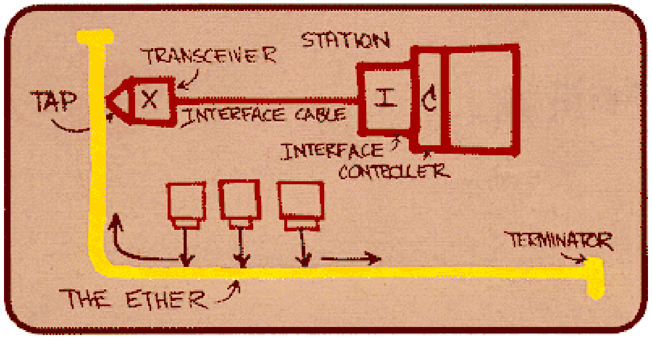
Basic Ethernet

What is Ethernet anyway?

Ethernet has evolved far from its roots of half-duplex CSMA/CD LANs and is hard to pin down today

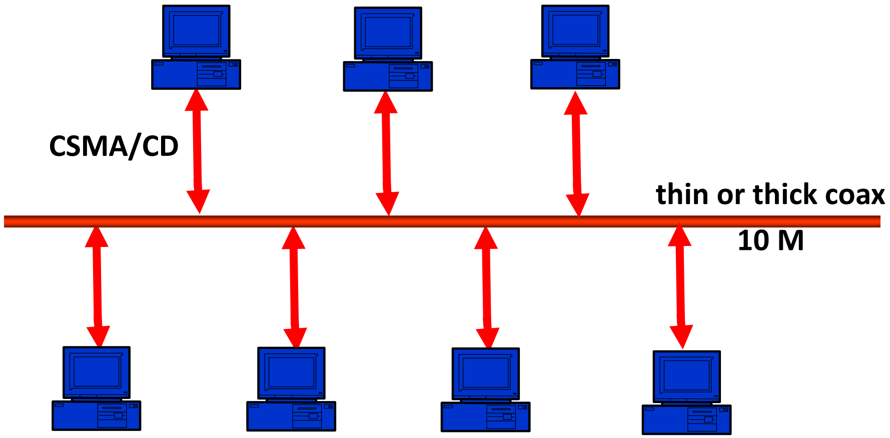


Metcalf's original sketch of Ethernet

We may use the term today to describe

- full duplex 10G point-to-point optical links
- wireless Ethernet (WiFi) hot spots
- *Ethernet in the first mile* DSL access
- passive optical *GEPON* networks
- metro Ethernet networks
- carrier-grade Ethernet services
- Ethernet **V**irtual **P**rivate **N**etworks
- etc.

Ethernet LANs, then and now



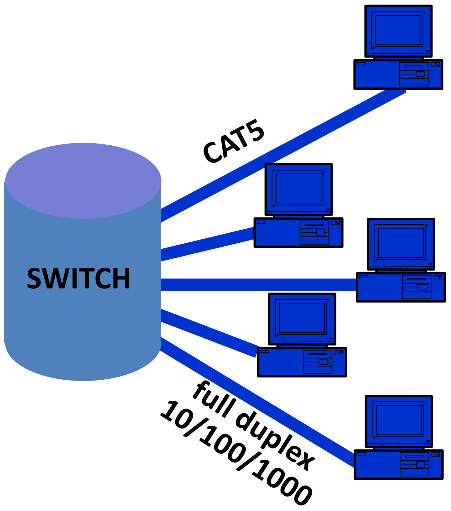
Bus topology

Single collision domain

Star topology

Independent FD transmission

Switch with buffering



IEEE 802, misc WGs, documents

Ethernet is defined by the **IEEE 802** LAN/MAN Standards Committee

- 802-2001
- 802.1 LAN protocols WG
  - **802.1D-2004**
  - **802.1Q-2005**
  - 802.1ad
  - 802.1ah
- 802.2 LLC
- 802.3 **Ethernet**
  - **802.3-2005**
  - 802.3z GbE
  - 802.3ad link aggregation
  - 802.3ah EFM
  - 802.3as 2000 byte frames
- 802.11 Wireless LAN (WiFi)
  - **802.11-2005**
  - 802.11a,b,g
- 802.16 Broadband Wireless Access (WiMax)
- 802.17 **R**esilient **P**acket **R**ing

Note:

working groups and study groups (e.g 802.1, 802.3) are semi-permanent

projects and task forces (e.g. 802.3z, EFM) are temporary

project outputs are usually absorbed into main WG document

MAC frame format

a *MAC frame* uses either of the following frame formats :

64 – 1518 B

DA (6B)	SA (6B)	T/L (2B)	data (0-1500B)	pad (0-46)	FCS (4B)

64/68 – 1522 B

DA(6B)	**SA**(6B)	**VT**(2B)	**VLAN**(2B)	**T/L**(2B)	data (0-1500B)	**pad**(0-46)	**FCS**(4B)
		8100					

T/L is *Ethertype* or *Length*

802.3as expanded frame size from 1500 to 2000B (since September 2006)

Physical frame format

When using the native IEEE (ETY) physical layer
the *physical frame* has the following formats :

PREAMBLE	SFD (1B)	MAC FRAME	IPG

- Preamble : 7 bytes of 10101010
- Start Frame Delimitator : 10101011
- InterPacket Gap : 12 (or 8) bytes of idle before next frame

But the MAC (ETH) layer network
is independent of the physical (ETY) layer network
and MAC frames can be transported over other many server networks :

- coaxial cable, twisted copper pairs, optical fibers (IEEE 802.3)
- synchronous (TDM) networks (PPP, HDLC, GFP, EoS, LAPS)
- packet switched networks, including
  - IP (EtherIP RFC 3378)
  - MPLS (Ethernet PW (RFC 4448, Y.1415), L2VPN (VPWS/VPLS))
  - Ethernet (MAC-in-MAC 802.1ah)

802.3

Actually, IEEE calls only 802.3 *Ethernet*

802.3 is a **large** standard, defining

- MAC frame format, including VLAN support
- medium specifications and attachment units (UTP, coax, fiber, PON)
- repeaters
- interfaces (e.g. MII, GMII)
- rate autonegotiation
- link aggregation (we will discuss later)

New projects continue to expand scope

Physical media are described by Rate-Modulation-CableLimits

- coax: 10BASE2, 10BASE5, 10BROAD36

- twisted pairs: 10BASE-T, 100BASE-TX, 1000BASE-T, 10PASS-TS, 2BASE-TL
- fiber-optic: 10BASE-FL, 100BASE-FX, 1000BASE-LX/SX, 10GBASE-SR/LR/ER/LX4

Ethernet Addressing

The most important part of any protocol's overhead are the *address fields*

Ethernet has both source (SA) and destination (DA) fields

The addresses need to be unique to the network

The fields are 6-bytes in length in EUI-48 format
(once called MAC-48, EUI = **E**xtended **U**nique **I**dentifier)
so that there are 2^{48} = 281,474,976,710,656 possible addresses

EUI-48 is shared by

- Ethernet (802.3)
- Token ring (802.5)
- WiFi (802.11)
- Bluetooth
- FDDI
- SCSI/fiber-channel

