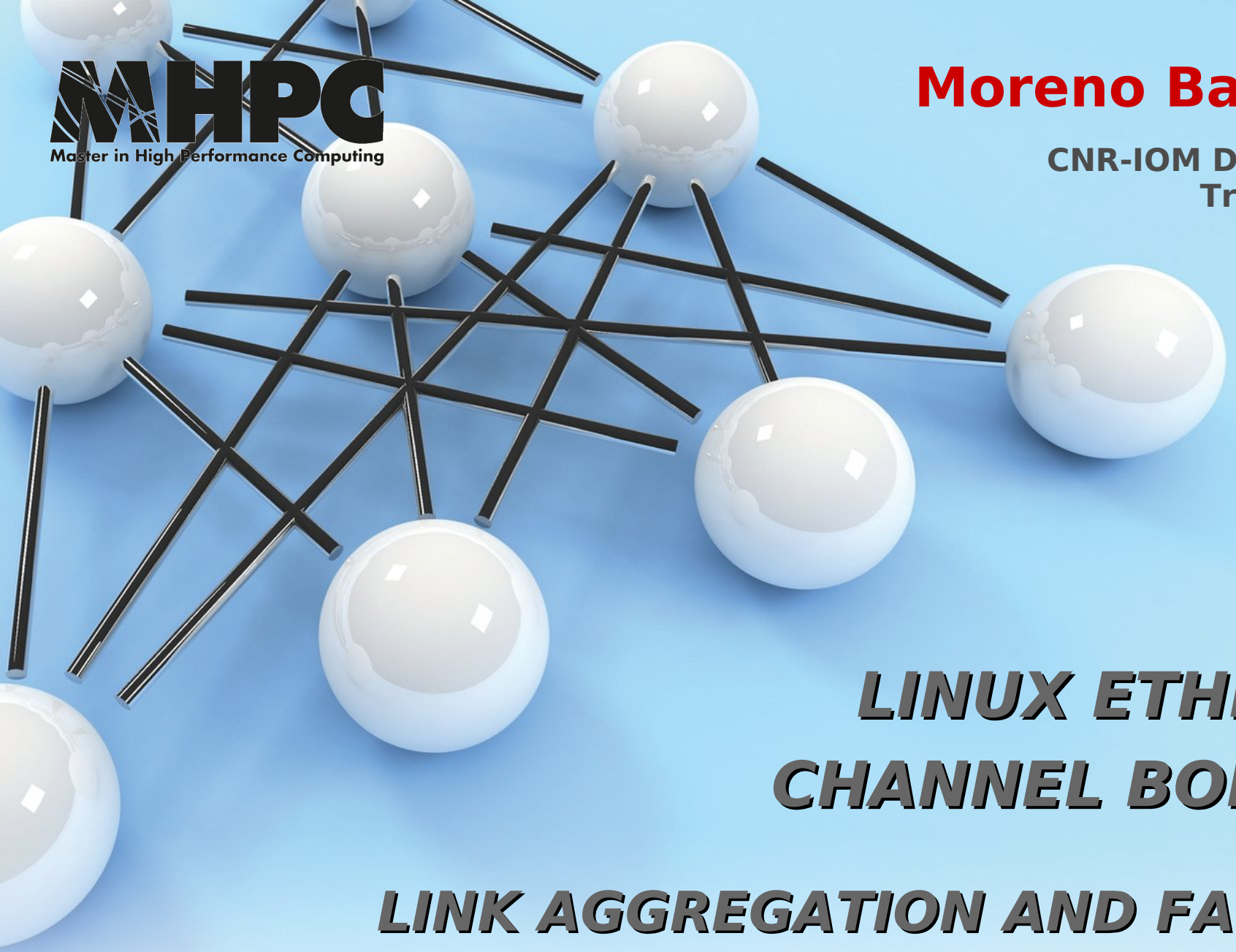


MHPC
Master in High Performance Computing


**Moreno Baricevic**

CNR-IOM DEMOCRITOS
Trieste, ITALY


# LINUX ETHERNET CHANNEL BONDING

## LINK AGGREGATION AND FAILOVER

SISSA
ma per seguir virtute e canoscenza
Scuola Internazionale Superiore di Studi Avanzati

ICTP
The Abdus Salam International Centre for Theoretical Physics

# Agenda

- Port Trunking, Link Aggregation, NIC Teaming, Ethernet Channel Bonding, Etherchannel?
- About throughput, latency and high availability
- LINUX Ethernet Bonding Driver
- Bonding modes
- Link aggregation and LACP
- Failover
- LINUX commands (configuration, diagnostic, benchmarking)
- Switch configuration (Cisco and HP examples)

# Port Trunking, Link Aggregation, NIC Teaming, Ethernet Channel Bonding, Etherchannel?

Different names for similar tecnologies.

