

Evoluzione delle architetture di rete e dei servizi di telecomunicazione

Che cos'è Internet?

- **Host** = sistema terminale
- Applicazioni di rete
- Collegamenti:
 - rame, fibra ottica, onde elettromagnetiche, satellite
 - Frequenza di trasmissione = ampiezza di banda
- **Router** = instrada i pacchetti verso la loro destinazione finale
- **Infrastruttura per applicazioni distribuite:**
 - Social networks, Web, VoIP, e-mail, giochi, e-commerce, condivisione di file
- **Servizi forniti alle applicazioni:**
 - Servizio affidabile dalla sorgente alla destinazione
 - Servizio "best effort" (non affidabile) senza connessione
- Un **protocollo** definisce il formato e l'ordine dei messaggi scambiati fra due o più entità in comunicazione:
 - es.: TCP, IP, HTTP, Skype, Ethernet
- Internet: "rete delle reti":
 - struttura gerarchica
 - Internet pubblica e intranet
- Standard Internet:
 - **RFC**: Request for comments
 - **IETF**: Internet Engineering Task Force

Cos'è un protocollo?

- Protocolli umani:
 - "Che ore sono?", "Ho una domanda", Presentazioni
- Protocolli di rete:
 - Dispositivi hardware e software
- Invio di specifici messaggi, quando il messaggio è ricevuto, vengono intraprese specifiche azioni, o si verificano altri eventi
- Tutta l'attività di comunicazione in Internet è governata dai protocolli



Struttura di rete

- Ai confini della rete:
 - Applicazioni
 - Sistemi terminali
- Reti, dispositivi fisici:
 - Collegamenti cablati
 - Wireless
- Al centro della rete:
 - Router interconnessi
 - La rete delle reti

Ai confini della rete

- Sistemi terminali (host):
 - Fanno girare programmi applicativi, es.: Web, e-mail
 - Situati all'estremità di Internet
- Architettura Client/Server:
 - L'host client richiede e riceve un servizio da un programma server in esecuzione su un altro terminale, es.: browser/server Web ; client/server e-client/server mail
- Architettura Peer To Peer:
 - Uso limitato (o inesistente) di server dedicati, es. Skype, Bit Torrent

Reti d'accesso e mezzi fisici

- **D:** Come collegare sistemi terminali e router esterni?
- Reti di accesso residenziale
- Reti di accesso aziendale (università, istituzioni, aziende)...
- Reti di accesso mobile

Ricordate:

- Ampiezza di banda (bit al secondo)?
- Condivise o dedicate?

Accesso residenziale: punto-punto

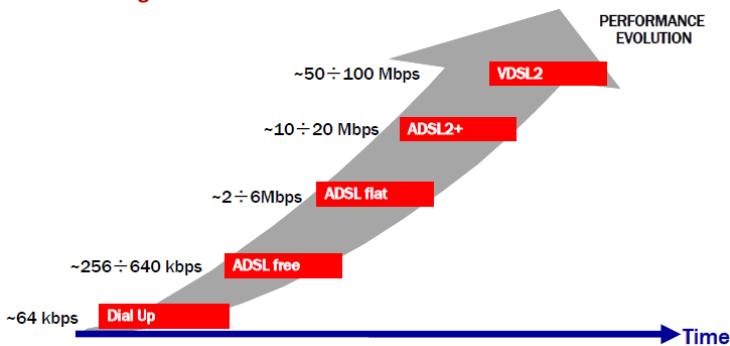
- **Modem dial-up:**

- Fino a 56 Kbps di accesso diretto al router (ma spesso è inferiore)
- Non è possibile "navigare" e telefonare allo stesso momento

- **DSL: digital subscriber line:**

- Installazione: in genere da un operatore di rete
- ~ 2 Mbps in upstream
- ~ 30 Mbps in downstream
- Linea dedicata

Accesso a Larga Banda di rete fissa



Rete di Distribuzione Telefonica

- **Obiettivo:**

- Trasporto e trattamento del segnale dalla centrale (SL) all'apparecchio del cliente

- È costituita da:

- Portanti fisici
- Attestazioni e terminazioni
- Apparati trasmisivi
- Altri dispositivi

- Si suddivide nelle seguenti sezioni:

- Rete Primaria (~ 1 km)
- Rete Secondaria (~ 200 m)
- Raccordo (~ 50 m)

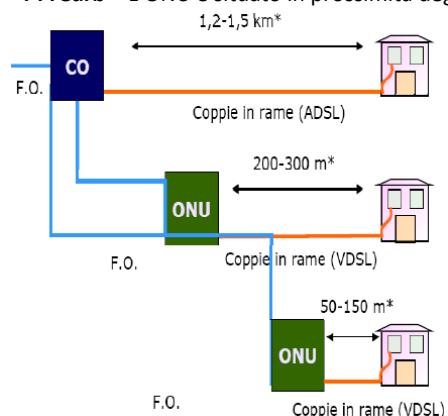


Architetture ibride rame-fibra (FTTx)

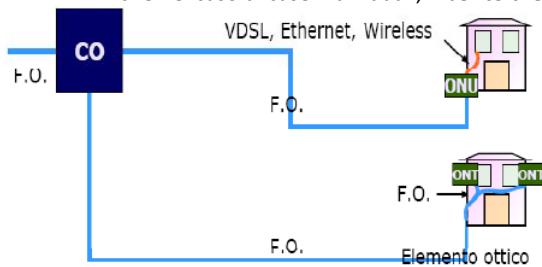
- **FTTE = Fiber to the Exchange**

• **FTTCab =** L'ONU è situato nell'armadio telefonico nelle strade, **Fiber to the Cabinet**

• **FTTCurb =** L'ONU è situato in prossimità degli edifici dove si trovano gli utenti, **Fiber to the Curb**



- **FTTB** = Nelle aree con edifici a sviluppo verticale, **Fiber to the Building**
- **FTTH** = Anche nel caso di case individuali, **Fiber to the Home**



Accesso aziendale: reti locali (LAN)

- Una LAN collega i sistemi terminali di aziende e università ad un router
- Ethernet:
 - 10 Mb/s, 100 Mb/s, 1 Gb/s, 10 Gb/s
 - Sistemi terminali collegati mediante uno switch

Accesso Wireless

- Una rete condivisa d'accesso wireless collega i sistemi terminali al router
 - Access Point (AP)
- Wireless LAN:
 - 802.11b/g (WiFi): 11 o 54 Mbps
- Rete d'accesso wireless geografica
 - Gestita da un provider di telecomunicazioni
 - ~ 1 Mbps per i sistemi cellulari (HSDPA)...
 - WiMax per aree più grandi

Reti domestiche

- Componenti di una tipica rete domestica
 - DSL o modem via cavo
 - Router/firewall/NAT
 - Ethernet
 - Punto d'accesso wireless

Mezzi trasmissivi

- **Mezzo fisico:**
 - Ciò che sta tra il trasmittente e il ricevente
- **Mezzi guidati:**
 - I segnali si propagano in un mezzo fisico: fibra ottica, filo di rame o cavo coassiale
- **Mezzi a onda libera:**
 - I segnali si propagano nell'atmosfera e nello spazio esterno
- **Twisted Pair (TP):**
 - Due fili di rame distinti:
 - Categoria 3: tradizionale cavo telefonico, 10 Mbps Ethernet
 - Categoria 5: 100 Mbps Ethernet

Mezzi trasmissivi: Cavo coassiale e fibra ottica

- **Cavo coassiale:**
 - Due conduttori in rame concentrici
 - Bidirezionale
 - Banda base:
 - Singolo canale sul cavo
 - Legacy Ethernet
 - Banda larga
- **Fibra ottica:**
 - Mezzo sottile e flessibile che conduce impulsi di luce
 - Alta frequenza trasmissiva:
 - Elevata velocità di trasmissione punto-punto (da 10 a 100 Gps)
 - Basso tasso di errore, immune all'interferenza elettromagnetica

Mezzi trasmissivi: Canali radio

- Trasportano segnali nello spettro elettromagnetico
- Non richiedono l'installazione fisica di cavi
- Bidirezionali
- Effetti dell'ambiente di propagazione:
 - Riflessione
 - Ostruzione da parte di ostacoli
 - Interferenza

- Tipi di canali radio:
 - Microonde terrestri:
 - Es. Canali fino a 45 Mbps
 - LAN (es. Wifi)
 - 11 Mbps, 54 Mbps
 - Wide-area (es. cellulari)
 - 3G: ~ 1 Mbps
 - Satellitari
 - Canali fino a 45 Mbps (o sottomultipli)
 - Ritardo punto-punto di 270 msec
 - Geostazionari/ a bassa quota

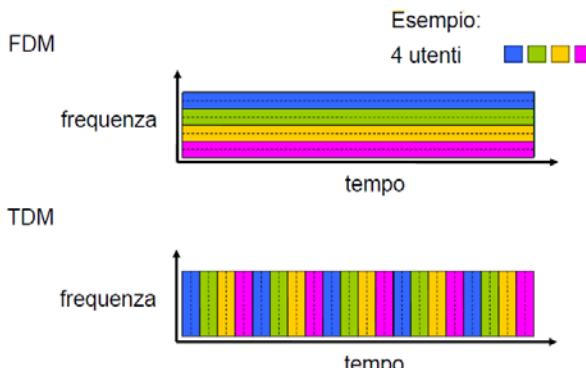
Il nucleo della rete

- Rete magliata da router che interconnettono i sistemi terminali
- Come vengono trasferiti i dati attraverso la rete?
 - **Commutazione di circuito:** Circuito dedicato per l'intera durata della sessione (rete telefonica)
 - **Commutazione di pacchetto:** I messaggi di una sessione utilizzano le risorse su richiesta, e di conseguenza potrebbero dover attendere per accedere a un collegamento

Commutazione di circuito (Circuit Switching – CS)

- Risorse punto-punto riservate alla "chiamata"
 - Ampiezza di banda, capacità del commutatore
 - Risorse dedicate: non c'è condivisione
 - Prestazioni da circuito (garantite)
 - Necessaria l'impostazione della chiamata
- Risorse di rete (banda) suddivise in "pezzi"
- Ciascun "pezzo" viene allocato ai vari collegamenti
- Le risorse rimangono inattive se non utilizzate (non c'è condivisione)
- Suddivisione della banda in "pezzi":
 - Divisione di frequenza
 - Divisione di tempo

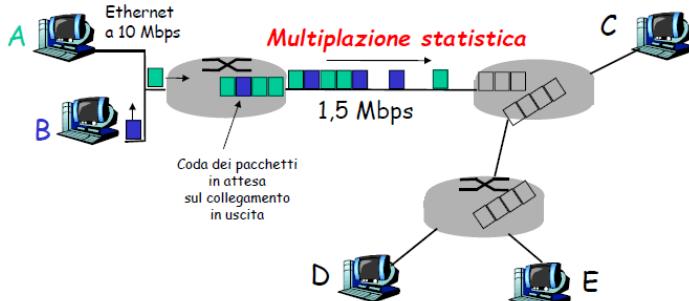
Commutazione di circuito: FDM e TDM



Commutazione di pacchetto (Packet Switching – PS)

- Il flusso di dati punto-punto viene suddiviso in pacchetti:
 - I pacchetti condividono le risorse di rete
 - Ciascun pacchetto utilizza completamente il canale
 - Le risorse vengono usate a seconda delle necessità
 - MULTIPLAZIONE STATISTICA
 - **NON** è necessario che:
 - La larghezza di banda venga suddivisa in prezzi
 - Allocazione dedicata
 - Risorse riservate
- Contesa per le risorse:
 - La richiesta di risorse può eccedere il quantitativo disponibile
 - **Congestione:** accodamento dei pacchetti, attesa per l'utilizzo del collegamento
 - **Store and forward:** il commutatore deve ricevere l'intero pacchetto prima di poter cominciare a trasmettere sul collegamento in uscita

Multiplicazione statistica



- La sequenza dei pacchetti A e B non segue uno schema prefissato.
- Condivisione di risorse su richiesta: multiplicazione statistica
- **TDM**: ciascun host ottiene uno slot di tempo dedicato unicamente a quella connessione.

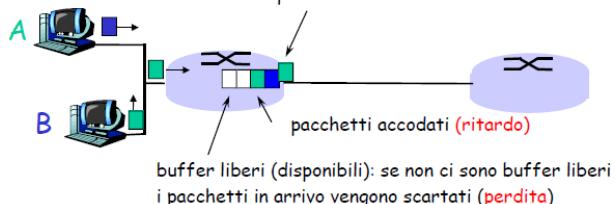
Confronto CS e PS

- La commutazione di pacchetto consente a più utenti di usare la rete:
 - 1 collegamento da 1 Mpbs
 - Ciascun utente usa 100 kbps quando è "attivo", attivo per il 10% del tempo
 - Commutazione di circuito:
 - 10 utenti
 - Commutazione di pacchetto:
 - con 35 utenti, la probabilità di averne > 10 attivi è inferiore allo 0,0004

- La commutazione di pacchetto è la "scelta vincente"?
 - Ottima per i dati a "burst":
 - Condivisione delle risorse
 - Più semplice, non necessita l'impostazione della chiamata
 - Eccessiva congestione: ritardo e perdita di pacchetti
 - Sono necessari protocolli per il trasferimento affidabile dei dati e per il controllo della congestione
 - D: Come ottenere un comportamento simile al circuito?
 - è necessario fornire garanzie di larghezza di banda per le applicazioni audio/video
 - è ancora un problema irrisolto

Ritardi e perdita

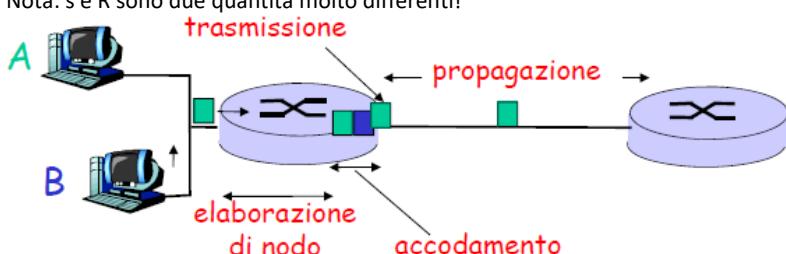
- I pacchetti si accodano nei buffer dei router
- Se il tasso di arrivo dei pacchetti eccede la capacità del collegamento i pacchetti si accodano, in attesa del proprio turno
pacchetti in attesa di essere trasmessi (**ritardo**)



Quattro cause di ritardo per i pacchetti

1. **Ritardo di elaborazione del nodo:**
 - a. Controllo errori sui bit
 - b. Determinazione del canale di uscita (instradamento)
2. **Ritardo di accodamento:**
 - a. Attesa di trasmissione
 - b. Livello di congestione del router
3. **Ritardo di trasmissione (L/R):**
 - a. R = frequenza di trasmissione del collegamento (in bps)
 - b. L = lunghezza del pacchetto (in bit)
 - c. Ritardo di trasmissione = L/R
4. **Ritardo di propagazione (d/s):**
 - a. d = lunghezza del collegamento fisico
 - b. s = velocità di propagazione del collegamento ($\sim 2 \times 10^8$ m/sec)
 - c. Ritardo di propagazione = d/s

Nota: s e R sono due quantità molto differenti!



Ritardo di link

$$d_{\text{link}} = d_{\text{elab}} + d_{\text{queue}} + d_{\text{trasm}} + d_{\text{prop}}$$

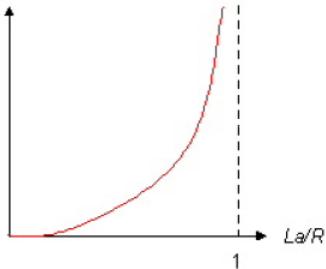
- d_{elab} = ritardo di elaborazione (processing delay)
 - In genere pochi microsecondi, o anche meno
- d_{queue} = ritardo di accodamento (queuing delay)
 - Dipende dalla congestione
- d_{trasm} = ritardo di trasmissione (transmission delay)
 - = L/R , significativo sui collegamenti a bassa velocità
- d_{prop} = ritardo di propagazione (propagation delay)
 - Da pochi microsecondi a centinaia di millisecondi

Ritardo di accodamento

- R = Frequenza di trasmissione (bps)
- L = Lunghezza del pacchetto (bit)
- a = tasso medio di arrivo dei pacchetti

$L \times a/R$ = intensità di traffico

- $L \times a/R \sim 0$: ritardo molto limitato
 - $L \times a/R \rightarrow 1$: il ritardo cresce in modo non lineare
 - $L \times a/R > 1$: più lavoro in arrivo di quanto possa essere effettivamente svolto, ritardo medio infinito
- average
queueing delay



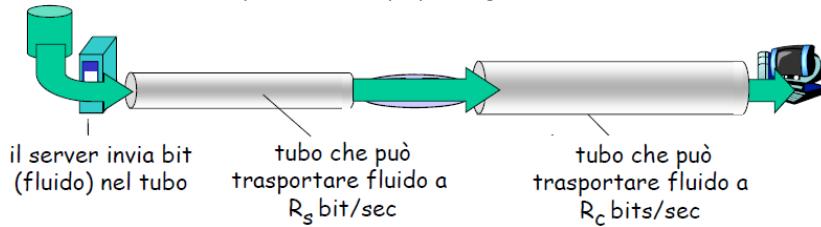
Perdita di pacchetti

- Una coda (detta anche buffer) ha capacità finita:

- Quando il pacchetto trova la coda piena, viene scartato (e quindi va perso)
- Un pacchetto perso può essere ritrasmesso dal nodo precedente, dal sistema terminale che lo ha generato, o non essere ritrasmesso affatto

Throughput

- Frequenza (bit/unità di tempo) alla quale i bit sono trasferiti tra mittente e ricevente:
 - Istantaneo: In un determinato istante
 - Medio: In un periodo di tempo più lungo



Collo di bottiglia (Bottleneck):

- Collegamento su un percorso punto-punto che vincola un throughput end to end

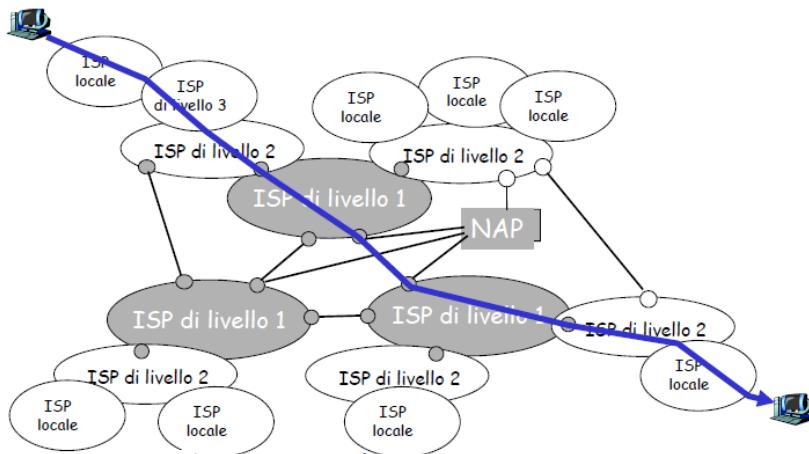
Throughput: scenario Internet:

- Throughput end to end per ciascuna connessione: $\min(R_c, R_s, R/10)$
- In pratica R_c o R_s è spesso nel collo di bottiglia

Struttura di Internet: rete di reti

- Fondamentalmente gerarchia
 - Al centro: "ISP di livello 1"
 - Verizon, Sprint, AT&T, Cable&Wireless
 - copertura nazionale/internazionale
 - Comunicano tra di loro come "pari"
- Gli ISP di livello 1 sono direttamente connessi a ciascuno degli altri ISP di livello 1
- ISP di livello 2: ISP più piccoli (nazionali o distrettuali)
 - Si può connettere solo ad alcuni ISP di livello 1 e possibilmente ad altri ISP di livello 2
- Un ISP di livello 2 paga l'ISP di livello 1 che gli fornisce la connettività per il resto della rete
- Un ISP di livello 2 è cliente di un ISP di livello 1
- Quando due ISP sono direttamente interconnessi vengono detti di pari grado (peer)

- ISP di livello 3 e ISP locali (ISP di accesso):
 - Reti “ultimo salto” (last hop network), le più vicine ai sistemi terminali
- ISP locali e di livello 3 sono clienti degli ISP di livello superiore che li collegano all’intera Internet
- Un pacchetto attraversa un numero anche molto elevato di reti



Elementi architetturali di una Computer Network

- Trasmissioni digitali
- Scambio di frames tra elementi di rete adiacenti:
 - Framing e error control
- Medium access control (MAC) regola l’accesso ai mezzi condivisi
- Indirizzi identificano il punto di accesso alla rete (interfaccia)
- Trasferimento dei pacchetti in rete
- Calcolo distribuito delle tabelle di routing
- Congestion control all’interno della rete
- Internetworking tra reti diverse
- Segmentazione e riassemblaggio dei messaggi in pacchetti all’ingresso e all’uscita da una rete
- Protocolli di trasporto end-to-end per comunicazioni tra processi
- Applicazioni che utilizzano le informazioni che attraversano la rete
- Intelligenza ai bordi della rete

Livelli di protocollo

- Le reti sono sistemi complessi:
 - Host
 - Router
 - Svariate tipologie di mezzi trasmissivi
 - Applicazioni
 - Protocolli
 - Hardware, Software
- Il processo complessivo è il prodotto di una sequenza di passi elementari
- Livelli: ciascun livello realizza un servizio
 - Effettuando determinate azioni all’interno del livello stesso, utilizzando i servizi del livello immediatamente inferiore

Perché la stratificazione?

- Quando si ha a che fare con sistemi complessi
 - Una struttura “esplicita” consente l’identificazione dei vari componenti di un sistema complesso e delle loro possibili interazioni
 - Modello di riferimento a strati
- La modularizzazione facilita la manutenzione e l’aggiornamento di un sistema
 - Modifiche implementative al servizio di uno dei livelli risultano trasparenti al resto del sistema
 - Es.: modifiche nelle procedure effettuate al gate non condizionano il resto del sistema
- Partiziona il processo di comunicazioni in parti indipendenti
- Semplifica il progetto, la realizzazione ed il test dei sistemi di telecomunicazione
- I protocolli:
 - Possono essere progettati separatamente ad ogni livello
 - Effettuano chiamate” ai servizi offerti dallo strato inferiore
 - Possono essere modificati senza cambiare i protocolli di altri strati sottostanti
- Architetture monolitiche sono costose, scarsamente flessibili e sono soggette a rapida obsolescenza

Open Systems Interconnection (OSI)

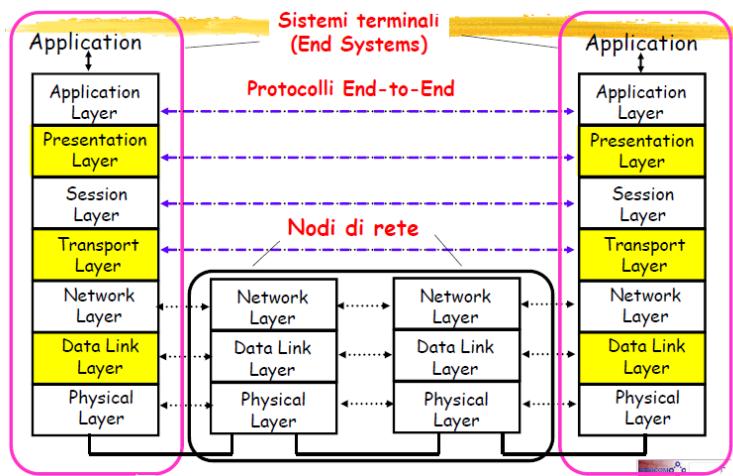
- Architettura di rete
 - Definizione dell’insieme degli strati
 - Definizione dei protocolli di ogni strato
- Dagli anni ’70 ogni produttore ha sviluppato la propria architettura a strati

- Problema
 - Computers di “vendor” diversi non possono essere interconnessi
- L’architettura OSI (Open Systems Interconnection) è stata creata per consentire l’interconnessione di sistemi “multivendor”

OSI Reference Model

- Describe un modello di riferimento a sette strati per l’architettura di una rete
- Fornisce un riferimento per lo sviluppo dei protocolli di comunicazione
- Il modello OSI definisce i concetti di **strato**, **protocollo** e **servizio** che hanno ancora oggi piena validità
- Sono stati definiti gli standard dei protocolli per i sette strati, ma nel tempo sono stati abbandonati
- Il modello a strati TCP/IP ha superato il modello OSI utilizzando esattamente gli stessi concetti

7-Layer OSI Reference Model



Physical Layer

- Ha lo scopo di trasferire i bit informativi sui mezzi trasmittivi
- Definizione delle caratteristiche fisiche di un link di comunicazione
 - Meccaniche
 - Tipo di cavi, connettori (plug, pin, ...)
 - Elettriche/ottiche
 - Modulazione, potenza dei segnali, livelli di tensione, temporizzazione, ...
 - Funzionali/procedurali
 - Procedure per attivare, mantenere e deattivare i link fisici
- Ethernet, xDSL, modem in banda fonica, ...
- Coppie in rame, cavi coassiali, fibre ottiche, mezzi radio, ...

Data Link Layer

- Ha lo scopo di realizzare il trasferimento affidabile delle informazioni in un link trasmittivo
- Formazione di unità dati denominate **trame (frame)**
- Rivelazione degli errori nelle trame ed eventuale loro correzione
- Attivazione, supervisione e deattivazione delle connessioni a livello di link
- **Funzioni MAC** (Medium Access Control) in reti locali (LAN)
- Controllo di flusso

Network Layer

- Trasferisce i **pacchetti** attraverso una serie di link o attraverso una serie di reti
- Gestisce **l’indirizzamento** di rete
- Definisce le procedure di **instradamento (routing)** eseguite dai nodi per la determinazione dei cammini di rete
- Definisce le procedure di **rilancio (forwarding)** dei pacchetti nei nodi
- Definisce le funzioni di **controllo di congestione**
- Definisce le procedure di setup, gestione e teardown delle connessioni di rete (modalità connection-oriented)

Internetworking

- Internetworking è una specifica funzione dello strato di rete, ha lo scopo di **gestire il trasferimento dei pacchetti attraverso una serie di reti diverse**
- I router rilanciano i pacchetti tra le reti

Transport Layer

- Trasferisce i dati end-to-end dal processo attivo in un host al processo residente nell’host remoto
- Garantisce **l’affidabilità** del trasferimento di stream di dati
- Offre un trasferimento rapido e semplice di singoli blocchi di dati
- Gestisce i “**port numbers**” (indirizzi interni ai sistemi terminali)
- Funzioni di **segmentation and reassembly** dei messaggi
- Connection setup, maintenance, and release

Application & Upper Layers

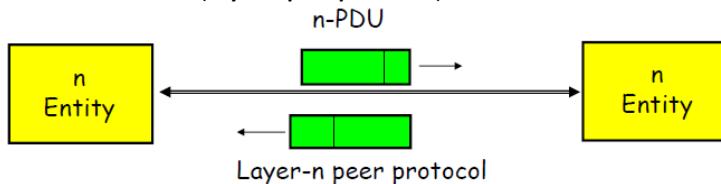
- Application Layer:
 - Fornisce i servizi richiesti dalle applicazioni
 - DNS, web access, file transfer, email...
- Presentation Layer:
 - Consente alle applicazioni di interpretare il significato dei dati (es. cifratura, compressione, convenzioni specifiche della macchina)
- Session Layer:
 - Sincronizzazione e controllo del dialogo,
 - Recupero dei dati

Pila di protocolli Internet

- **Applicazione:** supporta le applicazioni di rete
 - FTP, SMTP, HTTP
- **Trasporto:** Trasferimento dei messaggi a livello di applicazione tra il modulo client e server di un'applicazione
 - TCP, UDP
- **Rete:** trasferimento dei pacchetti dall'origine al destinatario
 - IP, protocolli di instradamento
- **Link (collegamento):** trasferimento dei pacchetti all'interno di una sottorete
 - PPP, Ethernet
- **Fisico:** trasferimento dei singoli bit

Concetto astratto di protocollo

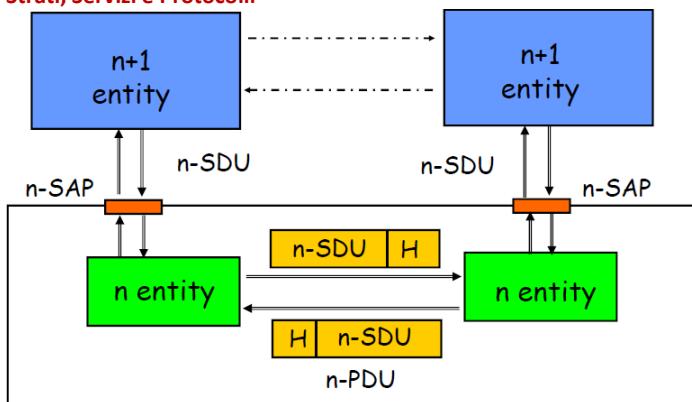
- Lo strato n in un sistema interagisce con lo strato n in un altro sistema per fornire servizio allo strato n+1
- Le entità che eseguono le funzioni di uno strato all'interno di sistemi comunicanti sono dette **peer processes**
- La cooperazione tra entità dello stesso strato è regolata dal protocollo di strato n (**layer-n protocol**)
- Le entità di strato n (**Layer-n peer processes**) si scambiano unità dati denominate **Protocol Data Unit (PDU)**



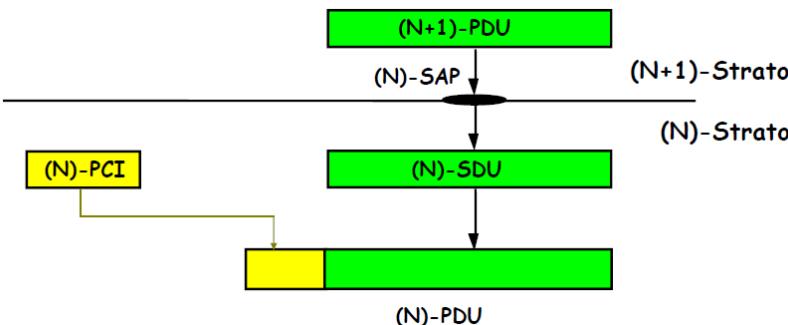
Concetto astratto di servizio

- La comunicazione tra due entità è **virtuale ed indiretta**
- Lo strato n+1 trasferisce le proprie informazioni invocando il servizio fornito dallo strato inferiore (strato n)
- I servizi sono dello strato n sono disponibili all'interfaccia tra i due strati (**Service Access Points - SAP**)
- Ogni strato passa dati e informazioni di controllo allo strato inferiore fino a che è raggiunto lo strato fisico che si occupa del trasferimento
- I dati che sono ricevuti da uno strato da quello superiore sono denominate **Service Data Unit (SDU)**
- Le SDU sono incapsulate nelle PDU nelle quali sono anche aggiunte le informazioni di controllo per l'esecuzione delle funzioni di strato

Strati, Servizi e Protocolli



Unità informative



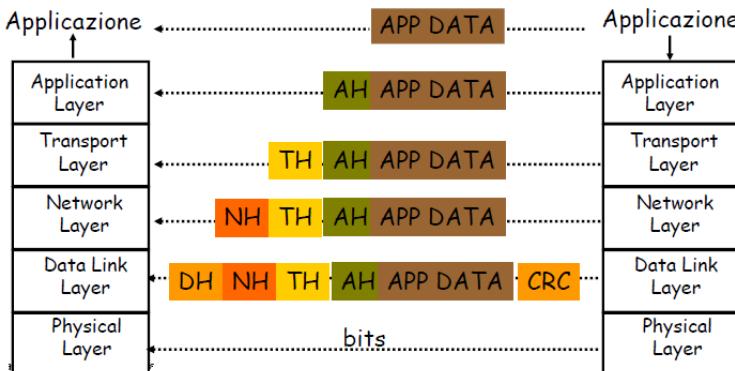
Segmentation & Reassembly

- Uno strato può impostare un limite massimo alla dimensione del blocco dati che può essere trasferito
- Se le n-SDU superano questo limite non possono essere trasferite in un'unica n-PDU
- Lato emittente: la SDU è segmentata in PDU multiple
- Lato ricevente: la SDU è riassemblata a partire dalla sequenza di PDU ricevute

Headers & Trailers

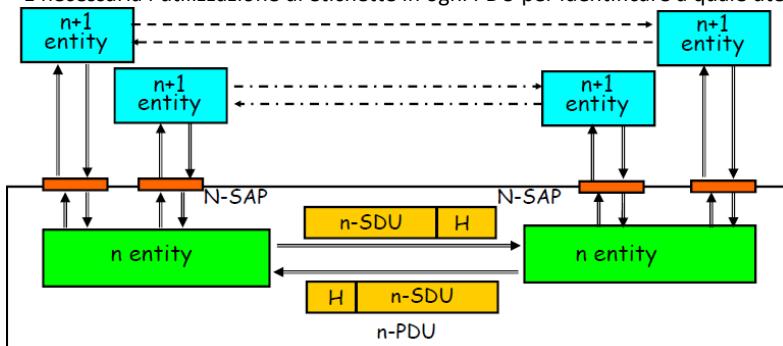
- Ogni protocollo usa un'intestazione (header) e un suffisso (trailer) che contengono le informazioni di controllo necessarie per l'esecuzione delle funzioni di strato

- Indirizzi, numeri di sequenza, flag, codici di controllo d'errore, ecc.



Multiplexing

- Condivisione del servizio di strato n da utenti multipli
- È necessaria l'utilizzazione di etichette in ogni PDU per identificare a quale utente appartiene la SDU



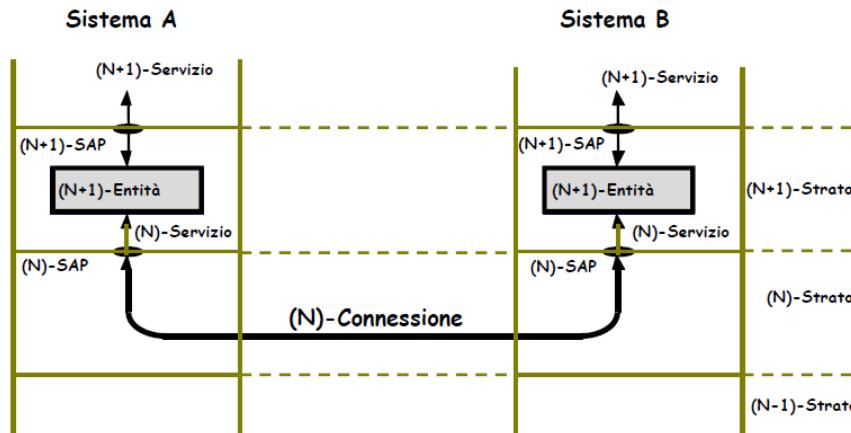
Modi di servizio

- Un servizio di strato dalle entità di strato superiore con o senza un'intesa preliminare
 - Nel caso in cui l'intesa sussista si parla di servizio con connessione (**connection oriented**)
 - Legame, almeno logico e in alcuni casi anche fisico, che viene stabilito tra le parti in comunicazione
 - Nel caso contrario si tratta di un servizio senza connessione (**connectionless**)

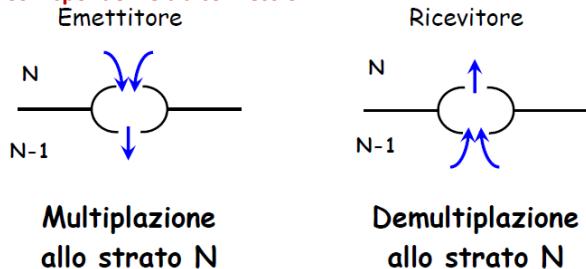
Servizio con connessione

- Servizio di strato **con connessione**
 - Strutturazione in **tre fasi temporali**
 - **Negoziazione** dei parametri di trasferimento
 - Indirizzamento con **identificatori di connessione**
 - **Legame logico** tra i segmenti informativi scambiati
- Es. TCP, Telefonia

Connessione di strato



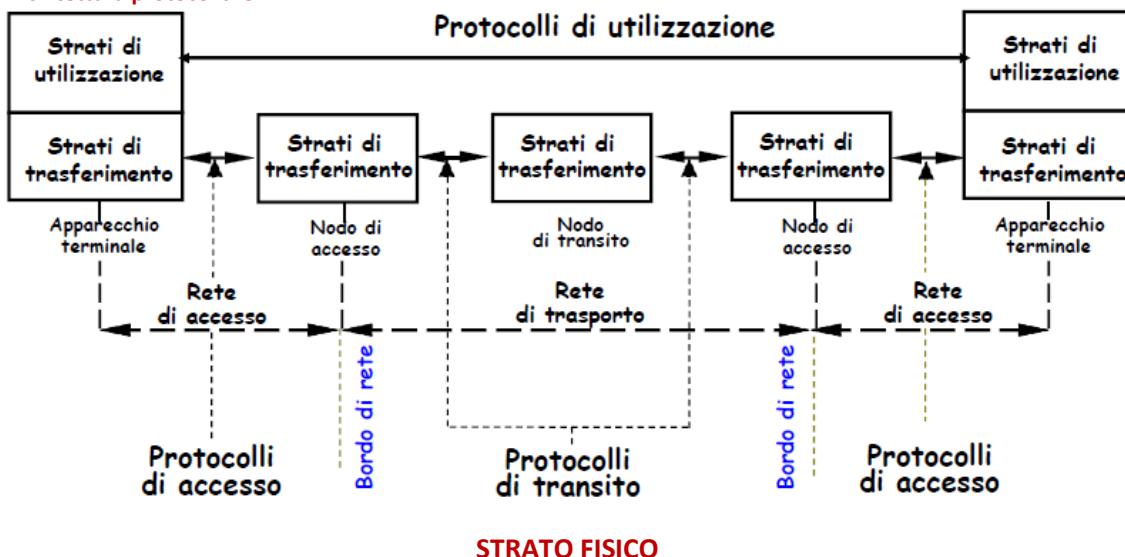
Corrispondenze tra connessioni



Servizio senza connessione

- Servizio di strato senza connessione
 - Una sola fase temporale
 - Assenza di negoziazione
 - Uso di indirizzi esplicativi per l'origine e la destinazione
 - Indipendenza e autoconsistenza dei segmenti informativi scambiati
- Es. IP, UDP

Architettura protocollare



Digital Networks

- Le tecniche di trasmissione digitale abilitano la rete al trattamento di qualsiasi flusso informativo

Informazione a Blocchi vs. Stream

- **Informazione a blocchi:**
 - L'informazione è naturalmente strutturata in unità indipendenti (blocchi)
 - Text message
 - Data file
 - JPEG image
 - MPEG file
- **Dimensione (size)**
 - Numero di bit (byte) per blocco
- **Informazione Stream:**
 - Informazione prodotta e trasmessa in modo continuo
 - Real-time voice
 - Streaming video
- **Bit rate**
 - Misura la quantità di bit prodotti dalla sorgente in una unità di tempo

Delay di trasferimento di un messaggio

- **L** = Numero di bit in un messaggio
- **R** = Velocità del sistema di trasmissione (bit/s)
- **t_{prop}** = tempo di propagazione lungo il mezzo trasmissivo
- **d** = lunghezza del collegamento
- **c** = velocità di propagazione sul mezzo trasmissivo (3x108 m/s nel vuoto, 2x108 m/s nei mezzi guidati)

- **L** si riduce mediante **tecniche di compressione**
- **R** si aumenta mediante adeguate **tecniche di trasmissione**
- **d** si riduce riducendo la lunghezza del collegamento (spesso impossibile...)

$$\text{Delay minimo} = t_{prop} + L/R = d/c + L/R$$

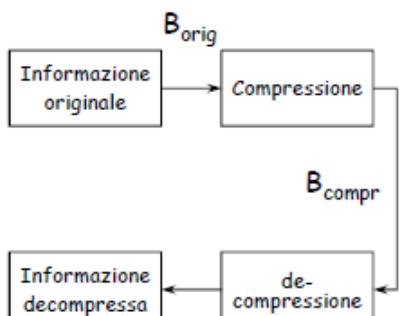
Compressione

- **Algoritmi di compressione dati:**

- Riducono il numero di bit necessari alla rappresentazione dell'informazione riducendo la ridondanza
- **Senza perdita (Lossless):** l'informazione originale è ricostruita esattamente
 - zip, GIF, fax
- **Con perdita (lossy):** l'informazione decompressa non è identica all'originale
 - JPEG

- **Rapporto di compressione (Compression Ratio) (Rc):**

- $R_c = B_{\text{orig}} / B_{\text{compr}}$ (#bits file originale / #bits file compresso)
- Compromesso tra numero di bit e qualità



$$R_c = \frac{B_{\text{orig}}}{B_{\text{compr}}} > 1$$

Esempi di informazione a blocchi

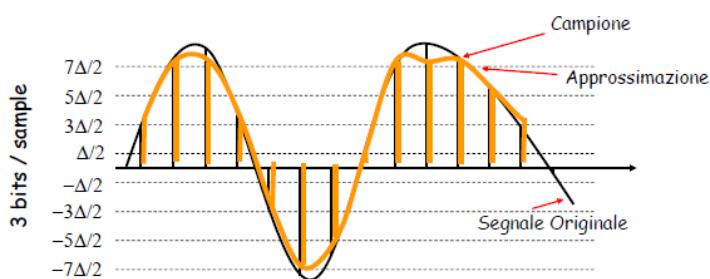
Tipologia	Metodo	Formato	Originale	Compressed Ratio
Text	Zip	ASCII	Kbyte-Mbyte	$2 < R_c < 6$
Fax	CCITT Group 3	A4 page 200x100 pixel/in ²	256 kbyte	5-54 kbyte ($5 < R_c < 50$)
Immagine a Colori	JPEG	8x10 in ² photo 400 ² pixel/in ²	38.4 Mbyte	1-8 Mbyte ($5 < R_c < 30$)

Stream Information

- Un segnale vocale nella forma originale è di tipo analogico
- Un segnale vocale deve essere digitalizzato e trasmesso in tempo reale: **Campionamento e Codifica (Sampling and Coding)**
- Il livello del **segnale analogico** varia nel tempo

Digitalizzazione di segnali analogici

- **Campionamento** (sampling) del segnale analogico nel tempo e **codifica** (coding) dell'ampiezza dei campioni
- Trovare la migliore approssimazione



$$R_s = \text{Bit rate} = \# \text{ bit/sample} \times \# \text{ sample/second}$$

Bit rate dei segnali digitalizzati

- Larghezza di banda (**Bandwidth**) Ws (Hz):
 - Indica quanto "velocemente" il segnale varia nel tempo
 - Maggiore bandwidth → campioni più frequenti
 - Frequenza di campionamento minima → $F_c = 2 \times W_s$
- **Accuratezza della rappresentazione:**
 - Maggiore accuratezza → numero maggiore di bit per campione (minore rumore di quantizzazione)

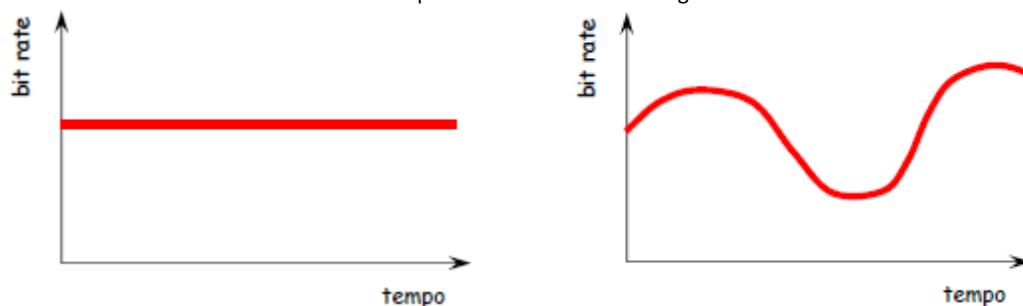
Segnale video

- Sequenza di "quadri" (picture frame):
 - Ogni picture è digitalizzata e compressa
- Frequenza di ripetizione delle frame:
 - 10-30-60 frame/sec in relazione all'obiettivo di qualità
- Risoluzione di ogni picture (Frame resolution):
 - Bassa risoluzione per servizio di videoconferenza
 - Risoluzione maggiore per servizio broadcast TV
 - HDTV frames

$$\text{Rate} = M \text{ bits/pixel} \times (W \times H) \text{ pixel/frame} \times F \text{ frame/second}$$

Tipologia di informazioni stream

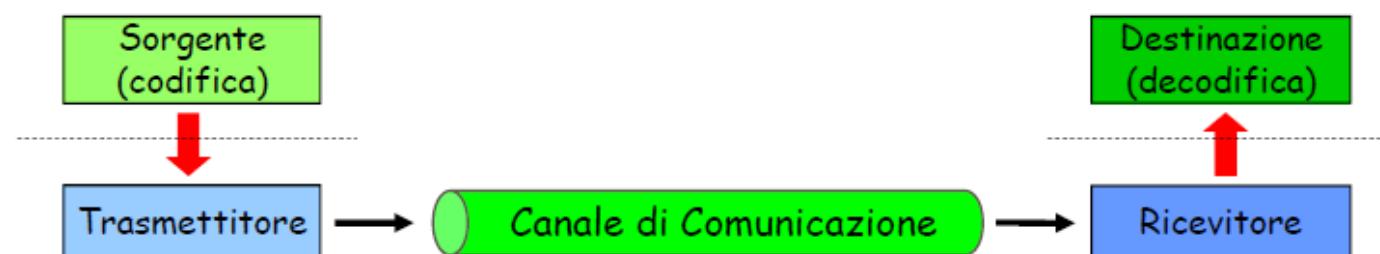
- Constant bit-rate (CBR):
 - Flussi informativi a bit rate costante
 - Es. sorgente telefonica produce un flusso stream a rate costante 64 kbit/s
 - La rete deve fornire un canale di comunicazione con banda almeno uguale al bit rate della sorgente
 - Es. Rete telefonica: canali di comunicazione (circuiti) a 64 kbit/s
- Variable bit-rate (VBR):
 - Flussi informativi con bit rate variabile nel tempo
 - Es. sorgente video a qualità costante produce un flusso in cui il bit rate varia in funzione del movimento tra due picture consecutive
 - La rete deve supportare in modo efficiente la variabilità del bit rate
 - Es. commutazione di pacchetto o rate-smoothing



Parametri di qualità per servizi di tipo Stream

- Possibili problemi introdotti dal transito in rete (Network Impairments):
 - Ritardo (Delay):
 - Per ogni servizio occorre individuare il vincolo sul ritardo massimo di attraversamento della rete
 - Variabilità del ritardo (Jitter):
 - Per ogni servizio occorre individuare il vincolo sulla variabilità massima consentita del ritardo di attraversamento della rete
 - Perdita di informazioni (Loss):
 - Per ogni servizio occorre individuare il vincolo sul percentuale massima di bit persi (per errori o congestione) sul totale dei bit trasmessi (**Probabilità di perdita**)
 - I protocolli di trasferimento sono progettati per gestire questi problemi

Schema di un sistema di trasmissione



• Trasmettitore:

- Converte il flusso informativo prodotto da una sorgente in un **segnale** adatto alla trasmissione
- Trasmette il segnale nel mezzo trasmittivo/canale di comunicazione

• Ricevitore:

- Riceve il segnale dal mezzo trasmittivo/canale di comunicazione
- Converte il segnale ricevuto in una forma utilizzabile dall'utente finale (destinazione)

• Canale di Comunicazione

- Coppie simmetriche
- Cavi coassiali
- Radio
- Fibra ottiche
- Light in air
- Infrarossi

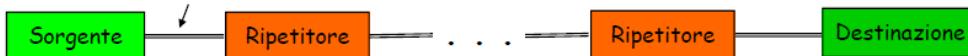
- **Transmission Impairments:**

- Attenuazione del segnale
- Distorsione del segnale
- Rumore additivo
- Interferenza con altri segnali

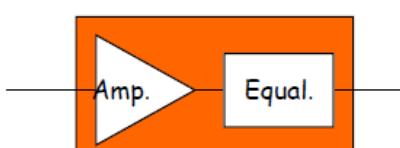
- I transmission impairments limitano la lunghezza del collegamento

Trasmissioni analogiche a lunga distanza

Tratta Tramssiva



- Ogni ripetitore ha lo scopo di rigenerare il segnale in uscita in modo che sia quanto più possibile simile a quello ricevuto in ingresso
- La rigenerazione è non ideale:
 - Le distorsioni non sono completamente eliminate
 - Il rumore e le interferenze sono solo parzialmente rimosse
- La qualità del segnale diminuisce al crescere del numero di ripetitori
- Le comunicazioni analogiche sono **distance-limited**
- Analogia:
 - Copie multiple di una cassetta musicale



Ripetitore ->

Trasmissione Analogica Vs Digitale

- **Trasmissioni analogiche:**

- Tutti i dettagli del segnale devono essere ricostruiti accuratamente



- **Trasmissioni numeriche:**

- Devono essere ricostruiti solo i livelli discreti del segnale
- L'impulso originale era positivo o negativo ?



Trasmissione numeriche a lunga distanza

Tratta Tramssiva

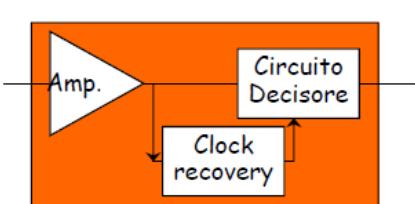


- **Un rigeneratore ricostruisce la sequenza iniziale di bit e la ritrasmette sulla tratta successiva**

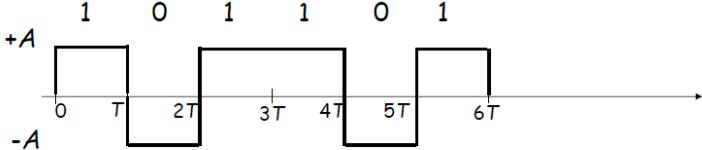
- È possibile progettare un rigeneratore in modo che la probabilità di errore sia piccola
- Il segnale rigenerato è in pratica identico a quello originale

- **Analogia:**

- Copie multiple di un file MP3
- Le comunicazioni numeriche sono possibili anche a lunghissima distanza
- **Sistemi numerici vs. sistemi analogici:**
 - Minore potenza, distanze maggiori, costi ridotti
 - Funzioni più semplici di monitoraggio, multiplazione, codifica, ecc.



Segnale numerico binario

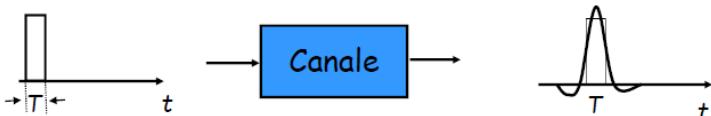


$$\text{Bit rate} = 1 \text{ bit} / T \text{ seconds} = 1/T$$

Trasmissione ad impulsi

- **Obiettivo:**

- Rendere massimo il rate di trasmissione degli impulsi in un canale, ovvero rendere T il più piccolo possibile



- Se in ingresso è trasmesso un impulso di breve durata, l'uscita sarà un impulso "allargato" e "arrotondato"

- Due impulsi consecutivi possono sovrapporsi tra loro

- **Domanda:**

- Qual è la frequenza massima F di trasmissione degli impulsi in modo che non ci sia interferenza tra loro?

- **Risposta:**

- $F = 2 \times W_c$ impulsi/secondo

- Dove W_c è la larghezza di banda del canale (Bandwidth)

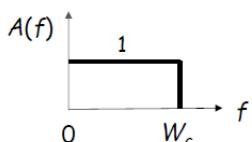
Larghezza di banda di un canale trasmissivo



- Se il segnale di ingresso ad un canale è una sinusoide di frequenza f allora:

- L'uscita sarà una sinusoide della stessa frequenza f
- Attenuata di un fattore $A(f)$ che dipende dalla frequenza f
- $A(f) \approx 1 \Rightarrow$ il segnale transita inalterato
- $A(f) \approx 0 \Rightarrow$ il segnale è bloccato

- La larghezza di banda W_c è definita come l'intervallo di frequenze per cui $A(f) \approx 1$



Canale passa basso ideale

Trasmissione ad impulsi multilivello

- Si consideri un canale con larghezza di banda W_c ad un rate $2W_c$ impulsi/s (senza interferenza)

- Se l'ampiezza degli impulsi può assumere due valori (-A o +A), ogni impulso può rappresentare un solo bit informativo, quindi:

- Bit Rate = 1 bit/impulso $\times 2W_c$ impulsi/sec = $2W_c$ bit/s

- Se l'ampiezza degli impulsi può assumere valori appartenenti all'insieme {-A, -A/3, +A/3, +A}, ogni impulso può rappresentare 2 bit quindi:

- Bit Rate = 2 bit/impulso $\times 2W_c$ impulsi/sec = $4W_c$ bit/s

- Se il segnale può assumere $M = 2^m$ livelli, si ha:

- Bit Rate = m bit/impulso $\times 2W_c$ impulsi/sec = $2mW_c$ bit/s

- In assenza di rumore il bit rate può essere incrementato aumentando il valore di m (livelli del segnale):

- **Attenzione:** aumentando m si riduce la distanza tra livelli adiacenti

Trasmissione multilivello (PAM)

- Raggruppa i bit in parole di dimensione $N=\log_2 M$

- **M:** numero di livelli

- **N:** numero di bit trasmessi in un unico impulso

- Assegna ad ogni parola di N bit un livello tra gli M disponibili:

- I livelli adiacenti corrispondono a parole di codice che differiscono per un solo bit (Codifica di Gray)

- Un errore tra due livelli adiacenti comporta un errore su un solo bit

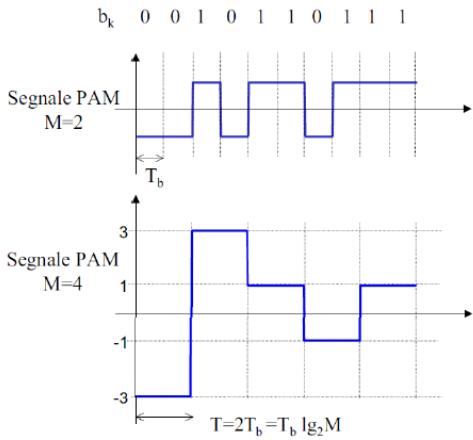
- Esempio:

- $N=3 \quad 000; 001; 011; 010; 110; 111; 101; 100$

Esempio

- Immaginiamo di voler trasmettere la sequenza binaria [0010110111], utilizzando un sistema PAM a $M=4$ livelli

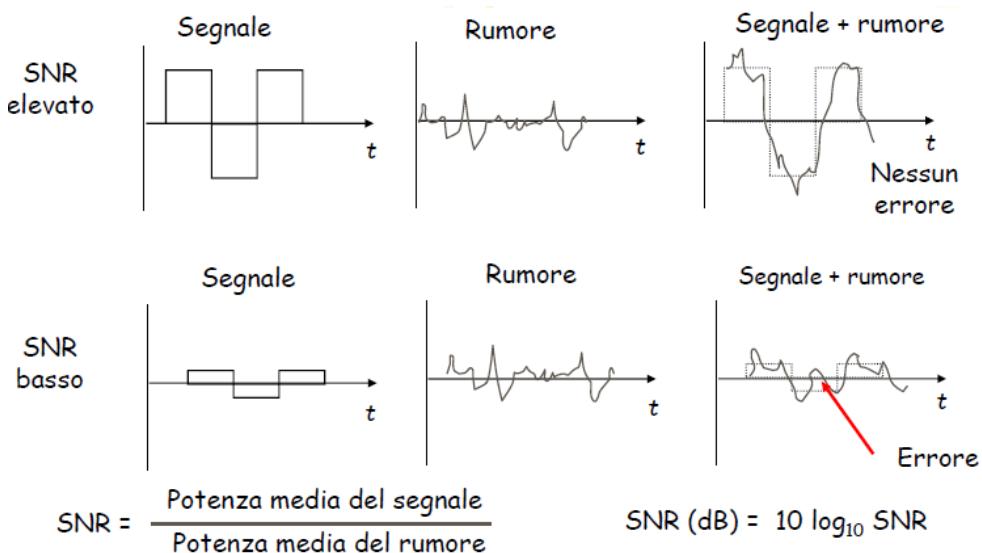
$b_{k-1} b_k$	a
0 0	-3
0 1	-1
1 1	+1
1 0	+3



Rumore

- Tutti i sistemi fisici introducono rumore:
 - Gli elettronni vibrano a temperature superiori allo zero assoluto, il moto degli elettronni introduce rumore
- La presenza di rumore limita l'accuratezza della misura dell'ampiezza del segnale ricevuto
- L'effetto del rumore è modellabile come un **segnale additivo** rispetto al segnale utile
- Una misura del rumore consiste nel **rappporto segnale-rumore (signal-to-noise ratio) (SNR)**
- Gli errori nella rivelazione del segnale ricevuto appaiono quando la separazione tra i livelli del segnale è comparabile con il livello di rumore
- Il **Bit Error Rate (BER)** aumenta quando diminuisce l'**SNR**
- Il rumore pone un limite al numero di livelli che possono essere utilizzati nella trasmissione di impulsi e quindi un limite al bit rate in trasmissione

Signal-to-Noise Ratio



Limite di Shannon alla capacità di un canale

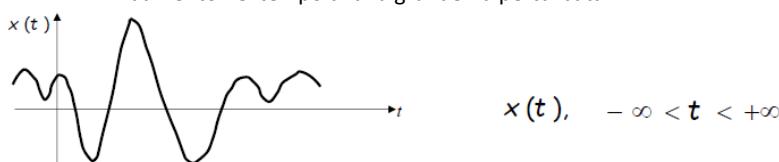
$$C_{\max} = W_c \cdot \log_2 (1 + \text{SNR}) \quad \text{bit/s}$$

- C_{\max} (capacità massima di canale) è una funzione della larghezza di banda e del rapporto segnale rumore
- Se il bit rate di trasmissione R è inferiore a C_{\max} ($R < C_{\max}$) è possibile ottenere un BER arbitrariamente piccolo:
 - è necessario introdurre una codifica di linea opportuna
- Se $R > C_{\max}$, non è possibile ridurre il BER a valori arbitrariamente piccoli
- La capacità C_{\max} può essere utilizzata come una misura di riferimento per stabilire quanto un sistema di trasmissione reale è vicino alle migliori prestazioni possibili

Rappresentazione dei segnali e teorema del campionamento

• Segnale analogico:

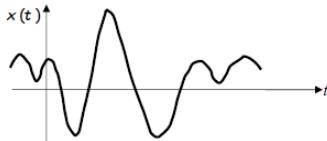
- Andamento nel tempo di una grandezza perturbata



- Esempi: Voce, temperatura ambiente, musica, televisione, tensione d'uscita di un microfono

Potenza di un segnale $x(t)$

$$P_x = \lim_{\Delta t \rightarrow +\infty} \frac{1}{\Delta t} \int_{-\Delta t/2}^{\Delta t/2} |x(t)|^2 dt \geq 0$$



- Un segnale è detto "di potenza" se $0 < P_x < +\infty$
- **Esempio (1):** segnale costante $x(t) = c$

$$P_x = \lim_{\Delta t \rightarrow +\infty} \frac{1}{\Delta t} \int_{-\Delta t/2}^{\Delta t/2} |c|^2 dt = |c|^2 \lim_{\Delta t \rightarrow +\infty} \frac{1}{\Delta t} \Delta t = |c|^2$$

- **Esempio (2):** segnale periodico sinusoidale

$$x(t) = A \cos(2\pi f_0 t + \varphi), \quad T = 1/f_0$$

$$\begin{aligned} P_x &= \frac{1}{T} \int_{-T/2}^{T/2} A^2 \cos^2(2\pi f_0 t + \varphi) dt = \frac{A^2}{T} \int_{-T/2}^{T/2} \frac{1}{2} [1 + \cos(4\pi f_0 t + 2\varphi)] dt = \\ &= \frac{A^2}{T} \int_{-T/2}^{T/2} \frac{1}{2} dt + \frac{A^2}{2T} \int_{-T/2}^{T/2} \cos(4\pi f_0 t + 2\varphi) dt = \\ &= \frac{A^2}{2T} \left(\frac{T}{2} + \frac{T}{2} \right) + 0 \quad (\text{il coseno ha area nulla}) = \frac{A^2}{2} \end{aligned}$$

Sviluppo in serie di Fourier per un segnale periodico

$$x(t) = x(t + T) = \sum_n g(t - nT)$$

- Segnale periodico, periodo T :
- Frequenza fondamentale: $F = 1/T$
- Armonica n -esima: $f_n = nF = n/T$

$$\begin{cases} x(t) = \sum_{n=-\infty}^{+\infty} X_n e^{j 2\pi f_n t} \\ X_n = \frac{1}{T} \int_{-T/2}^{T/2} x(t) e^{-j 2\pi f_n t} dt \end{cases}$$

Sviluppo in serie di Fourier

Coeffienti dello sviluppo

$$\{X_n\} = \{..., X_{-1}, X_0, X_1, ...\}$$

è una rappresentazione di $x(t)$

- Si osservi che:

$$e^{\pm j\varphi} = \cos \varphi \pm j \sin \varphi \quad \rightarrow \quad \cos \varphi = \frac{e^{j\varphi} + e^{-j\varphi}}{2}, \quad \sin \varphi = \frac{e^{j\varphi} - e^{-j\varphi}}{2j}$$

- Quindi:

$$\begin{aligned} X_n &= \frac{1}{T} \int_{-T/2}^{T/2} x(t) e^{-j 2\pi f_n t} dt = \\ &= \frac{1}{T} \int_{-T/2}^{T/2} x(t) \cos(2\pi f_n t) dt - j \int_{-T/2}^{T/2} x(t) \sin(2\pi f_n t) dt = R_n + j I_n = M_n e^{j \varphi_n} \end{aligned}$$

- Da cui:

$$\begin{aligned} R_n &= \frac{1}{T} \int_{-T/2}^{T/2} x(t) \cos(2\pi f_n t) dt = R_{-n} & M_n = \sqrt{R_n^2 + I_n^2} = M_{-n} \\ I_n &= -\frac{1}{T} \int_{-T/2}^{T/2} x(t) \sin(2\pi f_n t) dt = -I_{-n} & \varphi_n = \arctg \left(\frac{I_n}{R_n} \right) = -\varphi_{-n} \end{aligned}$$

$$X_{-n} = X_n^*$$

Sviluppo in serie di un segnale reale periodico

- $x(t)$ segnale periodico reale:

$$\begin{aligned} x(t) &= \sum_{n=-\infty}^{+\infty} X_n e^{j 2\pi f_n t} = X_0 + \sum_{n=1}^{+\infty} [X_n e^{j 2\pi f_n t} + X_{-n} e^{-j 2\pi f_n t}] = \\ &= R_0 + \sum_{n=1}^{+\infty} [M_n e^{j(2\pi f_n t + \varphi_n)} + M_{-n} e^{-j(2\pi f_n t + \varphi_{-n})}] = \\ &= R_0 + 2 \sum_{n=1}^{+\infty} M_n \cos(2\pi f_n t + \varphi_n) \end{aligned}$$