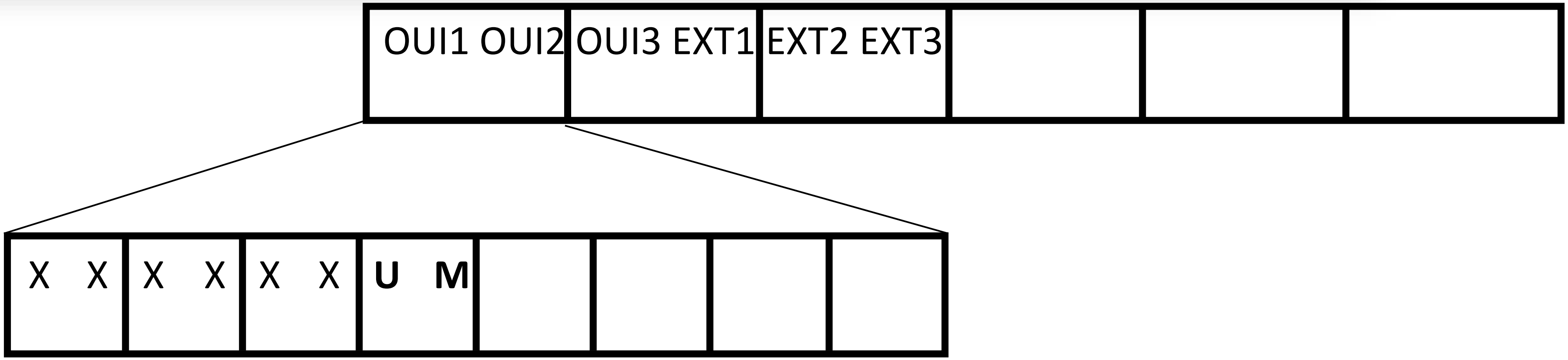
IEEE defined a "next generation" 8-byte address called EUI-64 used by

- IEEE 1394 (firewire)
- 802.15.4 (personal area networks)
- IPv6 (LSBs of non-temporary unicast address)

Addresses can be

- *universally administered* (burned in) or
- *locally administered* (SW assigned)

EUI format



EUI addresses usually expressed in *hex-hex-hex-hex-hex-hex* format

OUI (ex “company name”) is assigned by the IEEE Registration Authority

For each OUI there are 16M addresses (IEEE expects not to run out before 2100)

The LSB of the OUI is the **M**ulticast indicator (0=unicast, 1=multicast)
Broadcast address is FF-FF-FF-FF-FF-FF

The next to LSB is the **U**niversal / local bit
- 0 means UNIVERSALLY allocated address (all assigned OUIs have zero)
- 1 means there is no OUI - use any unique address
- WARNING – bit is reversed in IPv6!

OUIs are also used by *LLC SNAP* and in *slow protocols*

Ethernet and IP addresses

Ethernet is often used to carry IP packets
- since IP does not define lower layers
- since IP only forwards up to the LAN, not to the endpoint

Both IP and Ethernet use addresses
- but these addresses are not compatible (exception – IPv6 local address)

The **IETF** defined the **A**ddress **R**esolution **P**rotocol (RFC 826 / STD 37) to solve this problem

If you need to know the MAC address that corresponds to an IP address
- broadcast an ARP request (Ethertype 0806, address FF…FF)
- all hosts on LAN receive
- host with given IP address unicasts back an "ARP reply"

Other ARP protocols

Other related protocols (some use the ARP packet format)

- GARP (gratuitous ARP – WARNING not 802.1 GARP)
  host sends its MAC-IP binding without request (e.g. backup server)
- Proxy ARP
  router responds to ARP request to capture frames
- Reverse ARP, BOOTP, DHCP
  host sends its MAC and wants to know its IP address
- Inverse ARP
  frame-relay station unicasts DLCI to find out remote IP address
- ARP mediation
  mediate over L2VPN between networks using different ARPs
  (e.g. Ethernet on one side and FR on the other)

Ethernet clients

The 2-byte *Ethertype* identifies the client type

Some useful Ethertypes (assigned by IEEE Registration Authority):

- 0800 IPv4
- 0806 ARP
- 22F3 TRILL
- 22F4 IS-IS
- 8100 VLAN tag
- 8138 Novell IPX
- 814C SNMP over Ethernet
- 86DD IPv6
- 8809 slow protocols
- 8847 MPLS unicast
- 8848 MPLS multicast
- 88D8 CESoETH
- 88A8 Q-in-Q SVID / MAC-in-MAC BVID
- 88E7 PBT I-tag
- 88F5 MVRP
- 88F6 MMRP
- 88F7 IEEE 1588v2
- 8902 CFM OAM (1ag and Y.1731)

see them all at

http://standards.ieee.org/regauth/ethertype/eth.txt

get your own for only $2,500 !

LLC

Older applications don't differentiate clients using Ethertype

802.2 (**L**ogical **L**ink **C**ontrol)

first three bytes of payload :

- **D**estination **S**ervice **A**ccess **P**oint (1B)
- **S**ource **S**ervice **A**ccess **P**oint (1B) (usually the same as DSAP)
- Control Field (1 or 2 B)

DA	SA	len	LLC	payload

Example SAPs

04	IBM SNA
06	IP
42	STP
80	3Com
AA	SNAP
BC	Banyan
E0	Novel IPX/SPX
F4	FE CLNS

SNAP

DA	SA	len	LLC-SNAP	payload

- **S**ub-**N**etwork **A**ccess **P**rotocol

  LLC parameters plus expanded capabilities

  SNAP can support IPX/SPX, TCP/IP, AppleTalk Phase 2, etc.

  the first eight bytes of payload :
  - LLC **D**estination **S**ervice **A**ccess **P**oint (1B) = 0xAA
  - LLC **S**ource **S**ervice **A**ccess **P**oint (1B) = 0xAA
  - LLC Control Field (1B) = 0x03
  - OUI (3B)
  - Type (2B) (if OUI=00:00:00 then EtherType)

- IPX (old Netware method, “raw”) - first 2B of payload FF:FF
  - Note: standard DSAP/SSAP values can not be FF !

- RFC 1042 allows IPv4 over Ethernet with SNAP
  - DSAP=AA, SSAP=AA, Control=3, SNAP=0 followed by Ethertype

Ethernet header parsing

if EtherType/Length > ~~1500~~ then EtherType

else if payload starts with FF-FF then Netware

else if payload starts with AA then SNAP

else LLC

DA	SA	len/Ethertype	XX	payload

CSMA/CD

Ethernet LANs are broadcast domains (AKA collision domains)

The original multiple access methodology was

Carrier **S**ense **M**ultiple **A**ccess with **C**ollision **D**etection (CSMA/CD)

- Is medium idle? If not, wait until it is (+ IPG)
- Start transmitting but monitor for collision during transmission
- If a collision is detected
  - transmit *jam signal* for long enough to ensure that all receivers detect
  - increment retransmission counter
  - if exceed maximum retransmissions then abort
  - wait random backoff time
  - retry transmitting

In a broadcast domains only one host can transmit at a time

This imposes severe limitations on :

- number of hosts on LAN
- size (fast Ethernet can not exceed 200 m)
- throughput

Bridges

The solution is to segment hosts into small LANs, connected by **bridges**

Bridges are full Ethernet receivers, but have no Ethernet addresses
    they relay frames that need to pass to the other side of the bridge

As implied by their name, Ethernet bridges are not *forwarding* devices
    but rather *filtering* devices

That is, there is never a decision as to *where* to forward
    only *whether* to forward

However, Ethernet standards do not enforce particular internal mechanisms
    as long as external operation is correct

This led to the development of more efficient Ethernet **switches**

Learning bridges

Ethernet addresses are merely arbitrary *identifiers*, not *locators*

IP addresses and telephone numbers are *at lest partially* locators

Thus, in order to filter frames, bridges maintain a **filtering database**
listing addresses that need to traverse the bridge
Such databases may need to store thousands of addresses

This database may be
- configured (manually, or from a **N**etwork **M**anagement **S**ystem)
- learned

Learning involves:
- *observing* the SA of frames on each bridge port
- *aging* out addresses that have not been observed for some time
- *flooding* frames when the location of the address is unknown

