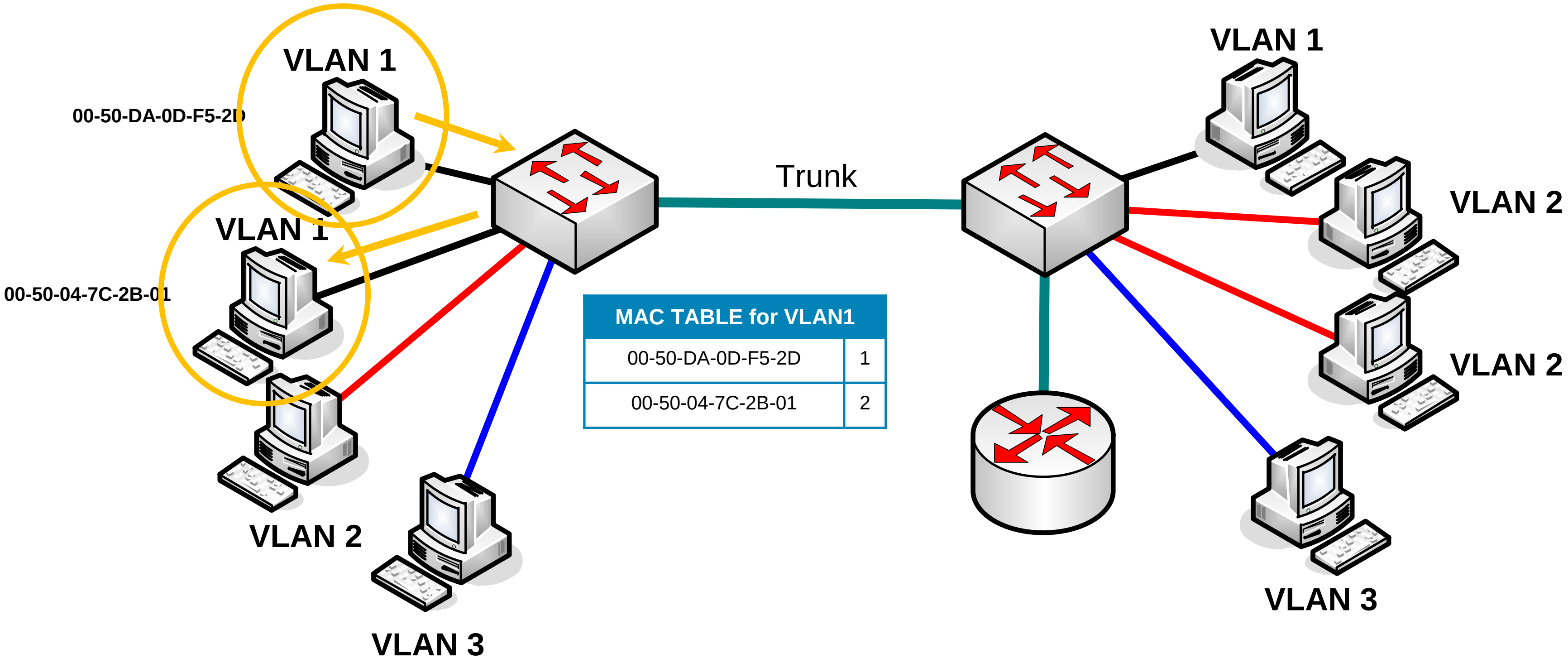
# 802.1Q

- 802.1Q is the IEEE standard VLAN trunking protocol
- Cross vendor interoperability
- 802.1Q inserts a tag into the original Ethernet header
  - Tag identifies a VLAN the frame belongs to
  - Tag is inserted into the frame – it is not encapsulation
- Tag is inserted
  - Between the fields Source MAC and Type/Length
  - Each frame (with exception) is tagged on the trunk link
  - FCS needs to be recalculated

# 802.1Q: Frame Transfer

- Sending switch
  1. Inserts the 4B  tag into the frame
  2. Recalculates the FCS field
  3. Sends the frame through the trunk
- Receiving switch
  1. Checks the FCS
  2. Analyzes the tag field and remove the tag from the frame
  3. Sends the frame to the appropriate VLAN
- End stations are not aware of tagging
  - Access ports receive the original frame
  - The tagging process is transparent to the end station

# 802.1Q: Intra VLAN Communication ①



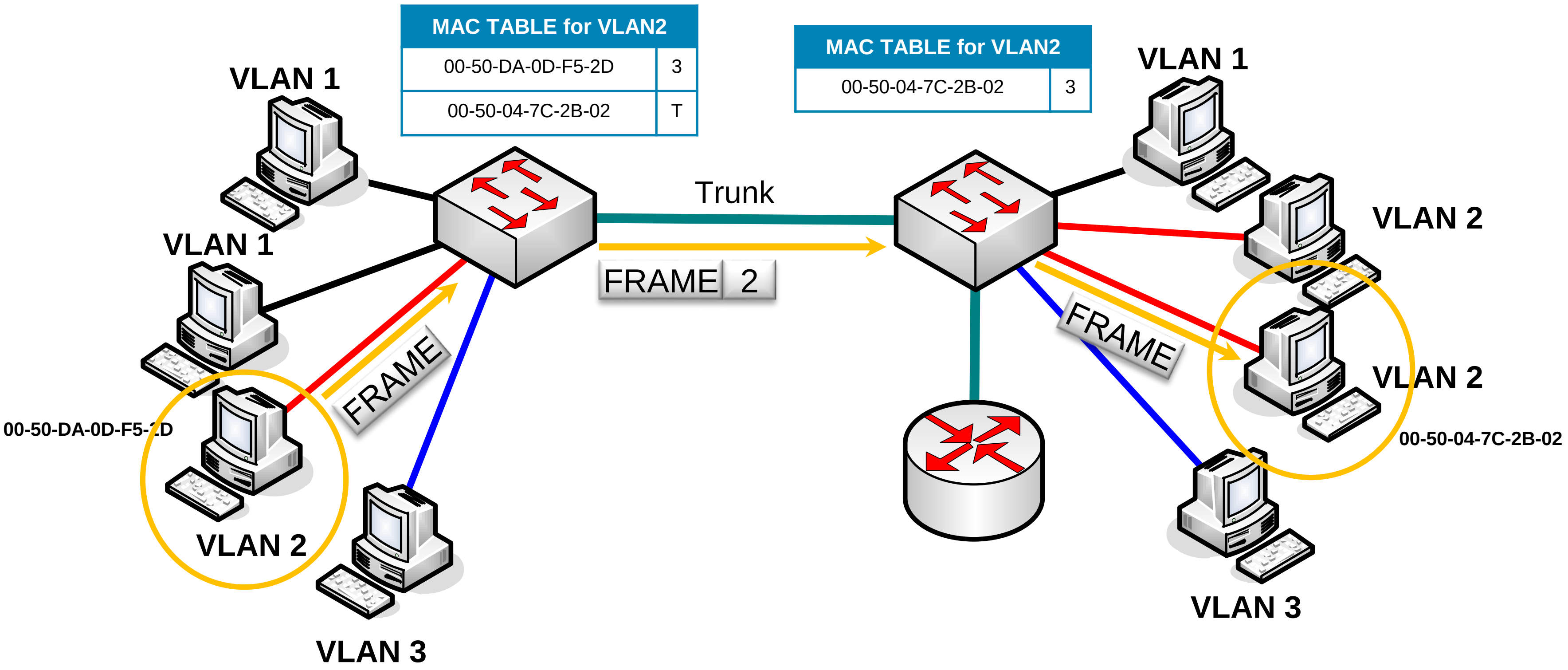
| MAC TABLE for VLAN1 | |
|---|---|
| 00-50-DA-0D-F5-2D | 1 |
| 00-50-04-7C-2B-01 | 2 |

***Example:***
*Communication between end stations in the same VLAN on the same switch*

1. *Receive the frame on an access port*
2. *Lookup in the appropriate VLAN MAC table*
3. *Send the frame to the port*

*The frame isn't sent through the trunk port thus it is not modified/tagged.*

# 802.1Q: Intra VLAN Communication ②



| MAC TABLE for VLAN2 | |
|---|---|
| 00-50-DA-0D-F5-2D | 3 |
| 00-50-04-7C-2B-02 | T |

| MAC TABLE for VLAN2 | |
|---|---|
| 00-50-04-7C-2B-02 | 3 |

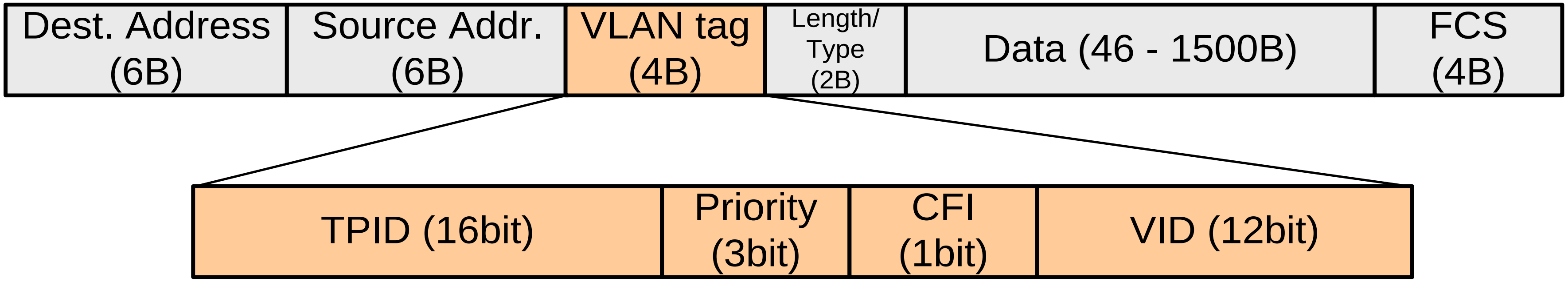
***Example:*** *Communication between end stations in the same VLAN and between distinct switches*

1. *Receive the frame on an access port*
2. *Lookup in the appropriate VLAN MAC table*
3. *Tag the frame*
4. *Send the tagged frame through the trunk port*

1. *Receive the frame on the trunk port*
2. *Lookup in the appropriate VLAN MAC table*
3. *If the end station is physically connected – remove the tag*

*The frame is sent through the trunk port tagged*

# 802.1Q: Frame Format



| Dest. Address (6B) | Source Addr. (6B) | VLAN tag (4B) | Length/ Type (2B) | Data (46 - 1500B) | FCS (4B) |
|---|---|---|---|---|---|

| TPID (16bit) | Priority (3bit) | CFI (1bit) | VID (12bit) |
|---|---|---|---|

- **TPID (Tag Protocol Identifier)**: 16 bits
  - Identifies the frame as the IEEE802.1Q frame, value is set to 0x8100
- **Priority (Class of Service)**: 3 bits
  - Frame priority according the 802.1p (CoS)
- **Drop eligible indicator (DEI)**  1bit
  - Formerly **CFI (Canonical Format Indicator)** bit used for signalling if MAC address is in canonical form or not: 0 for Ethernet, 1 for Token Ring/FDDI
  - Currently: can indicate frames eligible to be dropped in the presence of congestion
- **VID (VLAN Identifier)**: 12 bits
  - Specifying the VLAN to which the frame belongs, range 0-4095

# Native VLAN ①

- Cisco use the concepts of trunk ports and the native VLAN for that trunk
  - Native VLAN does not use tag
  - Each trunk port has own native VLAN
  - `IF` the frame belongs to native VLAN `THEN` it is sent untagged through the trunk port
  - `IF` the received frame on the trunk port does not have tag `THEN` it is put into the native VLAN
- The native VLAN must be the same on both ends of a trunk when using 802.1Q
  - VLAN 1 by default
  - Otherwise **native VLAN mismatch** – two different native VLANs are merged into one larger

# Native VLAN ②

- Native VLAN is confusing (Cisco vs. other vendors)
  - „802.1q trunking between different vendors“
- Best practices
  - Create separate unused VLAN as native VLAN on all trunks
  - Do not use VLAN1 as native VLAN
- Avoid using:
  - Access VLAN for end station same as native VLAN on trunk port
  - Management VLAN as native VLAN
- Using native VLAN can be deactivated (>3560)
  - Global configuration command

```
Switch(config)# vlan dot1q tag native
```

# Trunk Configuration

# Trunk Configuration

- Manual (static)

```
Switch(config-if)# switchport trunk encapsulation { dot1q | isl }
Switch(config-if)# switchport mode trunk
```

- Dynamically – using Dynamic Trunking Protocol (DTP)
    - Trunk is created dynamically
    - Both ISL and 802.1Q are supported

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

# Static Configuration

```
! Needed for  3550, 3560 (not required for platform 2950, 2960)
Switch(config-if)# switchport trunk encapsulation { isl | dot1q }

! Trunk configuration
Switch(config-if)# switchport mode trunk
Switch(config-if)# switchport nonegotiate ! Turn DTP off - recommended
Switch(config-if)# switchport trunk native vlan VLAN_ID
Switch(config-if)# switchport trunk allowed vlan ?
  WORD    VLAN IDs of allowed VLANs when this port is in trunking mode
  add     add VLANs to the current list
  all     all VLANs
  except  all VLANs except the following
  none    no VLANs
  remove  remove VLANs from the current list
```