- Sviluppo con solo coseni di opportuna ampiezza e fase

Teorema di Parseval

- Potenza di un segnale periodico $x(t)$:

$$\begin{aligned} P_x &= \frac{1}{T} \int_{-T/2}^{T/2} |x(t)|^2 dt = \frac{1}{T} \int_{-T/2}^{T/2} \left| \sum_n X_n e^{j 2\pi f_n t} \right|^2 dt = \\ &= \frac{1}{T} \int_{-T/2}^{T/2} \left[\sum_n X_n e^{j 2\pi f_n t} \sum_n X_n^* e^{-j 2\pi f_n t} \right] dt = \\ &= \frac{1}{T} \int_{-T/2}^{T/2} \sum_n X_n X_n^* = \sum_n |X_n|^2 \end{aligned}$$

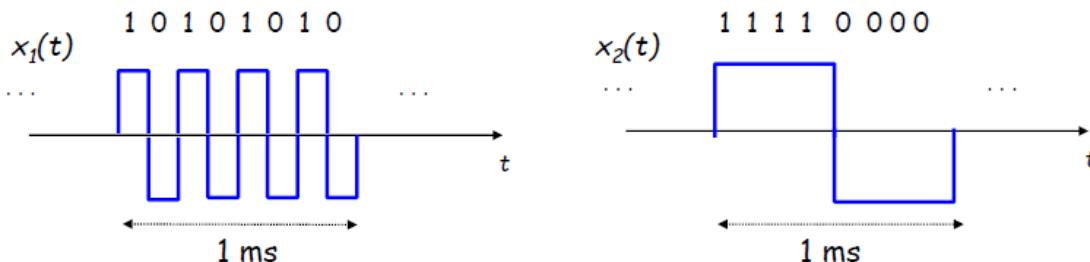
Ove $|X_n|^2$ è la potenza della singola armonica n/T

Digitalizzazione di segnali analogici

- Campionamento:** estrazione di campioni del segnale $x(t)$ uniformemente spaziati nel tempo
- Quantizzazione:** codifica di ogni campione con una stringa di bit (con precisione finita)
 - Telefonia: Pulse Code Modulation (PCM)
 - CD audio
- Compressione:** applicazione di metodi di riduzione del bit rate
 - Codifica differenziale: telefonia cellulare
 - Subband coding: MP3 audio

Frequenza di campionamento e larghezza di banda

- Segnali che variano più velocemente nel tempo devono essere campionati con maggiore frequenza
- Larghezza di banda (Bandwidth):** misura quanto velocemente varia un segnale



Segnali periodici

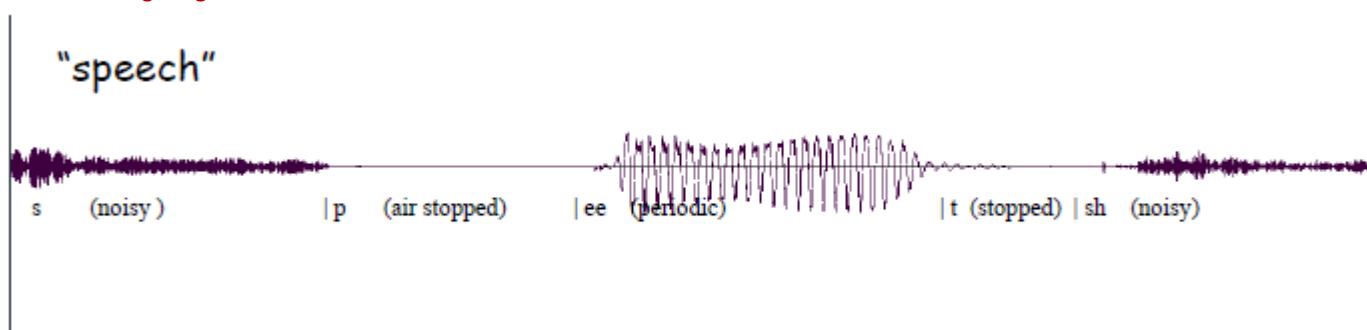
Un segnale reale periodico di periodo T può essere rappresentato come somma di sinusoidi usando lo sviluppo in serie di Fourier

$$x(t) = a_0 + a_1 \cos(2\pi f_0 t + \varphi_1) + a_2 \cos(2\pi 2f_0 t + \varphi_2) + \dots + a_k \cos(2\pi k f_0 t + \varphi_k) + \dots$$

"Componente continua";
 media a lungo termine Frequenza
 fondamentale $f_0 = 1/T$
 (prima armonica) k -ma armonica

- $|a_k|$ determina la potenza della k -ma armonica
- Spettro di ampiezza** = { $|a_0|, |a_1|, |a_2|, \dots$ }

Bandwidth di segnali generici

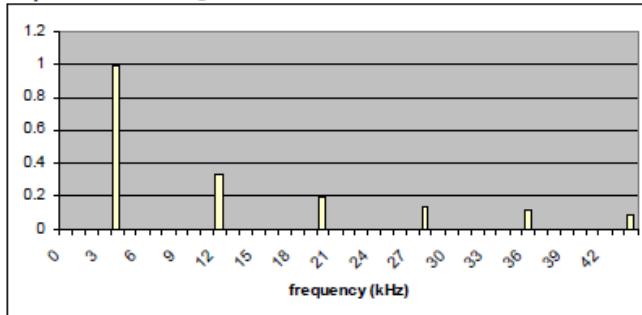


- Non tutti i segnali sono periodici:
 - Es. segnale vocale
- Per la determinazione dello spettro di un segnale generico si utilizza la **Trasformata di Fourier**
 - Segnale telefonico: 4 kHz
 - CD Audio: 22 kHz

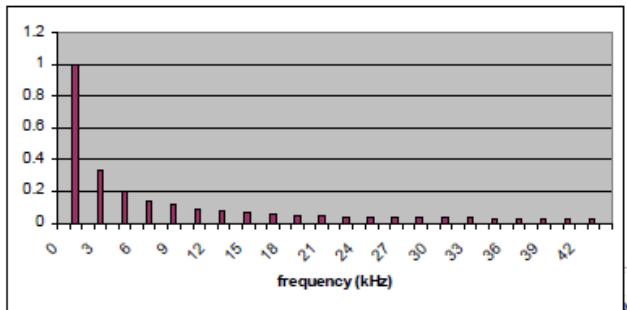
Spettro & Bandwidth di un segnale

- Lo Spettro di un segnale è rappresentato dalle ampiezze di ciascuna componente di frequenza
- $x_1(t)$ varia più velocemente nel tempo e quindi ha un contenuto di alte frequenze maggiore di $x_2(t)$
- La larghezza di banda (Bandwidth) W_s di un segnale è definita come l'intervallo di frequenze del segnale che hanno potenza non trascurabile:
 - Intervallo di banda che contiene il 99% della potenza totale del segnale

Spettro di $x_1(t)$



Spettro di $x_2(t)$



Trasformata di Fourier

- Dato un **segnale impulsivo** $x(t)$ per cui:

$$\int_{-\infty}^{\infty} |x(t)| dt < +\infty$$

- si ha:

$$FT : X(f) = \int_{-\infty}^{\infty} x(t) e^{-j 2\pi f t} dt = FT \{x(t)\} \quad -\infty < f < +\infty$$

$$FT^{-1} : x(t) = \int_{-\infty}^{\infty} X(f) e^{j 2\pi f t} df = FT^{-1}\{X(f)\} \quad -\infty < t < +\infty$$

- **$X(f)$ è una rappresentazione di $x(t)$ nel dominio della frequenza anziché del tempo**

Trasformata di Fourier di segnali reali

- Se $x(t)$ è un segnale reale:

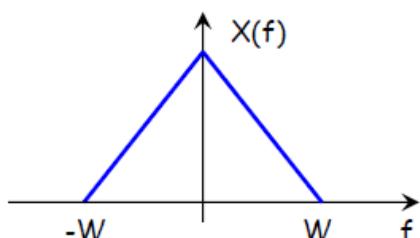
$$\begin{aligned} X(f) &= \int_{-\infty}^{+\infty} x(t) \cos(2\pi f t) dt - j \int_{-\infty}^{+\infty} x(t) \sin(2\pi f t) dt = \\ &= R(f) + j I(f) \end{aligned}$$

- Poiché:

$$\begin{aligned} R(f) &= R(-f) & I(f) &= -I(-f) \\ M(f) &= M(-f) & \varphi(f) &= -\varphi(-f) \end{aligned} \rightarrow \boxed{X(f) = X^*(-f)}$$

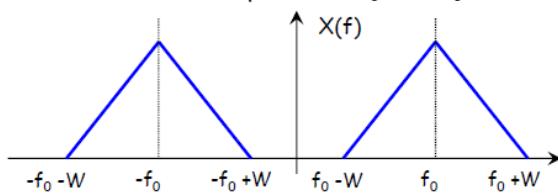
Larghezza di banda di un segnale

- Un segnale reale $x(t)$ si dice **limitato in banda** $[-W, W]$ se la sua trasformata di Fourier $X(f)$ è nulla per $f \notin [-W, W]$



- La quantità W è definita come la **Larghezza di Banda** del segnale $x(t)$
- Poiché $X(f) \neq 0$ in un intorno $[-W, W]$ di $f=0$, il segnale $x(t)$ si dice **segnale di banda base**
- Un segnale reale $x(t)$ si dice **limitato in banda, con banda $2W$ centrata intorno alla frequenza f_0** se:
 - $f_0 > W$

$X(f)$ è identicamente nulla per $f \in [-f_0 - W, -f_0 + W] \cup [f_0 - W, f_0 + W]$

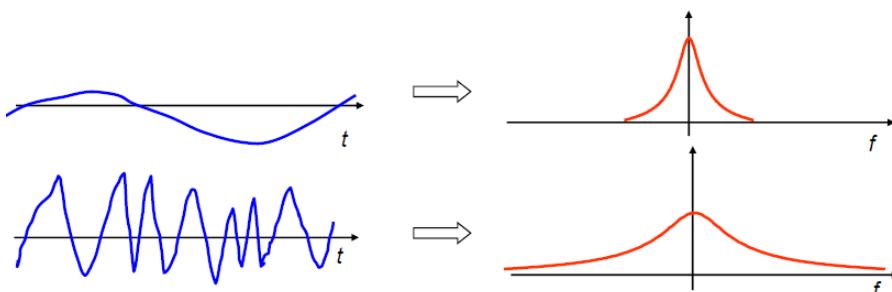


- La quantità $2W$ è la **larghezza di banda** del segnale $x(t)$
- Poiché $X(f) \neq 0$ in un intorno di $\pm f_1$ non adiacente all'origine, il segnale $x(t)$ si dice "segnale in banda traslata".

Relazioni tempo-frequenza

Segnali lentamente varianti in t → banda stretta (in f)

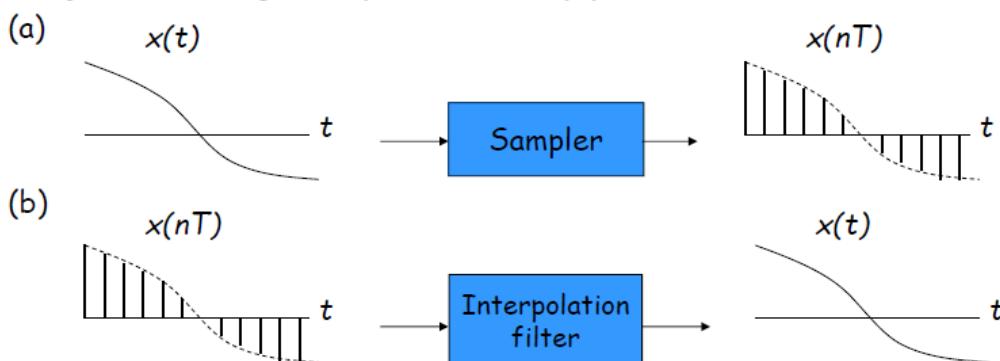
Segnali rapidamente varianti in t → banda larga (in f)



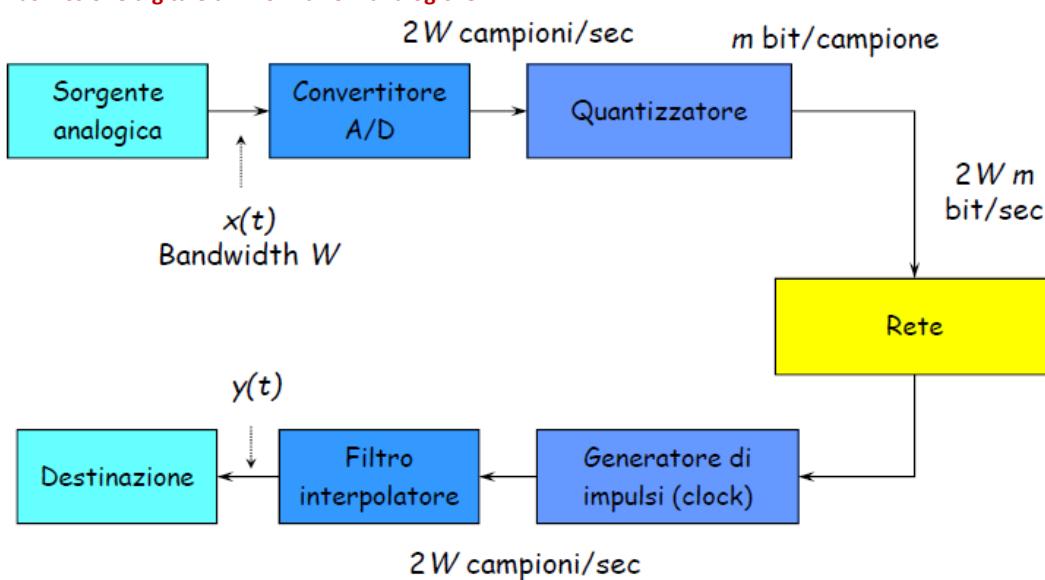
Teorema del campionamento

- Un segnale limitato in banda W_s può essere perfettamente ricostruito a partire dalla sequenza dei suoi campioni se la frequenza di campionamento

$$F_c = 1/T > 2W_s \text{ (Frequenza di Nyquist)}$$

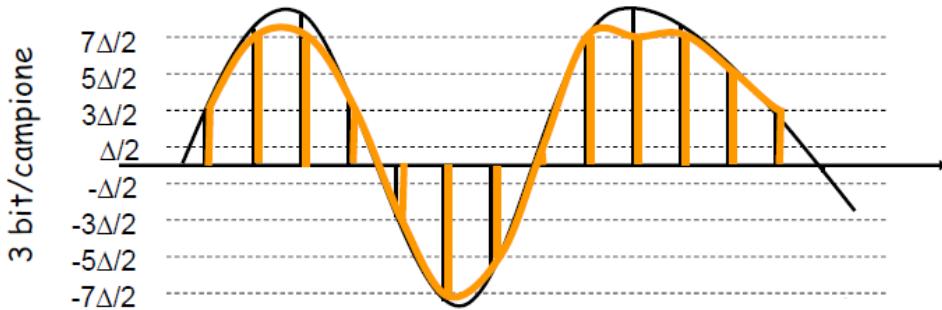
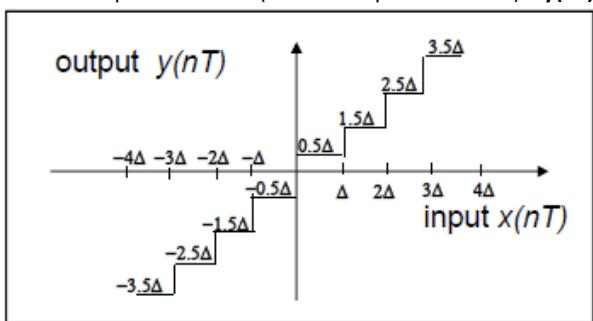


Trasmissione digitale di informazioni analogiche



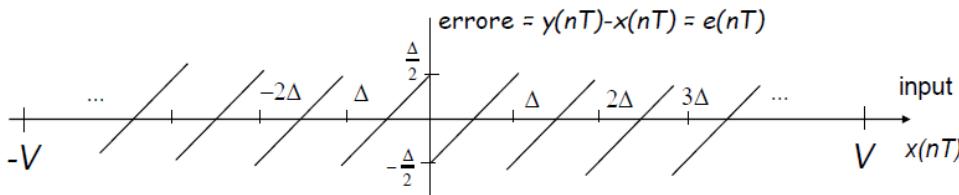
Quantizzazione di segnali analogici

- Il quantizzatore associa il valore di ingresso al valore 2^m (livello) più vicino
- Errore di quantizzazione (rumore di quantizzazione) = $y(nT) - x(nT)$



Prestazioni del quantizzatore

$$M = 2^m \text{ livelli}, \quad \text{Dynamic range}(-V, V); \quad \Delta = 2V/M$$



- Se il numero di livelli M è sufficientemente elevato, allora l'errore è uniformemente distribuito tra $(-\Delta/2, \Delta/2)$
- Potenza media del rumore di quantizzazione (Errore quadratico medio)

$$\sigma_e^2 = \frac{1}{\Delta} \int_{-\Delta/2}^{\Delta/2} x^2 dx = \frac{\Delta^2}{12}$$

Figura di merito:

- Signal-to-Noise Ratio (SNR) = Potenza media del segnale / Potenza media di rumore

- Sia σ_x^2 potenza del segnale si ha:

$$SNR = \frac{\sigma_x^2}{\sigma_e^2} = \frac{12\sigma_x^2}{\Delta^2/12} = \frac{12\sigma_x^2}{4V^2/M^2} = 3 \left(\frac{\sigma_x}{V} \right)^2 M^2 = 3 \left(\frac{\sigma_x}{V} \right)^2 2^{2m}$$

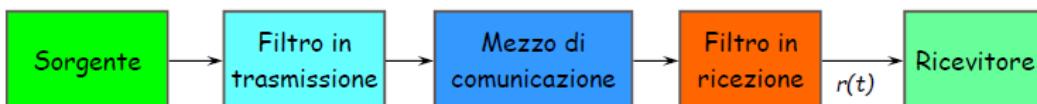
- Usualmente $V/\sigma=4$, quindi esprimendo SNR in dB

$$SNR (dB) = 10 \log_{10} \frac{\sigma_x^2}{\sigma_e^2} = 10 \log_{10} (3 \cdot 4^{-2} \cdot 2^{2m}) = 6m - 7.27 \quad dB$$

Caratterizzazione dei canali di comunicazione e limiti fondamentali delle comunicazioni digitali

Canali di comunicazione

- Per **canale di comunicazione** si intende l'unione dei mezzi trasmissivi e dei dispositivi (elettronici o ottici) che sono attraversati dal segnale lungo il percorso tra sorgente e destinazione:
 - Equalizzatori, amplificatori, ecc.
- Spesso si usa il termine **filtro** per indicare gli effetti del canale sul segnale che lo attraversa

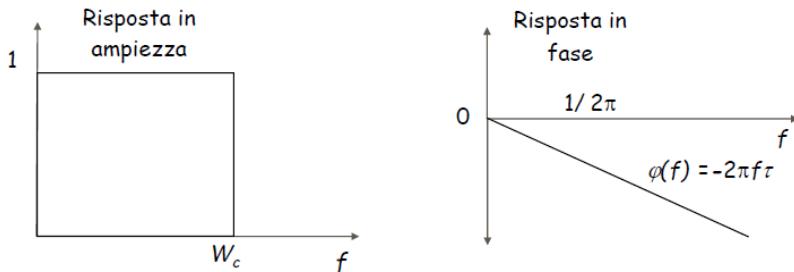


Filtro passa basso ideale

- Filtro passa basso ideale:

- Tutte le frequenze $f < W_c$ non subiscono attenuazione e sono ritardate di τ secondi
- Le frequenze $f > W_c$ sono bloccate

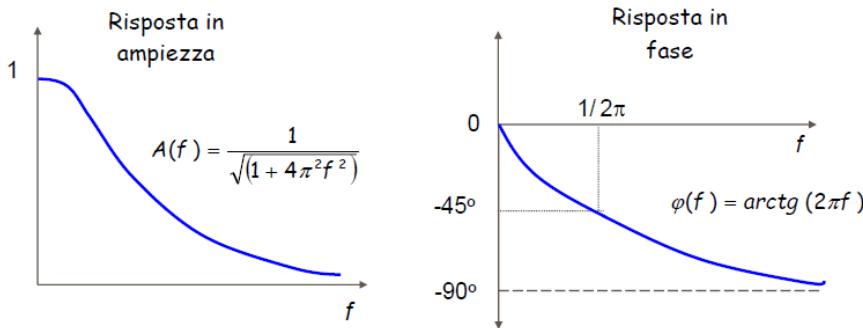
$$y(t) = A_{in} \cos(2\pi f t - 2\pi f \tau) = A_{in} \cos(2\pi f(t - \tau)) = x(t - \tau)$$



Filtro passa basso reale

- Filtro passa basso reale:

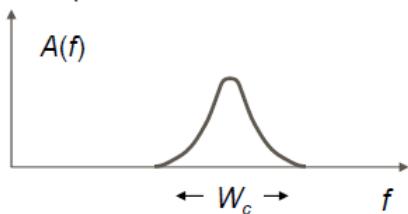
- Le frequenze sono attenuate in modo diverso e subiscono ritardi diversi



Canale passabanda

- Alcuni canali di comunicazione si comportano come un filtro passa-banda
 - Bloccano le basse e le alte frequenze
- La larghezza di banda è l'ampiezza dell'intervallo di frequenze per cui il segnale in uscita ha una potenza non trascurabile

Amplitude Response



Distorsione

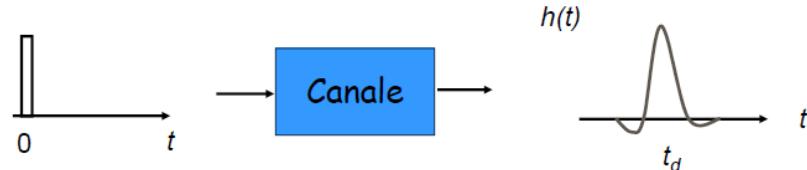
$$x(t) = \sum a_k \cos(2\pi f_k t + \theta_k) \rightarrow \text{Canale} \rightarrow y(t)$$

- Il canale introduce sul segnale in ingresso $x(t)$ due effetti:

- Se la risposta in frequenza non è "piatta", le componenti di frequenza del segnale d'uscita $y(t)$ avranno ampiezza diversa rispetto a quelle del segnale d'ingresso $x(t)$
- Se la risposta in fase non è "piatta", le componenti di frequenza del segnale d'ingresso $x(t)$ subiranno ritardi diversi

$$y(t) = \sum A(f_k) a_k \cos[2\pi f_k t + \theta_k + \phi(f_k)]$$

Caratterizzazione nel dominio del tempo



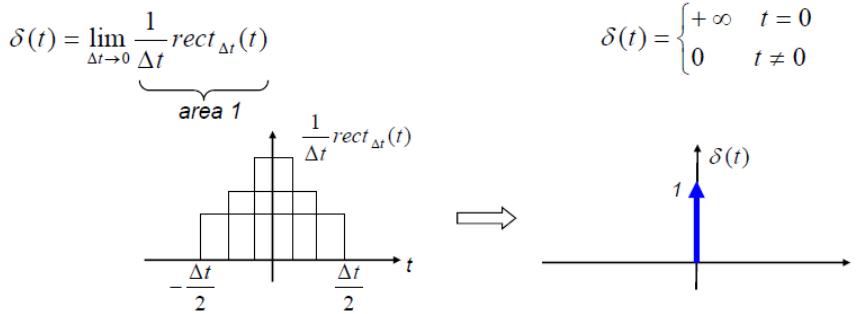
- La caratterizzazione di un canale nel dominio del tempo richiede la conoscenza della **risposta impulsiva $h(t)$** :

- Si applica in ingresso al canale un impulso di durata molto breve si osserva il segnale in uscita
- Tipicamente $h(t)$ è una copia ritardata e distorta dell'impulso in ingresso

- La larghezza della risposta impulsiva fornisce un'indicazione di quanto velocemente l'uscita segue l'ingresso e quindi di quanto velocemente possono essere trasmessi gli impulsi in ingresso

L'impulso matematico

- Rappresenta un segnale di durata brevissima (al limite, zero) e di ampiezza elevatissima (al limite, infinita) il cui integrale è unitario



• Proprietà:

- L'impulso matematico ha area unitaria

$$\int_{-\infty}^{+\infty} \delta(t) dt = 1$$

$$\int_{t_0-\varepsilon}^{t_0+\varepsilon} \delta(t - t_0) dt = 1, \quad \text{per ogni } \varepsilon > 0$$

- Proprietà di campionamento dell'impulso matematico

$$\int_{-\infty}^{+\infty} x(t) \delta(t - t_0) dt = x(t_0)$$

Risposta impulsiva di un sistema lineare



- La risposta impulsiva $h(t)$ di un sistema lineare e permanente (filtro) è definita come l'uscita $y(t)$ del sistema quando all'ingresso è applicato l'impulso unitario $x(t) = \delta(t)$

- Proprietà elementari di $h(t)$:

permanenza

$$x(t) = \delta(t - t_0) \rightarrow y(t) = h(t - t_0)$$

linearità

$$x(t) = a\delta(t - t_0) + b\delta(t - t_1) \rightarrow y(t) = ah(t - t_0) + bh(t - t_1)$$

Convoluzione

- Se un filtro è LP con risposta impulsiva $h(t)$, allora l'uscita $y(t)$ corrispondente ad un generico segnale di ingresso $x(t)$ è pari a:

$$y(t) = \int_{\tau=-\infty}^{\tau=+\infty} x(\tau)h(t - \tau)d\tau = x(t) * h(t), \quad -\infty < t < +\infty$$

- L'integrale precedente è detto **integrale di convoluzione** tra l'ingresso $x(t)$ e la risposta impulsiva $h(t)$ del filtro

- L'operazione di convoluzione è **commutativa**:

$$x(t) * h(t) = h(t) * x(t)$$

- L'operazione di convoluzione è **associativa**:

$$[x(t) * h(t)] * z(t) = x(t) * [h(t) * z(t)]$$

- L'operazione di convoluzione è **distributiva** rispetto alla somma:

$$[x(t) + z(t)] * h(t) = [x(t) * h(t)] + [z(t) * h(t)]$$

- La convoluzione di $x(t)$ con $\delta(t - t_0)$ trasla $x(t)$ di t_0

$$x(t) * \delta(t - t_0) = x(t - t_0)$$

Risposta in frequenza di un filtro



- La trasformata di Fourier della convoluzione è uguale al prodotto delle trasformate:

$$y(t) = x(t) * h(t) \longrightarrow Y(f) = X(f)H(f)$$

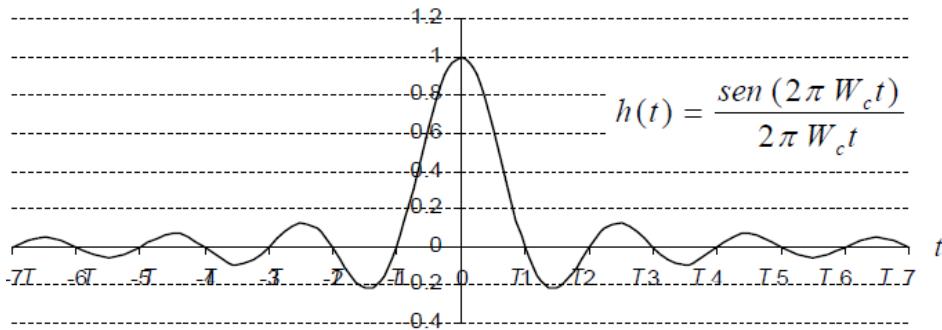
dove $H(f) = \text{FT}\{h(t)\}$

- $H(f)$ è detta **risposta in frequenza** del filtro o **funzione di trasferimento** del filtro

Risposta impulsiva di un filtro ideale

- Per canali ideali passa basso di larghezza di banda W_c , la risposta impulsiva è rappresentata dalla **funzione impulso di Nyquist** $h(t) = s(t - \tau)$, dove $T = 1/2 W_c$, e $s(t)$ vale zero in $t = kT$, $k = \pm 1, \pm 2, \dots$

- Gli impulsi possono essere inviati ogni T secondi senza interferenza (si sovrappongono in corrispondenza degli zeri)



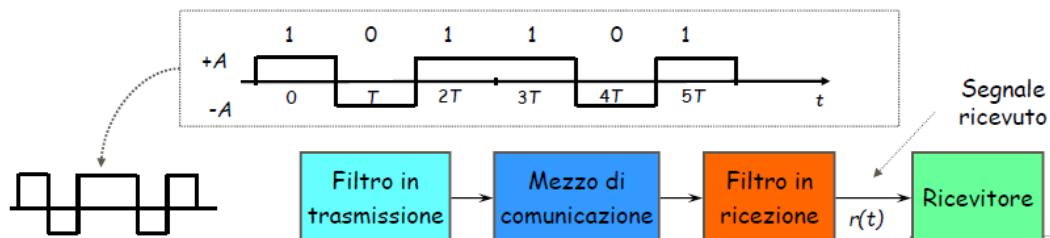
Trasmissione in banda base

Segnale PAM (Pulse Amplitude Modulation)

$$r(t) = \sum_k A_k p(t - kT)$$

$$r(0) = A_0 p(0) + \sum_{k \neq 0} A_k p(-kT)$$

- Sia $p(t)$ il segnale ricevuto dal ricevitore in risposta alla trasmissione di un singolo impulso
- Sia $r(t)$ il segnale che viene ricevuto a seguito della trasmissione di una sequenza di impulsi
- In generale se si campiona il segnale $r(t)$ negli istanti $t = kT$ il valore del campione è alterato dalla presenza di Interferenza Intersimbolica (ISI); ad esempio per $t = 0$
- Se $p(t) = s(t)$, quindi $p(t)$ sono impulsi di Nyquist, il segnale $r(t)$ ha ISI nulla negli istanti $t = kT$



Trasmissione in banda base

- Se il canale si comporta come un filtro passa basso ideale con larghezza di banda W_c , il massimo rate di trasmissione di una sequenza di impulsi senza ISI è uguale a $2W_c$ (**Nyquist Signalling Rate**):

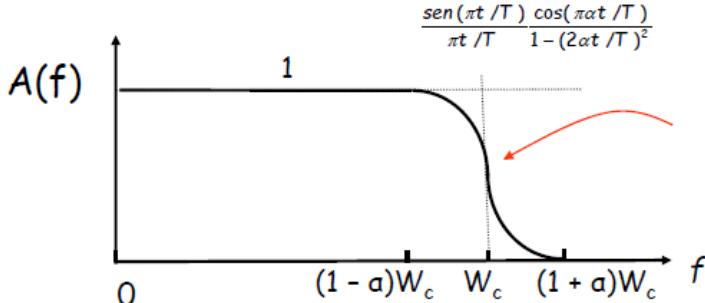
$$r_{\max} = 2W_c$$

- Si noti che $s(t)$ è un esempio della classe degli impulsi di Nyquist con ISI nulla:
 - L'ampiezza dei lobi laterali di $s(t)$ può causare errori anche notevoli se si commettono errori anche piccoli negli istanti di campionamento del segnale
 - Richiede una **sincronizzazione** molto accurata

- La funzione **coseno rialzato (raised cosine)** è un ulteriore esempio di funzione a zero ISI:

- Richiede una banda leggermente superiore a W_c

- I lobi laterali decadono come $1/t^3$ e quindi è più robusta ad errori di temporizzazione (**sincronizzazione meno accurata**)

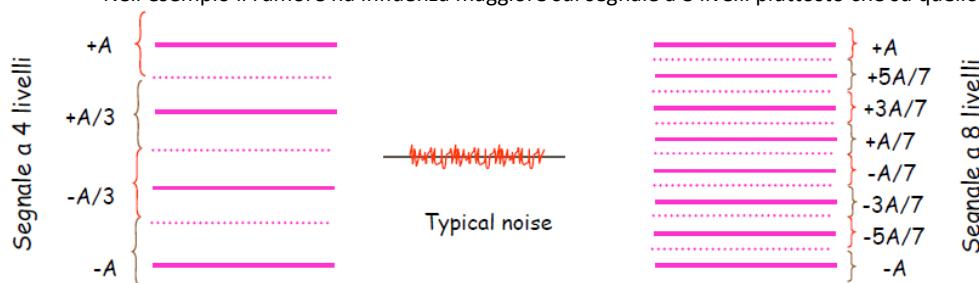


Trasmissione multilivello

- Il criterio di Nyquist impone che il massimo rate in trasmissione con ISI=0 sia:
 - $2W_c$ impulsi al secondo
 - $2W_c$ impulsi/ W_c Hz = 2 impulsi/Hz
- Se si usano due livelli di segnale ogni impulso trasporta 1 bit informativo:
 - Bit rate = $2W_c$ bit/s
- Con $M = 2^m$ livelli, ogni impulso trasporta m bit
 - Bit rate = $2W_c$ impulsi/s * m bit/impulso = $2W_c m$ bit/s
- Il bit rate può essere aumentato incrementando il numero di livelli, tuttavia ...
 - Il segnale $r(t)$ include il rumore additivo che limita il numero di livelli che possono essere usati

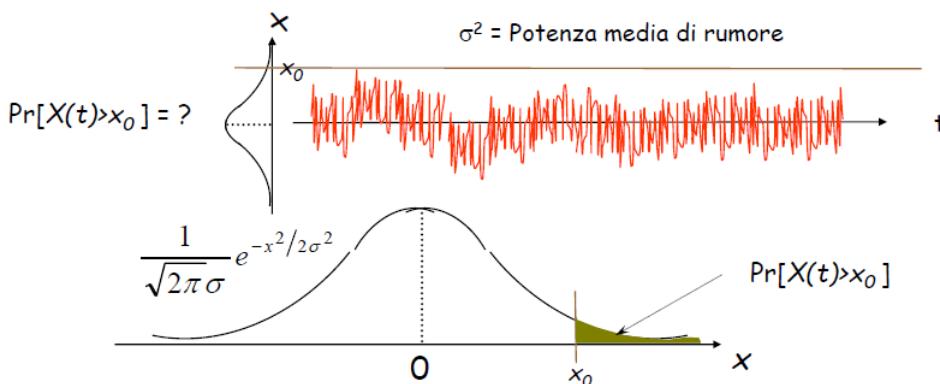
Effetto del rumore

- Il ricevitore prende le decisioni in base al segnale che è la somma dell'impulso trasmesso + rumore
- Il tasso di errore dipende dal valore relativo dell'ampiezza del rumore rispetto alla spaziatura tra i livelli
- Grandi valori di rumore possono comportare decisioni errate:
 - Nell'esempio il rumore ha influenza maggiore sul segnale a 8 livelli piuttosto che su quello a 4 livelli



Caratterizzazione del rumore

- Il **rumore termico** è inevitabile
- Il rumore può essere caratterizzato mediante la densità di probabilità dell'ampiezza dei campioni
- La distribuzione del rumore è Gaussiana

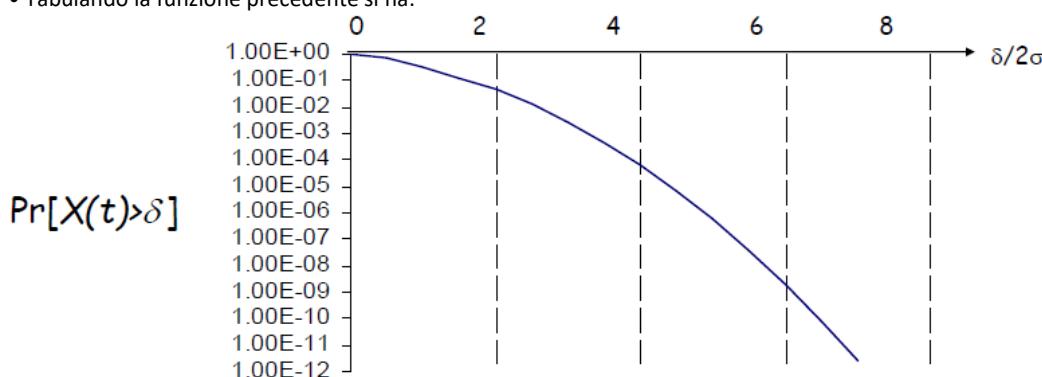


Probabilità di errore

- Un errore accade se il valore di rumore supera un determinato valore di ampiezza
- Si osservi che la probabilità di avere grandi valori di rumore decade rapidamente con la distribuzione Gaussiana
- In una trasmissione a M livelli di un segnale di ampiezza $[-A; A]$, la separazione δ tra livelli adiacenti è uguale a $\delta = 2A/(M-1)$
- La probabilità d'errore (P_e) è data dalla probabilità che il rumore superi il valore $\delta/2$ o sia inferiore a $-\delta/2$

$$P_e = \int_{-\infty}^{-\delta/2} \frac{1}{\sqrt{2\pi}\sigma} e^{-x^2/2\sigma^2} dx + \int_{\delta/2}^{\infty} \frac{1}{\sqrt{2\pi}\sigma} e^{-x^2/2\sigma^2} dx = 2 \int_{\delta/2\sigma}^{\infty} \frac{1}{\sqrt{2\pi}} e^{-x^2/2} dx = 2Q\left(\frac{\delta}{2\sigma}\right)$$

- Tabulando la funzione precedente si ha:



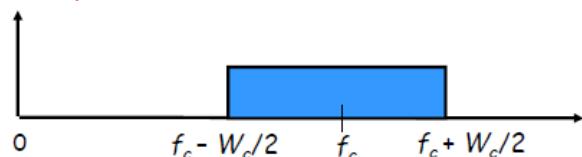
Capacità limite di Shannon

- Dato un canale con banda W e rumore Gaussiano e fissato un valore di S/N, il massimo rate di trasmissione raggiungibile per cui è ottenibile un BER arbitrariamente piccolo è dato da

$$C = W \log_2 (1 + S/N) \text{ bit/s}$$

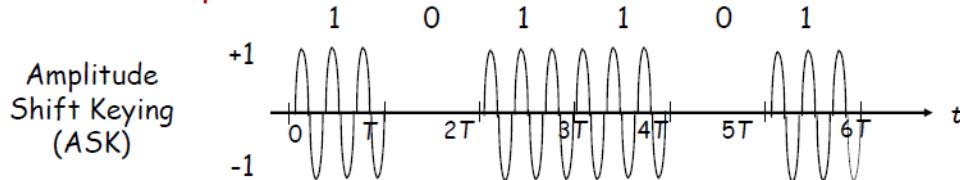
- Si ottiene un BER arbitrariamente piccolo mediante un'opportuna **Codifica di Linea**

Canali passa-banda



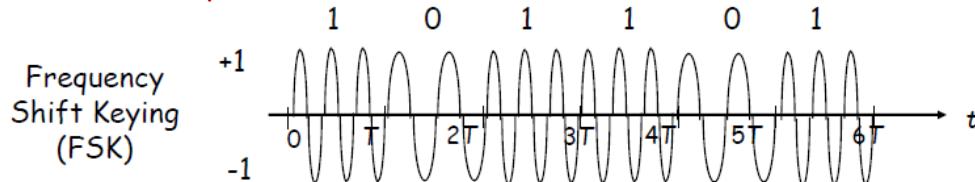
- I canali passa-banda sono passanti per un intervallo di frequenze centrate intorno ad una frequenza centrale f_c :
 - Canali radio channels, modem telefonici e xDSL
- I modulatori numerici (**Modem**) utilizzano forme d'onda che hanno frequenze che sono passanti per un canale passa-banda
- Un segnale sinusoidale di frequenza f_c è centrato nella banda del canale:
 - Un modulatore inserisce l'informazione in una sinusode [$\cos(2\pi f_c t)$]

Modulazione di Ampiezza



- Un modulatore ASK mappa ogni bit informativo nell'ampiezza di una sinusode a frequenza f_c :
 - "1" trasmissione del segnale sinusoidale
 - "0" nessun segnale
- Il demodulatore individua i periodi in cui è presente il segnale e i periodi in cui il segnale è assente

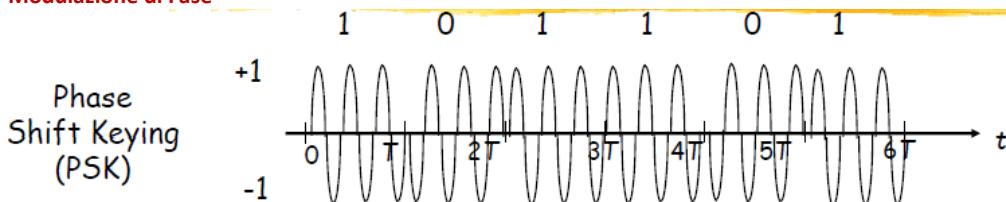
Modulazione di Frequenza



- Un modulatore FSK mappa ogni bit informativo nella frequenza di un segnale sinusoidale
 - "1" trasmissione di un segnale di frequenza $f_c + \delta$
 - "0" trasmissione di un segnale di $f_c - \delta$

Un demodulatore individua la potenza intorno alle frequenze $f_c + \delta$ o $f_c - \delta$

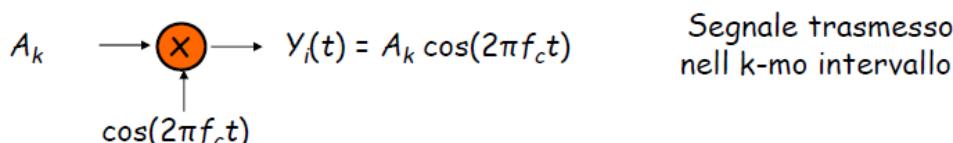
Modulazione di Fase



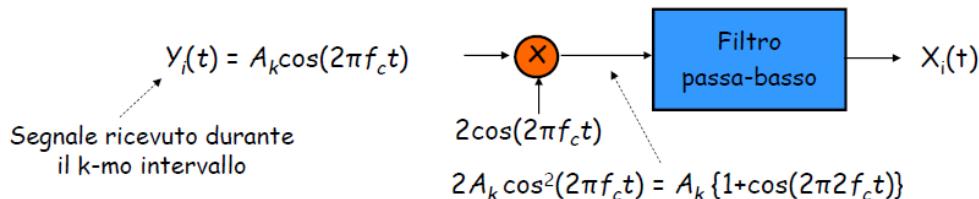
- Un modulatore PSK mappa ogni bit informativo nella fase di un segnale sinusoidale:
 - "1" trasmissione del segnale $A \cos(2\pi f t)$ \rightarrow fase 0
 - "0" trasmissione del segnale $A \cos(2\pi f t + \pi)$ \rightarrow fase π
- È equivalente a moltiplicare un segnale $\cos(2\pi f t)$ per $+A$ o $-A$
 - "1" trasmissione del segnale $A \cos(2\pi f t)$ \rightarrow moltiplicazione per A
 - "0" trasmissione del segnale $A \cos(2\pi f t + \pi) = -A \cos(2\pi f t)$ \rightarrow moltiplicazione per $-A$

Modulazione e Demodulazione PSK

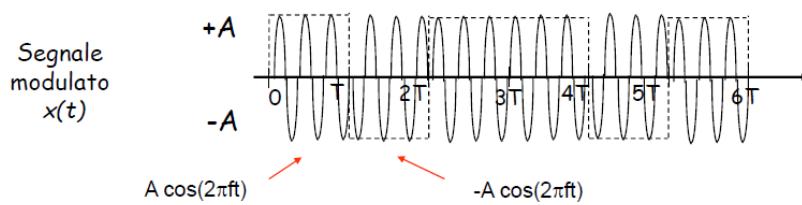
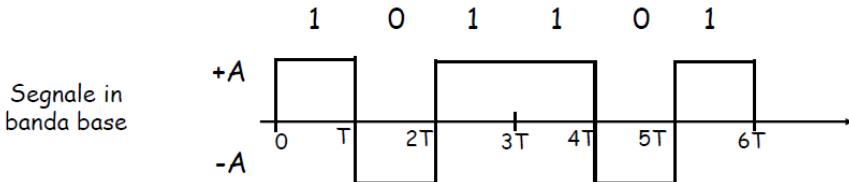
- Un segnale $\cos(2\pi f_c t)$ viene modulato moltiplicandolo per A_k per T secondi (durata di un simbolo)



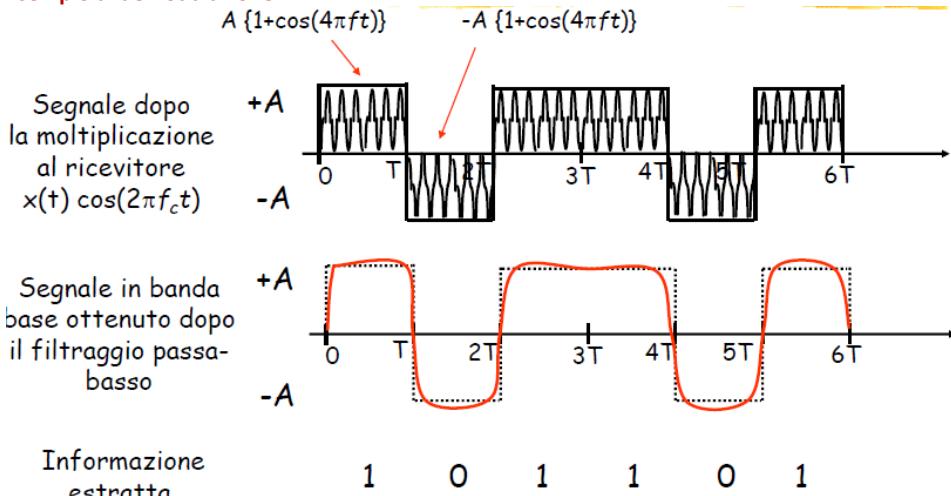
- Il segnale ricevuto viene demodulato moltiplicandolo per $2\cos(2\pi f_c t)$ per T secondi e successivamente filtrandolo con un filtro passa-basso



Esempio di modulazione

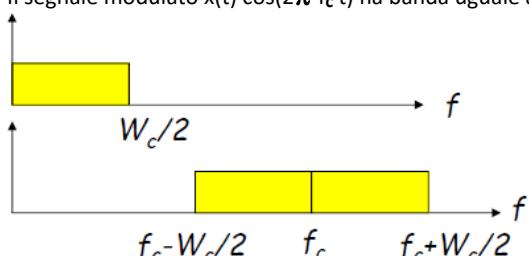


Esempio di demodulazione



Banda in trasmissione

- Se il segnale in banda base $x(t)$ ha banda $W_c / 2$ Hz
- Il segnale modulato $x(t) \cos(2\pi f_c t)$ ha banda uguale a W_c Hz

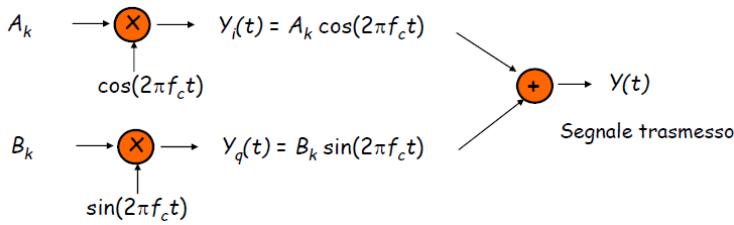


- Se il canale di comunicazione ha banda W_c Hz:
 - Il canale in banda base ha una larghezza di banda disponibile uguale a $W_c / 2$ Hz
 - Un sistema di modulazione supporta $(W_c / 2) \times 2 = W_c$ impulsi/secondo
 - Quindi W_c impulsi/secondo per W_c Hz = 1 impulso/Hz
 - Si ricorda che la trasmissione in banda base supporta 2 impulsi/Hz

Quadrature Amplitude Modulation (QAM)

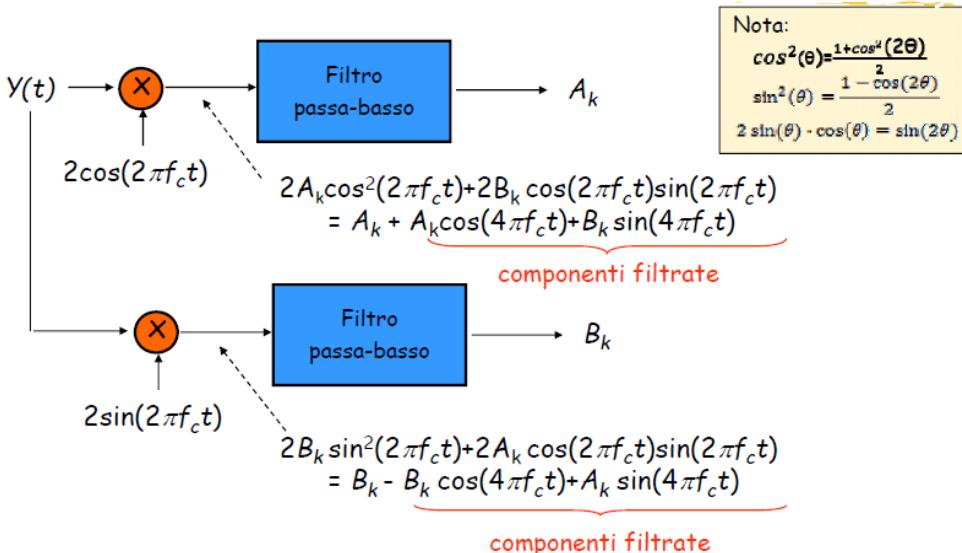
- QAM usa una trasmissione a due dimensioni:

- A_k modula il segnale in fase $\cos(2\pi f_c t)$ per T secondi
- B_k modula il segnale in quadratura $\cos(2\pi f_c t + \pi / 2) = \sin(2\pi f_c t)$ per T secondi
- Si trasmette la somma delle componenti in fase ed in quadratura



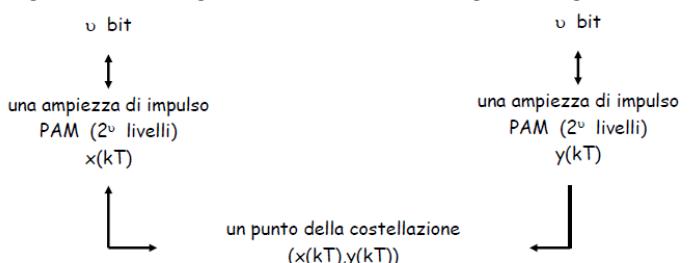
- I segnali $Y_i(t)$ e $Y_q(t)$ occupano entrambi la banda passante del canale
 - La modulazione QAM supporta 2 impulsi/Hz

Demodulazione QAM



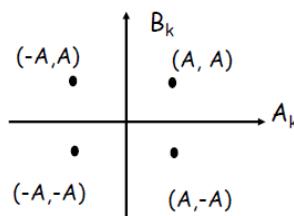
Costellazione dei segnali

- Ogni T secondi vengono trasmessi 2^v bit del segnale di ingresso

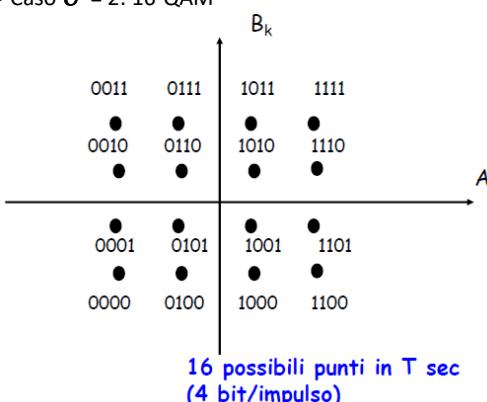


- Ogni coppia (A_k, B_k) definisce un punto nel piano
- La **costellazione** di un segnale è l'insieme dei punti che può assumere un segnale
- Caso $v = 1$: 4-QAM (4-PSK, QPSK)

**4 possibili punti in T sec
(2 bit/impulso)**



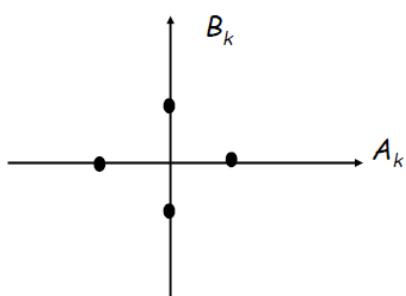
- Caso $v = 2$: 16-QAM



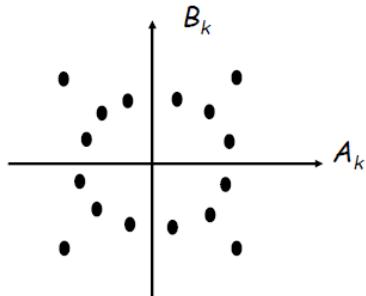
Altre costellazioni di segnale

Punti scelti in ampiezza e fase

$$A_k \cos(2\pi f_c t) + B_k \sin(2\pi f_c t) = \sqrt{A_k^2 + B_k^2} \cos(2\pi f_c t + \tan^{-1}(B_k/A_k))$$

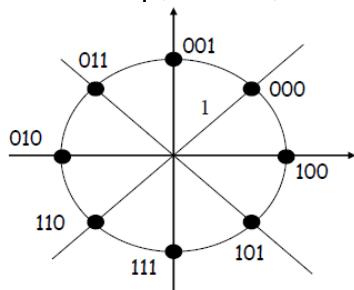


4 possibili punti in T sec

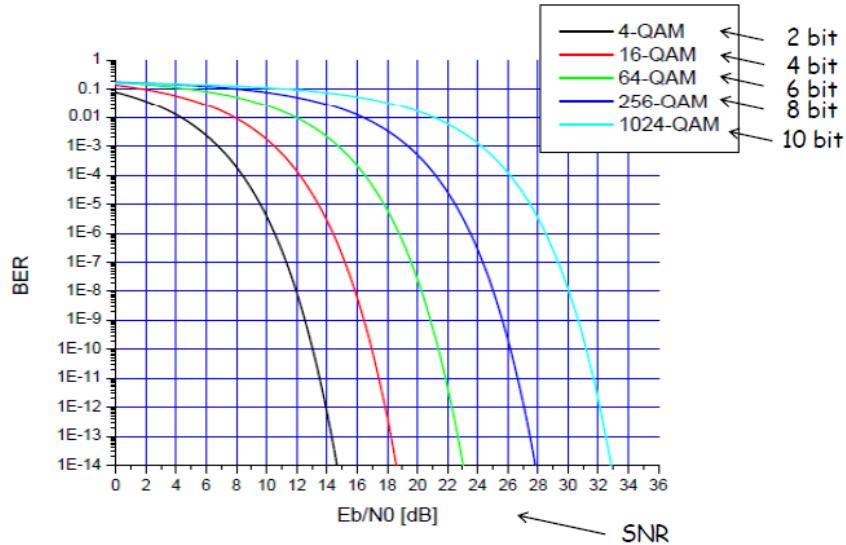


16 possibili punti in T sec

- Modulazione numerica con signal set a 8 punti disposti su una circonferenza di raggio 1, equidistanziati.
- Il nome 8-PSK (analogamente al 4-PSK) deriva dal fatto che le posizioni dei punti, in coordinate polari (r , ϕ) sono differenziate soltanto in base alla fase ϕ ($r = 1 = \text{cost}$).

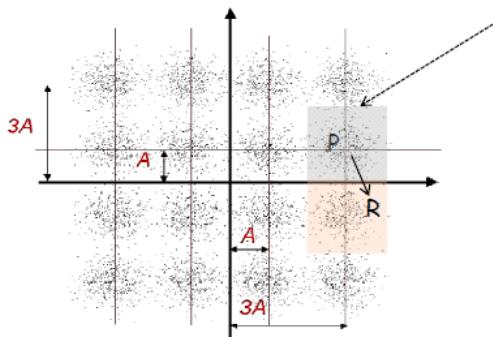


Prestazioni QAM



Effetto del rumore

- Si individua nel piano del signal set delle regioni di decisione associate ai punti della costellazione



La generica regione di decisione associata a un punto P è costituita da tutti i punti del piano più vicini a P che a tutti gli altri punti del signal set. Si ha una decisione errata (corrispondente a uno o più bit errati nel segnale binario demodulato) quando rumore è tale da far cadere il punto ricevuto R al di fuori della regione di decisione relativa al punto trasmesso P.

"Framing" e "Error control"

Strato di collegamento (Data Link)

- Gli host e i router sono i **nodi**
- I canali di comunicazione che collegano nodi adiacenti lungo un cammino sono i **collegamenti (link)**
 - Collegamenti cablati
 - Collegamenti wireless
 - LAN
- Le unità di dati scambiate dai protocolli a livello di link sono chiamate **frame**.
- I **protocolli di strato di collegamento** si occupano del trasporto dei pacchetti lungo un singolo canale di comunicazione (link)
- Un pacchetto può essere gestito da diversi protocolli su collegamenti differenti:
 - Es., un pacchetto può essere gestito da Ethernet sul primo collegamento, da PPP sull'ultimo e da un protocollo WAN nel collegamento intermedio
- I servizi erogati dai protocolli del livello di link possono essere differenti:
 - Ad esempio, non tutti i protocolli forniscono un servizio di consegna affidabile (controllo d'errore)

Servizi offerti dallo strato di link

• Framing:

- I protocolli incapsulano i pacchetti del livello di rete all'interno di un **frame** a livello di link
- Se necessario (reti ad accesso multiplo) il **protocollo MAC** controlla l'accesso al mezzo:
 - Per identificare origine e destinatario vengono utilizzati indirizzi "MAC"

• Rivelazione e correzione degli errori:

- Gli errori sono causati dal transito del segnale nel mezzo trasmissivo
- Il nodo ricevente individua la presenza di errori
 - è possibile grazie all'inserimento, da parte del nodo trasmittente, di bit di controllo di errore all'interno del frame
- Il nodo ricevente oltre a rivelare l'errore lo corregge

Servizi offerti dal livello di collegamento

• Controllo di flusso:

- Evita che il nodo trasmittente saturi quello ricevente

• Consegnare affidabile dei dati e ritrasmissione:

- Nel caso i requisiti dell'applicazione impongano una consegna affidabile dei dati il protocollo di link può effettuare la ritrasmissione delle frame affette da errore:
 - Questa funzione può essere eseguita anche nello strato di trasporto (es. TCP)
- È normalmente utilizzata nei collegamenti soggetti a elevati tassi di errori (es.: collegamenti wireless)

• Half-duplex e full-duplex:

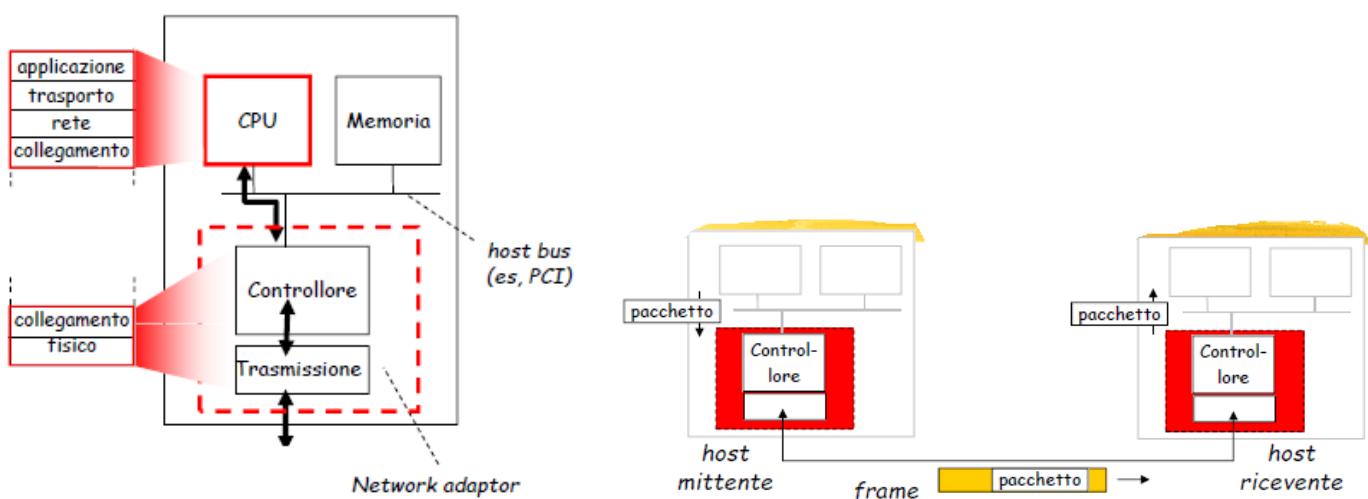
- Nella modalità full-duplex gli estremi di un collegamento possono trasmettere contemporaneamente
- Nella modalità half-duplex la trasmissione nei due versi è alternata

Esempio di implementazione

• In tutti gli host è realizzato in una **Network Interface Card (NIC)**

- Es. scheda Ethernet, PCMCIA, 802.11
- Implementa il livello di collegamento e fisico

• È una combinazione di hardware, software e firmware



• Lato mittente:

- Incapsula un pacchetto in una frame
- Imposta i bit di rilevazione degli errori, trasferimento dati affidabile, controllo di flusso, etc.