802.1D

802.1 discusses MAC bridges

The basic operational model, is called the **baggy pants** model

802.1D is also a **large** standard, defining

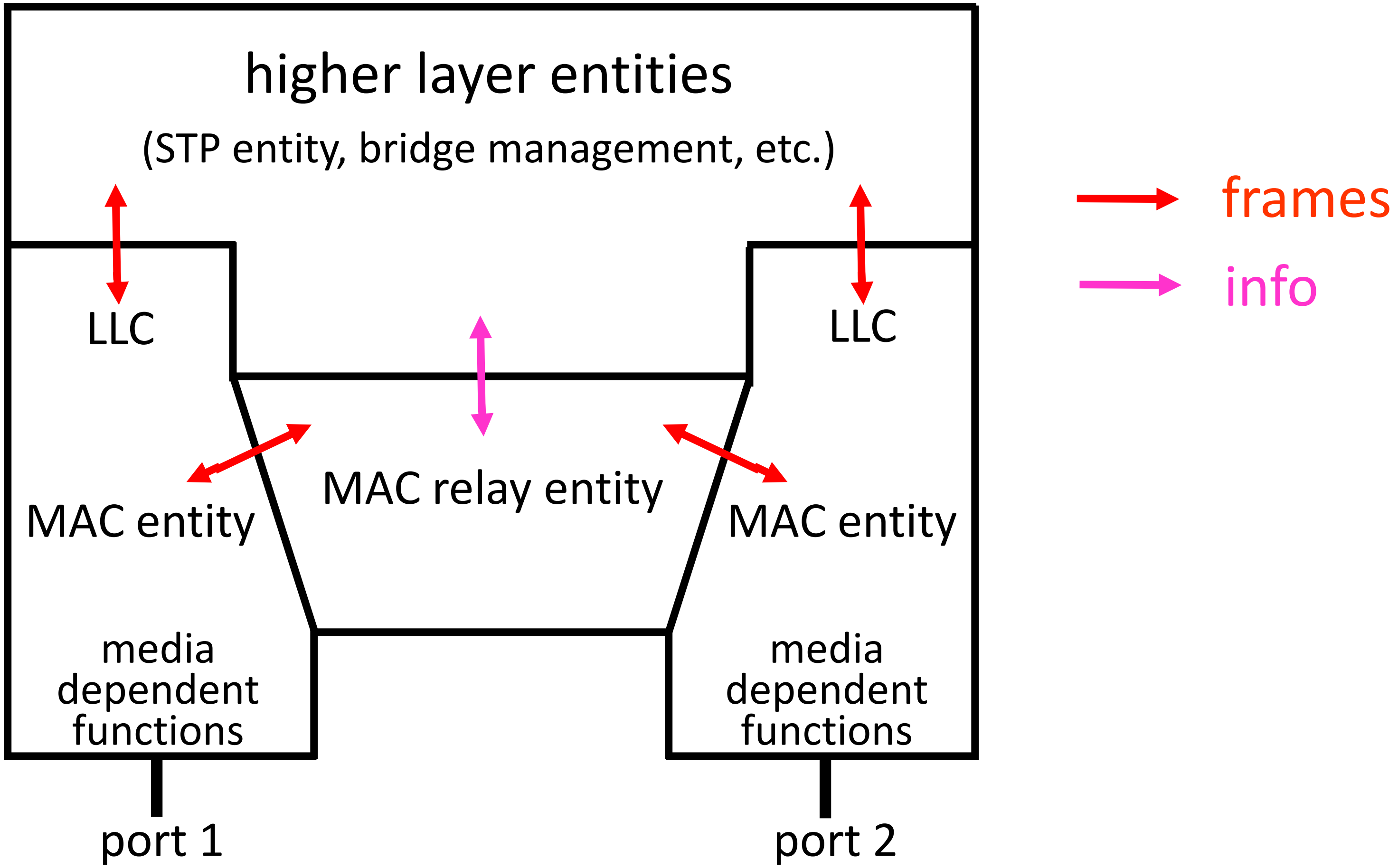
- bridge operation (learning, aging, STP, etc.)
- the architectural model of a bridge
- bridge Protocol and BPDUs
- GARP management protocols

802.1Q is a separate document on VLAN operation

New projects continue to expand scope

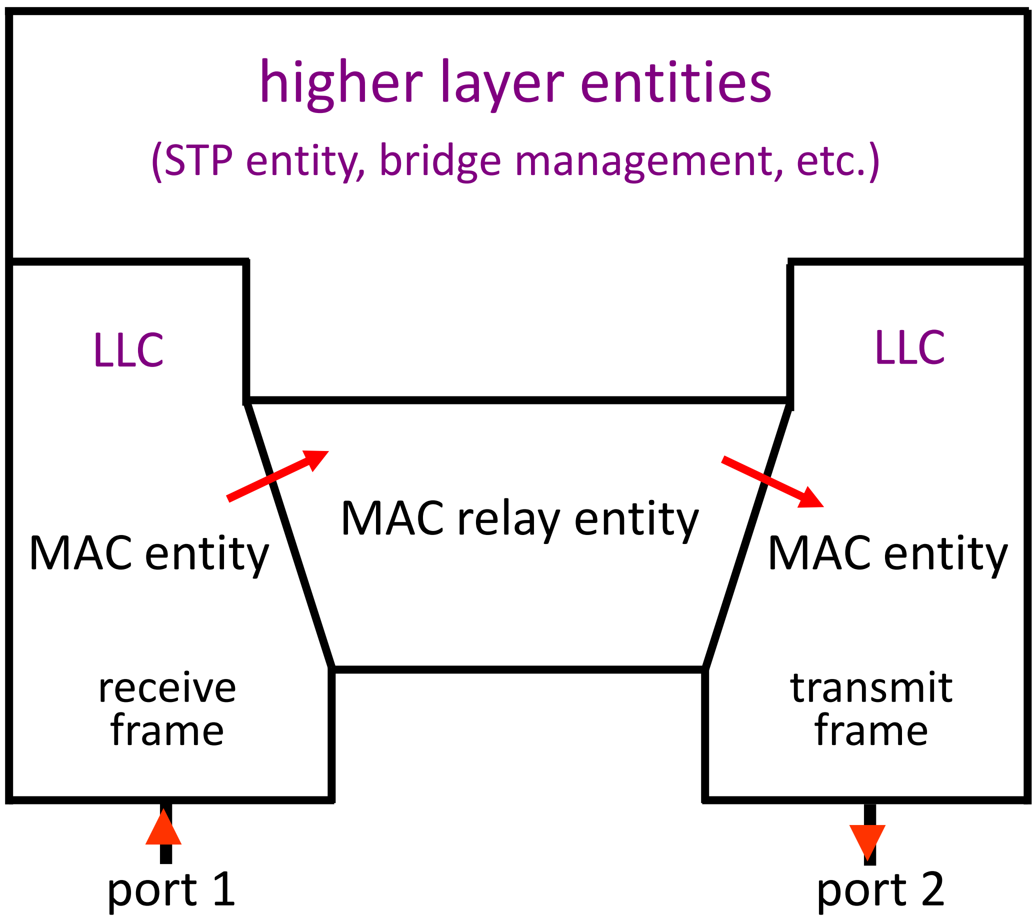
- 802.1ad – Q-in-Q
- 802.1ae – MACsec
- 802.1ag – OAM
- 802.1ah – MAC-in-MAC
- 802.1aj – 2-port MAC relay
- 802.1au – congestion notification