# Verify the Trunk Configuration ①

```
Switch# show interface trunk
Port        Mode         Encapsulation  Status        Native vlan
Gig1/1      on           802.1q         trunking      99
Gig1/2      auto         n-802.1q       trunking      99


Port        Vlans allowed on trunk
Gig1/1      1-1005
Gig1/2      1-1005

Port        Vlans allowed and active in management domain
Gig1/1      1,99,1002,1003,1004,1005
Gig1/2      1,99,1002,1003,1004,1005

Port        Vlans in spanning tree forwarding state and not pruned
Gig1/1      1,99,1002,1003,1004,1005
Gig1/2      1,99,1002,1003,1004,1005

Switch#
```

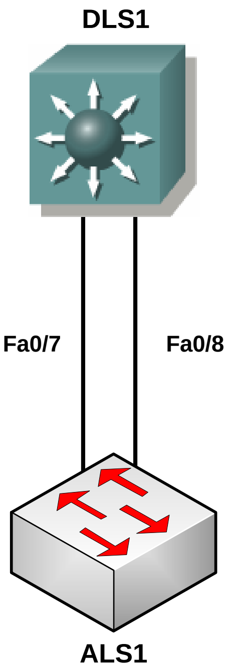
# Verify the Trunk Configuration ②

```
Switch#sh int gi 1/1 switchport
Name: Gig1/1
Switchport: Enabled
Administrative Mode: trunk
Operational Mode: trunk
Administrative Trunking Encapsulation: dot1q
Operational Trunking Encapsulation: dot1q
Negotiation of Trunking: On
Access Mode VLAN: 1 (default)
Trunking Native Mode VLAN: 99 (Management)
Voice VLAN: none
Administrative private-vlan host-association: none
Administrative private-vlan mapping: none
Administrative private-vlan trunk native VLAN: none
Administrative private-vlan trunk encapsulation: dot1q
Administrative private-vlan trunk normal VLANs: none
Administrative private-vlan trunk private VLANs: none
… output omitted
```

**switchport mode trunk** *command puts the port into the trunk mode permanently*

*DTP is still operational, so the far-end switch port can negotiate trunking*

# Static Trunk and Static Access Issue



```
DLS1(config)#int ran fa 0/7 - 8
DLS1(config-if-range)# switchport trunk encapsulation dot1q
DLS1(config-if-range)# switchport mode trunk
DLS1# sh int trunk

Port        Mode             Encapsulation  Status        Native vlan
Fa0/7       on               802.1q         trunking      1
Fa0/8       on               802.1q         trunking      1
```

```
ALS1# show interface trunk

Port        Mode             Encapsulation  Status        Native vlan
Fa0/7       on               802.1q         trunking      1
Fa0/8       on               802.1q         trunking      1

ALS1(config)# int fa 0/8
ALS1(config-if)# switchport mode access
ALS1(config-if)# ^Z
ALS1# show interface trunk

Port        Mode             Encapsulation  Status        Native vlan
Fa0/7       auto             802.1q         trunking      1
```

```
DLS1# show interface trunk

Port        Mode             Encapsulation  Status        Native vlan
Fa0/7       on               802.1q         trunking      1
Fa0/8       on               802.1q         trunking      1
```

# Dynamic Trunking Protocol

# Dynamic Trunking Protocol (DTP)

- Cisco proprietary protocol (U.S. Patent 6445715)
- Allows to create a trunk automatically by sending the DTP frames between switches
- Not supported on all Cisco boxes
  - Usually supported only on switches
  - Routers DO NOT understand the DTP and DO NOT generate the DTP messages
  - DTP DOES NOT have affect on trunk activity (sending/receiving frames, tagging, encapsulating etc.)

# DTP Modes

- **Dynamic Auto**
  - Default for platforms 2960, 3560
  - The local switch port advertises to the remote switch port that it is able to trunk but does not request to go to the trunking state

```
Switch(config-if)# switchport mode dynamic auto
```

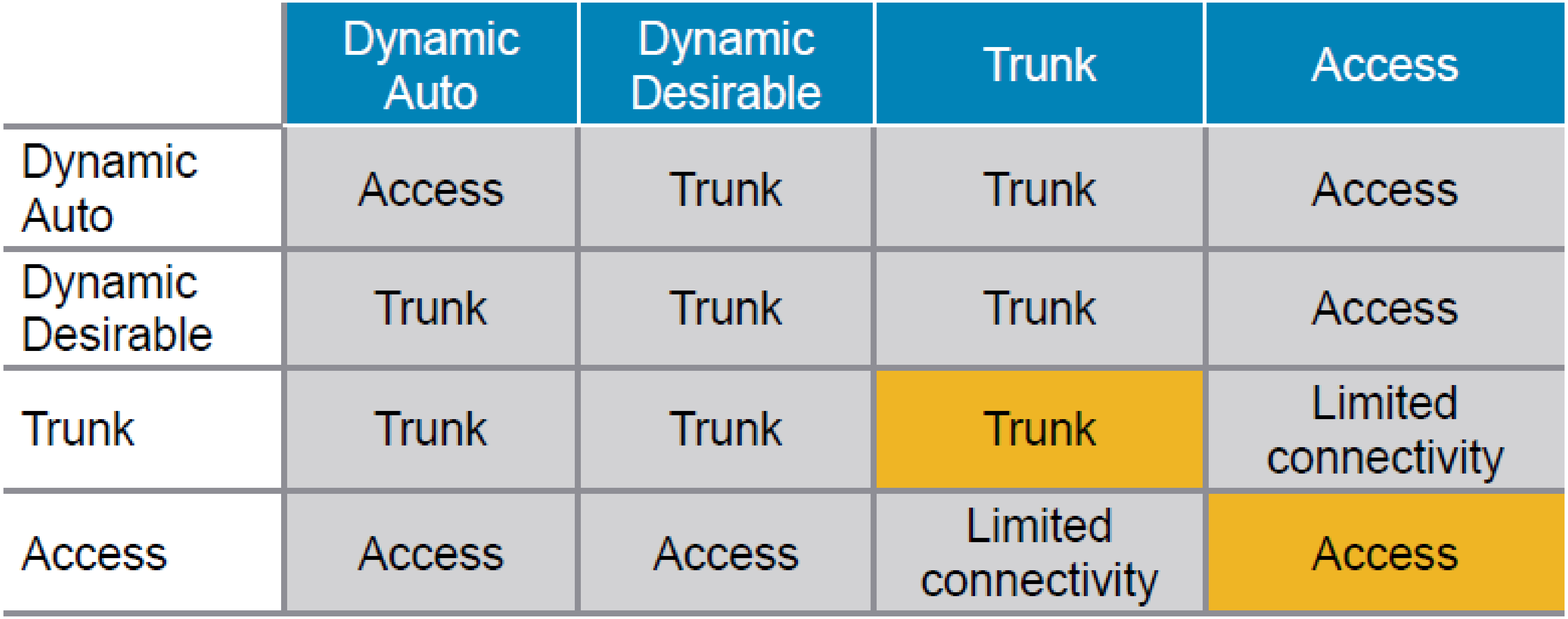
- **Dynamic Desirable**
  - Default for platform 2950, 3550
  - The local switch port advertises to the remote switch port that it is able to trunk and asks the remote switch port to go to the trunking state

```
Switch(config-if)# switchport mode dynamic desirable
```

- **Static configuration – trunk („On“)**
  - The local port is, regardless of what DTP information the remote port sends as a response to the advertisement, in the trunking state
- **Static configuration  („Off“)**
  - Trunk is not allowed on the port
- **Nonegotiate**
  - DTP is turned off
  - No DTP frames are being sent
  - Makes sense only for ports in static trunk configuration

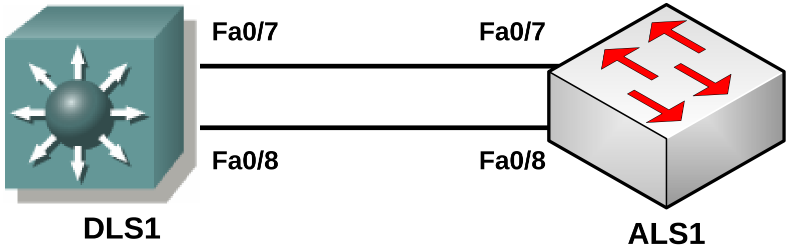
```
Switch(config-if)# switchport nonegotiate
```

# Switchport Mode Interactions



| | Dynamic Auto | Dynamic Desirable | Trunk | Access |
|---|---|---|---|---|
| Dynamic Auto | Access | Trunk | Trunk | Access |
| Dynamic Desirable | Trunk | Trunk | Trunk | Access |
| Trunk | Trunk | Trunk | Trunk | Limited connectivity |
| Access | Access | Access | Limited connectivity | Access |

# DTP: Static Trunk vs. Dynamic Auto



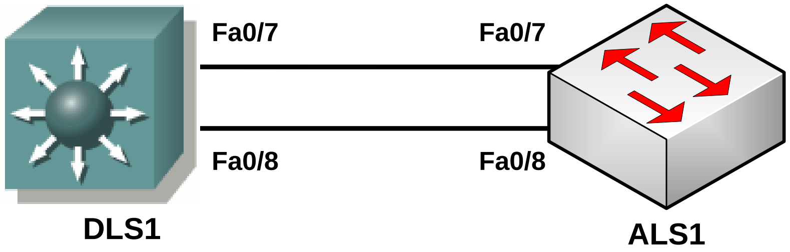
```
DLS1(config)#int ran fa 0/7 - 8
DLS1(config-if-range)#switchport trunk encapsulation dotlq
DLS1(config-if-range)#switchport mode trunk
DLS1#sh int trunk

Port      Mode        Encapsulation  Status      Native vlan
Fa0/7     on          802.1q         trunking    1
Fa0/8     on          802.1q         trunking    1

DLS1#show interface fa 0/7 switchport
Name: Fa0/7
Switchport: Enabled
Administrative Mode: trunk
Operational Mode: trunk
Administrative Trunking Encapsulation: dot1q
Operational Trunking Encapsulation: dot1q
Negotiation of Trunking: On
Access Mode VLAN: 1 (default)
Trunking Native Mode VLAN: 1 (default)
Administrative Native VLAN tagging: enabled
Voice VLAN: none
Administrative private-vlan host-association: none
Administrative private-vlan mapping: none
Administrative private-vlan trunk native VLAN: none
Administrative private-vlan trunk Native VLAN tagging: enabled
Administrative private-vlan trunk encapsulation: dot1q
Administrative private-vlan trunk normal VLANs: none
Administrative private-vlan trunk associations: none
Administrative private-vlan trunk mappings: none
Operational private-vlan: none
Trunking VLANs Enabled: ALL
Pruning VLANs Enabled: 2-1001
```

```
ALS1#sh int trunk

Port      Mode        Encapsulation  Status         Native vlan
Fa0/7     auto        802.1q         trunking       1
Fa0/8     auto        802.1q         trunking       1


ALS1#show interface fa 0/7 switchport
Name: Fa0/7
Switchport: Enabled
Administrative Mode: dynamic auto
Operational Mode: trunk
Administrative Trunking Encapsulation: dot1q
Operational Trunking Encapsulation: dot1q
Negotiation of Trunking: On
Access Mode VLAN: 1 (default)
Trunking Native Mode VLAN: 1 (default)
Administrative Native VLAN tagging: enabled
Voice VLAN: none
Administrative private-vlan host-association: none
Administrative private-vlan mapping: none
Administrative private-vlan trunk native VLAN: none
Administrative private-vlan trunk Native VLAN tagging: enabled
Administrative private-vlan trunk encapsulation: dot1q
Administrative private-vlan trunk normal VLANs: none
Administrative private-vlan trunk associations: none
Administrative private-vlan trunk mappings: none
Operational private-vlan: none
Trunking VLANs Enabled: ALL
Pruning VLANs Enabled: 2-1001
```

# DTP Information on the Port

```
DLS1# show dtp interface fa 0/7
DTP information for FastEthernet0/7:
  TOS/TAS/TNS:                              TRUNK/ON/TRUNK
  TOT/TAT/TNT:                              802.1Q/802.1Q/802.1Q
  Neighbor address 1:                       001B53A1A487
  Neighbor address 2:                       000000000000
  Hello timer expiration (sec/state):       20/RUNNING
  Access timer expiration (sec/state):      never/STOPPED
  Negotiation timer expiration (sec/state): never/STOPPED
  Multidrop timer expiration (sec/state):   never/STOPPED
  FSM state:                                S6:TRUNK
  # times multi & trunk                     0
  Enabled:                                  yes
  In STP:                                   no

  Statistics
  ----------
  524 packets received (524 good)
  0 packets dropped
      0 nonegotiate, 0 bad version, 0 domain mismatches,
      0 bad TLVs, 0 bad TAS, 0 bad TAT, 0 bad TOT, 0 other
  839 packets output (839 good)
      524 native, 315 software encap isl, 0 isl hardware native
  0 output errors
  0 trunk timeouts
  1 link ups, last link up on Mon Mar 01 1993, 00:06:49
  0 link downs
```

