

Università di Genova  
Facoltà di Ingegneria

---

Telematica  
**4. Internetworking L2/L3**

Prof. Raffaele Bolla

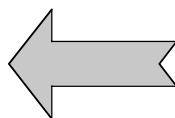


Telematica

**Internetworking L2/L3**

---

● **Bridge**



- Switch L2
- Switch L3
- Virtual LAN (VLAN)

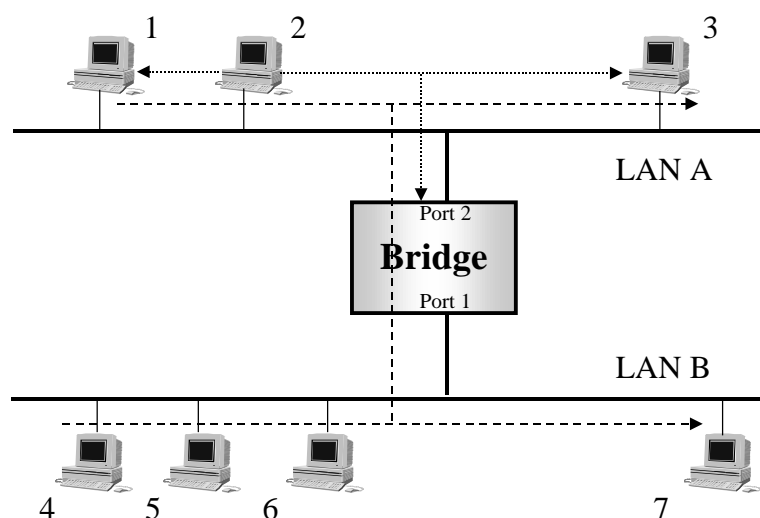
4.2

## Bridge

- Sia l'esistenza (almeno nella fase iniziale) di tecnologie diverse, la limitatezza delle prestazioni in caso di molti utenti e/o elevato traffico e la ridotta estensione geografica (specialmente nel caso di velocità a 100 o 1000 Mb/s) ha dato origine a degli apparati per l'interconnessione di LAN a livello 2, che prendono il nome di **Bridge**.
- L'interconnessione di LAN tramite bridge non ha quindi il solo obiettivo di far "parlare" calcolatori residenti su LAN differenti ma anche quello di permettere la creazione di LAN estese o BLAN (*Broad-LAN*) composte da più *shared* LAN realizzate con la stessa tecnologia e fra loro interconnesse.

4.3

## Bridge



4.4

## Bridge

---

- Il Bridge ha quattro funzioni principali:

- La ricezione dei pacchetti
- L'eventuale traduzione da un formato di liv. 2 ad un altro (es. da 802.3 a 802.5).
- Il filtraggio dei pacchetti sulla base dell'indirizzo tenendo conto:
  - » Della posizione della destinazione
  - » Di indicazione inclusive od esclusive del gestore (che possono anche riguardare l'ind. della sorgente, il tipo di protocollo)
- La ritrasmissione dei pacchetti.

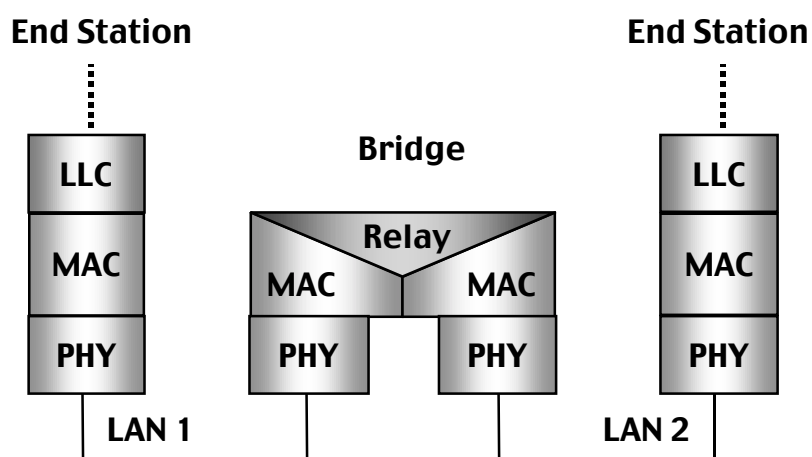
- In sostanza il bridge:

- mantiene separati i domini di collisione (IEEE 802.3)
- confina il traffico locale sul dominio di collisione (LAN) di pertinenza
- Converte i formati fra MAC di tipo diverso.

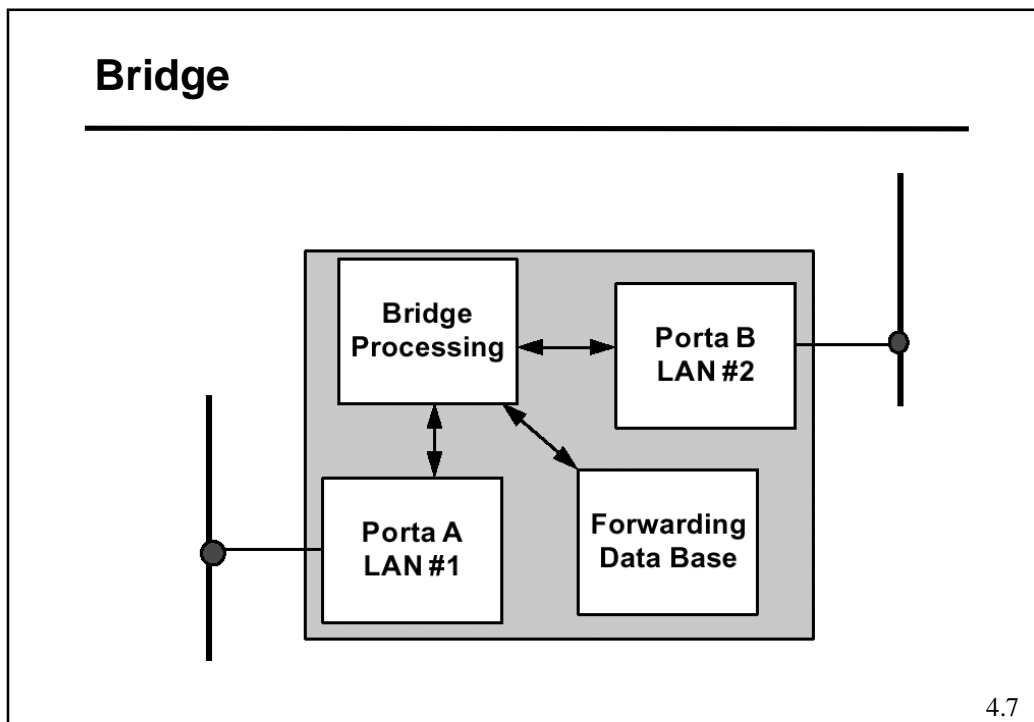
4.5

## Bridge

---



4.6

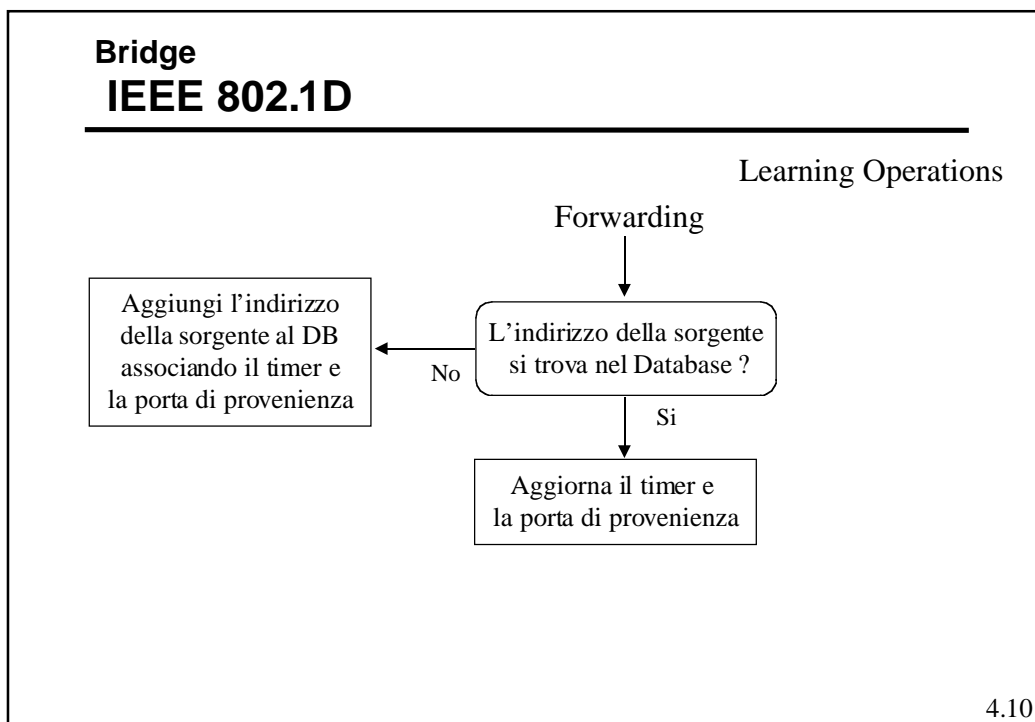
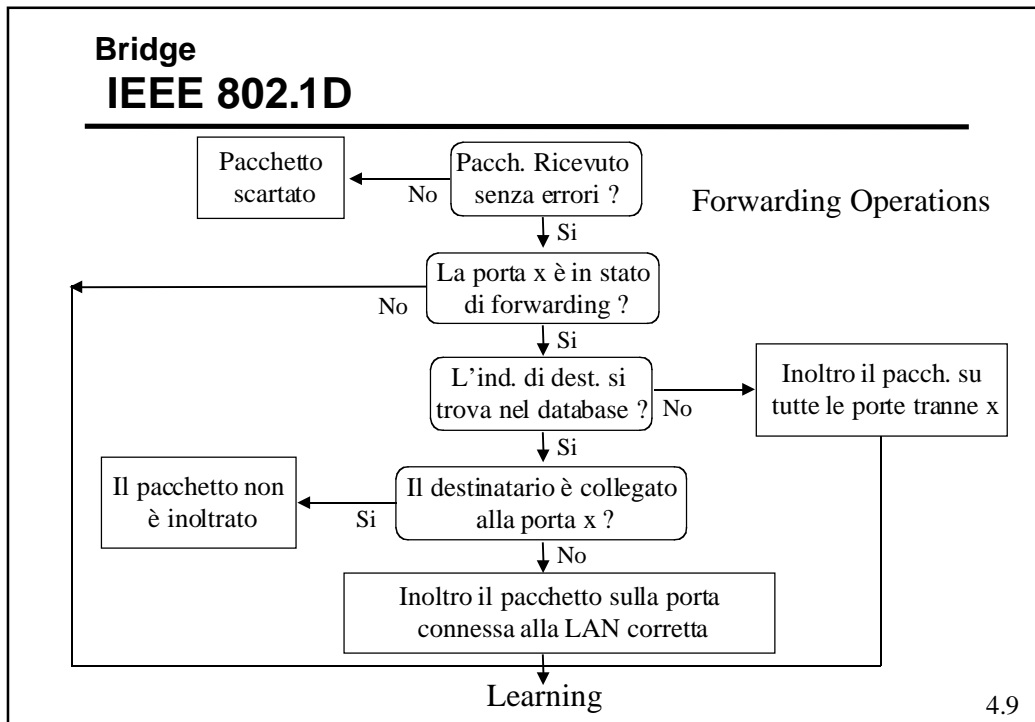