Same purpose: provide fault tolerance and/or greater bandwidth.

**Link Aggregation**: general term that describes various methods of combining multiple network connections

**LACP (Link Aggregation Control Protocol):** IEEE 802.3ad, independent standard (became 802.1ax in 2008)

**Ethernet Channel Bonding**: LINUX main and historical software implementation (kernel-space)

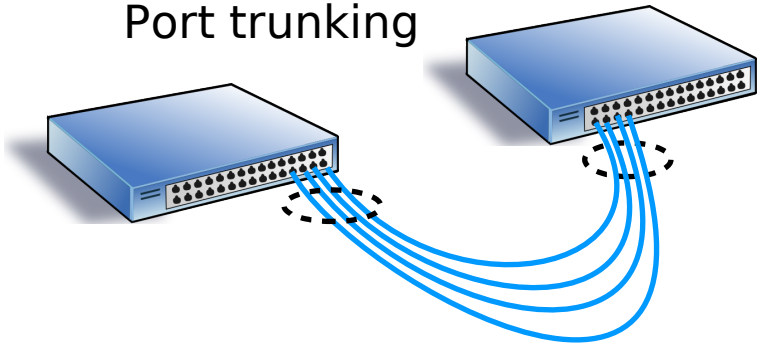
**Linux Team Driver (libteam)**: new LINUX project implemented in user-space (*teamd* daemon)

**Port Trunking**: (general term, switch configuration) method that combine more ports into a single virtual channel. Various protocols may define the (auto)configuration of the channel.

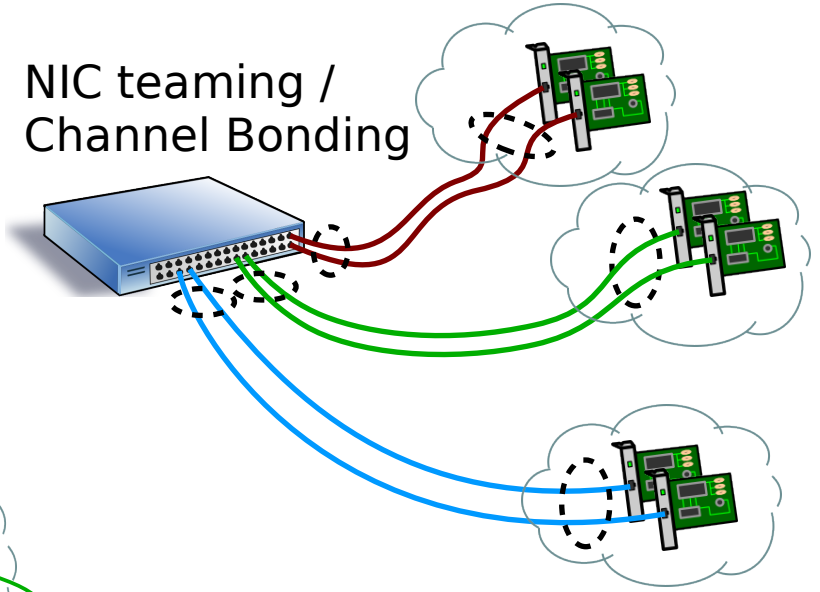
**EtherChannel**: as above, for Cisco technologies

