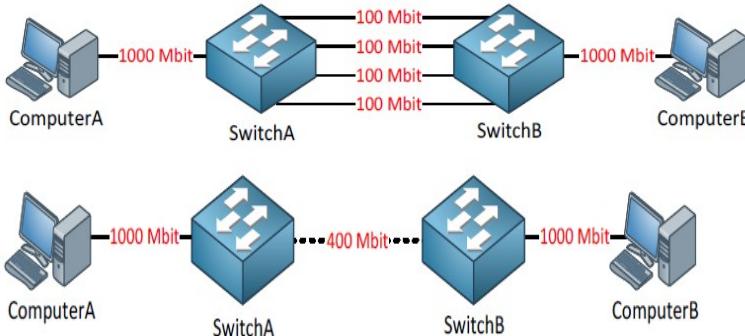


## EtherChannel:

**Etherchannel (link/bandwidth aggregation in the form of redundancy)** is a technology that lets you bundle multiple physical links into a single logical link. STP blocks redundant links whereas etherchannel can be used to provide redundancy i.e. load balance those links and will see it as one logical link. !(This doesn't work like MLPPP i.e. not true bandwidth increased)



### » EtherChannels consists of two parts

- Port-Channel interface
  - Logical interface representing the link bundle
- Member interfaces
  - Physical links that are part of the link bundle

### » LAG goal is to hide the member interfaces from the upper layer protocols

- E.g. STP sees one 2Gbps link not 2 x 1 Gbps link
- Result is active/active forwarding instead of active/standby with STP

### » Pro – cheap incremental upgrade solution

- E.g. 2 x 10GigE likely cheaper than moving to 40GigE or 100GigE

### » Pro – adds link layer redundancy

- E.g. 4 x 10GigE likely has better resiliency than 1 x 40GigE

### » Con – flows cannot exceed the bandwidth of an individual link

- E.g. 2Gbps LAG is not a 2Gbps pipe, but 2 x 1Gbps pipes
- LAG adds lanes to the highway but doesn't increase the speed limit

### » Con – flows can get polarized to one member of the LAG

- LAG supports load distribution but not load balancing
- E.g. 1 x 40GigE gives better throughput than 4 x 10GigE

1. Other vendor call it a LAG port (Link Aggregation Groups) instead of etherchannel
2. 2950 supports max 8 ports to aggregate/max 8 ports can channel together (platform dependent)
3. Even servers now support etherchannel (LACP compliant/NIC teaming) so LACP (LAG ports) can be configured between a server and a switch.
4. Port-Channel/Channel-Group is a logical Etherchannel Interface represents bonded links
5. Links should always be bundled as a multiple of 2 i.e. 2, 4, 8 etc.

**Maximum number of links:** 8 physical interfaces.

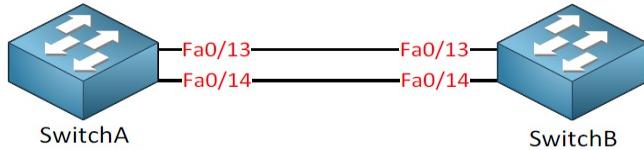
**Types:** PAgP (Port Aggregation Group Protocol) (Cisco proprietary) and LACP (Link Aggregation Control Protocol)(IEEE Standard/802.3ad)

**Must match on individual interface and the opposite side:** Duplex/Speed/Native VLAN/Allowed VLANs/Same switchport mode(access or trunk)!(No interfaces in a bundle can be SPAN ports)

**PagP Modes:** On/Desirable/Auto/Off

**LACP Modes:** On/Active/Passive/Off

## EtherChannel L2 Configuration:



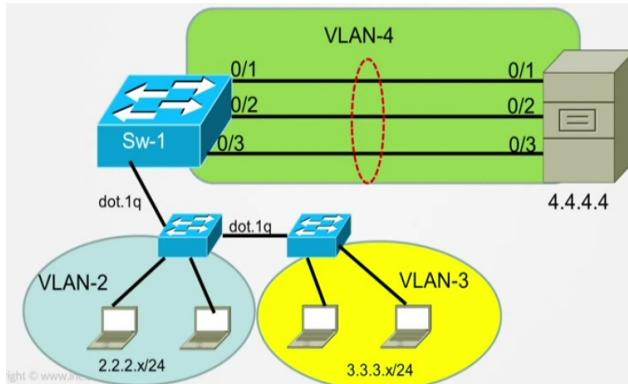
```

SwitchA(config)#default int range fa0/13 - 14
SwitchA(config)#int range fa0/13 – 14
SwitchA(config-if)#shut
SwitchA(config-if)#channel-group 1 mode desirable !(options(LACP/PAgp): active/desirable, passive/auto, on)
SwitchA(config-if)#no shut
SwitchA(config)#interface port-channel 1
SwitchA(config-if)#switchport trunk encapsulation dot1q
SwitchA(config-if)#switchport mode trunk
!(same an opposite config on SwitchB)
!(same config when using LACP except for modes are different)
!(Any changes now made on the port-channel interface will propagate down to member interfaces)
!(Some platforms even protects you to make any changes on member interfaces individually)
!(You can use the channel-protocol command to restrict anyone from selecting a mode that is not applicable to the selected protocol)