# Dynamic Desirable vs. Access



```
DLS1(config)#int fa 0/7
DLS1(config-if-range)#switchport mode dynamic desirable
DLS1#sh int trunk

Port      Mode         Encapsulation  Status        Native vlan
Fa0/7     desirable    802.1q         not-trunking  1
Fa0/8     on           802.1q         trunking      1

DLS1#show interface fa 0/7 switchport
Name: Fa0/7
Switchport: Enabled
Administrative Mode: dynamic desirable
Operational Mode: static access
Administrative Trunking Encapsulation: dot1q
Operational Trunking Encapsulation: native
Negotiation of Trunking: On
Access Mode VLAN: 1 (default)
Trunking Native Mode VLAN: 1 (default)
Administrative Native VLAN tagging: enabled
Voice VLAN: none
Administrative private-vlan host-association: none
Administrative private-vlan mapping: none
Administrative private-vlan trunk native VLAN: none
Administrative private-vlan trunk Native VLAN tagging: enabled
Administrative private-vlan trunk encapsulation: dot1q
Administrative private-vlan trunk normal VLANs: none
Administrative private-vlan trunk associations: none
Administrative private-vlan trunk mappings: none
Operational private-vlan: none
Trunking VLANs Enabled: ALL
Pruning VLANs Enabled: 2-1001
```

```
ALS1(config)#int fa 0/7
ALS1(config-if-range)#switchport mode access
ALS1#sh int trunk

Port      Mode         Encapsulation  Status        Native vlan
Fa0/7     off          802.1q         not-trunking  1
Fa0/8     auto         802.1q         trunking      1

ALS1#show interface fa 0/7 switchport
Name: Fa0/7
Switchport: Enabled
Administrative Mode: static access
Operational Mode: static access
Administrative Trunking Encapsulation: dot1q
Operational Trunking Encapsulation: native
Negotiation of Trunking: Off
Access Mode VLAN: 1 (default)
Trunking Native Mode VLAN: 1 (default)
Administrative Native VLAN tagging: enabled
Voice VLAN: none
Administrative private-vlan host-association: none
Administrative private-vlan mapping: none
Administrative private-vlan trunk native VLAN: none
Administrative private-vlan trunk Native VLAN tagging: enabled
Administrative private-vlan trunk encapsulation: dot1q
Administrative private-vlan trunk normal VLANs: none
Administrative private-vlan trunk associations: none
Administrative private-vlan trunk mappings: none
Operational private-vlan: none
Trunking VLANs Enabled: ALL
Pruning VLANs Enabled: 2-1001
```

# Verify the DTP operation

```
Switch# show interface gi 1/1 switchport
Name: Gig1/1
Switchport: Enabled
Administrative Mode: dynamic auto
Operational Mode: trunk
Administrative Trunking Encapsulation: dot1q
Operational Trunking Encapsulation: dot1q
Negotiation of Trunking: On
...
```

*Dynamic auto is set locally*

*Far end is set with dynamic desirable or trunk*

```
Switch# show dtp
Global DTP information
    Sending DTP Hello packets every 30 seconds
    Dynamic Trunk timeout is 300 seconds
    6 interfaces using DTP
```

# The debug dtp Command

```
DLS2# debug dtp ?
  aggregation  Show DTP debug user message aggregation
  all          All DTP debugging messages
  decision     Show DTP debug decision table
  events       DTP events
  oserrs       DTP OS errors
  packets      DTP packet processing
  queue        Show DTP debug packet queueing
  states       DTP state transitions
  timers       DTP timer events
```

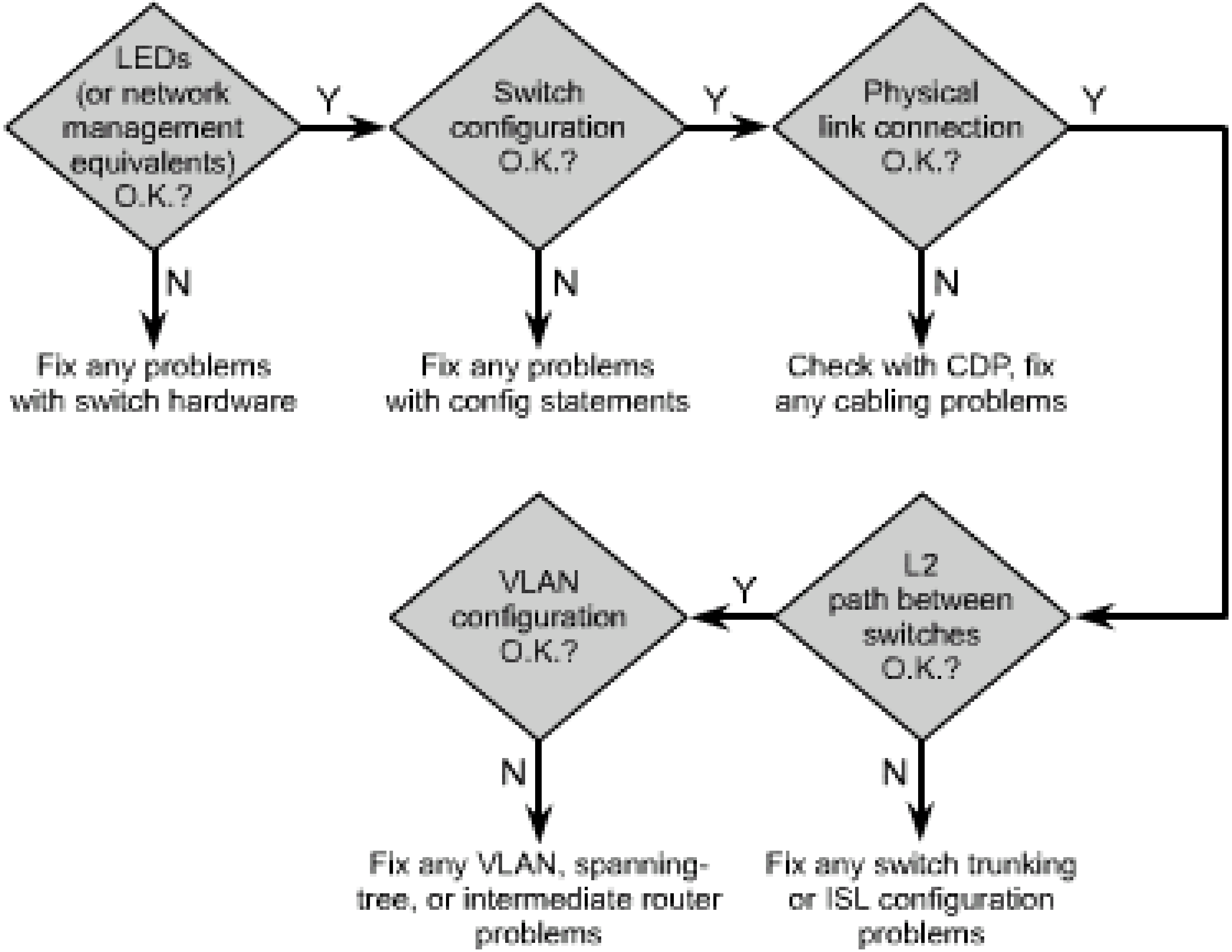
```
*Mar  1 01:21:41.505: DTP-event:Fa0/11:Received packet event
../dyntrk/dyntrk_process.c:2200
*Mar  1 01:21:41.513: DTP-event:Fa0/12:Received packet event
../dyntrk/dyntrk_process.c:2200
*Mar  1 01:21:41.539: DTP-event:Fa0/7:Received packet event
../dyntrk/dyntrk_process.c:2200
*Mar  1 01:21:41.555: DTP-event:Fa0/8:Received packet event
../dyntrk/dyntrk_process.c:2200
*Mar  1 01:21:41.622: DTP-event:Fa0/10:Received packet event
../dyntrk/dyntrk_process.c:2200
*Mar  1 01:21:41.715: DTP-event:Fa0/9:Received packet event
../dyntrk/dyntrk_process.c:2200
```

# Recommendations

- DTP can be used for initial configuration
  - After the initial config, DTP should be deactivated
  - Configure the port as trunk or access on both switches
  - Disable negotiation using **`switchport nonegotiate`**
- DTP sends a VTP domain name in the frames
  - Successfully negotiated trunk needs to have same domain name on both ends
  - `IF` the domain names are different `THEN` DTP fails to negotiate the trunk mode
- *Disabling DTP negotiation can save seconds of outage when restoring a failed link or node!*

# Troubleshooting L2 network

# Example of a Troubleshooting Process

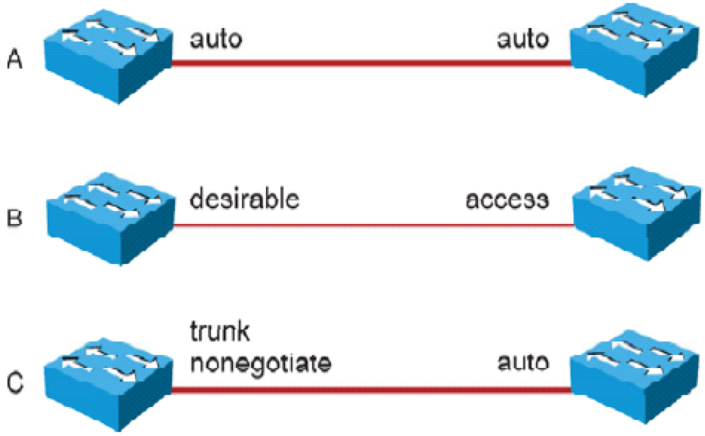

LEDs (or network management equivalents) O.K.?
Y
N
Fix any problems with switch hardware
Switch configuration O.K.?
Y
N
Fix any problems with config statements
Physical link connection O.K.?
Y
N
Check with CDP, fix any cabling problems
L2 path between switches O.K.?
Y
N
Fix any switch trunking or ISL configuration problems
VLAN configuration O.K.?
N
Fix any VLAN, spanning-tree, or intermediate router problems

# Common Trunk Link Problems

- **Native VLAN mismatch**

```
%CDP-4-NATIVE_VLAN_MISMATCH: Native VLAN mismatch discovered on
   GigabitEthernet1/1 (99), with Switch GigabitEthernet1/1 (1).
%CDP-4-NATIVE_VLAN_MISMATCH: Native VLAN mismatch discovered on
   GigabitEthernet1/1 (99), with Switch GigabitEthernet1/1 (1).
```

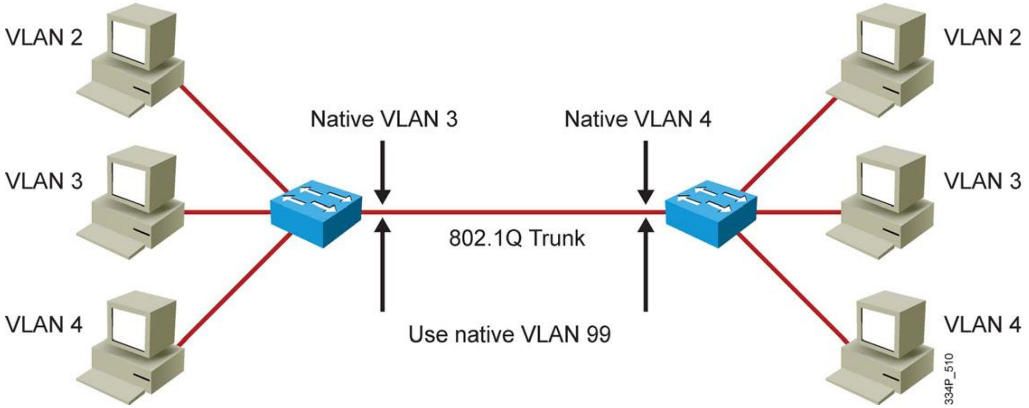
- **Creation of a trunk fails**
  - DTP is enabled on one interface. The far end has static trunk configuration and disabled DTP
  - VTP domain mismatch
  - Encapsulation mismatch



- **Misconfiguration of allowed VLANs on the trunk link**
  - VLAN is not allowed on all trunks

# Common Native VLAN Problems

- Native VLAN
  - Native VLAN frames are carried over the trunk link untagged
  - Native VLAN must be the same on both ends of a trunk
  - VLAN1 is used as the native VLAN in the default configuration
- Potential problems
  - Creating L2 loop
  - Merging traffic between VLANs
  - STP/CDP/DTP use VLAN1 – if the native VLAN is changed



- Cisco switches detect native VLAN mismatch using CDP/STP and disable the port until the problem is resolved