Baggy pants model



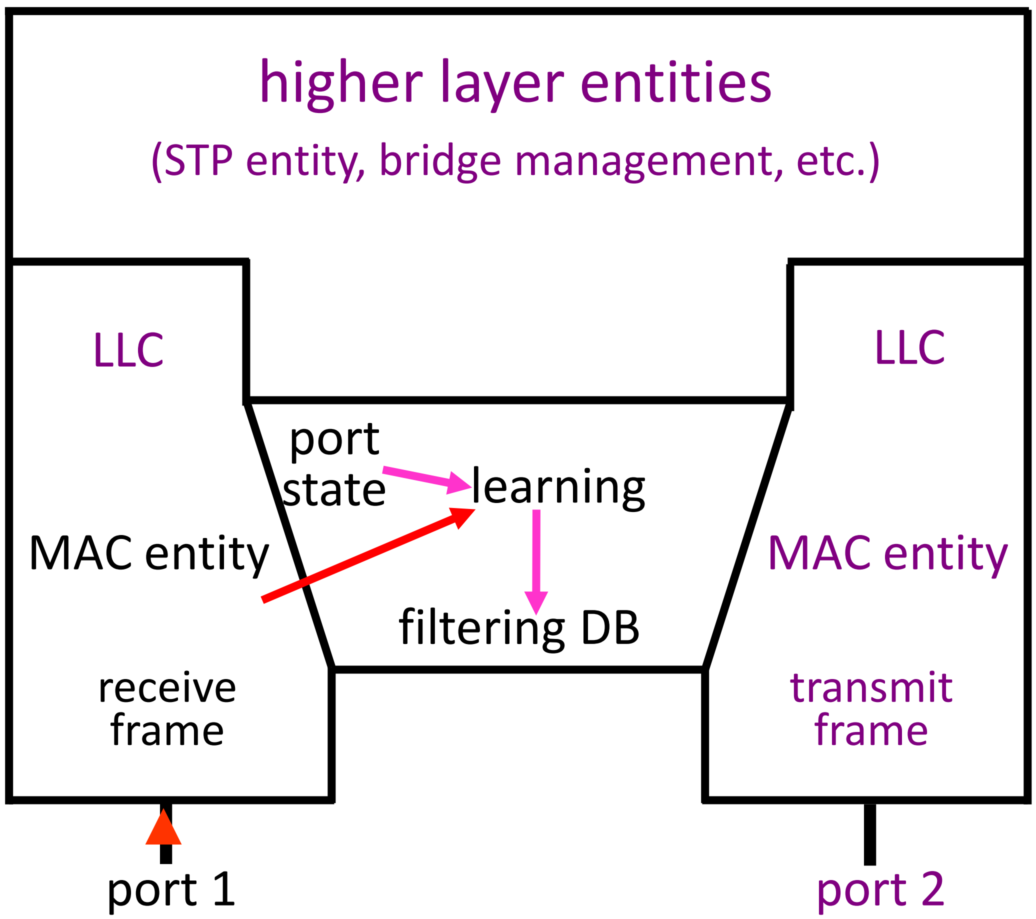
Note: a bridge must have at least 2 ports
here we depict exactly 2 ports

Baggy pants - filtering



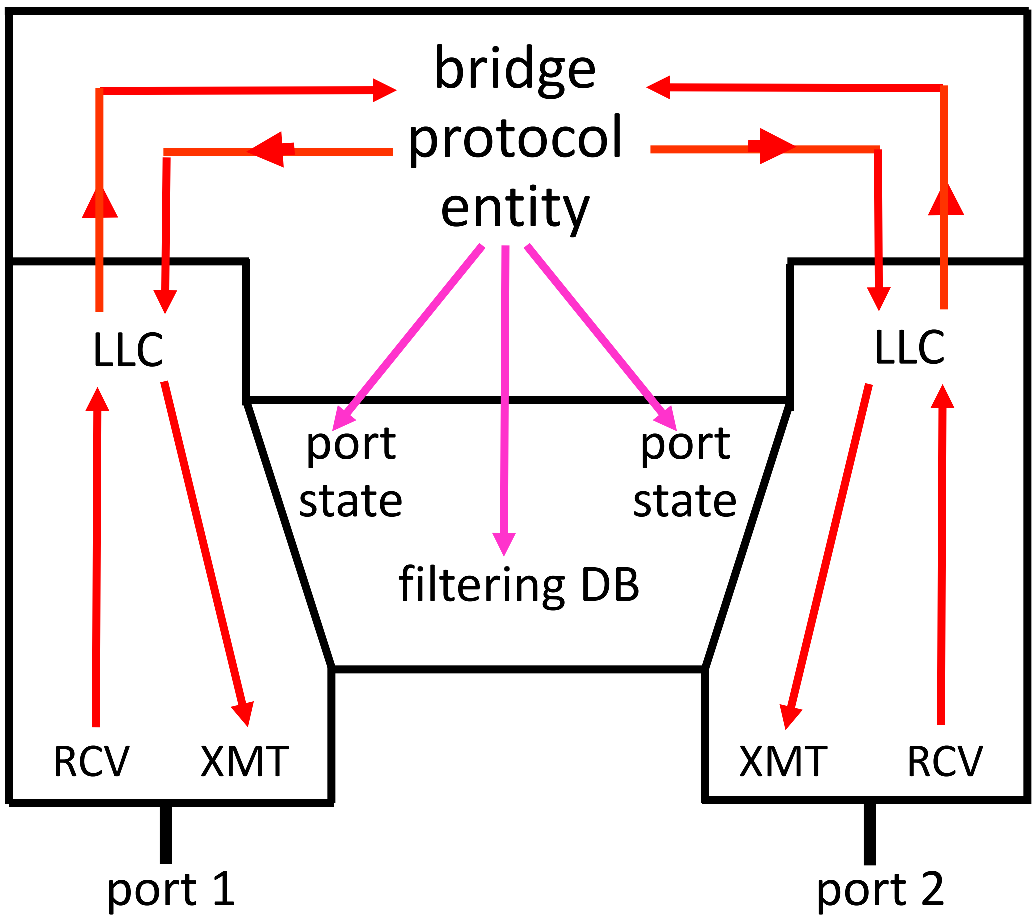
Note: relay entity passes frame to port 2 dependent on *port state* and *filtering database*

Baggy pants - learning



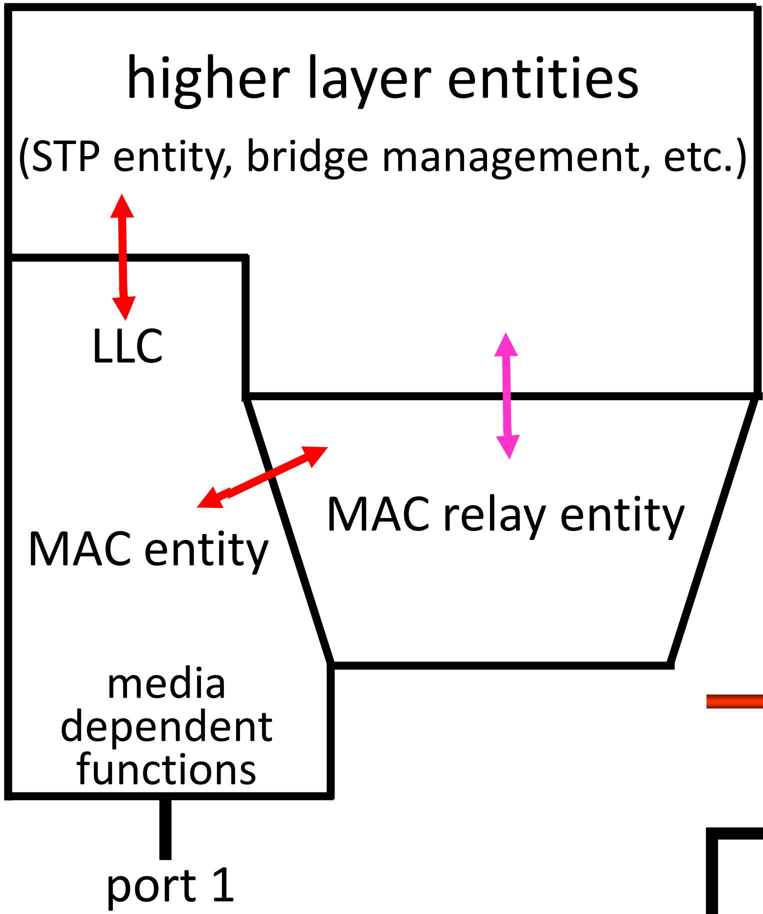
Note: we do not show forwarding of packet that *may* occur

Baggy pants - STP



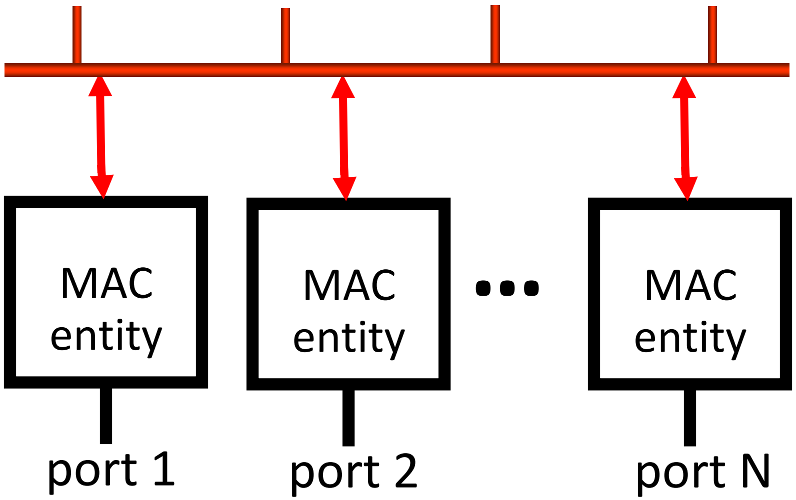
Note: PDUs are sent and received by the bridge protocol entity
bridge protocol entity updates filtering DB and port states

Extension to N ports



In the baggy pants diagram
port 1 and port 2 are identical
so it is enough to draw once

If there are many ports
the relay entity becomes
an internal LAN !