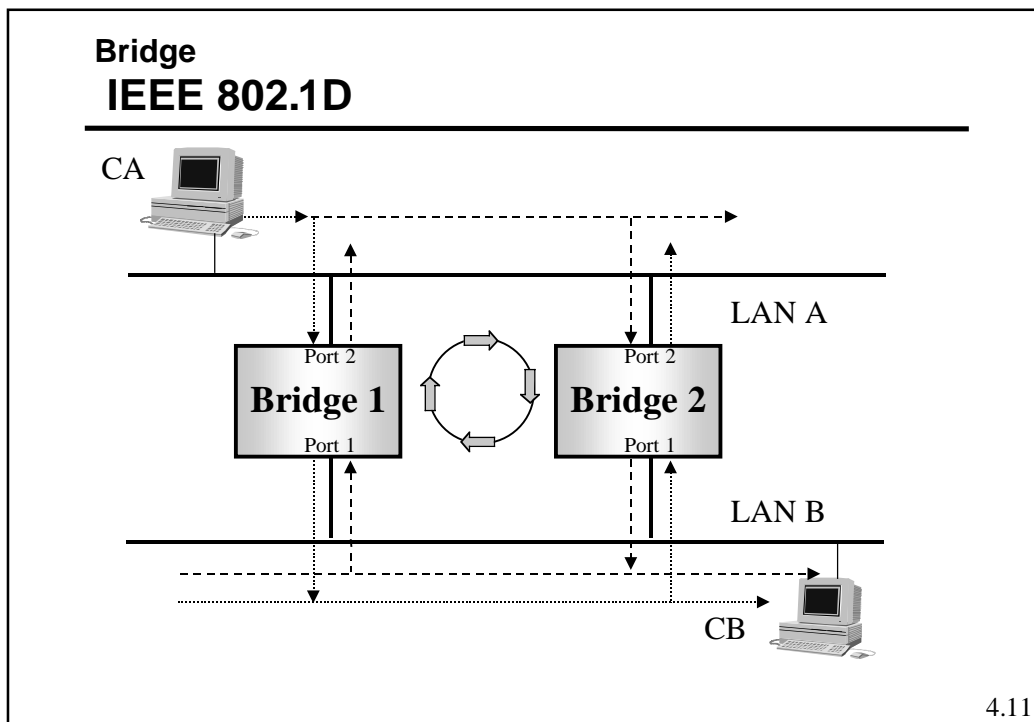
### Bridge

#### IEEE 802.1D - Transparent Bridge

- La IEEE ha standardizzato i bridge con lo standard 802.1D. Tali bridge sono noti anche con il nome di ***Transparent spanning-tree bridge***
- Fra le caratteristiche significative di tali tipi di bridge si possono citare
  - non richiedono una vera configurazione del database di filtraggio;
  - sono “trasparenti” alle stazioni collegate alla LAN.

4.8





**Bridge  
IEEE 802.1D**

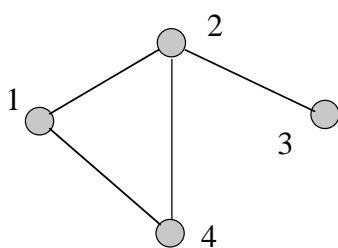
- Essendo un dispositivo di livello 2, il bridge ha capacità di instradamento ridotte, ed in particolare non è in grado di gestire la presenza di cicli (*loop*) che si creano quando su di una LAN è connesso più di un bridge.
- IEEE 802.1D prevede quindi un meccanismo per di “selezione” delle porte attive che elimina topologicamente gli eventuali *loop* realizzando fra i bridge uno ***Spanning tree***.

4.12

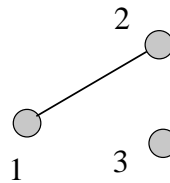
### Bridge - Algoritmi di instradamento

## Grafo

- Un **grafo** (*graph*)  $G = \{N, A\}$  è un insieme non vuoto  $N$  di nodi e un insieme  $A$  di coppie di nodi appartenenti ad  $N$  detti **archi** (*arc*).



$N = \{1, 2, 3, 4\}$   
 $A = \{(1, 2), (1, 4), (2, 3), (2, 4)\}$



$N = \{1, 2, 3\}$   
 $A = \{(1, 2)\}$



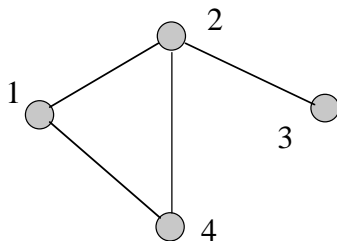
$N = \{1\}$   
 $A = \{ \}$

4.13

### Bridge - Algoritmi di instradamento

## Cammino e percorso

- Un **cammino** (*walk*) è una sequenza di nodi  $(n_1, n_2, \dots, n_L)$  tali che  $(n_l, n_{l+1})$ ,  $l=1, \dots, L-1$  sono archi di  $G$ .
- Un cammino  $(n_1, \dots, n_L)$  con  $L \geq 3$  e  $n_1 = n_L$  è detto **ciclo**.
- Un cammino senza nodi ripetuti è detto **percorso** (*path*).



Es. di cammini:

$(1, 4, 2, 1, 4, 1)$ ,  $(2, 3, 2)$ ,  $(1, 4, 2)$

Es. percorsi (path):

$(1, 4, 2, 3)$ ,  $(1, 2, 3)$

4.14

### Bridge - Algoritmi di instradamento

## Alberi

---

- Un grafo è detto **connesso** se per ogni nodo  $i$  esiste un percorso ( $i=n_1, \dots, n_L=j$ ) verso ogni nodo  $j$ .
- **Lemma**  
Sia  $G$  un grafo connesso e sia  $S$  un sottoinsieme di  $N$ , allora esiste almeno un arco  $(i, j)$  con  $i \in S$  e  $j \notin S$ .
- $G' = \{N', A'\}$  è un **sottografo** (*subgraph*) di  $G$  se  $N' \subset N$  e  $A' \subset A$ .

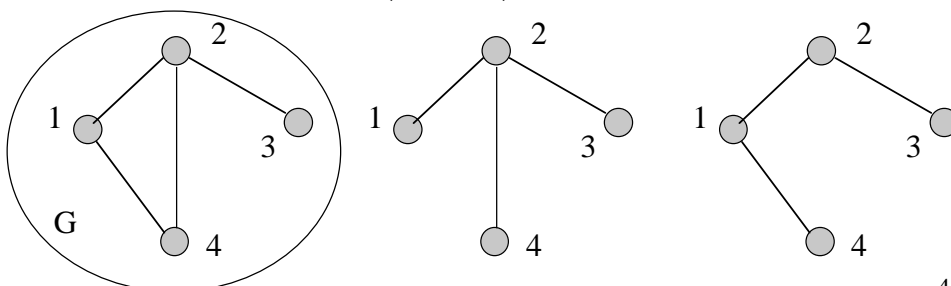
4.15

### Bridge - Algoritmi di instradamento

## Alberi

---

- Un **albero** (*tree*) è un grafo connesso che non contiene cicli.
- Uno **spanning tree** di un grafo  $G$  è un sottografo di  $G$  che è un albero e contiene tutti i nodi di  $G$  ( $N' = N$ ).



4.16



### Bridge - Algoritmi di instradamento

#### Spanning Tree

- Si può costruire uno *spanning tree* con il seguente algoritmo:
  - Sia  $N = \{n\}$ , con  $n$  un nodo arbitrario di  $N$ , e  $A'$  un insieme vuoto.
  - Se  $N' = N$ , allora  $G' = (N', A')$  è uno *spanning tree*, altrimenti si esegua il passo successivo
  - Sia  $(i, j) \in A$  un arco con  $i \in N'$ ,  $j \in N - N'$ , si ponga

$$N' = N' \cup \{j\}$$

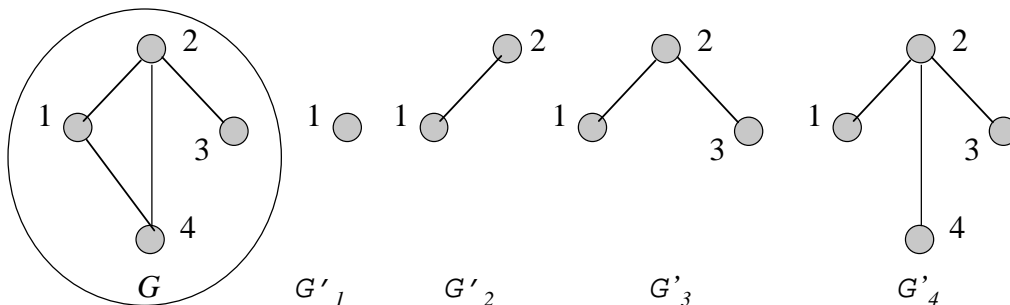
$$A' = A' \cup \{(i, j)\}$$

e si torni al passo precedente

4.17

### Bridge - Algoritmi di instradamento

#### Spanning Tree



- Sia  $G$  un grafo connesso con  $N$  nodi e  $A$  archi allora
  - $G$  contiene almeno uno *spanning tree*
  - $A \geq N-1$
  - $G$  è un albero solo se  $A = N-1$ ;

4.18

Bridge - Algoritmi di instradamento

**Minimum weight Spanning Tree (MST)**

---

- Se si associa ad ogni arco  $(i, j)$  un peso (*weight*)  $w_{ij}$  che riassume il “costo” di una trasmissione lungo di esso, la somma dei costi degli archi di un SP rappresenta il costo del *broadcast* di un messaggio sull’albero.
- Allora potrebbe essere conveniente cercare lo *spanning tree* la cui somma dei pesi degli archi sia minima, ossia il ***Minimum weight Spanning Tree (MST)***.

4.19

Bridge - Algoritmi di instradamento

**Minimum weight Spanning Tree (MST)**

---

- Si chiami frammento ogni sotto-albero (sub-tree) di un MST.

**Proposizione**

Dato un frammento  $F$  ed essendo  $\alpha = (i, j)$  l’arco con peso minimo per cui  $i$  appartiene ad  $F$  e  $j$  no, allora  $F$  esteso con  $(i, j)$  e  $j$  è a sua volta un frammento.

- Questa proposizione può essere usata per realizzare algoritmi che trovino un MST.

4.20

### Bridge - Algoritmi di instradamento

#### Grafi orientati

---

- Consideramo ora un Grafo Orientato (*directed graph*), i cui archi sono orientati ossia sono coppie ordinate di nodi.
- Tutte le definizioni date per il grafo, (ossia cammino, ciclo, percorso, albero) possono essere ripetute per il grafo orientato.
- Si associ ad ogni arco  $(i, j)$  un valore  $d_{ij}$  genericamente indicato come distanza. Dato un percorso diretto (*direct path*)  $p = (n_1, n_2, \dots, n_L)$  fra  $n_1$  e  $n_L$ , definiamo come lunghezza del percorso

$$D = \sum_{r=1}^{L-1} d_{n_r n_{r+1}}$$

- Si osservi che se  $d_{ij}=1$ , la lunghezza del percorso corrisponde al numero di archi che lo compone.

4.21

### Bridge - Algoritmi di instradamento

#### Shortest path

---

- Il problema del “percorso minimo” o *shortest path* è quello di trovare il percorso  $p$  fra  $i$  e  $j$  tale che  $D$  sia minimo.
- Esistono diversi metodi per risolvere questo problema, uno di questi prende il nome di algoritmo di Bellman-Ford.
- Tale algoritmo fissata una destinazione, trova il percorso minimo da ogni nodo a tale destinazione nell'ipotesi non ci siano distanze negative ( $d_{ij} \geq 0$ ).