# VLAN Trunking Protocol

# VLAN Trunking Protocol (VTP)

- Cisco proprietary protocol for VLAN management
  - Manages VLAN database on all switches
- VTP updates are exchanged only across trunk links
- Three versions
  - VTPv1 and VTPv2 are usually supported
  - VTPv3 originally only on high-end switches
    - Since IOS 12.2(52)SE supported on all Catalyst switches
  - VTPv1, VTPv2 will advertise VLANs 1–1005 only
  - VTPv3 advertises information about all VLANs
- Cisco Document ID: 10558, „Understanding VLAN Trunk Protocol (VTP)“
- Catalyst 6500 Series Software Configuration Guide, „Understanding How VTP Version 3 Works VTP Version 3“

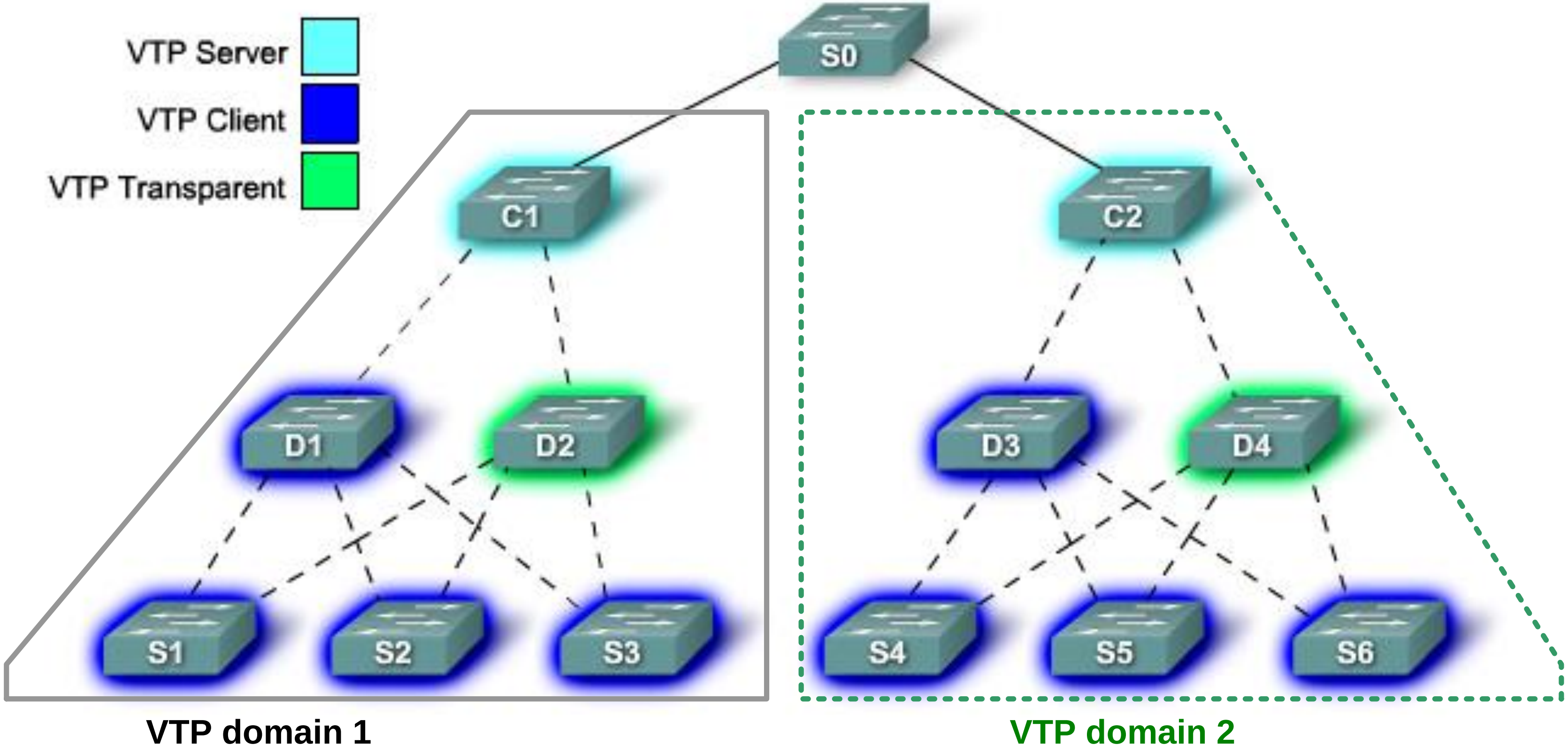
# Differences Between VTP Versions

- VTPv2 compared to VTPv1:
  - Support for Token Ring VLANs
  - Support for unknown TLV in VTP messages (TLV are forwarded; VTPv1 discards unknown TLV values)
  - Version independent transparent mode: VTPv2 Transparent switch forwards VTP messages without checking the version
  - Support for consistency check – when new information about the VTP domain is entered through command line/SNMP
- VTPv3:
  - Support for extended-range VLANs, Private VLANs
  - Improved server authentication
  - Configurable on a per-port basis
  - Protection from the “wrong” database accidentally being inserted (primary server)
  - General protocol – possibility to distribute any database (MSTP)

# VTP Modes

- **Server**
  - Creates, deletes and modifies VLANs
  - Sends and forwards advertisements to other switches
  - VLAN information are stored in `vlan.dat`
- **Client**
  - Cannot create, change, delete VLAN on CLI
  - Synchronizes VLAN configuration with latest information received from other switches in `vlan.dat`
  - Sends and forwards advertisements to other switches
- **Transparent**
  - Not the “real” member of VTP domain
  - Forwards advertisements to other switches
  - Does not synchronize its VLAN configuration
  - Store information to the startup-config
  - VTP revision number is always 0
- **Off**
  - DOES NOT forward messages, ignores them (only VTPv3 or CatOS)

# VTP Domain



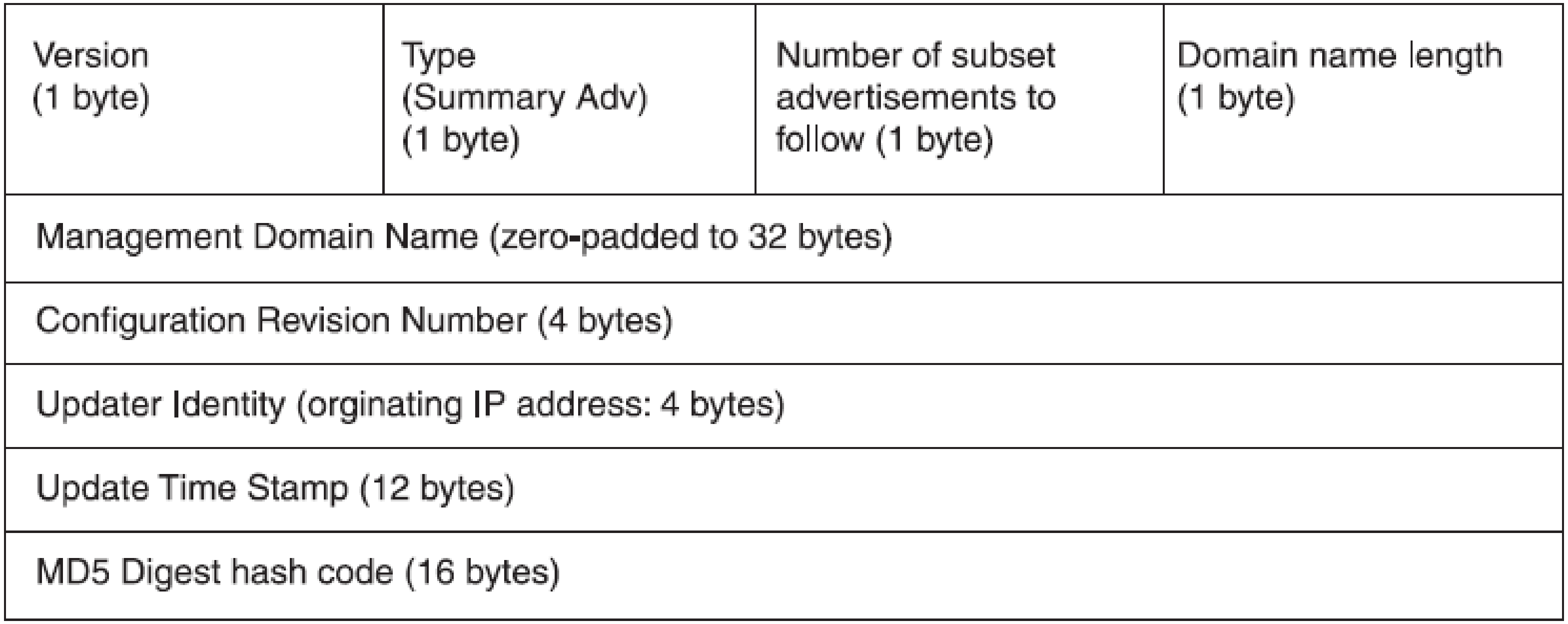
- Several interconnected switches sharing the same VTP environment
- Identified by same domain name
- Catalyst switches support only a single VTP domain – borders between domains are on the links

# Messages Types

- **Summary advertisement**
  - Information about VLAN domain name and revision number, sent in 5 minute increments
  - Includes VTP version, domain name, revision number, number of subset advertisement to follow
- **Subset advertisement**
  - Follows the Summary advertisements when server changes the database
  - Sends the list of VLAN information
- **Advertisement requests**
  - Request for VLAN database information if the Summary advertisement with a higher configuration number is received
  - Summary and subset advertisements are sent as a response
- **VTP Join**
  - Used for VTP Pruning

# VTP Summary Advertisement

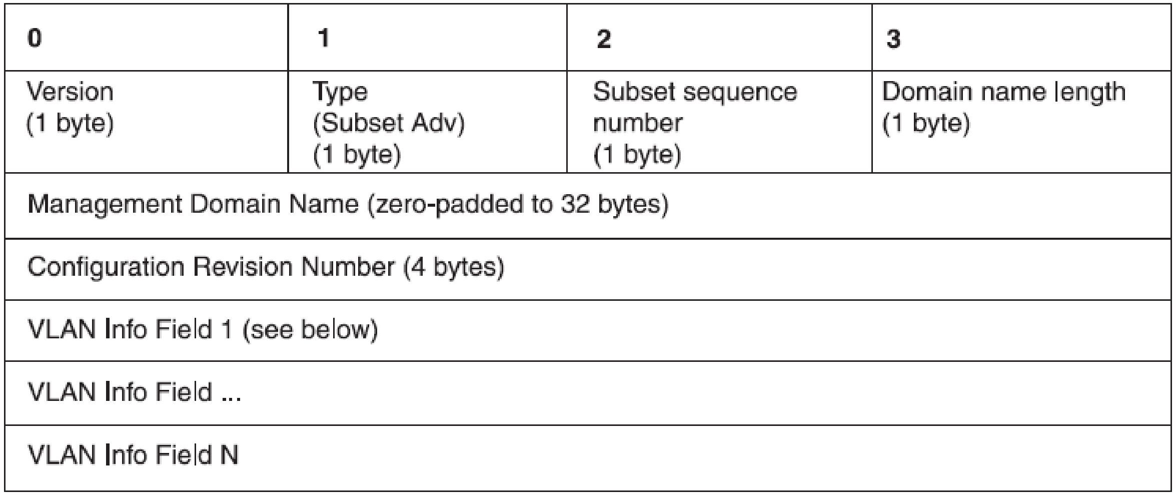
- Multicast address 01-00-0C-CC-CC-CC  (All-VTP)



| | | | |
|---|---|---|---|
| Version<br>(1 byte) | Type<br>(Summary Adv)<br>(1 byte) | Number of subset advertisements to follow (1 byte) | Domain name length<br>(1 byte) |
| Management Domain Name (zero-padded to 32 bytes) | | | |
| Configuration Revision Number (4 bytes) | | | |
| Updater Identity (orginating IP address: 4 bytes) | | | |
| Update Time Stamp (12 bytes) | | | |
| MD5 Digest hash code (16 bytes) | | | |