• Lato ricevente:

- Individua gli errori, trasferimento dati affidabile, controllo di flusso, etc.
- Estraie i pacchetti e li passa al nodo ricevente

Framing

- Ha lo scopo di formare la PDU di strato (frame) incapsulando la PDU di strato superiore (pacchetto)
- L'entità ricevente deve essere in grado di riconoscere senza ambiguità l'inizio e la fine di ogni frame (funzione di delimitazione)
- Ad ogni frame viene aggiunto all'inizio e alla fine una sequenza fissa di bit, denominata flag:
 - L'entità ricevente esamina il flusso binario entrante e delimita le frame riconoscendo i flag di apertura e di chiusura
- Problema della simulazione del flag all'interno della frame

Esempio di funzione di delimitazione

- Una possibile configurazione del Flag di delimitazione è:

0111110

- Per evitare la simulazione si utilizzano le funzioni di:

- **Bit stuffing:**

- In emissione, si aggiunge uno "0" dopo ogni sequenza di cinque "1" consecutivi all'interno della frame indipendentemente da quale sia il bit successivo

- **Bit destuffing:**

- In ricezione si contano gli "1" consecutivi
 - Quando sono ricevuti cinque "1" consecutivi, si esamina la cifra successiva:
 - Se è un "1": la sequenza di cifre binarie è un Flag
 - Se è un "0": questo è un bit di stuffing e deve quindi essere eliminato

Esempio bit stuffing

- Sequenza originale:

1 0 1 1 1 1 1 1 1 1 1 0 1 1 0 1 1 1 1 0 0 1 1 1 1 1 0 0

- Sequenza trasmessa:

1 0 1 1 1 1 1 0 1 1 1 1 0 1 0 1 1 0 1 1 1 1 0 0 0 1 1 1 1 1 0 1 0 0

- Sequenza ricevuta:

1 0 1 1 1 1 1 0 1 1 1 1 0 1 0 1 1 0 1 1 1 1 0 0 0 1 1 1 1 1 0 1 0 0

Byte stuffing e de-stuffing

- Utilizzata nel protocollo PPP (Point to Point Protocol)

- **Byte stuffing** (si usa una sequenza di «control escape» 01111101):

- In emissione, se in una parte della frame compare la sequenza "0111110" (ad eccezione del flag) o la sequenza "01111101" viene premesso il byte "01111101"

- **Byte destuffing:**

- In ricezione:
 - Se si ricevono due byte consecutivi "01111101" uno dei due viene eliminato
 - Se si riceve un "01111101" seguito da un "0111110" il primo viene eliminato
 - Se si riceve un solo "0111110" viene riconosciuto come flag

- **Esempio:**

- Sequenza originale

1 0 1 1 1 1 1 0 1 1 0 1 1 1 1 1 0 1 0 1 1 1 1 1 0 0

- Sequenza trasmessa

1 0 1 1 1 1 1 0 1 1 0 1 1 0 1 1 1 1 1 0 1 0 1 1 1 1 1 0 0

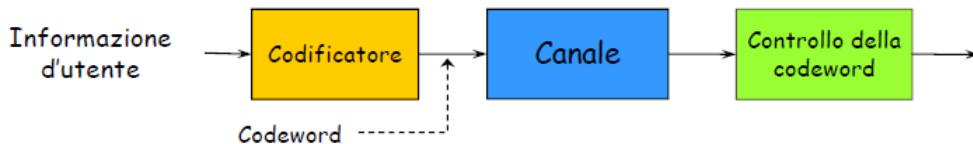
Rivelazione e correzione d'errore

Controllo d'errore

- La trasmissione introduce errori:
 - **Bit Error Rate (BER)**
- Il **controllo d'errore** si usa quando il livello trasmissivo non soddisfa i requisiti dell'applicazione:
- Il controllo d'errore assicura un determinato livello di accuratezza nel trasferimento di uno stream dati
- Due approcci possibili:
 - **Error detection & retransmission (ARQ)**
 - **Forward Error Correction (FEC)**

Principio base del controllo d'errore

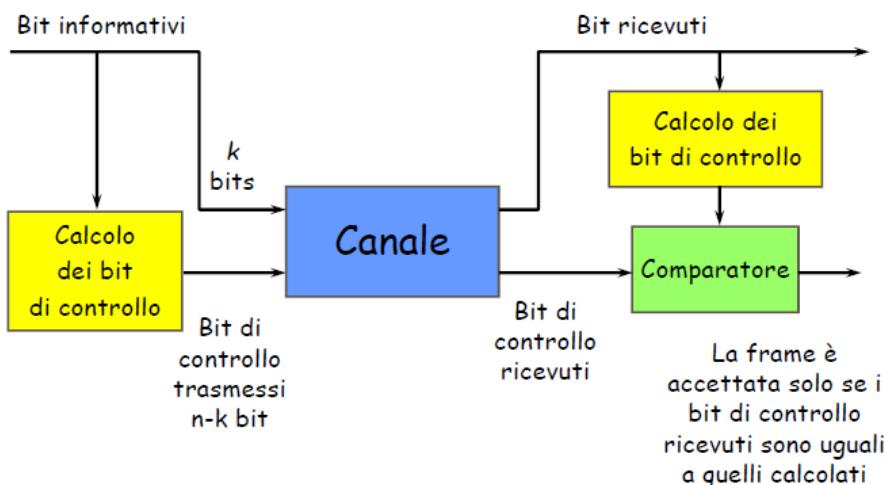
- Si organizza la trasmissione in modo da trasformare i blocchi di dati trasmessi in particolari "parole di codice" (**codeword**)
- Se il blocco ricevuto non è una parola di codice è considerato in errore
- È necessaria una **ridondanza (overhead)** costituita da un insieme di bit di controllo da aggiungere al blocco dati d'utente in modo che il blocco trasmesso sia una "codeword"
- È possibile che il canale trasformi la parola di codice trasmessa in una stringa di bit che è ugualmente una parola di codice



Rivelazione di errore

- Se:
 - k è la lunghezza del blocco da proteggere;
 - $n-k$ è il numero di bit di controllo
- Le **codeword** sono di lunghezza uguale a **n bit**
- Se una PDU è colpita da errore e se questi sono in configurazione tale da non essere rivelati (sostituzione di codeword), si verifica l'evento di "errori non rivelati"
- I metodi di codifica per rivelare errori rientrano usualmente nella categoria dei **codici con controllo di parità (parity check codes)**:
 - Codici a **parità singola**
 - Codici a **parità a blocchi**
 - Codici a **ridondanza ciclica (CRC, Cyclic Redundancy Check)**

Funzione di Error Detection



Controllo di parità singola

- Aggiunge un bit di parità a k bit informativi:

Info Bit	$b_1, b_2, b_3, \dots, b_k$
Check Bit	$b_{k+1} = (b_1 + b_2 + b_3 + \dots + b_k) \text{ modulo } 2$
Codeword	$(b_1, b_2, b_3, \dots, b_k, b_{k+1})$
- Un blocco dati trasmesso ha un numero pari di "1"
- Il ricevitore controlla se il numero di "1" è pari:
 - È rivelabile una qualsiasi configurazione di errore che modifica un numero dispari di bit
 - Tutte le configurazioni di errore che modificano un numero pari di bit non sono rilevabili
- **Esempio:**
 - Bit informativi (7 bit): $(0, 1, 0, 1, 1, 0, 0)$
 - Bit di parità: $b_8 = 0 + 1 + 0 + 1 + 1 + 0 + 0 = 1$
 - Codeword(8 bit): $(0, 1, 0, 1, 1, 0, 0, 1)$
 - Errore singolo nel bit 3 : $(0, 1, 1, 1, 1, 0, 0, 1)$
 - Numero di "1" è uguale a 5 (dispari)
 - Errore rivelato
 - Errore nei bit 3 and 5: $(0, 1, 1, 0, 0, 0, 1)$
 - Numero di "1" = 4 (pari)
 - Errore non rivelato

Prestazioni del controllo di parità

- **Ridondanza:**
 - Il controllo di parità aggiunge 1 bit di ridondanza ogni k bit informativi
 - Overhead = $1 / (k+1)$

- **Errori rivelati:**

- Una configurazione di errore è una stringa binaria composta da $(n=k+1)$ bit [($k+1$)-tuple], in cui sono presenti bit "1" nelle posizioni in cui si sono verificati gli errori, mentre gli altri bit sono uguali a "0"
- Tutte le configurazioni di errore con un numero dispari di bit modificati sono rivelati
- Tra tutte le 2^{k+1} ($k+1$)-tuple binarie, $\frac{1}{2}$ hanno un numero dispari di "1"
- Solo il 50% delle configurazioni di errore possono essere rivelate

- **Normalmente si assume l'ipotesi che i canali introducono errori sui bit in modo indipendente con probabilità p:**

- Una statistica più attendibile prevede **errori a burst**

- Alcune configurazioni di errore sono più probabili di altre:

$$P[10000000] = BER \cdot (1 - BER)^7 = (1 - BER)^8 \frac{BER}{1 - BER}$$

$$P[11000000] = BER^2 \cdot (1 - BER)^6 = (1 - BER)^8 \left(\frac{BER}{1 - BER} \right)^2$$

- Poiché si può assumere $BER \ll 0.5$ si ha $BER / (1 - BER) \ll 1$, quindi le configurazioni con 1 solo errore sono più probabili delle configurazioni con 2 errori e così via

- Qual è la probabilità di non rivelare gli errori?

Gli errori non rivelabili

- Configurazione d'errore con un numero pari di "1"

$$= \binom{n}{2} BER^2 (1 - BER)^{n-2} + \binom{n}{4} BER^4 (1 - BER)^{n-4} + \dots$$

- $\Pr\{\text{errore non rivelabile}\} = \Pr\{\text{config. di errore con \# pari di 1}\}$

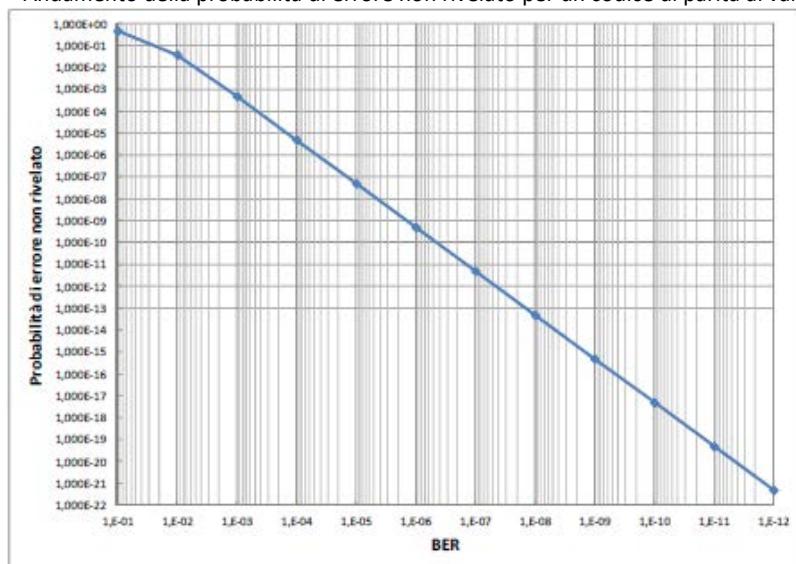
- Esempio: $n = 32$, $BER = 10^{-3}$

- $\Pr\{\text{errore non rivelabile}\} =$

$$\begin{aligned} & \binom{32}{2} (10^{-3})^2 (1 - 10^{-3})^{30} + \binom{32}{4} (10^{-3})^4 (1 - 10^{-3})^{28} + \dots \\ & = 496(10^{-6}) + 35960(10^{-12}) \approx 4.96 \cdot 10^{-4} \end{aligned}$$

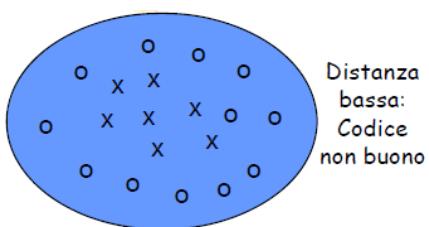
- Disegnare l'andamento della probabilità di errore non rilevabile al variare del BER

- Andamento della probabilità di errore non rivelato per un codice di parità al variare del BER per un blocco dati di lunghezza $n=32$ bit

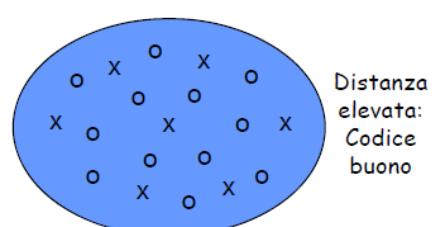


Quanto è "buono" un codice?

- In molti canali le configurazioni di errore più probabili sono quelle con un numero basso di bit errati
- Questi errori trasformano le codeword trasmesse in n -tuple "vicine"
- Se le codeword sono "vicine" tra loro allora la funzione di rivelazione può fallire
- I buoni codici massimizzano la "distanza" tra le codeword trasmesse



x = codewords
 o = noncodewords



Controllo di parità bi-dimensionale

- Un numero maggiore di bit di parità aumentano le prestazioni del codice:

- Si struttura la sequenza di bit informativi in colonne
- Si aggiunge un bit di parità per ogni colonna
- Si aggiunge una “colonna di parità”

1	0	0	1	0	0
0	1	0	0	0	1
1	0	0	1	0	0
1	1	0	1	1	0
<hr/>					1
1	0	0	1	1	1

La colonna finale è formata dai bit di parità di ogni riga

La riga finale è formata dai bit di controllo di ogni colonna

Capacità di rivelazione d'errore

1	0	0	1	0	0
0	0	0	0	0	1
1	0	0	1	0	0
1	1	0	1	1	0
<hr/>					1
1	0	0	1	1	1

↑

1	0	0	1	0	0
0	0	0	0	0	1
1	0	0	1	0	0
1	0	0	1	1	0
<hr/>					1
1	0	0	1	1	1

↑

1	0	0	1	0	0
0	0	0	1	0	1
1	0	0	1	0	0
1	0	0	1	1	0
<hr/>					1
1	0	0	1	1	1

↑

1	0	0	1	0	0
0	0	0	0	0	1
1	0	0	1	0	0
1	0	0	1	1	0
<hr/>					1
1	0	0	1	1	1

↑

1	0	0	1	0	0
0	0	0	1	0	1
1	0	0	1	0	0
1	0	0	1	1	0
<hr/>					1
1	0	0	1	1	1

↑

- Configurazioni con 1, 2, o 3 errori possono essere sempre rivelate.
- Non tutte le configurazioni di > 4 errori possono essere rivelate

Altri codici di rivelazione d'errore

- I codici a parità singola hanno scarse prestazioni:
 - Elevata probabilità di non rivelare errori
- I codici bi-dimensionalni hanno overhead elevato:
 - Richiedono un numero elevato di bit di controllo
- I codici più usati sono:
 - **Internet Checksums**: Strato di trasporto (implementazione software)
 - **Codici polinomiali a ridondanza ciclica (CRC)**: Strato di collegamento (implementazione hardware)

Internet Checksum

- Molti protocolli usati in Internet (es. IP, TCP, UDP) usano bit di controllo (**checksum**) per rivelare errori nell'header IP (o nell'header e nel campo dati delle unità dati TCP/UDP)
- Il checksum è inserito in uno specifico campo dell'header delle PDU (RFC 1071)
- Il checksum è ricalcolato in ogni router e quindi deve essere di facile implementazione in software
- Si considera che la stringa di bit da proteggere sia composta da L parole di 16 bit:

$$b_0, b_1, b_2, \dots, b_{L-1}$$

- Il checksum è una stringa b_L di 16 bit

Calcolo del Checksum

- Il checksum b_L è calcolato come segue
- Ciascuna stringa di 16-bit è considerata un intero
- $$x = b_0 + b_1 + b_2 + \dots + b_{L-1} \text{ modulo}(2^{16}-1)$$
- Il checksum è dato da:
- $$b_L = -x \text{ modulo}(2^{16}-1)$$
- Quindi, l'intero blocco trasmesso deve soddisfare la seguente proprietà:
- $$0 = b_0 + b_1 + b_2 + \dots + b_{L-1} + b_L \text{ modulo}(2^{16}-1)$$
- Il calcolo del checksum è eseguito in software

Codici CRC: i polinomi

- Le singole cifre binarie di una stringa da proteggere sono trattate come coefficienti (di valore "0" o "1") di un polinomio $P(x)$
- Le cifre binarie della stringa con lunghezza uguale a K sono considerate come i coefficienti di un polinomio completo di grado $K-1$

$$P(x) = a_{k-1}x^{k-1} + a_{k-2}x^{k-2} + \dots + a_1x^1 + a_0$$

- In particolare, l' i -esimo bit (a_i) della stringa è il coefficiente del termine x_{i-1} di $P(x)$
- Le entità emittente e ricevente utilizzano un polinomio comune $G(x)$, detto **polinomio generatore**
- Il polinomio $G(x)$ gode di **opportune proprietà** nell'ambito della **teoria dei campi algebrici**
- I coefficienti di $G(x)$ sono **binari**, come quelli di $P(x)$, supponiamo che questo polinomio sia di **grado z**
 - I coefficienti di $G(x)$ di grado massimo e di grado nullo debbono entrambi essere uguali a 1
 - Es. $x^{16} + x^{12} + x^5 + 1$ ($z = 16$)
- La entità emittente utilizza $G(x)$ come divisore del polinomio $x^z P(x)$

$$\frac{x^z P(x)}{G(x)} = Q(x) + \frac{R(x)}{G(x)}$$

- $Q(x)$ è il **polinomio quoziente**

- La particolarità della divisione risiede nel fatto che:
 - I coefficienti di dividendo e di divisore sono binari
 - L'aritmetica viene svolta modulo 2

Aritmetica polinomiale a coefficienti binari

• Addizione:

- Addizione e sottrazione sono operazioni identiche
- Equivalenti ad un XOR sui bit degli operandi

$$(x^7 + x^6 + 1) + (x^6 + x^5) = x^7 + \cancel{x^6 + x^6} + x^5 + 1 = x^7 + x^5 + 1$$

• Moltiplicazione:

- La moltiplicazione di una stringa binaria per 2^k equivale ad uno shift verso sinistra di k posizioni

$$(x + 1) \cdot (x^2 + x + 1) = x^3 + x^2 + x^2 + x + x + 1 = x^3 + 1$$

• Divisione:

Algoritmo di Euclide	Divisore	Quoziente [Q(x)]
	$x^3 + x + 1$	$x^6 + x^5$
		$x^6 + x^4 + x^3$
		$+ x^5 + x^4 + x^3$
		$+ x^5 + x^3 + x^2$
		$+ x^4 + x^2$
		$+ x^4 + x^2 + x$
		Resto [R(x)]

Osservazione

- Dato il grado del polinomio generatore, il grado del polinomio resto $R(x)$ è al più uguale a $Z-1$;
- Conseguentemente $R(x)$ può essere sempre rappresentato con Z coefficienti (binari), ponendo uguali a "0" i coefficienti dei termini mancanti

Codici CRC: l'emettitore

- Ottenuto il resto $R(x)$, l'entità emittente inserisce i coefficienti di questo polinomio in un apposito campo della PDU (**campo CRC**), che deve quindi avere lunghezza Z bit
- Nella PDU emessa trovano quindi posto le cifre binarie da proteggere (in numero uguale a K) e le cifre CRC (in numero uguale a Z): in totale $K+Z$ cifre binarie, che sono rappresentative di un polinomio $T(x)$ di grado $K+Z-1$

$$T(x) = x^Z P(x) + R(x)$$

e che costituiscono una parola di codice

- Tenendo conto che, per definizione,

$$x^Z P(x) = Q(x)G(x) + R(x)$$

e poiché addizione e sottrazione modulo 2 si equivalgono, si ottiene

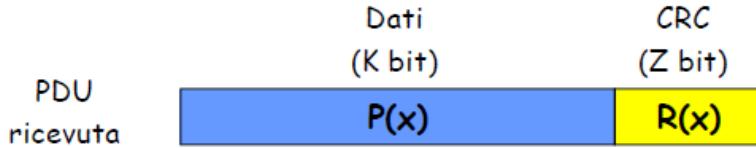
$$x^Z P(x) - R(x) = x^Z P(x) + R(x) = T(x) = Q(x)G(x)$$

cioè la stringa emessa (rappresentativa del polinomio $T(x)$) è divisibile per il polinomio generatore $G(x)$

- Si conclude che:
 - Tutte le parole di codice sono divisibili per il polinomio generatore
 - Tutti i polinomi divisibili per $G(x)$ sono parole di codice

Codici CRC: il ricevitore

- L'entità ricevente esegue, con il polinomio generatore, l'operazione di divisione effettuata in emissione
- In questo caso opera però sul polinomio rappresentato dalle $K+Z$ cifre binarie ricevute



- Supponiamo che nel trasferimento si siano verificati errori, con una sequenza rappresentata dal polinomio $E(x)$
 - Ogni errore nella PDU corrisponde ad un coefficiente non nullo in $E(x)$
- Allora le cifre binarie ricevute rappresentano il polinomio

$$T(x) + E(x)$$

Ove l'addizione è svolta in modulo 2 (XOR)

A	B	XOR
0	0	0
0	1	1
1	0	1
1	1	0

- Ogni bit "1" in $E(x)$ corrisponde ad un bit che è stato invertito e quindi a un **errore isolato**
- Un **errore a burst** di lunghezza n è caratterizzato in $E(x)$ da un "1" iniziale, una mescolanza di "0" e "1", e un "1" finale per un complesso di n coefficienti binari

$$E(x) = x^i (x^{n-1} + \dots + 1)$$

Ove i determina quanto il burst è lontano dall'estremità destra della PDU

- Il ricevitore calcola il resto della divisione di $T(x)+E(x)$ per $G(x)$
 - Le modalità sono le stesse utilizzate nell'emittitore
- poiché $T(x)$ è divisibile per $G(x)$, ne segue che:

$$\text{Resto} \left[\frac{T(x) + E(x)}{G(x)} \right] = \text{Resto} \left[\frac{E(x)}{G(x)} \right]$$

- Conseguentemente la regola applicata dal ricevitore è la seguente:
 - Se il resto della divisione $[T(x)+E(x)] / G(x)$ è nullo, la PDU ricevuta è assunta "senza errori"
 - In caso contrario, si sono verificati uno o più errori nel corso del trasferimento.
- Si nota che sono **non rivelabili** le configurazioni di errore per le quali il relativo polinomio $E(x)$ contiene $G(x)$ come fattore

Codici CRC: protezione contro gli errori

- Un codice polinomiale, in cui il polinomio generatore contiene $x+1$ come fattore primo, è in grado di rivelare:
 - Tutti gli **errori singoli o doppi**;
 - Tutti gli errori isolati con una **molteplicità dispari**
 - Tutti gli errori a burst di lunghezza $\leq Z$
- Se la lunghezza del burst è $z+1$ e se tutte le combinazioni della raffica sono considerate equiprobabili, la probabilità che l'errore a raffica non sia rivelato è uguale a 2^{-z}
- Infine, se il burst ha lunghezza maggiore di $z+1$, nell'ipotesi di equiprobabilità delle configurazioni di errore, la probabilità di errore non rivelato è uguale a 2^{-z}

Codici CRC: polinomi generatori

- Sono standard i seguenti polinomi generatori:

$$G(x) = x^{16} + x^{12} + x^5 + 1$$

$$G(x) = x^{32} + x^{26} + x^{23} + x^{22} + x^{16} + x^{12} + x^{11} + x^{10} + x^8 + x^7 + x^5 + x^4 + x^2 + x + 1$$
- Entrambi sono divisibili per $x+1$ e quindi danno luogo a codici CRC con le proprietà suddette

Forward Error Correction (FEC)

- Date due stringhe binarie di ugual lunghezza, X e Y e posto $W(A) =$ numero di bit 1 della stringa A , si definisce **distanza di Hamming tra X e Y** la quantità:

$$HD(X, Y) = W(X \text{ xor } Y)$$

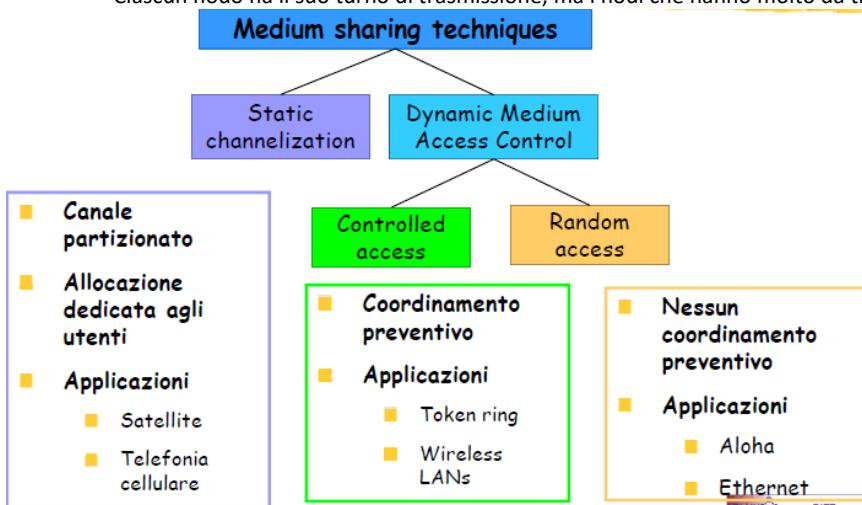
- Un codice con parole di n bit può rappresentare simboli di m bit e la capacità di correzione è funzione della ridondanza $r=n-m$; il valore minimo della HD tra tutte le coppie di parole di codice è la HD del codice
- Un codice con $HD=2d+1$ può correggere fino a d errori binari e può rivelarne fino a $2d$
- Un esempio di codice con $n=10$, $m=2$, $r=8$, $d=2$ è il seguente:

0000000000 0000011111 1111100000 1111111111

"Protocolli MAC"

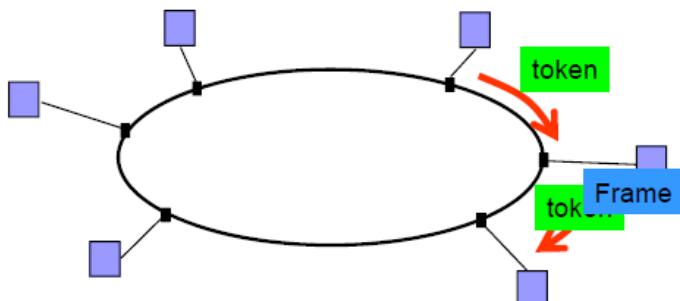
Protocolli di accesso multiplo

- Esistono due tipi di collegamenti di rete:
- **Collegamento punto-punto (PPP):**
 - Impiegato in connessioni telefoniche
 - Collegamenti punto-punto tra Ethernet e host
- **Collegamento broadcast (cavo o canale condiviso):**
 - Ethernet
 - Wireless LAN 802.11
- Centinaia o anche migliaia di nodi possono comunicare direttamente su un canale broadcast:
 - Si genera una collisione quando i nodi ricevono due o più frame contemporaneamente
- **Protocolli di accesso multiplo:**
 - Protocolli che fissano le modalità con cui i nodi regolano le loro trasmissioni sul canale condiviso
 - La comunicazione relativa al canale condiviso deve utilizzare lo stesso canale:
 - Non c'è un canale "out-of-band" per il coordinamento
- **Protocolli a suddivisione del canale (canalizzazione statica):**
 - Suddivide il canale in "parti più piccole" (slot di tempo, frequenza, codice)
 - Le parti vengono allocate ad un nodo per utilizzo esclusivo
- **Protocolli ad accesso dinamico:**
- **Protocolli ad accesso casuale (random access):**
 - I canali non vengono divisi e si può verificare una collisione
 - I nodi coinvolti ritrasmettono ripetutamente i pacchetti
- **Protocolli ad accesso controllato (controlled access):**
 - Ciascun nodo ha il suo turno di trasmissione, ma i nodi che hanno molto da trasmettere possono avere turni più lunghi.



Protocollo ad accesso controllato Token-Passing

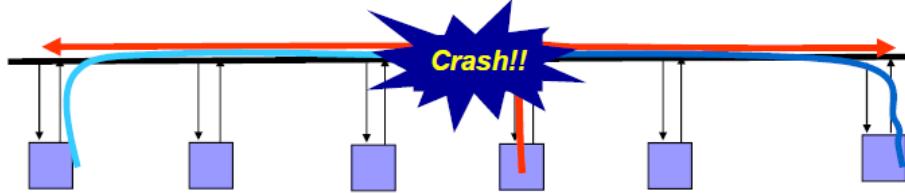
- Rete ad anello:



- La stazione che detiene il token può trasmettere
- Non sono possibili collisioni

Random Access

- Rete a bus:



- Una stazione trasmette quando è pronta
- Possibili collisioni, strategie di ritrasmissione

Protocolli ad accesso casuale

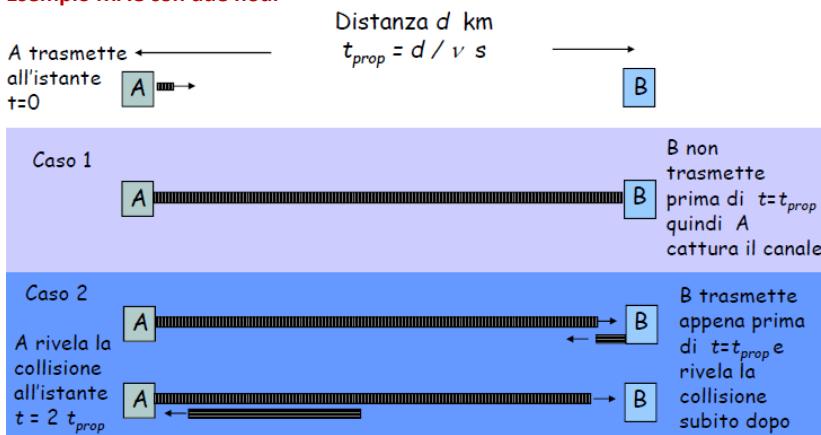
- Quando un nodo deve inviare un pacchetto:
 - Trasmette sempre alla massima velocità del canale, cioè **R bit/s**
 - Nessun coordinamento a priori tra i nodi
- Se due o più nodi trasmettono "contemporaneamente" si ha una "**collisione**"
- Un protocollo ad accesso casuale definisce:
 - Come rilevare un'eventuale collisione
 - Le politiche di ritrasmissione in caso di collisione
- Esempi di protocolli ad accesso casuale:
 - **ALOHA**
 - **slotted ALOHA**
 - **CSMA, CSMA/CD, CSMA/CA**

Prodotto Banda-Ritardo

$$PBR = Rd \text{ (bit)}$$

- **R** (bit/s): banda del canale
- **d** (sec): ritardo di propagazione end-to-end
- È il numero di bit che si trovano contemporaneamente sul canale:
 - Lunghezza elettrica del canale
- Parametro chiave dei protocolli MAC:
 - Il coordinamento tra i nodi richiede l'uso della banda del canale (in modo esplicito o implicito)
 - La difficoltà del coordinamento è legata al prodotto banda-ritardo

Esempio MAC con due nodi



Calcolo dell'efficienza

- Nel caso in esame, la trasmissione di una frame ha un intervallo di vulnerabilità uguale a $2t_{prop}$:
 - Il nodo B non deve iniziare la trasmissione un tempo t_{prop} prima e dopo rispetto all'inizio della trasmissione di A:
 - R bit rate del canale (bit/s)
 - L lunghezza di una frame (bit)

$$\text{Efficienza} = \rho_{\max} = \frac{L}{L + 2t_{prop}R} = \frac{1}{1 + 2t_{prop}R/L} = \frac{1}{1 + 2a}$$

$$\text{Throughput Massimo} = R_{\text{eff}} = \frac{L}{L/R + 2t_{prop}} = \frac{1}{1 + 2a} R \text{ bit/s}$$

$$a = \frac{t_{prop}}{L/R}$$

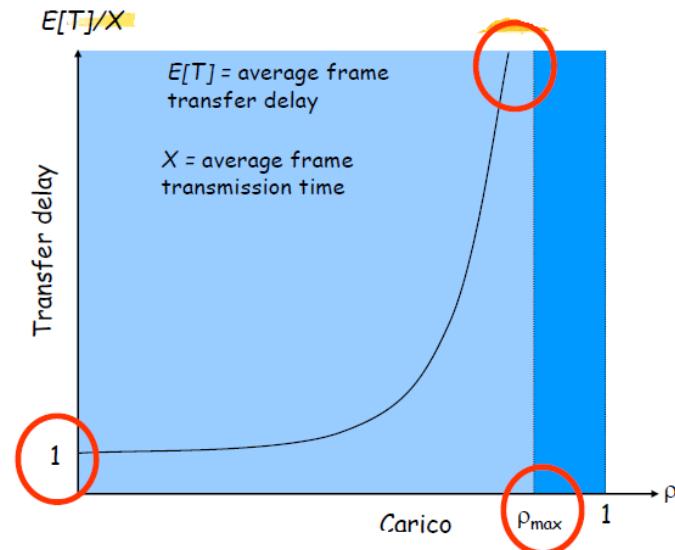
↓ Ritardo di Propagazione
 ↓ Tempo di trasmissione
 di una frame

Valori tipici del prodotto banda-ritardo

Distanza	Bit Rate			Tipo di rete
	10 Mbit/s	100 Mbit/s	1 Gbit/s	
1 m	5×10^{-2}	5×10^{-1}	5×10	Desk area network (DAN)
100 m	5×10^1	5×10^2	5×10^3	Local area network (LAN)
10 km	5×10^2	5×10^3	5×10^4	Metropolitan area network (MAN)
1000 km	5×10^4	5×10^5	5×10^6	Wide area network (WAN)
100000 km	5×10^6	5×10^7	5×10^8	Global area network

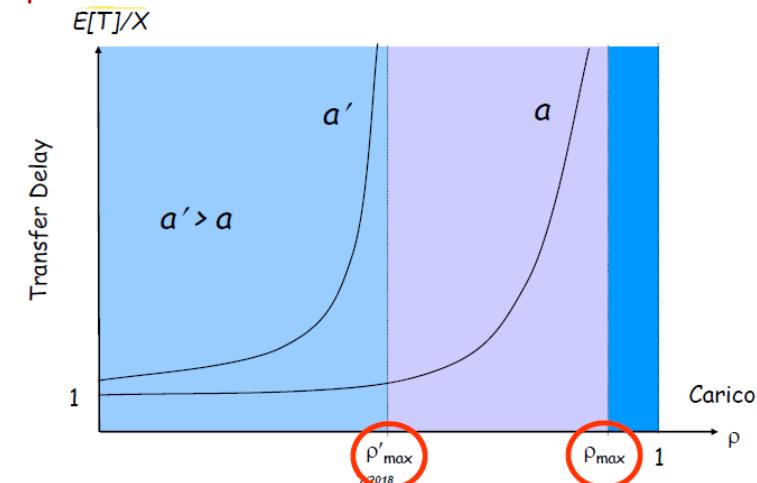
- Max size Ethernet frame = 1500 byte = 12000 bit = $1.2 \cdot 10^4$ bit
- Se aumenta il prodotto banda x ritardo l'efficienza di un protocollo MAC diminuisce

Prestazioni di ritardo



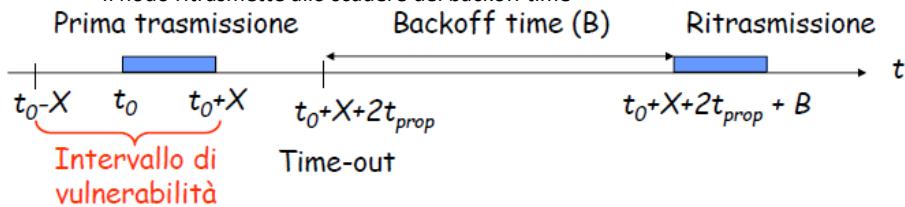
- A basso carico il ritardo è uguale al tempo di trasmissione
- Ad alto carico il ritardo cresce a causa delle attese per l'accesso al canale
- Carico massimo minore di 1

Dipendenza da a



Protocollo ALOHA

- Protocollo sviluppato per l'interconnessione tra dipartimenti dell'Università delle Hawaii:
 - Un nodo trasmette appena ha una frame pronta
 - Se viene trasmessa più di una frame si ha una collisione (frame persa)
 - Se un nodo non riceve un ACK entro un certo tempo (timeout), il nodo calcola il tempo di ritrasmissione (backoff time)
 - Il nodo ritrasmette allo scadere del backoff time



Modello prestazionale Aloha

- Definizioni:
 - X : frame transmission time (costante)
 - S : throughput (numero medio di trame trasmesse con successo in un intervallo di X secondi) ($0 < S < 1$)
 - G : load (numero medio di tentativi di trasmissione in un intervallo di X secondi)
 - P_{succ} : probabilità che una trama sia trasmessa con successo
- Si ha:

$$S = G P_{\text{succ}}$$

- L'intervallo di vulnerabilità nella trasmissione di una frame è uguale a $2X$

- Si consideri che il carico G comprenda anche le trasmissioni:

- Dividiamo X in n intervalli di durata $\Delta = X/n$
- Se p è la probabilità di una trasmissione in un intervallo Δ si ha:

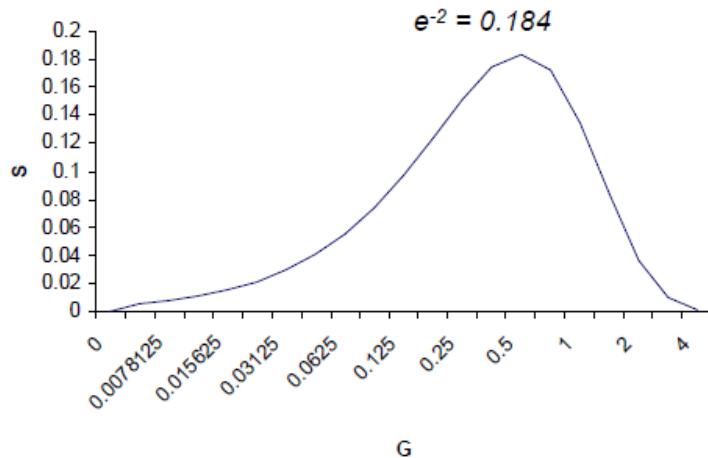
$$G = n p$$

$$r_{\text{succ}} = P[0 \text{ arrivi in } 2X] = P[0 \text{ arrivi in } 2n\Delta] = (1-p)^{2n} = (1-G/n)^{2n} \quad \text{per } n \rightarrow \infty$$

$$P_{\text{succ}} = e^{-2G} \quad e = 2,7182$$

Throughput Aloha

$$S = GP_{\text{success}} = Ge^{-2G}$$



- Max throughput $p_{\max} = 1/2e$ (18.4%)
- Comportamento bimodale:
 - Per valori bassi di G , $S \approx G$
 - Per valori elevati di G , $S \downarrow 0$
- Le collisioni sono in numero elevato il throughput tende a zero:
 - Instabilità

Slotted ALOHA

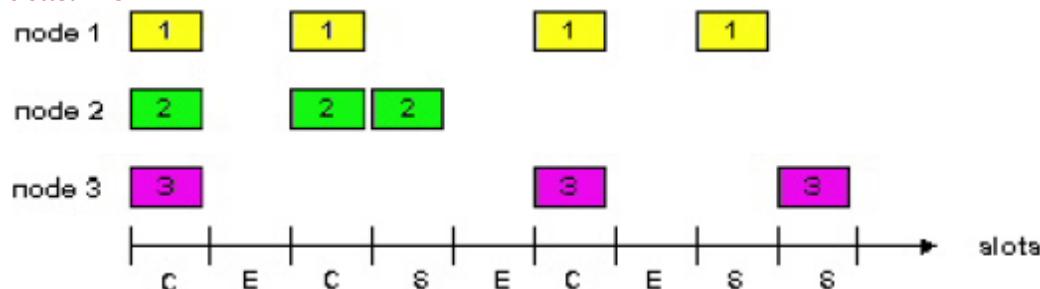
Ipotesi:

- Tutti i pacchetti hanno la stessa dimensione
- Il tempo è suddiviso in slot; ogni slot equivale al tempo di trasmissione di un pacchetto
- I nodi iniziano la trasmissione dei pacchetti solo all'inizio degli slot.
- I nodi sono sincronizzati
- Se in uno slot due o più pacchetti collidono, i nodi coinvolti rilevano l'evento prima del termine dello slot

Operazioni

- Quando a un nodo arriva un nuovo pacchetto da spedire, il nodo attende fino all'inizio dello slot successivo:
 - Se non si verifica una collisione: il nodo può trasmettere un nuovo pacchetto nello slot successivo
 - Se si verifica una collisione: il nodo la rileva prima della fine dello slot e ritrasmette con probabilità p il suo pacchetto durante gli slot successivi

Slotted ALOHA



Pro

- Consente a un singolo nodo di trasmettere continuamente pacchetti alla massima velocità del canale
- È fortemente decentralizzato, ciascun nodo rileva le collisioni e decide indipendentemente quando ritrasmettere.
- È estremamente semplice
- Intervallo di vulnerabilità = X**

Contro

- Una certa frazione degli slot presenterà collisioni e di conseguenza andrà "sprecata"
- Un'altra frazione degli slot rimane vuota, quindi inattiva

L'efficienza dello Slotted Aloha

- L'efficienza è definita come la frazione di slot in cui avviene una trasmissione utile in presenza di un elevato numero di nodi attivi, che hanno sempre un elevato numero pacchetti da spedire.
- Supponiamo N nodi con pacchetti da spedire, ognuno trasmette i pacchetti in uno slot con probabilità p
- La probabilità di successo di un dato nodo = $p(1-p)^{N-1}$
- La probabilità che un nodo arbitrario abbia successo = $Np(1-p)^{N-1}$
- Per ottenere la massima efficienza con N nodi attivi, bisogna trovare il valore p^* che massimizza $Np(1-p)^{N-1} \rightarrow p^*=1/N$
- Per un elevato numero di nodi, ricaviamo che:

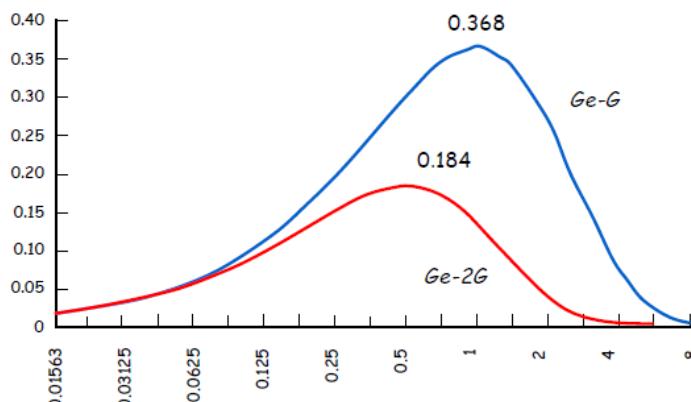
$$\lim_{N \rightarrow \infty} Np * (1 - p^*)^{N-1} = \\ = \lim_{N \rightarrow \infty} \left(1 - \frac{1}{N}\right)^N = \frac{1}{e} = 0,36$$

- Nel caso migliore: solo il 36% degli slot sono utilizzati in modo utile

Throughput Aloha

$$S = GP_{success} = GP[0 \text{ arrivi in } X] = GP[0 \text{ arrivi in } n\Delta] =$$

$$= G(1 - p)^n = G\left(1 - \frac{G}{n}\right)^n \rightarrow Ge^{-G}$$



Applicazioni slotted Aloha



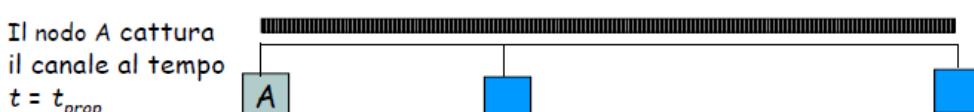
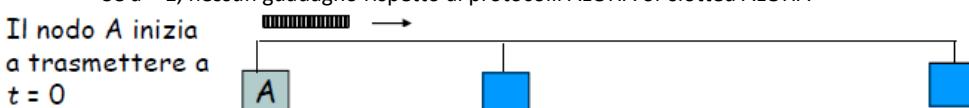
- Alcuni protocolli permettono la prenotazione di slot per effettuare la trasmissione delle frame
- L'asse dei tempi è suddiviso in cicli
- Ogni ciclo ha una serie di mini-slot per effettuare le prenotazioni
- I nodi usano il protocollo slotted Aloha nei mini-slot per effettuare le prenotazioni

Accesso multiplo a rilevazione della portante (CSMA)

- Carrier Sensing Multiple Access:
 - Un nodo ascolta prima di trasmettere
 - Se rileva che il canale è libero, trasmette l'intera frame
 - Se il canale è occupato, il nodo aspetta un altro intervallo di tempo
- Analogia: Se qualcun altro sta parlando, aspettate finché abbia concluso

CSMA

- Un nodo ascolta il canale prima di trasmettere:
 - Se il canale è occupato, attende o applica il backoff (varie opzioni)
 - Se il canale è libero, inizia la trasmissione
 - Intervallo di vulnerabilità è uguale a $2t_{prop}$ (effetto di cattura del canale)
 - Se avviene una collisione, questa interessa l'intera frame
 - Se $a > 1$, nessun guadagno rispetto ai protocolli ALOHA or slotted ALOHA



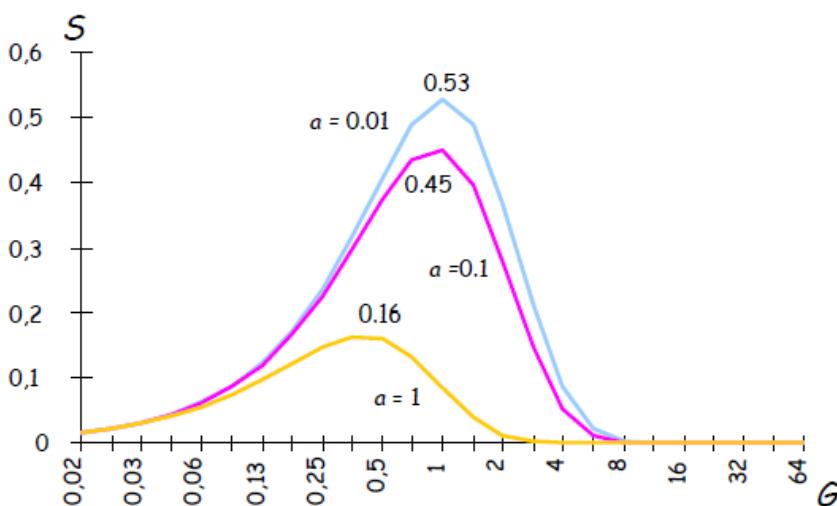
Algoritmi di persistenza

- Si applicano quando un nodo rivelà il canale occupato
- 1-persistent CSMA:
 - Il nodo inizia la trasmissione non appena il canale si libera
 - Basso ritardo e bassa efficienza

- Non-persistent CSMA:
 - Il nodo applica un backoff, quindi effettua un nuovo carrier sensing
 - Alto ritardo e alta efficienza

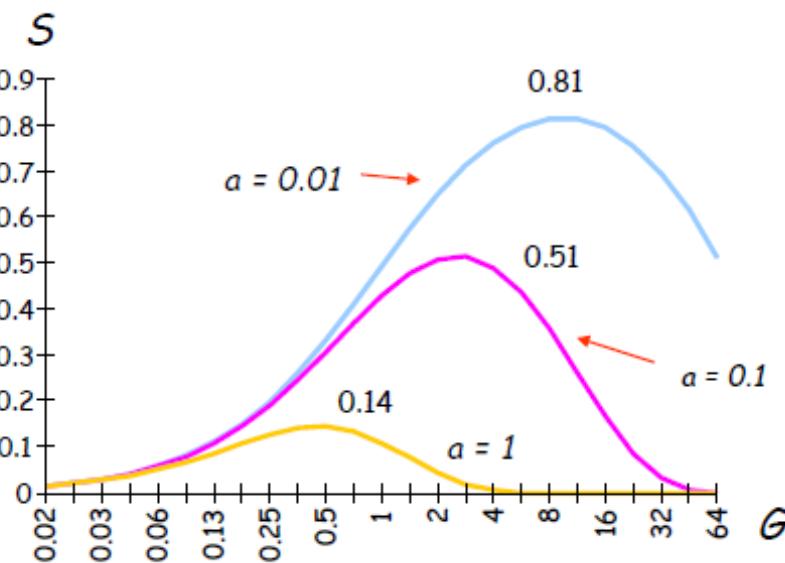
- P-persistent CSMA:
 - Il nodo attende fino a che il canale si libera, quindi:
 - Con probabilità p trasmette
 - Con probabilità $1-p$ attende un breve periodo (mini-slot) ed effettua nuovamente il carrier sensing

Prestazioni 1-persistent CSMA



- Prestazioni:
 - Migliori di Aloha e Slotted Aloha per piccoli valori di α
 - Peggiori di Aloha se $\alpha > 1$

Prestazioni non-persistent CSMA

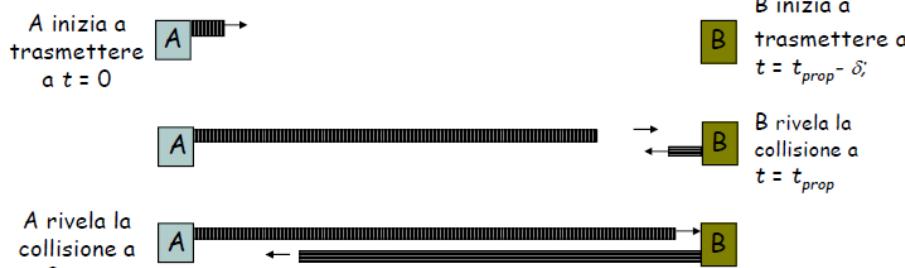


- Valori di throughput più alto rispetto a 1-persistent per piccoli valori di α
- Peggiori di Aloha se $\alpha > 1$

CSMA con Collision Detection (CSMA/CD)

- “Ascolta prima di parlare e mentre parli”:
 - Rivelà le collisioni ed interrompe la trasmissione
 - Un nodo ascolta il canale prima di trasmettere
 - Dopo l'inizio della trasmissione il nodo continua ad ascoltare il canale per rivelare le collisioni
 - Se viene rivelata una collisione, tutti i nodi coinvolti interrompono la trasmissione e rischedulano dopo un intervallo di backoff
- Nel protocollo CSMA, una collisione comporta un periodo di inutilizzazione del canale uguale a al tempo di trasmissione di una frame
- Il protocollo CSMA-CD riduce la durata delle collisioni e quindi aumenta l'efficienza

Rivelazione di una collisione



- Nel caso peggiore i nodi coinvolti nella collisione le rivelano dopo un tempo $t = 2t_{prop}$
 - Minore del tempo di trasmissione di una frame

Ethernet

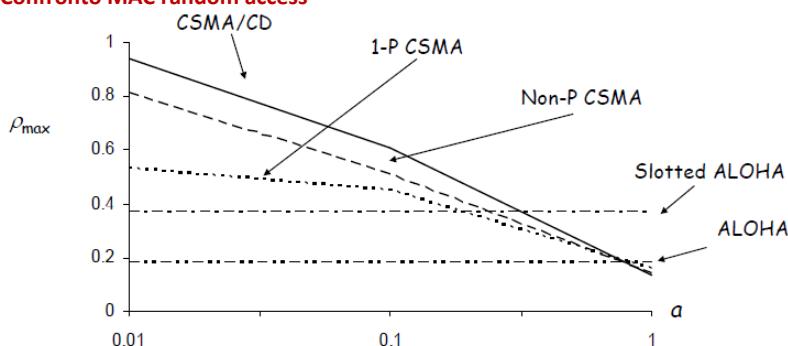
- Lo standard LAN Ethernet LAN è basato sul CSMA-CD:

- 1-persistent CSMA
- $R = 10 \text{ Mbit/s}$
- $t_{prop} = 51.2 \mu\text{s}$
 - 512 bit = 64 byte slot
 - Distanza massima 2.5 km + 4 repeaters

- Truncated Binary Exponential Backoff

- Dopo l' n -ma collisione, il tempo di backoff è scelto tra i valori $\{0, 1, \dots, 2k - 1\}$, dove $k = \min(n, 10)$

Confronto MAC random access



- Per piccoli valori di α : CSMA-CD ha il throughput migliore
- Per grandi valori di α : Le prestazioni migliori sono di Aloha & slotted Aloha

Protocolli MAC ad accesso controllato

• Protocolli MAC a suddivisione del canale:

- Condividono il canale equamente ed efficientemente con carichi elevati
- Inefficienti con carichi non elevati

• Protocolli MAC ad accesso casuale:

- Efficienti con carichi non elevati: un singolo nodo può utilizzare interamente il canale
- Carichi elevati: eccesso di collisioni

• Protocolli ad accesso controllato:

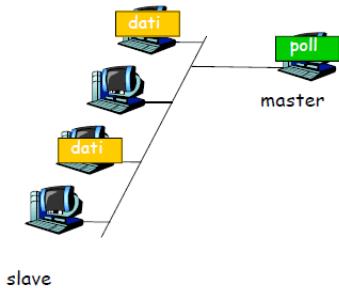
- Prendono il meglio dei due protocolli precedenti

Protocolli ad accesso controllato

Protocollo polling

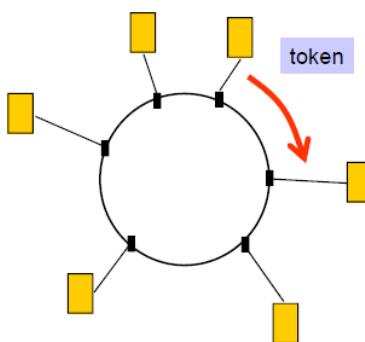
- Un nodo principale sonda "a turno" gli altri.
- In particolare:

- Elimina le collisioni
- Elimina gli slot vuoti
- Ritardo di polling
- Se il nodo principale (master) si guasta, l'intero canale resta inattivo.



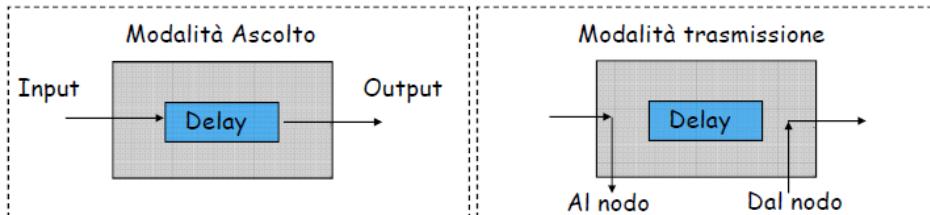
Protocollo token-passing

- Un messaggio di controllo circola fra i nodi seguendo un ordine prefissato
- Messaggio di controllo (token)
- In particolare:
 - Decentralizzato
 - Altamente efficiente
 - Il guasto di un nodo può mettere fuori uso l'intero canale



Application: Token-Passing Rings

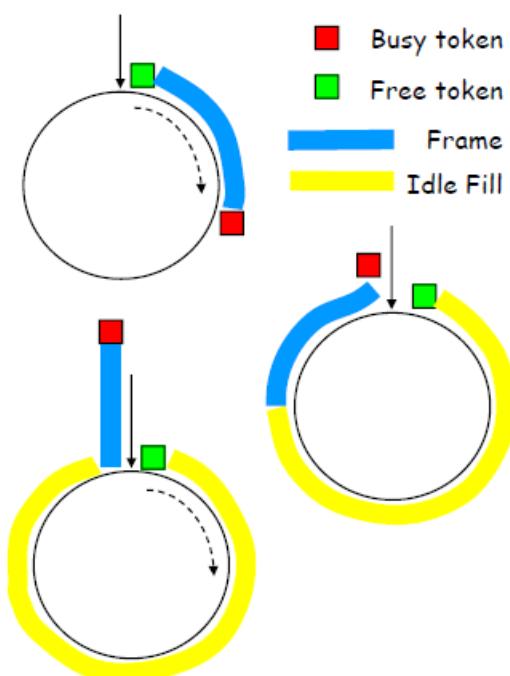
- Il flag delle frame può essere il token:
 - Free token = 01111110
 - Busy token = 01111111



- I nodi che sono pronti a trasmettere aspettano il token e cambiano il bit finale del flag per convertire il token da free a busy
- Il nodo che ha il token trasmette, al termine della trasmissione reinserisce il free token

Metodi di reinserimento del token

- **Ring Latency:**
 - Numero di bit che possono essere trasmessi simultaneamente sul ring
- **Multi-token operation:**
 - Il Free token è trasmesso immediatamente dopo l'ultimo bit di una frame
- **Single-token operation:**
 - Il Free token è inserito dopo che l'ultimo bit del busy token è ritornato al nodo origine
 - Il tempo di trasmissione uguale almeno alla ring latency
 - Se la frame è maggiore della ring latency, è equivalente al multi-token operation
- **Single-Frame operation:**
 - Il Free token è inserito dopo che il nodo emittente ha ricevuto l'ultimo bit della sua frame
 - È equivalente ad aggiungere alla frame un trailer uguale alla ring latency



Throughput del protocollo Token Ring

- **Definizioni:**

- τ : tempo richiesto ad un bit per circolare nel ring
- T : tempo di trasmissione di una frame

- **Multi-token operation:**

- Assumiamo che la rete è caricata al massimo, tutti gli M nodi trasmettono una frame dopo aver ricevuto il token
- Equivale ad un protocollo di tipo polling con un tempo di servizio limitato a X

$$\rho_{\max} = \frac{MT}{\tau + MT} = \frac{1}{1 + \tau / MT} = \frac{1}{1 + a / M}$$

$$a = \frac{\tau}{T} \quad \text{è la ring latency normalizzata}$$

- **Single-frame operation:**

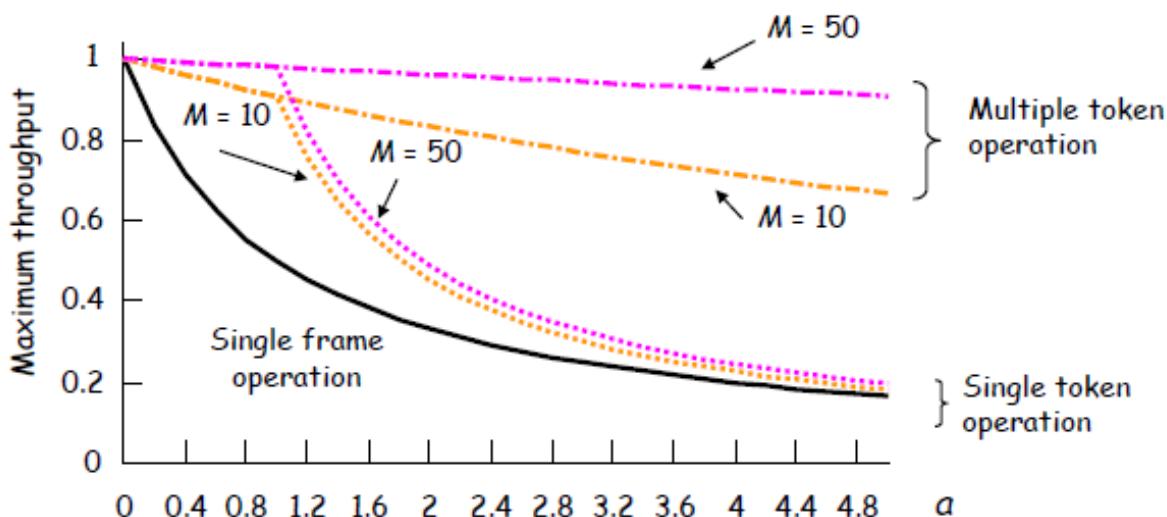
- Il tempo di trasmissione di una frame è uguale al massimo tra T e τ

$$\rho_{\max} = \frac{MT}{\tau + M \max(T, \tau)} = \frac{1}{\frac{a}{M} + \max(1, a)}$$

- **Single-token operation:**

- Il tempo di trasmissione di una frame è uguale a $t + \tau$

$$\rho_{\max} = \frac{MT}{\tau + M(T + \tau)} = \frac{1}{1 + a(1 + \frac{1}{M})}$$



- Se $a \ll 1$: è accettabile qualsiasi strategia di reinserimento del token
- Se $a \approx 1$: è accettabile la modalità single token operation
- Se $a > 1$: è necessaria la modalità multitone operation

Protocolli MAC: riepilogo

- Cosa si può fare con un canale condiviso?