4.22

### Bridge - Algoritmi di instradamento

## Shortest path - Bellman Ford

---

- Definendo

- 1 come il nodo destinazione
- $d_{ij} = \infty$  se  $(i, j)$  non è un arco (i nodi  $i$  e  $j$  non sono direttamente connessi),
- $D_i^h$  come la lunghezza del percorso più corto fra il nodo  $i$  ed il nodo 1, contenente al massimo  $h$  archi
- $D_i^h = \infty$  se tale percorso non esiste
- $D_1^h = 0, \forall h$ , per convenzione
- $D_i^0 = \infty$  per tutti gli  $i \neq 1$

- l'iterazione dell'algoritmo di Bellman-Ford è

$$D_i^{h+1} = \min_j \{d_{ij} + D_j^h\} \quad \text{per ogni } i \neq 1$$

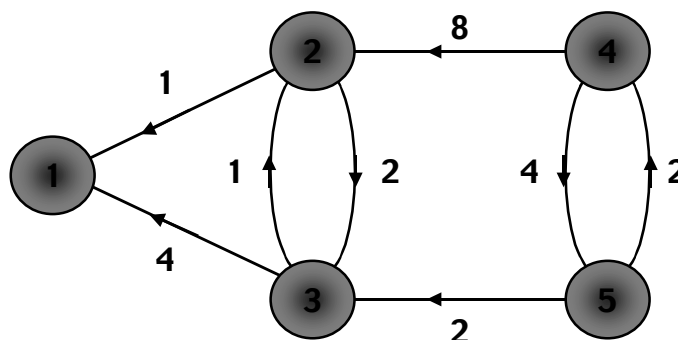
- L'algoritmo ha termine quando  $D_i^{h+1} = D_i^h \forall i$

4.23

### Bridge - Algoritmi di instradamento

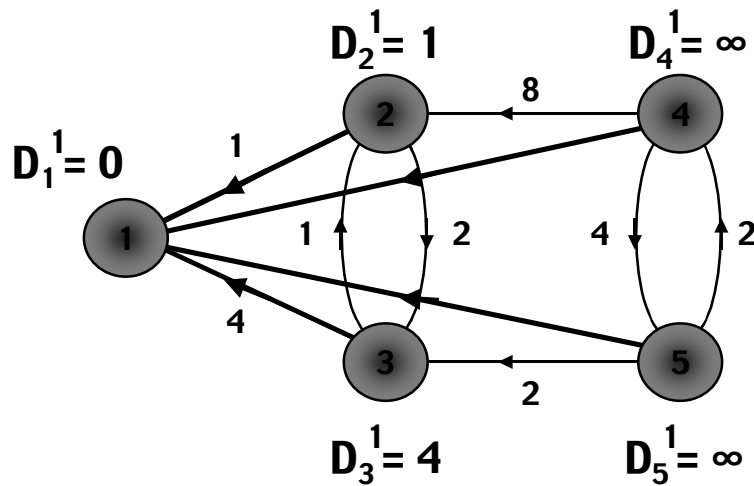
## Bellman-Ford (Es.)

---



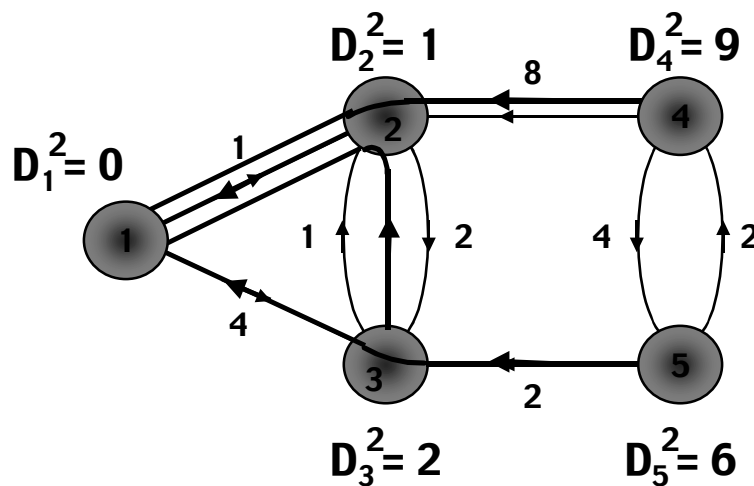
4.24

Bridge - Algoritmi di instradamento  
**Bellman-Ford (Es.)**



4.25

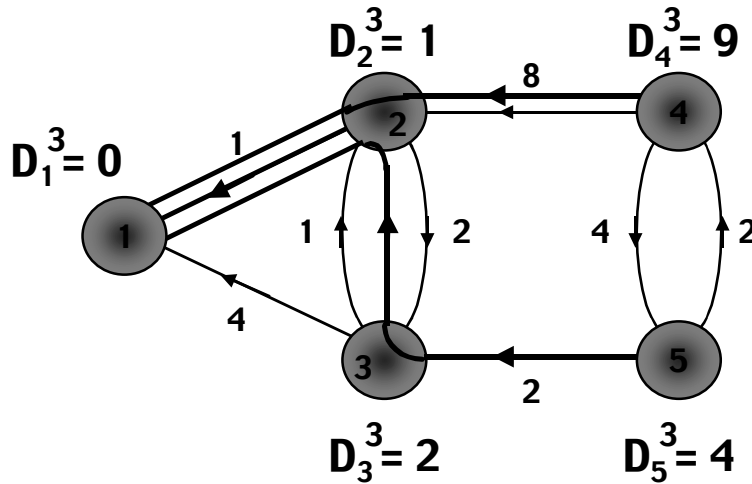
Bridge - Algoritmi di instradamento  
**Bellman-Ford (Es.)**



4.26

Bridge - Algoritmi di instradamento  
**Bellman-Ford (Es.)**

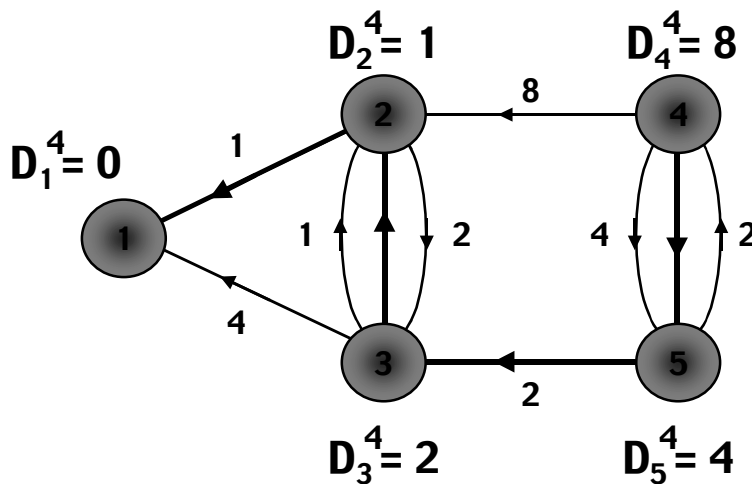
---



4.27

Bridge - Algoritmi di instradamento  
**Bellman-Ford (Es.)**

---



4.28

### Bridge - Algoritmi di instradamento

## Shortest path Spanning Tree

---

- L'applicazione equazione di Bellman seleziona un solo arco uscente da ogni nodo  $i$  (eccetto la dest. 1), cioè quello per cui la somma  $d_{ij} + D_j^h$  è minima.
- Quindi in sostanza gli archi scelti dall'algoritmo ed i nodi formano uno *spanning tree* perché:
  - Comprendono tutti i nodi per definizione
  - non possono formare cicli essendo le lunghezze positive
- Tale spanning tree viene chiamato *Shortest Path Spanning Tree* (SPST) ed il nodo destinazione è chiamato *root* (radice).

4.29

### Bridge - Algoritmi di instradamento

## Shortest path Spanning Tree

---

- Un grafo non orientato può essere rappresentato come un grafo orientato a cui ad ogni arco non orientato corrispondono due archi, uno per direzione, con egual peso.
- In generale però il MSP e il SPSP sono diversi:
  - Il MSP minimizza realmente il costo di un broadcasting;
  - Il SPSP invece minimizza il costo delle comunicazioni fra un qualunque nodo e la *root*.

4.30

### Bridge - IEEE 802.1D

## Spanning Tree

---

- Per risolvere il problema dei loop, lo standard 802.1D individua uno *spanning tree* fra i bridge e le LAN ed attiva le sole porte che appartengono a tale albero.
- Lo spanning tree è aggiornato dinamicamente nel tempo per adattarlo ad eventuali cambiamenti topologici (guasti o modifiche nella rete).
- L'algoritmo opera prima eleggendo un nodo di *root* e quindi applicando una particolare versione distribuita dell'algoritmo di Bellman-Ford con il nodo di *root* preso come destinazione.
- Lo spanning tree risultante è quindi uno *shortest path* SPST e non un minimum weight MST.

4.31

### Bridge - IEEE 802.1D

## Spanning Tree

---

- Ad ogni bridge ha un indirizzo di livello 2 (6 byte) ed è assegnato un numero (2 byte, 0 - 65535, di default 32768): la concatenazione fra i 2 byte del numero e i 6 byte dell'indirizzo determinano la priorità del bridge.
- Ad ogni porta di ogni bridge, identificata da un ID, viene a sua volta assegnata (da management) una priorità di un byte; essa rappresenta il "costo" del link in uscita e quindi valori più bassi indicano una priorità maggiore.

4.32



### Bridge - IEEE 802.1D Spanning Tree

---

- L'algoritmo nel suo complesso opera come segue
  - Determinazione del *root bridge*
  - Determinazione della *root port* (porta più "vicina" al *root bridge*) sugli altri *bridge*
  - Determinazione delle *designated port* su ogni LAN
- Tutte le interazioni fra i bridge avvengono tramite frame detti Bridge-PDU (BPDU).

4.33

### Bridge - IEEE 802.1D Spanning Tree

---

- Come *root bridge* è designato quello con una priorità più alta.
- Inizialmente ogni bridge assume di essere *root* e invia su ogni uscita BPDU con la propria priorità.
- Quando un bridge riceve una BPDU con priorità più alta (valore più basso), assume che il mittente sia la *root* e smette di inviare BPDU.
- Dopo un certo tempo la BPDU sono inviate solo dal *root bridge*.