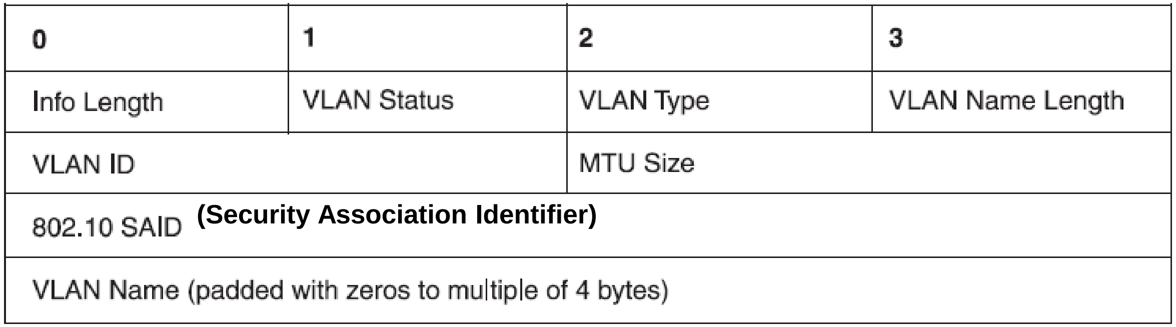
# VTP Subset Advertisement

- Information about VLAN DB changes



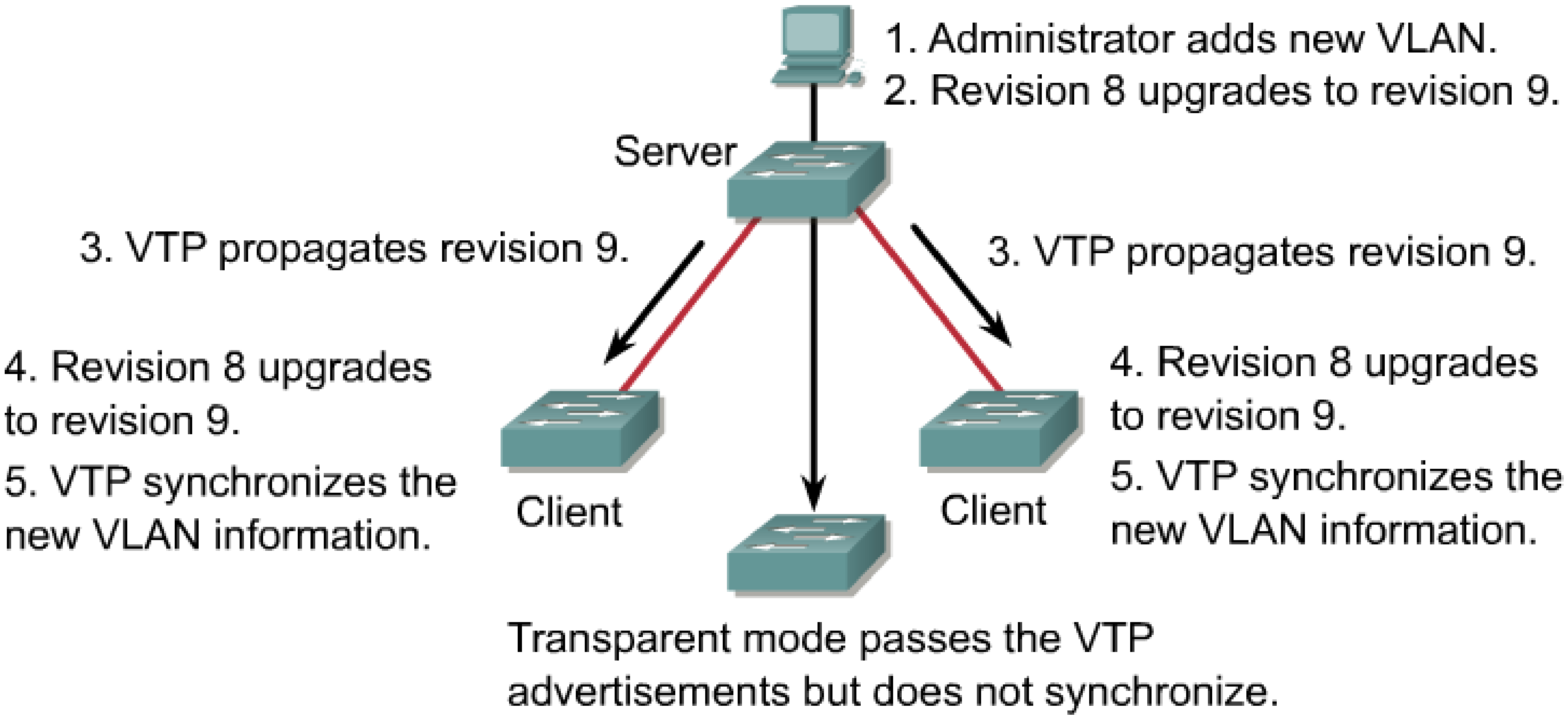
| 0 | 1 | 2 | 3 |
|---|---|---|---|
| Version<br>(1 byte) | Type<br>(Subset Adv)<br>(1 byte) | Subset sequence number<br>(1 byte) | Domain name length<br>(1 byte) |
| Management Domain Name (zero-padded to 32 bytes) | | | |
| Configuration Revision Number (4 bytes) | | | |
| VLAN Info Field 1 (see below) | | | |
| VLAN Info Field ... | | | |
| VLAN Info Field N | | | |

- VLAN Info Field

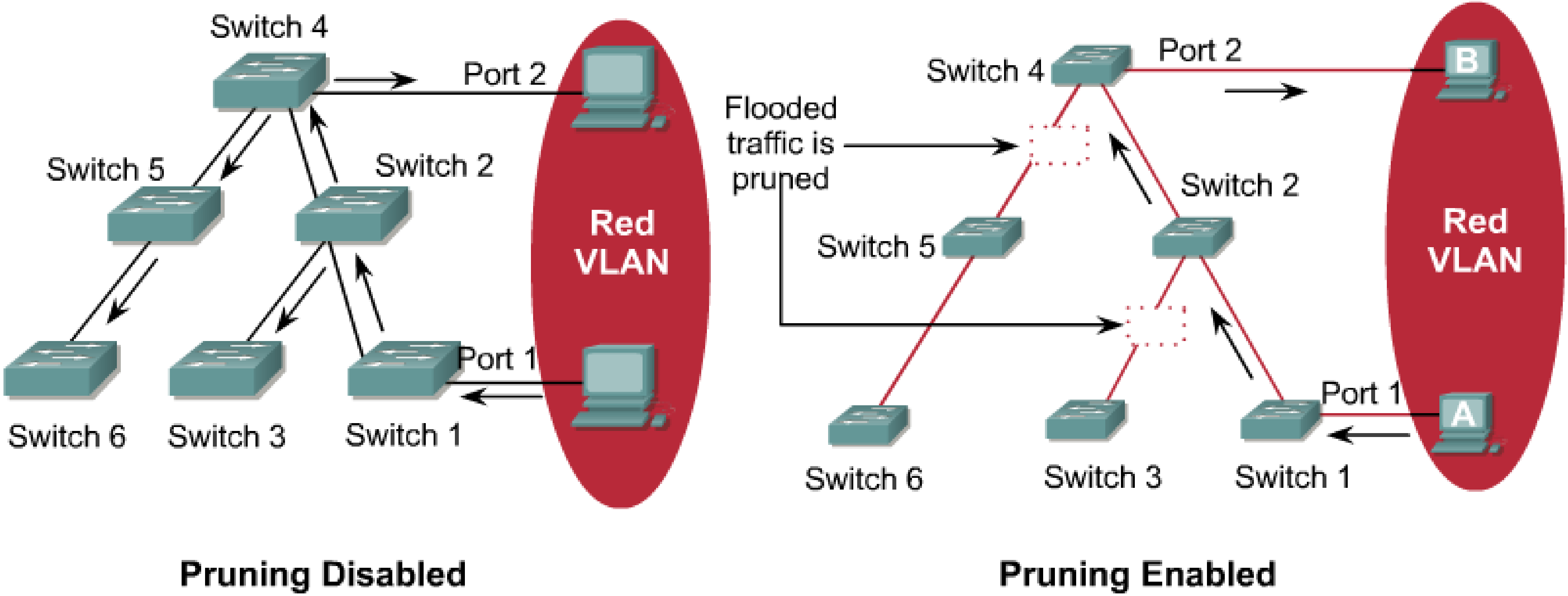


| 0 | 1 | 2 | 3 |
|---|---|---|---|
| Info Length | VLAN Status | VLAN Type | VLAN Name Length |
| VLAN ID | | MTU Size | |
| 802.10 SAID (Security Association Identifier) | | | |
| VLAN Name (padded with zeros to multiple of 4 bytes) | | | |

# Example


1. Administrator adds new VLAN.
2. Revision 8 upgrades to revision 9.
Server
3. VTP propagates revision 9.
3. VTP propagates revision 9.
4. Revision 8 upgrades to revision 9.
5. VTP synchronizes the new VLAN information.
Client
Client
4. Revision 8 upgrades to revision 9.
5. VTP synchronizes the new VLAN information.
Transparent mode passes the VTP advertisements but does not synchronize.

# VTP pruning



- Determines when a trunk connection is flooding traffic needlessly
- Reduce the broadcast scope
- Configured on VTP server
- *If something seems odd with VLAN communication then disable VTP pruning!*

# Basic Configuration

1. Choose VTP version
2. Choose domain
   - Establish boundaries
   - Name: Case sensitive
3. Choose mode for the switch
   - Recommended is one (two) server per domain, the rest should be clients
4. Secure the domain with password
   - Beaware of white space
   - Only HMAC is sent in VTP messages
5. Optionaly turn on VTP Pruning

# Configuring VTP

```
Switch(config)# vtp domain DOMAIN_NAME
Switch(config)# vtp mode { client | server | transparent }
! Hidden is supported in v3
Switch(config)# vtp password PASSWORD [ hidden ]

! Default mode is VTP v2 capable
Switch(config)# vtp version { 1 | 2 | 3 }

! VTP server can enable pruning
Switch(config)# vtp pruning
```

# Verify VTP Configuration

```
Switch# show vtp status
VTP Version                         : 2
Configuration Revision              : 0
Maximum VLANs supported locally     : 64
Number of existing VLANs            : 5
VTP Operating Mode                  : Server
VTP Domain Name                     : Null
VTP Pruning Mode                    : Disabled
VTP V2 Mode                         : Disabled
VTP Traps Generation                : Disabled
MD5 digest                          : 0x7D 0x5A 0xA6 0x0E 0x9A
   0x72 0xA0 0x3A
Configuration last modified by 0.0.0.0 at 0-0-00 00:00:00
Local updater ID is 0.0.0.0 (no valid interface found)
```

- Default Domain name is Null
- Switch will change it if a new domain name is received – only during initial configuration

# Displaying VTP statistics

```
Switch# show vtp counters
VTP statistics:
Summary advertisements received    : 1
Subset advertisements received     : 1
Request advertisements received    : 2
Summary advertisements transmitted : 5
Subset advertisements transmitted  : 5
Request advertisements transmitted : 0
Number of config revision errors   : 0
Number of config digest errors     : 0
Number of V1 summary errors        : 0


VTP pruning statistics:

Trunk            Join Transmitted Join Received    Summary advts received from
                                                   non-pruning-capable device
---------------- ---------------- ---------------- ---------------------------
```

# VTPv3

- VTPv3 has to be activated manually
- Switch can independently act as a server/client for different databases
  - VLAN
  - MST
  - Unknown (placeholder for future database)
- Two server modes
  - **Primary server**: can modified a database, only one in a domain
  - **Secondary server**: can become primary server
  - Primary server is not configured in config mode
    - Administrator request the function in privileged mode
    - Change to primary server mode can be protected with password
      - Password can be hidden

# VTPv3 Configuration

```
Switch(config)# vtp version 3
Switch(config)# vtp domain DOMAIN_NAME
Switch(config)# vtp mode { client | server | transparent | off }
                          [ vlan | mst | unknown   ]
Switch(config)# vtp password PASSWORD hidden
```

```
Switch(config)# vtp version 3
Switch(config)# vtp domain DOMAIN_NAME
Switch(config)# vtp mode server vlan
Switch(config)# vtp mode client mst
Switch(config)# vtp password PASSWORD hidden
Switch(config)# end
Switch# vtp primary mst
System can become primary server for Mst feature only when
configured as a server

Switch# vtp primary vlan
This system is becoming primary server for feature vlan
Enter VTP Password:
No conflicting VTP3 devices found.
Do you want to continue? [confirm]
Switch#
```

# Common VTP Problems

- Problems
  - Misconfigured trunk link
  - Incompatible VTP versions
  - Different VTP domain names
  - VTP password mismatch
  - Missing VTP server
- *Always check the VTP revision number when adding the device into the VTP domain!*
  - VTP switch can rewrite VTP domain database with its own
  - VTP revision number can be zeroed by changing the switch to transparent mode or by changing the domain name
  - VTP revision number is not zeroed by restarting the device – it is saved in `vlan.dat`

# Common VTP Pruning Problems

- VTP Pruning relies on receiving VTP Join messages on trunk link
  - Sender of VTP Join messages sends active VLANs
  - `IF` the messages are lost or ignored `THEN` all VLANs are blocked (there is no information about active VLAN)
- VTP Join could be ignored `IF`:
  - Domain names mismatch
  - VTP versions mismatch
  - Trunk encapsulation mismatch
- VTP Join could be absent `IF`:
  - VTP pruning configuration mismatch (different passwords, domains)
  - Device does not support VTP protocol (router, server)

# VTP Pruning

- VTP can be configured not to prune a VLAN manually:

```
Switch# configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.
Switch(config)# int fa 0/1
Switch(config-if)# switchport trunk pruning vlan ?
  WORD     VLAN IDs of the allowed VLANs when this port is in trunking mode
  add      add VLANs to the current list
  except   all VLANs except the following
  none     no VLANs
  remove   remove VLANs from the current list
Switch(config-if)# switchport trunk pruning vlan none
Switch(config-if)# end
```