- Suddivisione del canale per: tempo, frequenza, codice:

- TDM, FDM

- Accesso casuale:

- ALOHA, S-ALOHA, CSMA, CSMA/CD
- Rilevamento della portante: facile in alcune tecnologie (cablate), difficile in altre (wireless)
- CSMA/CD usato in Ethernet
- CSMA/CA usato in 802.11

- Ad accesso controllato:

- Polling con un nodo principale; a passaggio di token
- Bluetooth, FDDI, IBM Token Ring

"Error Recovery"

End-to-End vs. Hop-by-Hop

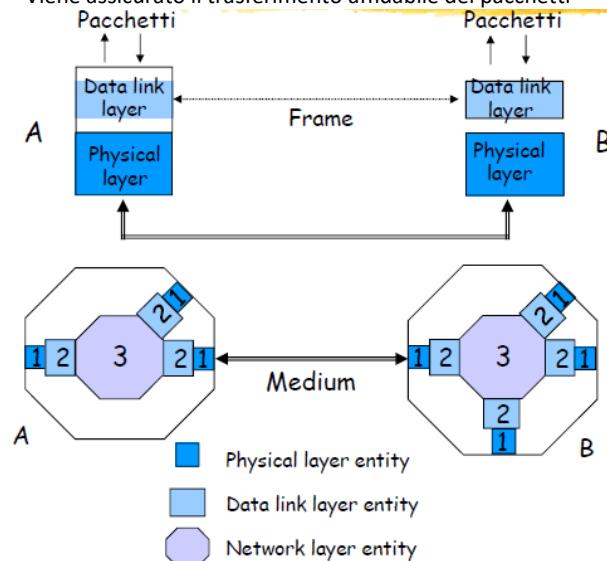
- Una funzione può essere eseguita:
 - Da estremo a estremo (**end-to-end**) (**Strato di trasporto**)
 - Tratta per tratta (**hop-by-hop**) (**Strato di Data link**)

Esempi:

- Controllo d'errore eseguito in ogni hop del percorso di rete oppure solamente tra sorgente e destinazione
- Controllo di flusso eseguito in ogni hop del percorso di rete oppure solamente tra sorgente e destinazione

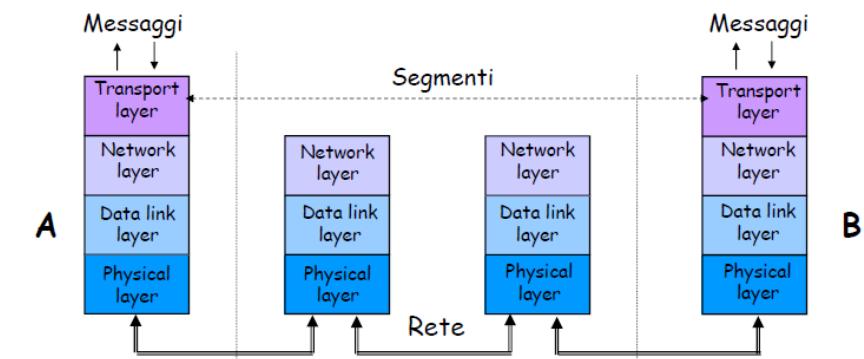
Controllo d'errore nello strato di Data Link

- Lo strato di data link opera punto-punto tra due elementi di rete direttamente connessi
- Le frame possono subire errori, ma è preservata la sequenza
- Lo strato di Data link esegue la rivelazione degli errori e la ritrasmissione delle frame errate
- Viene assicurato il trasferimento affidabile dei pacchetti

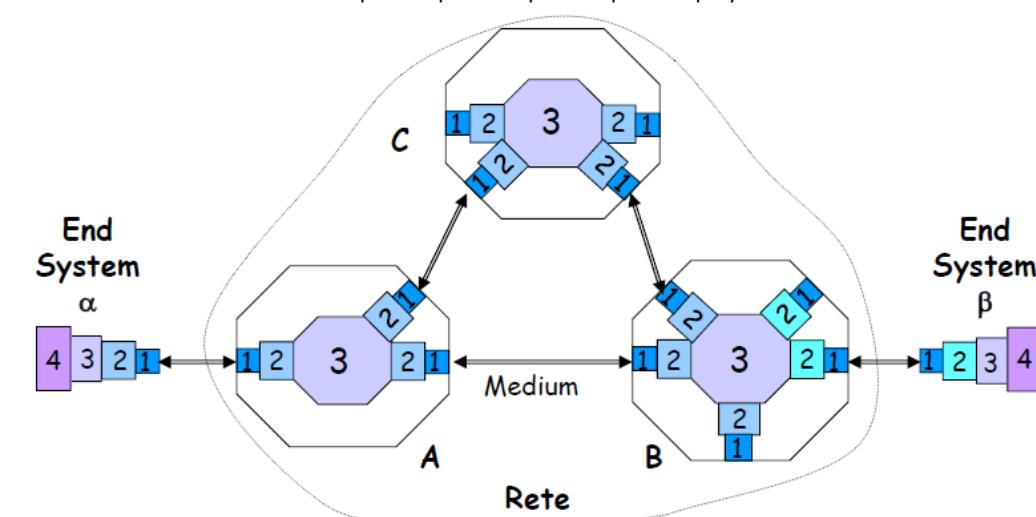


Controllo d'errore nello strato di Trasporto

- Il protocollo di strato di trasporto (es. TCP) emette i segmenti ed esegue end-to-end sia la rivelazione d'errore che la ritrasmissione
- La rete è considerata inaffidabile



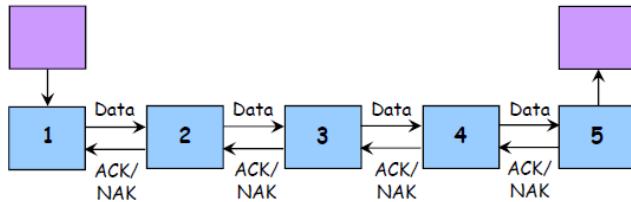
- I segmenti possono subire ritardi elevati, subire errori, essere persi o arrivare fuori sequenza
- Il controllo d'errore end-to-end è più complesso rispetto a quello hop-by-link



End-to-End vs. Hop-by-Hop

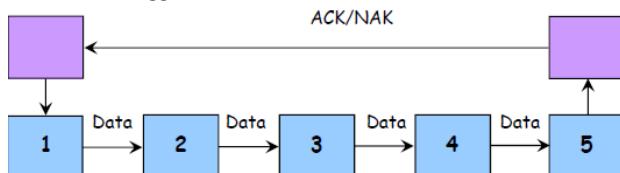
- **Hop-by-hop:**

- Non può assicurare la correttezza end-to-end
- Recupero più veloce



- **End-to-end:**

- Semplicità delle procedure di rete
- Maggiore scalabilità



Automatic Repeat Request (ARQ)

- **Obiettivo:**

- Assicurare che una sequenza di PDU sia consegnata in ordine e senza errori o duplicazioni in presenza di un servizio offerto dagli strati sottostanti che introduce errori e/o perdite

- **Possibili procedure alternative:**

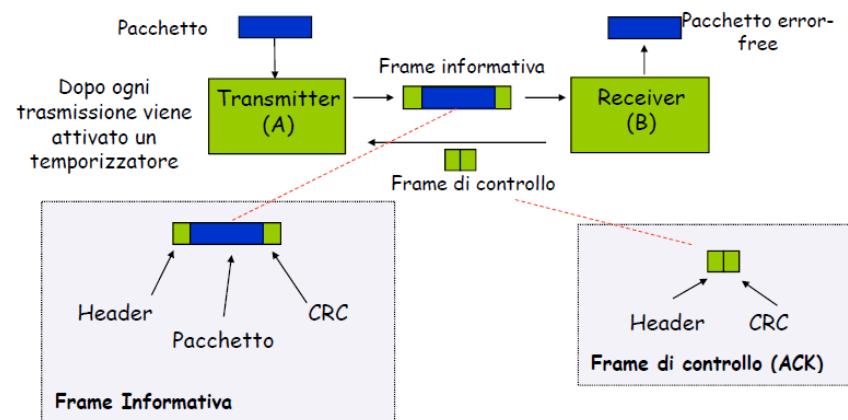
- Stop-and-Wait
- ARQ Go-Back N ARQ
- Selective Repeat ARQ

- Elementi chiave delle procedure ARQ:

- Codici di rivelazione d'errore
- Riscontri positivi (ACK)
- Riscontri negativi (NACK)
- Timeout

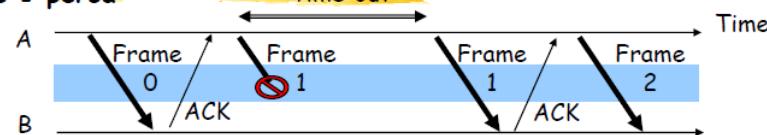
Stop-and-Wait ARQ

- L'entità A trasmette una frame ed aspetta l'ACK

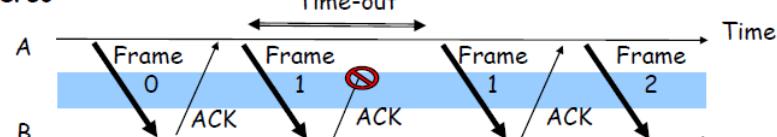


Numeri di Sequenza

Frame 1 perso



ACK perso

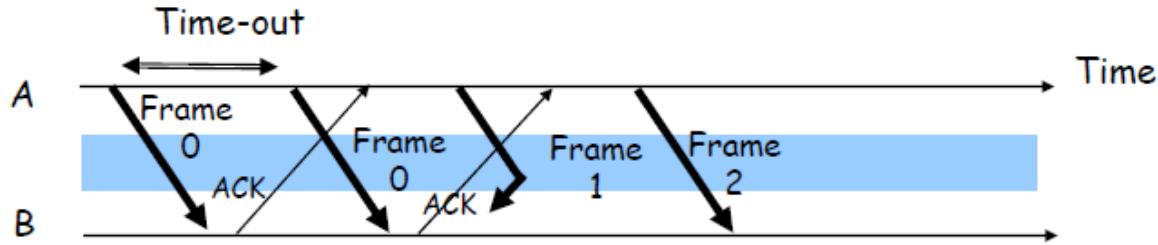


- L'entità emittente (A) si comporta sempre nello stesso modo

- Nel secondo caso, l'entità ricevente (B) riceve la frame 1 due volte (duplicazione)

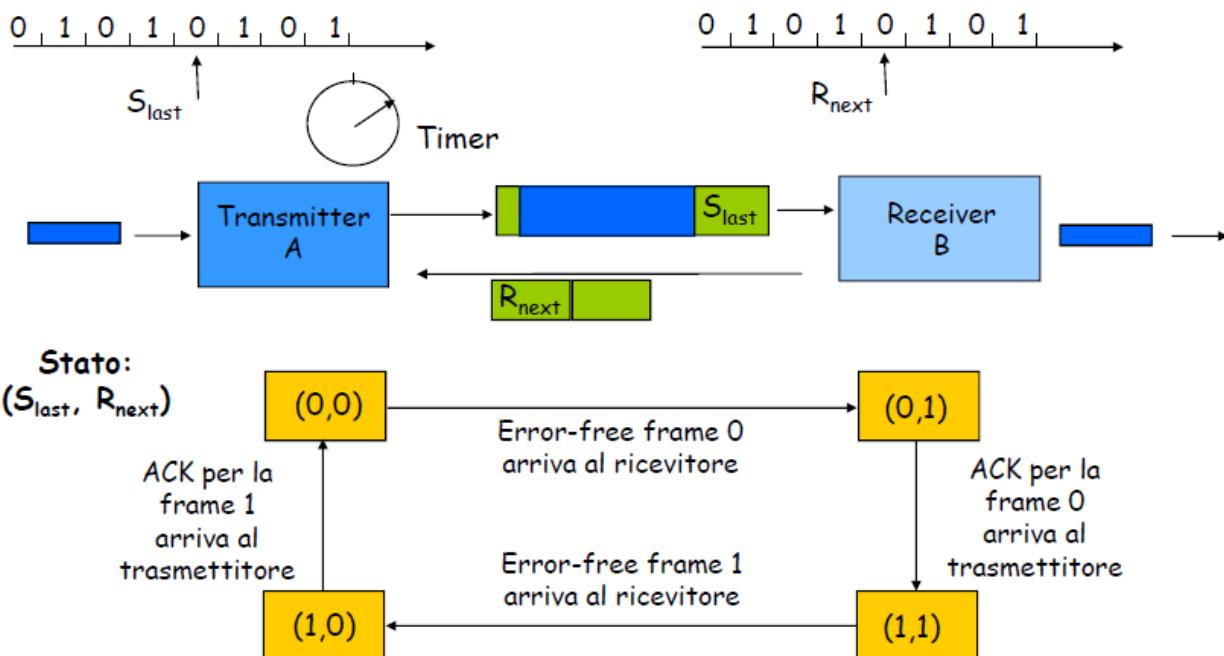
- B rivela la duplicazione mediante il numero di sequenza (Slast) contenuto nell'header di ciascuna frame

- Esaurimento prematuro del time-out:



- La stazione emittente interpreta in modo scorretto gli ACK:
 - Per il secondo ACK riscontra la frame 1 che invece è persa
- Occorre inserire il numero di sequenza anche negli ACK (R_{next}):
 - Indica il numero di sequenza della prossima frame che il ricevitore si aspetta di ricevere
 - Implicitamente riscontra tutte le frame con numero di sequenza $R' < R$

Numero di sequenza 1-Bit



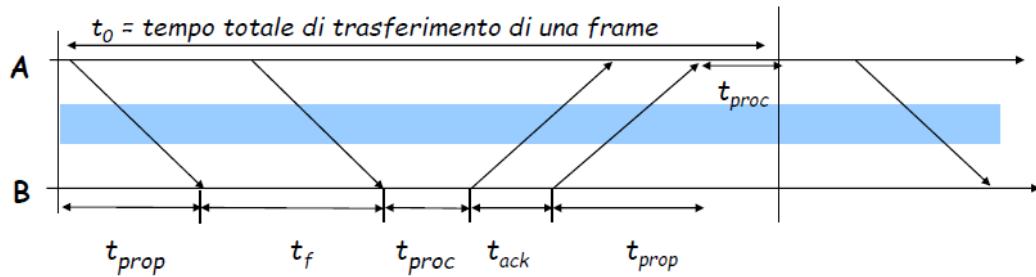
Stop-and-Wait ARQ (Trasmettitore)

- Stato Ready:
 - Attesa di una richiesta di invio di un pacchetto dallo strato superiore
 - Quando arriva una richiesta, si trasmette la frame con numero di sequenza S_{last} e completa di CRC
 - Transizione nello stato Wait
- Stato Wait:
 - Attesa del riscontro della frame emessa o dell'esaurimento del timeout (la ricezione delle richieste dallo strato superiore sono bloccate)
 - Se il timeout scade viene ritrasmessa la frame e viene riavviato il timer
 - Se viene ricevuto un ACK:
 - Se il numero di sequenza non è corretto l'ACK è ignorato
 - Se il numero di sequenza è corretto ($R_{next} = S_{last} + 1$), la frame è accettata e si torna nello stato Ready

Stop-and-Wait ARQ (Receiver)

- Sempre nello stato Ready:
 - Attesa dell'arrivo di una nuova frame
 - Quando arriva una frame viene eseguito il controllo d'errore (CRC)
 - Se non sono rivelati errori e il numero di sequenza è corretto ($S_{last} = R_{next}$):
 - La frame viene accettata
 - Viene aggiornato il valore di R_{next}
 - Viene emesso l'ACK con valore R_{next}
 - Il pacchetto è consegnato allo strato superiore
 - Se non sono rivelati errori e il numero di sequenza non è corretto:
 - La frame viene scartata
 - Viene emesso un ACK con R_{next} (ACK duplicato)
 - Se sono rivelati errori:
 - La frame viene scartata

Modello Stop-and-Wait ARQ



$$\begin{aligned} t_0 &= 2t_{prop} + 2t_{proc} + t_f + t_{ack} && \text{Lunghezza di una frame} \\ &= 2t_{prop} + 2t_{proc} + \frac{n_f}{R} + \frac{n_a}{R} && \text{Lunghezza di un ACK} \\ &&& \text{Bit rate canale} \end{aligned}$$

Efficienza su un canale senza errori

- Rate di trasmissione efficace:

$$R_{eff}^0 = \frac{\text{numero di bit informativi consegnati a destinazione}}{\text{tempo totale necessario per la consegna dei bit informativi}} = \frac{n_f - n_o}{t_0},$$

bit di overhead

- Efficienza di trasmissione:

$$\eta_0 = \frac{R_{eff}}{R} = \frac{\frac{n_f - n_o}{t_0}}{R} = \frac{\frac{n_f - n_o}{t_0}}{1 + \frac{n_a}{n_f} + \frac{2(t_{prop} + t_{proc})R}{n_f}}$$

Effetto dell'overhead di una frame

Effetto di un ACK

Effetto del prodotto Banda-Ritardo

Esempio: Impatto del prodotto banda-ritardo

- $n_f=1250$, byte = 10000 bits, $n_a = n_o = 25$ byte = 200 bit

2xDelayxBW Efficiency	1 ms	10 ms	100 ms	1 sec
	200 km	2000 km	20000 km	200000 km
1 Mbit/s	10^3	10^4	10^5	10^6
	88%	49%	9%	1%
1 Gbit/s	10^6	10^7	10^8	10^9
	1%	0.1%	0.01%	0.001%

- La tecnica Stop-and-Wait non è efficiente in link ad alta velocità o con elevati ritardi di propagazione

Efficienza su un canale con errori

- Sia $1 - P_f$ = probabilità che una frame arrivi senza errori
- $1 / (1 - P_f)$ = numero medio di trasmissioni necessarie per avere una trasmissione corretta di una frame
- $T_0 / (1 - P_f)$ = tempo medio di trasferimento di una frame

$$\eta_{SW} = \frac{R_{eff}}{R} = \frac{\frac{n_f - n_o}{t_0}}{\frac{1 - P_f}{R}} = \frac{\frac{n_f - n_o}{t_0}}{1 + \frac{n_a}{n_f} + \frac{2(t_{prop} + t_{proc})R}{n_f}} \cdot (1 - P_f)$$

Effetto della probabilità di perdita delle frame

Esempio: Impatto del Bit Error Rate

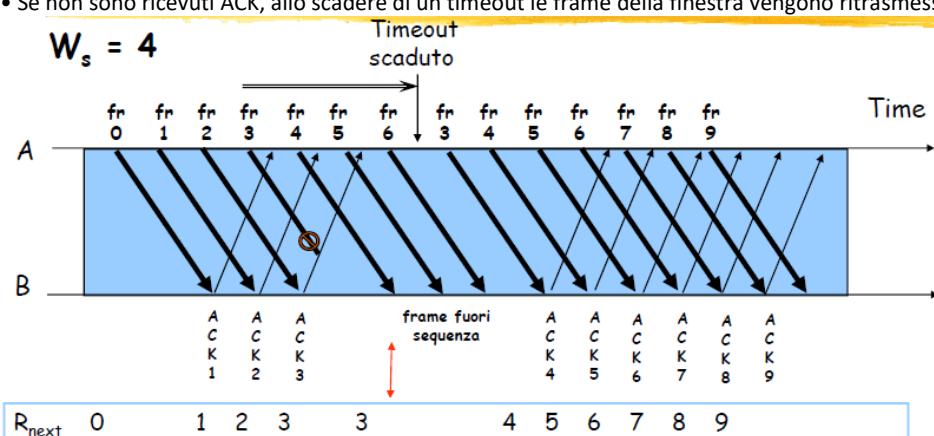
- $n_f = 1250 \text{ byte} = 10000 \text{ bit}$, $n_a = n_o = 25 \text{ byte} = 200 \text{ bit}$
- Calcolo dell'efficienza per un BER $p=0, 10^{-6}, 10^{-5}, 10^{-4}$

$1 - P_f$ Efficiency	0	10^{-6}	10^{-5}	10^{-4}
R=1 Mbps	1	0.99	0.905	0.368
$T_{\text{prop}}=1 \text{ ms}$	88%	86.6%	79.2%	32.2%

- Gli errori introducono un effetto significativo quando il prodotto $n_f p$ si avvicina ad 1

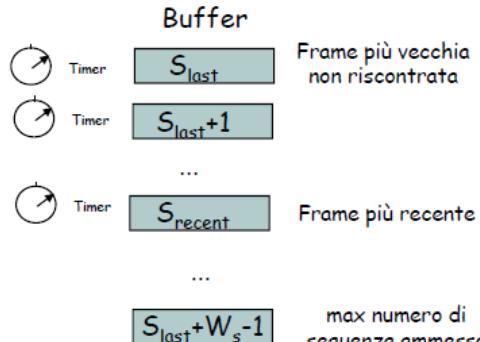
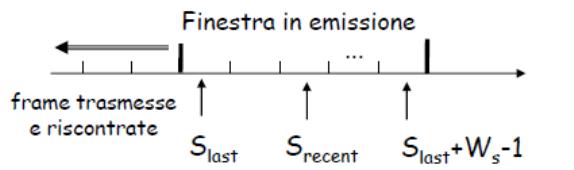
Go-back N ARQ

- Miglioramento del protocollo Stop-and-Wait
- Elimina le attese dei riscontri:
 - Il canale è mantenuto occupato inviando altre frame
 - Utilizza una **finestra in trasmissione** di ampiezza W_s frame
 - Usa m bit per la numerazione delle frame
- Se vengono ricevuti gli ACK delle frame emesse prima di esaurire la finestra, la finestra è aggiornata e la trasmissione delle frame può continuare
- Se la finestra si esaurisce, la trasmissione viene interrotta in attesa degli ACK
- Se non sono ricevuti ACK, allo scadere di un timeout le frame della finestra vengono ritrasmesse

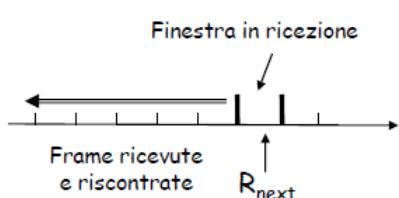


Go-Back-N Transmitter & Receiver

- Transmitter:



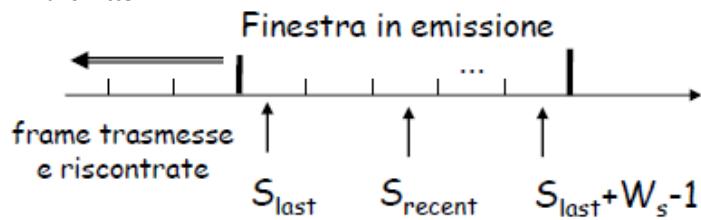
- Receiver:



- Il Receiver accetta solo frame corrette e in sequenza (con numero di sequenza = R_{next})
- Quando arriva una nuova frame in sequenza, viene incrementato di uno R_{next} , quindi la finestra in ricezione slitta di una unità

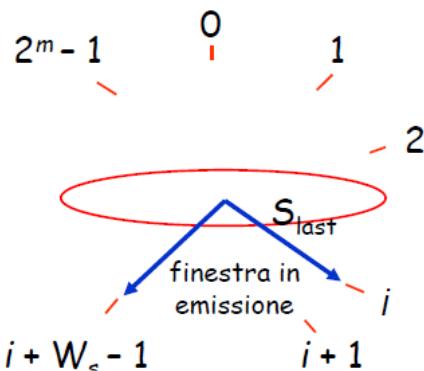
Sliding window

- Transmitter:



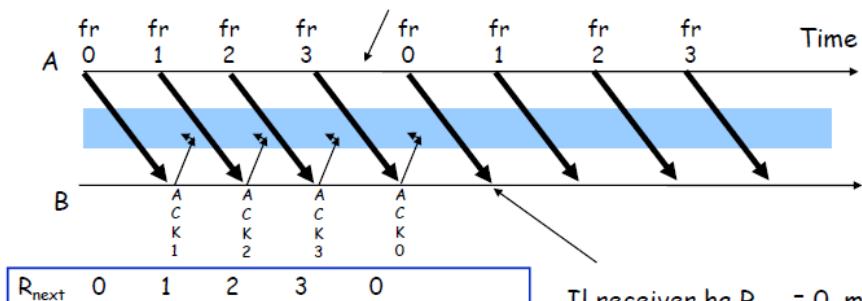
- Il Transmitter attende gli ACK (con numero di sequenza $S \geq S_{last}$)
- Quando arriva un ACK, con numero di sequenza S , viene posto $S_{last} = S$
- L'estremo superiore della finestra sarà quindi $S_{last} + W_s - 1$

Numeri di sequenza a m bit



Dimensione massima della finestra

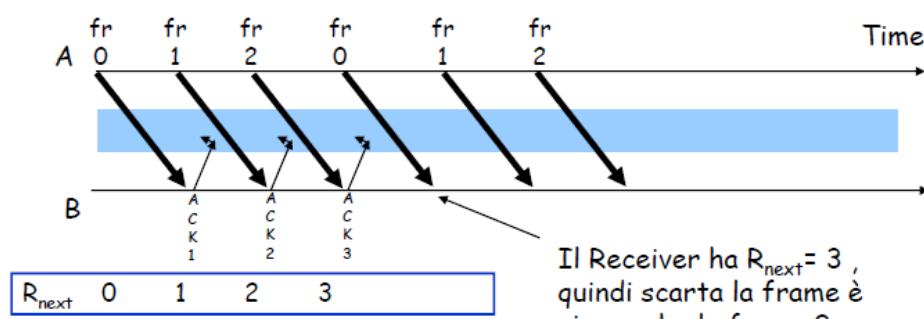
$$W_s = M = 2^m = 4$$



Il massimo valore della finestra è uguale a $W_s = M = 2^m$

Il receiver ha $R_{next} = 0$, ma non è in grado di distinguere se il suo ACK per la frame 0 è stato ricevuto e quindi la frame arrivata è nuova oppure si tratta della ritrasmissione della vecchia frame 0

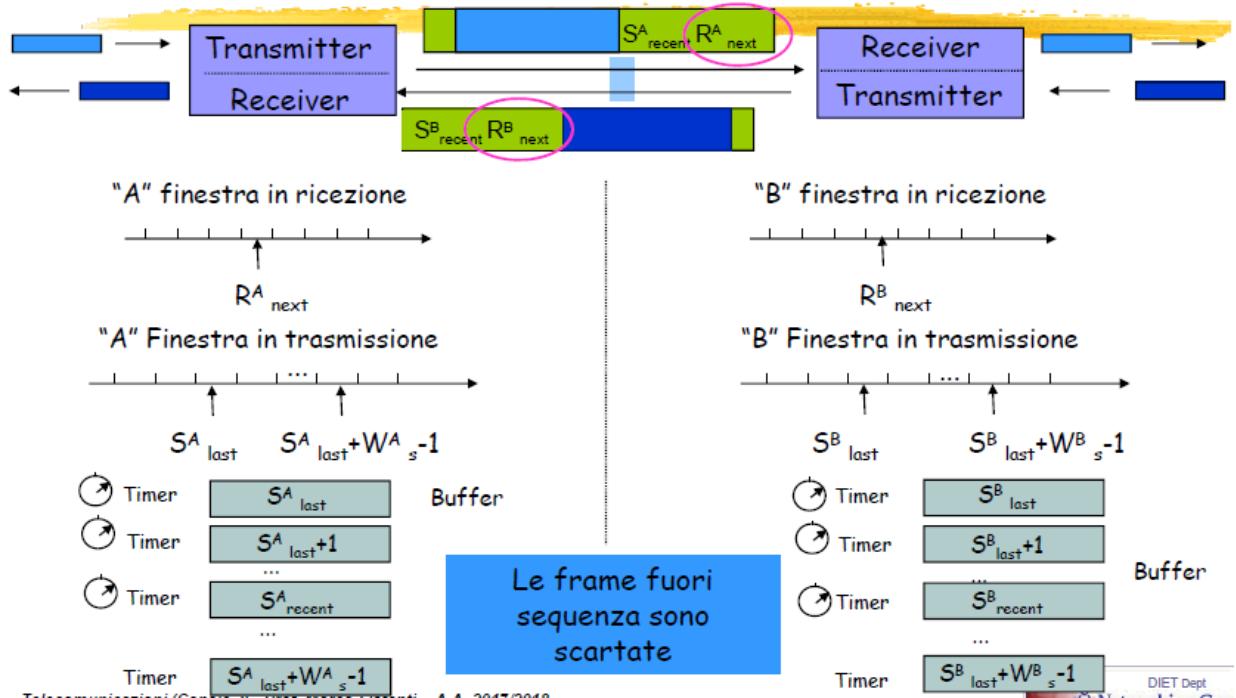
$$W_s = M = 2^m - 1 = 3$$



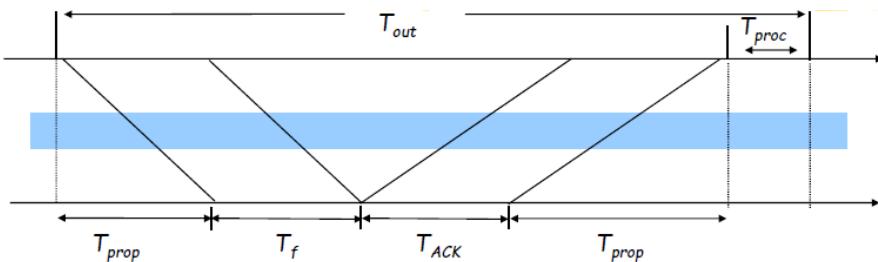
Il Receiver ha $R_{next} = 3$, quindi scarta la frame 0, sicuro che la frame 0 arrivata è una duplicazione e quindi la scarta

Il massimo valore della finestra è uguale a $W_s = M - 1 = 2^m - 1$

Piggybacking



Dimensionamento della finestra e del timeout



- Il valore del Timeout (T_{out} componenti) deve essere la somma delle seguenti:

- Due tempi di propagazione + un tempo di processing = $2 T_{prop} + T_{proc}$
- Un tempo di trasmissione di una frame informativa T_f
- Un tempo di trasmissione della frame ACK, T_{ACK}

- W_s deve essere grande abbastanza da poter mantenere il canale occupato per tutto il periodo T_{out}

Dimensione della finestra vs. prodotto banda-ritardo

Frame = 1250 bytes = 10,000 bits, $R = 1 \text{ Mbps}$		
$2(t_{prop} + t_{proc})$	$2 \times \text{Delay} \times \text{BW}$	Window
1 ms	1000 bits	1
10 ms	10,000 bits	2
100 ms	100,000 bits	11
1 second	1,000,000 bits	101

Efficienza del Go-Back-N

- Tempo di trasferimento di una frame:

$$t_{GBN} = t_f (1 - P_f) + P_f \left\{ t_f + \frac{W_s t_f}{1 - P_f} \right\} = t_f + P_f \frac{W_s t_f}{1 - P_f}$$

- Efficienza:

$$\eta_{GBN} = \frac{\frac{n_f - n_o}{t_{GBN}}}{R} = \frac{1 - \frac{n_o}{n_f}}{1 + (W_s - 1)P_f} (1 - P_f)$$

Impatto del BER su GBN

- $n_f = 1250 \text{ bytes} = 10000 \text{ bits}$, $n_a = n_o = 25 \text{ bytes} = 200 \text{ bits}$
- Random bit errors with $p=0, 10^{-6}, 10^{-5}, 10^{-4}$
- $R = 1 \text{ Mbps}$, Delay = 100 ms
- $1 \text{ Mbps} \times 100 \text{ ms} = 100000 \text{ bits} = 10 \text{ frames} \rightarrow W_s = 11$

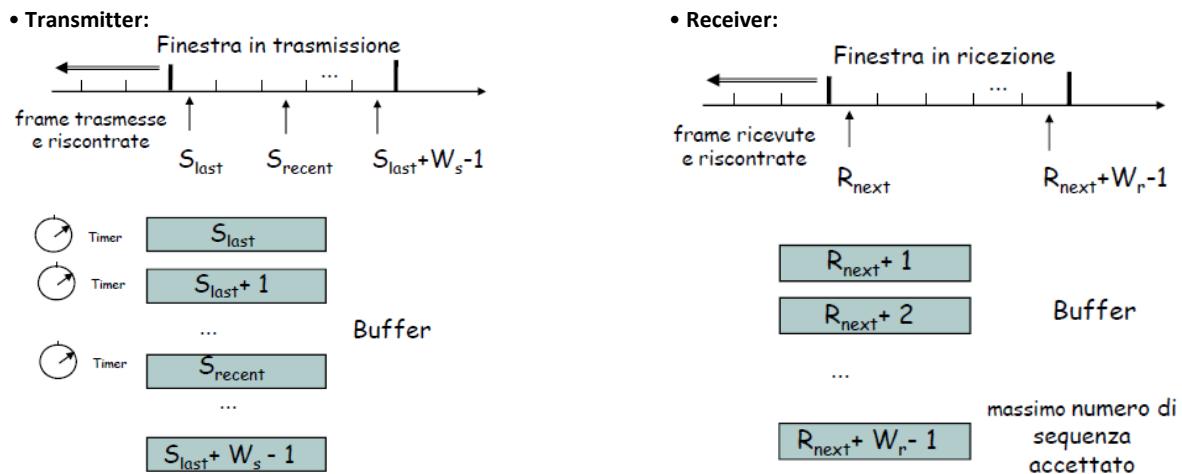
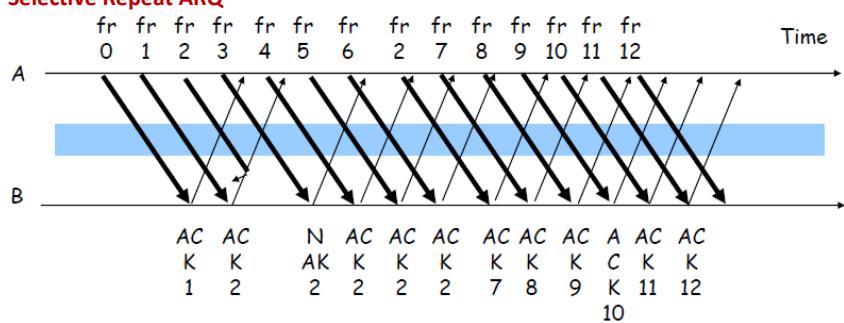
Efficiency	0	10^{-6}	10^{-5}	10^{-4}
S&W	8.9%	8.8%	8.0%	3.3%
GBN	98%	88.2%	45.4%	4.9%

- Go-Back-N è migliore di S&W nei casi di elevato valore del prodotto banda ritardo
- Go-Back-N diviene inefficiente se il BER cresce

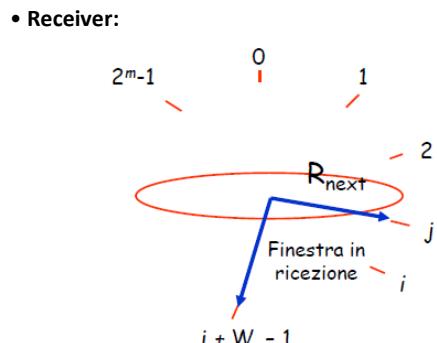
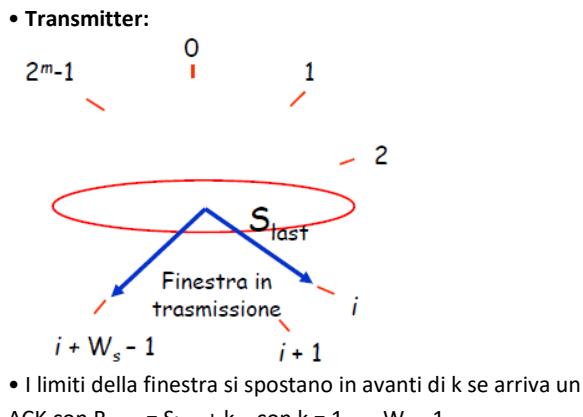
Selective Repeat ARQ

- Go-Back-N ARQ è inefficiente poiché, in caso di ritrasmissione, viene riemesso un numero elevato di frame, anche se ricevute correttamente dal receiver
- Selective Repeat ritrasmette solo le frame che sono state perse:
 - L'esaurimento del Timeout determina la ritrasmissione solo del frame corrispondente
 - La ricezione di un NAK causa la ritrasmissione della trama non riscontrata più vecchia
- Il Receiver gestisce una finestra in ricezione che indica i numeri di sequenza che possono essere accettati:
 - Frame corrette, ma fuori sequenza con numero di sequenza compreso nella finestra in ricezione non sono scartate, ma sono bufferizzate
 - Un arrivo di una frame con R_{next} determina lo scorrimento della finestra in trasmissione

Selective Repeat ARQ

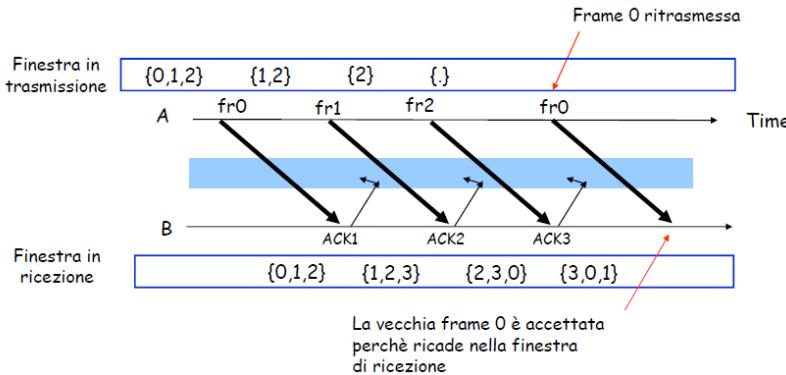


Finestre in trasmissione e ricezione

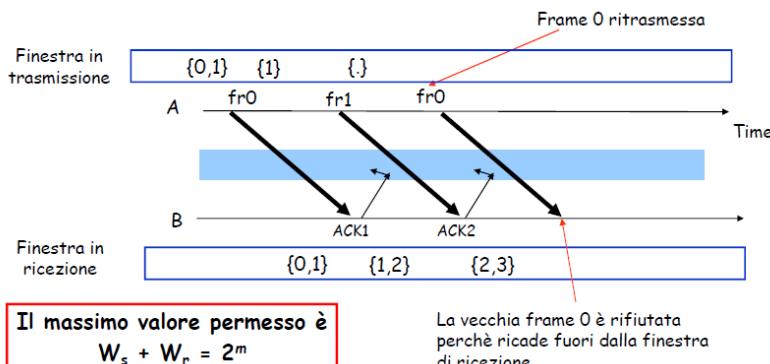


Valori massimi di W_s e W_r

- Esempio: $M=2^2=4$, $W_s=3$, $W_r=3$

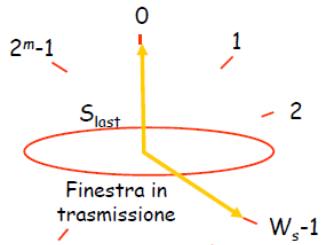


- Esempio: $M=2^2=4$, $W_s = 2$, $W_r = 3$

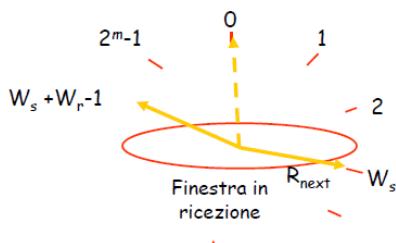


Perchè $W_s + W_r = 2^m$

- Il Transmitter emette le frame da 0 a W_s-1 ; la finestra di trasmissione è vuota
- Tutte le frame arrivano al receiver
- Tutti gli ACKs sono persi
- Il Transmitter riemette la frame 0



- La finestra di ricezione inizia a {0, ..., Wr}
- La finestra di ricezione slitta a {Ws, ..., Ws+Wr-1}
- Il ricevitore rifiuta la frame 0 perché è fuori dalla finestra di ricezione



Efficienza del Selective Repeat

- Assumiamo P_f = frame loss probability
- Il numero di trasmissioni richieste per trasferire una frame è: $1 / (1 - P_f)$
- Il tempo di trasferimento è quindi: $t_f / (1-P_f)$
- L'efficienza è data da:

$$\eta_{SR} = \frac{\frac{n_f - n_o}{t_f / (1 - P_f)}}{R} = \left(1 - \frac{n_o}{n_f}\right)(1 - P_f)$$

Esempio: Impatto del BER sul Selective Repeat

- $n_f = 1250 \text{ bytes} = 10000 \text{ bits}$, $n_a = n_o = 25 \text{ bytes} = 200 \text{ bits}$
- Random bit errors with $p=0, 10^{-6}, 10^{-5}, 10^{-4}$
- $R = 1 \text{ Mbps}$, Delay = 100 ms

Efficiency	0	10^{-6}	10^{-5}	10^{-4}
S&W	8.9%	8.8%	8.0%	3.3%
GBN	98%	88.2%	45.4%	4.9%
SR	98%	97%	89%	36%

- Il Selective Repeat ha prestazioni migliori rispetto a GBN e S&W, ma l'efficienza diminuisce al crescere del BER

Confronto tra i metodi ARQ

Assumiamo n_a e n_o trascurabili rispetto a n_f , e $L = 2(t_{prop} + t_{proc}) R / n_f = (W_s - 1)$

- Selective-Repeat:

$$\eta_{SR} = (1 - P_f) \left(1 - \frac{n_o}{n_f}\right) \approx (1 - P_f)$$

- Go-Back-N:

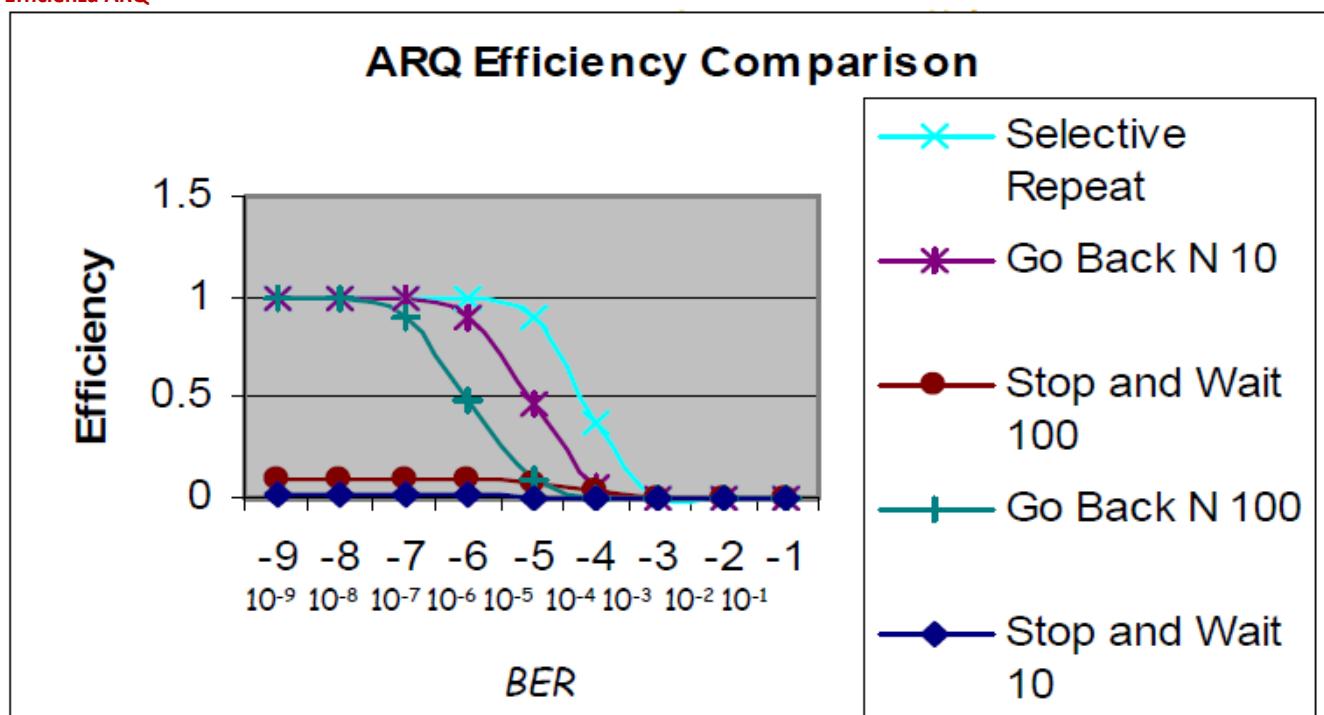
$$\eta_{GBN} = \frac{1 - P_f}{1 + (W_s - 1)P_f} = \frac{1 - P_f}{1 + LP_f}$$

- Stop-and-Wait:

$$\eta_{SW} = \frac{(1 - P_f)}{1 + \frac{n_a}{n_f} + \frac{2(t_{prop} + t_{proc})R}{n_f}} \approx \frac{1 - P_f}{1 + L}$$

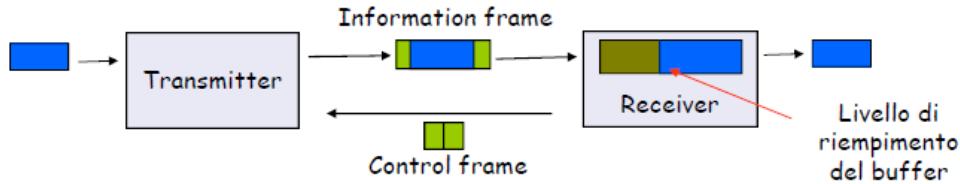
- Per $P_f \approx 0$, SR & GBN uguali
- Per $P_f \rightarrow 1$, GBN & SW uguali

Efficienza ARQ



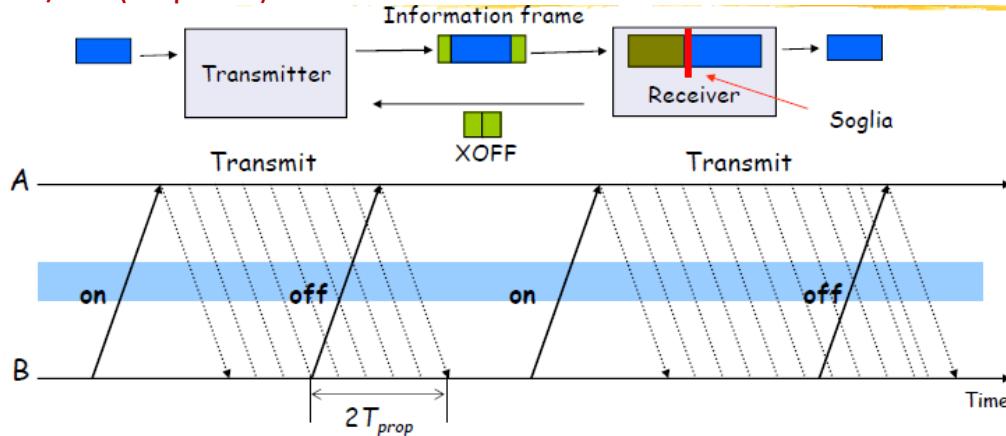
"Flow Control" e "Protocolli PPP e HDLC"

Flow Control



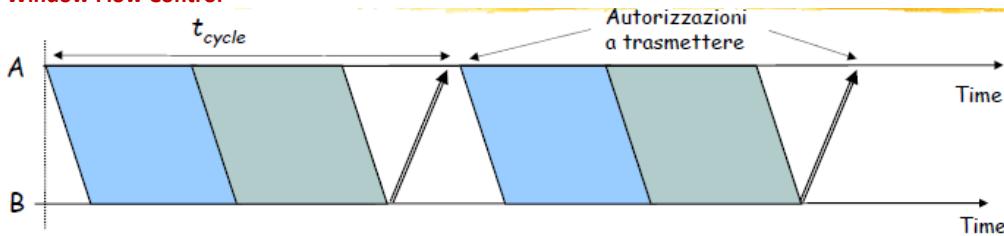
- Il ricevitore dispone di un buffer limitato per memorizzare le frame entranti
- Nel buffer di ricezione si possono verificare fenomeni di overflow a causa di:
 - Differenza tra il rate di arrivo delle frame e il rate con cui il ricevitore elabora le frame
 - Picchi nell'arrivo delle frame
- Il Flow Control ha lo scopo di prevenire gli overflow del buffer di ricezione regolando il tasso di emissione delle frame da parte del Transmitter

XON / XOFF (Backpressure)



- Si deve attivare il segnale di Off in modo da evitare la perdita di pacchetti:
 - Lo spazio disponibile nel buffer deve essere almeno uguale a $2 T_{prop} R$ bit

Window Flow Control



- Finestra scorrevole di ampiezza W_s uguale al buffer disponibile:
 - Il Transmitter non può in nessun caso emettere più di W_s frame
- Gli ACK possono essere interpretati come permessi a trasmettere e possono regolare il rate di trasmissione
- Problemi:**
 - Scelta della dimensione della finestra
 - Interazione tra rate di trasmissione e ritrasmissioni
 - TCP separa error & flow control

Il protocollo PPP

Protocolli di data link punto-punto

- Un mittente, un destinatario:**
 - Non è necessaria la funzione di controllo di accesso al mezzo (MAC)
 - Non occorre indirizzamento MAC esplicito
 - Il collegamento potrebbe essere una linea telefonica seriale commutata
- Protocolli punto-punto DLC più diffusi:**
 - PPP (point-to-point protocol) [RFC 1547]
 - HDLC (high-level data link control)

Funzioni del PPP

- Framming dei pacchetti:**
 - Il protocollo PPP incapsula un pacchetto a livello di rete all'interno del un pacchetto PPP a livello di link
- Trasparenza:**
 - Il protocollo PPP non deve porre alcuna restrizione ai dati che sono contenuti nel pacchetto a livello di rete
- Rilevazione degli errori (ma non la correzione)**
- Disponibilità della connessione:**
 - Il protocollo deve rilevare la presenza di eventuali guasti a livello di link e segnalare l'errore al livello di rete
- Negoziazione degli indirizzi di rete:**
 - PPP deve fornire un meccanismo alle entità di strato di rete per ottenere o configurare gli indirizzi di rete

Funzioni non coperte dal PPP

- Correzione degli errori
- Controllo di flusso
- Controllo di sequenza:
 - Il protocollo PPP non deve necessariamente trasferire le frame al ricevente mantenendo lo stesso ordine
- Tutte le funzioni elencate sono delegate ai livelli superiori

Applicazioni del PPP

• Point-to-point applications:

- Telephone Modem Links (30-54 kbit/s)
- Packet over SDH (600 Mbit/s to 10 Gbit/s)

• Shared links:

- Supporto di funzioni di autenticazione
- PPP over Ethernet (RFC 2516)

• xDSL

Formato dei pacchetti dati PPP

- **Flag**: ogni frame inizia e termina con un byte con valore 01111110
- **Address**: unico valore (11111111)
- **Control**: unico valore; ulteriori valori potrebbero essere stabiliti in futuro
- **Protocol**: indica al PPP del ricevente qual è il protocollo del livello superiore cui appartengono i dati incapsulati
- **Information**: incapsula la PDU (es. pacchetto IP) trasmesso da un protocollo del livello superiore sul collegamento PPP
- **Checksum**: utilizzato per rilevare gli errori nei bit contenuti in un pacchetto; utilizza un codice a ridondanza ciclica a due o a quattro byte

1 o 2 byte variabile 2 or 4 byte

Flag 01111110	Address 11111111	Control 00000011	Protocol	Information	FCS	Flag 01111110
------------------	---------------------	---------------------	----------	-------------	-----	------------------

Delimitazione (Byte stuffing)

• Requisito di trasparenza:

- Nel campo informazioni deve essere possibile inserire una stringa <01111110>

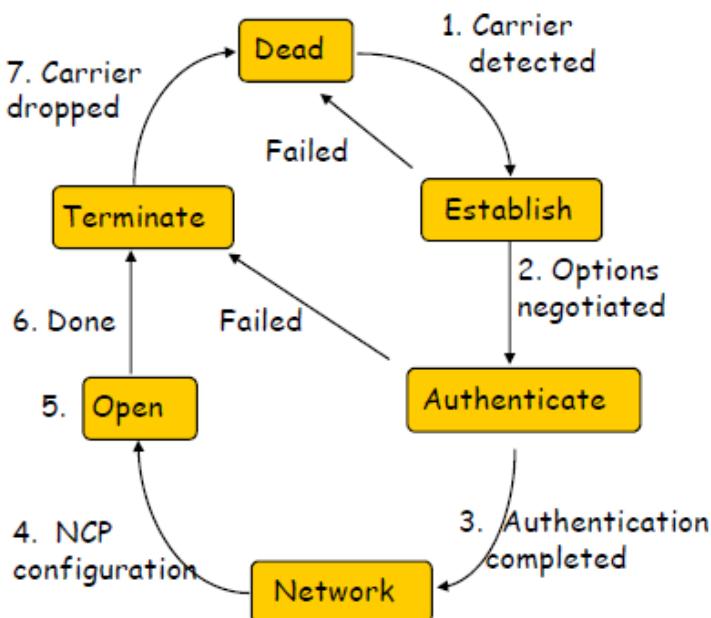
• Transmitter:

- Si aggiunge un byte <01111101> prima di ogni byte di dati <01111110> o <01111101>

• Receiver:

- Se si rivelano due byte <01111101> consecutivi si scarta il primo e continua la ricezione dei dati
- Se si rivelano una sequenza <01111101> <01111110> si scarta il primo byte e continua la ricezione dei dati
- Se si rivelava un singolo byte <01111110> si tratta di un flag

Collegamento PC-ISP: fasi del PPP



1. Il PC si connette all'ISP via modem
2. Il PC and l'ISP scambiano pacchetti LCP per negoziare i parametri del protocollo PPP
3. Controllo delle identità
4. Scambio di pacchetti NCP per configurare lo strato di rete (es. IP address assignment)
5. Emissione e ricezione di pacchetti IP send/receive IP packets
6. Il protocollo NCP è usato per abbattere lo strato di rete (rilascio degli IP address); Il protocollo LCP abbatte la connessione di data link layer
7. Il Modem si disconnette

PPP Authentication

- **Password Authentication Protocol:**

- La parte "Initiator" deve inviare la coppia [userID & password]
- La parte "Authenticator" replica indicando il successo o il fallimento dell'autenticazione
- Dopo alcuni tentativi falliti, il collegamento viene chiuso
- Se la trasmissione non è cifrata, la coppia [userID & password] può essere intercettata

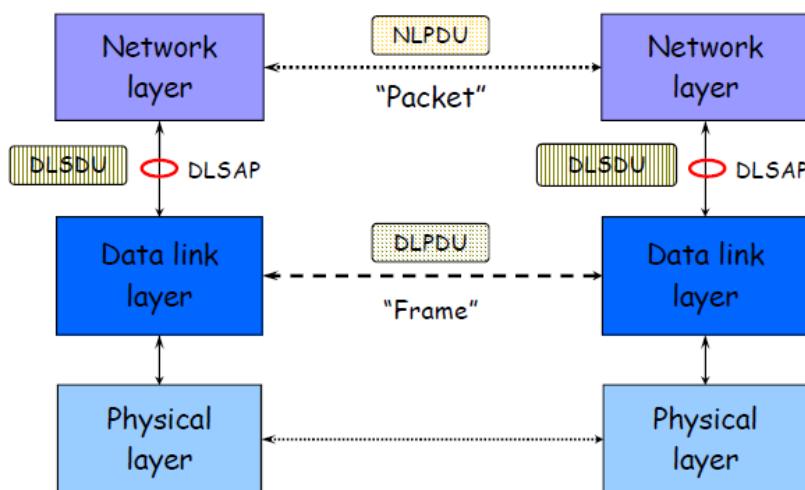
- **Challenge-Handshake Authentication Protocol (CHAP):**

- **Initiator & authenticator** condividono una chiave segreta
- L'Authenticator emette una "sfida" (un numero random)
- L'Initiator e L'Authenticator calcolano la versione cifrata della "sfida" utilizzando la chiave segreta condivisa
- L'Initiator trasmette la versione cifrata della sfida verso l'Authenticator
- L'Authenticator confronta la risposta dell'Initiator con la propria versione della sfida cifrata

Il protocollo HDLC

High-Level Data Link Control (HDLC)

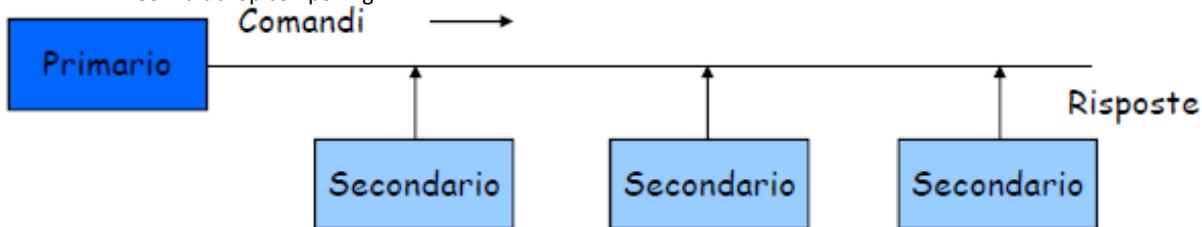
- Bit-oriented data link control
- Derivato dal protocollo Synchronous Data Link Control (SDLC) della IBM
- È la base dei protocolli della famiglia Link Access Procedure Balanced (LAPB):
 - LAPD in ISDN
 - LAPDm nel GSM



Modalità di trasferimento

- **Normal Response Mode:**

- Linee multidrop con polling



- **Asynchronous Balanced Mode:**

- Link full-duplex, point-to-point



HDLC Frame Format

Flag	Address	Control	Information	FCS	Flag
------	---------	---------	-------------	-----	------

- **Flag:** funzioni di delimitazione (01111110)

- **Address:** identifica la stazione secondaria (1 otetto):

- Nella modalità ABM, una station può agire come primario o secondario quindi il valore del campo può cambiare

- **Information:** dati d'utente (lunghezza variabile)

- **Frame Check Sequence:** CRC 16 o 32-bit

Campo di controllo

- Information Frame:

1	2-4	5	6-8
0	N(S)	P/F	N(R)

- Supervisory Frame:

1	0	S	S	P/F	N(R)
---	---	---	---	-----	------

- Unnumbered Frame:

1	1	M	M	P/F	M	M	M
---	---	---	---	-----	---	---	---

- S: Supervisory Function Bits
- N(R): Receive Sequence Number
- N(S): Send Sequence Number
- M: Unnumbered Function Bit
- P/F: Poll/final bit used in interaction between primary and secondary

Information frame (I-frame)

- Ogni I-frame contiene un numero di sequenza N(S)
- Positive ACK piggybacked:
 - N(R) = Numero di sequenza della prossima frame che il receiver si aspetta di ricevere
 - Riscontra tutte le frame fino a quella numerata con N(R)-1
- Numero di sequenza composto da 3 (modulo 8) o 7 (modulo 128):
 - Massima dimensione della finestra in trasmissione 7 o 127
- Poll/Final Bit:
 - Il primario indica i comandi con bit P=1
 - Il Secondario impone F=1 nell'ultima I-frame in risposta

Supervisory frame

- Implementano le funzioni di error control (ACK, NAK) e flow control
- Receive Ready (RR), "SS" = "00":
 - Hanno il significato di ACK quando non è possibile il piggyback
- REJECT (REJ), "SS" = "01":
 - Hanno il significato Negative ACK
 - Indicano che la frame numerata con N(R) è la prima frame ricevuta non correttamente
 - Il Transmitter deve ritrasmettere tutte le frame a partire da quella numerata con N(R)
- Receive Not Ready (RNR), "SS" = "10":
 - Riscontra le frame fino a quella numerata con N(R)-1
 - Blocca la trasmissione delle frame successive
- Selective REJECT (SREJ), "SS" = "11":
 - Richiede che sia ritrasmessa solo la frame numerata con N(R)

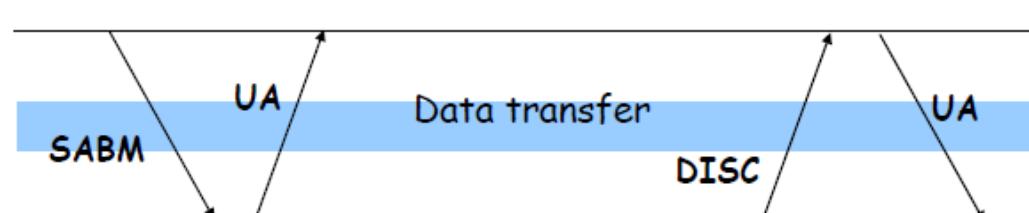
Unnumbered Frame

- Modalità del protocollo:
 - SABM: Set Asynchronous Balanced Mode
 - UA: indicano l'accettazione della modalità di trasmissione
 - DISC: termina la connessione di strato di link
- Information Transfer tra stazioni:
 - UI: Unnumbered information
- Funzioni di Recovery:
 - FRMR: frame con FCS corretto, ma non comprensibile
 - RSET: indicano il reset del collegamento

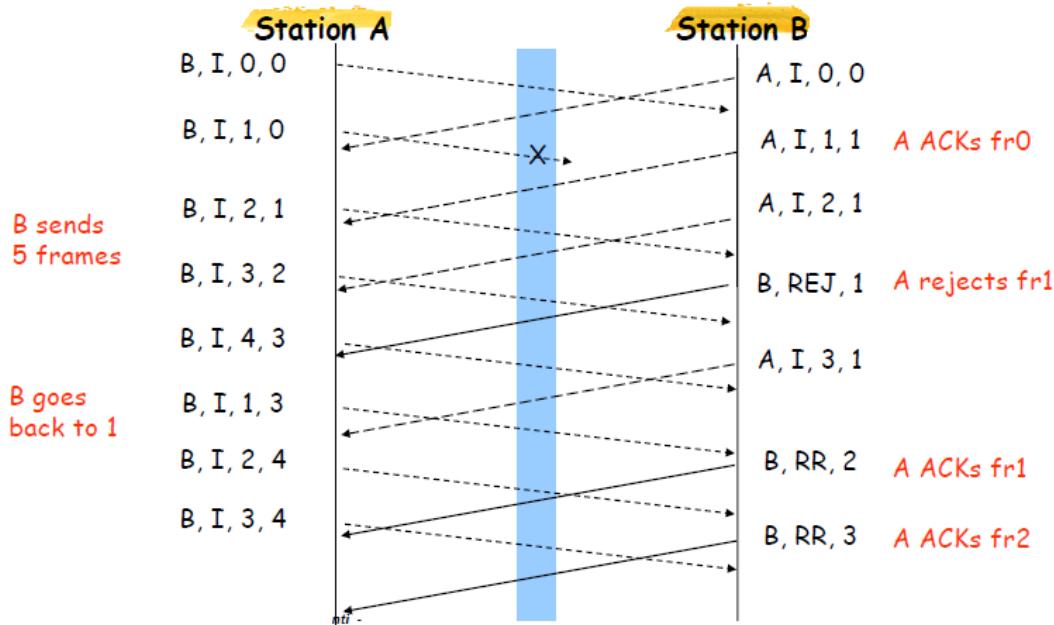
Connection Establishment & Release

- Le Supervisory frame sono usate per stabilire e rilasciare la connessione di link

- In HDLC:
 - Set Asynchronous Balanced Mode (SABM)
 - Disconnect (DISC)
 - Unnumbered Acknowledgment (UA)

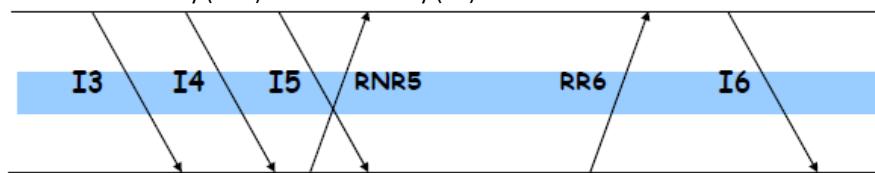


Frame Exchange using Asynchronous Balanced Mode



Flow Control

- Il controllo di flusso è richiesto per evitare la perdita di PDU in caso di overflow del buffer
- Il ricevitore può controllare il flusso ritardando l'emissione dei riscontri
- Il ricevitore può usare le supervisory frame per controllare esplicitamente il transmitter:
 - Receive Not Ready (RNR) & Receive Ready (RR)

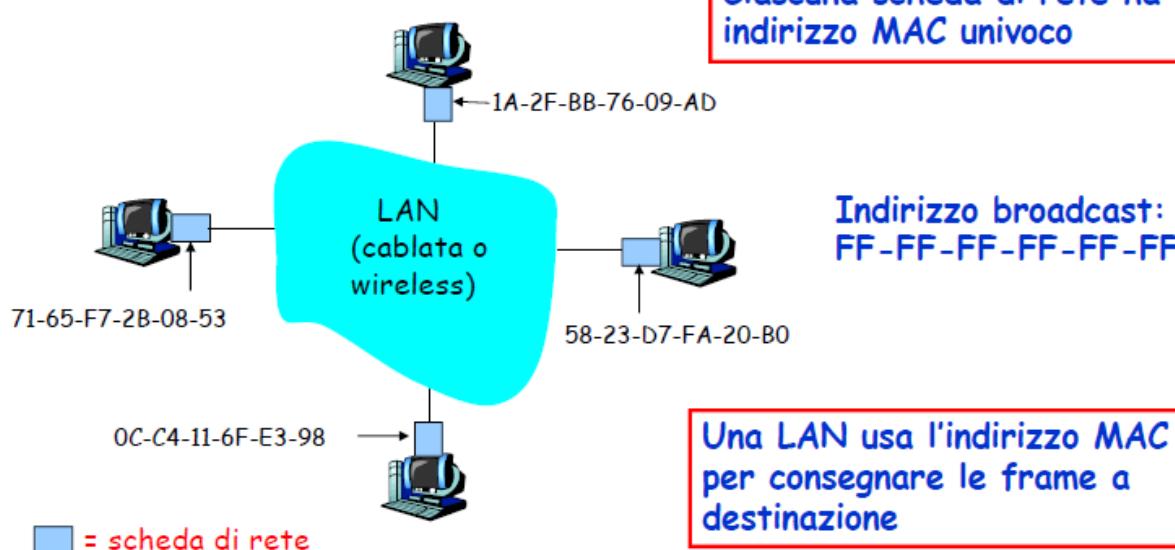


"Addressing", "Ethernet" "Hub e Switch"

Indirizzi MAC

- Indirizzo IP a 32 bit (Strato di rete):
 - Indirizzo a livello di rete
 - Analogico all'indirizzo postale di una persona
 - Ha una struttura gerarchica e deve esser aggiornato quando una persona cambia residenza (cambia rete)
- Indirizzo MAC (strato di data link):
 - Analogico al numero di codice fiscale di una persona
 - Ha una struttura orizzontale e non varia a seconda del luogo in cui la persona si trasferisce (indipendente dalla rete)
 - Indirizzo a **48 bit** (per la maggior parte delle LAN)

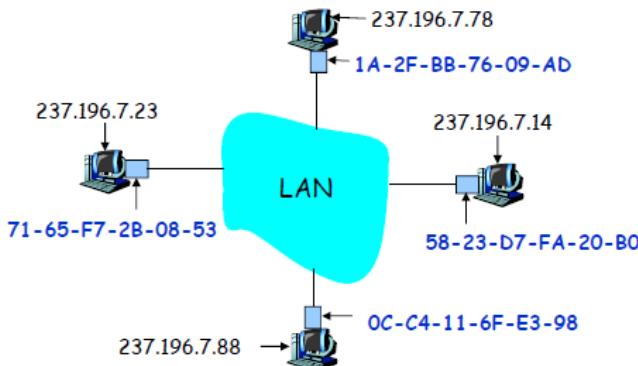
Notazione esadecimale



- La IEEE sovrintende alla gestione degli indirizzi MAC
- Quando una società vuole costruire schede di rete, compra un blocco di spazio di indirizzi (**unicità degli indirizzi**)
- Indirizzo orizzontale MAC --> **portabilità**:
 - È possibile spostare una scheda LAN da una LAN a un'altra
- Gli indirizzi IP hanno una struttura gerarchica e devono essere aggiornati se il terminale cambia rete:
 - Dipendono dalla sottorete IP cui il nodo è collegato.

Address Resolution Protocol (ARP)

Come si determina l'indirizzo MAC di un nodo se si conosce solo l'indirizzo IP del nodo?