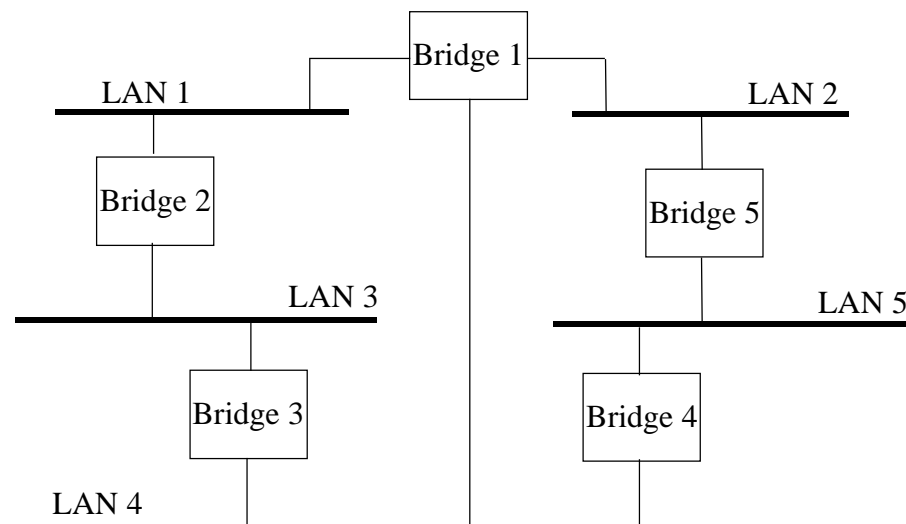
4.34

### Bridge - IEEE 802.1D Spanning Tree

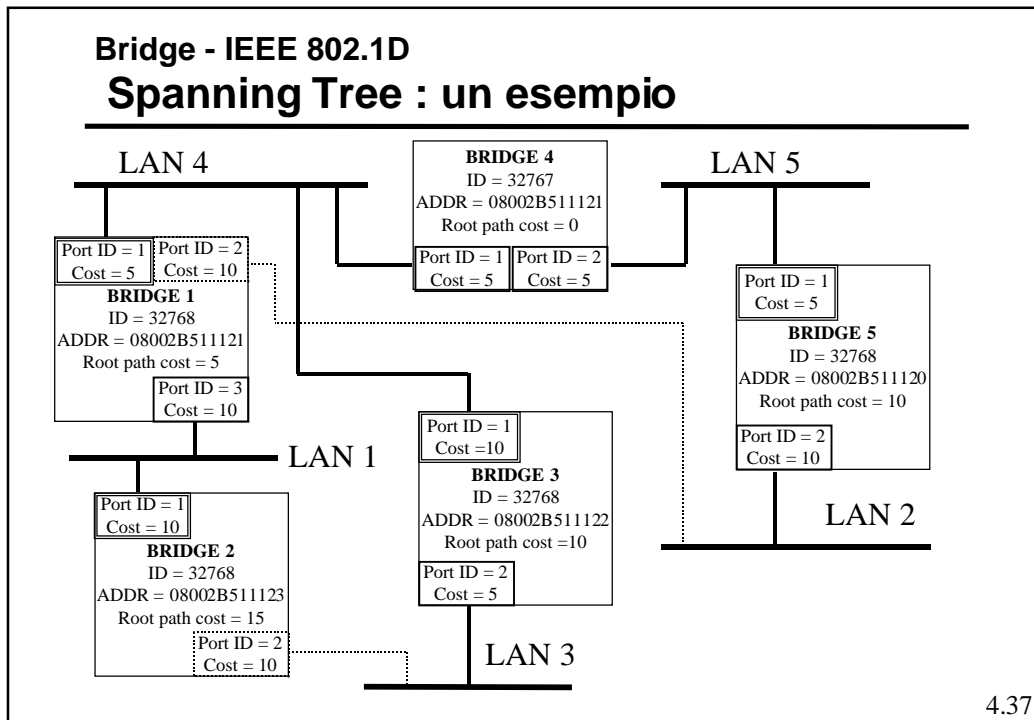
- Ogni BPDU contiene un *root path cost*, che la *root* inizialmente fissa a zero; ogni bridge che riceve la BPDU sceglie come *root port* quella da cui riceve la BPDU con il *root path cost* più basso e propaga solo quella.
- Prima di propagarla aggiorna il *path cost* aggiungendo il costo relativo alla porta da cui l'ha ricevuta.
- Le porte da cui riceve BPDU con un *root path cost* più basso del proprio vengono disabilitate.

4.35

### Bridge - IEEE 802.1D Spanning Tree: un esempio



4.36

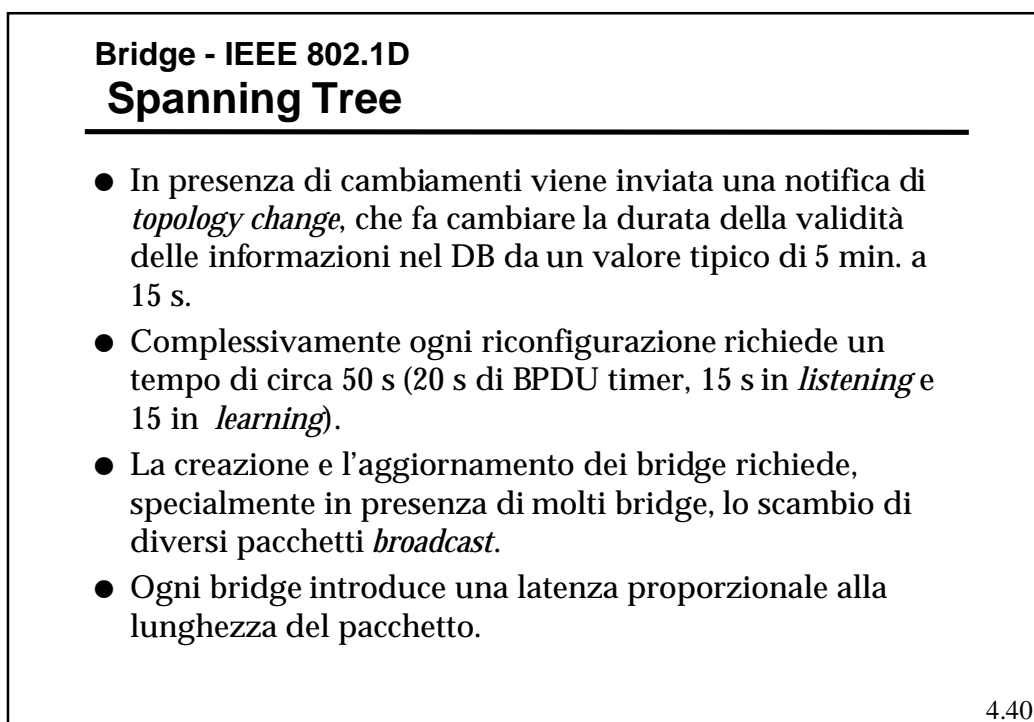
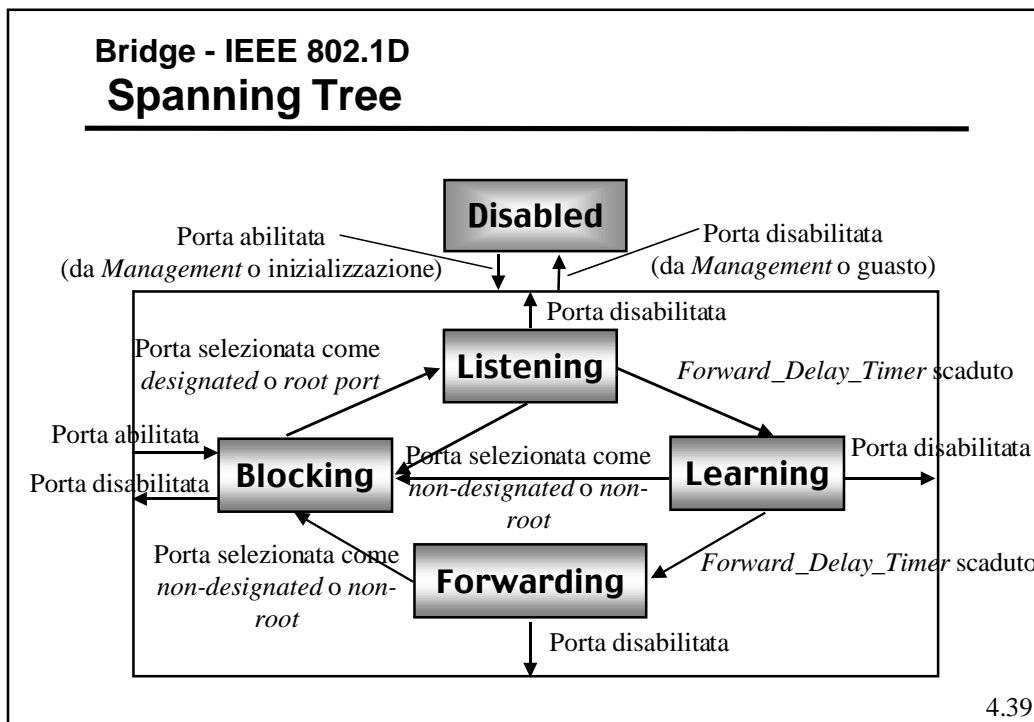


### Bridge - IEEE 802.1D

## Spanning Tree

- Per poter mantenere l'albero sempre aggiornato e corretto, il *root bridge* continua a mandare periodicamente (l'intervallo d'invio è un parametro, detto *hello time*, che può essere modificato; in genere vale 2 s) le BPDU.
- Ogni stazione aggiorna un timer, che viene azzerato ogni qualvolta si riceve una BPDU; se il valore di tale timer supera i 20 s (tipicamente, ma può essere anche diverso), la stazione presuppone che la *root* non sia più raggiungibile.
- Ogni qual volta che la *root* non appare più raggiungibile o il valore dei *root path cost* cambia, l'albero viene ricalcolato e le porte possono passare da *blocked* a *forwarding* e viceversa.

4.38



### Bridge - IEEE 802.1D

#### Remote Bridge

---

- I Bridge remoti sono bridge che hanno porte adatte ad una interconnessione su scala geografica, per esempio possono essere interconnessi a
  - fibra ottica (sino a 50Km nel caso FDDI)
  - linee dedicate con velocità maggiori o uguali a 64kb/s (anche via satellite)
  - fasci di microonde (difficile in Italia) sino a 10km
  - reti *Frame relay*
  - reti a larga banda: SMDS, ATM
- Non sono specificati nello standard, che prevede questo tipo di interconnessione solo a livello 3.

4.41

### Bridge - IEEE 802.1D

#### Remote Bridge

---

- Spesso non è accettabile (specialmente in presenza di connessione in area geografica) che le linee in *blocking state* non portino traffico (alcuni costruttori propongono algoritmi proprietari per usare le linee in *blocking state*)
- Anche con questi algoritmi i bridge sono poco idonei a fare *internetworking* su scala geografica:
  - E' difficile garantire il comportamento FIFO e la non generazione dei duplicati durante le fasi di transizione dello *spanning tree*
  - possono innescarsi pericolosi *loop*
  - si propaga comunque il traffico di *broadcast* e *multicast*

4.42

## Bridge - IEEE 802.1D

### Spanning Tree

---

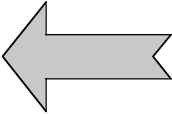
- Perché i bridge sia realmente efficaci devono:
  - Essere avere prestazioni tali da non generare rallentamenti sulla rete (pacch. /s)
  - Essere posti in modo da confinare il più possibile il traffico locale su un *collision domain*.
  - Essere configurati in modo che l'algoritmo di *spanning tree* generi configurazioni efficienti.

