# Capitolo 1

# Link Layer e LAN

## 1.1 Il link layer

I dispositivi che supportano un protocollo link-layer sono detti *nodi*. I canali di comunicazione che connettono nodi adiacenti sono detti *collegamenti*. Un nodo incapsula il datagramma ricevuto dal network layer sovrastante in un *link-layer frame* e lo trasmettono sul collegamento.

### 1.1.1 I servizi forniti dal link layer

#### Incapsulazione

Quasi tutti i protocolli link-layer incapsulano i datagrammi ricevuti dal network layer prima di trasmetterli sul collegamento. Il frame è composto da un campo dati, dove viene inseito il datagramma, e degli header.

#### Accesso al collegamento

Un protocollo di medium access control (MAC) specifica come il frame deve essere trasmesso sul collegamento.

#### Trasporto affidabile

Un protocollo di trasferimento affidabile garantisce che ogni frame raggiunga la sua destinazione senza errori.

#### Individuazione e correzione degli errori

Il nodo mittente fornisce un meccanismo per individuare gli errori, che verranno poi corretti dal destinatario.

### 1.1.2 Implementazione del link layer

Le funzionalità Ethernet sono integrate nella scheda madre o in un chip Ethernet. Il link layer è implementato su un chip detto *network adapter* o *NIC*.

## 1.2 Individuazione e correzione degli errori

### 1.2.1 Controlli di parità

La forma più semplice di error detection è l'utilizzo di un bit di parità. Gli schemi di parità possono essere pari o dispari. Con uno schema di parità *bidimensionale*, dove i bit sono disposti a matrice, è

possibile identificare il bit corrotto e correggerlo. Questo schema non può correggere due errori in un singolo pacchetto, ma li può individuare.

### 1.2.2 CRC

I codici CRC (*cyclic redundancy check*) sono anche detti codici polinomiali in quanto è possibile considerare la stringa da inviare come un polinomio i quali coefficienti sono i valori 0 e 1 della stringa di bit. I CRC possono individuare $resto \leq$ bit errati consecutivi.

## 1.3 Multiple access link

Esistono due tipi di collegamento. Il collegamento *point to point* consiste in un solo mittente e un solo destinatario. Il collegamento *broadcast* può avere più nodi mittenti e destinatari connessi allo stesso canale di broadcast. Per coordinare la trasmissione di pacchetti all'interno di una rete broadcast, si utilizzano *protcolli di accesso multiplo*. Quando due o più nodi trasmettono frame allo stesso momento, i frame *collidono* e vengono persi. Esistono tre tipi di protocolli ad accesso multiplo.

### 1.3.1 Protocolli a partizionamento di canale

- *TDMA*: l'accesso al canale avviene in "round", e ogni nodo ha uno slot di tempo in cui può trasmettere in ognuno di questi round. Il tasso di trasmissione è $R/N$, e i canali inutilizzati vengono sprecati.
- *FDMA*: divide il canale da $R$ bps in multiple frequenze e assegna ciascuna di queste frequenze ai nodi.
- *CDMA*: assegna un codice a ogni nodo, e ogni nodo usa quel codice per codificare i bit da inviare.

### 1.3.2 Protocolli ad accesso casuale

Con questi protocolli, il nodo mittente trasmette alla velocità concessa dal canale. Quando avviene una collisione, i nodi affetti ritrasmettono il pacchetto finchè non viene ricevuto, attendendo prima di ritrasmettere il pacchetto. Ogni nodo sceglie indipendentemente l'intervallo di ritrasmissione.

#### Slotted ALOHA

Assumiamo che i frame abbiano tutti la stessa dimensione $L$, il tempo sul canale sia diviso in slot di dimensione $L/R$, i nodi siano sincronizzati e se due frame collidono in un slot, tutti i nodi rilevano la collisione in quello slot. Sia $p$ una probabilità tra 0 e 1.

- Quando un nodo ha un frame da inviare, attende fino al prossimo slot e lo trasmette.
- Se non si verifica una collisione, il nodo non ritrasmette.
- Altrimenti, il nodo rileva una collisione e ritrasmette il suo frame negli slot successivi con probabilità $p$ finchè non riesce a ritrasmettere senza collisione

La probabilità introduce casualità, che rende più efficiente il protocollo. Slotted ALOHA permette a ogni nodo di trasmettere a full rate ed è decentralizzato, ma nelle collisioni gli slot vengono persi, possono esserci slot inutilizzati ed è necessario un meccanismo di sincronizzazione. Inoltre, non è molto efficiente: con un gran numero di nodi, solo il 37% degli slot viene effettivamente utilizzato.