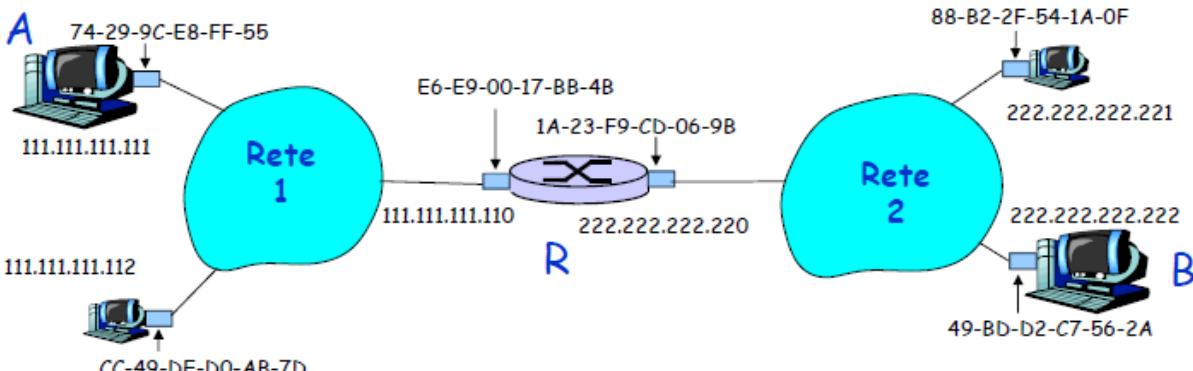
- Ogni nodo IP (host, router) nella LAN ha una **tavella ARP**
- **Tabella ARP:**
 - Contiene la corrispondenza tra indirizzi IP e MAC
- <**Indirizzo IP; Indirizzo MAC; TTL**>:
 - TTL (tempo di vita) valore che indica quando bisognerà eliminare una data voce nella tabella, il tempo di vita tipico è di 20 min

Protocollo ARP nella stessa sottorete

- Un host A vuole inviare un messaggio ad un host B:
 - L'indirizzo MAC di B non è nella tabella ARP di A
- A trasmette in una frame broadcast il **messaggio di richiesta ARP**, contenente l'indirizzo IP di B:
 - Indirizzo MAC del destinatario
 - FF-FF-FF-FF-FF-FF, Tutti gli host della LAN ricevono la richiesta ARP
- L'host B riceve la frame ARP e risponde ad A comunicandogli il proprio indirizzo MAC:
 - Il frame viene inviato all'indirizzo MAC di A che è scritto nel messaggio ARP
- Il messaggio di risposta ARP è inviato in una frame standard
- ARP è "plug-and-play":
 - La tabella ARP di un nodo si costituisce automaticamente e non deve essere configurata dall'amministratore del sistema

Invio verso un nodo esterno alla sottorete

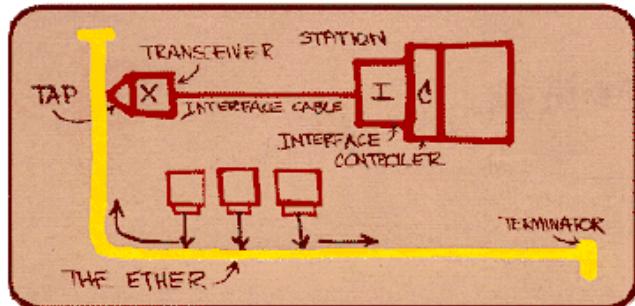
- Invio di un pacchetto tra due host A a B, localizzati in due LAN diverse (**Reti 1 e 2**) attraverso un router R
- È necessario che A conosca l'indirizzo IP di B
- Il router R ha due tabelle ARP, una per ciascuna LAN



- A crea un pacchetto con origine A, e destinazione B
- A usa ARP per ottenere l'indirizzo MAC di R (scheda della rete 1)
- A invia il pacchetto a R
- R rimuove il pacchetto IP dalla frame Ethernet, e vede che la destinazione è B
- R usa ARP per ottenere l'indirizzo MAC di B
- R crea un frame contenente il pacchetto IP e lo invia a B

Ethernet Standard IEEE 802.3

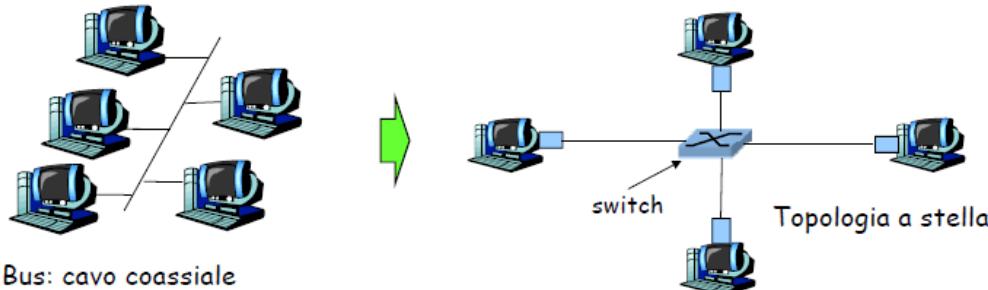
- 1970 ALOHAnet radio network deployed in Hawaiian islands
- 1973 Metcalf and Boggs invent Ethernet, random access in wired net
- 1979 DIX Ethernet II Standard
- 1985 IEEE 802.3 LAN Standard (10 Mbps)
- 1995 Fast Ethernet (100 Mbps)
- 1998 Gigabit Ethernet
- 2002 10 Gigabit Ethernet



- Il progetto originale di Bob Metcalfe che portò allo standard Ethernet

Topologia a bus o a stella

- La topologia a bus originale è stata sostituita dalla **topologia a stella** alla metà degli anni 90
- Al centro della stella è collocato un elemento denominato **switch** che esegue le funzioni di commutazione delle frame sui rami della stella
- Ciascun nodo esegue un protocollo Ethernet separato e non entra in collisione con gli altri



IEEE 802.3 MAC

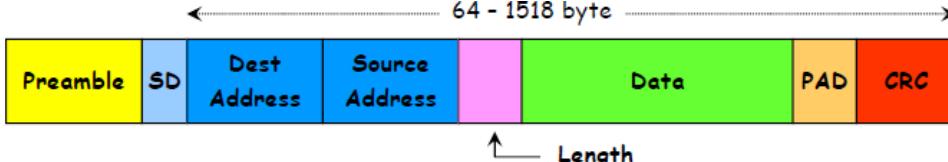
- CSMA/CD
- Parametro principale di sistema: **Slot Time**:
 - Limite superiore per rivelare una collisione ($2t_{prop}$)
 - Limite superiore per acquisire il canale in trasmissione
 - Limite superiore per la lunghezza di una frame in caso di collisione
 - Quanto per il calcolo del tempo di ritrasmissione in caso di collisione
 - $\max\{\text{round-trip propagation, MAC jam time}\}$

IEEE 802.3: Parametri originali

- Transmission Rate: 10 Mbit/s
- Lunghezza minima di una frame: 512 bit = 64 byte
- **Slot time:** 512 bit/10 Mbit/s = **51.2 μ sec**
 - $51.2 \mu\text{sec} \times 2 \times 105 \text{ km/sec} = 10.24 \text{ km}$ (round trip delay)
 - 5.12 km estensione massima della rete
- Lunghezza massima della rete: 2500 metri + 4 repeater (5 tratte di 500 metri ciascuna)
- **Regola:**
 - Ogni incremento di 10 volte del bit rate, determina la diminuzione di 10 volte della lunghezza massima della rete

Frame Ethernet (IEEE 802.3)

- La scheda di rete trasmittente incapsula i pacchetti IP in una frame Ethernet



- **Preambolo** (7 byte):
 - Ogni byte ha la configurazione 10101010 (onda quadra)
 - Serve per "attivare" le schede di rete dei riceventi e a sincronizzare i loro clock con quello del trasmittente

- **Start Delimiter** (1 byte):
 - Ha configurazione 10101011
 - Indica l'inizio della frame

- **Source e Destination Address** (6 byte ciascuno):
 - Sono gli indirizzi MAC del mittente e del destinatario della frame
 - Quando una scheda di rete riceve una frame contenente nel campo destination address il proprio indirizzo MAC o l'indirizzo broadcast (es.: un pacchetto ARP), copia la frame nel buffer di ricezione
 - Le frame con altri indirizzi MAC vengono ignorate

- **Length** (2 byte):
 - Indica il numero di byte del campo informativo
 - Lunghezza massima della frame 1518 byte (esclusi preamble e SD)
 - Lunghezza massima del campo informativo 1500 bytes

- **PAD:**
 - Assicura che la lunghezza minima di una frame sia 64 byte

- **CRC** (4 byte)

Fasi operative del protocollo CSMA/CD

- La scheda di rete prepara una frame Ethernet (802.3)
- Se il canale è inattivo, inizia la trasmissione.
- Se il canale risulta occupato, resta in attesa fino a quando non rileva più il segnale
- Verifica, durante la trasmissione, la presenza di eventuali segnali provenienti da altri terminali
- Se non ne rileva considera il pacchetto spedito
- Se rileva segnali da altri adattatori (evento di **collisione**), interrompe immediatamente la trasmissione del pacchetto e invia un segnale di **disturbo (jam)**
- La scheda di rete calcola l'intervallo di backoff
- Se si è arrivati all' n -esima collisione consecutiva, stabilisce un valore K tra {0,1,2,...,2 n -1}
- La scheda di rete aspetta un tempo pari a K volte 512 bit (slot size)

Protocollo CSMA/CD di Ethernet

- **Segnale di disturbo (jam):**
 - La finalità è di avvisare della collisione tutti gli altri adattatori che sono in fase trasmittiva
 - Ha lunghezza 48 bit

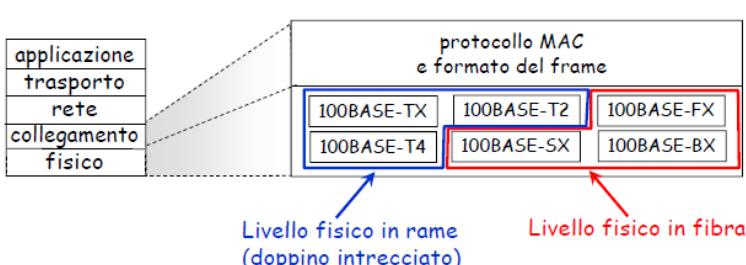
- **Intervallo di bit:**
 - Corrisponde a 0,1 μ s per Ethernet a 10 Mbps
 - Per K=1023, il tempo di attesa complessivo è di circa 50 ms:

$$- 1023 \times 512 \times 0.1 \mu\text{s} \cong 50 \text{ ms}$$

- **Intervallo di backoff:**
 - Ha lo scopo di adattare il tempo di attesa al numero di nodi coinvolti nella collisione
 - Prima collisione: sceglie K tra {0,1}; il tempo di attesa è pari a K volte 512 bit.
 - Dopo la seconda collisione: sceglie K tra {0,1,2,3}...
 - Dopo dieci collisioni, sceglie K tra {0,1,2,3,4,...,1023}

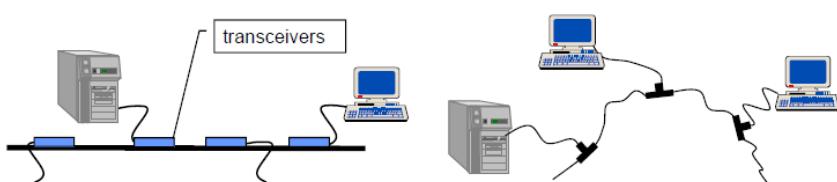
Ethernet 802.3: livelli di collegamento e fisico

- **Standard Ethernet:**
 - Protocollo MAC e formato della frame unici
 - Differenti velocità: 10 Mbit/s, 100 Mbit/s, 1 Gbit/s, 10 Gbit/s
 - Differenti mezzi trasmissivi: fibra, cavo



IEEE 802.3 Physical Layer (10 Mbit/s)

	10base5	10base2	10baseT	10baseFX
Medium	Thick coax	Thin coax	Twisted pair	Optical fiber
Max. Segment Length	500 m	200 m	100 m	2 km
Topology	Bus	Bus	Star	Point-to-point link



Fast Ethernet (100 Mbit/s)

	100baseT4	100baseT	100baseFX
Medium	Twisted pair category 3 UTP 4 pairs	Twisted pair category 5 UTP two pairs	Optical fiber multimode Two strands
Max. Segment Length	100 m	100 m	2 km
Topology	Star	Star	Star

- Identico formato di frame rispetto alla versione a 10 Mbit/s
- La topologia a bus non è prevista
- Standard attualmente prevalente

Gigabit Ethernet (1 Gbit/s)

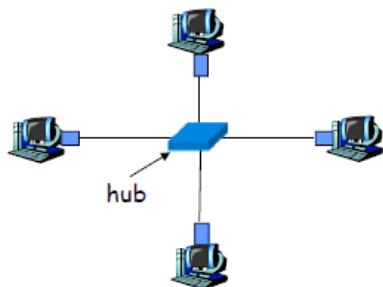
	1000baseSX	1000baseLX	1000baseCX	1000baseT
Medium	Optical fiber multimode Two strands	Optical fiber single mode Two strands	Shielded copper cable	Twisted pair category 5 UTP
Max. Segment Length	550 m	5 km	25 m	100 m
Topology	Star	Star	Star	Star

- Il time slot è incrementato a 512 byte
- Le frame di lunghezza minima sono estese a 512 byte
- Adozione del “Frame bursting” in modo che le stazioni possano trasmettere un insieme di frame di lunghezza breve
- Il CSMA-CD è sostanzialmente abbandonato
- Questo tipo di reti è largamente adottato nel backbone di reti aziendali

Hub e switch

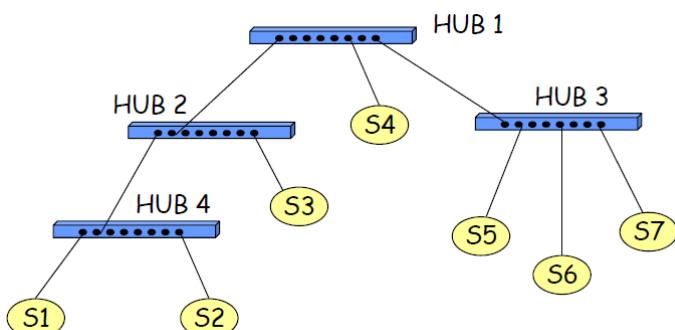
Hub (repeater)

- Opera allo strato fisico
- Rigenera il segnale analogico (re-shaping, re-timing re-transmitting) e lo ritrasmette su tutte le interfacce uscenti:
 - Decodifica e ri-codifica il codice di linea (Manchester)
 - Rileva collisioni e le inoltra su tutte le porte
 - Isola segmenti di rete se si verificano 30 collisioni consecutive
- Permette di aumentare le dimensioni di una LAN rispettando:
 - a) Limite teorico imposto dal CSMA/CD
 - b) Limiti al numero massimo di ripetitori utilizzabili



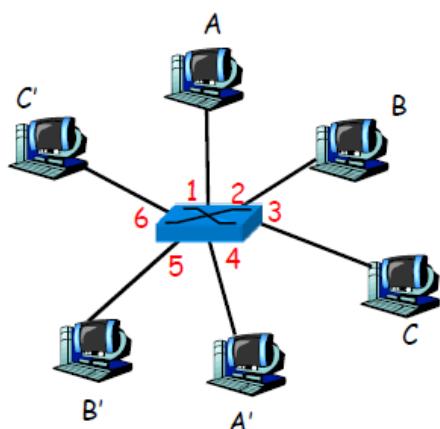
- **Dominio di collisione:**
 - Sezione di rete in cui qualsiasi coppia di stazioni che trasmettono contemporaneamente generano una collisione
- La sezione di rete collegata da hub (repeater) fa parte di un unico dominio di collisione

Esempio di dominio di collisione



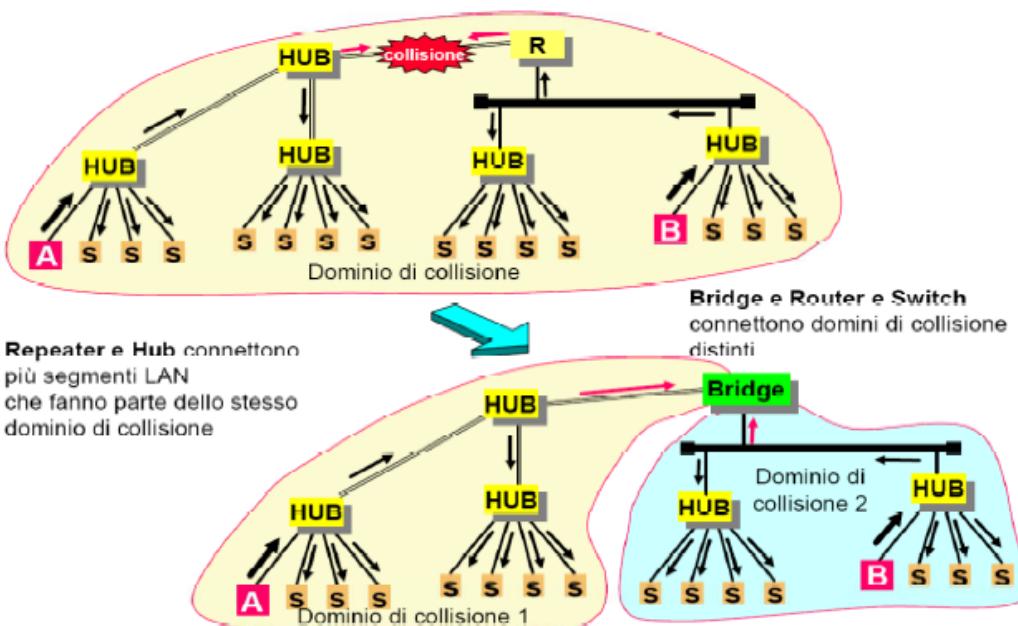
Switch (Bridge)

- **Dispositivo intelligente a livello di link, svolge un ruolo attivo:**
 - Filtra e inoltra le frame Ethernet
 - Esamina l'indirizzo MAC di destinazione e, se possibile, lo invia all'interfaccia corrispondente alla sua destinazione
 - Quando un pacchetto è stato inoltrato nel segmento, usa il protocollo CSMA/CD per accedere al segmento
- **Permette di collegare tra loro differenti domini di collisione**
- **Trasparente:**
 - Gli host sono inconsapevoli della presenza di switch
- **Plug-and-play, autoapprendimento:**
 - Gli switch non hanno bisogno di essere configurati, apprendono autonomamente la topologia di rete e le regole di instradamento delle frame
- Gli host hanno (normalmente) collegamenti dedicati e diretti con lo switch
- Gli switch, se necessario, bufferizzano le frame
- Il CSMA/CD è usato su ciascun collegamento in entrata, anche se non si verificano collisioni:
 - Collegamenti full duplex
- Trasmissione simultanea da A ad A' e da B a B', senza collisioni:
 - La trasmissione simultanea non è switch con sei interfacce possibile con gli hub



*switch con sei interfacce
(1,2,3,4,5,6)*

Domini di collisione

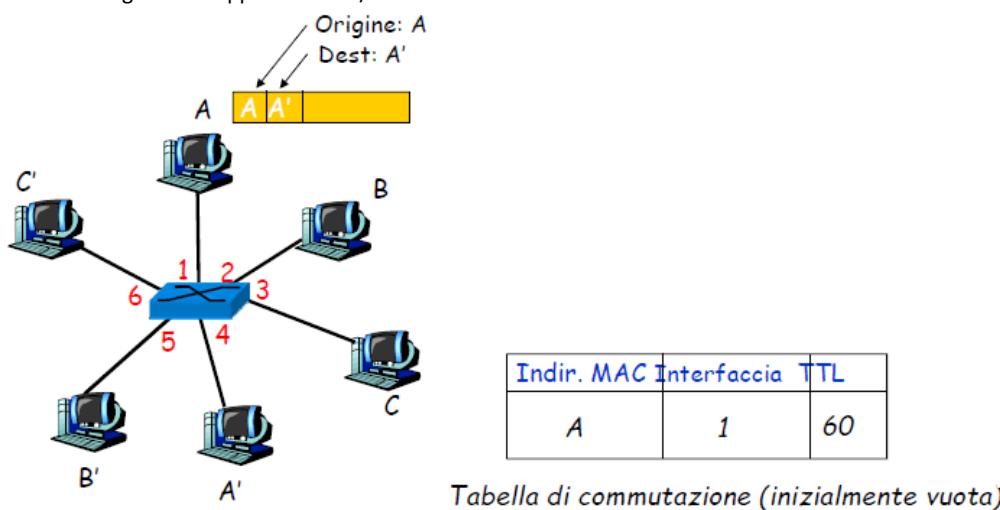


Switch Table

- Ogni switch ha una tabella di commutazione (**switch table**)
- Ogni record della switch table comprende:
 - L'indirizzo MAC di un nodo
 - L'interfaccia a cui è connesso il nodo
 - Time stamp
- Come si creano e si mantengono i record di una tabella di commutazione?
 - Auto apprendimento

Switch: autoapprendimento

- Lo switch **apprende** quali nodi possono essere raggiunti attraverso determinate interfacce:
 - Quando riceve una frame, lo switch "impara" l'indirizzo MAC del mittente
 - Registra la coppia mittente/indirizzo nella sua tabella di commutazione



Switch: filtraggio e inoltro

- Quando uno switch riceve un pacchetto:
 1. Registra l'interfaccia associata all'host mittente
 2. Accede alla tabella utilizzando gli indirizzi MAC
- if entry trovato tramite l'indirizzo MAC di destinazione (dest)
- ```

 then{
 if dest risiede sull'interfaccia su cui è arrivata la frame
 then scarta la frame
 else rilancia la frame sull'interfaccia indicata
 }
 else flood ← Lo inoltra su tutte le interfacce tranne quella dalla quale è arrivata la frame

```

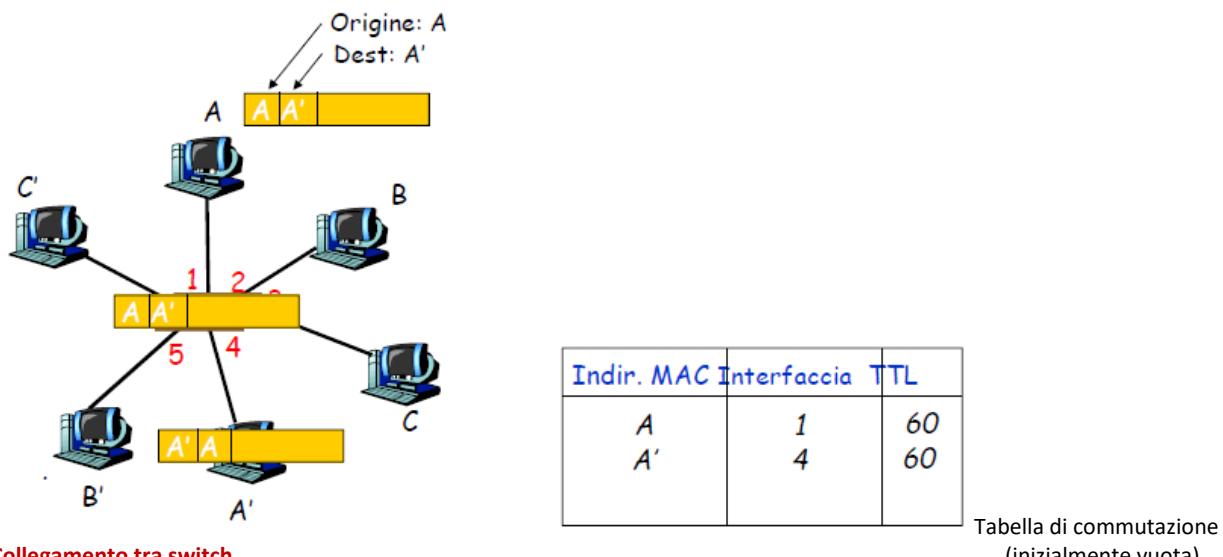
### Esempio

- Destinazione del frame ignota:

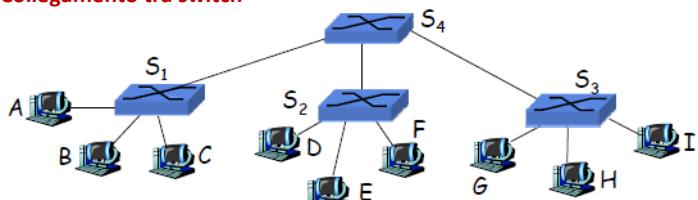
flood

- Destinazione A, location nota:

selective send

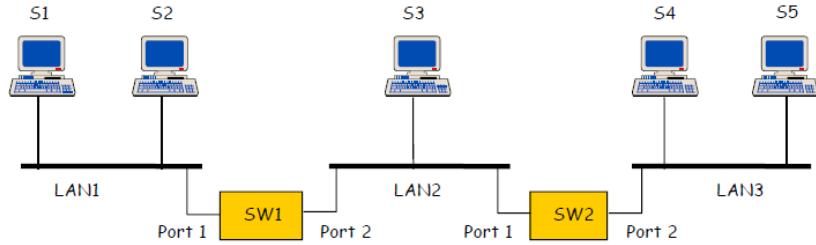


### Collegamento tra switch



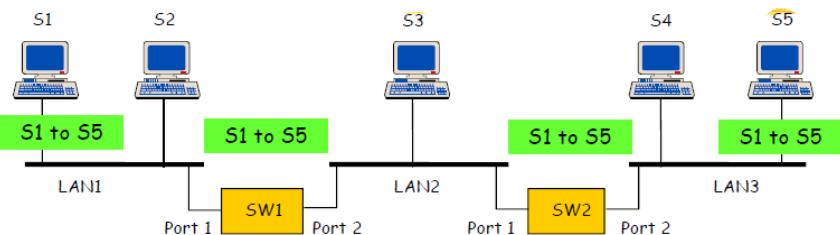
- Gli switch possono essere interconnessi tra loro
- Per inviare una frame da A a G, come fa  $S_1$  a sapere che deve inoltrare il frame attraverso  $S_4$  e  $S_3$ ?
- Autoapprende (funziona esattamente come nel caso di un singolo switch)

### Esempio



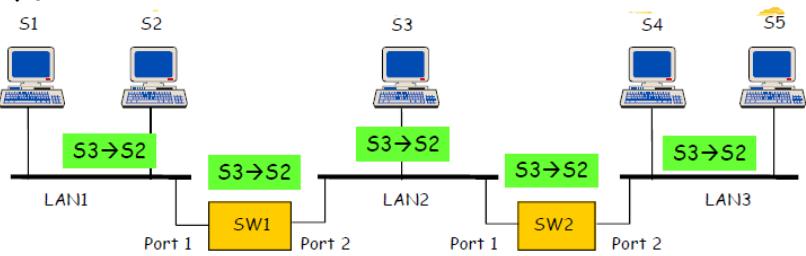
| Address | Port |
|---------|------|
|         |      |
|         |      |
|         |      |
|         |      |
|         |      |
|         |      |
|         |      |

S1 → S5



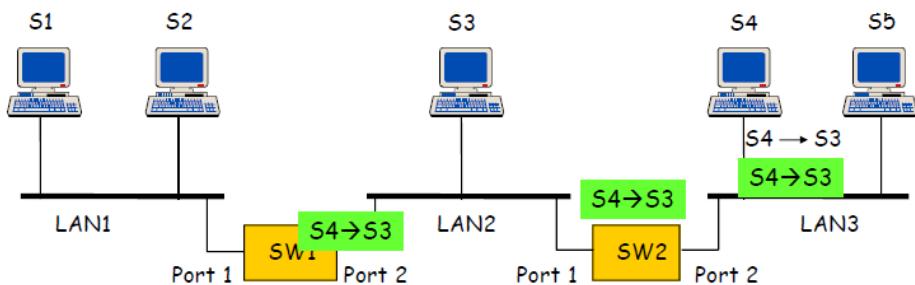
| Address | Port |
|---------|------|
|         |      |
| S1      | 1    |
|         |      |
|         |      |
|         |      |
|         |      |
|         |      |
|         |      |

S3 → S2



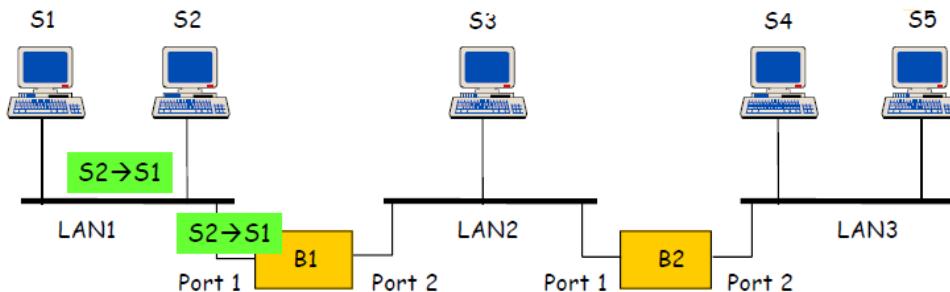
| Address | Port |
|---------|------|
|         |      |
| S1      | 1    |
| S3      | 2    |
|         |      |
|         |      |
|         |      |
|         |      |
|         |      |

S4 → S3



| Address | Port |
|---------|------|
|         |      |
| S1      | 1    |
| S3      | 2    |
| S4      | 2    |
|         |      |
|         |      |
|         |      |
|         |      |
|         |      |

S2 → S1



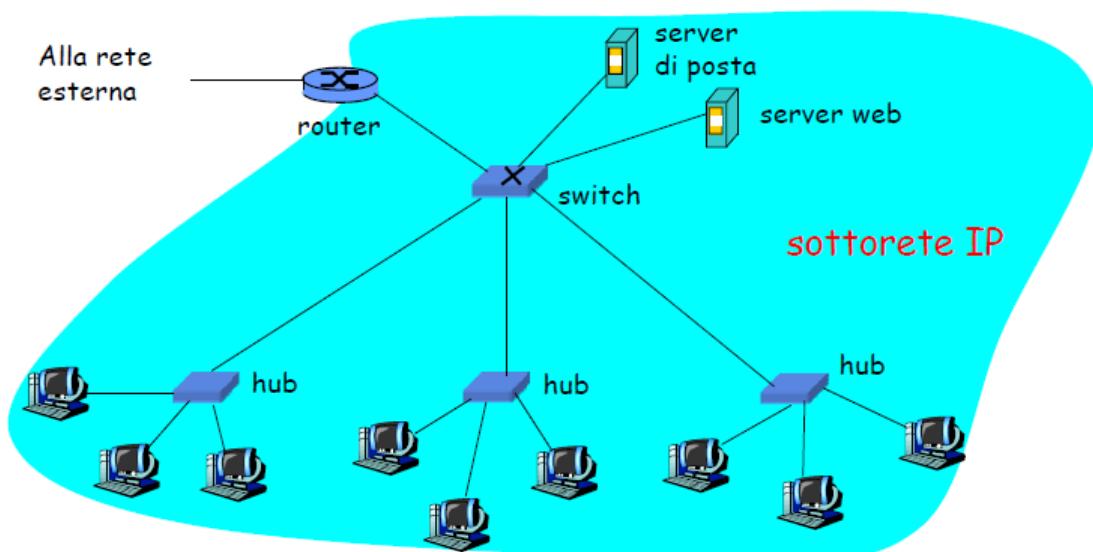
| Address | Port |
|---------|------|
| S1      | 1    |
| S3      | 2    |
| S4      | 2    |
| S2      | 1    |

| Address | Port |
|---------|------|
| S1      | 1    |
| S3      | 1    |
| S4      | 2    |

### Adaptive Learning

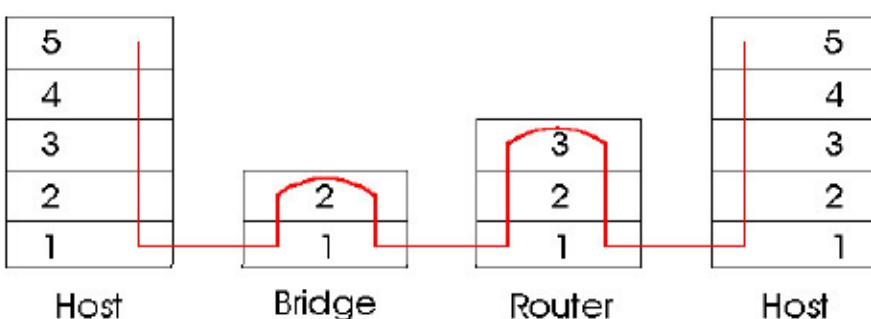
- In una rete statica il processo di apprendimento conduce ad uno stato in cui tutti gli indirizzi sono memorizzati nelle switch table
- In situazioni pratiche, in una rete i nodi sono aggiunti, rimossi o spostati:
  - Si introduce un timeout che forza periodicamente la ripetizione dell'apprendimento di ogni indirizzo
  - Le informazioni che non vengono rinfrescate sono cancellate dopo un tempo massimo (ageing time – 300 s valore consigliato dallo standard)
  - Se una frame arriva su una porta che differisce da quella memorizzata nella switch table, questa viene aggiornata immediatamente

### Esempio di rete di un'istituzione



### Switch e router a confronto

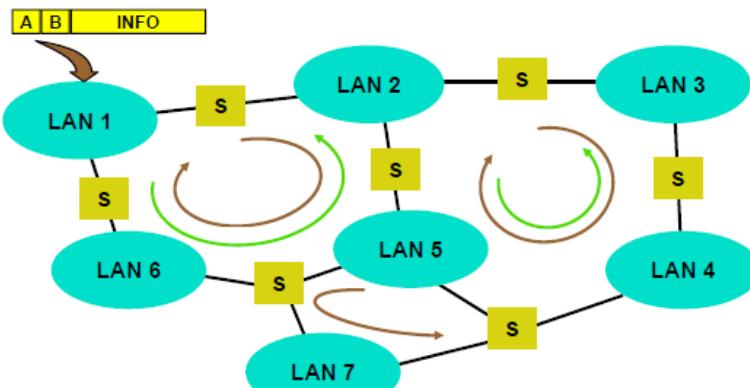
- Entrambi sono dispositivi store-and-forward:
  - Router: dispositivi a livello di rete**
  - Switch: dispositivi a livello di collegamento**
- I router mantengono tabelle di routing e implementano algoritmi di instradamento
- Gli switch mantengono tabelle di commutazione e implementano il filtraggio e algoritmi di autoapprendimento



## Spanning Tree Protocol (STP)

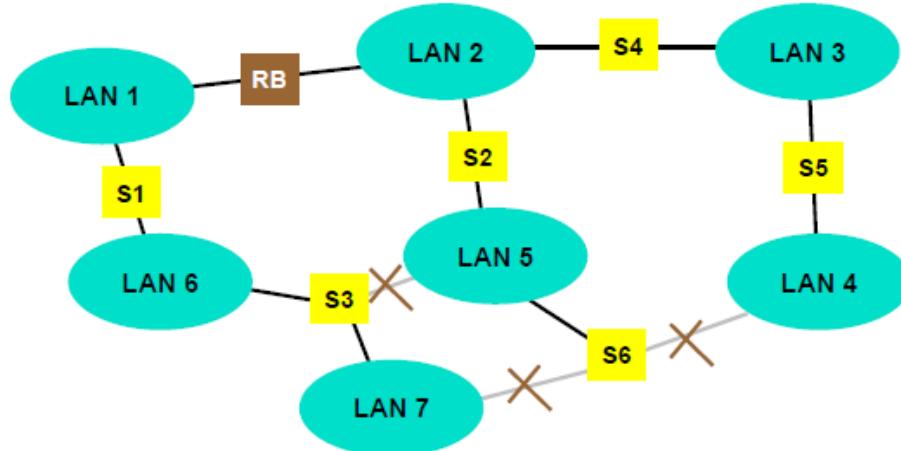
### Interconnessione di LAN tramite switch

- Problema dei "cicli infiniti"



- La rete magliata deve essere trasformata in albero:

- Protocollo Spanning Tree (IEEE 802.1D) regola il processo di forwarding in presenza di loop nella rete

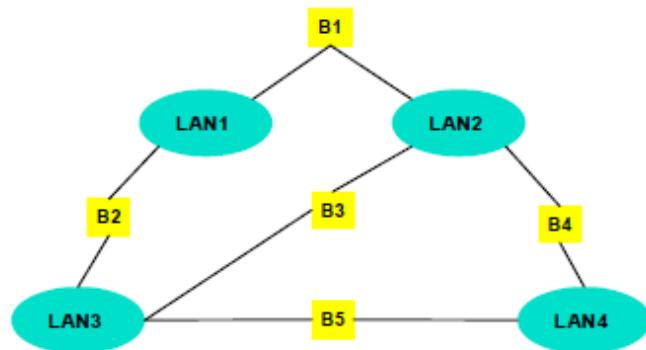


- Per ottenere un albero si deve:

- Determinare lo switch radice (**Root Bridge – RB**)
- Disabilitare alcune porte degli switch

- Quali salvare?

- Verso la radice ...
- Verso le LAN ...



- Opera in tre fasi:

1. Elezione del Root Bridge: radice dell'albero
2. Selezione della **root port**: per ogni switch la porta per raggiungere il Root Bridge
3. Selezione della **designated port**: per ogni LAN la porta utilizzata per inoltrare e ricevere le trame della LAN

- Alla fine vengono abilitate al forwarding solo le porte root e designated (le altre vengono bloccate)

- Utilizza trame denominate **BPDU Bridge Protocol Data Unit** trasmesse in multicast:

- Topology Change Notification BPDU

- Configuration BPDU inviate periodicamente contengono:

- **Root id**: l'identificativo del bridge candidato a diventare il Root Bridge
- **Switch id**: identificativo del bridge che trasmette la BPDU
- **Root path cost**: costo totale del percorso per raggiungere il Root Bridge (posto a 0 dal Root Bridge e aggiornato da ogni altro switch)
- **Flag**:Topology Change (TC), TC Acknowledgment (TCA)

- **Topology Change Notification BPDU** trasmesso solo a seguito di un cambiamento nella topologia verso il Root Bridge

## Configuration BPDU

| Octet                       |    |
|-----------------------------|----|
| Protocol Identifier         | 1  |
| Protocol Version Identifier | 2  |
| BPDU Type                   | 3  |
| Flags                       | 4  |
| Root Identifier             | 5  |
|                             | 6  |
|                             | 7  |
|                             | 8  |
|                             | 9  |
|                             | 10 |
|                             | 11 |
|                             | 12 |
|                             | 13 |
|                             | 14 |
|                             | 15 |
|                             | 16 |
|                             | 17 |
| Root Path Cost              | 18 |
|                             | 19 |
|                             | 20 |
|                             | 21 |
|                             | 22 |
|                             | 23 |
|                             | 24 |
|                             | 25 |
| Bridge Identifier           | 26 |
| Port Identifier             | 27 |
| Message Age                 | 28 |
| Max Age                     | 29 |
| Hello Time                  | 30 |
| Forward Delay               | 31 |
|                             | 32 |
|                             | 33 |
|                             | 34 |
|                             | 35 |

### • Elezione del Root Bridge:

- Tutti gli switch si "credono" Root e trasmettono Configuration BPDU con Root id = Switch id
- Alla ricezione di una Configuration BPDU trasmessa dallo switch j lo switch i verifica:
  - se Switch id<sub>i</sub> > Root id<sub>j</sub> interrompe la tx delle Conf. BPDU e ritrasmette solo le ricevute
  - se Switch id<sub>i</sub> < Root id<sub>j</sub> continua a tx Conf BPDU

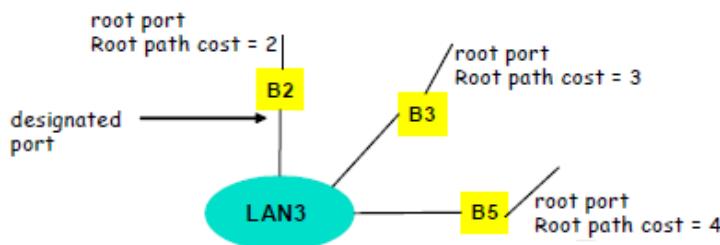
• Alla fine solo lo switch con l'identificativo più piccolo genera Configuration BPDU → Root Bridge

### • Selezione della root port:

- È la porta attraverso la quale ogni switch raggiunge il Root Bridge (riceve le Conf. BPDU)
- Ogni switch somma al campo Root path cost delle Conf. BPDU il costo del percorso (della LAN) associato alla porta di ricezione
- Ogni switch seleziona la **root port come la porta a costo minimo per raggiungere il Root Bridge**

### • Selezione della designated port:

- Ogni switch ritrasmette le Conf. BPDU ricevute dalla root port su tutte le altre porte
- Se esistono più switch sulla stessa LAN questi switch riceveranno Conf. BPDU da porte non root
- In ogni LAN lo switch con Root path cost minore è scelto come designated switch e la sua porta verso tale LAN è la designated port



## STP: cambiamenti di topologia

- Cambiamenti della topologia vengono notificati al Root Bridge attraverso Topology Change Notification BPDU
- Il Root Bridge invia Conf. BPDU con flag TC = 1 verso tutti gli altri bridge
- I bridge reagiscono al cambiamento della topologia impostando il timer ageing-time al valore forward delay (trasportato nelle Conf. BPDU ... raccomandato 15 s)

| Octet                       |   |
|-----------------------------|---|
| Protocol Identifier         | 1 |
| Protocol Version Identifier | 2 |
| BPDU Type                   | 3 |
|                             | 4 |

## Topology Change Notification BPDU

### Posizione dello Switch?

- Accessi dedicati:
  - Utilizzo pesante e continuativo di risorse di rete:
    - Server
    - Stazioni per video-comunicazione
    - etc.
- Accessi condivisi:
  - Stazioni che generano traffico discontinuo

Per esempio sulle STP consultare slide originale(11)

## "Wireless LAN, il protocollo IEEE 802.11"

### Caratteristiche ambiente wireless

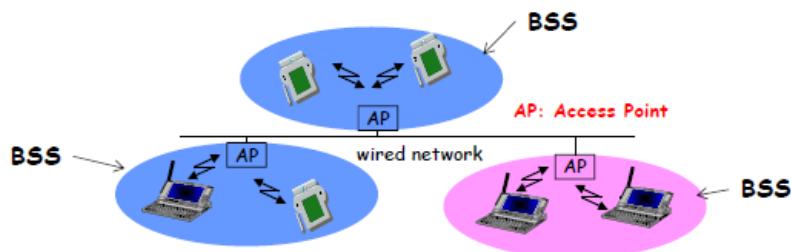
- Il Bit Error Rate (**BER**) è molto più elevato rispetto ad un ambiente "wired"
- Le operazioni di **Collision Detection** sono difficili perché una stazione non è in grado di ascoltare le proprie trasmissioni e quindi rivelare eventuali collisioni
- Problema del terminale nascosto (**Hidden Terminal**)
- Altri problemi:
  - Gestione della mobilità, variazioni della qualità del link, limitazione delle batterie, sicurezza ecc.

### IEEE 802.11 standards

| Protocol | Date | Frequency       | Date Rate (Typical) | Date Rate (Max) | Range (indoor) |
|----------|------|-----------------|---------------------|-----------------|----------------|
| 802.11a  | 1999 | 5 GHz           | 25 Mbit/s           | 54 Mbit/s       | ~30 metri      |
| 802.11b  | 1999 | 2.4 – 2.5 GHz   | 6.5 Mbit/s          | 11 Mbit/s       | ~50 metri      |
| 802.11g  | 2003 | 2.4 – 2.5 GHz   | 11 Mbit/s           | 54 Mbit/s       | ~30 metri      |
| 802.11n  | 2009 | 2.4 GHz o 5 GHz | 200 Mbit/s          | 540 Mbit/s      | ~100 metri     |
| 802.11ac | 2014 | 5 GHz           | 500 Mbit/s          | 1 Gbit/s        | ~100 metri     |

### Topologie di rete

- Infrastructure network:



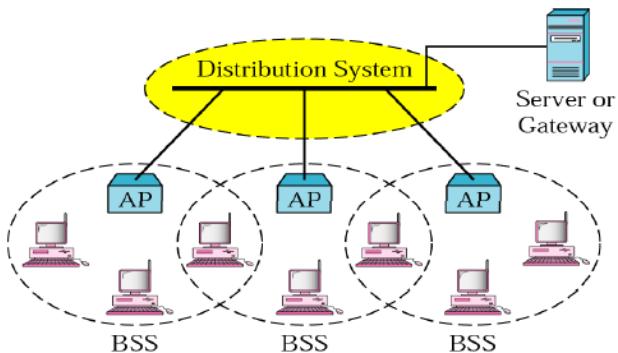
- Le comunicazioni avvengono esclusivamente tra i terminali e l'Access Point (AP) e non direttamente tra i terminali
- L'AP gestisce l'accesso al mezzo trasmittivo e si comporta da bridge verso altre reti

- Basic Service Set (BSS):

- Terminali e AP all'interno della stessa area di copertura

- Extended Service Set (ESS):

- Diverse BSS connesse tra loro



- Ad-hoc network:



- Nessun AP, ogni terminale comunica direttamente con gli altri
- I nodi comunicano tra loro se questi sono all'interno della copertura radio reciproca
- Complessità più elevata

## Strato MAC

- Sono definiti due modalità di accesso:
  - **Distributed Coordination Function (DCF)** basato su un protocollo CSMA/Collision Avoidance (**CSMA/CA**):
    - Metodo opzionale **RTS/CTS** per superare il problema dell'hidden terminal
  - **Point Coordination Function (PCF)**:
    - Metodo di polling di tipo contention-free polling adatto a servizi con requisiti stringenti di ritardo
    - L'AP interroga i terminali terminals in accordo ad una lista
- Il DCF offre un servizio di trasferimento asincrono, mentre il PCF offre sia un servizio asincrono sia un servizio time- bounded
- La configurazione ad-hoc network mode offre solo il trasferimento asincrono

### Distributed Coordination Function (DCF)

- I nodi non possono rivelare le collisioni quindi una frame sarà trasmessa sempre per intero

#### Funzionamento base CSMA/CA:

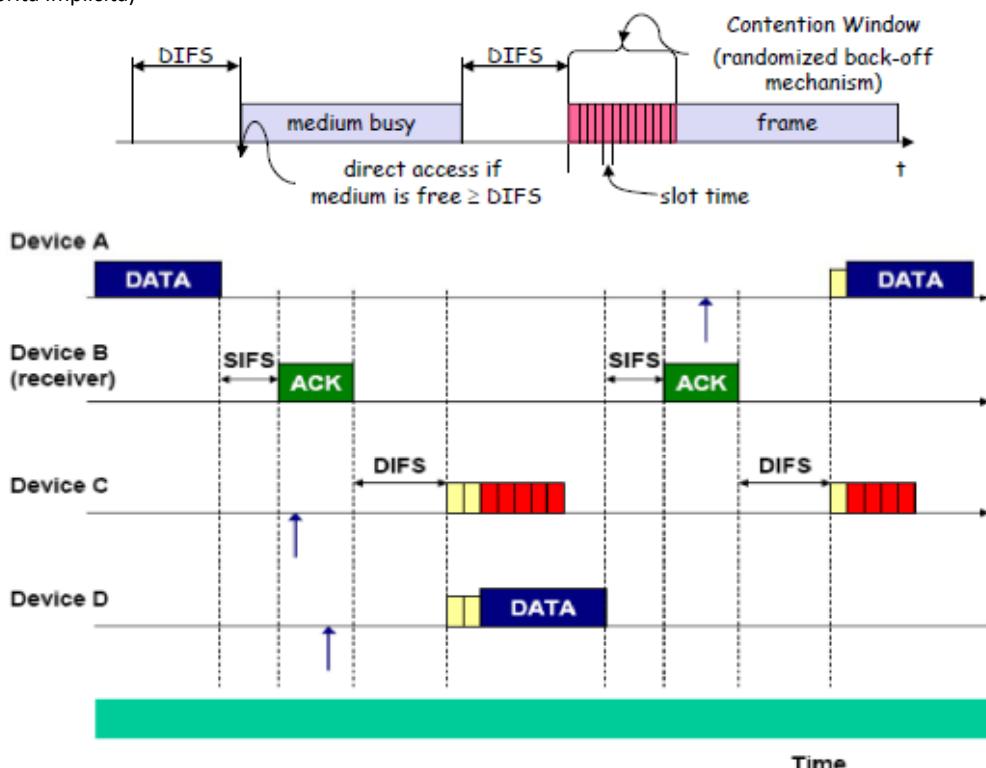
- **Se il mezzo è libero:**
  - Il terminale attende un intervallo di tempo denominato DIFS
  - Dopo l'intervallo DIFS trasmette la frame
- **Se il mezzo è occupato:**
  - Il terminale effettua il backoff della trasmissione
  - Backoff prima della collisione

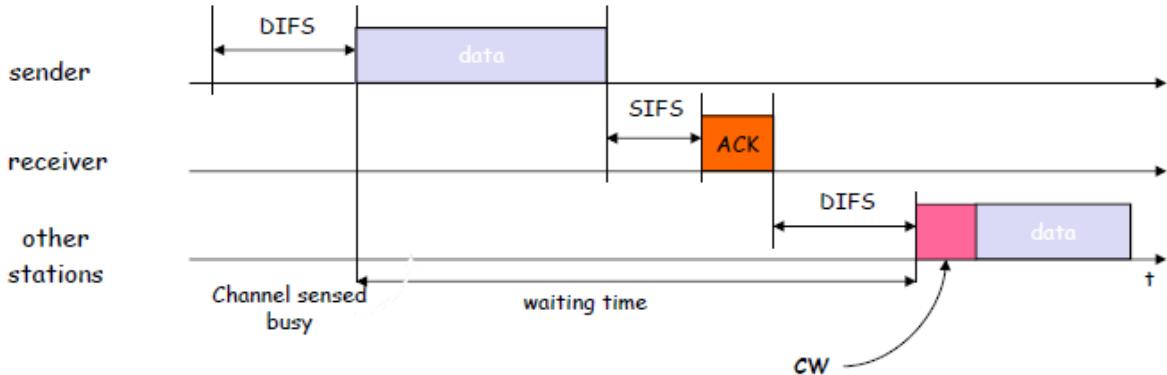
### Interframe Spacing (IFS) e priorità

- **Time slot:**
  - 9 µs (802.11g)
- **SIFS (Short IFS):**
  - 10 µs (802.11g)
  - Tempo tra frame del tipo: ACK, CTS, Poll Messages, Poll responses, CF-End
- **PIFS (PCF IFS):**
  - 19 µs (802.11g)
  - PCF operation mode, including Beacon, Retransmitted poll messages
- **DIFS (DCF IFS):**
  - 28 µs (802.11g)
  - DCF operation mode, including back-off, RTS

### Collision Avoidance

- Lo scopo è quello di ridurre la probabilità di collisione quando il mezzo ritorna libero:
  - è probabile che ci siano molti nodi in attesa di trasmettere
- Se il mezzo è rivelato libero, il nodo continua a testare il mezzo per un intervallo uguale a DIFS:
  - Si effettua la trasmissione solo se il mezzo rimane libero per l'intero intervallo DIFS
- Se il mezzo è rivelato occupato, un nodo deve attendere che per un intero intervallo DIFS il mezzo sia libero, inoltre deve attendere un back-off time (**Contention Window**), se il mezzo si è mantenuto libero per DIFS+CW, il nodo effettua la trasmissione:
  - La durata della Congestion Window varia con il carico della rete
- Se un altro nodo trasmette durante la congestion window, the back-off timer viene fermato, il timer sarà riattivato durante la fase successiva (priorità implicita)





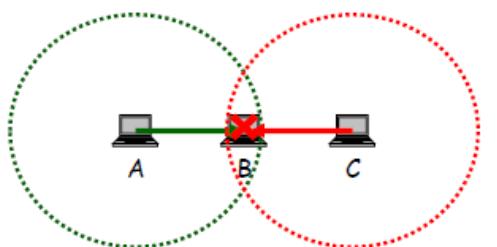
- Un nodo deve attendere un DIFS prima di emettere una frame
- Il receiver può emettere il riscontro (ACK) dopo un tempo inferiore Short Inter-Frame Space (SIFS)

### Congestion Window

- Il valore della **Contention Window (CW)** varia in funzione del carico:
  - Lo scopo è quello di ridurre al minimo le collisioni
- Nel caso di collisione, la durata della CW è progressivamente raddoppiata: 15, 31, 63,...1023, fino a che avviene con successo un trasmissione
  - Dopo una trasmissione la CW viene riportata al valore minimo (CW=15 slot)
  - 1 slot ha durata 9 µs (802.11g)
- Dato il valore della CW (= 15, 31 ... 1023 slots), un terminale calcola il random backoff come valore uniformemente distribuito nell'intervallo (0, CW)
- Lo standard 802.11 non fissa il minimo ed il massimo valore della CW:
  - Sono consigliati un valore minimo uguale a 15 slot ed un valore massimo uguale a 1023 slot

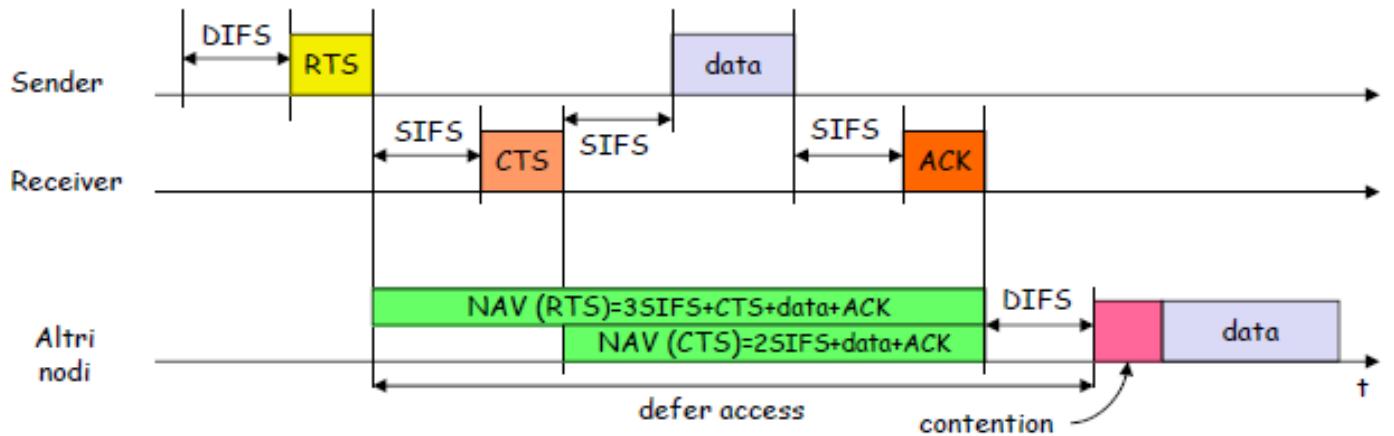
### Hidden Terminal Problem

- **Hidden terminals:**
  - I nodi A e C non possono ascoltarsi (fuori dalle rispettive aree di copertura radio)
  - A emette una frame verso B, C non può ascoltare la trasmissione di A
  - C vuole emettere una frame verso B, C "ascolta" a "free" medium (CS commette un errore)
  - Si verifica una collisione in B
  - A non può rivelare la collisione (il meccanismo CD non funziona)
- **Soluzione:**
  - Il problema dell'hidden terminal è specifico dell'ambiente wireless
  - Occorre effettuare l'operazione di Carrier Sensing al nodo "receiver" non al nodo "sender"
  - "Virtual Carrier Sensing":
    - Il Sender "chiede" al receiver se il canale è libero



### Meccanismo RTS/CTS

- Meccanismo di "prenotazione" del canale (opzionale):
  - Permette al sender di "prenotare" il canale invece di competere per il suo utilizzo attraverso un accesso casuale
  - Si evitano completamente le collisioni
- Il "sender", prima di emettere una frame, trasmette verso il receiver (o l'AP) un pacchetto, di lunghezza molto breve, denominato **"Request-To-Send (RTS)"**:
  - Il pacchetto RTS è trasmesso usando il meccanismo CSMA
  - Il pacchetto RTS può subire collisioni con altri pacchetti RTS, le collisioni sono poco probabili perché il pacchetto RTS è breve
- Il "receiver" (o l'AP) quando riceve l'RTS emette un pacchetto **"clear-to-send" (CTS)**:
  - Il pacchetto CTS è ricevuto da tutti i nodi
- Il sender quando riceve il pacchetto CTS può trasmettere la frame:
  - Gli altri terminali posticipano le proprie trasmissioni (per quanto tempo?)
- All'interno dei pacchetti RTS e CTS è indicato l'intervallo di tempo in cui il canale sarà occupato per la trasmissione della frame
- **Net Allocation Vector (NAV):**
  - È un temporizzatore che indica l'intervallo di tempo che le altre stazioni devono attendere per effettuare il test del canale e verificare se il canale libero
  - Ogni nodo alla ricezione dell'RTS inizializza il proprio NAV che specifica l'istante in cui il nodo può tentare nuovamente di accedere al mezzo
  - Virtual Carrier Sensing



### Formato delle frame

| 2 bytes          | 2 bytes | 6 bytes   | 6 bytes   | 6 bytes   | 2 bytes   | 6 bytes   | 0 to 2312 bytes | 4 bytes   |       |       |
|------------------|---------|-----------|-----------|-----------|-----------|-----------|-----------------|-----------|-------|-------|
| FC               | D       | Address 1 | Address 2 | Address 3 | SC        | Address 4 | Frame body      | FCS       |       |       |
| Protocol version | Type    | Subtype   | To DS     | From DS   | More flag | Retry     | Pwr mgt         | More data | WEP   | Rsvd  |
| 2 bits           | 2 bits  | 4 bits    | 1 bit           | 1 bit     | 1 bit | 1 bit |

- **Frame control (FC)** (2 byte):
  - Definisce il tipo di frame e contiene alcune informazioni di controllo

- **Duration (D)** (2 byte):
  - Nella maggioranza delle frame indica la durata della trasmissione
  - E' usato dagli altri nodi per definire il NAV

- **Addresses** (4 x 6 Byte):
  - Ci sono 4 campi di indirizzo MAC ognuno di lunghezza 6 byte (48 bit)
  - Il significato di questi indirizzi dipende dai flag "To DS" e "From DS" contenuti nel campo FC

- **Sequence control (SC)** (2 byte):
  - Numero di sequenza della frame
  - Usato per la funzione di flow control

- **Frame body** (0 – 2312 byte):
  - Contiene le informazioni d'utente (payload)

- **Frame Check Sequence (FCS)** (4 byte):
  - Contiene un CRC-32 per la rivelazione di errore

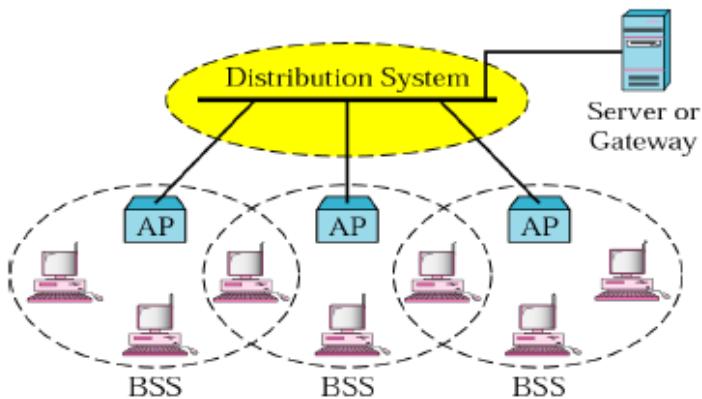
### Tipo di frame

| Field     | Explanation                                                      |
|-----------|------------------------------------------------------------------|
| Version   | Current version is 0                                             |
| Type      | Type of information: management (00), control (01), or data (10) |
| Subtype   | Subtype of each type (see Table 14.2)                            |
| To DS     | Defined later                                                    |
| From DS   | Defined later                                                    |
| More flag | When set to 1, means more fragments                              |
| Retry     | When set to 1, means retransmitted frame                         |
| Pwr mgt   | When set to 1, means station is in power management mode         |
| More data | When set to 1, means station has more data to send               |
| WEP       | Wired equivalent privacy (encryption implemented)                |
| Rsvd      | Reserved                                                         |

| Subtype | Meaning               |
|---------|-----------------------|
| 1011    | Request to send (RTS) |
| 1100    | Clear to send (CTS)   |
| 1101    | Acknowledgment (ACK)  |

## Addressing

- Il meccanismo di indirizzamento specifica quattro casi
- I quattro casi sono individuati dal valore dei due flag “To DS” e “From DS” contenuti nel campo FC
- DS (Distribution System):**
  - Infrastruttura di interconnessione tra gli AP e con il router di accesso alla rete fissa

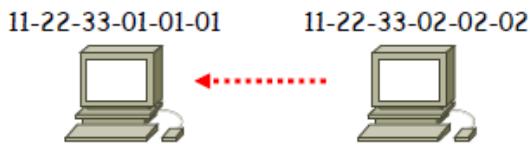


| To DS | From DS | Address 1    | Address 2  | Address 3   | Address 4 |
|-------|---------|--------------|------------|-------------|-----------|
| 0     | 0       | Destination  | Source     | BSS ID      | N/A       |
| 0     | 1       | Destination  | Sending AP | Source      | N/A       |
| 1     | 0       | Receiving AP | Source     | Destination | N/A       |
| 1     | 1       | Receiving AP | Sending AP | Destination | Source    |

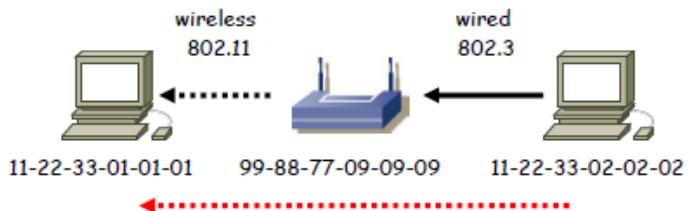
- Caso 1 (00)** : “To DS”= 0 , “From DS”= 0:
  - La frame non proviene da e non è diretta verso il Distribution System
  - I nodi sorgente e destinazione sono interni allo stesso Basic Service Set (BSS):
    - Address 1: Destination Address
    - Address 2: Source Address
    - Address 3: Identificatore del BSS a cui appartengono i nodi sorgente e destinazione (indirizzo dell'AP del BSS)
    - Address 4: non usato
- Il riscontro ACK deve essere inviato direttamente al nodo sorgente
- Caso 2 (01)** : “To DS”=0 , “From DS”=1:
  - La frame proviene dal Distribution System
  - La frame è emessa dall'Access Point (AP):
    - Address 1: Destination Address
    - Address 2: Identificatore del BSS a cui appartiene l'AP sorgente (AP address)
    - Address 3: Source address (indirizzo del nodo sorgente che si trova in un altro BSS)
    - Address 4: non usato
- Il riscontro ACK deve essere inviato all'AP
- Caso 3 (10)** : “To DS”=1 , “From DS”=0:
  - La frame è diretta verso il Distribution System
  - La frame è diretta verso un AP diverso rispetto a quello del BSS a cui appartiene il nodo sorgente:
    - Address 1: identificatore del BSS a cui appartiene il nodo di destinazione (AP address di destinazione)
    - Address 2: Source Address
    - Address 3: Destination address (indirizzo del nodo destinazione che si trova in un altro BSS)
    - Address 4: non usato
- Il riscontro ACK deve essere inviato al nodo sorgente
- Caso 4 (11)** : “To DS”=1 , “From DS”=1:
  - È il caso in cui una frame è emessa da un AP ed è diretta verso un altro AP dello stesso BSS
  - I nodi sorgente e destinazione sono interni allo stesso Basic Service Set (BSS):
    - Address 1: Receiving Address (Indirizzo dell'AP di destinazione)
    - Address 2: Transmitting Address (Indirizzo dell'AP di origine)
    - Address 3: Destination Address (indirizzo dell'effettivo nodo di destinazione)
    - Address 4: Source Address (indirizzo dell'effettivo nodo sorgente)