4.43

Telematica

## Internetworking L2/L3

---

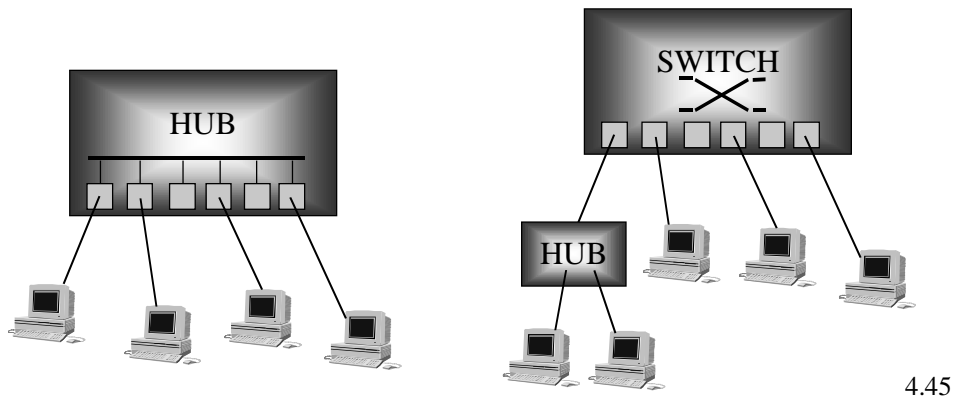
- Bridge
- **Switch L2** 
- Switch L3
- Virtual LAN (VLAN)

4.44

## L2 Switch

---

- Inizialmente i bridge si limitavano a interconnettere una o due LAN, l'evoluzione della topologia da bus a stella ha favorito la nascita di bridge multiporta come centro stella, che diventano dei veri e propri commutatori (switch).



4.45

## L2 Switch

---

- Se ogni porta ha una sola stazione connessa la condivisione del mezzo in pratica non esiste più (fra le stazioni direttamente connesse allo *switch*), le collisioni avvengono solo per ricezioni e trasm. contemporanee; i pacchetti vengono eventualmente persi nel buffer del bridge quando il carico si concentra verso una o poche porte.

4.46

## L2 Switch

---

- Tecniche di Ethernet *switching*
  - *Store-and-Forward*: utilizzata dai bridge (prevista da IEEE 802.1d) il pacchetto viene ricevuto interamente e poi ritrasmesso
  - ***Cut through*** o ***On-The-Fly Switching***: la decisione di inoltrare viene presa durante il transito del pacchetto nello switch
  - ***Fragment free***: prima di iniziare a ritrasmettere il pacchetto si aspetta comunque un tempo pari alla *collision window* (51.2 m ms)

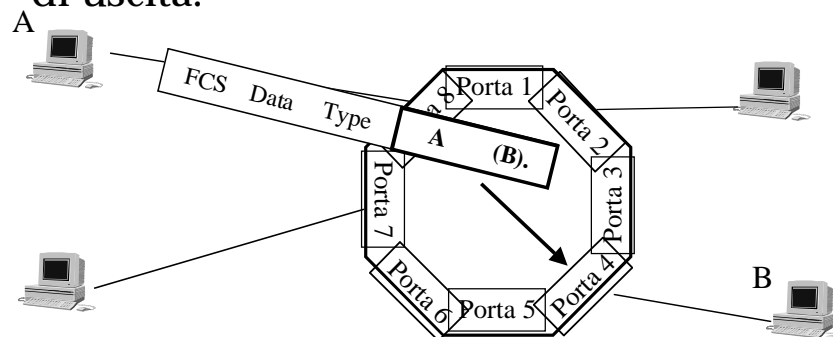
4.47

### L2 Switch

## Cut-through switch

---

- I tempi di latenza sono molto bassi (40-60  $\mu$ s) perché quando lo *switch* legge l'indirizzo di destinazione decide la porta di uscita.



4.48



## L2 Switch

**Cut-through switch**

---

- Le tecniche Cut through e *Fragment free* possono essere utilizzate solo se:
  - su tutte le porte è presente lo stesso tipo di MAC
  - tutte le porte hanno la stessa velocità trasmissiva
  - la porta di destinazione è libera
  - il pacchetto non è broadcast o multicast
- Altrimenti occorre fare *Store & Forward*
- Per i pacchetti corti *Cut through*, *Fragment free* e *Store & Forward* sono equivalenti
- Con velocità elevate (100 Mb/s o 1 Gb/s) la latenza di uno *store and forward* è comunque molto piccola.
- Il *Cut through* inoltra anche i frammenti di collisione

4.49

## L2 Switch

**Store and forward switch**

---

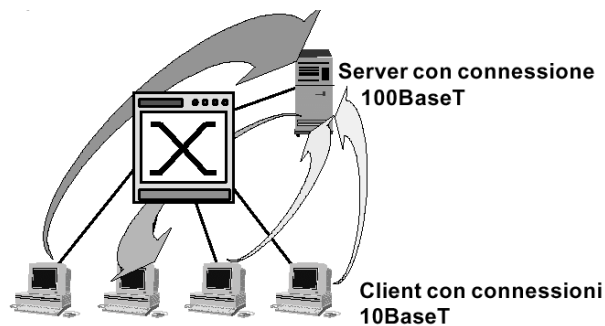
- Opera come un Bridge Multiporta ad alte prestazioni
- Può interconnettere MAC diversi: Ethernet, FDDI, ATM
- Può operare a velocità diverse: 10 Mb/s (802.3), 100 Mb/s (802.3u) e 1 Gb/s
- Non inoltra pacchetti contenenti errori poiché controlla il CRC
- Non inoltra i frammenti di collisione.
- Con velocità elevate ha comunque una latenza molto piccola.

4.50

## L2 Switch

**Esempio**

- Uno *switch* con connessioni a velocità diverse è adatto per applicazioni *Client-Server*
  - server connesso ad alte prestazioni (esempio 100Mb/s - 1 G/s), i *client* a medie



4.51

## L2 Switch

**Half and Full Duplex**

- Le LAN sono strutture intrinsecamente *half-duplex*: trasmette una sola stazione per volta
- Lo *switching* ridimensiona molto il ruolo del mezzo fisico condiviso:
  - spesso il mezzo trasmissivo diventa punto-punto: sono collegati unicamente la stazione e lo *switch*
- I mezzi trasmissivi punto-punto possono
  - essere *full-duplex*: entrambe le stazioni possono trasmettere contemporaneamente le trasmissioni avvengono su canali fisici diversi

4.52

## L2 Switch

**Ethernet Full Duplex**

---

- Ethernet full-duplex è attualmente utilizzata principalmente per le dorsali: raddoppia la banda disponibile sulle connessioni *Bridge-to-Bridge* o *Switch-to-Switch*
- Necessita di particolari *transceiver* in cui non viene rilevata la collisione. (i *transceivers* normali inviano un segnale di collisione all'interfaccia quando si ha la presenza di attività contemporanea su TX e RX)
- La distanza tra due stazioni full-duplex
  - dipende solo dalle caratteristiche del mezzo trasmissivo (fino a 50 Km con fibra monomod.)
- è indipendente dal diametro del dominio di collisione<sup>4.53</sup>

## L2 Switch

**Tipi di prodotti**

---

- ***Workgroup Switch***: apparati di medie prestazioni, con numero limitato di porte e costi bassi
- ***Switch modulari***: apparati di elevate prestazioni, con elevata modularità e flessibilità
- ***HUB/Switch***: apparati modulari che realizzano le funzionalità di *hub (repeater)* e di *switch*

4.54

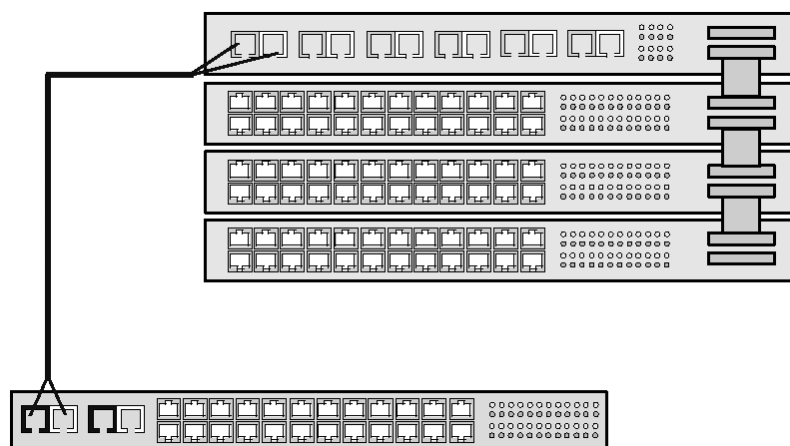
## L2 Switch

**Workgroup Switch**

- **Caratteristiche generali:**
  - possono utilizzare le tecniche *cut-through*, *fragment free* e *store and forward*
  - VLAN
  - management SNMP, RMON
- alcuni dispongono di una connessione, proprietaria a alta velocità per impilare un limitato numero di apparati (modelli *stackable*)
- alcuni sono dotati di alimentatore supplementare di *backup*

4.55

## L2 Switch

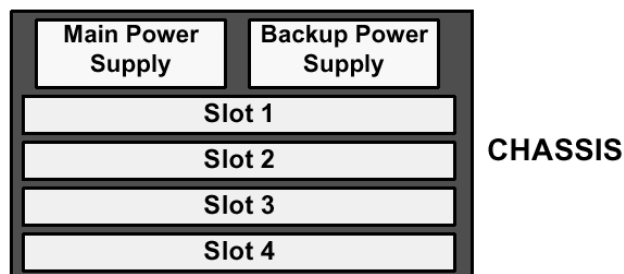
**Workgroup Switch**

4.56

## L2 Switch

**Switch Modulari**

- Costituiti da uno chassis che può alloggiare diversi tipi di moduli *Switching*
- Caratteristiche dello chassis:
  - numero di slot variabile da 2 a 8
  - alimentatore ridondante



4.57

## L2 Switch

**Switch Modulari**

- Costituiti da uno chassis che può alloggiare diversi tipi di moduli *Switching, Repeater, Router, Terminal Server*
- I moduli *switch* possono avere
  - ogni porta di tipo *switching*
  - diversi gruppi di porte che fanno parte dello stesso dominio di collisione
    - » ogni gruppo di porte dispone di un *multiport repeater* integrato
    - » lo *switching* viene realizzato tra i gruppi di porte
- Lo chassis ha un numero di slot variabile e alimentatore ridondante

4.58

Telematica

## Internetworking L2/L3

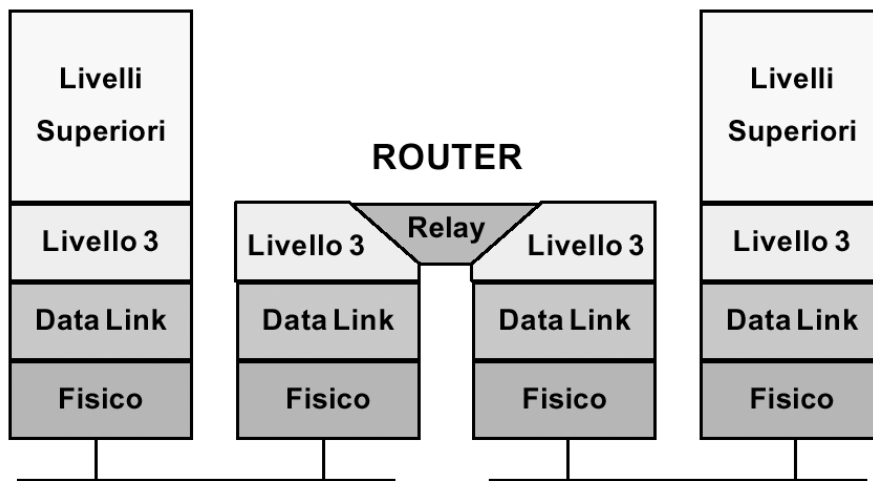
---

- Bridge
- Switch L2
- **Switch L3** ←
- Virtual LAN (VLAN)

4.59

### L3 Switch Router

---



4.60

**L3 Switch**  
**Router**

---

- Lavorano a livello 3
- Sono limitati ad un solo o comunque a pochi protocolli
- Sono adeguati a gestire topologie anche molto complesse, utilizzando tutte le linee a disposizione
- Funzionano bene anche su linee lente
- Non propagano incondizionatamente traffico *broadcast* o *multicast*
- Permettono un *routing* gerarchico suddividendo le reti in aree

4.61

**L3 Switch**  
**Router - L3 Switch**

---

- Un *L3 switch* è un *router* ad elevata velocità
- Per poter ottenere tali velocità spesso le funzionalità del *router* vengono semplificate
- L'eventuali semplificazioni lo rendono adatto in particolare ad interconnettere VLAN.
- Rispetto ad *L2 switch* è in genere più costoso ed ha meno porte.