# Debuging VTP Commands

```
DLS1# debug sw-vlan vtp ?
  events        vtp events
  packets       vtp packets
  pruning       vtp pruning events
  redundancy    vtp redundancy
  xmit          vtp packets transmitted
```

```
DLS1(config)# debug sw-vlan vtp events
DLS1(config)# vlan 10
DLS1(config-vlan)# exit
DLS1#
*Mar  1 05:30:08.908: VTP LOG RUNTIME: Transmit vtp summary, domain netlab,
   rev 6, followers 1, tlv blk size 5 (inc #tlv field),
   MD5 digest calculated = 99 45 75 AA D3 0B 5A C4 9F 25 E1 FE BC 4E 39 59

*Mar  1 05:30:08.925: VTP LOG RUNTIME: Summary packet received, domain =
   netlab, rev = 6, followers = 1, length 77, trunk Fa0/7

*Mar  1 05:30:08.925: VTP LOG RUNTIME: Summary packet rev 6 equal to domain
   netlab rev 6

*Mar  1 05:30:08.925: VTP LOG RUNTIME: Subset packet received, domain =
   netlab, rev = 6, seq = 1, length = 224
```

# Domain and Password Mismatch

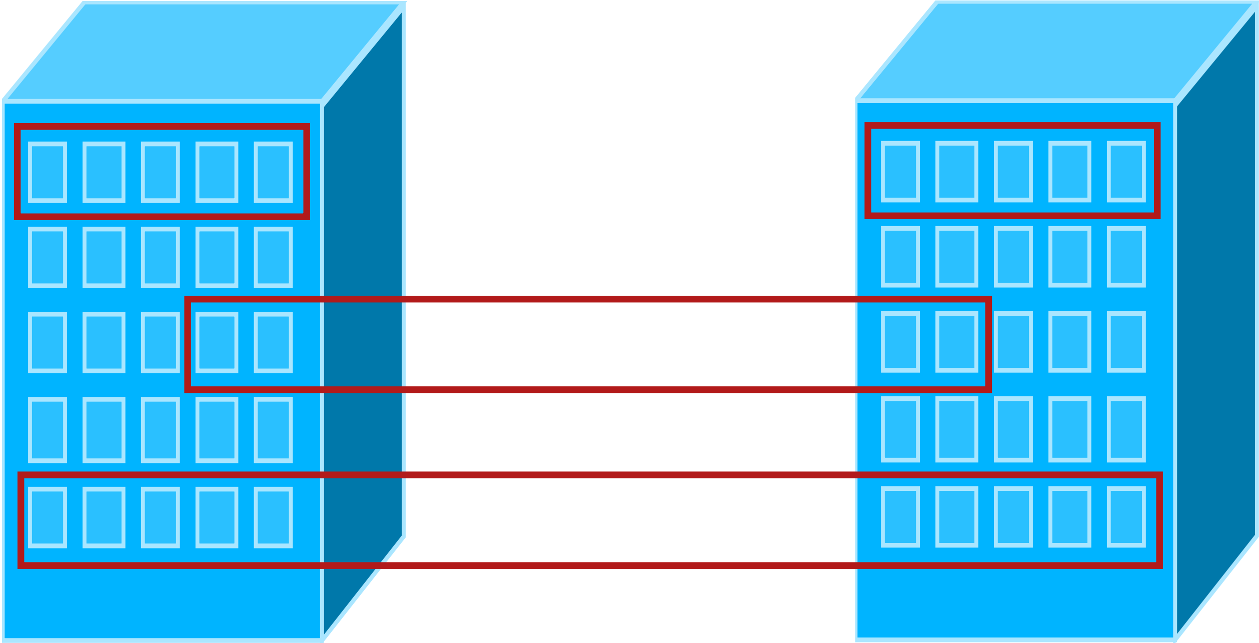
```
*Mar  1 00:36:39.828: %SW_VLAN-6-VTP_DOMAIN_NAME_CHG: VTP domain name changed
to swlab.

*Mar  1 00:36:40.960: %DTP-5-DOMAINMISMATCH: Unable to perform trunk
negotiation on port Fa0/11 because of VTP domain mismatch.

*Mar  1 00:36:40.969: %DTP-5-DOMAINMISMATCH: Unable to perform trunk
negotiation on port Fa0/12 because of VTP domain mismatch.
```

```
*Mar  1 00:51:04.912: VTP LOG RUNTIME: Summary packet received, domain = swlab, rev = 2,
followers = 1, length 77, trunk Fa0/9

*Mar  1 00:51:04.912: VTP LOG RUNTIME: Summary packet rev 2 greater than domain swlab rev 1

*Mar  1 00:51:04.912: VTP LOG RUNTIME: Domain swlab currently not in updating state

*Mar  1 00:51:04.912: VTP LOG RUNTIME: pdu len 77, #tlvs 1

*Mar  1 00:51:04.912: VTP LOG RUNTIME: Subset packet received, domain = swlab, rev = 2, seq
= 1, length = 280

*Mar  1 00:51:04.912: VTP LOG RUNTIME: MD5 digest failing
calculated = 16 98 BB 99 5F 15 60 04 11 73 1D B3 17 A3 8D 8B
transmitted = 37 76 F2 00 3B 16 04 91 5C 1A F0 ED 79 90 7C DD
```

# Private VLAN

# Motivation

- Two apartment blocks
- All apartments have access to Internet
- Some apartments need to have mutual connectivity.



- Rest of apartments should be isolated
- *Solutions that saves IPv4 address space?*

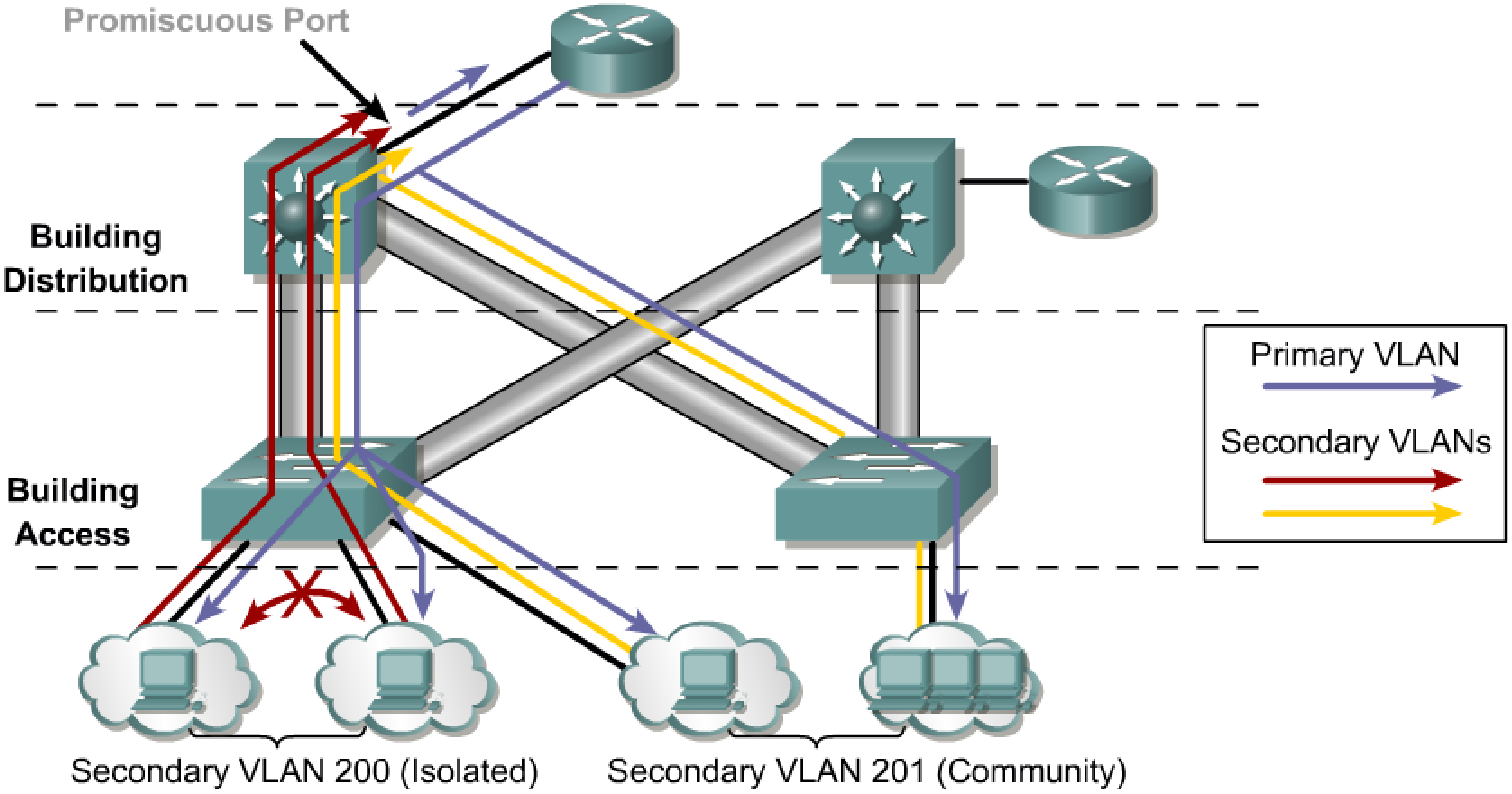
# Private VLAN ①

- Private VLANs (RFC 5517) enable the isolation at L2 of devices in the same IP subnet
- Private VLANs “separate” one VLAN to several groups
  - Original VLAN = **primary VLAN**
  - Every subgroup is represented by **secondary VLAN**
- There are two types of secondary VLANs
  - **Community:** Ports belonging to one community VLAN can communicate with other ports in the same community
  - **Isolated:** Devices belonging to isolated VLAN CAN NOT communicate between each other
- It is internal structure only. From the outside view, there is only one VLAN (primary) and one IP network
  - The main design goal for PVLAN is to save IP address space + provide an isolation
- “Configuring Private VLANs”

# Private VLAN ②

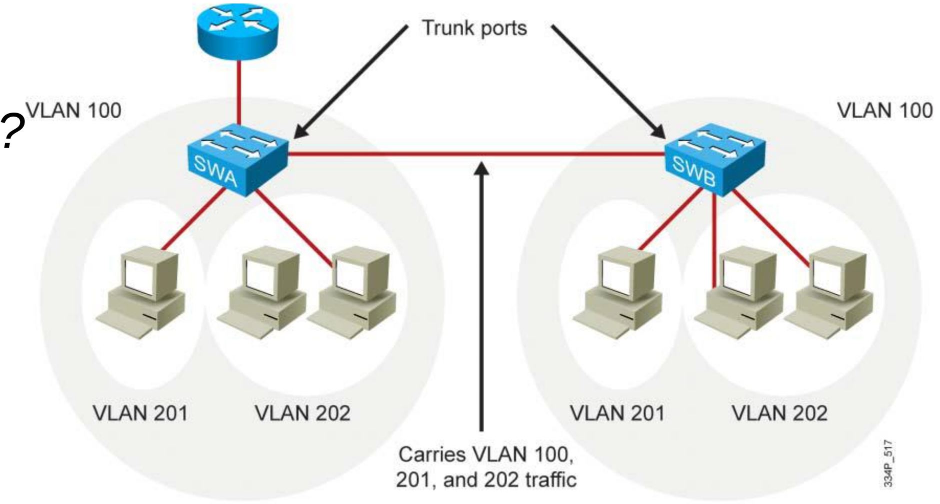
- Private VLAN can consist of:
  - several community VLANs
  - one isolated VLAN
- An entrance/exit point is necessary
  - **Promiscuous port**: port can communicate with all ports within the Private VLAN, including community and isolated ports
- Communication in private VLAN
  - Port in a community VLAN can communicate with other ports in the community, trunk ports and promiscuous port
  - Port in isolated VLAN can communicate only with trunk ports and promiscuous port

# Use-Case

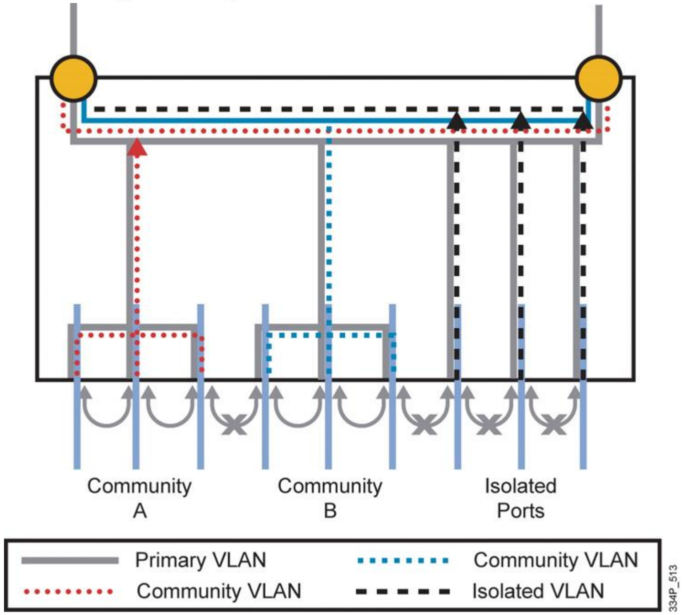

Promiscuous Port
Building Distribution
Building Access
Primary VLAN
Secondary VLANs
Secondary VLAN 200 (Isolated)
Secondary VLAN 201 (Community)