## Esempi

### Caso 1 (00)



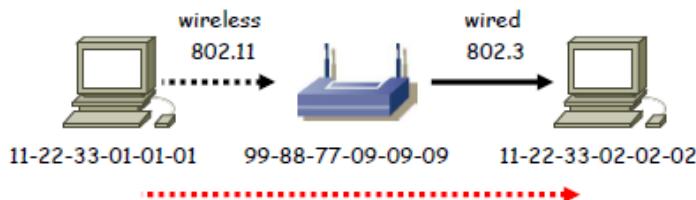
### Caso 2 (01)



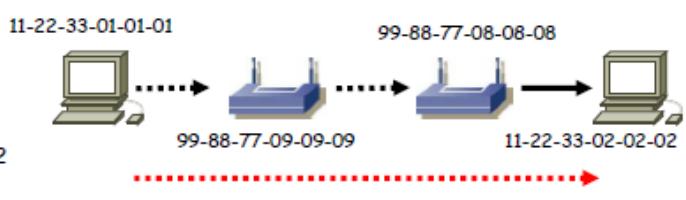
| Campo     | Valore            | Significato |
|-----------|-------------------|-------------|
| Address 1 | 11-22-33-01-01-01 | DA          |
| Address 2 | 11-22-33-02-02-02 | SA          |
| Address 3 | xx.xx.xx.xx.xx.xx | BSS ID      |
| Address 4 |                   |             |

| Campo     | Valore            | Significato |
|-----------|-------------------|-------------|
| Address 1 | 11-22-33-01-01-01 | DA          |
| Address 2 | 99-88-77-09-09-09 | Sending AP  |
| Address 3 | 11-22-33-02-02-02 | SA          |
| Address 4 |                   |             |

### Caso 3 (10)



### Caso 4 (11)



| Campo     | Valore            | Significato  |
|-----------|-------------------|--------------|
| Address 1 | 99-88-77-09-09-09 | Receiving AP |
| Address 2 | 11-22-33-01-01-01 | SA           |
| Address 3 | 11-22-33-02-02-02 | DA           |
| Address 4 |                   |              |

| Campo     | Valore            | Significato  |
|-----------|-------------------|--------------|
| Address 1 | 99-88-77-08-08-08 | Receiving AP |
| Address 2 | 99-88-77-09-09-09 | Sending AP   |
| Address 3 | 11-22-33-02-02-02 | DA           |
| Address 4 | 11-22-33-01-01-01 | SA           |

## Roaming

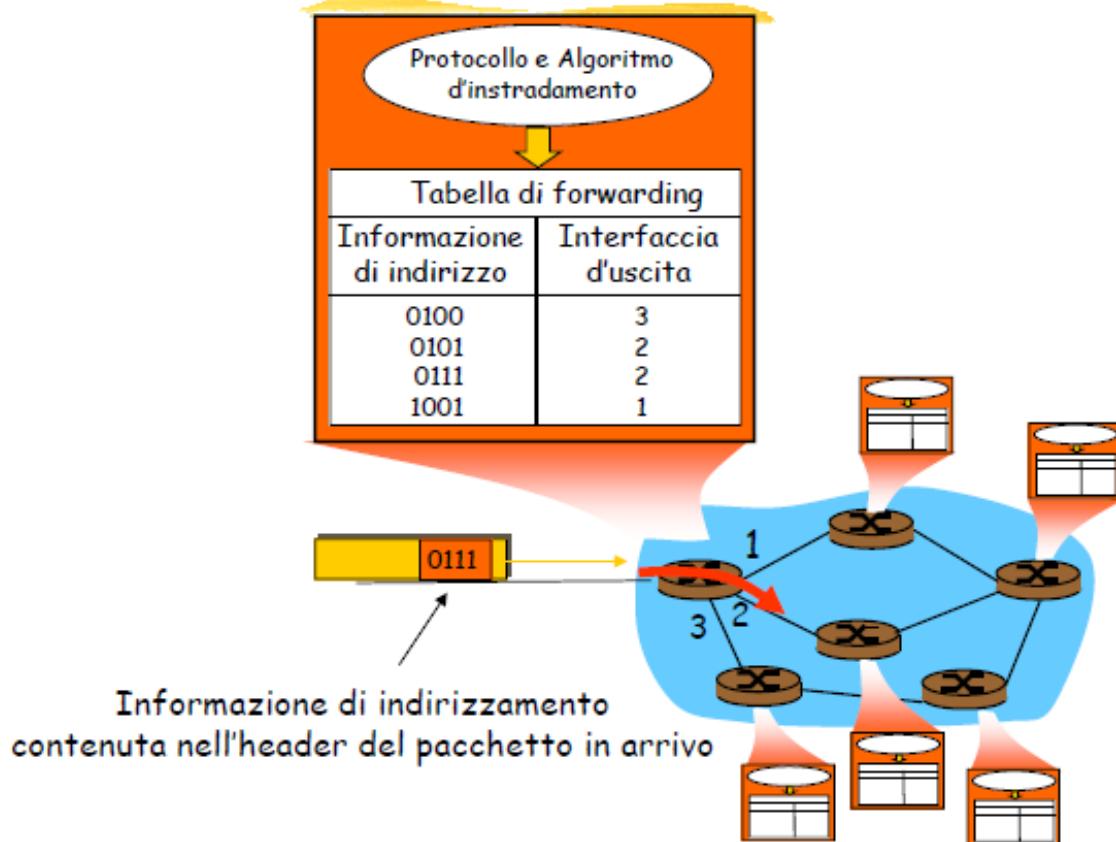
- Un nodo può migrare da una BSS ad un'altra, da un'area coperta da un AP ad un area coperta da un altro AP
- Procedura di "Re-association":
  - Un nodo decide che il collegamento verso l'AP non è affidabile
  - Il nodo esegue la funzione di "scanning" del mezzo radio per trovare un altro AP
  - In caso di esito positivo, il nodo emette una "Re-association Request" verso il nuovo AP
  - Se la "Re-association Response" è positiva il nodo entra a far parte della BSS gestita dal nuovo AP (roaming), altrimenti cerca un ulteriore AP
  - Se un AP accetta una "Re-association Request":
    - Indica la riassociazione al Distributed System (DS)
    - Le informazioni del DS sono aggiornate

## "Reti a pacchetto" e "Protocollo IP"

### Funzioni del livello di rete

- **Forwarding** (inoltro):
  - **Funzione attuativa**
  - Trasferisce i pacchetti da un interfaccia di ingresso di un nodo (router) verso un'opportuna interfaccia di uscita
- **Routing** (instradamento):
  - **Funzione decisionale**
  - Determina il percorso seguito dai pacchetti dall'origine alla destinazione:
    - Per ciascun router determina l'interfaccia di uscita su cui deve essere inoltrato un pacchetto
  - Utilizza protocolli e algoritmi specifici

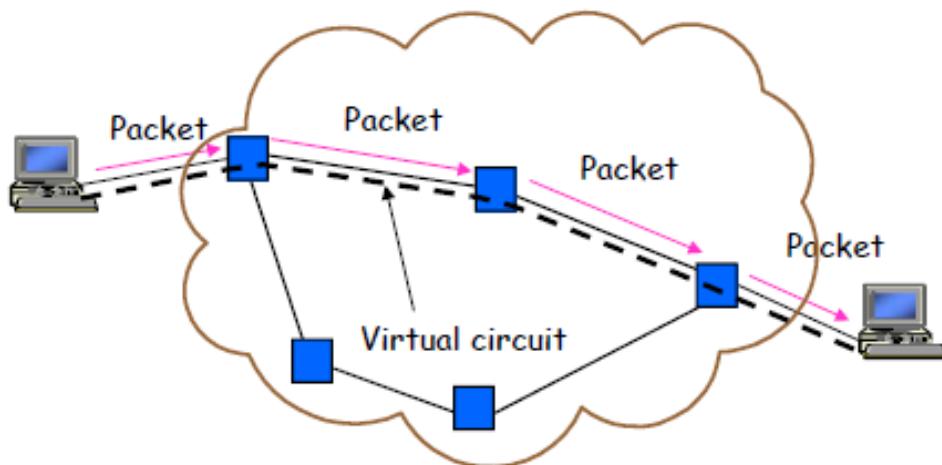
## Routing e forwarding



### Tipologie di servizio di rete

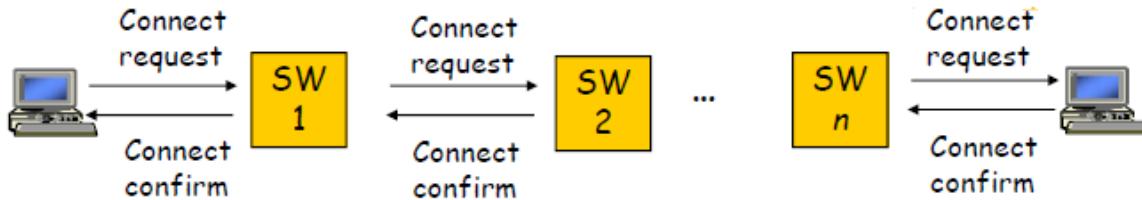
- Servizio senza connessione:
  - Reti a "datagramma"
  - I pacchetti sono inviati da una sorgente senza un preventivo accordo sia con la destinazione sia con la rete
  - I pacchetti sono trattati dalla rete e quindi da ciascun nodo come entità indipendenti
  - L'instradamento è deciso pacchetto per pacchetto
  - I router hanno un funzionamento "stateless"
- Servizio connection-oriented:
  - Reti a "circuito virtuale" (VC)
  - Prima dell'invio dei pacchetti viene instaurata una **connessione di rete**
  - Il cammino (path) di instradamento dei pacchetti è deciso al momento dell'instaurazione della connessione
  - I nodi hanno un funzionamento "statefull":
    - I nodi mantengono informazioni sullo stato delle connessioni

### Packet Switching – Virtual Circuit



- È necessaria una fase di set-up della connessione
- È necessario un protocollo di segnalazione
- Tutti i pacchetti seguono lo stesso path in rete
  - Consegnata in sequenza dei pacchetti
- L'informazione di indirizzamento contenuta nell'header di ogni pacchetto è l'**identificatore della connessione** a cui appartiene:
  - L'identificazione della connessione avviene "per link"

### Connection Setup



- I messaggi di segnalazione:
  - Sono trasmessi lungo il path della connessione
  - Determinano in ogni nodo l'esecuzione della funzione di routing che identifica il nodo successivo sul path
  - Inizializzano le tabelle di forwarding nei nodi

- La connessione è identificata su ogni link da un "local tag" (**Virtual Circuit Identifier – VCI**)
- Ogni nodo (**switch**) memorizza la relazione tra input tag e output tag e interfaccia di uscita nella tabella di forwarding
- Una volta che le tabelle di forwarding sono inizializzate i pacchetti possono essere trasmessi in rete

### Connection Setup Delay



- Il ritardo di instaurazione della connessione (**connection setup delay**) si somma al ritardo di transito dei pacchetti
- Tale ritardo addizionale è:
  - Tollerabile se è inferiore al tempo di trasferimento dei pacchetti dati
  - Inaccettabile se devono essere trasferiti pochi pacchetti

### Virtual Circuit Forwarding Table

| Input VCI | Output port | Output VCI |
|-----------|-------------|------------|
| 12        | 13          | 44         |
| 15        | 15          | 23         |
| 27        | 13          | 16         |
| 58        | 7           | 34         |

A blue arrow points from the '27' entry in the first row to the 'Input VCI' column of the table, indicating the search path for a packet with VCI 27.

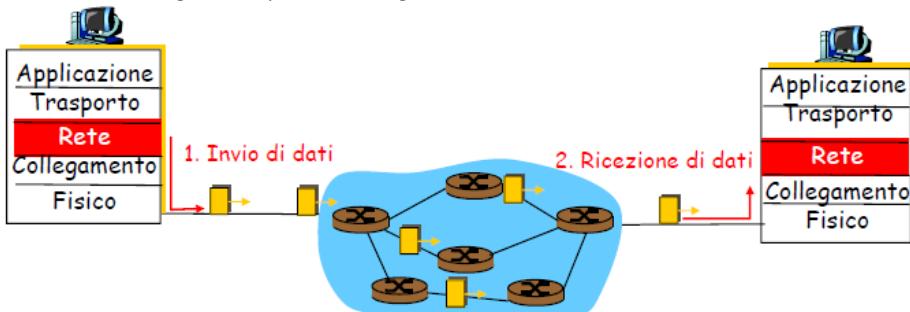
- Ogni porta di ingresso ha una propria forwarding table
- Si utilizza il VCI contenuto nell'header del pacchetto come indice di accesso della tabella
- Si individua il record corrispondente al VCI, si legge la porta di uscita e il valore del VCI sul link d'uscita
- Il valore del VCI d'uscita è scritto nell'header del pacchetto
- Commutazione molto veloce

### Riassumendo

- Un circuito virtuale consiste in:
  - Un percorso tra origine e destinazione
  - Identifieri di connessione (VCI), uno per ciascun link
  - Righe nella tabella di forwarding in ciascun nodo (switch)
- Il pacchetto di un circuito virtuale ha un VCI nella propria intestazione
- Il VCI del pacchetto cambia su tutti i collegamenti lungo un percorso:
  - Un nuovo VCI viene rilevato dalla tabella di forwarding

## Reti a datagramma

- I router della rete sono "stateless"
  - Non esiste il concetto di "connessione" a livello di rete
- I router utilizzano gli **indirizzi di destinazione** per effettuare il forwarding:
  - I pacchetti possono seguire percorsi diversi in rete
  - La consegna in sequenza non è garantita



## Esempio di Tabella di routing

| Intervallo degli indirizzi di destinazione | Interfaccia |
|--------------------------------------------|-------------|
| da 11001000 00010111 00010000 00000000     | 0           |
| a 11001000 00010111 00010111 11111111      |             |
| da 11001000 00010111 00011000 00000000     | 1           |
| a 11001000 00010111 00011000 11111111      |             |
| da 11001000 00010111 00011001 00000000     | 2           |
| a 11001000 00010111 00011111 11111111      |             |
| altrimenti                                 | 3           |

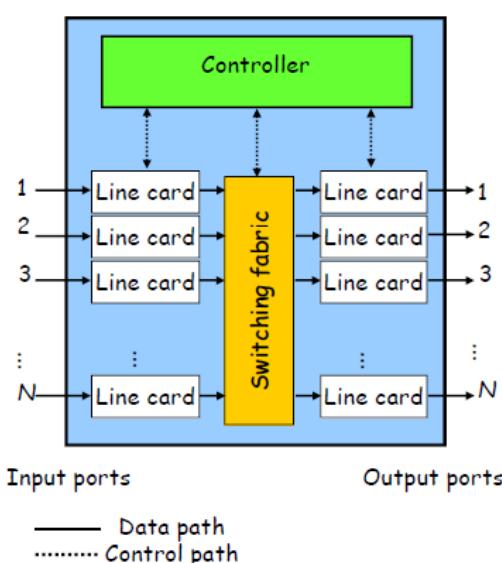
$2^{32} = \text{circa 4 miliardi di possibili indirizzi}$

## Concetto di prefisso

| Prefisso                   | Interfaccia |
|----------------------------|-------------|
| 11001000 00010111 00010    | 0           |
| 11001000 00010111 00011000 | 1           |
| 11001000 00010111 00011    | 2           |
| altrimenti                 | 3           |

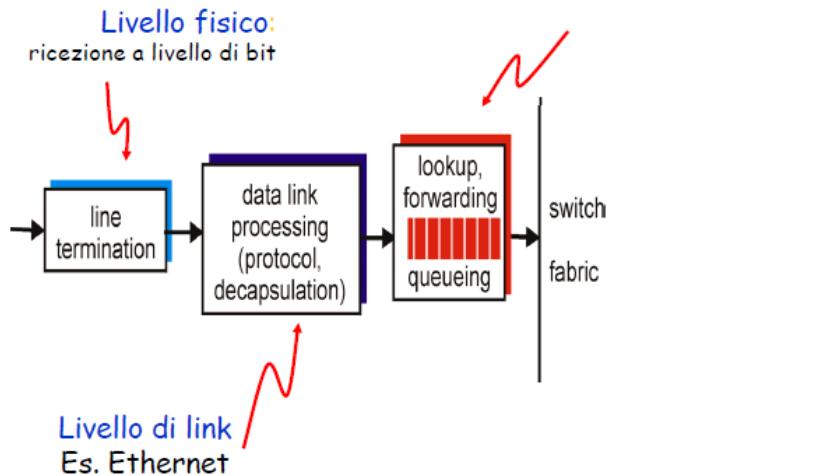
- È necessaria una fase di ricerca del prefisso nella tabella:
  - Algoritmi di lookup

## Architettura di un router



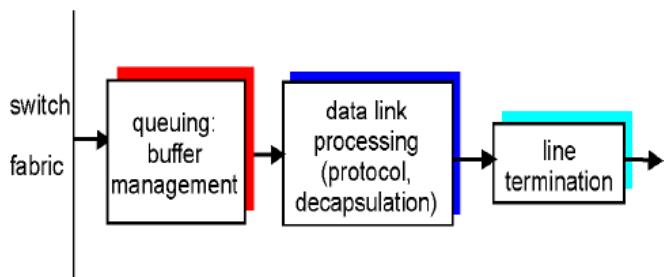
- **Input Line Card:**
  - Funzioni di strato 1 & 2
  - Header processing
  - Routing
- **Controller:**
  - Segnalazione & resource allocation
- **Switching Fabric:**
  - Funzione di forwarding tra porte di ingresso e di uscita
- **Output Line Card:**
  - Scheduling & priority
  - Multiplexing

#### Porte d'ingresso (Input Line Card)



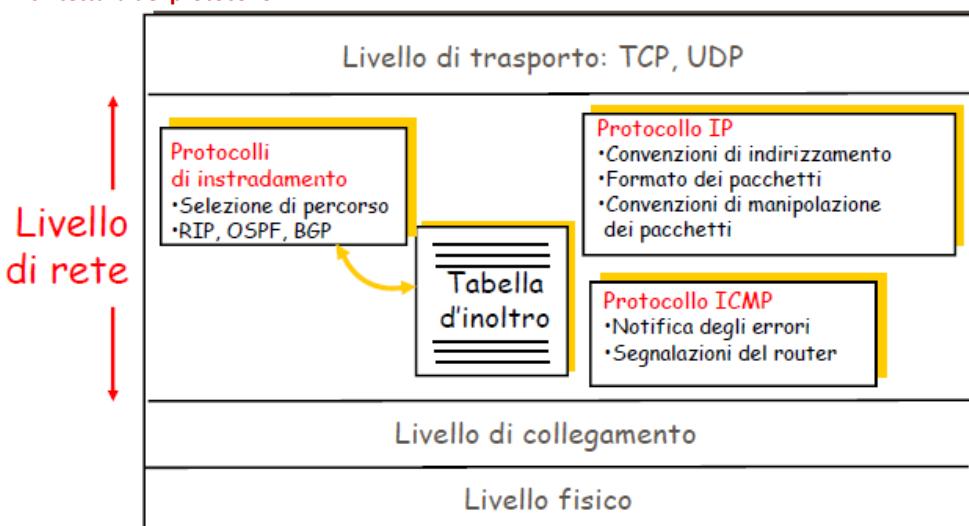
- Determina la porta d'uscita dei pacchetti utilizzando le informazioni della tabella di routing:
  - Obiettivo: completare l'elaborazione allo stesso tasso della linea
- Funzione di bufferizzazione se il tasso di arrivo dei pacchetti è superiore a quello di inoltro

#### Porte d'uscita



- **Bufferizzazione:**
  - Quando la struttura di commutazione consegna pacchetti alla porta d'uscita a una frequenza che supera quella del collegamento uscente
- **Packet scheduling:**
  - Stabilisce in quale ordine trasmettere i pacchetti accodati

#### Architettura del protocollo IP



## Architettura protocollare

- **Il protocollo IP (RFC 791, 919, 922, 950, 1349):**

- è un protocollo di strato di rete
- Opera con modalità di trasferimento senza connessione
- Non fornisce alcuna garanzia sulla QoS (servizio "best effort")

- **Il protocollo IP esegue le seguenti funzioni:**

- Definisce il formato dei pacchetti:
  - La lunghezza massima di un pacchetto è di 65536 byte
- Definisce lo schema di indirizzamento
- Definisce le modalità di instradamento dei pacchetti
- Esegue, se necessario, la frammentazione e il ri-assemblaggio delle unità dati

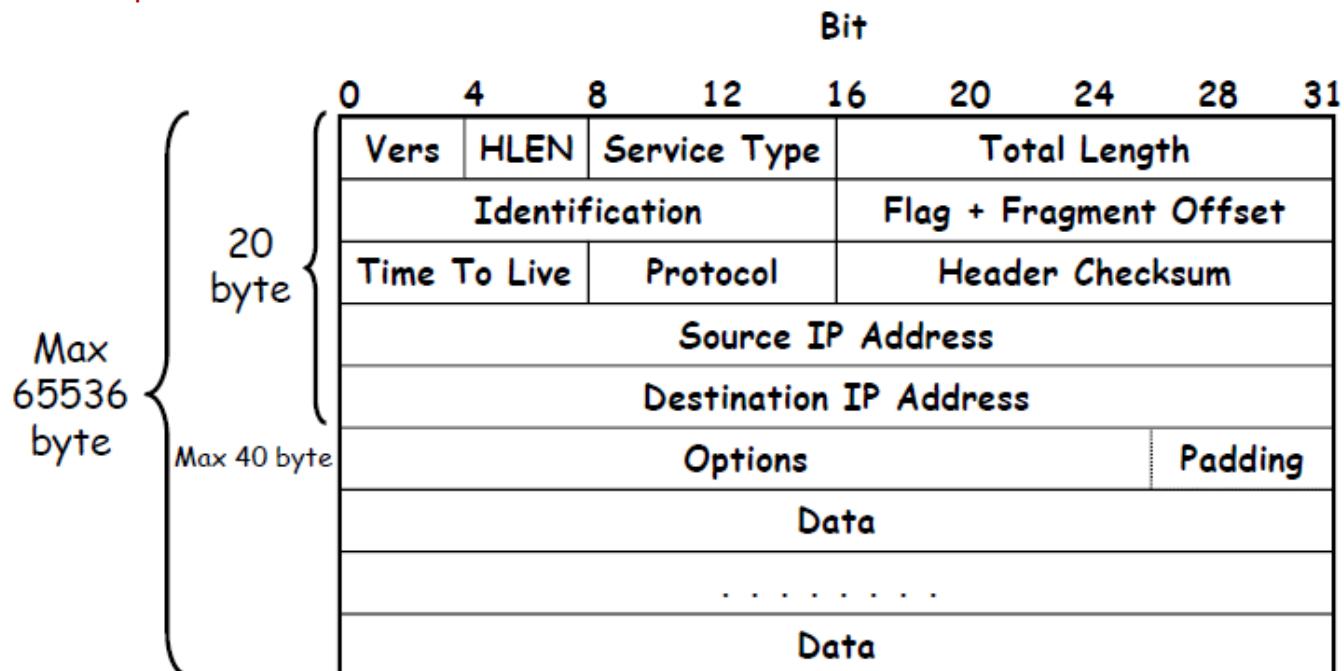
- **Il protocollo ICMP (Internet Control Message Protocol):**

- è un protocollo senza connessione
- è orientato a:
  - Risolvere eventuali situazioni anomale
  - Controllare il trasferimento (controllo di flusso di tipo On-Off)
  - Comunicare alle sorgenti eventuali problemi (ad es. di indirizzamento)

- **Esempi:**

- **Source Quench:** inviato dal destinatario, interrompe l'emissione di pacchetti del mittente;
- **Redirect:** il destinatario segnala al mittente di re-instradare il pacchetto verso un altro host;
- **Echo:** controlla se un possibile destinatario è attivo,
- **Destination Unreachable:** notifica il mittente la non-raggiungibilità di un host

## Formato del pacchetto IPv4



## Formato del pacchetto IP

- **Vers (4 bit):**

- Versione del protocollo, è possibile la coesistenza di più versioni di IP

- **Header Length (HLEN) (4 bit):**

- Lunghezza dell'intestazione (specificata in parole di 32 bit) comprende la parte fissa (20 bytes) e la parte opzionale
- Valore massimo: 60 byte

- **Total length (16 bit):**

- Lunghezza complessiva del pacchetto (specificata in byte)
- Comprende la lunghezza dell'header e del payload
- Valore massimo: 65536 byte

- **Service Type (8 bit):**

- Specifica i parametri di qualità di servizio richiesti dall'utente per il pacchetto

- **Precedence (3 bit):**

- Indicano il livello di priorità del pacchetto
- In passato non sono stati utilizzati
- Ora implementano i meccanismi DiffServ

|            |       |       |         |      |   |
|------------|-------|-------|---------|------|---|
| Precedence | Delay | Thput | Reliab. | Cost | 0 |
| 1          | 2     | 3     | 4       | 5    | 6 |

### Service Type (8 bit)

- **Type Of Service (TOS) (4 bit):**

- Indicano il tipo di servizio richiesto per il pacchetto
- Il servizio normale si ha se tutti i quattro bit sono a 0
- Solo uno dei quattro bit può essere posto a 1

|         |                        |
|---------|------------------------|
| 1 0 0 0 | Minimize delay         |
| 0 1 0 0 | Maximize Throughput    |
| 0 0 1 0 | Maximize Reliability   |
| 0 0 0 1 | Minimize Monetary Cost |
| 0 0 0 0 | Normal Service         |

- Ogni rete fisica ha un valore massimo di lunghezza della propria unità informativa:

- Maximum Transmission Unit - MTU

- La frammentazione di un pacchetto IP è necessaria se il valore della MTU nella sottorete fisica è inferiore alla lunghezza del pacchetto:
  - Il valore minimo di una MTU è 68 byte

- La frammentazione è effettuata dal router/host prima del rilancio nella sottorete
- La ricomposizione del pacchetto originale è effettuata dall'host di destinazione

- **Identification (16 bit):**

- Numero identificativo del pacchetto da frammentare
- è assegnato dal processo sorgente

- **Flags (3 bit):**

- X: non usato e posto a zero
- DF: Don't Fragment (0: frammentazione permessa; 1: frammentazione vietata)
- MF: More Fragment (0: ultimo frammento del pacchetto; 1: non è l'ultimo frammento)

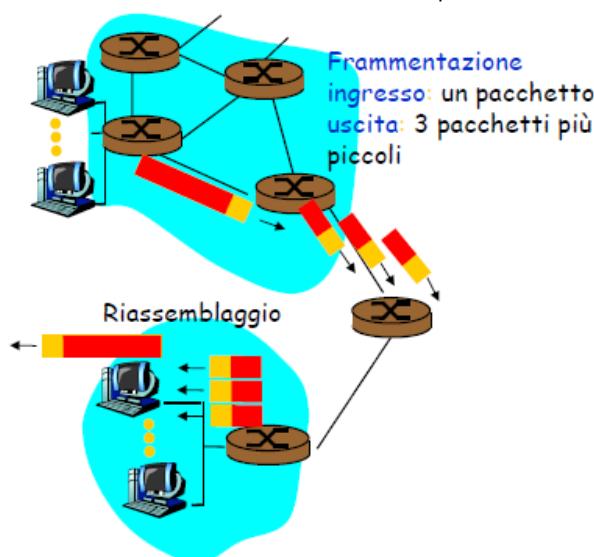
- **Fragment Offset (13 bit):**

- Posizione del frammento all'interno del pacchetto (espresso in unità di 8 byte) consente di valutare l'integrità del pacchetto

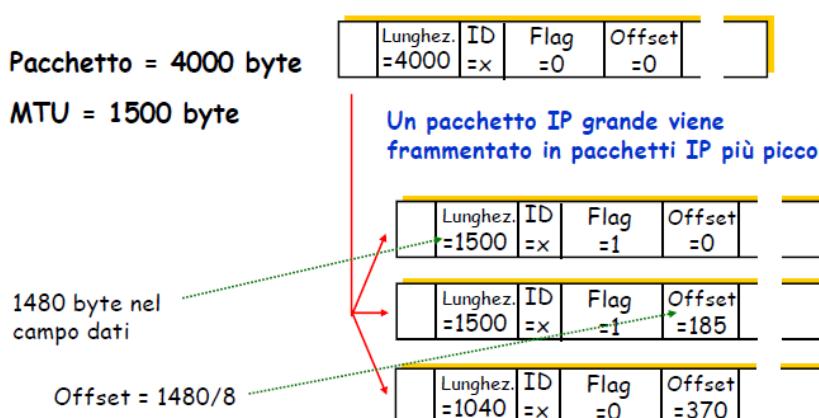
### Frammentazione dei pacchetti IP

- L'unità massima di trasmissione (MTU) è la massima quantità di dati che un frame a livello di collegamento può trasportare:
  - Differenti tipi di link, differenti MTU

- Pacchetti IP grandi vengono suddivisi ("frammentati") in pacchetti IP più piccoli:
  - I frammenti saranno riassemblati solo una volta raggiunta la destinazione
  - I bit dell'intestazione IP sono usati per identificare e ordinare i frammenti



### Esempio



### Formato del pacchetto IP

- **Time To Live (TTL) (8 bit):**
  - Indica il numero massimo di router che possono essere ancora attraversati dal pacchetto
  - è inizializzato dall'host sorgente ed è decrementato di una unità da ogni router
  - Quando il valore del campo è nullo, il pacchetto è scartato e viene emesso un messaggio ICMP di notifica verso l'host sorgente
- **Protocol (8 bit):**
  - Indica a quale protocollo dello stato superiore deve essere trasferito il contenuto informativo del pacchetto (es. TCP=6, UDP=17, ICMP=1)
- **Header Checksum (16 bit):**
  - Protegge solo l'intestazione del pacchetto
  - Se viene rivelato un errore il pacchetto è scartato
- **Source Address (32 bit) e Destination Address (32 bit)**
- **Options (lunghezza variabile a multipli di 8 bit):**
  - **Record Route Option (RRO):**
    - Lista vuota di indirizzi IP, ogni router attraversato inserisce il suo indirizzo
  - **Timestamp Option:**
    - Come RRO con in più l'istante in cui il pacchetto attraversa ogni router
  - **Loose Source Routing Option (LSRO):**
    - Specifica una lista di router che devono essere attraversati dal pacchetto
  - **Strict Source Route Option (SSRO):**
    - Specifica tutti i router attraverso i quali deve transitare il pacchetto
- **Padding:**
  - Rende l'intestazione multipla di 32 bit mediante introduzione di zeri

### Il protocollo ICMP (Internet Control Message Protocol)

- Il protocollo ICMP (RFC 792, 950) consente ai router di inviare all'host sorgente informazioni riguardanti anomalie nel processamento di un pacchetto:
  - Errori di instradamento
  - TTL scaduto
  - Congestione eccessiva
- ICMP è una parte integrante di IP e deve essere incluso in ogni implementazione di IP
- Un messaggio ICMP è incapsulato nella parte dati di un pacchetto IP

### ICMP

- ICMP ha lo scopo esclusivo di notificare errori all'host di origine:
  - ICMP non specifica le azioni che devono essere prese per rimediare ai malfunzionamenti
  - Spetta all'host di origine decidere le azioni da intraprendere per correggere il problema
- I messaggi ICMP non sono elaborati dai router intermedi
- Non vengono generati nuovi messaggi ICMP in seguito ad errori causati da pacchetti contenenti messaggi ICMP:
  - Evita messaggi di errore relativi a messaggi di errore
- Un messaggio ICMP si riferisce ad uno specifico pacchetto
- Un messaggio ICMP contiene l'indicazione del particolare pacchetto IP che ha generato l'errore:
  - Nel caso di frammentazione, un messaggio ICMP viene emesso solo per il frammento 0
- Formato messaggio ICMP

| Byte      |      |          |    |    |
|-----------|------|----------|----|----|
| 0         | 8    | 16       | 24 | 31 |
| Type      | Code | Checksum |    |    |
| ICMP data |      |          |    |    |

- **Type (4 bit):**
  - Identificano il particolare messaggio ICMP

|   |                   |    |                    |
|---|-------------------|----|--------------------|
| 0 | Echo reply        | 11 | Time exceeded      |
| 3 | Dest. Unreachable | 13 | Time stamp request |
| 4 | Source Quench     | 14 | Time stamp replay  |
| 5 | Redirect          | 17 | Address mask req.  |
| 8 | Echo              | 18 | Address mask rep.  |

- **Code (4 bit):**
  - Contiene il codice di errore
- **Data:**
  - Consente l'individuazione del pacchetto che ha causato l'errore
  - Contiene parte del pacchetto IP

- **Redirect message:**

- Se è emesso da un router significa che i successivi pacchetti emessi dall'host verso la rete dovranno essere indirizzati verso il router indicato nel messaggio ICMP
- Causa una modifica della tabella di instradamento dell'host sorgente

- **Source quench:**

- Se è emesso da un router intermedio indica che il router non ha buffer sufficiente per memorizzare il pacchetto
- Se è emesso dall'host di destinazione indica che il pacchetto non è stato processato dall'host
- Il messaggio è utilizzato dal TCP

- **Time exceeded:**

- Indica che il TTL si è esaurito

- **Echo e Echo replay:**

- Sono utilizzati per stabilire l'attività di un elemento di un host

- **Destination unreachable:**

- Indica che l'instradamento di un pacchetto non è stato completato

- **Time Stamp Request e Time Stamp Replay:**

- Sono utilizzati per effettuare misure di prestazioni (es. ritardi di transito)

- **Address mask request e Address mask replay:**

- Sono usati per determinare la maschera della sotto-rete a cui è connesso un host
- Sono usati da host molto semplici (diskless) dopo aver individuato il proprio indirizzo con il protocollo RARP

## Ping

- **Si utilizza per verificare:**

- L'installazione della pila TCP/IP
- L'attività di un host
- Il tempo di transito tra host sorgente e host destinazione

- **Utilizza i messaggi ICMP Echo e Echo Replay**

## Traceroute

- **Il programma invia una serie di pacchetti IP alla destinazione:**

- Il primo con TTL =1
- Il secondo con TTL=2, ecc.
- Numero di porta qualsiasi

- **Quando l'n-esimo pacchetto arriva all'n-esimo router:**

- Il router scarta il pacchetto
- Invia all'origine un messaggio di allerta ICMP (tipo 11, codice 0)
- Il messaggio include il nome del router e l'indirizzo IP

- **Quando il messaggio ICMP arriva, l'origine può calcolare RTT**

- **Traceroute lo fa per 3 volte**

- **Criteri di arresto dell'invio:**

- Quando un segmento UDP arriva all'host di destinazione
- L'host di destinazione restituisce un messaggio ICMP di porta non raggiungibile (tipo 3, codice 3).
- Quando l'origine riceve questo messaggio ICMP, si blocca

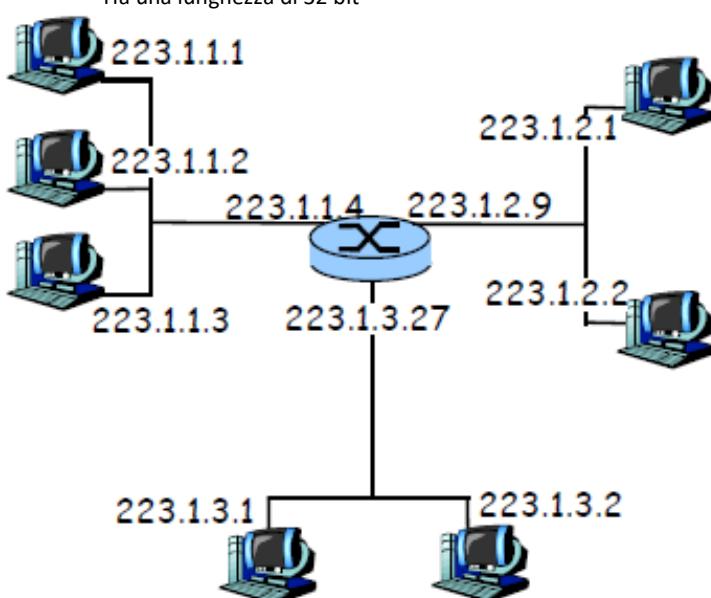
## Indirizzamento in IPv4

- Un indirizzo IP (IP Address) identifica un'interfaccia di rete:

- Se un host è connesso a più di una rete (multi-homed) avrà un indirizzo IP per ogni interfaccia
- Un router ha tanti indirizzi IP quanto sono le interfacce di rete che gestisce

- Un indirizzo IP pubblico è unico in tutta la rete:

- Ha una lunghezza di 32 bit



## Schema di indirizzamento

- **Notazione numerica:**

- L'indirizzo è espresso da una stringa di 32 bit

- **Notazione "dotted":**

- Ogni gruppo di 8 bit della notazione numerica è sostituito dall'equivalente numero decimale

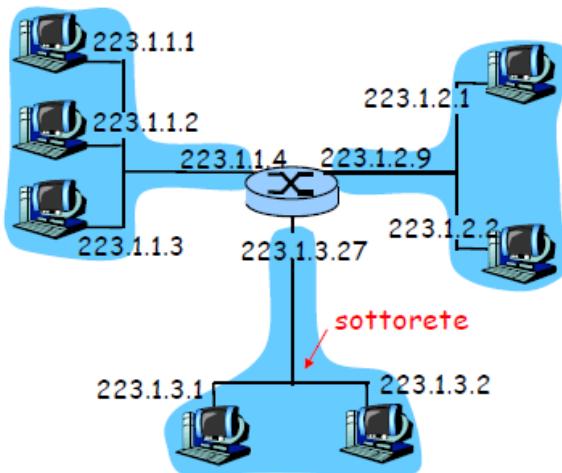
**Notazione Numerica** 10010111 01100100 00001000 00010010

**Notazione Dotted**

151. 100. 8. 18

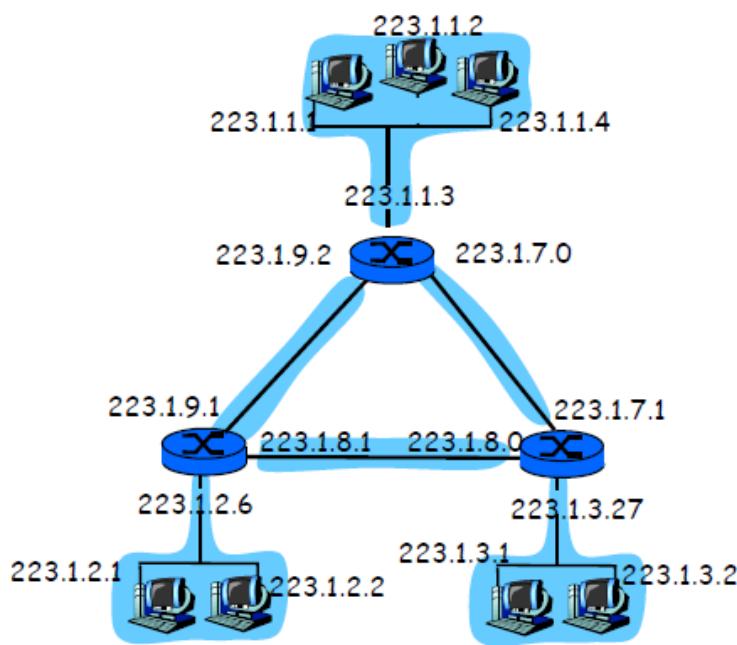
## Indirizzamento IP

- Una **sottorete** è una rete isolata i cui punti terminali sono collegati all'interfaccia di un host o di un router
  - Esempio: LAN
- Una sottorete è anche detta **rete IP**



rete composta da 3 sottoreti

- Un link diretto tra due router è una sottorete con due interfacce



- Un indirizzo IP è formato da due parti:

- **Prefisso, Net\_Id:** identificativo di sotto-rete  
- **Host\_Id:** identificativo di host all'interno della sotto-rete

- **IP\_Address = Net\_Id . Host\_Id:**

- La divisione tra Net\_Id e Host\_Id non è fissa

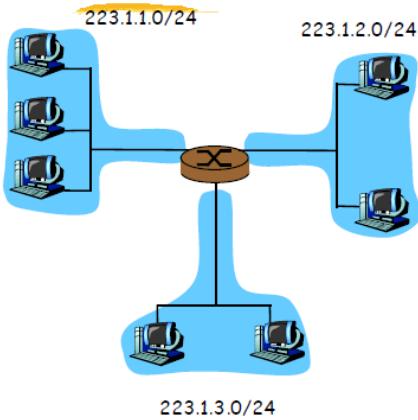
- Una **sottorete** è una rete isolata i cui punti terminali sono collegati all'interfaccia di un host o di un router

## Concetto di prefisso di sottorete

- Una sottorete è identificata da un **prefisso** (Net\_Id):

- Parte dell'indirizzo IP identica per tutte le interfacce che appartengono alla sottorete

- Gli indirizzi IP delle interfacce di una stessa sottorete sono caratterizzati dallo stesso prefisso



Maschera di sottorete: /24

#### Schema di indirizzamento "Classfull"

- In origine (1981, RFC 1166) le sotto-reti erano divise in classi:

- La classe era individuata dai bit iniziali dell'indirizzo
- I prefissi (Net\_Id) di sottorete avevano lunghezza fissa

| Classe | Bit iniziali | Net_Id                                                               | Host_Id | "Reti" disponibili | "Host" disponibili |
|--------|--------------|----------------------------------------------------------------------|---------|--------------------|--------------------|
| A      | 0            | 7 bit                                                                | 24 bit  | 128                | 16.777.216         |
| B      | 10           | 14 bit                                                               | 16 bit  | 16384              | 65.536             |
| C      | 110          | 21 bit                                                               | 8 bit   | 2.097.152          | 256                |
| D      | 1110         | Indirizzo multicast: 28 bit<br>Indirizzi possibili: 268.435.456      |         |                    |                    |
| E      | 11110        | Riservata per usi futuri: 27 bit<br>Indirizzi possibili: 134.217.728 |         |                    |                    |

- Classi di indirizzi IP:

0                    8                    16                    24                    31

Classe A 0 Net\_Id Host\_Id

Classe B 1|0 Net\_Id Host\_Id

Classe C 1|1|0 Net\_Id Host\_Id

Classe D 1|1|1|0 Multicast Address

Classe E 1|1|1|1|0 Reserved

#### Convenzioni speciali

- Se un host si muove dalla rete in cui si trova, il suo indirizzo deve essere cambiato:

- Mobilità: protocollo Mobile IP

- Convenzioni speciali:

Questo host (fase di boot)      Tutti "0"

Host nella rete locale      Tutti "0"      Host\_Id

Broadcast sulla rete locale      Tutti "1"

Broadcast sulla rete Net\_Id      Net\_Id      Tutti "1"

#### Subnetting

- La struttura di indirizzamento a due livelli gerarchici era sufficiente nella fase iniziale di Internet

- Nel 1984 è stato aggiunto un terzo livello gerarchico:

- Il livello di Sottorete (Subnet)

- Si utilizzano alcuni bit dell'Host\_Id per codificare il Subnet\_Id

|                  |     |        |         |
|------------------|-----|--------|---------|
| Original address | 1 0 | Net ID | Host ID |
|------------------|-----|--------|---------|

|                   |     |        |           |         |
|-------------------|-----|--------|-----------|---------|
| Subnetted address | 1 0 | Net ID | Subnet ID | Host ID |
|-------------------|-----|--------|-----------|---------|

- Il campo Subnet\_Id è identificato da una maschera denominata **“Subnet Mask”**
  - Una Subnet Mask è una parola di 32 bit in cui:
    - I bit uguali a “1” identificano i bit del Net\_Id e del Subnet\_Id
    - I bit uguali a “0” identificano i bit dell’Host\_Id
  - La Subnet\_Id ha significato solo nel router a cui sono connesse le sottoreti

|            |       |        |           |         |
|------------|-------|--------|-----------|---------|
| IP address | 1   0 | Net ID | Subnet ID | Host ID |
|------------|-------|--------|-----------|---------|

|             |   |   |   |   |   |   |   |   |     |   |   |   |   |   |   |   |   |   |   |   |
|-------------|---|---|---|---|---|---|---|---|-----|---|---|---|---|---|---|---|---|---|---|---|
| Subnet Mask | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | ... | 1 | 1 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
|-------------|---|---|---|---|---|---|---|---|-----|---|---|---|---|---|---|---|---|---|---|---|

## **Subnetting Statico (lunghezza fissa)**

- Tutte le subnet hanno la stessa maschera
  - Esempio:

| 0 | Net_id | Host_id |
|---|--------|---------|
|---|--------|---------|

| Subnet Mask | 1                                                                         | 8   | 16  | 24  | 32  |
|-------------|---------------------------------------------------------------------------|-----|-----|-----|-----|
|             | 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 0 0 0 0 0 0 0 | 255 | 255 | 255 | 192 |
|             |                                                                           |     |     |     |     |

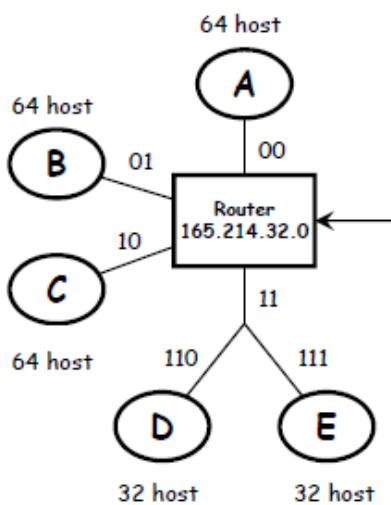
- Numero massimo di sottoreti possibili =  $2^{18} = 262.144$

- Numero massimo di host per sottorete =  $2^6 - 2 = 62$

## **Subnetting a lunghezza variabile**

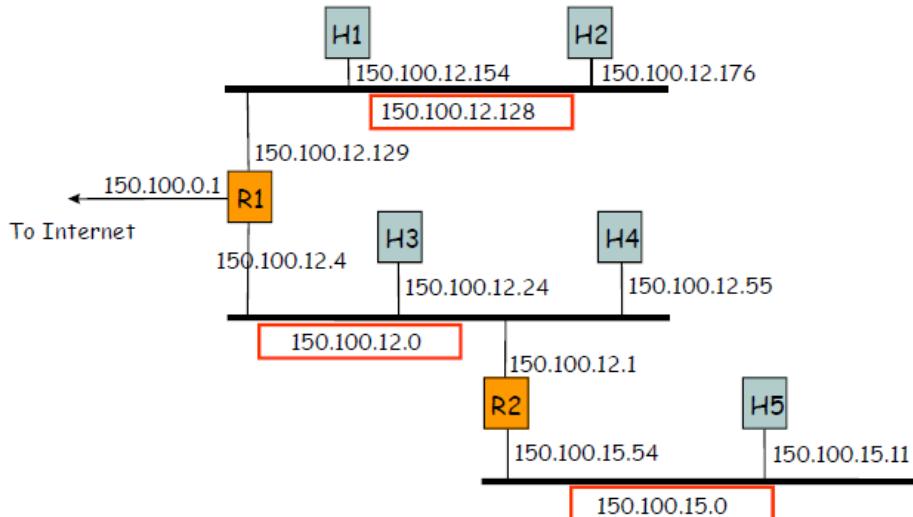
- Le sotto-reti di una rete usano maschere diverse:
    - Consente di gestire reti di dimensione diversa

- Esempio:
    - Router con un indirizzo di classe C:
      - 193.214.32.0
    - 5 Sottoreti:
      - Subnet A, Subnet B, Subnet C: 50 host
      - Subnet D, Subnet E: 30 host
    - Subnetting:
      - 3 sottoreti con 64 host ciascuna (Host\_id: 6 bit) (subnet mask 255.255.255.192)
      - 1 sottorete divisa in due ulteriori sottoreti con 32 host ciascuna (Host\_id: 5 bit) (subnet mask 255.255.255.224)



## Esempio 2

- Un provider ha un indirizzo di classe B (Host Id = 16 bit) con Net\_Id = 150.100.0.0
  - Si devono creare sottoreti con un numero massimo di 100 host ciascuna:
    - 7 bit sufficienti per ciascuna sottorete
    - $16 - 7 = 9$  bit per il Subnet\_Id
  - Si applicano le subnet mask per individuare la sottorete:
    - Esempio: trovare la sottorete per 150.100.12.176
    - IP add = 10010110 01100100 00001100 10110000
    - Mask = 11111111 11111111 11111111 10000000
    - AND = 10010110 01100100 00001100 10000000
    - Subnet = 150.100.12.128
    - L'indirizzo di sottorete è usato dai router del provider



### Routing in reti IP

- Sia gli host che i router hanno una **Tabella di Routing** (Routing table)

- Host origine:**

- Se la destinazione è sulla stessa rete, il pacchetto è emesso direttamente usando l'interfaccia di rete:  
- La frame in cui viene incapsulato il pacchetto conterrà l'indirizzo MAC della destinazione
- Se la destinazione non è nella stessa rete, il pacchetto è inviato al **default router**:  
- La frame in cui viene incapsulato il pacchetto conterrà l'indirizzo MAC del router

- Router:**

- Esamina l'indirizzo IP di destinazione (IP destination address) nel pacchetto entrante
- Se la destinazione è su una delle reti a cui è connesso il router, il pacchetto è emesso direttamente usando l'interfaccia di rete
- Se la destinazione non è su una delle reti a cui è connesso il router, il router accede alla routing table per determinare il next-hop verso cui inoltrare il pacchetto

### Routing Table

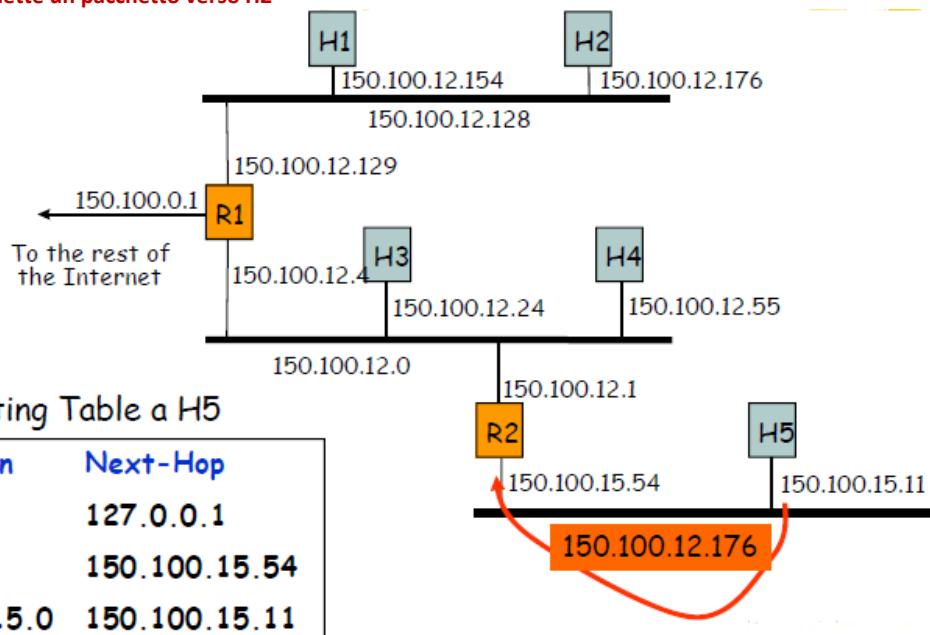
- Ogni riga contiene:**

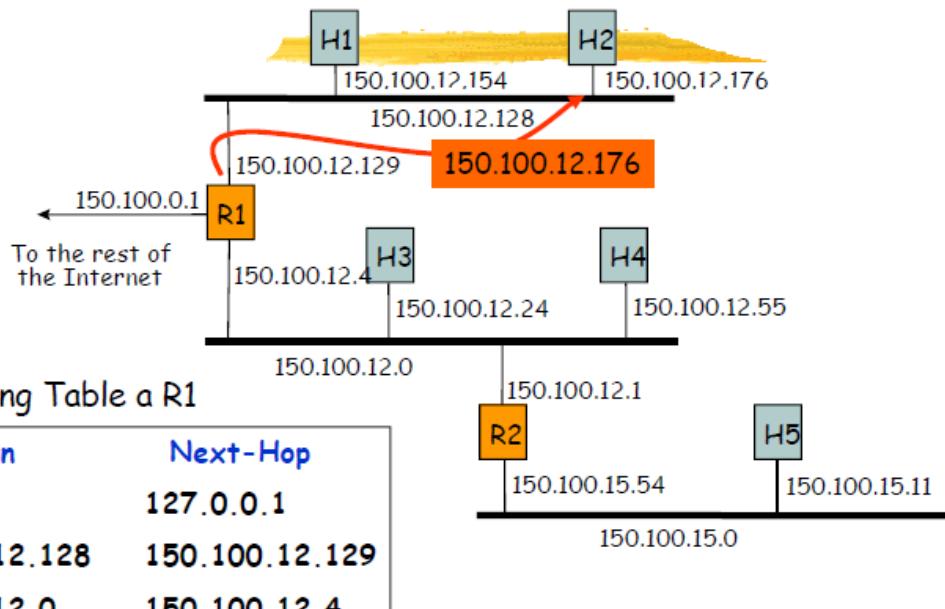
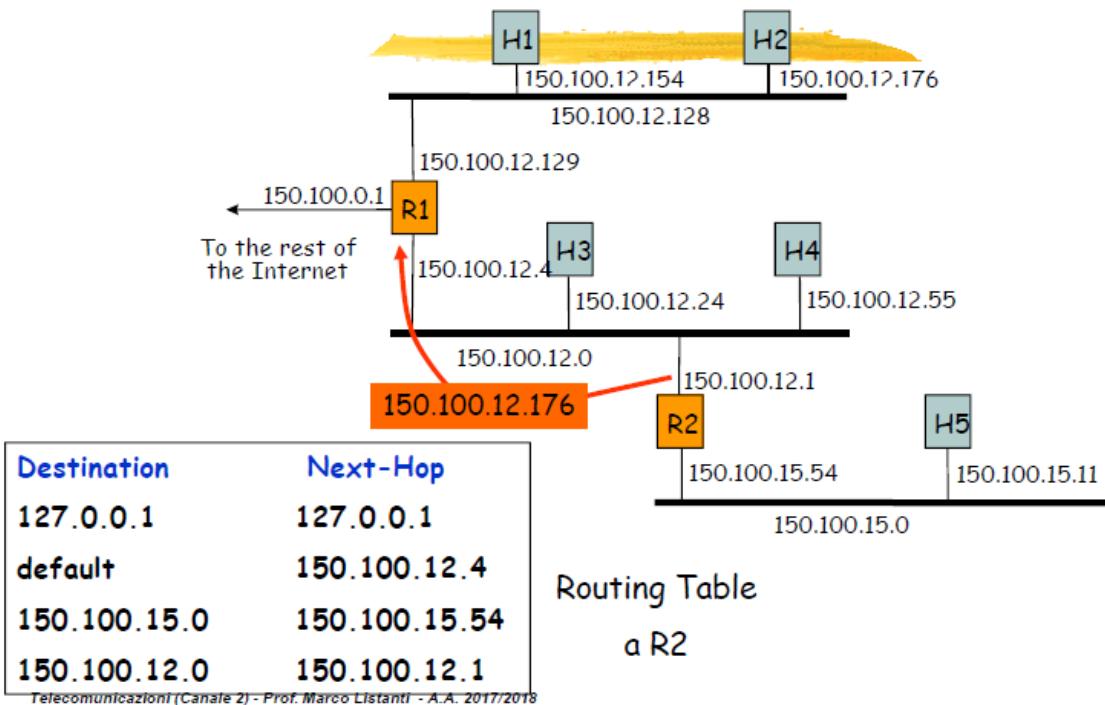
- Destination IP address
- IP address del next-hop router
- Identificatore della porta di uscita
- Informazioni statistiche

- Criteri di ricerca e relative azioni:**

1. Destination address completo
2. Destination Net\_ID (prefisso)
3. Default router
4. Altrimenti "Declare packet undeliverable":  
- Emissione di un pacchetto ICMP "host unreachable error" verso l'host mittente

### Esempio: H5 emette un pacchetto verso H2





### Classless Inter Domain Routing CIDR: Problemi dell'indirizzamento IP

- Nel 1990, sono apparsi chiari due problemi:
  - Gli indirizzi IP erano in via di esaurimento
  - Le tabelle di routing stavano crescendo di dimensione
- Esaurimento degli indirizzi IP:
  - La struttura Classful era inefficiente:
    - Indirizzi di Classe B troppo grandi per la maggior parte delle organizzazioni
    - Indirizzi di Classe C troppo piccoli
    - Con la frequenza di allocazione di indirizzi di Classe B se ne prevedeva l'esaurimento entro il 1994
- Dimensione delle IP routing table:
  - La crescita del numero di reti IP si rifletteva nella crescita del numero di entry delle tabelle di routing:
    - Dal 1991 al 1995, la dimensione delle routing table raddoppiava ogni 10 mesi
    - Aumento del tempo di processing e della dimensione dell'impegno di memoria
- Soluzione Short-term:
  - Classless Interdomain Routing (CIDR), RFC 1518
  - New allocation policy (RFC 2050)
  - Uso di indirizzi Privati per le Intranet
- Long-term solution:
  - Aumento dello spazio di indirizzamento (IPv6, indirizzi a 128 bit)

### Classless Inter Domain Routing (CIDR)

- CIDR è stato ideato per:
  - Rendere più efficiente l'impiego dello spazio di indirizzamento di IP
  - Diminuire la complessità delle tabelle di instradamento nei router
- Ad una rete è assegnato un certo numero di blocchi contigui di indirizzi (**Supernetting**):
  - La rete sarà caratterizzata da un unico **prefisso** (insieme dei bit più significativi)
  - La rete sarà individuata nei router solo dal suo **prefisso**
- Un insieme di reti caratterizzato da blocchi di indirizzi contigui sarà identificato da un unico prefisso

### Address Allocation Policy

- Indirizzi di Classe A e B sono assegnati solo in caso di dimostrata necessità
- Sono assegnati blocchi consecutivi di classe C (fino a 64 blocchi):
  - Tutti gli IP addresses hanno un common **prefix**
  - La lunghezza del prefisso può essere arbitraria
- La metà inferiore degli indirizzi di classe C è assegnata su base geografica

| Address Requirement | Address Allocation |
|---------------------|--------------------|
| < 256               | 1 Class C          |
| 256 < , < 512       | 2 Class C          |
| 512 < , < 1024      | 4 Class C          |
| 1024 < , < 2048     | 8 Class C          |
| 2048 < , < 4096     | 16 Class C         |
| 4096 < , < 8192     | 32 Class C         |
| 8192 < , < 16384    | 64 Class C         |

### CIDR

- Pianificazione geografica degli indirizzi di classe C:

|                       |         |             |
|-----------------------|---------|-------------|
| Multiregional         | 192.0.0 | 193.255.255 |
| Europe                | 194.0.0 | 195.255.255 |
| Others                | 196.0.0 | 197.255.255 |
| North America         | 198.0.0 | 199.255.255 |
| Central/South America | 200.0.0 | 201.255.255 |
| Pacific Rim           | 202.0.0 | 203.255.255 |
| Others                | 204.0.0 | 205.255.255 |
| Others                | 206.0.0 | 207.255.255 |

- Tutte le reti appartenenti ad una regione geografica sono identificate dagli stessi 7 bit di prefisso:

- Esempio: Europa
  - da 194 = 11000010 0 a 195 = 11000011 1

### Supernetting

- Esempio: 150.158.16.0/20

- IP Address (150.158.16.0); lunghezza della maschera (20)
  - IP add = 10010110 10011110 00010000 00000000
  - Mask = 11111111 11111111 11110000 00000000
  - Contiene 16 blocchi di Classe C
  - Da **10010110 10011110 00010000 00000000**
    - 150.158.16.0
  - Fino a **10010110 10011110 00011111 00000000**
    - 150.158.31.0

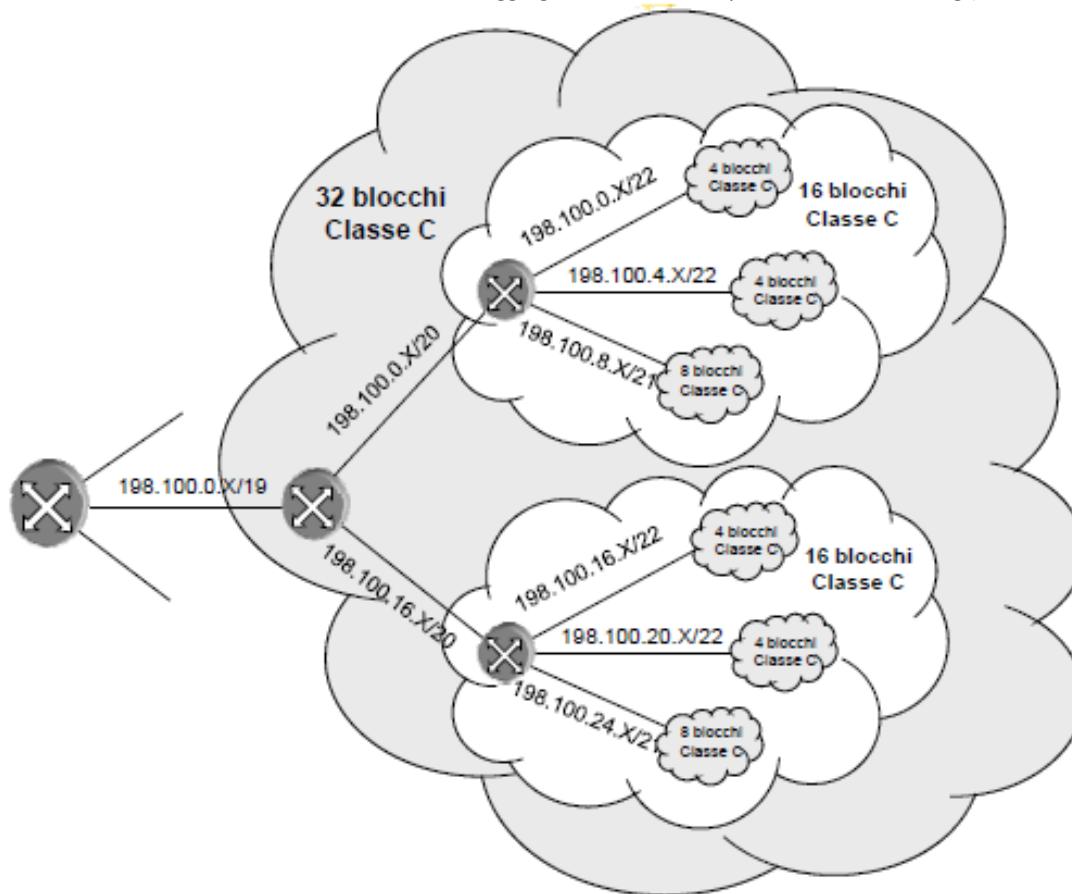
### Classless Inter-Domain Routing

- Il CIDR rallenta la crescita della dimensione delle Routing Table:
  - Una rete è rappresentata da un prefisso e da una maschera
  - **Pre-CIDR**: Una rete con 16 blocchi di classe C contigui richiedeva 16 entry
  - **Post-CIDR**: Una rete con 16 blocchi di classe C contigui richiedeva 1 entry
- L'instradamento è effettuato in base al prefisso:
  - Un entry di una Routing table entry contiene <IP address, network mask>
  - Esempio: 192.32.136.0/21

|          |          |          |                      |
|----------|----------|----------|----------------------|
| 11000000 | 00100000 | 10001000 | 00000001 min address |
| 11111111 | 11111111 | 11111--- | ----- mask           |
| 11000000 | 00100000 | 10001--- | ----- IP prefix      |
| 11000000 | 00100000 | 10001111 | 11111110 max address |
| 11111111 | 11111111 | 11111--- | ----- mask           |
| 11000000 | 00100000 | 10001--- | ----- same IP prefix |