4.62

L3 Switch

## Router - Bridge

---

- Indirizzamento
  - I *router* sono indirizzati esplicitamente, i bridge sono trasparenti ai nodi
- Calcolo instradamento
  - I *router* ricevono ed usano molte informazioni, mentre i bridge usano solo gli indirizzi di mittente e destinatario di livello 2
- Pacchetti/ *frame*
  - I *router* operano su pacchetti di livello 3 e possono dividere/riunire i messaggi per adattarli a reti con lunghezze massime diverse. I bridge non possono toccare i campi dati.

4.63

L3 Switch

## Router - Bridge

---

- *Feedback*
  - I *router* possono fornire informazioni sullo stato della rete all'utente finale
- *Forwarding*
  - I *router* ritrasmettono i messaggi cambiando gli indirizzi di livello 2
- *Priority*
  - I *router* possono utilizzare priorità
- *Security*
  - I *router* possono realizzare tecniche di *firewall*

4.64



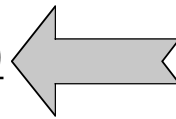
Telematica

## Internetworking L2/L3

---

- Bridge
- Switch L2
- Switch L3

- Virtual LAN (VLAN)



4.65

## Virtual LAN (VLAN)

---

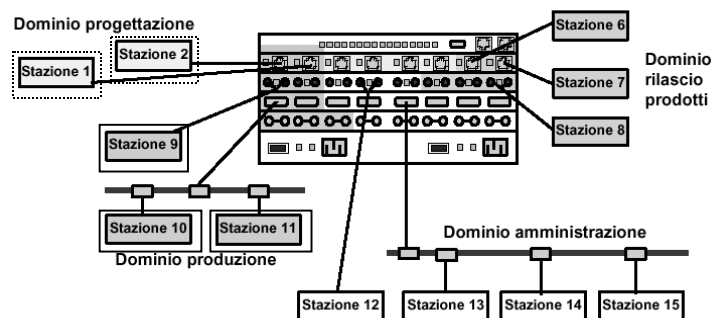
- Le LAN estese (BLAN), quando crescono troppo di dimensione, sono fonte di problemi:
  - elevato traffico di *multicast/ broadcast*
  - *routing* tra le sottoreti (IP)
  - Sicurezza
- Allora si è introdotto il concetto di LAN virtuali:
  - unica infrastruttura fisica
  - definizione di più sottoreti logiche separate (interconnesse tramite *router*)
- Le LAN virtuali possono estendersi su:
  - il singolo *switch*
  - l'intera LAN estesa

4.66

## VLAN

**Single switch VLAN**

- Due o più porte dello *switch* possono essere raggruppate in un dominio di *broadcast* (per interconnettere i domini ci vuole un *router*)

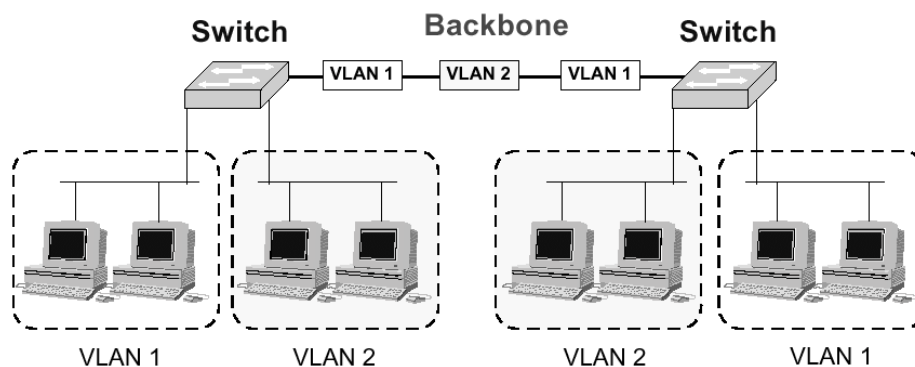


4.67

## VLAN

**Multi switch VLAN**

- Nel caso *multi-switch*, le macchine raggruppate in una singola VLAN possono essere connesse a switch differenti.



4.68

## L2 Switch

**Membership**

---

- Un singola stazione può appartenere ad una o più VLAN.
- L'appartenenza (*membership*) può essere di tipo:
  - **Statica**
    - » per porta (*Port Grouping*)
  - **Dinamica**
    - » per indirizzo MAC (*MAC Address Grouping*)
    - » per protocollo o per indirizzo di livello 3 (*Protocol Grouping*)
    - » per indirizzo multicast
    - » “policy-based” (per applicazione, per utente,...)

4.69

## VLAN

**Port Grouping**

---

- **Vantaggi**
  - E' facile da capire e da gestire.
  - Esiste una metodologia comune a diversi fornitori per realizzarlo.
  - Dato che ad una singola porta possono essere connessi hub, può essere usato per riunire reti a mezzo condiviso.
- **Svantaggi:**
  - Si deve essere riconfigurare se una stazione viene mossa da una porta ad un'altra.
  - Non permette ad una porta di appartenere a più di una VLAN.
  - Richiede uno scambio di informazione di controllo tra gli switch

4.70

VLAN

## **MAC Address Grouping**

---

- **Vantaggi**

- Si può muovere le stazioni senza dover riconfigurare la VLAN

- **Svantaggi**

- Più utenti appartenenti a VLAN diverse attaccati alla stessa porta danno luogo a forti decadimenti di prestazioni
- Ogni stazione deve appartenere ad almeno una VLAN.
- Richiede un elevato scambio di informazione fra gli *switch*.

4.71

VLAN

## **Protocol Grouping**

---

- **Vantaggi**

- Permette di riunire per tipo di protocollo.
- Non richiede riconfigurazioni in corrispondenza di spostamenti.
- Una singola porta può partecipare a più VLAN.
- Non richiede uno scambio di informazioni per identificare le VLAN.
- Può essere realizzato in modo efficace legandolo alle *subnet* IP.

- **Svantaggi**

- Può avere prestazioni ridotte.
- Non funziona con protocolli di solo livello 2 (*NetBios*).
- In taluni casi comporta limitazioni.

4.72

## VLAN

**Frame Tagging**

---

- Il *tag* (etichetta) contiene l'informazione relativa all'appartenenza ad una VLAN di un *frame*. Si può avere
  - ***Implicit tagging***
    - » nessun tag viene aggiunto al frame
    - » Facile da realizzare in ambienti orientati alla connessione
    - » Difficile per multicast/broadcast frames
  - **Explicit tagging**
    - » Viene aggiunto un tag ad ogni frame che trasporta la "VLAN *membership*" fra i bridge.
    - » Il tag può essere usato per trasportare anche altre informazioni (per es. priorità)

4.73

## VLAN

**Explicit tagging**

---

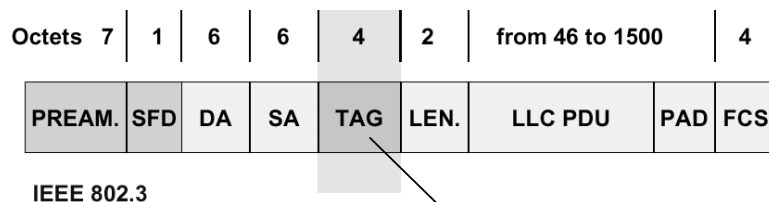
- Due possibili posizioni del tag nel frame:
  - *One level tagging (Internal tagging)*
  - *Two level tagging (External tagging)*
- Ambedue richiedono una realizzazione hardware

4.74

## VLAN

**One level tagging**

- Il frame originale viene modificato con l'aggiunta del *tag* all'interno
- Il nuovo frame continua ad avere un formato valido anche per gli apparati "non consci" (unware) delle VLAN.
- Gli indirizzi MAC SA e DA rimangono immutati
- Nasce l'eventuale problema dei "baby giant"



Campo aggiunto

4.75

## VLAN

**Two level tagging**

- Il frame originale non viene modificato ma un *header* esterno viene aggiunto in testa.
- Vengono inseriti un nuovo SA, DA, Ethertype e VLAN-ID
- Non esistono problemi di "giant frames"



Inter-Switch Link (esempio)

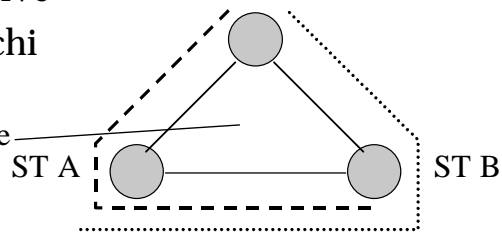
4.76

## VLAN

**VLAN Spanning Tree**

- Tre diverse alternative:
  - Un unico ST
  - Un ST per ogni VLAN
  - Più ST eventualmente condivisi
- Un singolo ST non permette:
  - Multiple topologie attive
  - Bilanciamento di carichi

Tutte le linee vengono utilizzate



4.77

## VLAN

**VLAN**

- Due possono essere le interpretazioni delle VLAN:
  - Come metodo efficace per determinare filtri locali in una bridge-LAN e quindi creare reti di accesso che limitino la connettività *end-to-end*.
  - Come metodo efficace per realizzare più *bridged* LAN su di un'unica struttura fisica.