#### CSMA

I protocolli CSMA e CSMA/CD seguono due regole:

- *carrier sensing*: un nodo ascolta sul suo canale prima di trasmettere; se il canale è occupato, attende
- *collision detection*: se un nodo rileva un altro nodo che sta trasmettendo sul canale, annulla la trasmissione

Nonostante il carrier sensing, le collisioni possono comunque avvenire a causa del ritardo di propagazione. Quando avviene una collisione, tutto il tempo impiegato a inviare un pacchetto viene sprecato. Il CSMA semplice non effettua collision detection.

#### CSMA/CD

CSMA/CD effettua collision detection, annullando una trasmissione se nota che il canale è occupato. Prima di ritrasmettere, il nodo attende una quantità di tempo casuale. Algoritmo CSMA/CD di Ethernet:

1. Ethernet riceve il datagramma dal livello rete e lo incapsula in un frame.
2. Se il canale è libero, trasmette, altrimenti attende.
3. Se non vengono rilevate collisioni, la trasmissione è andata a buon fine.
4. Altrimenti, la trasmissione viene annullata e viene inviato un segnale.
5. Ethernet entra in fase di *binary exponential backoff*: dopo la $m$-esima collisione, sceglie un numero $K$ con $0 \leq 2^{m-1}$. Attende $512K$ bit volte e ritorna al punto 2, ripetendo fino al completamento della trasmissione.

### 1.3.3 Protocolli a turni

I protocolli a turni permettono di ottenere buone prestazioni sia con carichi leggeri che con carichi pesanti.

- *Polling protocol*. Un nodo viene scelto come nodo master. Questo interpella ognuno dei nodi con metodo round-robin. In questo modo, vengono eliminate le collisioni e non ci sono tempi morti. I principali svantaggi sono l'introduzione di un ritardo di polling, la latenza e il potenziale fallimento del nodo master.
- *Token-passing protocol*. Un frame apposito, detto token, viene passato in ordine tra i nodi. Quando un nodo riceve un token, lo tiene finchè ha dei frame da trasmettere. I principali svantaggi sono l'overhead introdotto dal token, la latenza e il fatto che il token rappresenta un potenziale punto di rottura.

## 1.4 Switched Local Area Networks

Gli switch utilizzano indirizzi propri al link layer per inoltrare frame.

### 1.4.1 Indirizzamento nel link layer e ARP

#### Indirizzi MAC

Le interfacce di rete degli host e dei router hanno indirizzi link layer, ma non gli switch. Un indirizzo link layer viene chiamato *indirizzo MAC*. Gli indirizzi MAC sono di solito lunghi 6 bytes (48 bit) e sono espressi in motazione esadecimale. Ogni interfaccia nella LAN ha un unico indirizzo MAC e un

(localmente) unico indirizzo IP. Ogni interfaccia possiede un unico indirizzo MAC perché questi sono controllati dalla IEEE.

Quando un'interfaccia vuole inviare un frame, inserisce l'indirizzo MAC di destinazione nel frame e lo inoltra sulla LAN. Se un mittente vuole inviare un frame a tutte le interfacce presenti sulla LAN, questo inserisce un indirizzo speciale, l'*indirizzo broadcast* nel frame.

#### ARP

ARP è il protocollo che si occupa della traduzione tra indirizzi IP e MAC, ma solo per interfacce sulla stessa sottorete.

Ogni host e router possiedono una *tabella ARP*, che contiene mappature tra indirizzi IP e MAC, e un valore di time-to-live che indica la durata di quella voce nella tabella.

Per ottenere l'indirizzo MAC del destinatario, un mittente invia sull'indirizzo MAC di broadcast un *pacchetto ARP*, attendendo la risposta del nodo con l'indirizzo IP corrispondente. Una volta ricevuta la risposta, il mittente aggiorna la sua tabella ARP.