### CIDR Allocation Principles (RFC 1518-1520)

- L'assegnazione degli IP address riflette la topologia fisica della rete
- La topologia di rete segue i confini continentali e nazionali:
  - Gli indirizzi IP devono essere assegnati su questa base
- I domini di transito (TRD) hanno un prefisso IP unico:
  - Trasportano traffico tra domini terminali
  - La maggior parte dei domini terminali sono single-homed: connessi ad un solo TRD
  - A tali domini sono assegnati indirizzi con lo stesso prefisso del TRD
  - Tutte le reti connesse ad un TRD sono aggregate in un solo entry delle tabelle di routing (BGPv4, RFC 1520)



### Longest Prefix Matching

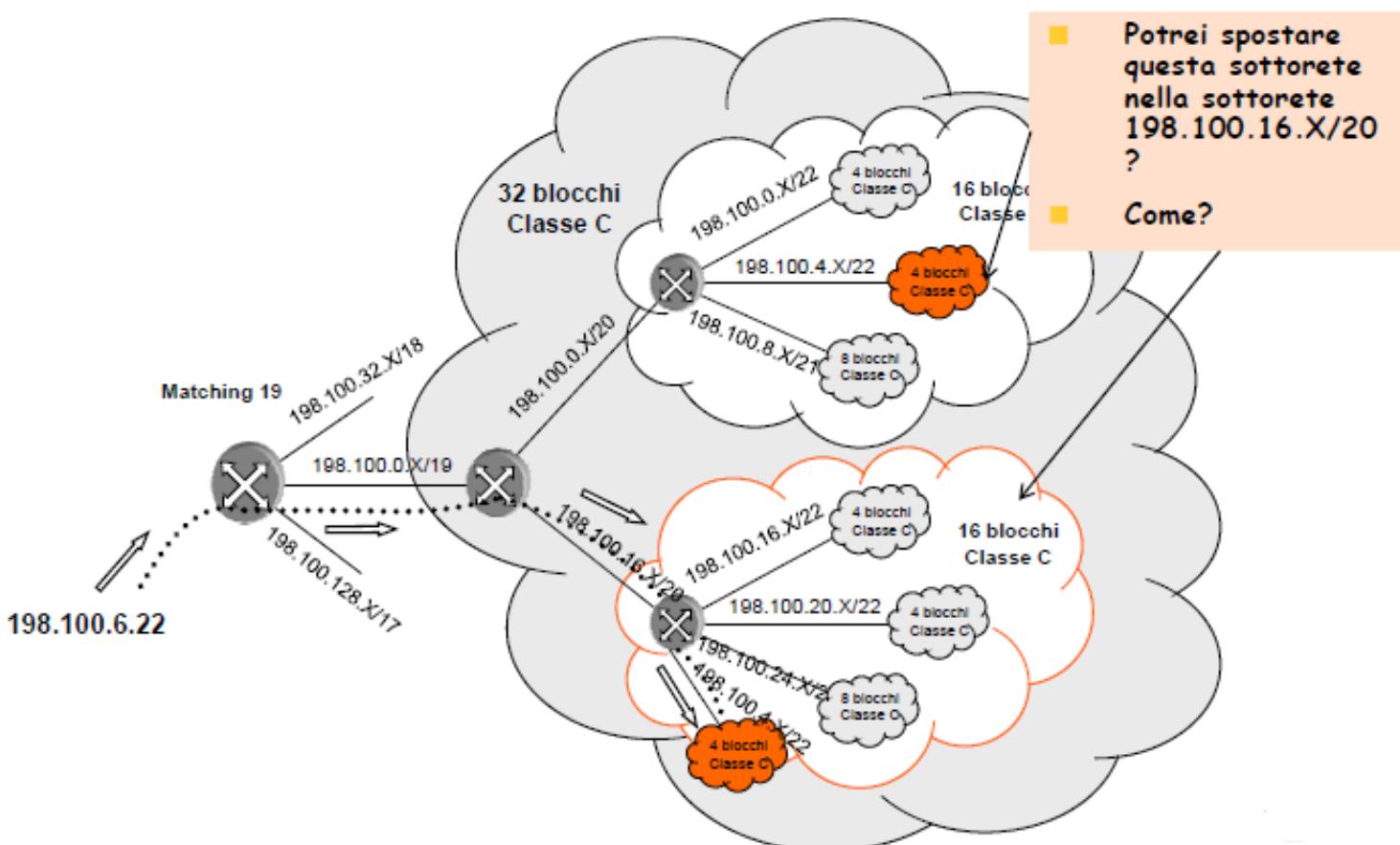
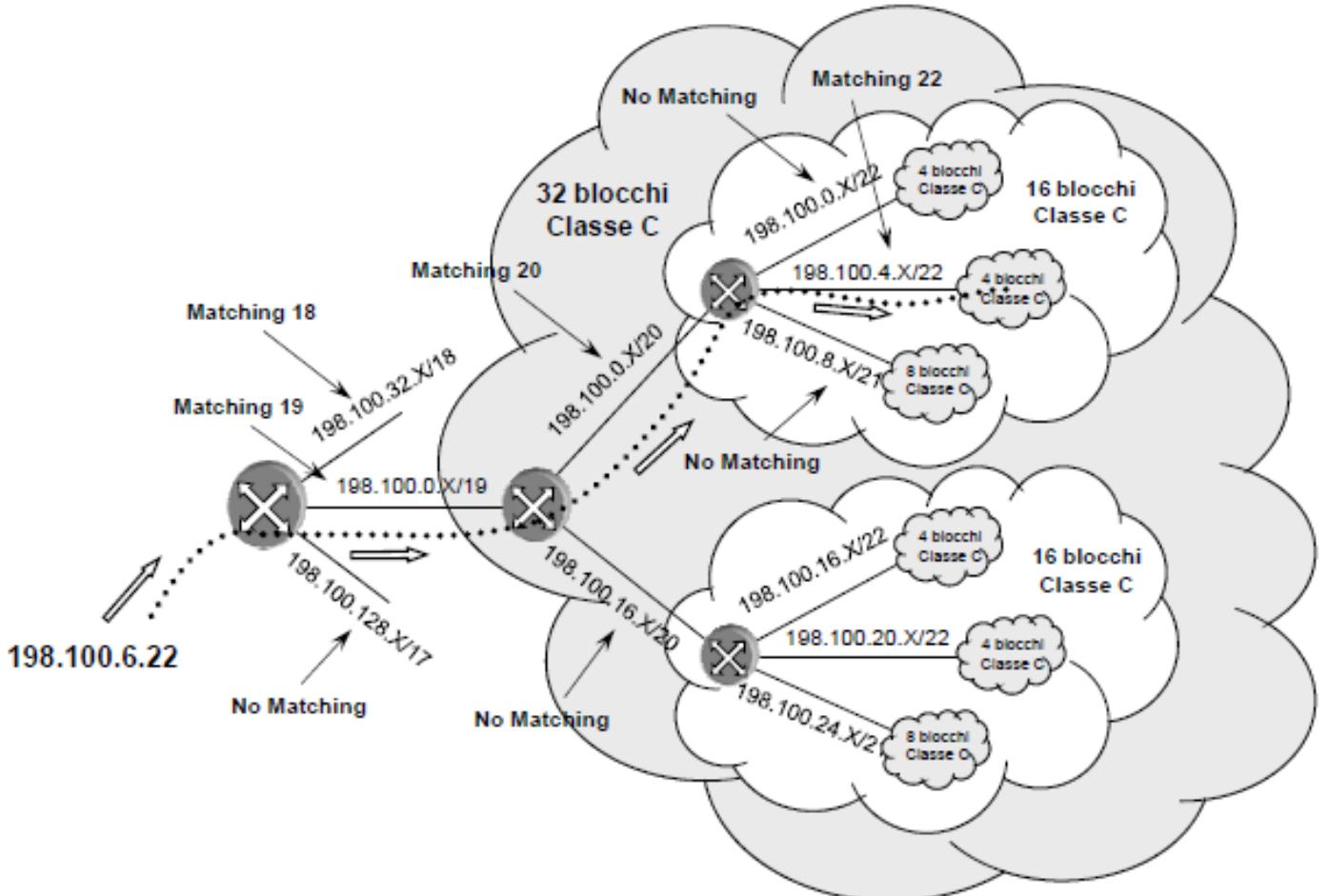
- In una routing table una Super rete può essere rappresentata da un unico elemento corrispondente al suo prefisso
- Per ogni pacchetto entrante, un router sceglie l'instradamento verso la direzione corrispondente al **prefisso di lunghezza maggiore**
- **Instradamento:**
  - Indirizzo 198.15.7.3
  - Indirizzo 198.15.7.4
- 198.15.7.3:
  - porta 1: matching prefisso 16
  - porta 7: matching prefisso 24
  - porta 4: matching prefisso 32
- 198.15.7.4:
  - porta 1: matching prefisso 16
  - porta 7: matching prefisso 24 porta 4: no matching

| Prefix        | Porta d'uscita |
|---------------|----------------|
| 198.15.0.0/16 | 1              |
| 198.15.7.0/24 | 7              |
| 198.15.7.3/32 | 4              |

198.15.7.3  $\Rightarrow$  porta 4

198.15.7.4  $\Rightarrow$  porta 7

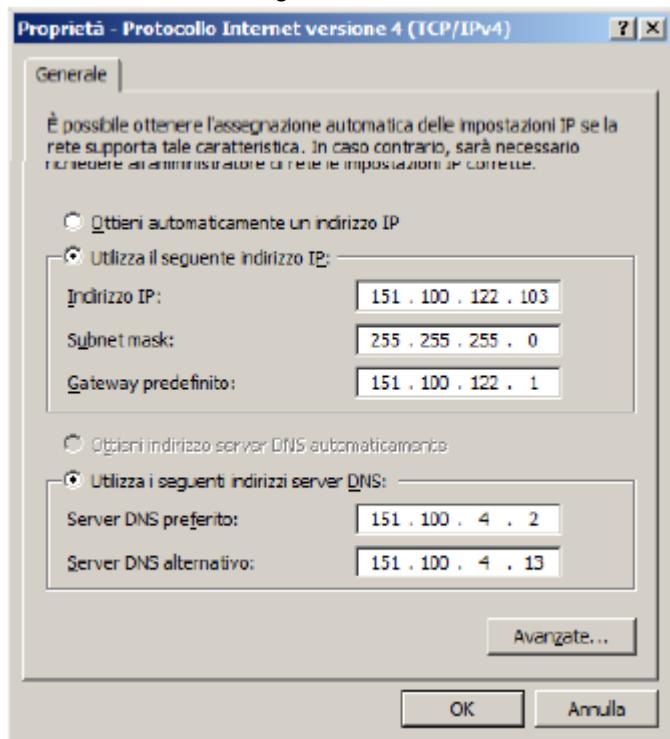
## Longest Prefix Matching



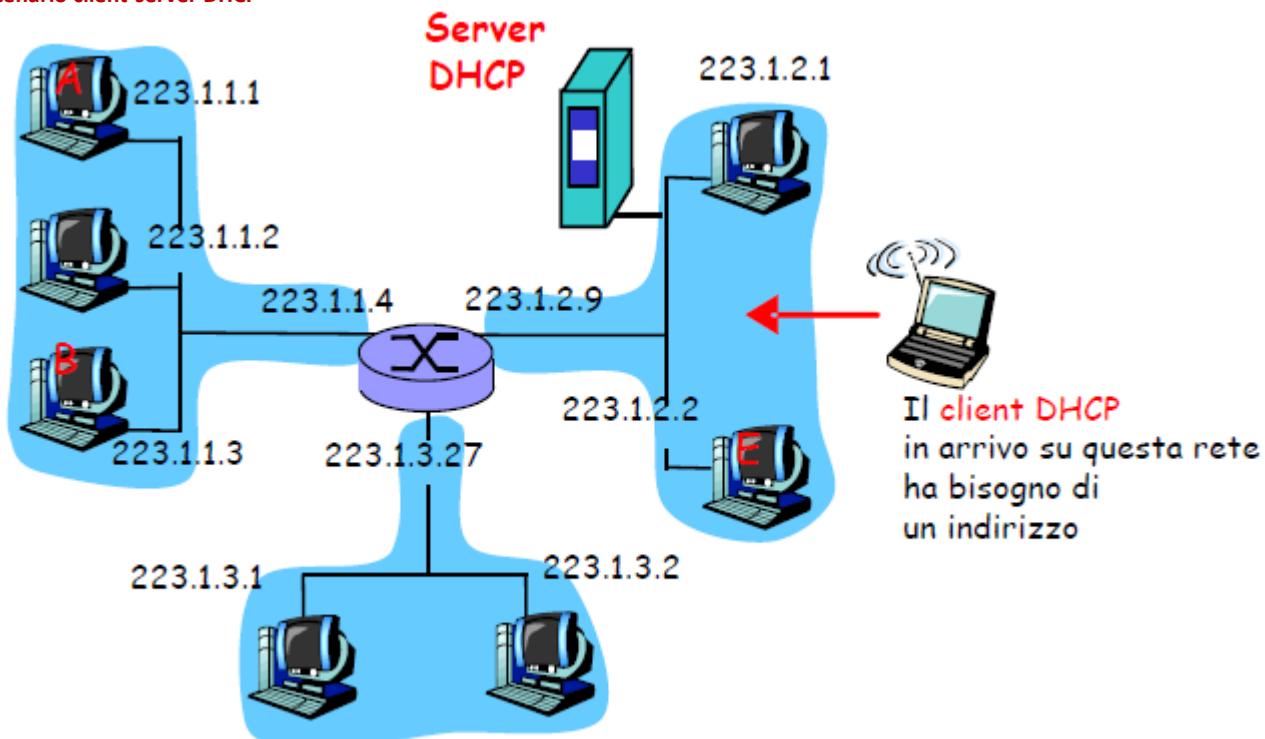
## "DHCP" e "NAT"

### Autoconfigurazione Protocollo DHCP

- Un host deve essere configurato:
  - IP address
  - Subnet mask
  - Default router
  - Server DNS
- Procedura manuale
- Necessità di procedure di autoconfigurazione:
  - DHCP (Dynamic Host Configuration Protocol)
  - Plug and play
  - Uso efficiente degli indirizzi



### Scenario client-server DHCP



## DHCP: Dynamic Host Configuration Protocol

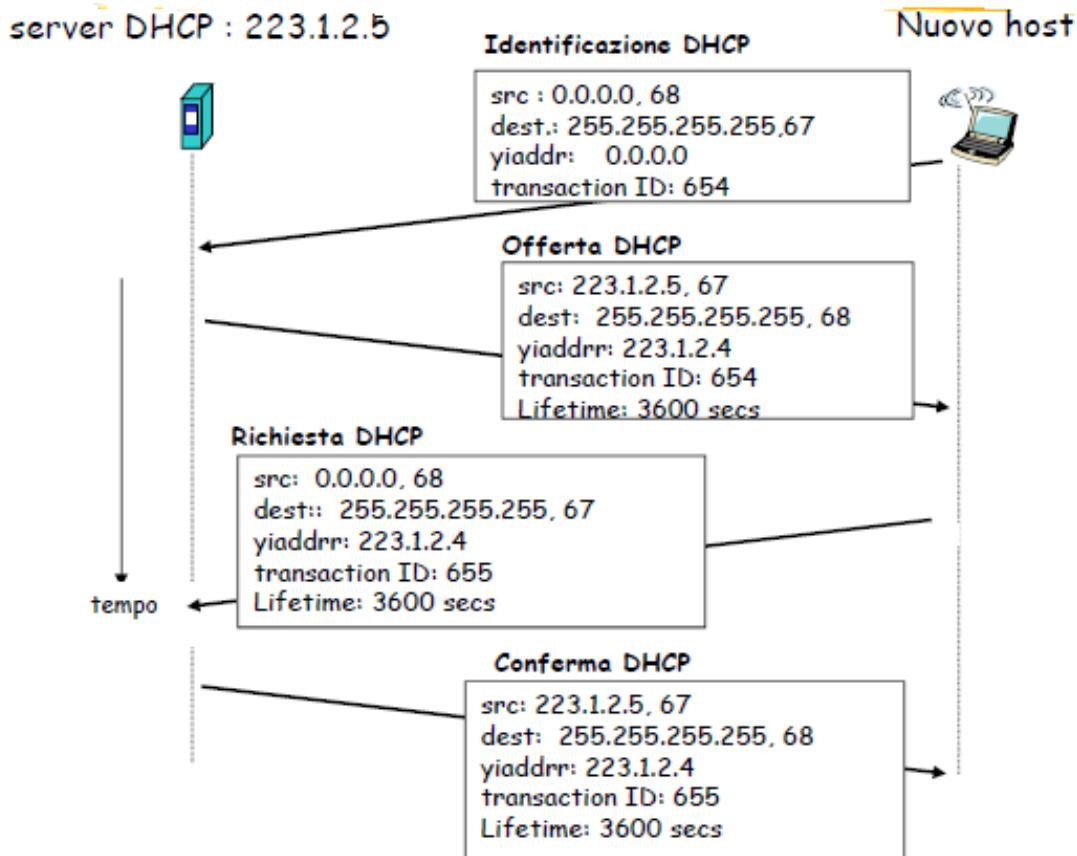
- Consente ad un host di ottenere dinamicamente il suo indirizzo IP dal server di rete:
  - È possibile rinnovare la proprietà dell'indirizzo in uso
  - È possibile il riuso degli indirizzi
  - Supporta anche gli utenti mobili che si vogliono unire alla rete
- Panoramica di DHCP:
  - L'host invia un messaggio broadcasts "DHCP discover"
  - Il server DHCP invia l'indirizzo con il messaggio "DHCP offer"
  - L'host richiede la configurazione con il messaggio "DHCP request"
  - Il server DHCP invia la configurazione con il messaggio "DHCP ack"
- Supporta tre meccanismi per a gestione degli indirizzi IP:
  - Allocazione automatica:
    - DHCP assegna permanentemente un indirizzo IP
  - Allocazione dinamica:
    - DHCP assegna un indirizzo IP per un intervallo limitato di tempo (lease)
  - Allocazione manuale
    - L'indirizzo IP è assegnato dall'amministratore di rete
- Code:
  - Indica una richiesta o una risposta
- HW type:
  - Tipo di hardware (es. ethernet, IEEE 802)
- Length:
  - Lunghezza del campo client HW address
- Transaction ID:
  - Pacchetti di richiesta e di risposta hanno lo stesso numero
- Seconds:
  - Indica il tempo trascorso dall'avvio della procedura di boot
- Flag:
  - Indica se il pacchetto è unicast o broadcast
- Client IP address:
  - È settato dal client, se il client non conosce il proprio indirizzo il suo valore è 0.0.0.0
- Your IP address:
  - Indirizzo IP del client assegnato dal server
- Server IP address:
  - Indirizzo IP del server
- Client HW address:
  - Indirizzo MAC del client
- Options:
  - Parametri di configurazione addizionali: router di default, subnet mask, domain name server, ecc.

| code                         | HW type | length      | hops |  |  |
|------------------------------|---------|-------------|------|--|--|
| Transaction ID               |         |             |      |  |  |
| Seconds                      |         | Flags field |      |  |  |
| Client IP address            |         |             |      |  |  |
| Your IP address              |         |             |      |  |  |
| Server IP address            |         |             |      |  |  |
| Router IP address            |         |             |      |  |  |
| Client HW address (16 bytes) |         |             |      |  |  |
| Server host name (64 bytes)  |         |             |      |  |  |
| Boot file name (128 bytes)   |         |             |      |  |  |
| Options (3124 bytes)         |         |             |      |  |  |

## Messaggi DHCP

- **DHCP\_Discover:**
  - È emesso in modo broadcast da un client per trovare un DHCP server
- **DHCP\_Offer:**
  - È la risposta di un DHCP server ad un messaggio DHCP\_Discover e assegna l'indirizzo IP richiesto
- **DHCP\_Request:**
  - È emesso da un DHCP client verso un server
  - Richiede i parametri di configurazione ad un server e rifiuta le offerte degli altri in caso di ricezione multipla di messaggi DHCP\_Offer
  - Verifica la consistenza della propria configurazione in caso di cambio di rete o di sistema (es. reboot)
  - Richiede l'estensione temporale dell'uso di un indirizzo (lease extension)
- **DHCP\_Ack:**
  - Riscontro inviato dal DHCP Server al client ad un DHCP\_request
  - Contiene la configurazione richiesta dal client

## Procedura DHCP

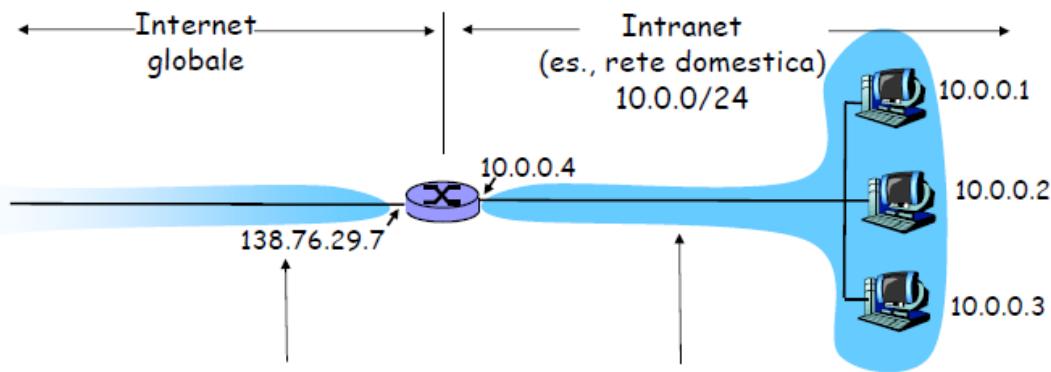


## DHCP

- **Pro:**
  - Semplifica la gestione amministrativa degli indirizzi in rete
  - Semplifica l'accesso in rete di utenti in mobilità (Nomadic Computing)
  - Rende possibile l'uso efficiente di un insieme di indirizzi IP dimensionando opportunamente il tempo di lease
- **Contro:**
  - Non garantisce un vero e proprio "plug and play":
    - Deve essere previsto un server DHCP in rete
    - Gli host devono essere configurati per usare DHCP
  - DHCP non è sicuro, un utente non autorizzato può accedere alla rete
  - Problemi di interoperabilità con DNS in caso di riallocazione dinamica degli indirizzi

## Network Address Translator (NAT)

- Riduce l'utilizzazione dello spazio di indirizzi IP
- È utilizzato in una Intranet:
  - Ad una Intranet è assegnato un insieme di indirizzi IP pubblici che sono visibili dalla rete esterne
  - All'interno della Intranet possono essere utilizzati liberamente **indirizzi IP privati**, anche non unici in rete, appartenenti alla seguenti classi:
    - Indirizzi di classe A: 10.0.0.0
    - Indirizzi di classe B: da 172.16.0.0 a 172.31.0.0
    - Indirizzi di classe C: da 192.168.0.0 a 192.168.255.0
- **Il dispositivo NAT:**
  - Assegna un indirizzo pubblico ad un host solo nel momento che questi deve comunicare con l'esterno
  - Esegue la traslazione dell'indirizzo privato con un indirizzo pubblico



I router abilitati alla funzione NAT appaiono come un *unico* dispositivo con un *unico* indirizzo IP.

Indirizzo IP origine: 138.76.29.7,  
e tutto il traffico verso Internet deve riportare lo stesso indirizzo

Spazio di indirizzi riservato alle reti private, molte delle quali usano un identico spazio, 10.0.0/24 per scambiare pacchetti tra i loro dispositivi

- **Un NAT nasconde i dettagli di una Intranet al mondo esterno:**

- Non è necessario allocare un intervallo di indirizzi
- Un unico indirizzo IP è sufficiente per tutti gli host di una rete locale
- È possibile cambiare gli indirizzi delle macchine di una rete privata senza doverlo comunicare all'Internet globale
- È possibile cambiare ISP senza modificare gli indirizzi delle macchine della rete privata
- Dispositivi interni alla rete non esplicitamente indirizzabili e visibili dal mondo esterno (un plus per la sicurezza)

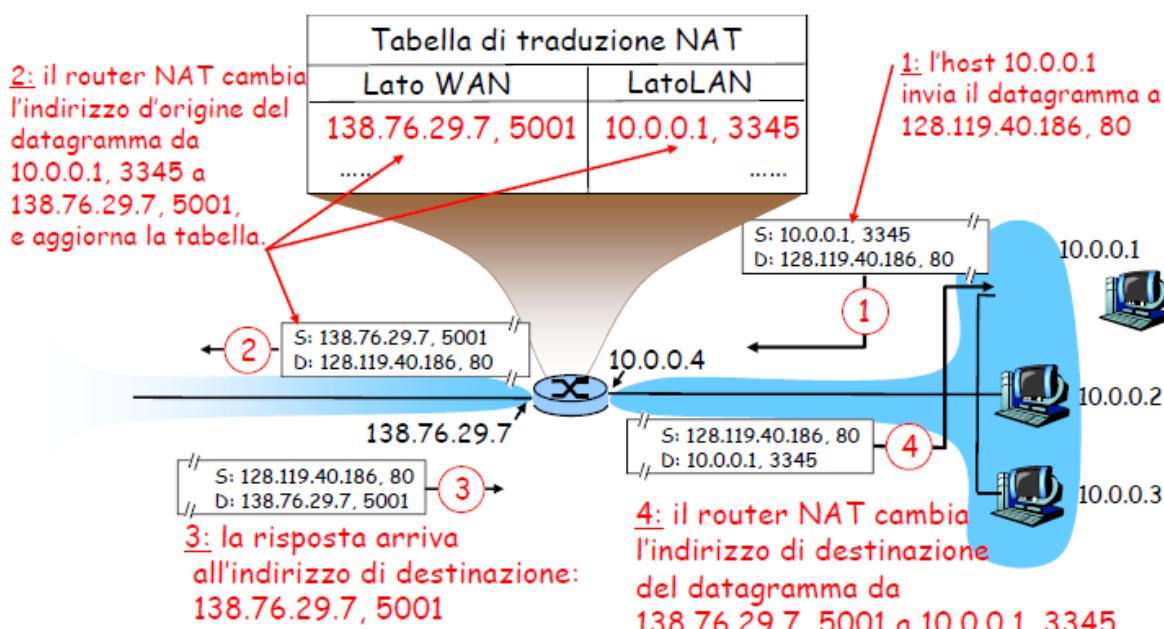
- **Quando un router NAT riceve il pacchetto dalla rete locale:**

- Genera un nuovo numero di porta d'origine (es. 5001)
- Sostituisce l'indirizzo IP di sorgente (privato) con il proprio indirizzo IP (pubblico) sul lato WAN (es. 138.76.29.7)
- Sostituisce il numero di porta origine iniziale (es. 3348) con il nuovo numero (5001)

- **Quando un router NAT riceve il pacchetto da Internet:**

- Legge il numero di porta (es. 5001) ed individua il mapping con l'indirizzo interno
- Sostituisce l'indirizzo IP di destinazione con l'indirizzo IP privato dell'host di destinazione
- Sostituisce il numero di porta di destinazione (5001) con il numero di porta iniziale (3348)

### Traduzione degli indirizzi di rete



- **Il campo numero di porta è lungo 16 bit:**

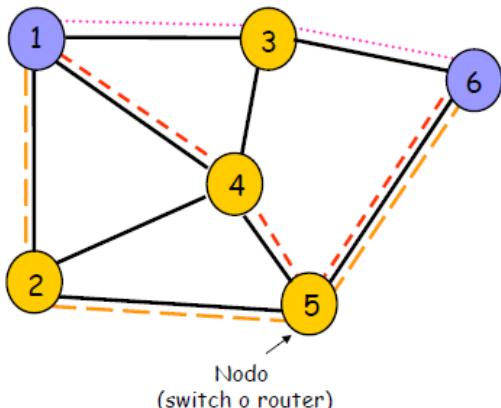
- Il protocollo NAT può supportare più di 60.000 connessioni simultanee con un solo indirizzo IP sul lato WAN

- **NAT è contestato perché:**

- È contrario ai principi dell'architettura a strati TCP/IP:
  - I dispositivi di rete dovrebbero elaborare i pacchetti fino al livello 3
- Un host non è visibile dall'esterno della rete a cui appartiene
- Interferisce con le applicazioni P2P
- Al momento di un cambio di indirizzo IP deve essere ricalcolato il checksum dei pacchetti UDP e TCP
- Incompatibilità con il protocollo ICMP

## "Routing in reti IP"

### Instradamento in reti a pacchetto



- Tre possibili (loopfree) cammini dal nodo 1 al nodo 6:
  - 1-3-6, 1-4-5-6, 1-2-5-6
- Qual è il cammino migliore?
  - Minimo ritardo Minimo numero di hop Minimo costo Massima affidabilità

### Creazione delle tabelle di routing

- È necessario definire la tipologia di informazioni sullo stato dei link:
  - Link up/down; stato di congestione; delay o altre metriche
- Occorre distribuire le informazioni di stato dei link usando un protocollo di routing:
  - Quali informazioni devono essere scambiate?
  - Con quale frequenza?
  - Scambio di informazioni con i vicini, broadcast, flooding
- Occorre calcolare i cammini migliori:
  - Algoritmo di instradamento
  - Metriche singole o multiple

### Requisiti

- Risposta alle variazioni di stato:
  - Variazioni di topologia o banda dei link
  - Stato di congestione
  - Rapida convergenza
  - Assenza di loop
- Ottimalità:
  - Utilizzazione ottima delle risorse di rete
  - Minimizzazione della lunghezza dei cammini
- Robustezza:
  - Continuità di servizio in presenza di condizioni anomale (alto carico, congestione di rete, guasti, errori di implemetazione)
- Semplicità:
  - Basso carico di elaborazione

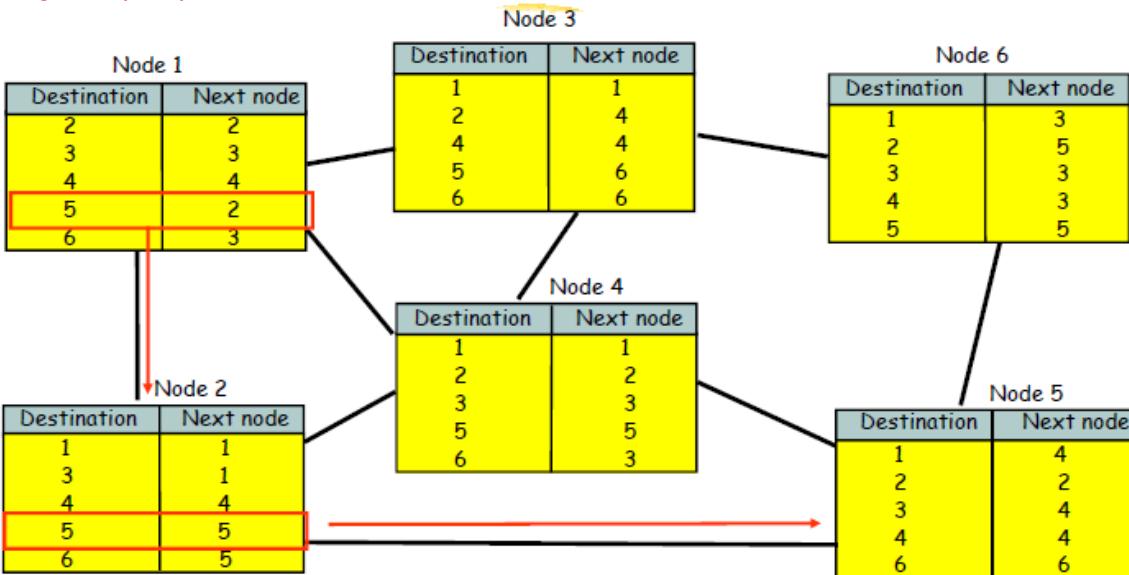
### Instradamento Centralizzato o Distribuito

- Routing Centralizzato:
  - I cammini sono determinati da un elemento (nodo) centralizzato
  - Le informazioni di stato sono inviate al nodo centrale
  - Difficili adattamenti ai cambi di topologia
  - Soluzione non scalabile e di scarsa affidabilità
- Routing Distribuito:
  - I router determinano i cammini usando un algoritmo distribuito
  - Le informazioni di stato sono scambiate tra i router
  - Maggiore adattabilità alle variazioni di stato della rete
  - Alta scalabilità

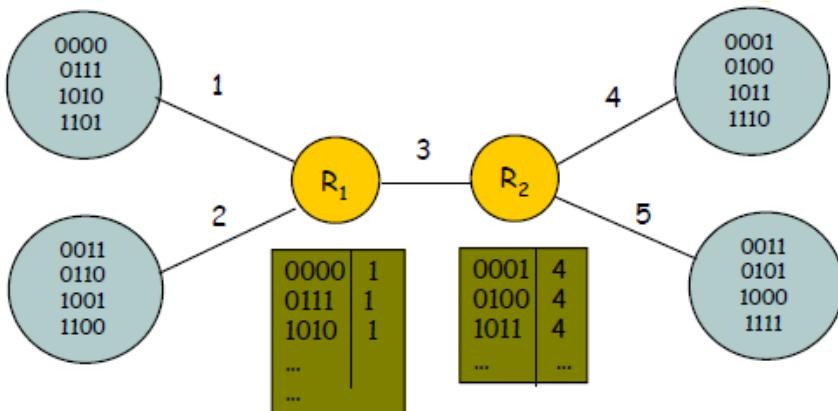
### Instradamento Statico o Dinamico

- Instradamento Statico:
  - Cammini configurati manualmente, non variano nel tempo
  - Adatto al caso di reti semplici con traffico predibile
  - Usato per impostare alcuni cammini particolari
  - Usato per fornire un instradamento di default (default router)
- Instradamento Dinamico:
  - Adatto a sostenere variazioni dello stato della rete
  - Calcolo automatico dei cammini
  - Cammini determinati in base alle informazioni di stato della rete ricevute per mezzo di un protocollo di instradamento

## Routing Table: principio di funzionamento

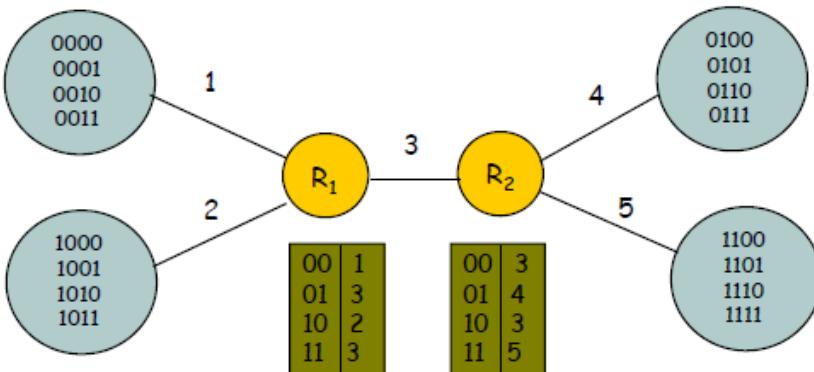


## Indirizzamento e instradamento non gerarchici



- Nessuna relazione tra indirizzi e localizzazione geografica (vicinanza) delle destinazioni
- Routing table composta da 16 record ciascuna:
  - Possibilità di routing table explosion

## Indirizzamento e instradamento gerarchici



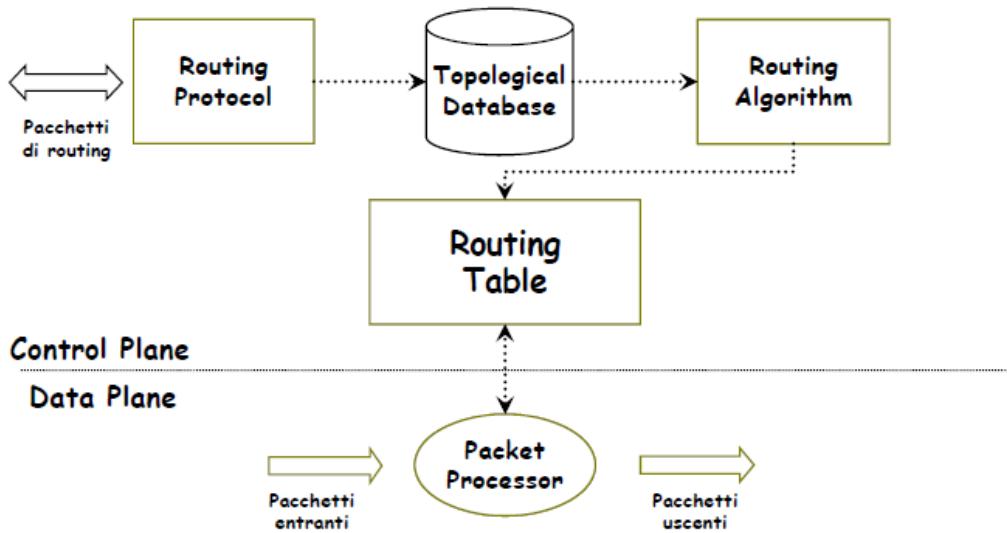
- I prefissi indicano la rete a cui un host è connesso
- Reti con lo stesso prefisso sono “vicine”
- Routing table composta da 4 record ciascuna

## Instradamento Flat o Gerarchico

- **Routing Flat:**
  - Tutti i router sono allo stesso livello (peer)
  - Scarsa scalabilità
- **Routing Gerarchico:**
  - Suddivisione della rete: Domini, sistemi autonomi, aree...
  - Alcuni router fanno parte del backbone della rete
  - Alcuni router comunicano solo con router della stessa area
  - Soluzione efficiente (ricalca le relazioni di traffico)
  - Soluzione scalabile

## Instradamento in reti IP

- La scelta del router verso cui inviare il pacchetto avviene utilizzando la Tabella di Instradamento (Routing Table - RT) contenuta in ogni host e in ogni router
- **Ogni elemento di una RT contiene:**
  - Indirizzo IP di destinazione (host address o network address)
  - Indirizzo del router successivo (next hop router) sul cammino verso la rete di destinazione
  - Indicazione dell'interfaccia fisica di uscita
- Un router non conosce il cammino completo verso la destinazione
- **Un router esegue i seguenti passi:**
  - Estrae dal pacchetto entrante il contenuto del campo Destination Address
  - Ricerca all'interno della RT il record che contiene il "longest prefix matching" con il DA del pacchetto entrante
  - In caso di fallimento del passo precedente, ricerca l'indirizzo del "router di default"
  - Se nessuno dei passi precedenti da esito positivo, il pacchetto è classificato come "undeliverable" ed è scartato ed inviato un messaggio ICMP all'host sorgente
- Un router possiede un Database Topologico in cui sono memorizzate le informazioni sulla topologia della rete:
  - Le informazioni sulla topologia di rete sono aggiornate dai messaggi del protocollo di routing
- L'algoritmo di routing, sulla base delle informazioni contenute nel Database Topologico, determina periodicamente i percorsi a costo minimo tra il router e le possibili reti di destinazione (network prefix)
- La Routing Table è costruita inserendo, per ogni destinazione, sulla base dei risultati del passo precedente, l'informazione relativa al next hop verso cui instradare il pacchetto



## Le Routing Table sono dinamiche:

- Ogni router ed ogni host aggiornano nel tempo le informazioni relative alla topologia di rete

## L'aggiornamento dinamico è necessario perché:

- Internet non può essere considerata stabile
- In caso di guasti alcuni cammini non sono utilizzabili
- È consigliabile scegliere il cammino in base allo stato di occupazione delle risorse di rete

## Le RT devono essere aggiornate continuamente (anche ad intervalli di pochi secondi)

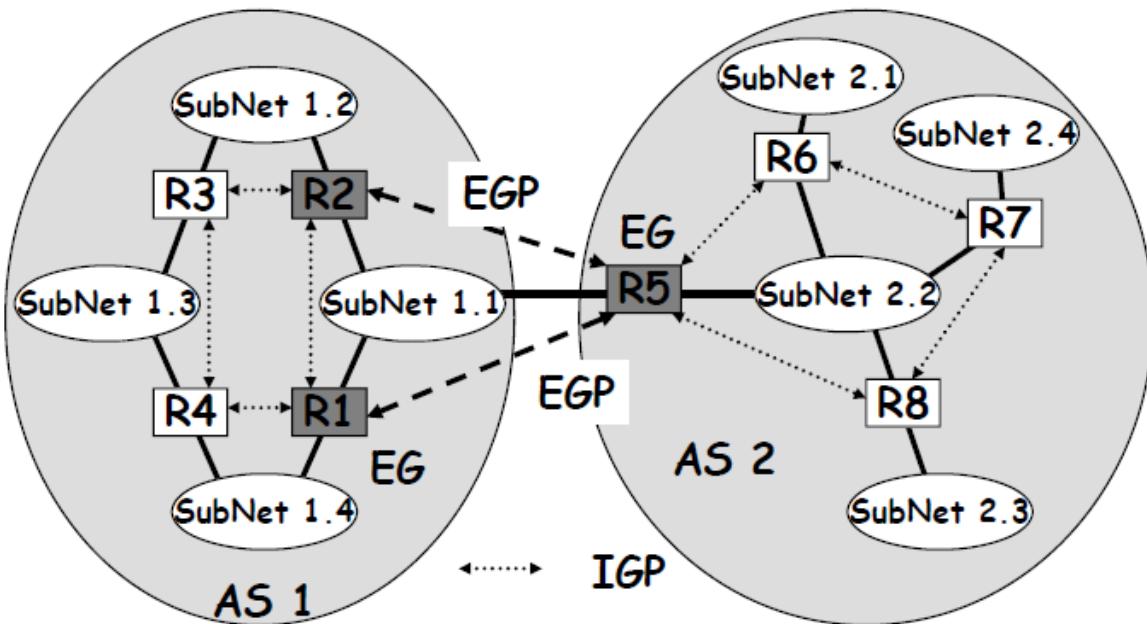
## L'aggiornamento delle RT è attuato mediante protocolli di colloquio tra i router (Routing Protocol)

## Sistemi autonomi

- Un sistema autonomo (Autonomous System - AS) è un insieme di host e router controllato da una singola autorità amministrativa (es. ISP):
  - Un particolare AS è detto "Core AS" e costituisce il backbone di Internet
  - Un router del core AS è detto Core Router
  - Gli altri AS sono detti "Stub AS"
- Ogni AS ha il proprio protocollo di instradamento
- Uno Stub AS deve aver almeno un router connesso ad un core router; questi router sono detti **Exterior Gateway**
- Un router interno ad un AS è detto **Interior Gateway**

## IGP e EGP

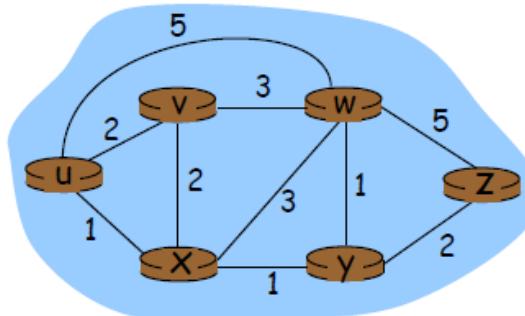
- I protocolli di instradamento all'interno di un AS sono detti **Interior Gateway Protocols (IGP)**
- Le informazioni di instradamento che coinvolgono più di un sistema autonomo sono gestite mediante gli **Exterior Gateway Protocols (EGP)**
- Le informazioni di instradamento degli EGP vengono inviate agli Exterior Gateway di ogni sistema autonomo
- L'instradamento all'interno di un sistema autonomo e la raccolta di dati da inviare ai core router avviene per mezzo degli IGP
- **Un EGP svolge tre funzioni:**
  - Individuazione dei router adiacenti con cui scambiare le informazioni di instradamento
  - Verifica continua della funzionalità dei router interlocutori
  - Scambio periodico delle informazioni di instradamento, queste riguardano la sola raggiungibilità delle reti, non la distanza



### Algoritmi di instradamento: Generalità

Modello a grafo di una rete

Grafo Pesato  
 $G = (N, E, c)$



- $N$  = insieme di nodi (router) = { u, v, w, x, y, z }
- $E$  = insieme di archi (collegamenti) = { (u,v), (u,x), (v,x), (v,w), (x,w), (x,y), (w,y), (w,z), (y,z) }
- $c$  = insieme dei costi associati ai rami:  
 -  $c(x,x')$  = costo associato ramo  $(x,x')$

#### Costo di un cammino

Il costo di un cammino è definito dalla somma di tutti i costi degli archi lungo il cammino

$$\text{Costo di un cammino } (x_1, x_2, x_3, \dots, x_p) = c(x_1, x_2) + c(x_2, x_3) + \dots + c(x_{p-1}, x_p)$$

- Il protocollo di instradamento mette in grado ogni router di determinare il modello a grafo della rete
- L'algoritmo di instradamento determina il cammino a costo minimo tra due nodi della rete

#### Metriche

- Misurano la "qualità" di un link o di un cammino:
  - Costo basso: link ad alta qualità (es. banda elevata), da includere se possibile nei cammini
  - Costo elevato: link di bassa qualità (es. banda limitata), da escludere se possibile nei cammini
- Lunghezza di un cammino (Path Length) = somma dei costi dei link componenti (Distanza)
- Possibili metriche:
  - Numero di hop: misura approssimata delle risorse utilizzate
  - Affidabilità: grado di disponibilità del cammino;
  - BER Ritardo: somma dei ritardi lungo il path
  - Bandwidth: capacità disponibile lungo un path
  - Carico: Grado di utilizzazione dei link e dei router lungo il path

#### Approcci Shortest Path

- Distance Vector Protocol:
  - Nodi adiacenti si scambiano la lista delle distanze verso le destinazioni
  - Viene determinato il next-hop migliore per ogni destinazione
  - Algoritmo di Bellman-Ford
- Link State Protocol:
  - Le informazioni sullo stato dei link (costi) sono diffuse in rete (flooding)
  - I router conoscono l'intera topologia della rete
  - Ogni router calcola lo shortest path ed il next-hop verso ogni destinazione
  - Algoritmo di Dijkstra

## Algoritmo di Bellman-Ford

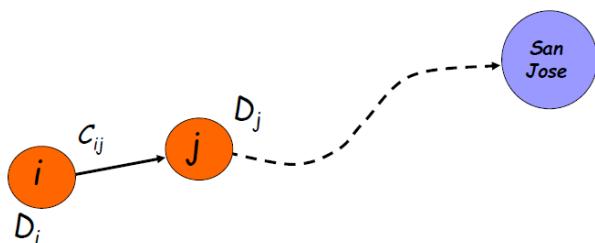
### Distance Vector Protocol

- **Routing Table:**
  - Per ogni destinazione sono memorizzati
  - Next Hop
  - Distanza (costo del cammino minimo)
- Router vicini si scambiano i **Distance Vector**:
  - DV = (destinazione, distanza)
  - Periodicamente
  - Dopo un cambio di stato
- **Ogni nodo determina per ogni destinazione il next- hop migliore**

| Dest | Next | Dist |
|------|------|------|
|      |      |      |
|      |      |      |
|      |      |      |
|      |      |      |
|      |      |      |
|      |      |      |

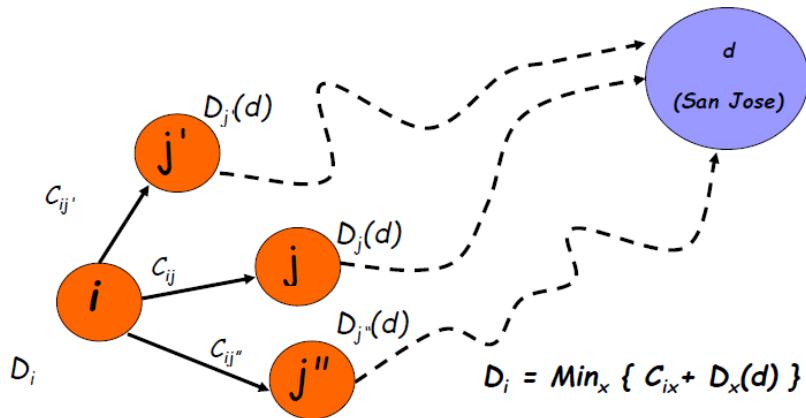
### Calcolo dei cammini minimi

- **Obiettivo:**
  - Calcolo del percorso minimo tra il nodo  $i$  ed un nodo di destinazione (es. SJ)



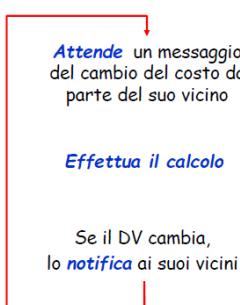
- Se  $D_i$  è la distanza minima dal nodo  $j$  e SJ e se  $j$  è il nodo adiacente a  $i$  che si trova sul percorso a costo minimo dal nodo  $i$  verso SJ, si ha  

$$D_i = C_{ij} + D_j$$
- Il nodo  $i$ :
  - riceve le informazioni dai nodi vicini:  $D_x(d)$
  - conosce i costi dei rami verso i vicini ( $C_{ix}$ )



### Algoritmo con vettore distanza

- **Iterativo, asincrono**
- **Ogni iterazione locale è causata da:**
  - Cambio del costo di uno dei collegamenti locali
  - Ricezione da qualche vicino di un vettore distanza aggiornato
- **Distribuito:**
  - Ogni nodo aggiorna i suoi vicini solo quando il suo DV cambia
  - I vicini avvisano i vicini solo se necessario
- **Ciascun nodo:**



### Algoritmo di Bellman-Ford

- Consideriamo il calcolo parallelo per tutte le destinazioni d
- Inizializzazione:
  - Ogni nodo ha 1 riga per ogni destinazione d
  - La distanza del nodo d a se stesso è posta a zero:  $D_d(d)=0$
  - La distanza del nodo d verso un altro nodo j è posta uguale ad infinito:
 
$$D_j(d) = \infty, \text{ for } j \neq d$$

#### • Passo di emissione:

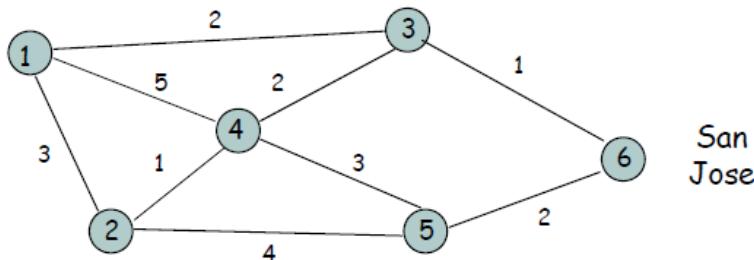
- Il nodo emette il nuovo distance vector verso i nodi vicini

#### • Passo di ricezione:

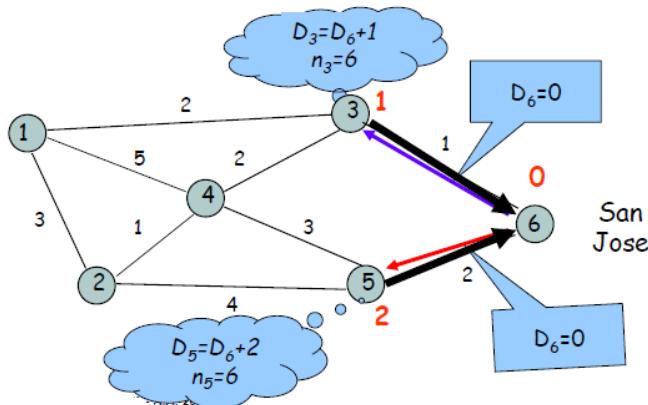
- Per ogni destinazione d, un nodo calcola il next hop che fornisce la minima distanza verso il nodo d,
  - $\min_j \{ C_{ij} + D_j(d) \}$
  - Si sostituisce il vecchio record  $(n_j, D_i(d))$  con il nuovo record  $(n_j^*, D_j^*(d))$

| Iteration | Node 1          | Node 2          | Node 3          | Node 4          | Node 5          |
|-----------|-----------------|-----------------|-----------------|-----------------|-----------------|
| Initial   | (-1, $\infty$ ) |
| 1         |                 |                 |                 |                 |                 |
| 2         |                 |                 |                 |                 |                 |
| 3         |                 |                 |                 |                 |                 |

Record della RT  
 del nodo 1 per la  
 destinazione SJ      Record della RT  
 del nodo 3 per la  
 destinazione SJ

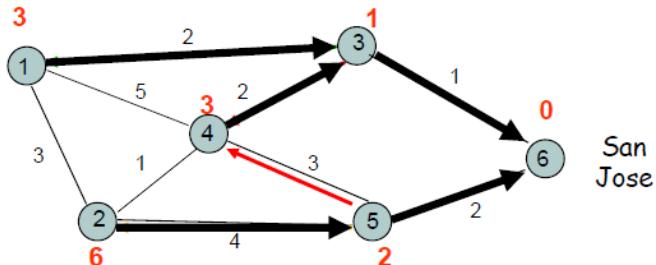


| Iteration | Node 1          | Node 2          | Node 3          | Node 4          | Node 5          |
|-----------|-----------------|-----------------|-----------------|-----------------|-----------------|
| Initial   | (-1, $\infty$ ) |
| 1         | (-1, $\infty$ ) | (-1, $\infty$ ) | (6, 1)          | (-1, $\infty$ ) | (6, 2)          |
| 2         |                 |                 |                 |                 |                 |
| 3         |                 |                 |                 |                 |                 |

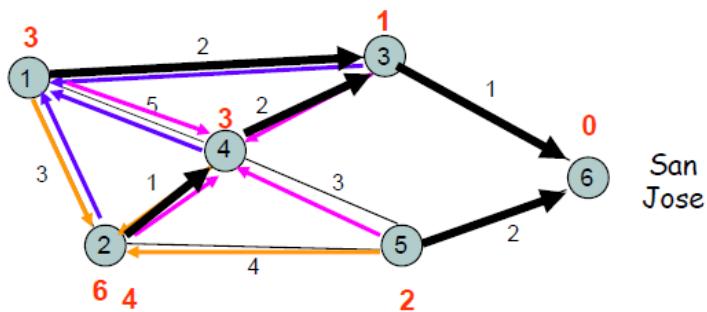




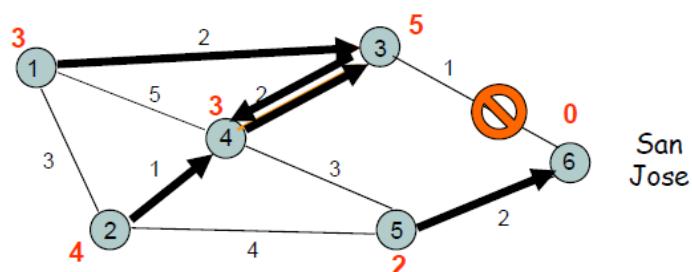
| Iteration | Node 1          | Node 2          | Node 3          | Node 4          | Node 5          |
|-----------|-----------------|-----------------|-----------------|-----------------|-----------------|
| Initial   | (-1, $\infty$ ) |
| 1         | (-1, $\infty$ ) | (-1, $\infty$ ) | (6, 1)          | (-1, $\infty$ ) | (6, 2)          |
| 2         | (3, 3)          | (5, 6)          | (6, 1)          | (3, 3)          | (6, 2)          |
| 3         |                 |                 |                 |                 |                 |



| Iteration | Node 1          | Node 2          | Node 3          | Node 4          | Node 5          |
|-----------|-----------------|-----------------|-----------------|-----------------|-----------------|
| Initial   | (-1, $\infty$ ) |
| 1         | (-1, $\infty$ ) | (-1, $\infty$ ) | (6, 1)          | (-1, $\infty$ ) | (6, 2)          |
| 2         | (3, 3)          | (5, 6)          | (6, 1)          | (3, 3)          | (6, 2)          |
| 3         | (3, 3)          | (4, 4)          | (6, 1)          | (3, 3)          | (6, 2)          |

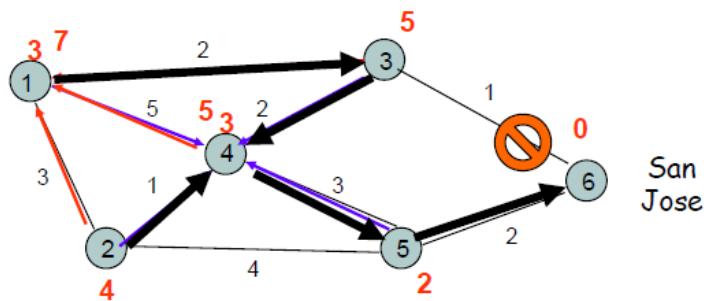


| Iteration | Node 1 | Node 2 | Node 3 | Node 4 | Node 5 |
|-----------|--------|--------|--------|--------|--------|
| Initial   | (3, 3) | (4, 4) | (6, 1) | (3, 3) | (6, 2) |
| 1         | (3, 3) | (4, 4) | (4, 5) | (3, 3) | (6, 2) |
| 2         |        |        |        |        |        |
| 3         |        |        |        |        |        |

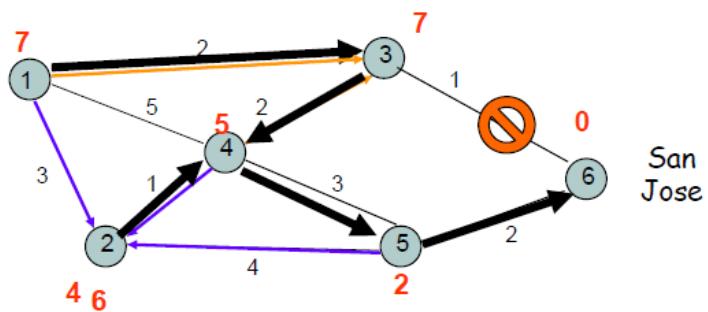


Rete disconnessa: si crea un loop tra i nodi 3 e 4

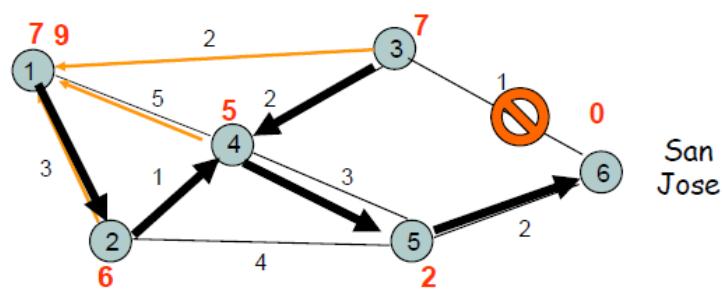
| Iteration | Node 1 | Node 2 | Node 3 | Node 4 | Node 5 |
|-----------|--------|--------|--------|--------|--------|
| Initial   | (3, 3) | (4, 4) | (6, 1) | (3, 3) | (6, 2) |
| 1         | (3, 3) | (4, 4) | (4, 5) | (3, 3) | (6, 2) |
| 2         | (3, 7) | (4, 4) | (4, 5) | (5, 5) | (6, 2) |
| 3         |        |        |        |        |        |



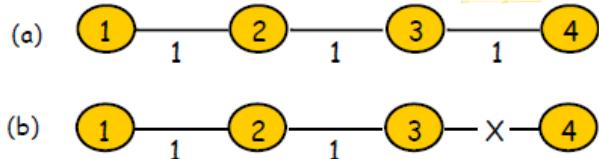
| Iteration | Node 1 | Node 2 | Node 3 | Node 4 | Node 5 |
|-----------|--------|--------|--------|--------|--------|
| Initial   | (3, 3) | (4, 4) | (6, 1) | (3, 3) | (6, 2) |
| 1         | (3, 3) | (4, 4) | (4, 5) | (3, 3) | (6, 2) |
| 2         | (3, 7) | (4, 4) | (4, 5) | (5, 5) | (6, 2) |
| 3         | (3, 7) | (4, 6) | (4, 7) | (5, 5) | (6, 2) |



| Iteration | Node 1 | Node 2 | Node 3 | Node 4 | Node 5 |
|-----------|--------|--------|--------|--------|--------|
| 1         | (3, 3) | (4, 4) | (4, 5) | (3, 3) | (6, 2) |
| 2         | (3, 7) | (4, 4) | (4, 5) | (2, 5) | (6, 2) |
| 3         | (3, 7) | (4, 6) | (4, 7) | (5, 5) | (6, 2) |
| 4         | (2, 9) | (4, 6) | (4, 7) | (5, 5) | (6, 2) |



### Conteggio all'infinito



I nodi credono che il  
esista un cammino in  
realtà non disponibile  
**Destinazione nodo 4**

| Passo            | Nodo 1 | Nodo 2 | Nodo 3 |
|------------------|--------|--------|--------|
| Prima del guasto | (2, 3) | (3, 2) | (4, 1) |
| Dopo il guasto   | (2, 3) | (3, 2) | (2, 3) |
| 1                | (2, 3) | (3, 4) | (2, 3) |
| 2                | (2, 5) | (3, 4) | (2, 5) |
| 3                | (2, 5) | (3, 6) | (2, 5) |
| 4                | (2, 7) | (3, 6) | (2, 7) |
| 5                | (2, 7) | (3, 8) | (2, 7) |
| ...              | ...    | ...    | ...    |

### Soluzioni al conteggio all'infinito

- **Split Horizon:**

- Un router non trasmette il proprio DV aggiornato verso il router da cui ha ricevuto l'aggiornamento

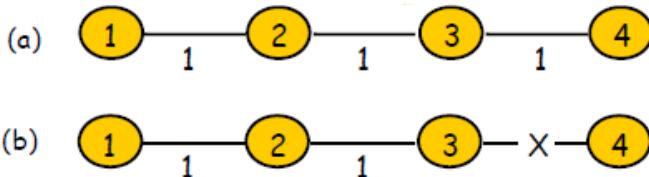
- **Poisoned Reverse:**

- Un router trasmette il proprio DV aggiornato anche verso il router da cui ha ricevuto l'aggiornamento, ma indicando per la distanza aggiornata al valore  $\infty$

- Si interrompe immediatamente il loop di conteggio

- Questa soluzione non funziona in caso di loop più complessi

### Split Horizon con Poison Reverse



**Destinazione nodo 4**

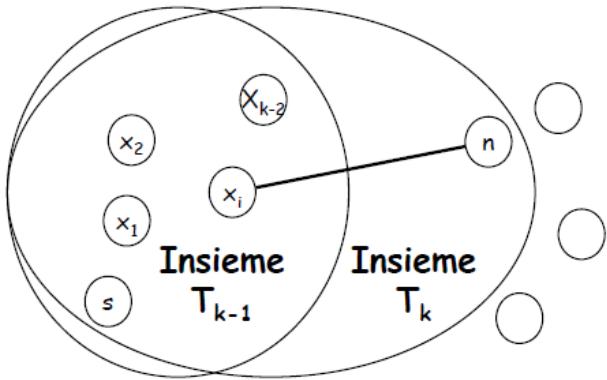
| Update           | Nodo 1          | Nodo 2          | Nodo 3          | Eventi                                                                                                                                        |
|------------------|-----------------|-----------------|-----------------|-----------------------------------------------------------------------------------------------------------------------------------------------|
| Prima del guasto | (2, 3)          | (3, 2)          | (4, 1)          |                                                                                                                                               |
| Dopo il guasto   | (2, 3)          | (3, 2)          | (-1, $\infty$ ) | Nodo 2 annuncia al nodo 3 il suo cammino verso il nodo 4 con distanza infinita;<br>Il nodo 3 deduce che non esiste un cammino verso il nodo 4 |
| 1                | (2, 3)          | (-1, $\infty$ ) | (-1, $\infty$ ) | Nodo 1 annuncia al nodo 2 il suo cammino verso il nodo 4 con distanza infinita<br>Il nodo 2 deduce che non esiste un cammino verso il nodo 4  |
| 2                | (-1, $\infty$ ) | (-1, $\infty$ ) | (-1, $\infty$ ) | Il nodo 1 deduce che non esiste un cammino verso il nodo 4                                                                                    |