Layer 2 control protocols

The IEEE (and others) have defined Ethernet *control protocols* (L2CPs) :

protocol	DA	reference
STP/RSTP/MSTP	01-80-C2-00-00-00 802.2 LLC	802.1D §8,9 802.1D§17 802.1Q §13
PAUSE	01-80-C2-00-00-01 EtherType 88-08	802.3 §31B 802.3x
LACP/LAMP	01-80-C2-00-00-02 EtherType 88-09 Subtype 01 and 02	802.3 §43 (ex 802.3ad)
Link OAM	01-80-C2-00-00-02 EtherType 88-09 Subtype 03	802.3 §57 (ex 802.3ah)
ESMC	01-80-C2-00-00-02 EtherType 88-09 Subtype 10	G.8264
Port Authentication	01-80-C2-00-00-03 EtherType 888E	802.1X
E-LMI	01-80-C2-00-00-07 EtherType 88-EE	MEF-16
Provider MSTP	01-80-C2-00-00-08	802.1D § 802.1ad
Provider MMRP	01-80-C2-00-00-0D	802.1ak
LLDP	01-80-C2-00-00-0E EtherType 88-CC	802.1AB-2009
GARP (GMRP, GVRP)	Block 01-80-C2-00-00-20 through 01-80-C2-00-00-2F	802.1D §10, 11, 12

Note: we won't discuss autonegotiation as it is a *physical layer* protocol (uses link pulses)

Slow protocol frames

Slow protocols are slow – no more than 5 (or 10) frames per second
no more than 100 frames per link or ONU

Slow protocol frames must be untagged, and must be padded if needed

Slow protocols are for single links – they do not traverse bridges

There is a specific multicast address for multi-cast slow protocols
01-80-C2-00-00-02

There can not be more than 10 slow protocols

DA SA	8809	subtype		…

802-3 Annex 43B

Subtype:

- 1 is Link Aggregation Control Protocol (LACP)
- 2 is link aggregation marker protocol
- 3 is EFM OAM

STP

We want to connect up Ethernet bridges in every possible way

When doing this the network topology graph may have **loops**

Ethernet packets do not have **T**ime **T**o **L**ive fields
  thus if a packet enters a loop, it will continue looping forever

Eventually all the network bandwidth will go into the looping packets
  and the network will need to be shut down

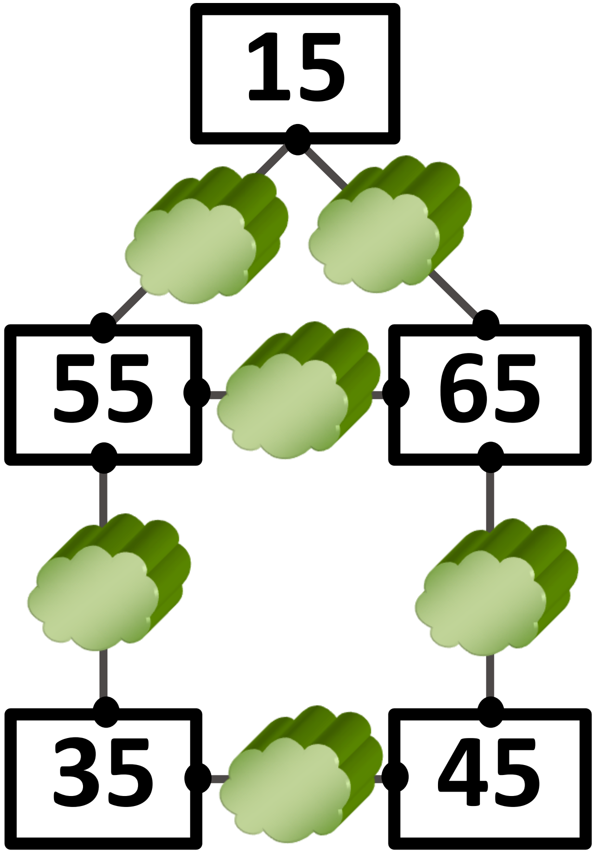
One solution is to ensure that there are no loops in the active topology

This means blocking links to obtain an active topology is a **tree**
  but that still **spans** the network

A protocol that accomplishes this is the **S**panning **T**ree **P**rotocol
  which involves bridges transmiting **B**ridge **P**rotocol **D**ata **U**nits

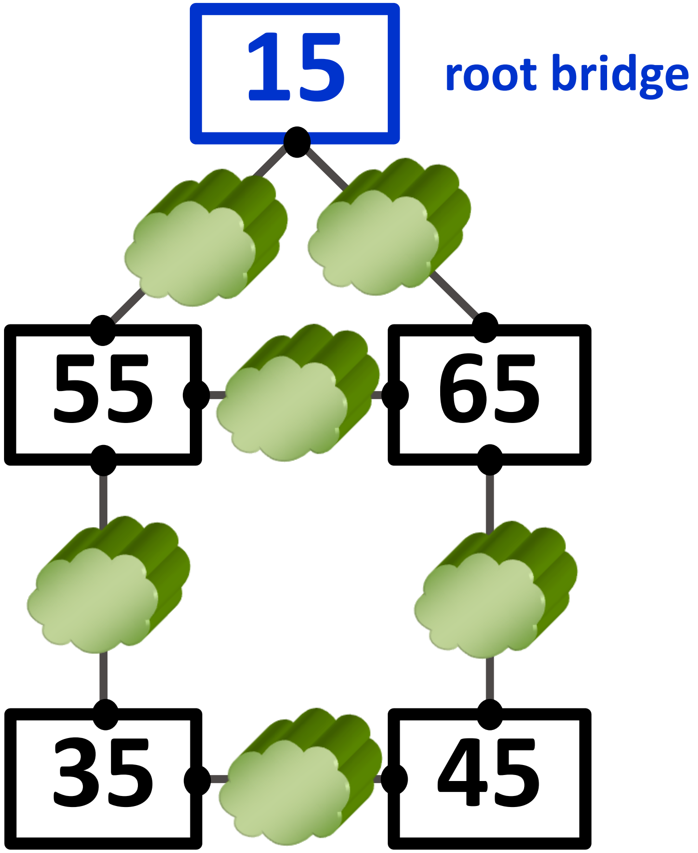
Here is a network with loops

The numbers are bridge IDs (where ID = priority and MAC address)