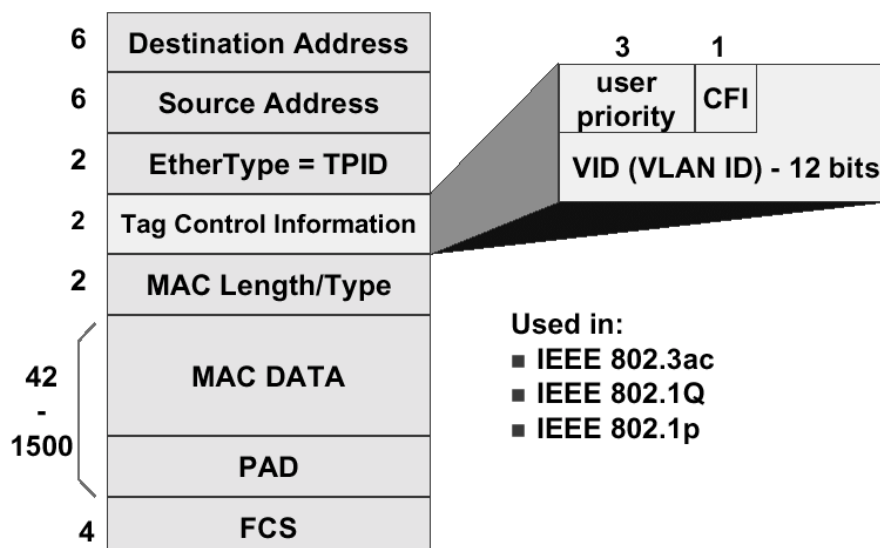
4.78

## VLAN IEEE Standard

- All'interno dell'IEEE 802 a partire dal 1995 sono stati proposti un insieme di standard per il supporto delle VLAN
- IEEE 802.1
  - 802.1p Priorities/GARP/GMRP
  - 802.1Q VLANs/GVRP
- 802.3: CSMA/CD (Ethernet)
  - 802.3ac

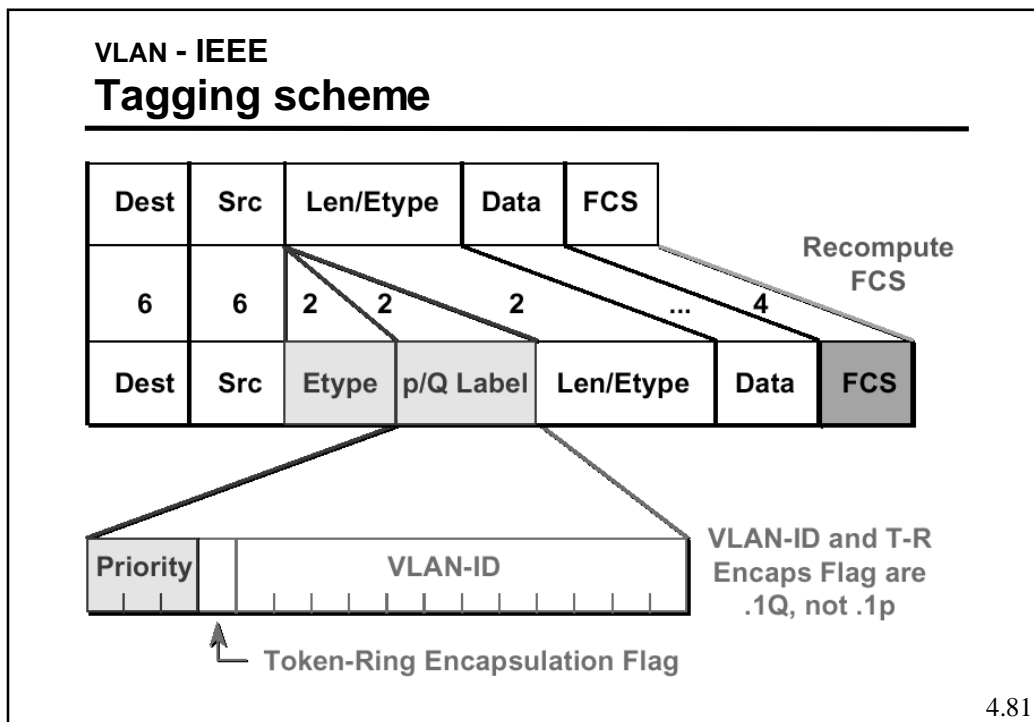
4.79

## VLAN - IEEE Tagging scheme



4.80





**VLAN - IEEE**  
**802.1p**

- IEEE Standard per LAN e MAN:  
*“Supplement to Media Access Control (MAC) bridges. Traffic Class Expediting and Dynamic Multicast Filtering”*
- Due obiettivi principali:
  - *“Expedited traffic capabilities”*
  - *“Filtering services to support the dynamic use of Group MAC addresses”*

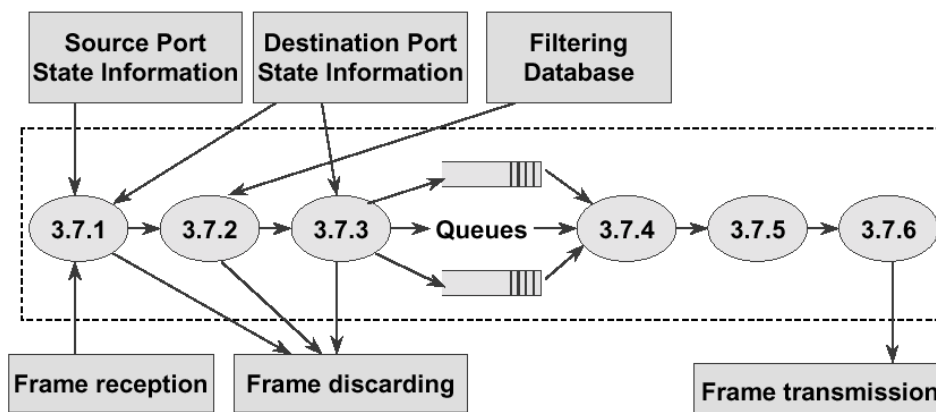
4.82

### VLAN - IEEE 802.1p ***Expedited traffic***

- Uso della priorità
  - MAC - *layer priority* del *tag*
  - La priorità non viene ricavata dall'indirizzo MAC
- Code di uscita multiple per ogni porta
  - selezione della coda sulla base della priorità
  - Mantenimento dell'ordine dei *frame* solo per priorità uguali
- 802.1 coopera con IETF's ISSLL (*Integrated Services over Specific Lower Layers*)

4.83

### VLAN - IEEE 802.1p ***Expedited traffic***



3.7.1 Enforcing topology restriction  
3.7.2 Filtering Frames  
3.7.3 Queueing Frames

3.7.4 Selecting frames for transmission  
3.7.5 Mapping priority  
3.7.6 Recalculating FCS

4.84

### VLAN - IEEE 802.1p Expedited traffic

---

- Sono definite 8 priorità (da 0 a 7), la 0 è la più bassa (servita per ultima) la 7 la più alta.
- Il bridge deve assicurare l'ordine dei pacchetti appartenenti alla stessa priorità.
- Sono previste fino a 8 classi di traffico, che corrispondono ad una coda per classe per porta di uscita.
- Una coda di classe  $i$  è servita solo se le code di classe  $> i$  sono vuote.
- La tabella indica il legame fra la priorità di ingresso e quella di uscita a seconda del numero di classi supportate dal bridge.

4.85

### VLAN - IEEE 802.1p Priority mapping

---

		Number of Available Traffic Classes							
		1	2	3	4	5	6	7	8
User Priority	0 (Default)	0	0	0	1	1	1	1	2
	1	0	0	0	0	0	0	0	0
	2	0	0	0	0	0	0	0	1
	3	0	0	0	1	1	2	2	3
	4	0	1	1	2	2	3	3	4
	5	0	1	1	2	3	4	4	5
	6	0	1	2	3	4	5	5	6
	7	0	1	2	3	4	5	6	7

4.86

VLAN - IEEE 802.1p

**GARP**

---

- Generic Attribute Registration Protocol
  - Serve per disseminare dichiarazioni di attributi nella rete
  - Viene usato dai partecipanti alle “GARP *applications*” (GARP *participants*).
  - Ogni GARP *application* ha i propri tipi e valori per gli attributi
  - E’ stato progettato per registrare qualunque cosa.

4.87

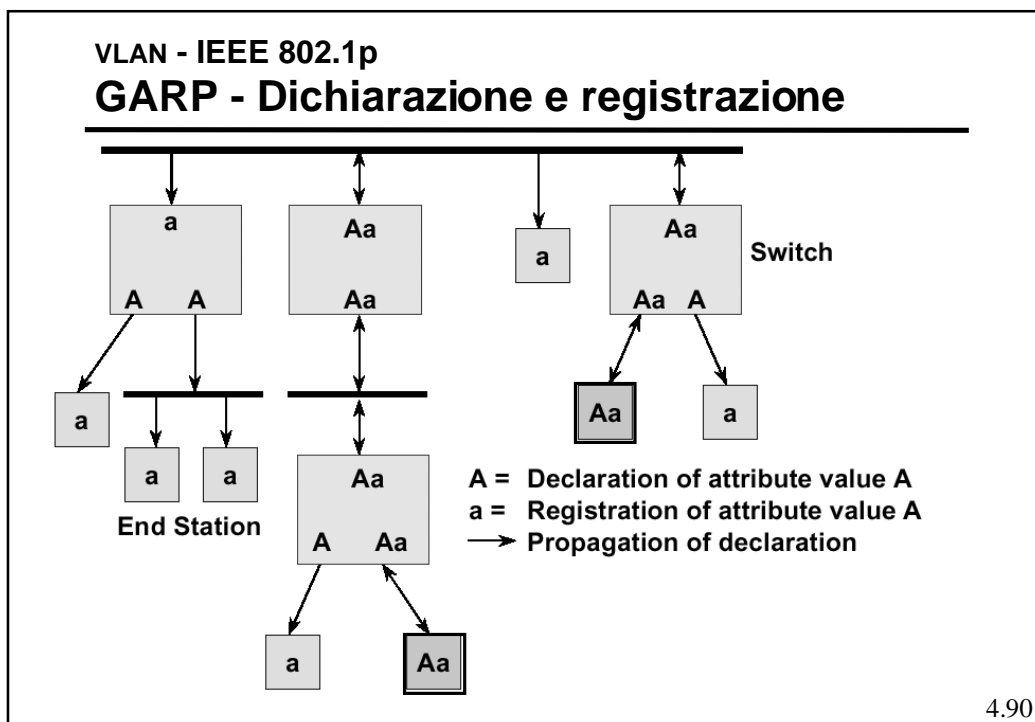
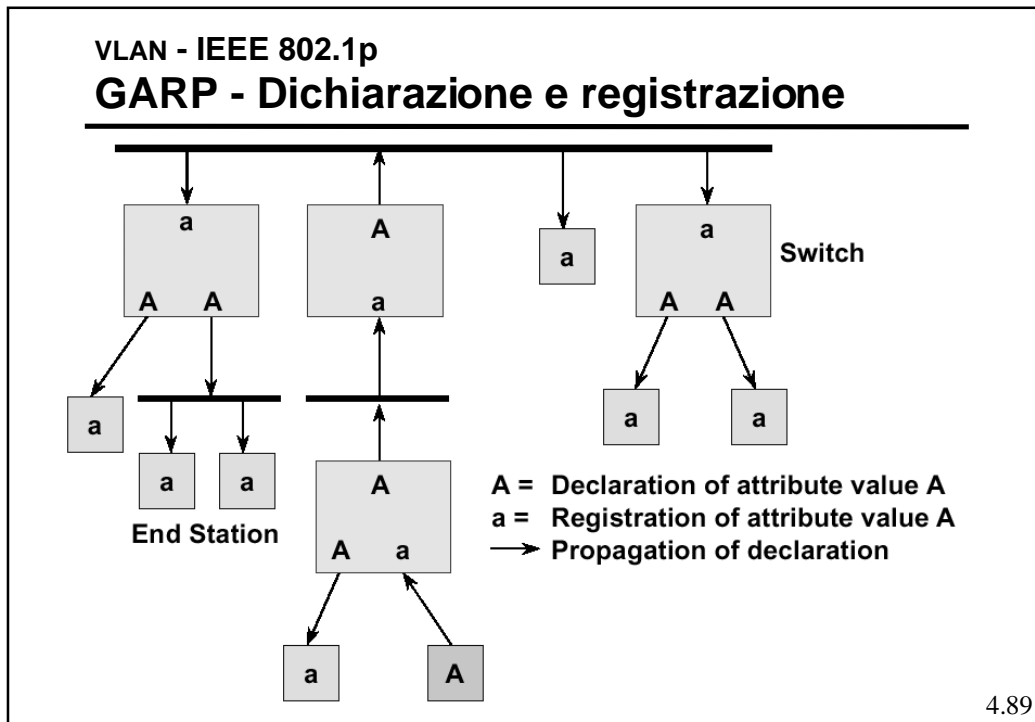
VLAN - IEEE 802.1p