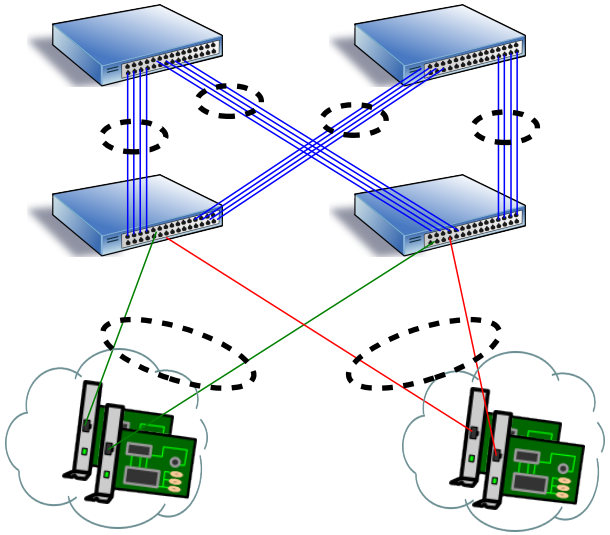
# Port trunking and NIC teaming

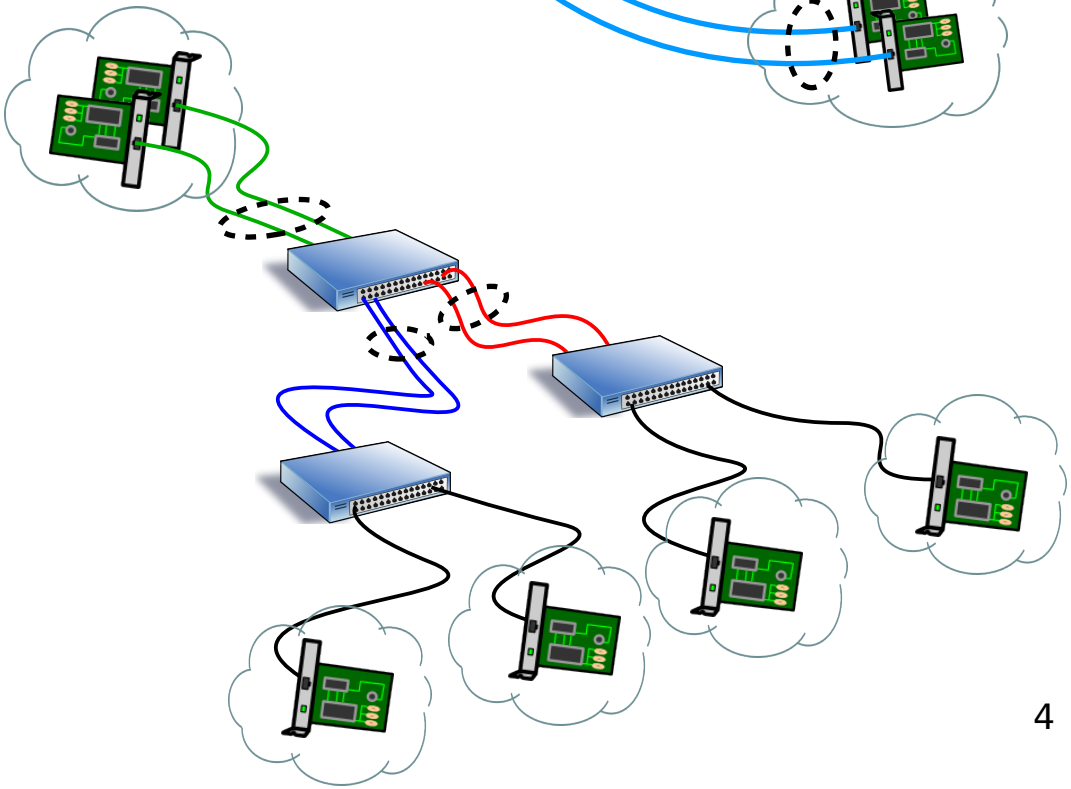
Port trunking



NIC teaming /
Channel Bonding

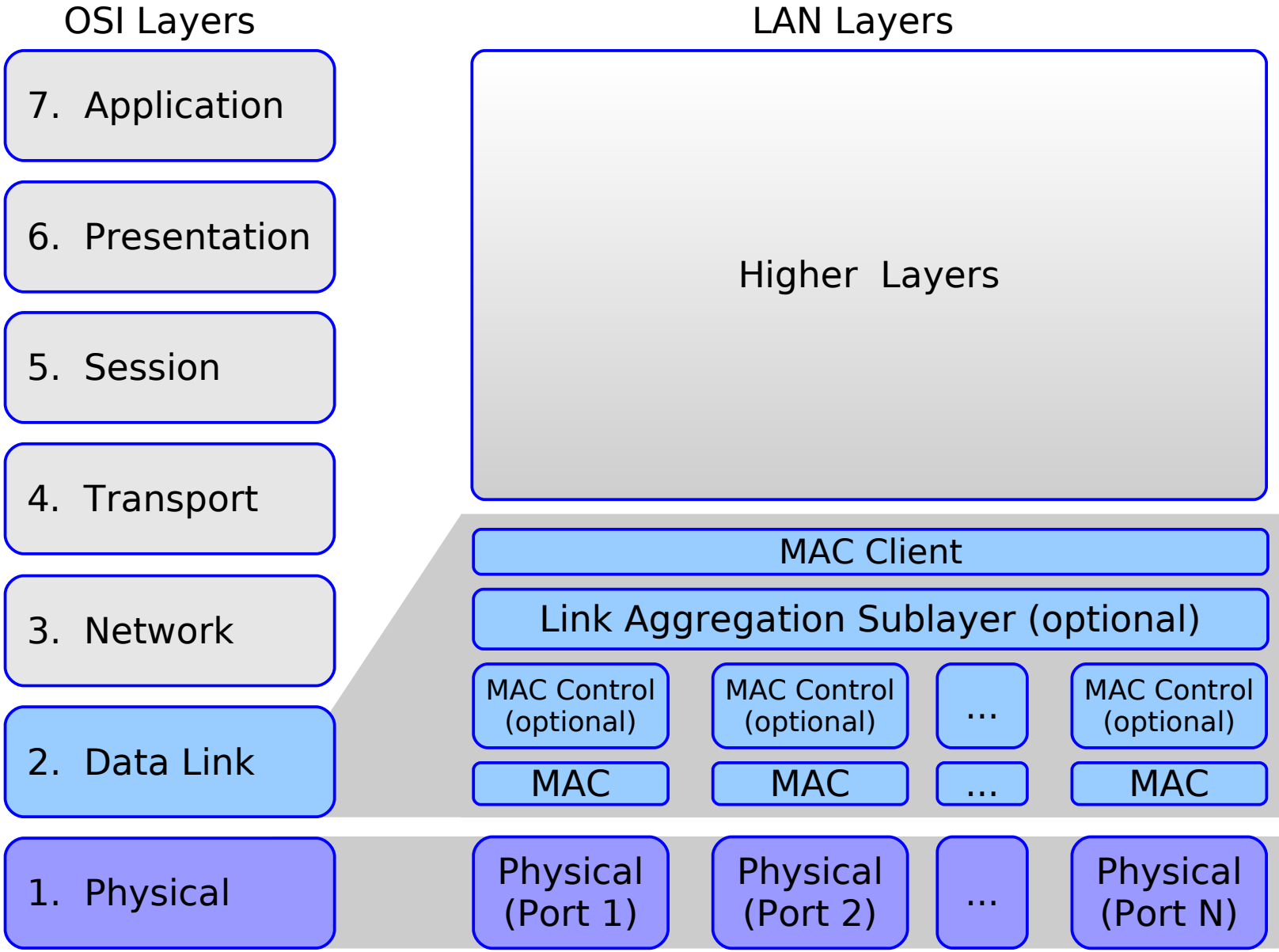




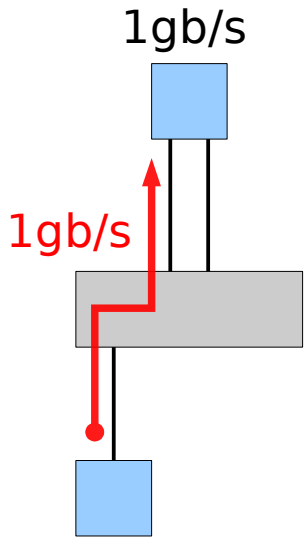