First select a root bridge

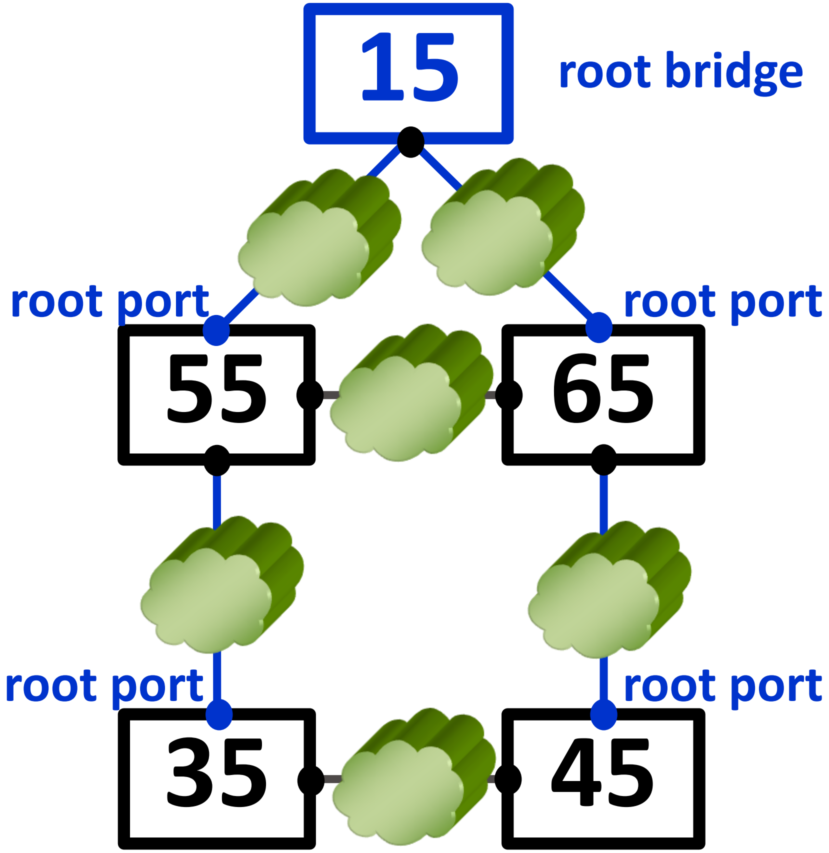
The bridge with minimal ID is designated the **root bridge**



Next identify root ports

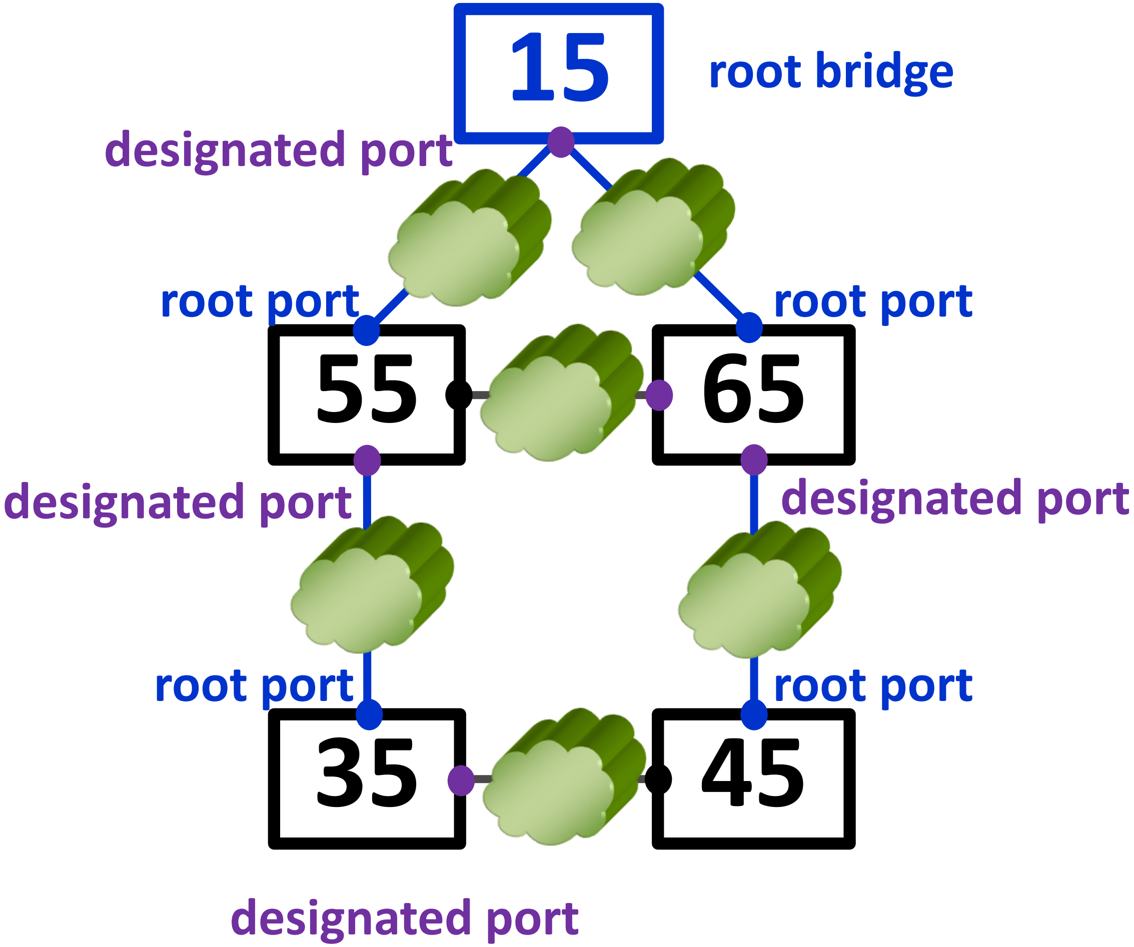
find the least cost path from the root bridge to every other bridge

ingress ports on this path are **Root Ports**



Next identify designated ports

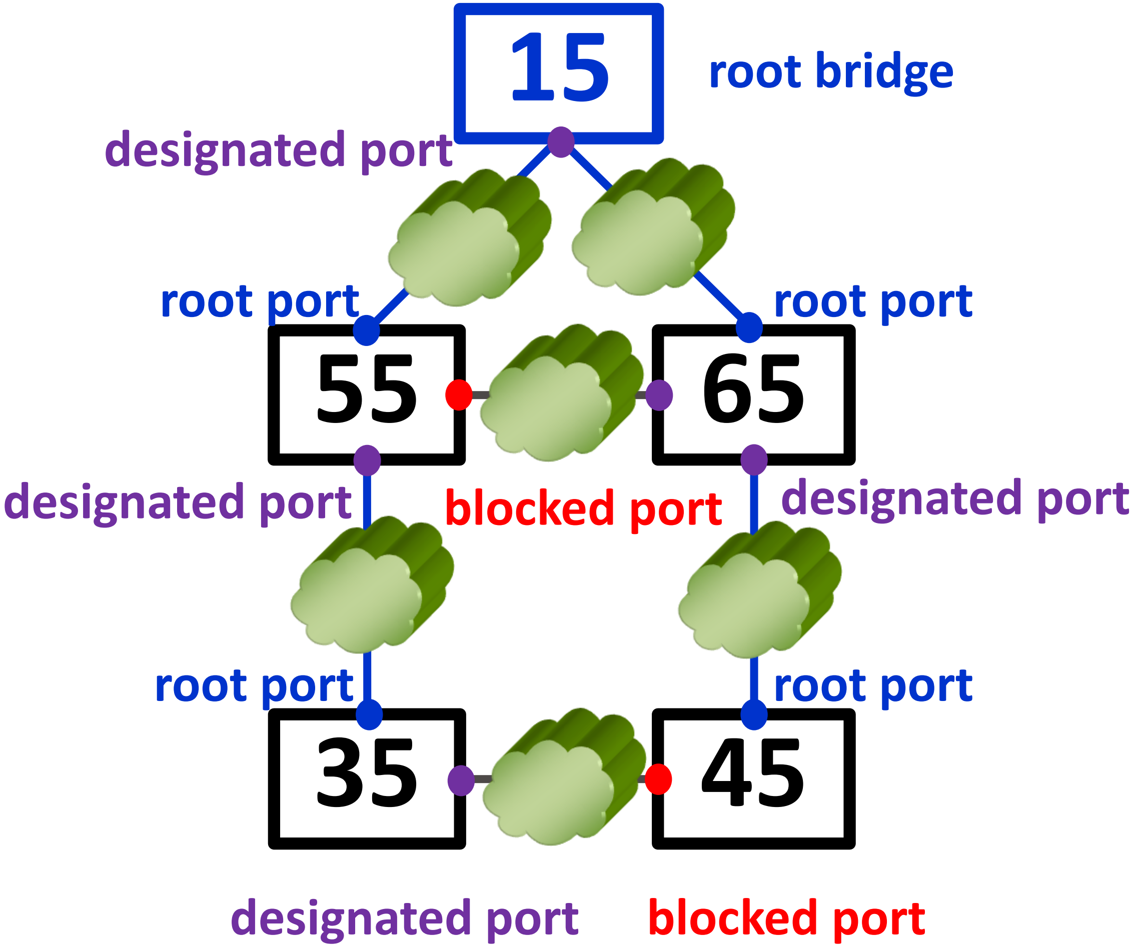
All other ports on a least cost path are **Designated Ports**