```

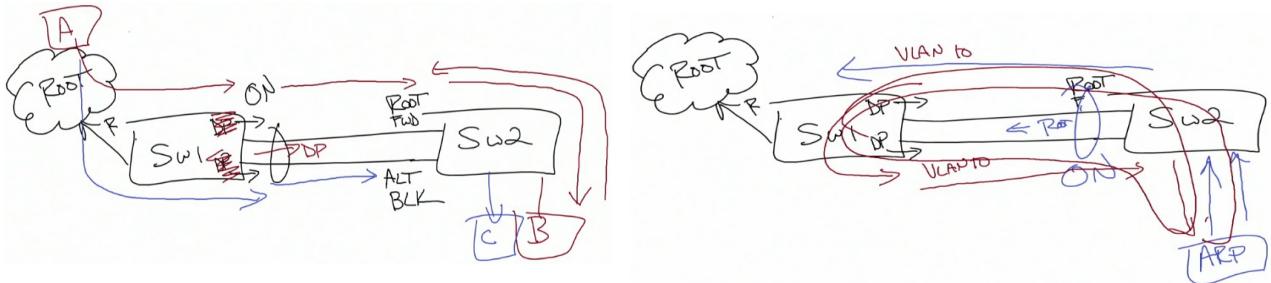
	<b>On</b>	<b>Desirable/Active</b>	<b>Auto/Passive</b>	<b>Off</b>
<b>On</b>	Yes	No	No	No
<b>Desirable/Active</b>	No	Yes	Yes	No
<b>Auto/Passive</b>	No	Yes	No	No
<b>Off</b>	No	No	No	No

## Problems with L2 EtherChannel:



1. Put the server and the PCs in different VLANs to avoid broadcasts (flooded traffic) in the same domain (inherent problem in L2 etherchannel).
2. If any of the interfaces goes down switch transitions through listening/learning etc. states. Also spanning tree converges again if the cost has changed (inherent problems in L2 etherchannel again).

## L2 Problem: Order of operation when using ON mode is important:



!(If you are using mode ON then the order of operation is really key so always use 'shut' and 'no shut' commands as a protection mechanism)

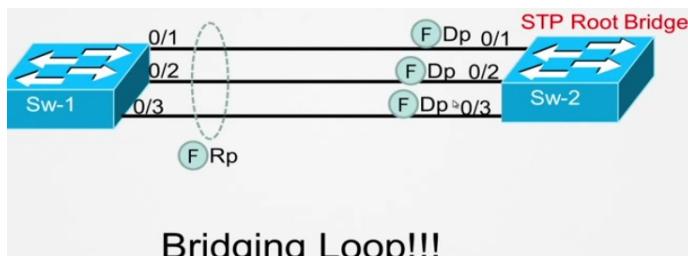
!(We are not allowed to learn mac addresses on blocked interface so the frame will be dropped)

!(Also could cause loops, based on the order of operations, if ON mode is used)

!(This is why we use negotiation i.e. LACP or PAgP instead of just ON mode)

## EtherChannel misconfiguration guard:

!(All BPDUs transmitted from individual ports in an etherchannel have the same Sending-Port-ID)



1. ON by default
2. Detects (by inspecting STP Port-ID of received BPDUs) if remote end etherchannel is misconfigured
3. Places ports in err-disable if misconfiguration detected

SW1(config)#spa etherchannel guard misconfig !(ON by default)

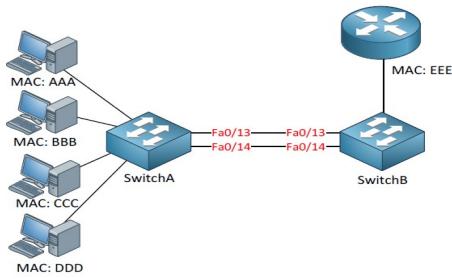
## EtherChannel L3 Configuration:

```

SW1(config)# interface port-channel 1
SW1(config-if)# no switchport
SW1(config-if)# ip address 172.16.1.11 255.255.255.0
SW1(config)# default int range fastethernet0/1 - 2
SW1(config)# int range fastethernet0/1 - 2
SW1(config-if-range)# no switchport
SW1(config-if-range)# no ip address
SW1(config-if-range)# channel-group 1 mode desirable
!(same an opposite config on SW2)
!(same config when using LACP except for modes are different)

```

**Load-Balancing (default 'src-mac')**: dst-ip/dst-mac/src-dst-ip/src-dst-mac/src-ip/src-mac



!(Switch creates a hash based on the type of load balancing method)

!(OK to have default 'src-mac' on SwitchA, but ideal to have 'dst-mac' on SwitchB)

!(Some newer switch platforms can even do load balancing based on port number i.e. tcp/udp ports)

**SwitchB(config)#port-channel load-balance dst-mac**

!(Higher level platforms like Nexus7K allows you to configure load balancing per port-channel, but Catalyst switches only allows it as a global command)

### Single vs. MultiChassis EtherChannel:

#### » EtherChannels cause fate sharing at the access layer

- E.g. servers may have redundant power, redundant NICs, but what happens if the access switch goes down?