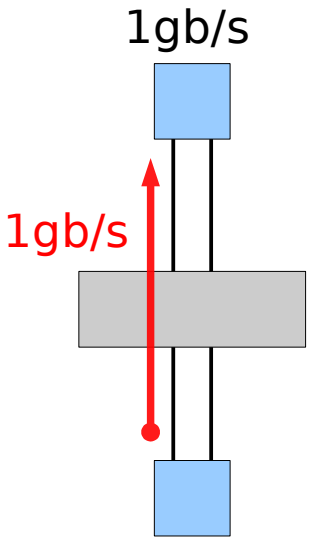
# LACP

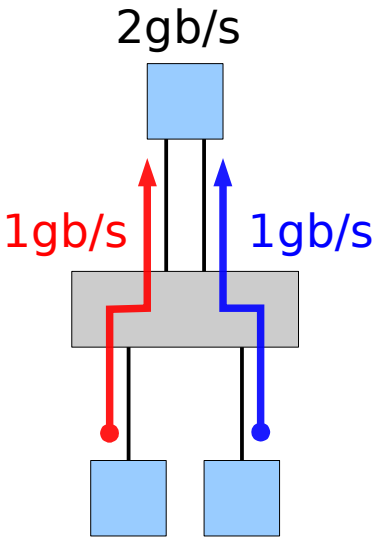
***IEEE Std 802.1AX-2008***

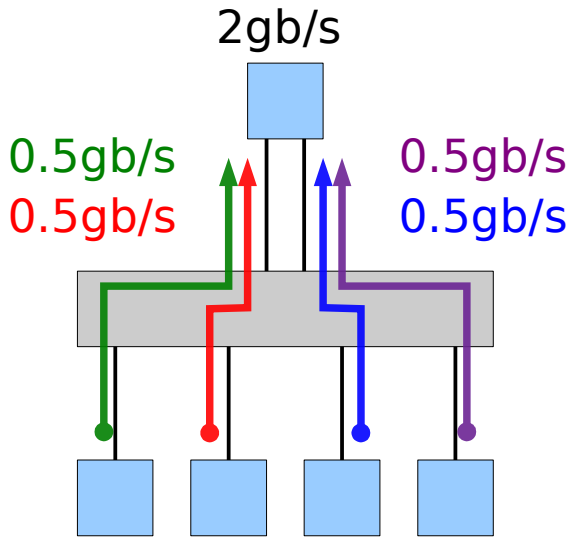


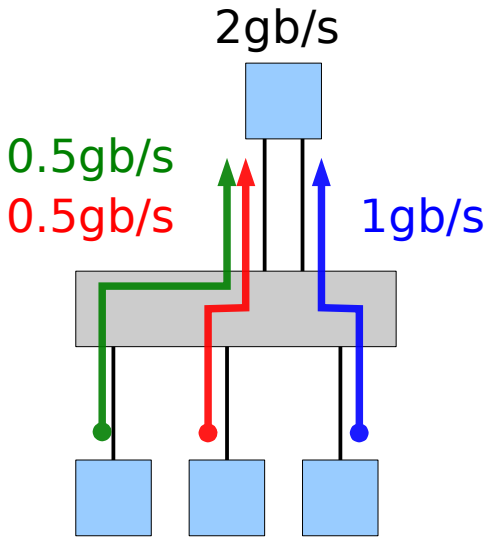
# Aggregated bandwidth and (not-so-intuitive) balancing 1/2