All other ports are blocked ports

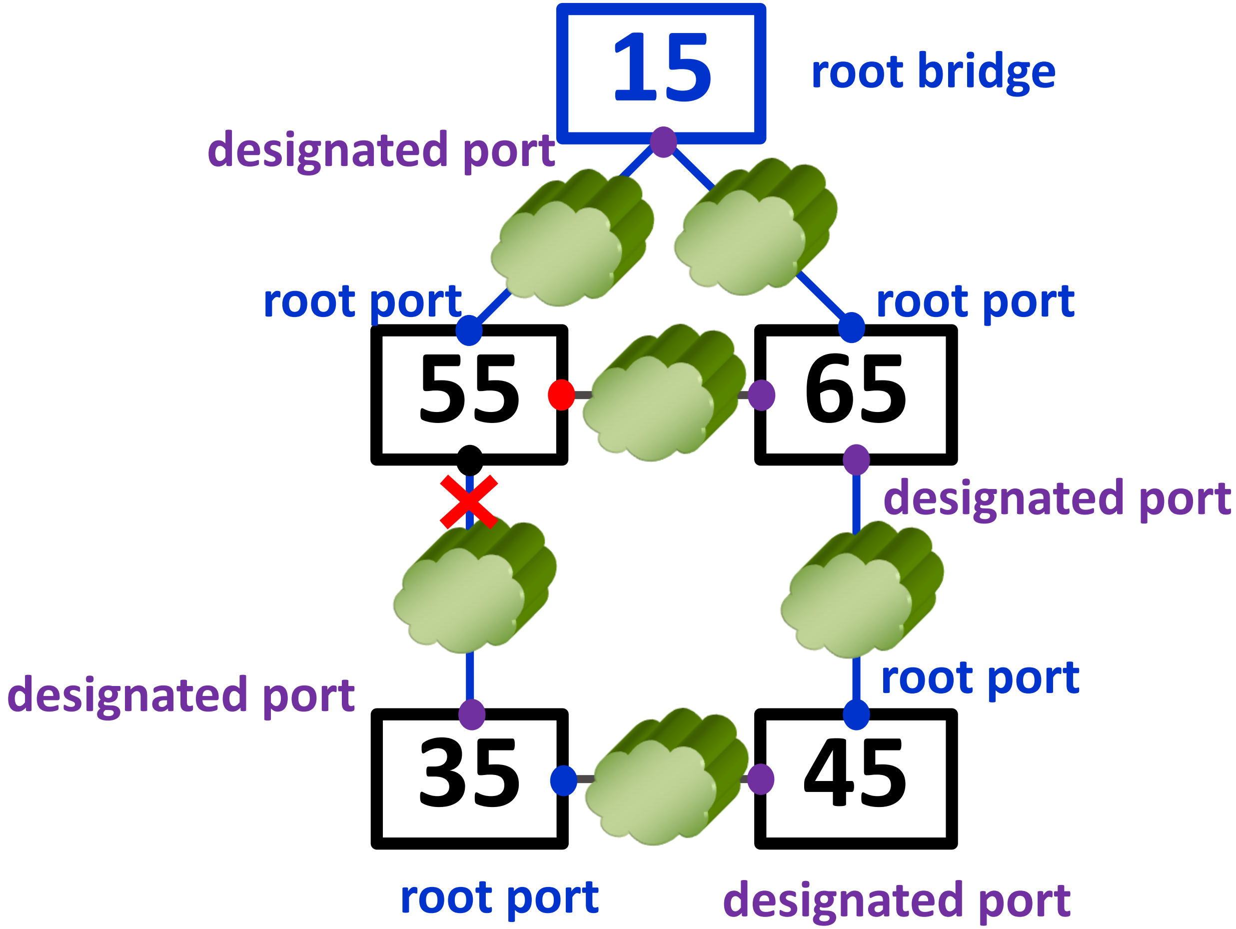
All other ports are **Blocked Ports**

packets are not forwarded over blocked ports



Reconvergence

If a link failure occurs, STP must reconverge to a new path



Algorhyme by Radia Perlman

I think that I shall never see
a graph more lovely than a tree.

A tree whose crucial property
is loop-free connectivity.

A tree that must be sure to span
so packet can reach every LAN.

First, the root must be selected.
by ID, it is elected.

Least-cost paths from root are traced.
in the tree, these paths are placed.

A mesh is made by folks like me,
then bridges find a spanning tree.

Multiple Spanning Tree Protocol 802.1s

Conventionally, all VLANs use the same spanning tree
(even if IVL switches use different FIDs)

so links blocked by STP will **never** carry **any** traffic

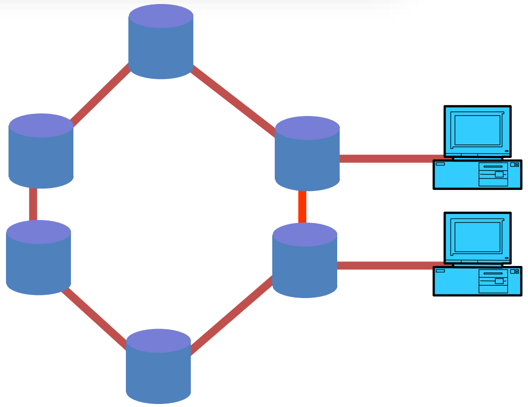
We **can** utilize these links
if different VLANs could use different spanning trees

Multiple Spanning Tree Protocol - 1998 amendment to 802.1Q
the protocol and algorithm are now in 802.1Q-2003 clauses 13 and 14

MSTP configures a separate spanning tree for each VLAN
blocks redundant links separately for each spanning tree

Another alternative is **TRILL**

TRILL



STP needs no configuration
but may make inefficient trees
  hosts that are actually close become far apart
  high bandwidth links may be blocked

We *could* use IP routing protocols
  but that requires allocating IP addresses, etc.

A new solution from Radia Perlman is
TRansparent **I**nterconnection of **L**ots of **L**inks

TRILL defines a combination of router and bridge called an Rbridge
  that runs a link state protocol (IS-IS)
  but based on Ethernet addresses

Rbridges have the advantages of both with the disadvantages of neither
- optimized paths
- but no configuration
- no IP layer

Algorhyme v2

I hope that we shall one day see
a graph more lovely than a tree.

A graph to boost efficiency
while still configuration-free.

A network where RBridges can
route packets to their target LAN.

The paths they find, to our elation,
are least cost paths to destination.

With packet hop counts we now see,
the network need not be loop-free.

RBridges work transparently.
without a common spanning tree.