# Frame Tagging

- *How are the frames tagged on trunks?*
  - `IF` the frame is received on the community or isolated VLAN port `THEN` tag of appropriate secondary VLAN is added on a trunk port
  - `IF` the frame is received on the promiscuous port `THEN` tag of a primary VLAN is added on the trunk port
- Frame tagged with primary VLAN tag can flow to:
  - other trunks ports
  - promiscuous ports
  - all associated secondary VLANs ports
- Frame tagged with the secondary VLAN tag can flow to:
  - other trunk ports
  - promiscuous ports
  - appropriate secondary VLAN ports (Community = to any other ports in the same community VLAN; Isolated: to none of other ports)



VLAN 100 = Primary VLAN
VLAN 201 = Secondary isolated VLAN
VLAN 202 = Secondary community VLAN



# PVLAN Trunk Ports

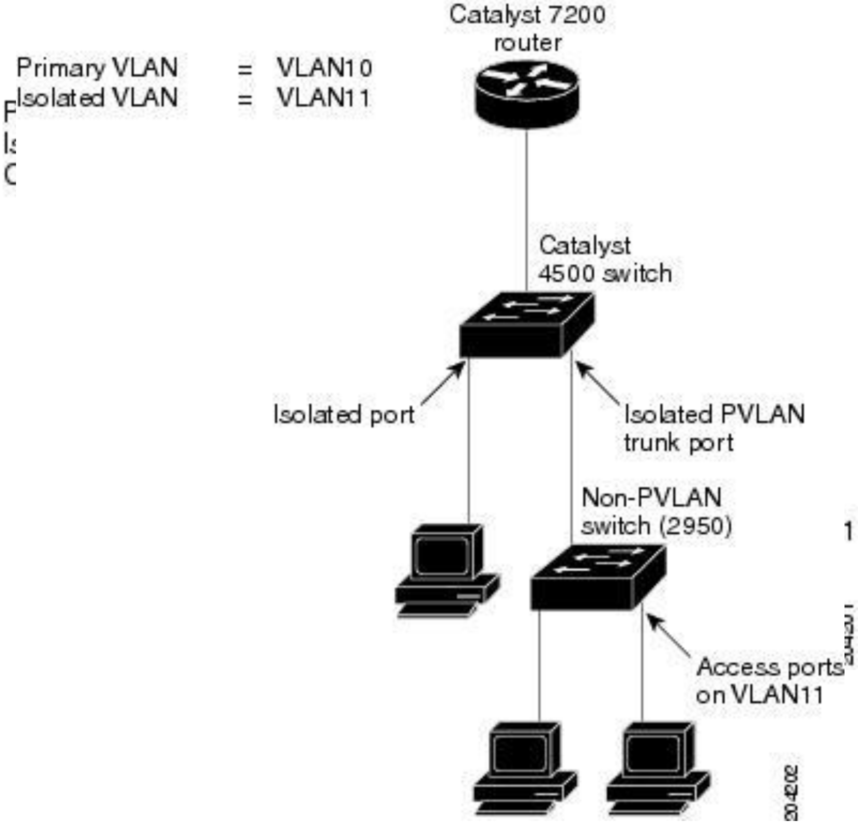
- Not widely supported (Catalyst 4500)
- Ordinary trunk ports are usually sufficient
- Sometimes is necessary to use more appropriate trunk port:

## Promiscuous PVLAN trunk port

- Tag of secondary VLAN of an outgoing frame is changed to Primary VLAN tag
- Use case: Trunk is connected to upstream device which does not support PVLAN, but needs tagging

## Isolated PVLAN trunk port

- Tag of primary VLAN of an outgoing frame is changed to secondary VLAN tag of isolated VLAN
- Use case:
  - Can connect a downstream switch that does not support PVLANs thus extend the isolated VLAN scope. Downstream switch without PVLAN support can use **`switchport protected`** command to separate connected clients



# Configuration

- Mark VLAN as primary

```
Switch(config-vlan)# private-vlan primary
```

- Mark VLAN as isolated or community

```
Switch(config-vlan)# private-vlan {community|isolated}
```

- Associate secondary VLAN(s) with primary

```
! On primary VLAN
Switch(config-vlan)#
   private-vlan association PriVID ListOfSecVIDs
```

- Setup interface as promiscuous

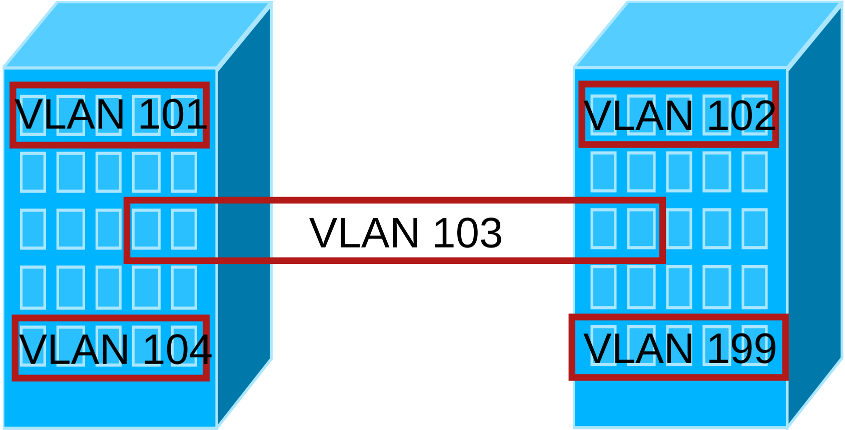
```
! Router on a stick
Switch(config-if)# private-vlan mapping PriVID ListOfSecVID
! L3 switch SVI
Switch(config-if)# private-vlan mapping ListOfSecVIDs
```

- Setup interface as access for user in secondary VLAN

```
Switch(config-if)# switchport mode private-vlan host
Switch(config-if)# sw private-vlan host-association PriVID SecVID
```

# Configuration Example

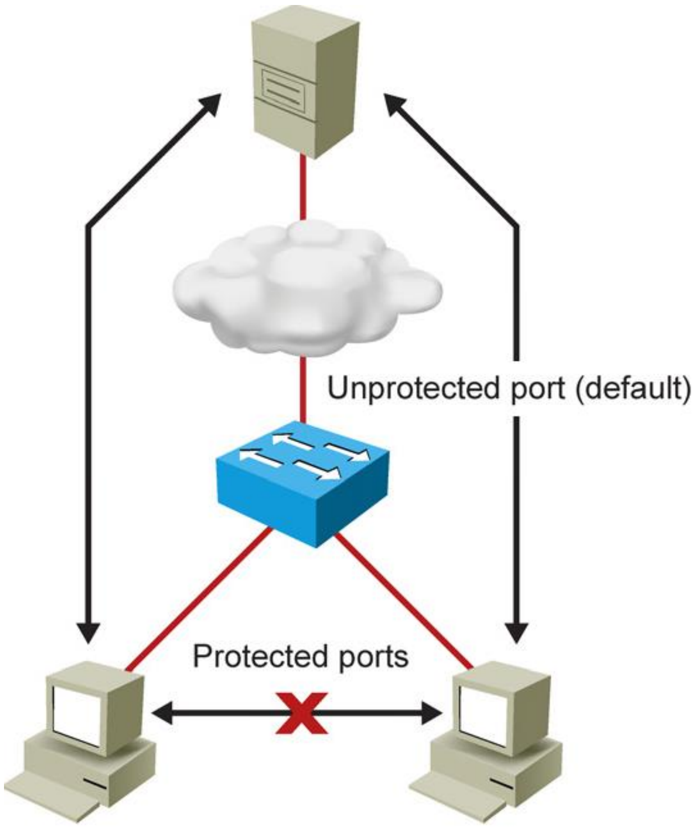
```
vtp transparent !Only for VTPv1/2
vlan 199
 private-vlan isolated

vlan 101-104
 private-vlan community

vlan 100
 private-vlan primary
 private-vlan association 101-104
 private-vlan association add 199
```



```
!Community port
interface fa0/1
 switchport mode private-vlan host
 switchport private-vlan
   host-association 100 101

!Isolated port
interface fa0/2
 switchport mode private-vlan host
 switchport private-vlan
   host-association 100 199


!In case of router on a stick
!Promisc port
interface fa0/3
 switchport mode private-vlan prom
 switchport private-vlan
  mapping 100 101-104,199


!In case of L3 switch
!Promisc SVI
interface Vlan100
 private-vlan mapping 101-104,199
```

# Protected Ports on Access Switch

- VTPv1 and VTPv2 do not support private VLAN
  - Recommendation is to switch VTP mode to transparent
  - VTPv3 supports private VLAN
- *Private VLANs are supported only on multilayer switches Catalyst 3560 and higher! What to do on L2 switch?*
  - 2950/2960/3550 can configure protected ports using the `switchport protected` command
    - Protected ports on one switch cannot communicate with each other – similar as a member of isolated VLAN
  - Protected ports cannot be on separate switches – Isolated PVLAN trunk is needed
  - Protected ports are also known as Private VLAN Edge



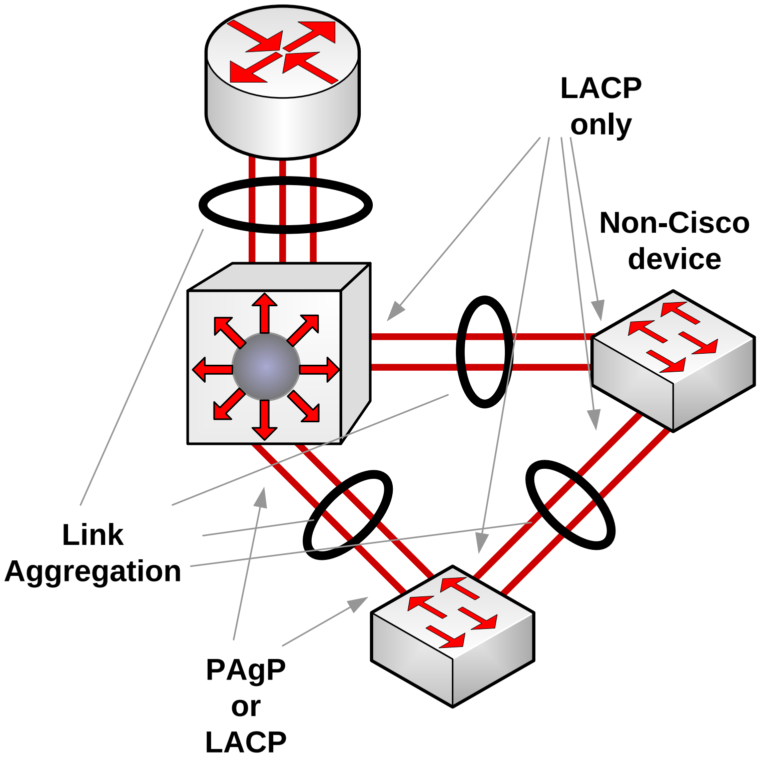
# EtherChannel

# Link Aggregation with EtherChannel

- LAN switch-to-switch technique of grouping several Fast or Gigabit Ethernet ports into one logical channel
- *Advantages*
  - Increase the throughput
  - Provides redundancy
    - As long as at least one physical link is present, the EtherChannel is functional
  - Several methods of load-balancing
    - MAC, IP, IP+TCP/UDP
  - Simplify the configuration and ensure the configuration consistency
    - Only the logical port is configured, physical ports inherit the configuration
  - Simplify the function of some protocols
    - STP works with one logical channel (instead of several physical links)

# Link Aggregation

- EtherChannel uses one of the two management protocols to create Port-Channel
  - Verifies that all ports have the same type of configuration
  - Verifies that all links are connected to the same device
  - Verifies that far-end also uses EtherChannel
- **PAgP (Port Aggregation Protocol)**
  - Cisco proprietary, U.S. Patent 6163543
- **LACP (Link Aggregation Protocol)**
  - IEEE standard 802.3ad



# EtherChannel PAgP and LACP modes

| PAgP | LACP |
|---|---|
| **Auto**<br>The interface responds to the PAgP packets that it receives but does not initiate PAgP negotiation (default) | **Passive**<br>Same as PAgP Auto |
| **Desirable**<br>The interface initiates negotiations with other interfaces by sending PAgP packets | **Active**<br>Same as PAgP Desirable |
| **On**<br>Forces the interface to channel without PAgP | **On**<br>Same as PAgP On |

# Prerequisites

- Aggregated ports should/must share similar characteristics
    - Same speed and duplex mode
    - Same mode (access/trunk/dynamic)
        - If the ports operate in access mode, same access VLAN
        - Same trunk protocol (ISL/802.1Q), allowed VLAN and same pruning-eligible list
    - STP cost, priority and mode (edge/non-edge) should be the same
    - Port security on ports and EC port should be disabled

# Configuration of Etherchannel

- Choose the protocol (not necessary)

```
Switch(config)# channel-protocol { pagp | lacp }
```