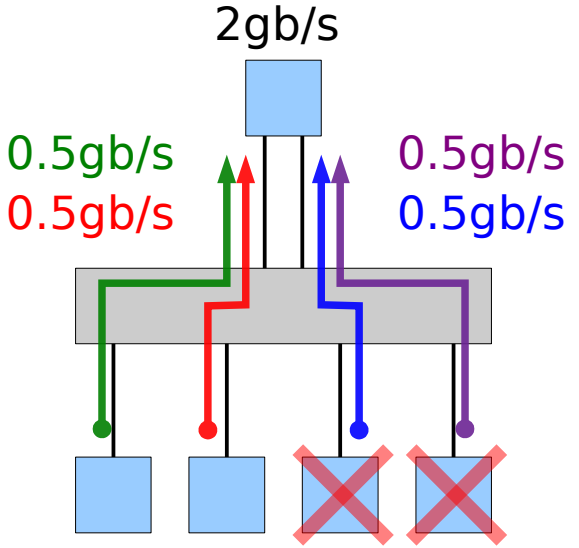



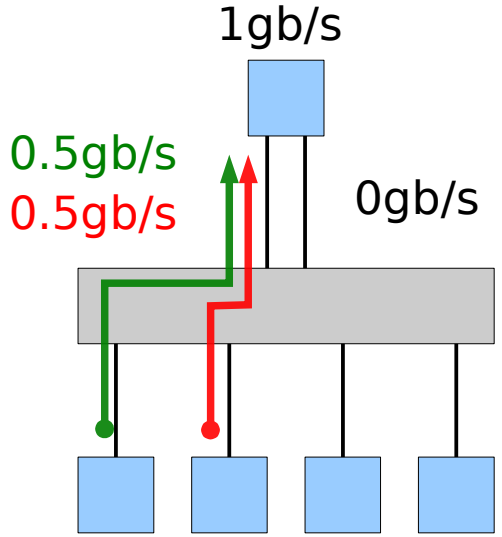



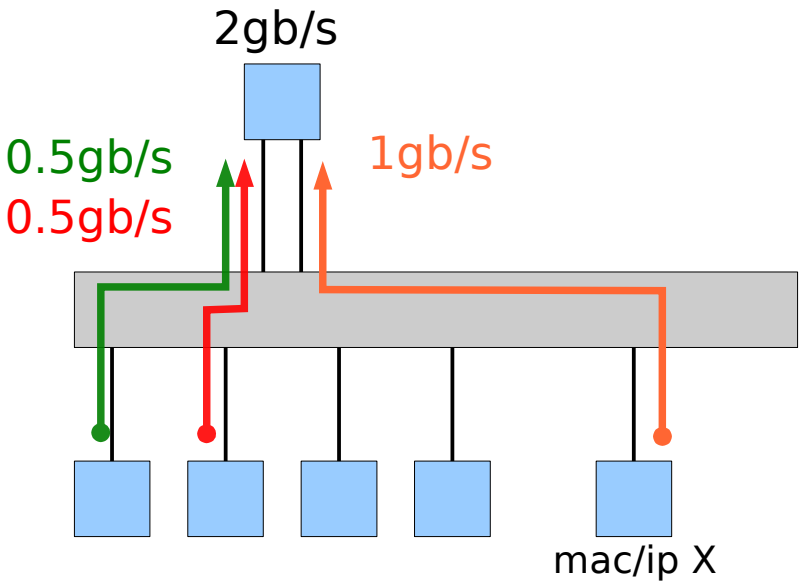



# Aggregated bandwidth and (not-so-intuitive) balancing 2/2



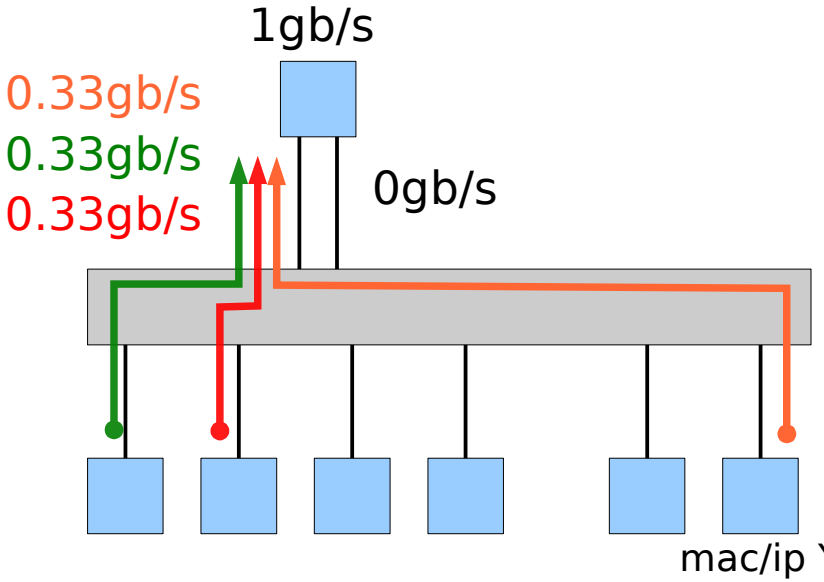




Same src-dst connection use always the same link in order to avoid retransmissions (due to out-of-order