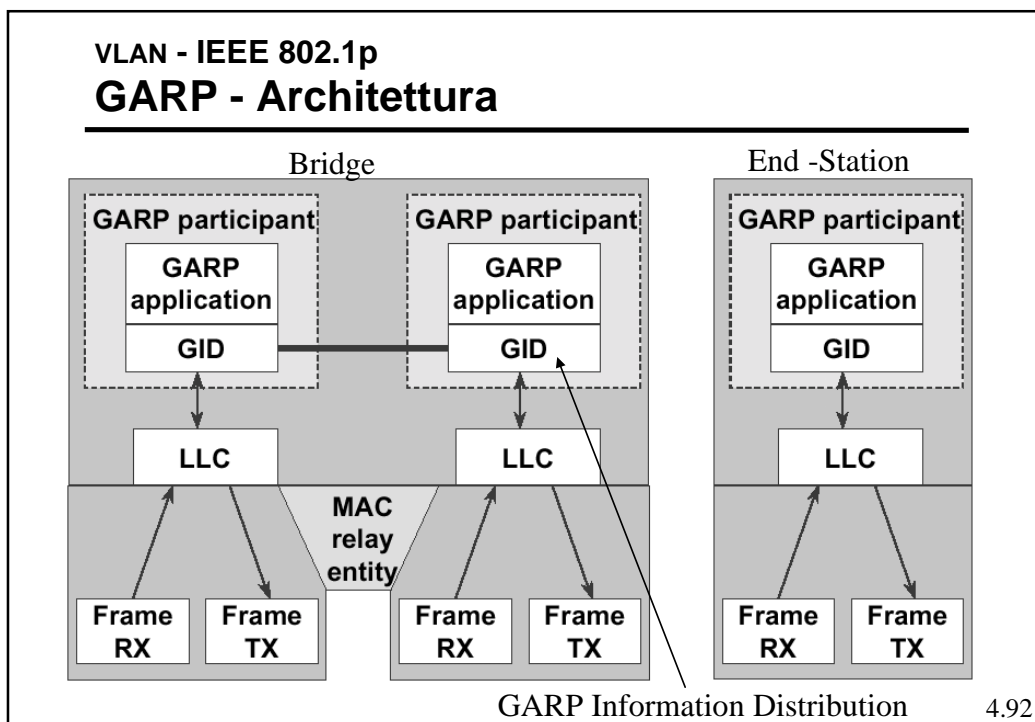
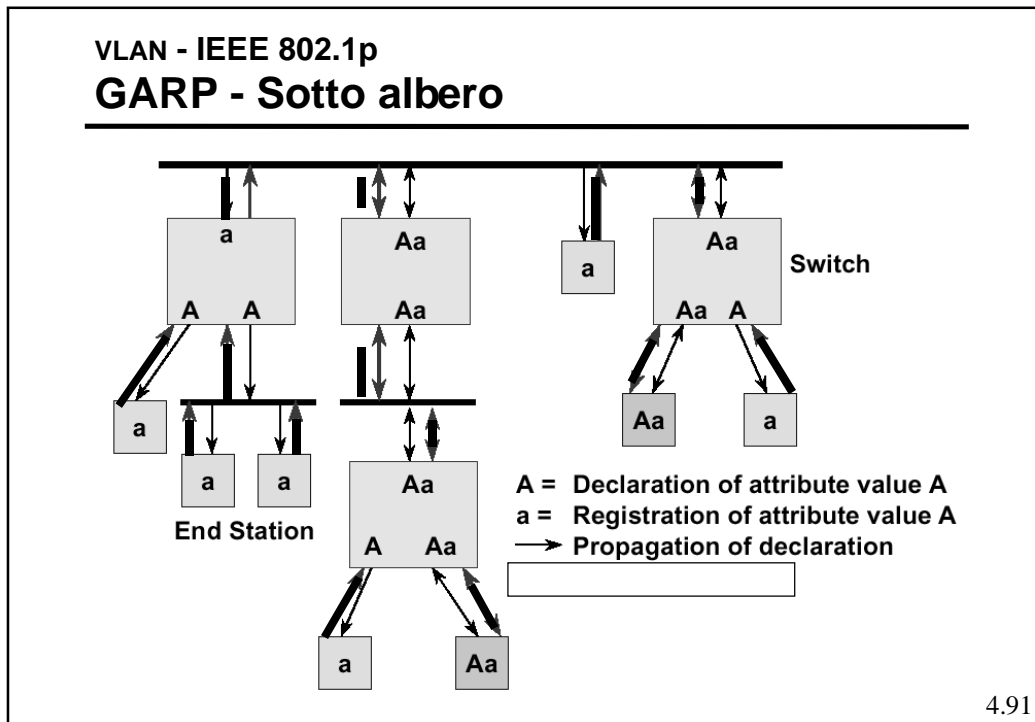
**GARP**

---

- Ogni GARP *Application*
  - Fa o annulla dichiarazioni relative a valori di attributi
  - Tali azioni danno luogo alla registrazione / cancellazione di valori di attributi negli altri GARP *participants*
  - Le registrazioni sono registrate in una variabile di stato
    - » Solo le porte che ricevono la GARP PDU contengono la registrazione
    - » Anche le porte non in forwarding
  - Le dichiarazioni registrate da porte attive vengono propagate dai bridge anche a tutte le altre porte appartenenti alla “*active topology*” (cioè allo *spanning tree* attivo al momento).

4.88





VLAN - IEEE 802.1p

**GMRP**

---

- GMRP Multicast Registration Protocol
- Permette la gestione delle appartenenze a gruppi *multicast* a livello di MAC (quello che fa IGMP a livello 3 nel TCP-IP)
  - Permette alle stazioni finali di registrarsi in un gruppo *multicast* o in tutti.
  - Registra le richieste da ogni porta per ogni indirizzo *multicast*
  - Permette la commutazione dei *frame* solo dove è necessario

4.93

VLAN - IEEE 802.1p

**Priorità**

---

- Il tag 1Q/1p può essere usato anche in assenza di VLAN
- In questo caso serve solo per il trasporto di priorità.

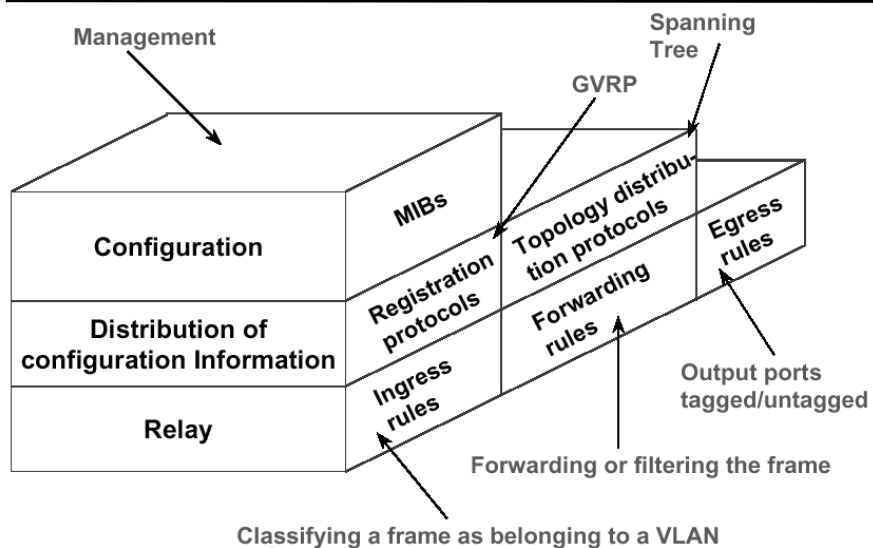
4.94

## VLAN IEEE 802.1Q

- Definisce le funzionalità dei bridge “VLAN-*aware*”
- E’ in grado di gestire configurazioni miste (bridge VLAN-*aware* e VLAN-*unaware*)
- Scelte architetturali
  - Per-port VLAN (Protocol Grouping è permesso ma non specificato)
  - Un unico *spanning tree*
  - *Explicit Tagging* (ad un livello)

4.95

## VLAN - IEEE 802.1Q Architettura



4.96



## VLAN - IEEE 802.1Q

**GVRP**

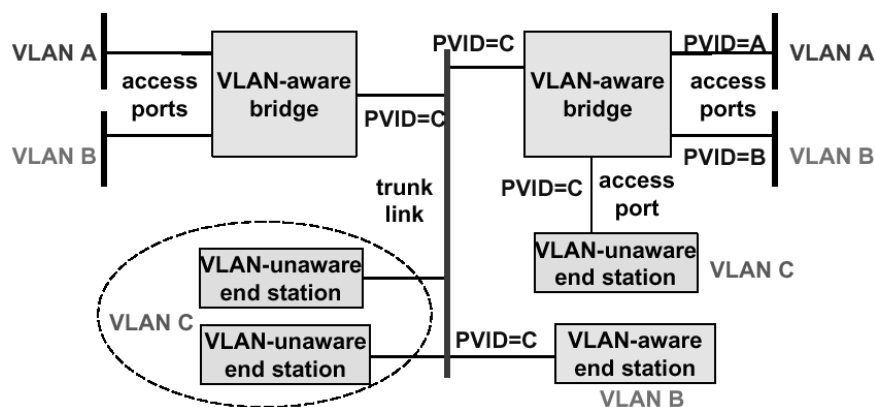
- GARP VLAN Registration Protocol
- Gestisce l'appartenenza alle VLAN
  - Le stazioni terminali o i bridge possono fare o revocare dichiarazioni relative all'appartenenza a VLAN
  - Il tipo di attributo è il 12 bit VID (VLAN-ID)

4.97

## VLAN - IEEE 802.1Q

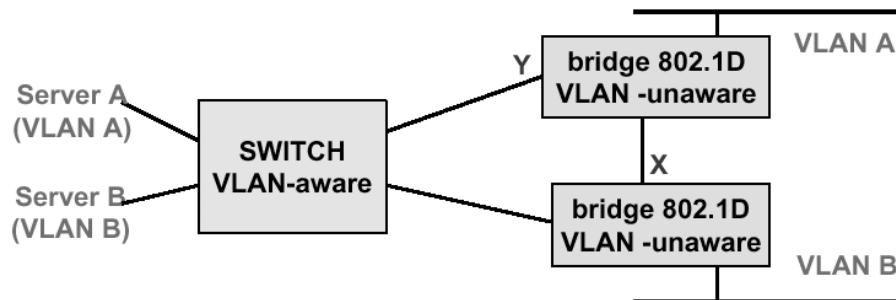
**Presenza di stazioni VLAN-unaware**

- Ogni porta fisica ha un PVID (Port-VLAN-ID) che viene assegnato a tutti i frame “*untagged*”



4.98

## VLAN - IEEE 802.1Q

**Presenza di bridge VLAN-unaware**

- Se la porta X è bloccata tutto funziona bene
- Se viene bloccata la porta Y invece, la VLAN A risulta partizionata (il server A è irraggiungibile dalla VLAN A)

4.99

## VLAN - IEEE 802.1Q

**Interazioni fra 1Q e 1p**

- 1p introduce il GARP
  - Specifica anche come gestire i *frame* in un ambiente *multicast* usando il GMRP
- 1Q introduce il GVRP e specifica come gestire i *frame* in un ambiente VLAN
  - Su quali porte
    - » Su quali VLAN
    - » Con *tag* o senza
  - 1p specifica anche come decodificare e trattare le priorità

4.100