#### Inviare datagrammi in un'altra sottorete

Come inviare un datagramma da un host $A$ a un host $B$? Supponiamo $A$ conosca l'indirizzo IP di $B$, l'indirizzo IP del router e l'indirizzo MAC del router (via ARP). Allora:

- $A$ crea un datagramma con destinazione l'indirizzo IP $B$ (non può conoscere il suo MAC)
- $A$ incapsula il datagramma in un frame indirizzato al router
- il frame è ricevuto dal router e passato a IP
- il router determina la giusta interfaccia sulla quale inoltrare il datagramma
- il router incapsula il datagramma con destinazione indirizzo MAC di $B$

### 1.4.2 Ethernet

La LAN Ethernet originalmente utilizzava un bus coassiale per connettere i nodi, e i frame erano trasmessi in broadcast. Successivamente, il bus è stato rimpiazzato con uno switch, che a differenza dei router opera esclusivamente sul link layer.

#### Struttura di un frame Ethernet

- *Preambolo.* Serve a sincronizzare le interfacce.
- *Indirizzo di destinazione.* Contiene l'indirizzo MAC di destinazione.
- *Indirizzo di provenienza.* Contiene l'indirizzo MAC di provenienza.
- *Tipo.* Serve all'interfaccia di destinazione per indirizzare il frame al giusto protocollo di livello rete.
- *Campo dati.* Contiene il datagramma IP.
- *CRC.*

Ethernet è connectionless e non affidabile.

### 1.4.3 Switch

Gli switch sono dispositivi operanti nel link layer. Sono trasparenti gli host e ai router nella sottorete. Gli switch, come i router, hanno dei buffer per contenere i frame in eccesso.

#### Forwarding e filtering

Il *filtering* è la funzionalità degli switch che determina se un frame deve essere inoltrato o scartato. Il *forwarding* è la funzionalità che determina le interfacce alle quali il frame va inoltrato. Queste funzioni sono svolte tramite una *tabella di switching*. Questa tabella contiene informazioni su alcuni dispositivi presenti in LAN. Ogni voce contiene un indirizzo MAC, il numero di interfaccia dello switch che porta a quell'indirizzo, e quando quell'indirizzo è stato aggiunto alla tabella.

Supponiamo che un frame con indirizzo di destinazione DD-DD-DD-DD-DD-DD giunga a uno switch sull'interfaccia $x$. Lo switch consulta la sua tabella. Si possono verificare tre casi.

- Nessuna voce nella tabella corrisponde all'indirizzo DD-DD-DD-DD-DD-DD. Lo switch inoltra copie del frame a tutte le interfacce eccetto l'interfaccia $x$.
- È presente una voce corrispondente nella tabella, che associa l'indirizzo DD-DD-DD-DD-DD-DD con l'interfaccia $x$. Non serve imoltrare il frame ad altre interfacce, quindi viene scartato (filtering).
- È presente una voce corrispondente nella tabella, che associa l'indirizzo DD-DD-DD-DD-DD-DD con l'interfaccia $y \neq x$. Il frame viene inoltrato sul segmento della LAN corrispondente all'interfaccia $y$ (forwarding).

#### Self-Learning

La tabella di uno switch viene riempita automaticamente, dinamicamente e autonomamente.

1. La tabella è inizialmente vuota.
2. Per ogni frame in arrivo, lo switch memorizza nella sua tabella l'indirizzo MAC del mittente del frame, l'interfaccia dal quale è arrivato e quando.
3. Lo switch elimina dalla tabella un indirizzo se nessun frame con quell'indirizzo viene ricevuto in un certo lasso di tempo.

#### Proprietà

- *Eliminazione delle collisioni.* In una LAN gestita tramite switch non ci sono collisioni. Gli switch bufferizzano i frame e non trasmettono più di un frame su uno stesso segmento.
- *Collegamenti eterogenei.* Poiché uno switch isola segmenti di LAN l'uno dall'altro, questi possono avere velocità differenti o operare su architetture differenti.
- *Gestione.* Uno switch semplifica la gestione della rete. Se un'interfaccia di un dispositivo si guasta e continua a inviare frame in broadcast, uno switch può rilevare il problema e disconnettersi dall'interfaccia malfunzionante.

### 1.4.4 VLAN

Le reti LAN tradizionali sono affette da tre problemi:

- *Nessuna isolazione del traffico.*

- *Uso non efficiente degli switch.*
- *Gestione degli utenti.*