packets) and the consequent impact on performance.

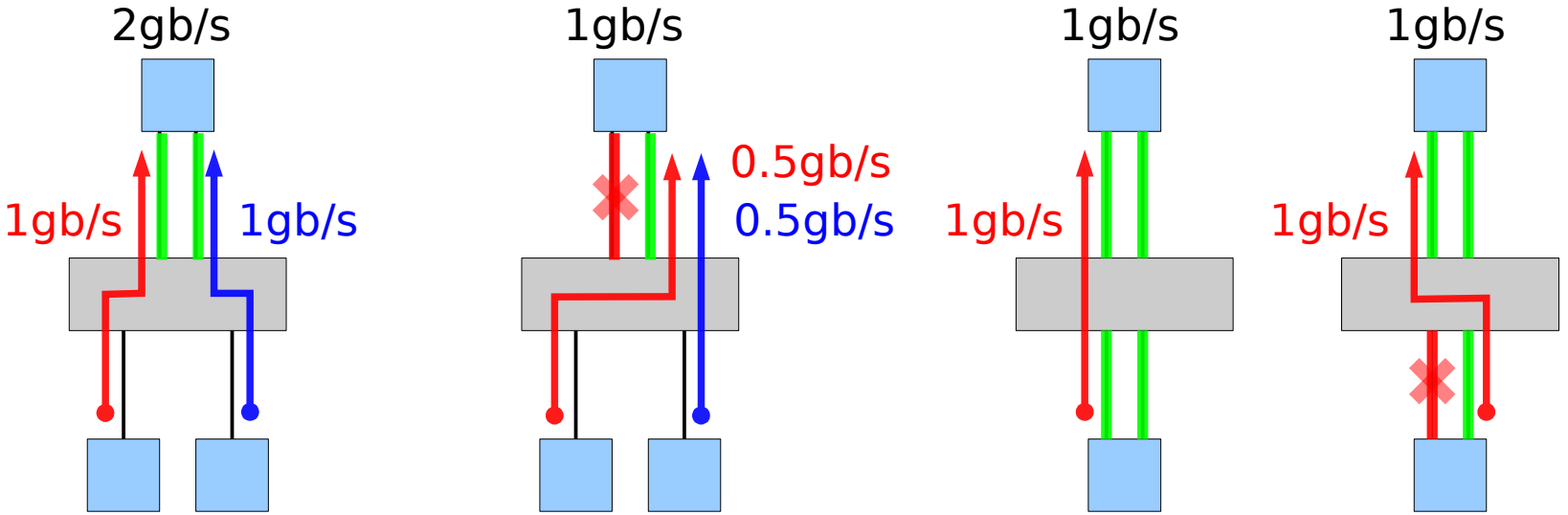
The balancing is usually based on src/dst MAC (L2) or src/dst IP (L3) addresses, or various combinations of both. (L4 balancing is also implemented on some high-end devices virtually reaching 50/50)

(cisco: show etherchannel load-balance)