#### » One solution - use two access layer switches

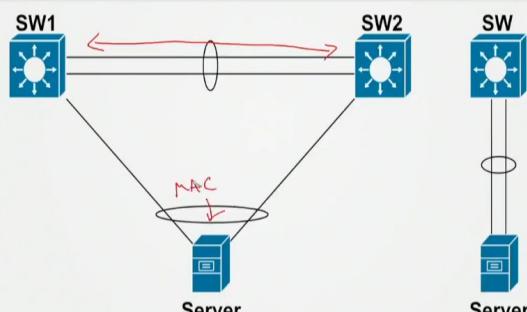
- Problem is that NICs usually only support active/standby outside of LAG
- I.e. 50% of bandwidth is lost unless the application supports load distribution at upper layers

#### » Better solution - trick the server into running LAG

- Form a logical chassis between two physical switches
- Result is that the server thinks it has two connections to the same switch
- 100% of bandwidth can be used, and more resiliency is added

### MLAG (MultiChassis Link Aggregation)

MultiChassis EtherChannel Physical vs. Logical How MultiChassis EtherChannel (MCEC) Works



» Goal is to turn physical triangle into logical P2P link

- Downstream box thinks upstream boxes are one
- This is what allows active/active

» MCEC/MLAG relies on synchronized control plane between switches

- Forward the data plane out member ports of both chassis
- Synchronize the control plane on the backend transparently
- Synchronization is proprietary and not inter-operable

### » Virtual Switching System (VSS)

### Cisco's MCEC Implementations

- Aggregation platforms, e.g. Catalyst 4500/6500/6800
- Control plane sync over Virtual Switch Link (VSL)
  - Typically 2 x 10GigE LAG
- One control plane is shared among VSS
- One management plane is shared among VSS
- Usually 1 active supervisor and 3 standby supervisors

#### » StackWise Cross-Stack EtherChannel

- Access platforms, e.g. Catalyst 3750/3850
- Control plane sync over dedicated stacking cables
- Stack cable creates a bidirectional closed-loop
- One control plane is shared among stack members
- One management plane is shared among stack members

## » Virtual Port Channel (vPC)

- Data Center platforms, e.g. Nexus 5000/7000/9000
- Control plane synchronized over a vPC Peer Link
  - Typically 2 x 10GigE LAG
- *Two independent control planes* in the vPC
- *Two independent management planes* in the vPC
- Usually 2 active supervisors and 2 standby supervisors

## StackWise vs. VSS vs. vPC

- » StackWise can have more than 2 members, up to the stack limit
  - Stack limit depends on platform, e.g. 9 on 3750-X
- » VSS and vPC are always a pair of switches
- » Logical result of all three is the same
  - Turns a physical triangle into a logical P2P link

### **Verification and Troubleshooting:**

1. Make sure you use the same etherchannel modes (types) on both sides and verify it using show commands. Cannot mix PagP and LACP
2. Make sure at least one of the two switches uses desirable/active mode
3. Make sure all the interfaces that needs adding to a port-channel have exact same config e.g. Speed/duplex/allowed VLANs/switchport (access/trunk)
4. Do not configure a GigaStack gigabit interface converter (GBIC) as a part of EtherChannel
5. A SPAN interface will not join etherchannel until SPAN is disabled
6. Do not configure a secure port as a part of an etherchannel

```
SW1#clear pagp 1 counters
SW1#clear lacp 1 counters
```

```
sh etherchannel summary !(best command to see if channel group is up or down)
sh etherchannel summary | begin Group !(to see the grouped interfaces)
sh etherchannel 1 port-channel
sh interface port-channel 1
sh spa active !(to verify etherchannel misconfiguration i.e. Bridging loop)
sh spa summary !(to see if etherchannel misconfig guard is enabled. ON by default)
sh etherchannel
sh etherchannel 1 detail !(to see the modes)
sh etherchannel 1 detail | inc reason !(to see the reason why port-channel was unable to bundle)
sh ip int bri | inc Port !(to see if the port-channel is up)
sh int fa0/12 etherchannel
sh pagp neighbor
sh pagp counters
sh lacp neighbor
sh lacp counters
sh etherchannel load-balance
sh spa int port-channel 1 !(to see the spa cost for port-channel)
sh int trunk
sh int trunk port-channel 1
sh spa vlan 1
sh mac-address-table int po1 !(mac addresses are associated with po1 not member interfaces)
```