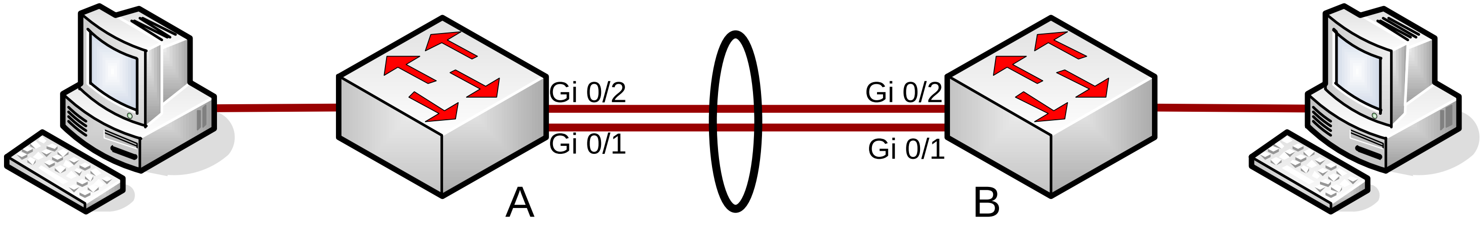
- Assign physical ports into the channel with channel number and mode

```
Switch(config-if)# channel-group GROUP mode {MODE}
```

- Configure logical EtherChannel port

```
Switch(config)# interface port-channel GROUP
```

# Example: PAgP L2 Etherchannel



```
B(config)# int range gi 0/1 - 2
B(config-if-range)# channel-group 1 mode desirable
Creating a port-channel interface Port-channel 1
B(config-if-range)# exit
B(config)# int port-channel 1
B(config-if)# switchport mode trunk
B(config-if)# end
```

```
A(config)# int ra gi 0/1 - 2
A(config-if-range)# channel-group 1 mode desirable
A(config-if-range)#end
```

# Verify the Configuration

```
show etherchannel summary

show etherchannel [ GROUP ] port-channel

show etherchannel detail

show interface [ IFACE GROUP ] etherchannel

show etherchannel
```

# The show etherchannel summary Command

```
A# show etherchannel summary
Flags:  D - down        P - in port-channel
        I - stand-alone s - suspended
        H - Hot-standby (LACP only)
        R - Layer3      S - Layer2
        U - in use      f - failed to allocate aggregator
        u - unsuitable for bundling
        w - waiting to be aggregated
        d - default port


Number of channel-groups in use: 1
Number of aggregators:           1

Group  Port-channel  Protocol    Ports
------+-------------+-----------+-----------------------------------------------
1      Po1(SU)         PAgP      Gi0/1(P)    Gi0/2(P)

A#
```

```
B# show etherchannel summary
...
Group  Port-channel  Protocol    Ports
------+-------------+-----------+-------------
1   Po1(SU)         PAgP      Gi0/1(I)   Gi0/2(I)

B#
```

# The show interface trunk Command

```
A# show int trunk
Port        Mode         Encapsulation  Status        Native vlan
Po1         auto         802.1q         trunking      1

Port        Vlans allowed on trunk
Po1         1-4094

Port        Vlans allowed and active in management domain
Po1         1

Port        Vlans in spanning tree forwarding state and not pruned
Po1         1
A#
```

```
B# show int trunk

Port        Mode         Encapsulation  Status        Native vlan
Po1         on           802.1q         trunking      1

Port        Vlans allowed on trunk
Po1         1-4094

Port        Vlans allowed and active in management domain
Po1         1

Port        Vlans in spanning tree forwarding state and not pruned
Po1         1
B #
```

# The show int etherchannel Command

```
B #sh interface etherchannel
----
Giga 0/1:
Port state    = Up Mstr In-Bndl
Channel group = 1          Mode = Desirable-Sl    Gcchange = 0
Port-channel  = Po1        GC    = 0x00010001      Pseudo port-channel = Po1
Port index    = 0          Load = 0x00             Protocol =   PAgP

Flags:  S - Device is sending Slow hello.  C - Device is in Consistent state.
        A - Device is in Auto mode.        P - Device learns on physical port.
        d - PAgP is down.
Timers: H - Hello timer is running.        Q - Quit timer is running.
        S - Switching timer is running.    I - Interface timer is running.

Local information:
                                Hello    Partner  PAgP     Learning  Group
Port      Flags State   Timers  Interval Count   Priority   Method  Ifindex
Gi0/1     SC    U6/S7   H       30s      1        128        Any      5001

Partner's information:

          Partner              Partner          Partner         Partner Group
Port      Name                 Device ID        Port       Age  Flags   Cap.
Gi0/1     A                    0017.9446.ad00   Gi0/1      26s  SC     10001

Age of the port in the current state: 0d:00h:06m:33s
...
```

# Disbanding the EtherChannel Link

```
B(config)# no int port-channel 1
B(config)# int range gi 0/1-2
B(config-if-range)# no channel-group 1 mode
B(config-if-range)# no shut
```

```
A(config)# no int port-channel 1
A(config)# int range gi 0/1-2
A(config-if-range)# no channel-group 1 mode
A(config-if-range)# no shut
```

# Recommendation for EC configuration

- `IF` the EC is configured using **`on`** mode `THEN` it is necessary to shutdown aggregated ports first before creating Etherchannel
  - Otherwise there is a danger of creating a loop!
  - Ports are in shutdown state when disabling the EC
  - Avoid using mode **`on`** unless it is necessary
- Removing Port-channel interface removes also configuration from belonging ports
- Recommended is to use LACP instead of PAgP
  - Except for Virtual Switching System (VSS)
- EtherChannel CAN NOT be configured as SPAN ports

# Load-balancing Configuration

- Load-balancing can be based on these variables according to platform
  - src-mac: Source MAC (by default)
  - dst-mac: Destination MAC
  - src-dst-mac: Source XOR destination MAC
  - src-ip: Source IP
  - dst-ip: Destination IP
  - src-dst-ip: Source XOR destination IP
  - src-port: Source TCP/UDP port
  - dst-port: Destination TCP/UDP port
  - src-dst-port: Source XOR destination TCP/UDP port

```
! Load balancing is applied globally for all EtherChannel bundles
Switch(config)# port-channel load-balance TYPE
Switch(config)# exit
…
Switch# show etherchannel load-balance
```

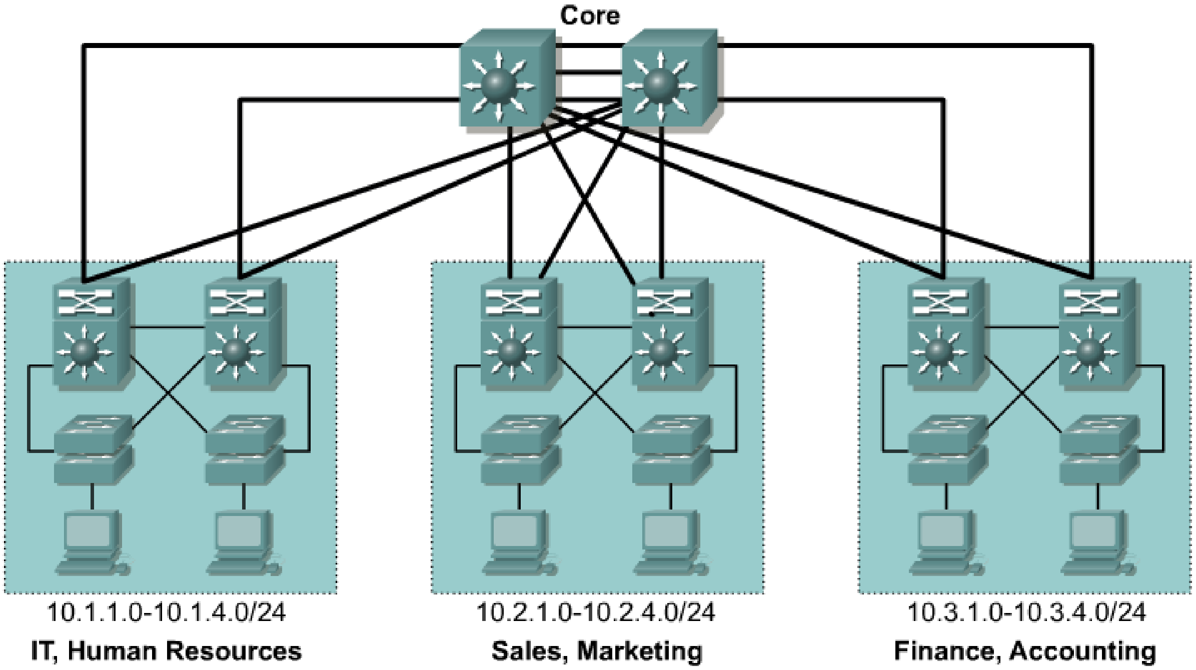
# Load-balancing Notes

- Equal balancing can be achieved only if the EC has 2, 4 or 8 ports
  - Different number of ports leads to unequal load balancing
  - EC can contain arbitrarily number of ports (from 1 to 8)
- Document ID: 12023, „Understanding EtherChannel Load Balancing and Redundancy on Catalyst Switches“

# VLAN Design Recommendation

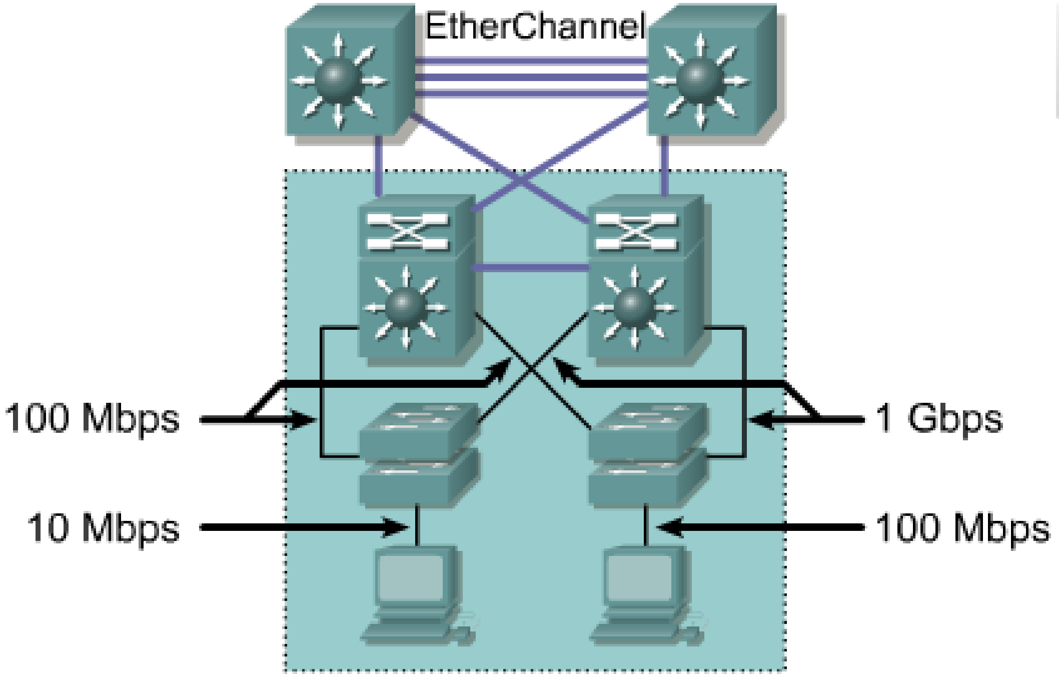
# VLAN Design and Addressing

- IP address space should be allocated in continuous block
- Single IP network = single VLAN
- VLAN should terminates at the distribution switch



# Recommended technology

- Fast Ethernet – end devices connected to access switch
- GigaEthernet – connection between access/distribution and distribution/core layer (even servers but nowadays also connection to users)
- 10 GigaEthernet – at the core layer
- EtherChannel bonding
- Connection should consider
  - Future growth
  - Max 20:1 oversubscription between access and distribution layer
  - Max 4:1 oversubscription between distribution and core layer



# Best Practices

- Minimize number of VLANs on access switches, scope of the VLAN should be limited to access and distribution layer
- Avoid using VLAN 1
- Remove VLAN1 from allowed VLANs on the trunk ports
- Separate VLANs for voice, data, management, default, parking, native
- Consider using VTP
- Trunk port – static configuration, disable DTP
- Using 802.1Q tagging instead of ISL
- Configure non-trunk ports as access ports
- Unused ports assign to parking VLAN, which should be suspended
- Using secure protocols in management VLAN

# Terminology and other notes

- Use technical terms carefuly especially when speaking with other network admins
  - Different vendors have different terms for the same technology
  - Untagger/tagged vs. Access and trunk
  - Trunk vs etherchannel vs bridge aggregation
- VLANs must be usually explicitly allowed (except Cisco)
- Be carefull with data analysis – network card or OS often strip VLAN tag
- Don‘t forget the difference between L2 and L3 topology
  - VLAN can create L2 link in routed topology which is sometime confusing

Slides adapted by Matěj Grégr and Vladimír Veselý
partially from official course materials
but most of credit goes to CCIE#23527 Ing. Peter Palúch, Ph.D.

The last update: 2016-10-05