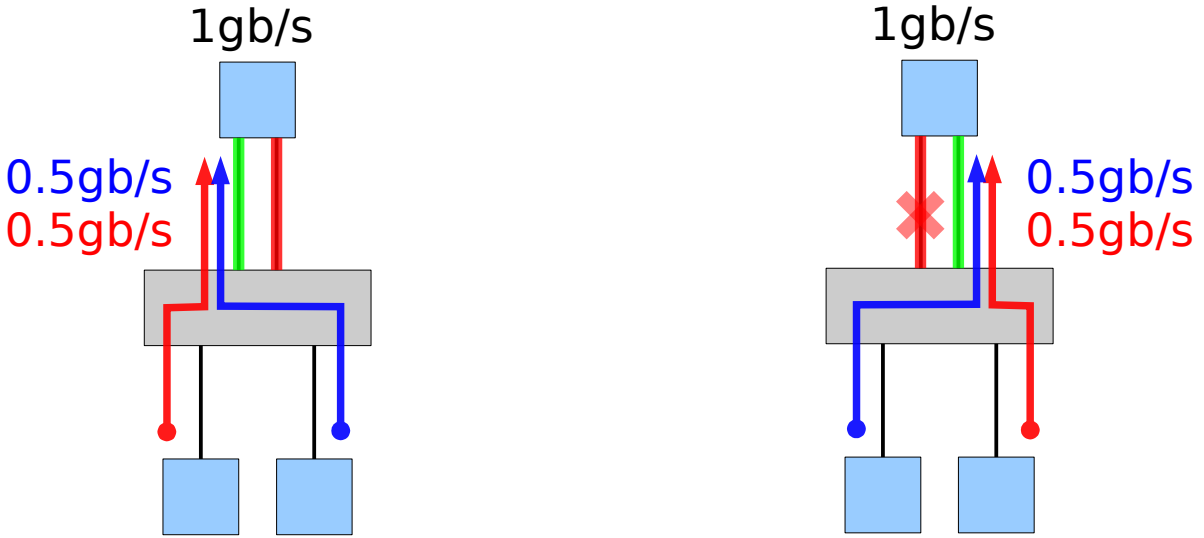


# Fault tolerance

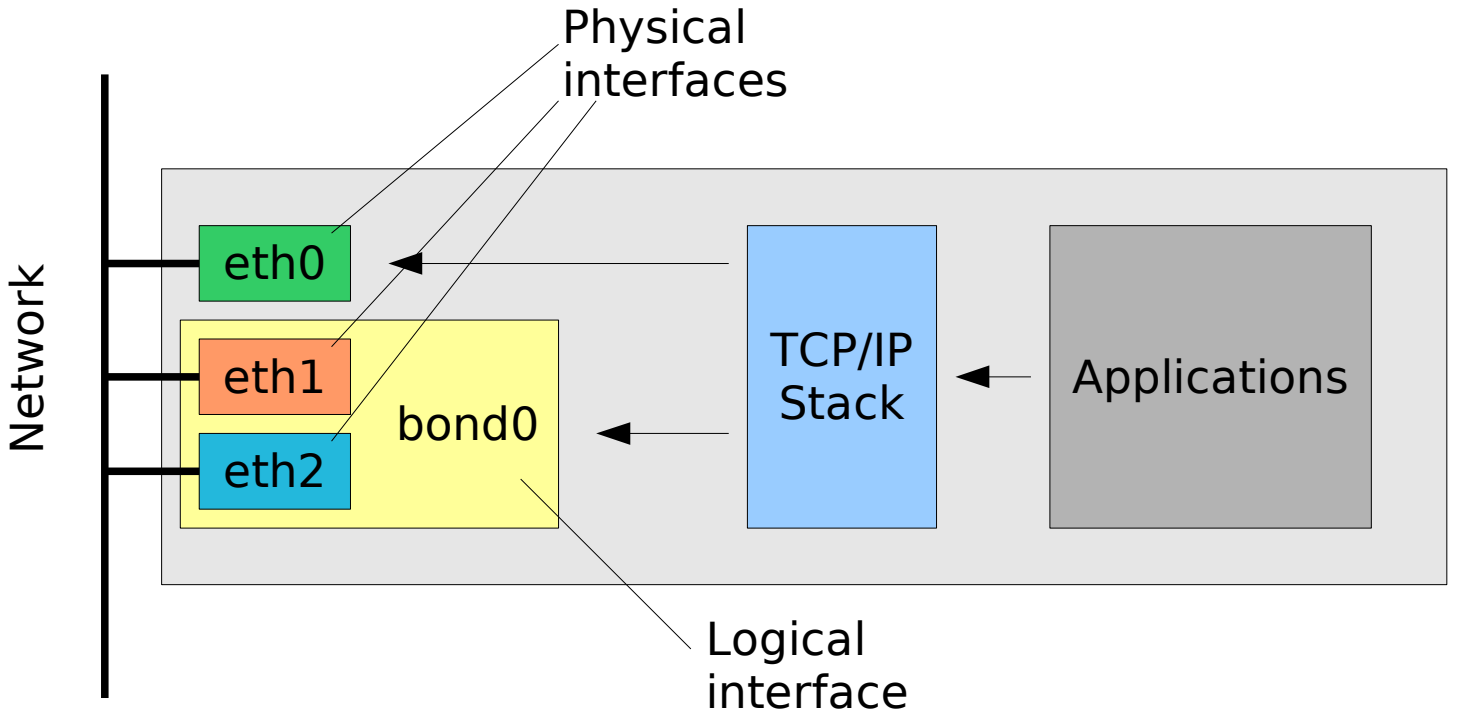
Active-active / Link Aggregation




Active-passive / Fault Tolerance



# LINUX Ethernet Channel Bonding




**eth0**: has it's own MAC and IP address, configured as usual

**bond0**:
- forces the **same** MAC address on both the slaves (**eth1** and **eth2**);
- the MAC address used is the one of the first interface enslaved;
- the IP address belongs to bond0, not eth* (ifconfig bond0 ...);
- depending on the bonding mode adopted, addictional configuration may be required on the switch.

# Bonding modes on LINUX

**balance-rr / 0** (Round-robin)
load balancing and failover

**active-backup / 1**
fault tolerance

**balance-xor / 2**
load balancing and failover

**broadcast / 3**
fault-tolerance

**802.3ad / 4**
IEEE 802.3ad Dynamic link aggregation (LACP)

**balance-tlb / 5** (adaptive transmit load balancing)
load balancing and failover

**balance-alb / 6** (adaptive load balancing)
load balancing and failover

DOES NOT REQUIRE ANY SPECIAL SWITCH SUPPORT OR CONFIGURATION

REQUIRES A SWITCH THAT SUPPORT LACP AND A SPECIAL CONFIGURATION IS NEEDED

# LINUX COMMANDS
# command line, using std. utilities

Configuration:

```
# modprobe bonding mode=4 miimon=100
# ifconfig bond0 10.1.0.1 netmask 255.255.0.0 up
# ifenslave bond0 eth1 eth2
```

De-configuration:

```
# ifenslave -d bond0 eth1 eth2
# ifconfig bond0 down
# modprobe -r bonding
```

LACP: `mode=4` or `mode=802.3ad`
FAILOVER: `mode=1` or `mode=active-backup`

# LINUX COMMANDS
## command line, using sysfs+iputils

```