Ray Perlner

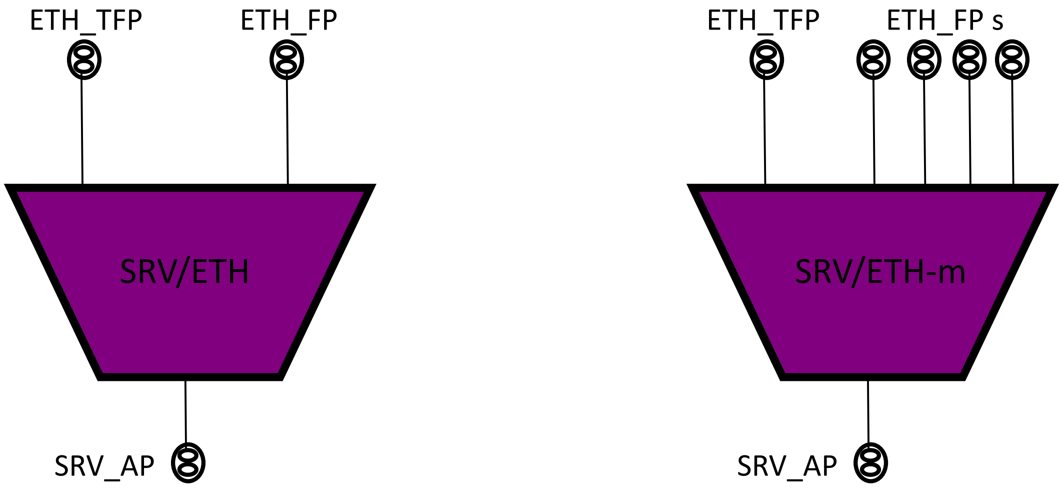
ETH layer network

ETH is a packet/frame-based layer network
it maintains client/server relationships with other networks
- networks that use Ethernet are Ethernet *clients*
- networks that Ethernet uses are Ethernet *servers*
- sometimes Ethernet ETY is the lowest server
  - i.e. there is no lower layer server network

ETH is usually *connectionless*
- but *connection-oriented* variants have been proposed (PBT, PVT, etc)

ETH is a relatively simple layer network
- it has no real forwarding operations
  - just filtering and topology pruning
- it has no real control plane
  - just STP, GARP, “slow protocol frames”, etc.
- until recently it had no OAM
  - but now there are two

ETH adaptations



The adaptation from ETH to the server layer (e.g. ETY) has

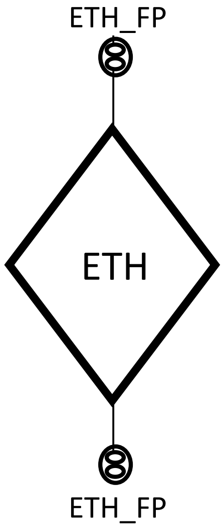
- 1 ETH Termination Flow Point responsible for DA, SA, P bits, OAM
- 1 (for ETH-m between 1 and 4094) ETH Flow Point(s)
  where the ETH CI enters
- 1 SRV Access Point (SRV can be ETY, but can be other server networks)

Traffic conditioning

G.8010 defines a new function (not in G.805/G.809)

The traffic conditioning function:

- inputs CI
- classifies traffic units according to configured rules
- meters traffic units within class to determine eligibility
- polices/shapes non-conformant traffic units
- outputs remaining traffic units as CI

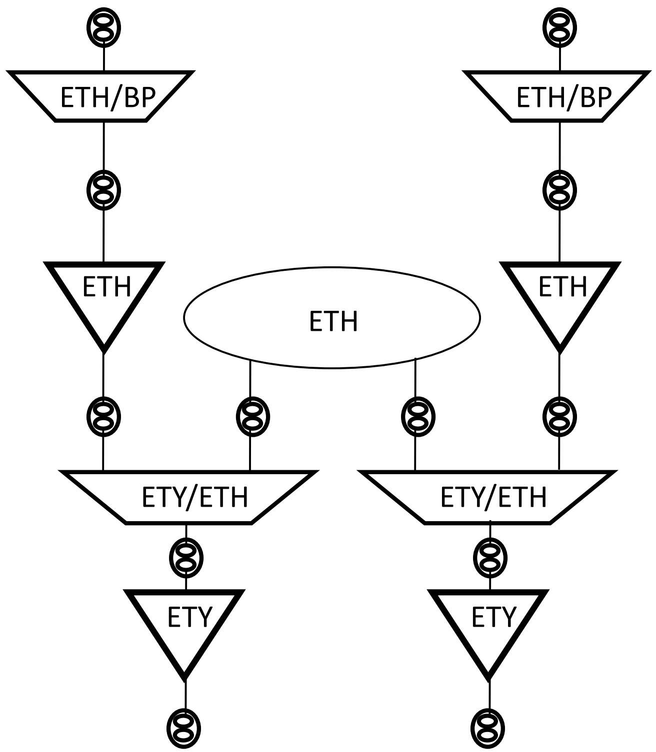


The TC function is obtained by expanding the ETH Flow Point

Translation to G.805

We can redraw the baggy pants model per G.805

(from G.8010 Appendix II) Note: drawn for CO case only



Flow control

When an Ethernet switch receives traffic faster than it can process it
it needs to tell its immediate neighbor(s) to slow down

On half-duplex links the *back pressure* can be employed
- overloaded device jams the shared media
  by sending preambles or idle frames
- detected by other devices as collisions causing senders to wait (CSMA/CD)

On full-duplex point-to-point links, *PAUSE frames* are sent

Since they are sent on a point-to-point link, the DA is unimportant,
and the standard multicast address 01-80-C2-00-00-01 is used
making the PAUSE frame easy to recognize

The PAUSE frame encodes the requested pause period
as a 2-byte unsigned integer representing units of 512 bit times

Handling QoS

Ethernet switches have **FIFO** buffers
on each port's input and output

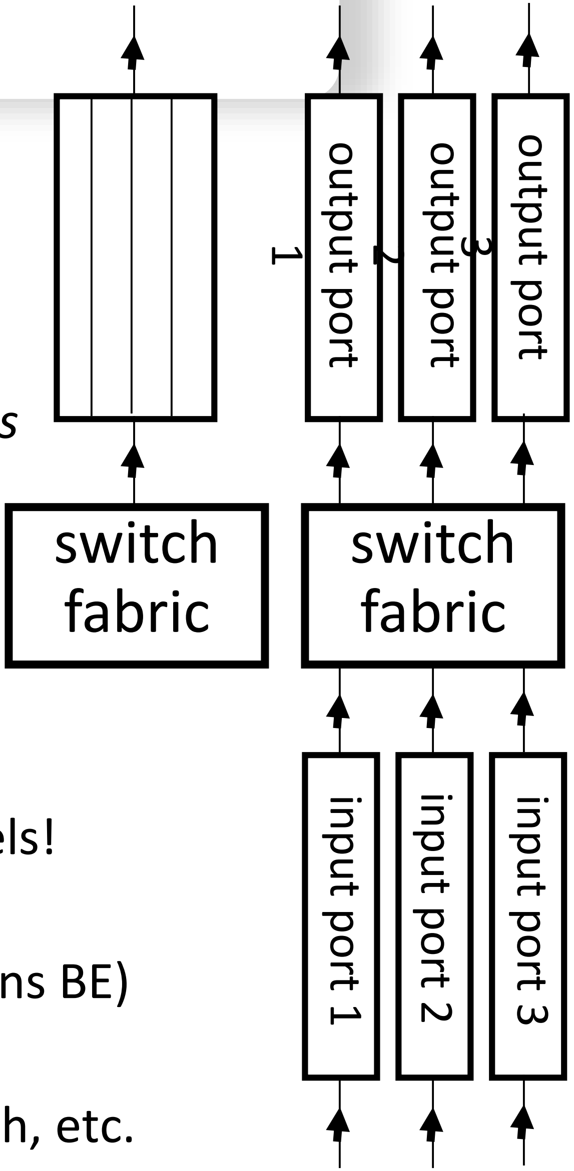
But prioritization is often needed

So output buffers may be divided into multiple *queues*
Outgoing frames put into queues of specified priority

We *could* base priority on input port
but then for the next switch to know the priority too
we would need to send to its appropriate port too
so the number of both input and output ports
would be multiplied by the number of priority levels!

A better way is to mark the frames
using the 3 bit **P**riority **C**ode **P**oint field (PCP=0 means BE)

User priority levels map to traffic classes (CoS)
indicating drop probability, latency across the switch, etc.
but there are no BW/latency/jitter guarantees
802.1Q recommends specific mappings from PCP to traffic class



Research topics

- CSMA/CD and WiFi/EPON (broadcast DS, TDMA US)
  are 2 multiple access strategies used in Ethernet.
  What other mechanisms can be used?
  Which is best for specific scenarios?
- Low rate Ethernet was quiet when there were no packets to send but high rate Ethernet continually transmits IDLE patterns, because
  - difficulty in high accuracy synchronization with preambles
  - lengthy laser turn-on times.

  How can Ethernet be made more energy-efficient?
- How can we make a zero-configuration but secure Ethernet ?