# modprobe bonding
# echo +bond-lacp > /sys/class/net/bonding_masters
# ip link set dev bond-lacp down
# echo '4' > /sys/class/net/bond-lacp/bonding/mode
# echo '100' > /sys/class/net/bond-lacp/bonding/miimon
# ip link set dev bond-lacp up
# echo +eth1 > /sys/class/net/bond-lacp/bonding/slaves
# echo +eth2 > /sys/class/net/bond-lacp/bonding/slaves
# ip addr add 10.1.0.1/16 brd 10.1.255.255 dev bond-lacp
  label bond-lacp
```

# LINUX COMMANDS
# permament configuration

```
# /etc/modprobe.d/bonding.conf
alias bond0 bonding
```

```
# /etc/sysconfig/network-scripts/ifcfg-bond-lacp
DEVICE=bond-lacp
NM_CONTROLLED=no
ONBOOT=yes
USERCTL=no
BOOTPROTO=dhcp
LINKDELAY=10
BONDING_OPTS="mode=X miimon=100"
IPADDR=A.B.C.D
...
```

```
# /etc/sysconfig/network-scripts/ifcfg-eth[12]
DEVICE=eth[12]
HWADDR=AA:BB:CC:DD:EE:FF
NM_CONTROLLED=no
ONBOOT=yes
USERCTL=no
SLAVE=yes
MASTER=bond-lacp
```

# LINUX COMMANDS
# check running configuration

```
# lsmod | grep bonding
# ip addr
# ip addr ls DEVICE
# ifconfig
# ifconfig DEVICE
# ifenslave -a
# cat /proc/net/bonding/DEVICE
# grep . /sys/devices/virtual/net/DEVICE/bonding/*
```

# Final remarks

- Link aggregation:
  - increases the bandwidth for servers/nodes that must be accessed by multiple clients, useless otherwise
  - requires high-end switch ($$$)
  - fault-tolerance/failover is provided too
- Active-backup:
  - no particular requirements about the switch
  - fault-tolerance/failover is native
- What to use, then?
  - consider the needs AND the cost
  - fault-tolerance using active-backup comes free (just one more cable and a switch port), so why not?
  - sometimes, upgrading to NICs and switch with higher base-operating speed may be a better option

# That's All Folks!

"In case you need the network administrator, he's hiding under his desk in the fetal position."

```
( questions ; comments ) | mail -s uheilaaa baro@democritos.it

( complaints ; insults ) &>/dev/null
```