### Algoritmo di Djikstra

- Individua il cammino a lunghezza minima tra un nodo s e tutti gli altri nodi di un grafo G procedendo in modo da aumentare progressivamente la distanza

- L'algoritmo procede a passi successivi:

- Al passo k-mo sono individuati i k nodi raggiungibili dal nodo sorgente tramite i cammini a costo più basso
- Tali k nodi formano l'insieme  $T_k$
- Al passo  $k+1$ -mo si individua il nodo n che è caratterizzato dal cammino dal costo più basso dal nodo s che transita esclusivamente nei nodi dell'insieme  $T_k$
- Viene formato l'insieme  $T_{k+1}$  aggiungendo il nodo n all'insieme  $T_k$
- L'algoritmo termina quando sono stati esplorati tutti i nodi



Situazione al passo K

- Al passo k viene aggiunto all'insieme  $T_{k-1}$  il nodo n caratterizzato dal cammino a costo minimo con il nodo sorgente s che transita esclusivamente in nodi dell'insieme  $T_{k-1}$

- Notazioni:

- $N$  : insieme dei nodi del grafo
- $s$  : nodo sorgente
- $T_k$  : insieme dei nodi raggiunti dall'algoritmo al passo k
- $c(i,j)$  : peso (costo) del ramo  $(i,j)$ 
  - $c(i,i) = 0$
  - $c(i,j) \geq 0$  se i vertici i e j sono connessi direttamente
  - $c(i,j) = \infty$  se i vertici i e j non sono connessi direttamente
- $L_k(n)$  : costo del cammino minimo, individuato dall'algoritmo fino al passo k, tra il nodo s ed un generico nodo n

- Inizializzazione ( $k=1$ )

- $T_1 = \{s\}$
- $L_1(n) = c(s,n)$  per  $n \neq s$

- Aggiunta di un nodo (passo  $1 \leq k \leq N$ )

- Trovare  $x \notin T_{k-1}$  tale che:

$$L_{k-1}(x) = \min_{j \notin T_{k-1}} \{L_{k-1}(j)\}$$

- Aggiungere all'insieme  $T_{k-1}$  il nodo x ed il ramo incidente a x

- Aggiornamento dei cammini minimi:

$$L_k(n) = \min [L_{k-1}(n), L_{k-1}(x) + c(x,n)] \quad \text{per tutti i valori di } n \notin T_k$$

- Al termine:

- L'insieme  $T_N$  è uno spanning tree del grafo di partenza contenente i cammini a costo minimo tra il nodo sorgente e tutti gli altri nodi del grafo
- $L_N(n)$  indica il costo del cammino a costo minimo tra il nodo s ed il nodo n

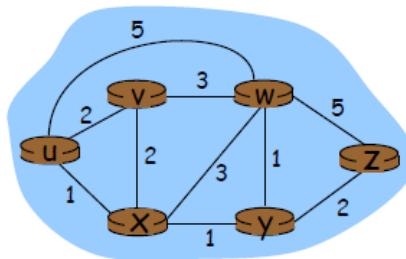
- Si noti che:

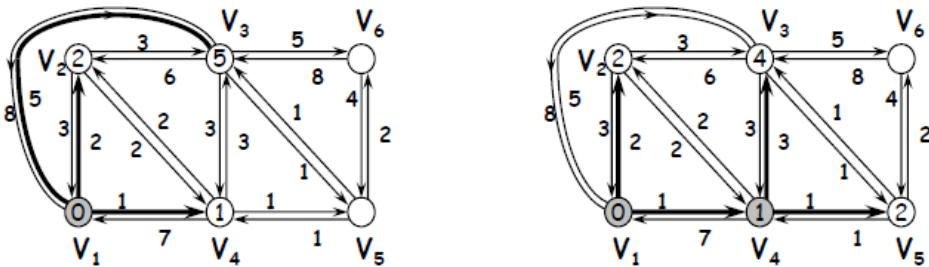
- Al passo k-mo viene aggiunto all'insieme  $T_{k-1}$  il k-mo nodo ed è individuato il cammino a costo minimo tra il tale nodo ed il nodo sorgente
- Questo cammino transita esclusivamente attraverso i nodi sinora compresi nell'insieme  $T_{k-1}$

- La complessità dell'algoritmo è  $O(N^2)$

**Esempio**

| passo | $N'$   | $D(v), p(v)$ | $D(w), p(w)$ | $D(x), p(x)$ | $D(y), p(y)$ | $D(z), p(z)$ |
|-------|--------|--------------|--------------|--------------|--------------|--------------|
| 0     | u      | 2,u          | 5,u          | 1,u          | $\infty$     | $\infty$     |
| 1     | ux     | 2,u          | 4,x          |              | 2,x          | $\infty$     |
| 2     | uxy    | 2,u          | 3,y          |              | 4,y          |              |
| 3     | uxyv   |              | 3,y          |              | 4,y          |              |
| 4     | uxyvw  |              |              |              | 4,y          |              |
| 5     | uxyvwz |              |              |              |              |              |

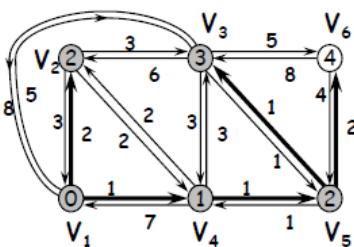
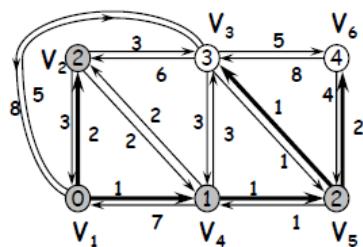




$$T_1 = \{1\}$$

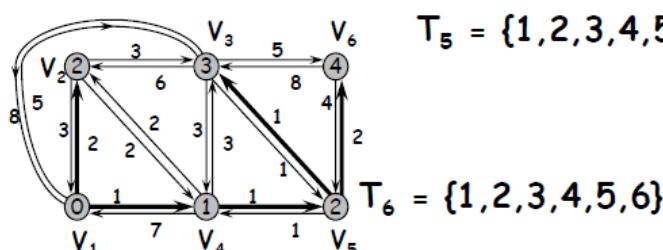
$$T_2 = \{1, 4\}$$

$$T_3 = \{1, 2, 4\}$$



$$T_4 = \{1, 2, 4, 5\}$$

$$T_5 = \{1, 2, 3, 4, 5\}$$



### Algoritmo di Dijkstra

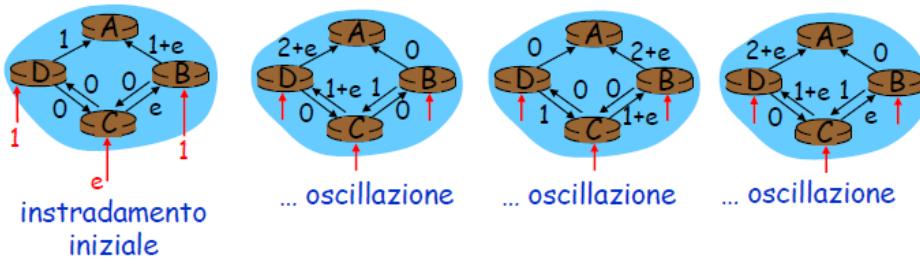
- Complessità dell'algoritmo (n nodi):

- Ciascuna iterazione: controllo su tutti i nodi, w, non in N
- $$n(n+1)/2 \rightarrow O(n^2)$$

- La più efficiente implementazione possibile:

$$O(n \log n)$$

- Possibili oscillazioni
- Es. costo del collegamento = quantità di traffico trasportato



### Instradamento gerarchico

- Ipotesi utilizzate:

- Ciascun router era indistinguibile dagli altri
- Visione omogenea della rete

- Problemi:

- Autonomia amministrativa:
  - Internet: rete di reti
  - Ogni ISP deve essere in grado di amministrare la propria rete nel modo desiderato, pur mantenendo la possibilità di connetterla alle reti esterne
- Scalabilità:
  - 200 milioni di destinazioni

- Archiviare le informazioni d'instradamento su ciascun host richiederebbe un'enorme quantità di memoria
- Il traffico generato dagli aggiornamenti LS non lascerebbero banda per i pacchetti di dati

- **Organizzazione di router in sistemi autonomi (AS, Autonomous System).**

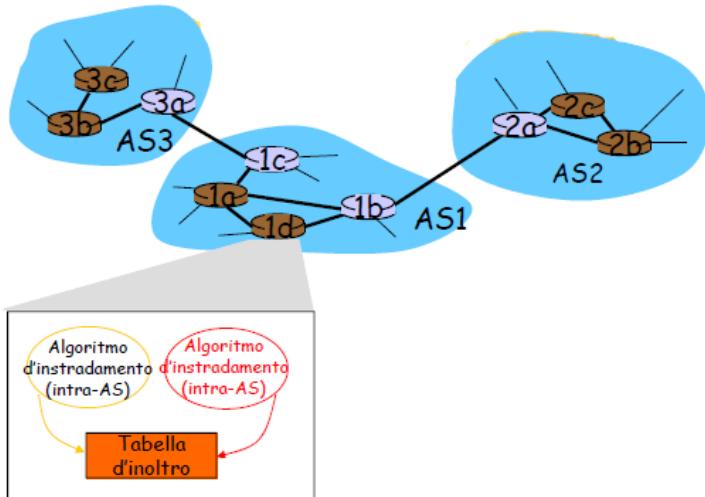
- **I router di un AS eseguono lo stesso algoritmo d'instradamento:**

- Protocollo d'instradamento interno al sistema autonomo (intra-AS) (IGP)
- I router appartenenti a differenti AS possono eseguire protocolli d'instradamento inter-AS (IGP diversi)

- **Router gateway:**

- Hanno il compito aggiuntivo d'inoltrare pacchetti a destinazioni esterne ad un AS

### Sistemi autonomi interconnessi



- Ciascun router interno ad un AS sa come inoltrare pacchetti lungo il percorso ottimo verso qualsiasi destinazione interna

- I sistemi AS2 e AS3 hanno tre router
- I protocolli d'instradamento dei tre sistemi autonomi non sono necessariamente gli stessi
- I router 1b, 1c, 2a e 3a sono **Gateway**

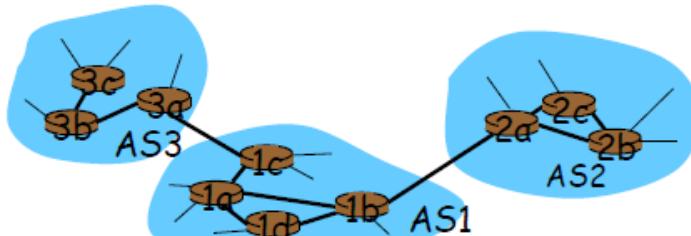
### Instradamento tra sistemi autonomi

- Un router in AS1 riceve un pacchetto con destinazione esterna a AS1:

- Il router dovrebbe inoltrare il pacchetto verso uno dei due gateway
- Quale?

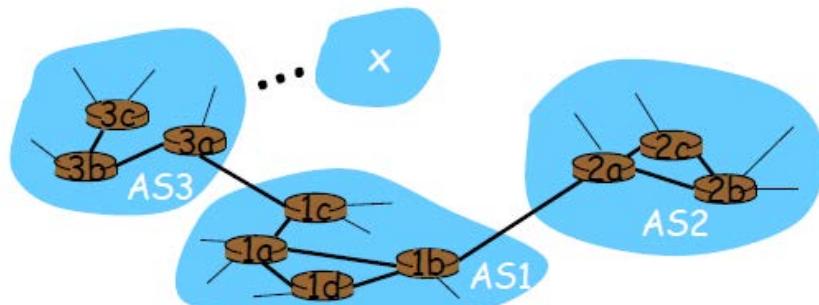
- AS1 deve:

- Sapere quali destinazioni sono raggiungibili attraverso AS2 e quali attraverso AS3
- Informare tutti i router all'interno dell'AS in modo che ciascuno possa configurare la propria tabella d'inoltro per gestire destinazioni esterne



### Esempio: Gateway unico

- AS1 apprende dal proprio protocollo d'instradamento inter-AS (EGP) che la sottorete x è raggiungibile da AS3 (gateway 1c), ma non da AS2
- Il protocollo inter-AS (EGP) propaga questa informazione a tutti i propri router
- Il router 1d determina, partendo dall'informazione fornita dal protocollo intra-AS (IGP), l'interfaccia I del router sul percorso a costo minimo dal router 1d al gateway 1c.
- Il router 1d può inserire la riga (x,I) nella propria tabella d'inoltro.

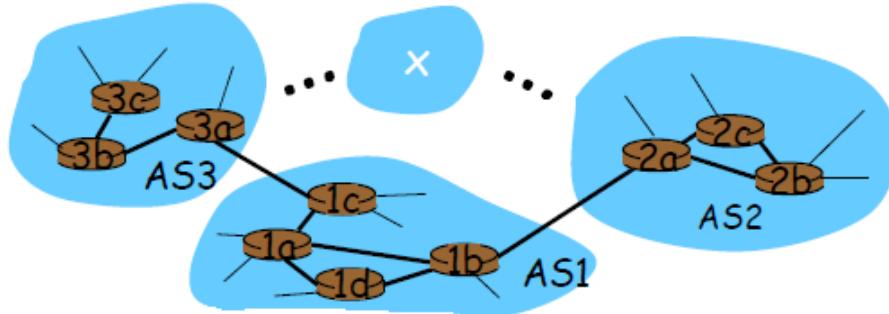


### Esempio: Gateway multiplo

- Supponiamo che AS1 apprenda dal protocollo d'instradamento tra sistemi autonomi che la sottorete x è raggiungibile da AS2 e da AS3
- Al fine di configurare la propria tabella d'inoltro, il router 1d dovrebbe determinare a quale gateway, 1b o 1c, indirizzare i pacchetti destinati alla sottorete x.

#### Instradamento a hot potato:

- Si sceglie il gateway con percorso a costo minimo (protocollo IGP)



Dal protocollo inter-AS si apprende che la sottorete x è raggiungibile attraverso più gateway

Si usa l'informazione d'instradamento proveniente dal protocollo intra-AS per determinare i costi dei percorsi a costo minimo verso i gateway

Instradamento "hot potato": si sceglie il gateway che ha il costo minimo inferiore

Della tabella di routing si determina l'interfaccia I che conduce al gateway a costo minimo. Si scrive (x,I) nella tabella di routing

### "RIP" e "OSPF"

#### Protocolli di instradamento

- I protocolli d'instradamento intra-AS sono noti anche come **Interior Gateway Protocol(IGP)**

#### I protocolli IGP più comuni sono:

- RIP (Routing Information Protocol)
- OSPF (Open Shortest Path First)
- IGRP (Interior Gateway Routing Protocol) (protocollo proprietario Cisco)

#### Routing Information Protocol RIP

- RFC 1058

#### Distance Vector Routing Protocol:

- La metrica dei rami dipende normalmente dal loro stato (sano/guasto)
- Conteggio degli hop come metrica di costo (max = 15 hop)

- È utilizzato in reti di piccole-medie dimensioni

- È molto semplice, ma:

- La convergenza è lenta
- Lo stato di equilibrio può essere un sub-ottimo

- Utilizza il protocollo UDP (port number 520)

- Due tipi di messaggi RIP:

- Request per chiedere ai vicini il distance vector
- Response per annunciare il distance vector

- I router adiacenti si scambiano gli aggiornamenti d'instradamento ogni 30 secondi

- Ogni messaggio contiene un elenco comprendente fino a 25 sottoreti e la distanza del mittente rispetto a ciascuna di tali sottoreti

### RIP v1

|                      |         |   |
|----------------------|---------|---|
| Command              | Version | 0 |
| Address Identifier   |         | 0 |
| IP Address 1         |         |   |
| 0                    |         |   |
| 0                    |         |   |
| Metric for address 1 |         |   |
| Address Identifier   |         | 0 |
| IP Address 2         |         |   |
| 0                    |         |   |
| 0                    |         |   |
| Metric for address 2 |         |   |
| ⋮                    |         |   |
| Address Identifier   |         | 0 |
| IP Address N         |         |   |
| 0                    |         |   |
| 0                    |         |   |
| Metric for address N |         |   |

### Header

- Command

- request
- response

- Version

### Block

- IP address

- rete, sottorete o host

- Metric

- distanza dalla rete indicata nell'IP address

### RIP v2

|                      |         |          |
|----------------------|---------|----------|
| Command              | Version | Reserved |
| Address Identifier   |         | Reserved |
| IP Address 1         |         |          |
| Subnet Mask          |         |          |
| Next Hop             |         |          |
| Metric for address 1 |         |          |
| Address Identifier   |         | Reserved |
| IP Address 2         |         |          |
| Subnet Mask          |         |          |
| Next Hop             |         |          |
| Metric for address 2 |         |          |
| ⋮                    |         |          |
| Address Identifier   |         | Reserved |
| IP Address N         |         |          |
| Subnet Mask          |         |          |
| Next Hop             |         |          |
| Metric for address N |         |          |

- IP address

- rete, sottorete o host

- Subnet Mask

- specifica come interpretare i bit dell'indirizzo

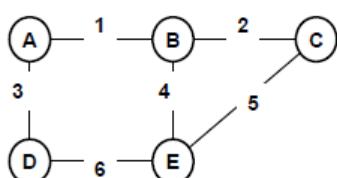
- Next Hop

- indica a quale next hop router il router emittente il messaggio RIP invierà i pacchetti diretti all'indirizzo specificato

- Metric

- distanza dalla rete indicata nell'IP address

### Esempio RIP: Inizializzazione



### Routing Table

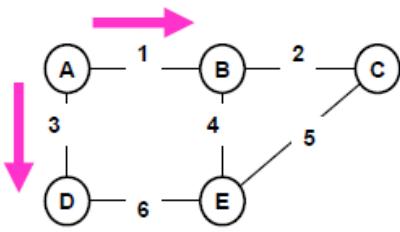
| A | Destinazione | A     | B     | C     | D     | E     |
|---|--------------|-------|-------|-------|-------|-------|
|   | Distanza     | 0     | ?     | ?     | ?     | ?     |
|   | Link         | local | ?     | ?     | ?     | ?     |
| B | Destinazione | A     | B     | C     | D     | E     |
|   | Distanza     | ?     | 0     | ?     | ?     | ?     |
|   | Link         | ?     | local | ?     | ?     | ?     |
| C | Destinazione | A     | B     | C     | D     | E     |
|   | Distanza     | ?     | ?     | 0     | ?     | ?     |
|   | Link         | ?     | ?     | local | ?     | ?     |
| D | Destinazione | A     | B     | C     | D     | E     |
|   | Distanza     | ?     | ?     | ?     | 0     | ?     |
|   | Link         | ?     | ?     | ?     | local | ?     |
| E | Destinazione | A     | B     | C     | D     | E     |
|   | Distanza     | ?     | ?     | ?     | ?     | 0     |
|   | Link         | ?     | ?     | ?     | ?     | local |

- Condizione iniziale:

- Routing table vuote

- Metrica:

- Distanza



## Routing Table

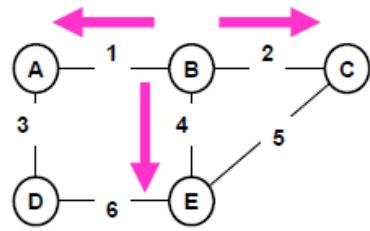
| A | Destinazione | A     | B | C | D | E |
|---|--------------|-------|---|---|---|---|
|   | Distanza     | 0     | ? | ? | ? | ? |
|   | Link         | local | ? | ? | ? | ? |

| B | Destinazione | A | B     | C | D | E |
|---|--------------|---|-------|---|---|---|
|   | Distanza     | 1 | 0     | ? | ? | ? |
|   | Link         | 1 | local | ? | ? | ? |

| C | Destinazione | A | B | C     | D | E |
|---|--------------|---|---|-------|---|---|
|   | Distanza     | ? | ? | 0     | ? | ? |
|   | Link         | ? | ? | local | ? | ? |

| D | Destinazione | A | B | C | D     | E |
|---|--------------|---|---|---|-------|---|
|   | Distanza     | 1 | ? | ? | 0     | ? |
|   | Link         | 3 | ? | ? | local | ? |

| E | Destinazione | A | B | C | D | E     |
|---|--------------|---|---|---|---|-------|
|   | Distanza     | ? | ? | ? | ? | 0     |
|   | Link         | ? | ? | ? | ? | local |



## Routing Table

| A | Destinazione | A     | B | C | D | E |
|---|--------------|-------|---|---|---|---|
|   | Distanza     | 0     | 1 | ? | ? | ? |
|   | Link         | local | 1 | ? | ? | ? |

| B | Destinazione | A | B     | C | D | E |
|---|--------------|---|-------|---|---|---|
|   | Distanza     | 1 | 0     | ? | ? | ? |
|   | Link         | 1 | local | ? | ? | ? |

| C | Destinazione | A | B | C     | D | E |
|---|--------------|---|---|-------|---|---|
|   | Distanza     | 2 | 1 | 0     | ? | ? |
|   | Link         | 2 | 2 | local | ? | ? |

| D | Destinazione | A | B | C | D     | E |
|---|--------------|---|---|---|-------|---|
|   | Distanza     | 1 | ? | ? | 0     | ? |
|   | Link         | 3 | ? | ? | local | ? |

| E | Destinazione | A | B | C | D | E     |
|---|--------------|---|---|---|---|-------|
|   | Distanza     | 2 | 1 | ? | ? | 0     |
|   | Link         | 4 | 4 | ? | ? | local |

## Step 3:

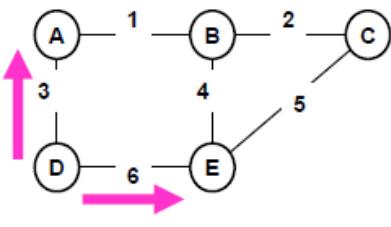
- B emette un messaggio verso A, C e E

| B | Address | A | B | --- | --- | --- |
|---|---------|---|---|-----|-----|-----|
|   | Metric  | 1 | 0 | --- | --- | --- |

## Step 4:

- D emette un messaggio verso A e E

| D | Address | A | --- | --- | D | --- |
|---|---------|---|-----|-----|---|-----|
|   | Metric  | 1 | --- | --- | 0 | --- |



## Routing Table

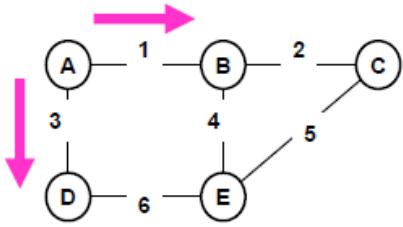
| A | Destinazione | A     | B | C | D | E |
|---|--------------|-------|---|---|---|---|
|   | Distanza     | 0     | 1 | ? | 1 | ? |
|   | Link         | local | 1 | ? | 3 | ? |

| B | Destinazione | A | B     | C | D | E |
|---|--------------|---|-------|---|---|---|
|   | Distanza     | 1 | 0     | ? | ? | ? |
|   | Link         | 1 | local | ? | ? | ? |

| C | Destinazione | A | B | C     | D | E |
|---|--------------|---|---|-------|---|---|
|   | Distanza     | 2 | 1 | 0     | ? | ? |
|   | Link         | 2 | 2 | local | ? | ? |

| D | Destinazione | A | B | C | D     | E |
|---|--------------|---|---|---|-------|---|
|   | Distanza     | 1 | ? | ? | 0     | ? |
|   | Link         | 3 | ? | ? | local | ? |

| E | Destinazione | A | B | C | D | E     |
|---|--------------|---|---|---|---|-------|
|   | Distanza     | 2 | 1 | ? | 1 | 0     |
|   | Link         | 4 | 4 | ? | 6 | local |



## Routing Table

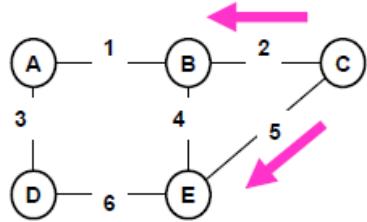
| A | Destinazione | A     | B | C | D | E |
|---|--------------|-------|---|---|---|---|
|   | Distanza     | 0     | 1 | ? | 1 | ? |
|   | Link         | local | 1 | ? | 3 | ? |

| B | Destinazione | A | B     | C | D | E |
|---|--------------|---|-------|---|---|---|
|   | Distanza     | 1 | 0     | ? | 2 | ? |
|   | Link         | 1 | local | ? | 1 | ? |

| C | Destinazione | A | B | C     | D | E |
|---|--------------|---|---|-------|---|---|
|   | Distanza     | 2 | 1 | 0     | ? | ? |
|   | Link         | 2 | 2 | local | ? | ? |

| D | Destinazione | A | B | C | D     | E |
|---|--------------|---|---|---|-------|---|
|   | Distanza     | 1 | 2 | ? | 0     | ? |
|   | Link         | 3 | 3 | ? | local | ? |

| E | Destinazione | A | B | C | D | E     |
|---|--------------|---|---|---|---|-------|
|   | Distanza     | 2 | 1 | ? | 1 | 0     |
|   | Link         | 4 | 4 | ? | 6 | local |



## Routing Table

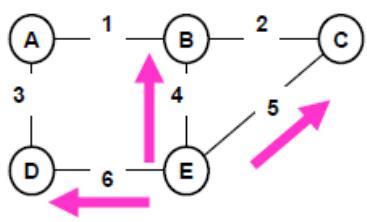
| A | Destinazione | A     | B | C | D | E |
|---|--------------|-------|---|---|---|---|
|   | Distanza     | 0     | 1 | ? | 1 | ? |
|   | Link         | local | 1 | ? | 3 | ? |

| B | Destinazione | A | B     | C | D | E |
|---|--------------|---|-------|---|---|---|
|   | Distanza     | 1 | 0     | 1 | 2 | ? |
|   | Link         | 1 | local | 2 | 1 | ? |

| C | Destinazione | A | B | C     | D | E |
|---|--------------|---|---|-------|---|---|
|   | Distanza     | 2 | 1 | 0     | ? | ? |
|   | Link         | 2 | 2 | local | ? | ? |

| D | Destinazione | A | B | C | D     | E |
|---|--------------|---|---|---|-------|---|
|   | Distanza     | 1 | 2 | ? | 0     | ? |
|   | Link         | 3 | 3 | ? | local | ? |

| E | Destinazione | A | B | C | D | E     |
|---|--------------|---|---|---|---|-------|
|   | Distanza     | 2 | 1 | 1 | 1 | 0     |
|   | Link         | 4 | 4 | 5 | 6 | local |



## Routing Table

| A | Destinazione | A     | B | C | D | E |
|---|--------------|-------|---|---|---|---|
|   | Distanza     | 0     | 1 | ? | 1 | ? |
|   | Link         | local | 1 | ? | 3 | ? |

| B | Destinazione | A | B     | C | D | E |
|---|--------------|---|-------|---|---|---|
|   | Distanza     | 1 | 0     | 1 | 2 | 1 |
|   | Link         | 1 | local | 2 | 1 | 4 |

| C | Destinazione | A | B | C     | D | E |
|---|--------------|---|---|-------|---|---|
|   | Distanza     | 2 | 1 | 0     | 2 | 1 |
|   | Link         | 2 | 2 | local | 5 | 5 |

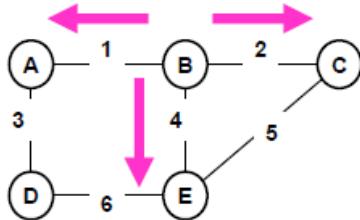
| D | Destinazione | A | B | C | D     | E |
|---|--------------|---|---|---|-------|---|
|   | Distanza     | 1 | 2 | 2 | 0     | 1 |
|   | Link         | 3 | 3 | 6 | local | 6 |

| E | Destinazione | A | B | C | D | E     |
|---|--------------|---|---|---|---|-------|
|   | Distanza     | 2 | 1 | 1 | 1 | 0     |
|   | Link         | 4 | 4 | 5 | 6 | local |

### Step 7:

- E emette un messaggio verso B, C e D

| E | Address | A | B | C | D | E |
|---|---------|---|---|---|---|---|
|   | Metric  | 2 | 1 | 1 | 1 | 0 |



## Routing Table

| A | Destinazione | A     | B | C | D | E |
|---|--------------|-------|---|---|---|---|
|   | Distanza     | 0     | 1 | 2 | 1 | 2 |
|   | Link         | local | 1 | 1 | 3 | 1 |

| B | Destinazione | A | B     | C | D | E |
|---|--------------|---|-------|---|---|---|
|   | Distanza     | 1 | 0     | 1 | 2 | 1 |
|   | Link         | 1 | local | 2 | 1 | 4 |

| C | Destinazione | A | B | C     | D | E |
|---|--------------|---|---|-------|---|---|
|   | Distanza     | 2 | 1 | 0     | 2 | 1 |
|   | Link         | 2 | 2 | local | 5 | 5 |

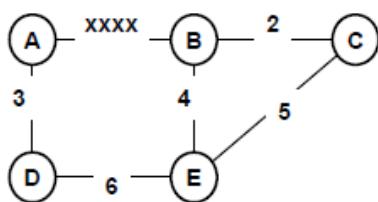
| D | Destinazione | A | B | C | D     | E |
|---|--------------|---|---|---|-------|---|
|   | Distanza     | 1 | 2 | 2 | 0     | 1 |
|   | Link         | 3 | 3 | 6 | local | 6 |

| E | Destinazione | A | B | C | D | E     |
|---|--------------|---|---|---|---|-------|
|   | Distanza     | 2 | 1 | 1 | 1 | 0     |
|   | Link         | 4 | 4 | 5 | 6 | local |

### RIP: guasto sul collegamento e recupero

- Se un router non riceve notizie dal suo vicino per 180 sec, il nodo adiacente viene considerato spento o guasto:
  - RIP modifica la tabella d'instradamento locale
  - Propaga l'informazione mandando annunci ai router vicini
  - I vicini inviano nuovi messaggi (se la loro tabella d'instradamento è cambiata)
  - L'informazione che il collegamento è guasto si propaga su tutta la rete
  - L'utilizzo dell'inversione avvelenata evita i loop (distanza infinita = 16 hop)

### Esempio RIP: Guasto di un ramo



## Routing Table

| A | Destinazione | A     | B   | C   | D | E   |
|---|--------------|-------|-----|-----|---|-----|
|   | Distanza     | 0     | inf | inf | 1 | inf |
|   | Link         | local | 1   | 1   | 3 | 1   |

| B | Destinazione | A   | B     | C | D   | E |
|---|--------------|-----|-------|---|-----|---|
|   | Distanza     | inf | 0     | 1 | inf | 1 |
|   | Link         | 1   | local | 2 | 1   | 4 |

| C | Destinazione | A | B | C     | D | E |
|---|--------------|---|---|-------|---|---|
|   | Distanza     | 2 | 1 | 0     | 2 | 1 |
|   | Link         | 2 | 2 | local | 5 | 5 |

| D | Destinazione | A | B | C | D     | E |
|---|--------------|---|---|---|-------|---|
|   | Distanza     | 1 | 2 | 2 | 0     | 1 |
|   | Link         | 3 | 3 | 6 | local | 6 |

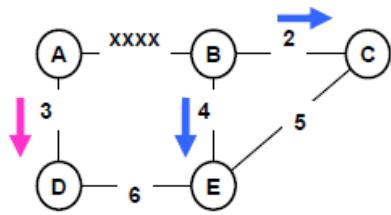
| E | Destinazione | A | B | C | D | E     |
|---|--------------|---|---|---|---|-------|
|   | Distanza     | 2 | 1 | 1 | 1 | 0     |
|   | Link         | 4 | 4 | 5 | 6 | local |

### Condizione iniziale

- rete a regime
- guasto del ramo AB

### Metrica

- Distanza



## Routing Table

|   | Destinazione | A     | B   | C   | D | E   |
|---|--------------|-------|-----|-----|---|-----|
| A | Distanza     | 0     | inf | inf | 1 | inf |
|   | Link         | local | 1   | 1   | 3 | 1   |

### Step 1

Messaggio di A verso D

| A | Address | A | B   | C   | D | E   |
|---|---------|---|-----|-----|---|-----|
|   | Metric  | 0 | inf | inf | 1 | inf |

Messaggio di B verso C ed E

| B | Address | A   | B | C | D   | E |
|---|---------|-----|---|---|-----|---|
|   | Metric  | inf | 0 | 1 | inf | 1 |

### Step 2

Messaggio di C verso B, E

| C | Address | A   | B | C | D | E   |
|---|---------|-----|---|---|---|-----|
|   | Metric  | inf | 1 | 0 | 2 | inf |

Messaggio di D verso A, E

| D | Address | A | B   | C | D | E |
|---|---------|---|-----|---|---|---|
|   | Metric  | 1 | inf | 2 | 0 | 1 |

Messaggio di E verso B, C, D

| E | Address | A   | B | C | D | E |
|---|---------|-----|---|---|---|---|
|   | Metric  | inf | 1 | 1 | 1 | 0 |

### Step 3

Messaggio di A verso D

| A | Address | A | B   | C | D | E |
|---|---------|---|-----|---|---|---|
|   | Metric  | 0 | inf | 3 | 1 | 2 |

Messaggio di B verso E, C

| B | Address | A   | B | C | D | E |
|---|---------|-----|---|---|---|---|
|   | Metric  | inf | 0 | 1 | 2 | 1 |

Messaggio di D verso A, E

| D | Address | A | B | C | D | E |
|---|---------|---|---|---|---|---|
|   | Metric  | 1 | 2 | 2 | 0 | 1 |

Messaggio di E verso B, C, D

| E | Address | A | B | C | D | E |
|---|---------|---|---|---|---|---|
|   | Metric  | 2 | 1 | 1 | 1 | 0 |

## Routing Table

|   | Destinazione | A     | B   | C | D | E |
|---|--------------|-------|-----|---|---|---|
| A | Distanza     | 0     | inf | 3 | 1 | 2 |
|   | Link         | local | 1   | 3 | 3 | 3 |

|   | Destinazione | A   | B     | C | D | E |
|---|--------------|-----|-------|---|---|---|
| B | Distanza     | inf | 0     | 1 | 2 | 1 |
|   | Link         | 1   | local | 2 | 4 | 4 |

|   | Destinazione | A   | B | C     | D | E |
|---|--------------|-----|---|-------|---|---|
| C | Distanza     | inf | 1 | 0     | 2 | 1 |
|   | Link         | 2   | 2 | local | 5 | 5 |

|   | Destinazione | A | B | C | D     | E |
|---|--------------|---|---|---|-------|---|
| D | Distanza     | 1 | 2 | 2 | 0     | 1 |
|   | Link         | 3 | 6 | 6 | local | 6 |

|   | Destinazione | A | B | C | D | E     |
|---|--------------|---|---|---|---|-------|
| E | Distanza     | 2 | 1 | 1 | 1 | 0     |
|   | Link         | 6 | 4 | 5 | 6 | local |

## Routing Table

|   | Destinazione | A | B | C | D | E |
|---|--------------|---|---|---|---|---|
| A | Distanza     | 0 | 3 | 3 | 1 | 2 |
|   | Link         | 3 | 3 | 3 | 3 | 3 |

|   | Destinazione | A | B     | C | D | E |
|---|--------------|---|-------|---|---|---|
| B | Distanza     | 3 | 0     | 1 | 2 | 1 |
|   | Link         | 4 | local | 2 | 4 | 4 |

|   | Destinazione | A | B | C     | D | E |
|---|--------------|---|---|-------|---|---|
| C | Distanza     | 3 | 1 | 0     | 2 | 1 |
|   | Link         | 5 | 2 | local | 5 | 5 |

|   | Destinazione | A | B | C | D     | E |
|---|--------------|---|---|---|-------|---|
| D | Distanza     | 1 | 2 | 2 | 0     | 1 |
|   | Link         | 3 | 6 | 6 | local | 6 |

|   | Destinazione | A | B | C | D | E     |
|---|--------------|---|---|---|---|-------|
| E | Distanza     | 2 | 1 | 1 | 1 | 0     |
|   | Link         | 6 | 4 | 5 | 6 | local |

## **Open Shortest Path First (OSPF)**

- Le specifiche del protocollo sono pubblicamente disponibili (“Open”) (RFC 2328)
- È un protocollo “link state”:
  - Utilizza il flooding di informazioni sullo stato dei link (**Link State Advertisement – LSA**)
  - Utilizza l’algoritmo di Dijkstra per la determinazione del percorso a costo minimo
- Ogni volta che si verifica un cambiamento nello stato di un link, il router emette un LSA verso tutti gli altri router
- Gli LSA sono trasferiti nel sistema autonomo utilizzando il flooding:
  - I messaggi OSPF vengono trasportati direttamente da IP (e non da TCP o UDP) con un protocollo di livello superiore
  - Rapida convergenza in caso di cambiamenti di stato

## **Vantaggi di OSPF**

- **Sicurezza:**
  - Gli scambi tra router sono autenticati
- **Multipath:**
  - Quando più percorsi verso una destinazione hanno lo stesso costo, OSPF consente di usarli senza doverne scegliere uno, come invece avveniva in RIP
- **Su ciascun collegamento, vi possono essere più metriche di costo per differenti TOS:**
  - Es: il costo del satellite sarà “basso” per un best effort; elevato per un real time
- **Supporto integrato per l’instradamento unicast e multicast:**
  - Per consentire l’instradamento multicast viene impiegato MOSPF (OSPF multicast) che utilizza il database di collegamenti OSPF
- **Supporto alle gerarchie in un dominio d’instradamento**

## **Link State Routing**

- **Gli LSA sono emessi:**
  - Quando un router contatta un nuovo router vicino
  - Quando un link si guasta
  - Quando il costo di un link varia
  - Periodicamente ogni fissato intervallo di tempo
- **La rete trasporta gli LSA mediante la tecnica di flooding:**
  - Un LSA è rilanciato da un router su tutte le sue interfacce tranne quella da cui è stato ricevuto
  - Gli LSA trasportano dei riferimenti temporali (time stamp) o numeri di sequenza per:
    - Evitare il rilancio di pacchetti già rilanciati
    - Consentire un corretto riscontro dal ricevente

## **Tecnica Flooding**

- **Assicura che tutti i router di una rete:**
  - Riescano a costruire un database contenente lo stato della rete
  - Abbiano le stesse informazioni sullo stato dei link
- **Alla ricezione di un LSP:**
  - Un router esamina i campi di un LSP: link identifier, metrica, time stamp o numero di sequenza
  - Se il dato non è contenuto nel database, viene memorizzato e l’LSP è rilanciato su tutte le interfacce del router tranne quella di ricezione
  - Se il dato ricevuto è più recente di quello contenuto nel database, il suo valore è memorizzato e l’LSP è rilanciato su tutte le interfacce del router tranne quella di ricezione
  - Se il dato ricevuto è più vecchio di quello contenuto nel database, viene rilanciato un LSP con il valore contenuto nel database esclusivamente sull’interfaccia di arrivo dell’LSP
  - Se i due dati sono della stessa età non viene eseguita alcuna operazione
- **La tecnica flooding ha i seguenti vantaggi:**
  - Esplora tutti i possibili cammini tra origine e destinazione
  - è estremamente affidabile e robusta
  - Almeno una copia di ogni LSP seguirà la via a minor costo
- **Il traffico generato dipende dalle dimensioni della rete e può essere molto elevato**

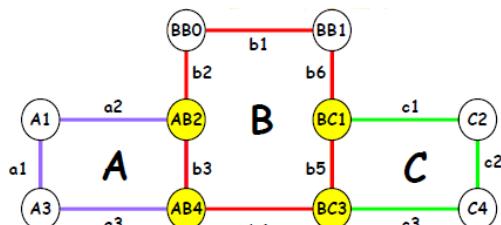
## **Suddivisione di grandi reti in aree**

- **Se la rete è di grandi dimensioni:**
  - Cresce il numero di record del database e quindi la memoria necessaria in ogni router
  - Cresce il tempo necessario al calcolo dei percorsi
  - Cresce il traffico di segnalazione dovuto all’invio degli LSP
- **OSPF risolve questo problema adottando un instradamento di tipo gerarchico:**
  - Una rete è suddivisa in aree:
    - Sezioni indipendente di rete Database separati
    - Meccanismi di flooding indipendenti
  - Le singole aree sono interconnesse da un area di backbone
- **Alcuni router (Area Border Router - ABR) apparterranno ad aree diverse:**
  - Ogni area ha almeno un ABR
  - Ogni area è almeno connessa all’area di backbone
- **Un ABR:**
  - Contiene i database delle aree a cui appartiene
  - Emette degli appositi messaggi (**summary records**) che contengono la lista delle sottoreti raggiungibili attraverso le aree a cui appartiene

## Instradamenti esterni

- Un AS è connesso ad altri AS attraverso uno o più “**AS Border Router**”
- Se l’AS Border Router è unico, è sufficiente indicare a tutti i router interni l’instradamento di default verso l’esterno
- Se gli AS Border Router sono più di uno, ognuno di essi indicherà ai router interni il costo della via verso l’esterno:
  - External record

## Suddivisione di grandi reti in aree



### Il database di un router dell’area A conterrà:

- I record dei link a1, a2, a3, comunicati dai router A1, A3, AB2, AB4
- I summary record relativi alle sottoreti comprese nell’area di backbone e nell’area C, comunicati dai router AB2 e AB4
  - Ad ogni sottorete sarà associato il costo di raggiungimento
  - Analogia con i protocolli distance vector
- Gli external record emessi dai router BB0 e BB1 e rilanciati dai router AB2 e AB4
  - Ad ogni destinazione sarà associato il costo di raggiungimento

## Open Shortest Path First

### • OSPF è il protocollo IGP più utilizzato nelle reti più recenti e di grandi dimensioni:

- è basato sullo scambio di LSP detti Link State Advertisement (LSA)
- Supporta metriche relativi a diversi valori del campo TOS
- Supporta l’uso del concetto di variable length subnet mask (CIDR)
- Supporta il servizio di autenticazione tra router supporta l’indicazione di specific routes
- Riduzione delle dimensioni delle tabelle di routing con l’uso del concetto di Designated Router (DR)
- Supporto del concetto di virtual link per l’interconnessione di aree non contigue

## Terminologia OSPF

### • Area:

- è un insieme logico di reti e di router (geografico, amministrativo, ...)
- Ha lo scopo di limitare la dimensione dei database di descrizione della topologia di rete all’interno dei router
- All’interno di un’area i router devono avere database identici che descrivono la topologia di rete
- Informazioni sulla parte di rete esterna all’area sono contenute in router speciali denominati Area Border Router
- Un Area Border Router trasmette LSA contenenti informazioni sulle reti esterne all’interno dell’area (costo di raggiungimento)
- Tutte le reti OSPF devono essere composte da almeno un area, denominata area di backbone

### • Intra-Area Router (IAR):

- Sono i router che sono situati all’interno di una area OSPF
- Scambiano LSA con tutti gli altri router dell’area
- Gestiscono il database relativo alla topologia dell’area

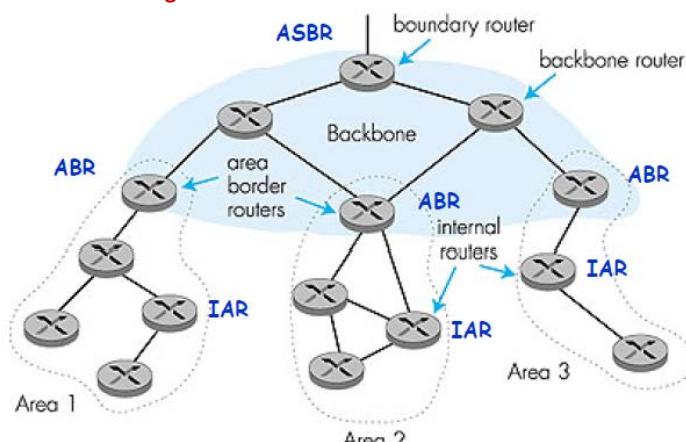
### • Area Border Router (ABR):

- Sono i router che sono connessi a due o più aree OSPF
- Gestiscono i database di topologia di tutte le aree a cui sono connessi
- Trasmettono all’interno di ogni area LSA relativi alle reti presenti in ogni area

### • AS Boundary Router (ASBR):

- Sono i router che sono situati a bordo del dominio OSPF
- Scambiano LSA contenenti informazioni di raggiungibilità di reti di altri AS
- Inviano LSA all’interno del dominio con informazioni sui percorsi esterni

## OSPF strutturato gerarchicamente

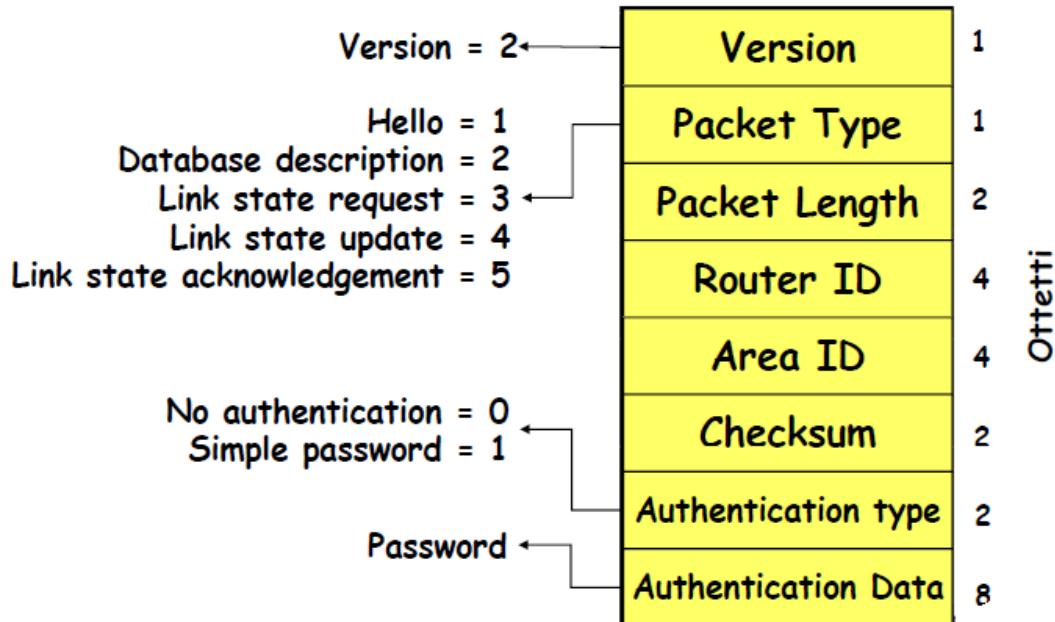


## Tipi di LSA

### • Link State Advertisements (LSA):

- Sono i pacchetti scambiati tra router OSPF per aggiornare i link state database e i percorsi inter-area e inter-AS
- **Router link advertisement:**
  - Indicano lo stato dei link uscenti da un router, sono inviati all'interno di una singola area
- **Summary link advertisement:**
  - Sono generati dagli ABR e individuano le reti contenute nelle altre aree ed i relativi costi di raggiungimento, sono inviati all'interno di tutte le aree gestite da un ABR
- **AS external link advertisement:**
  - sono generati dagli ASBR e indicano i cammini verso le reti esterne al dominio OSPF, sono inviati all'interno di tutte le aree di un dominio OSPF

## Header pacchetti OSPF



## Link State Advertisement

### OSPF Link State Header

|                            |   |         |
|----------------------------|---|---------|
| Link State Age             | 2 |         |
| Options                    | 1 |         |
| Link State Type            | 1 |         |
| Link State ID              | 4 |         |
| Advertising Router         | 4 | Ottetti |
| Link State Sequence Number | 4 |         |
| Link State Checksum        | 2 |         |
| Length                     | 2 |         |

- Tutti i tipi di LSA hanno lo stesso header:

- **Link State Age:**

- Indica il tempo (in secondi) di emissione dell'advertisement

- **Link State Type:**

- 1: Router link
- 2: Network link
- 3: Summary link
  - Inter-area, intra-AS route
- 4: Summary link
  - Route verso l'AS Boundary Router
- 5: AS External link
  - Route verso reti esterne all'AS

- **Link State ID:**

- Indica il tipo di link a cui si riferisce il messaggio
- Tipo 1 e 4: indirizzo IP del Router emittente
- Tipo 3 e 5: indirizzo IP della rete a cui si riferisce il messaggio
- Tipo 2: indirizzo IP del DR emittente

- **Advertising Router:**

- Indirizzo IP del router che ha emesso il messaggio
- Tipo 1 : identico al campo Link State ID
- Tipo 2: indirizzo IP del DR
- Tipo 3 e 4: indirizzo IP del ABR Tipo 5: indirizzo IP del ASBR
- Tipo 5: indirizzo IP del ASBR

### Router Link Ad

|                   |    |
|-------------------|----|
| Link State Header | 20 |
| Reserved          | 1  |
| Reserved          | 1  |
| Number of links   | 2  |
| Link ID           | 4  |
| Link Data         | 4  |
| Type              | 4  |
| Number of TOS     | 1  |
| TOS 0 metric      | 2  |
| TOS               | 1  |
| Reserved          | 1  |
| Metric            | 2  |

Ripetuto per ogni link

### External Link Ad

|                    |    |
|--------------------|----|
| Link State Header  | 20 |
| Network Mask       | 4  |
| Reserved           | 1  |
| Metric             | 3  |
| Forwarding Address | 4  |
| External Route Tag | 4  |
| TOS                | 1  |
| TOS metric         | 3  |
| Forwarding Address | 4  |
| External Route Tag | 4  |

Ripetuto per tutti i valori di TOS

### Network Link Ad

|                   |    |
|-------------------|----|
| Link State Header | 20 |
| Network Mask      | 4  |
| Attached Router   | 4  |

Ripetuto per ogni attached router

### Summary Link Ad

|                   |    |
|-------------------|----|
| Link State Header | 20 |
| Network Mask      | 4  |
| Reserved          | 1  |
| Metric            | 3  |
| TOS               | 1  |
| Metric            | 3  |

Ripetuto per ogni TOS

- **Network Mask:**

- Maschera della rete a cui si riferisce il pacchetto, l'indicazione della rete è contenuta nell'header

- **Metric:**

- Costo del cammino

- **Forwarding Address:**

- Indirizzo IP a cui deve essere inviato il traffico diretto alla rete indicata

- **External Route Tag:**

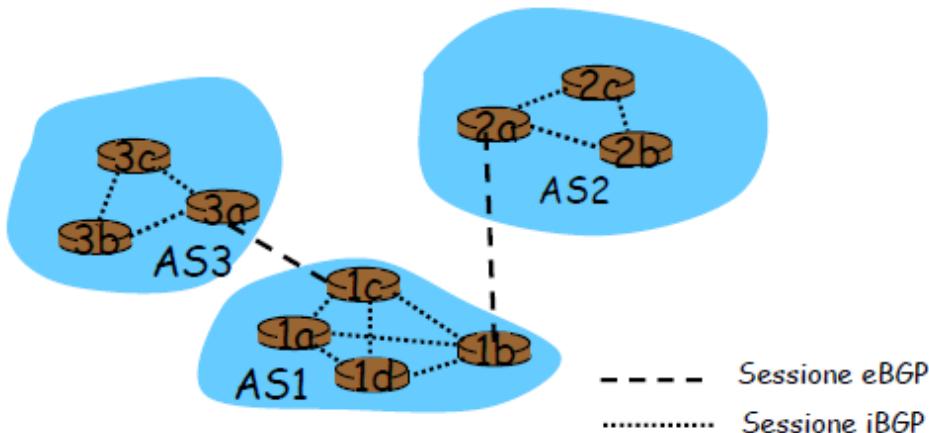
- Suffisso ad uso degli ASBR

## Border Gateway Protocol BGP

- Rappresenta l'attuale standard de facto per i protocolli EGP
- BGP mette a disposizione di ciascun AS un modo per:
  - Ottenere informazioni sulla raggiungibilità delle sottoreti da parte di AS confinanti
  - Propagare le informazioni di raggiungibilità a tutti i router interni di un AS
  - Determinare percorsi "buoni" verso le sottoreti sulla base delle informazioni di raggiungibilità e delle politiche dell'AS
- BGP consente a ciascuna sottorete di comunicare la propria esistenza al resto di Internet

### Terminologia BGP

- **BGP speaker:**
  - Un router che supporta il protocollo BGP
  - Un BGP router non necessariamente coincide con un border router
- **BGP Neighbors:**
  - Una coppia di BGP speaker che si scambiano informazioni di instradamento inter-AS
  - Possono essere di due tipi:
    - Interni: se appartengono allo stesso AS
    - Esterni: se appartengono ad AS diversi
- **BGP session:**
  - La connessione TCP che supporta il colloquio tra due BGP speaker



- **AS number:**
  - Identificatore a 16-bit che identifica univocamente un AS
- **AS path:**
  - è la lista di AS che sono attraversati in un cammino
- **Politiche di routing:**
  - nel protocollo BGP non sono definite regole fisse per la scelta dei cammini inter-AS, ma le regole sono definite dal gestore di ogni AS
    - Un AS multi-homed può rifiutare di operare come AS di transito
    - Un AS multi-homed può operare come AS di transito solo per alcuni AS
    - Un AS può scegliere a quale altro AS affidare il traffico di transito
- Tra le possibili scelte un BGP speaker sceglie quella da preferire in base alla politica di routing fissata dal gestore
- In caso di cammini alternativi, un BGP speaker li mantiene tutti ma ne comunica uno solo agli altri AS
- **Traffico:**
  - **Locale:**
    - Traffico generato o terminato nell'AS
  - **Transito:**
    - Traffico che non è locale
- **AS type:**
  - **Stub:**
    - Uno stub AS ha una singola connessione inter-AS, trasporta solo traffico locale
  - **Multihomed:**
    - Ha un insieme di connessioni verso una molteplicità di altri AS, ma non trasporta traffico di transito
  - **Transit:**
    - Ha un insieme di connessioni verso una molteplicità di altri AS, e trasporta anche traffico di transito

### Attributi del percorso e rotte BGP

- Quando un router annuncia un prefisso per una sessione BGP, include anche un certo **numero di attributi BGP**:
  - Prefisso + attributi = "rotta"
- Due dei più importanti attributi:
  - **AS-PATH:** elenca i sistemi autonomi attraverso i quali è passato l'annuncio del prefisso; es. AS 67 AS 17
  - **NEXT-HOP:** quando si deve inoltrare un pacchetto tra due sistemi autonomi, questo potrebbe essere inviato su uno dei vari collegamenti fisici che li connettono direttamente.
- Quando un router gateway riceve un annuncio di rotta, utilizza le proprie politiche d'importazione per decidere se accettare o filtrare la rotta

### Selezione dei percorsi BGP

- Un router può ricavare più di una rotta verso un determinato prefisso, e deve quindi sceglierne una
- **Regole di eliminazione:**
  - Alle rotte viene assegnato come attributo un valore di preferenza locale. Si selezionano quindi le rotte con i più alti valori di preferenza locale
  - Si seleziona la rotta con valore AS-PATH più breve
  - Si seleziona quella il cui router di NEXT-HOP è più vicino: instradamento “hot potato”
  - Se rimane ancora più di una rotta, il router si basa sugli identificatori BGP

### Messaggi BGP

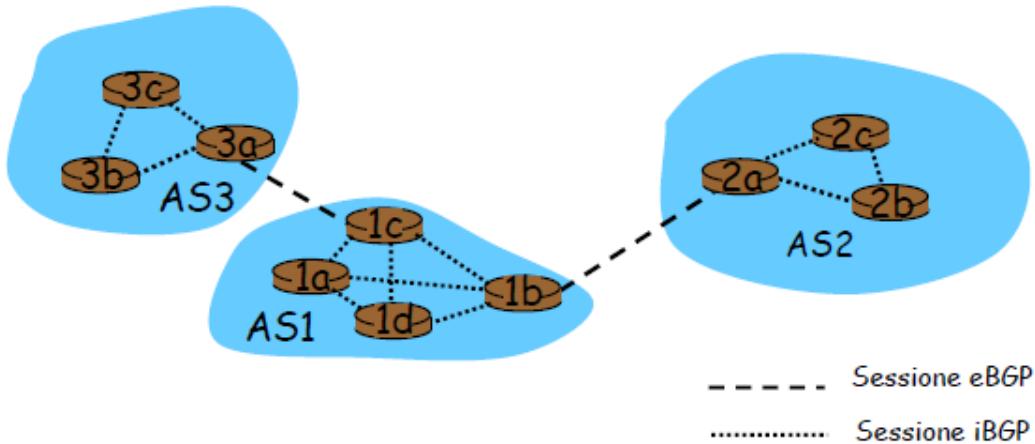
- I messaggi BGP vengono scambiati attraverso TCP

- **Messaggi BGP:**

- **OPEN:**
  - Apre la connessione TCP e autentica il mittente
- **UPDATE:**
  - Annuncia il nuovo percorso (o cancella quello vecchio)
- **KEEPALIVE:**
  - Mantiene la connessione attiva in mancanza di UPDATE
- **NOTIFICATION:**
  - Riporta gli errori del precedente messaggio; usato anche per chiudere il collegamento

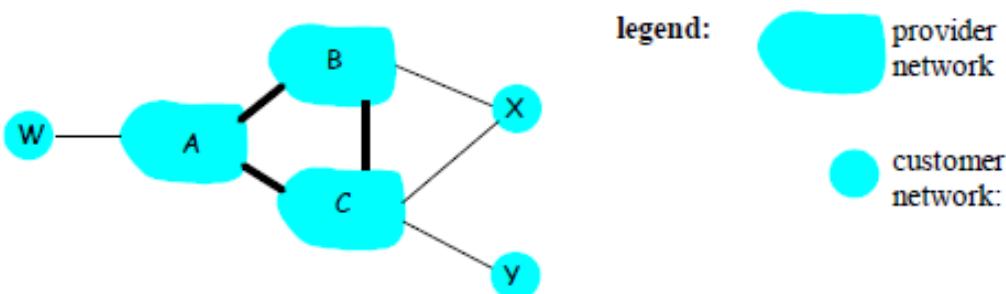
### BGP

- In una sessione eBGP tra i gateway 3a e 1c, AS3 invia ad AS1 la lista di prefissi raggiungibili:
  - 1c utilizza le proprie sessioni iBGP per distribuire i prefissi agli altri router del sistema autonomo.
  - Anche AS1 e AS2 si scambiano informazioni sulla raggiungibilità dei prefissi attraverso i propri gateway 1b e 2a.
- Quando un router viene a conoscenza di un nuovo prefisso, lo memorizza in una nuova riga della propria tabella d'inoltro.



### Politiche d'instradamento BGP

- A, B, C sono reti di provider di dorsale
- X, W, Y sono reti stub
- X è una rete stub multihomed:
  - X non vuole che il traffico da B a C le passi attraverso
  - ... e così X non annuncerà a B la rotta verso C



**Figure 4.5-BGPnew: a simple BGP scenario**

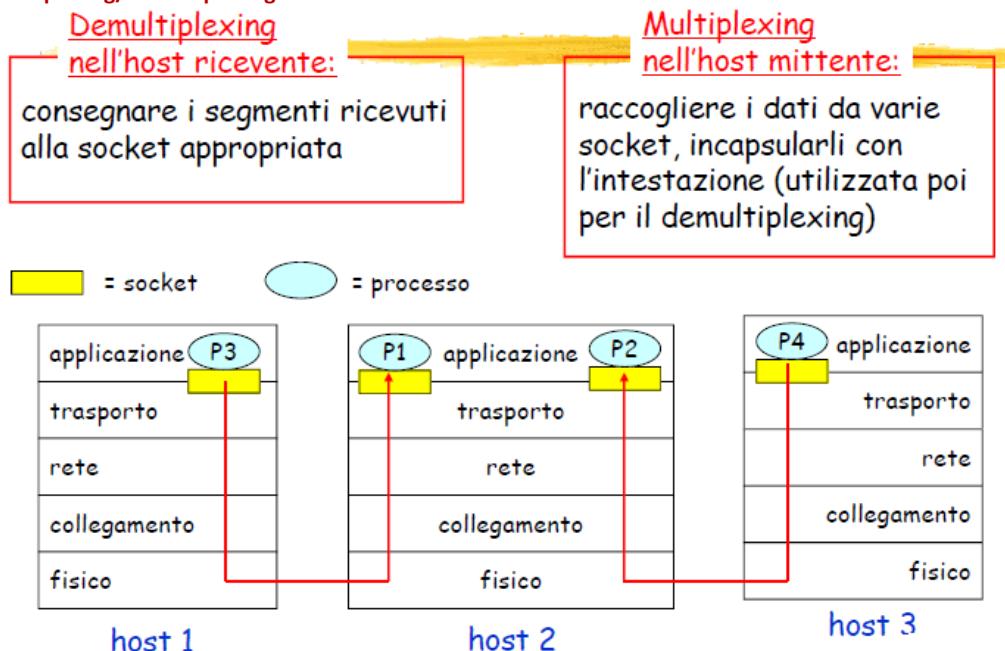
- A annuncia a B del percorso AW
- B annuncia a X del percorso BAW
- B deve annunciare a C del percorso BAW?
  - No! B non ha nessun “interesse” nella rotta CBAW poiché né W né C sono clienti di B
  - B vuole costringere C ad instradare verso W attraverso A
  - B vuole instradare solo da/verso i suoi clienti

## Protocolli inter-AS vs. protocolli intra-AS

- **Politiche:**
  - **Inter-AS:** il controllo amministrativo desidera avere il controllo su come il traffico viene instradato e su chi instrada attraverso le sue reti.
  - **Intra-AS:** unico controllo amministrativo, e di conseguenza le questioni di politica hanno un ruolo molto meno importante nello scegliere le rotte interne al sistema
- **Scala:**
  - L'instradamento gerarchico fa "risparmiare" sulle tabelle d'instradamento, e riduce il traffico dovuto al loro aggiornamento
- **Prestazioni:**
  - Intra-AS: orientato alle prestazioni
  - Inter-AS: le politiche possono prevalere sulle prestazioni

## Strato di Trasporto "TCP": Multiplazione a livello di trasporto

### Multiplexing/demultiplexing



### Demultiplexing

- L'host riceve i pacchetti IP:
  - Ogni pacchetto ha un indirizzo IP di origine e un indirizzo IP di destinazione
  - Ogni pacchetto trasporta 1 segmento a livello di trasporto
  - Ogni segmento ha un numero di porta di origine e un numero di porta di destinazione
- L'host usa gli indirizzi IP e i numeri di porta per inviare il segmento alla socket appropriata

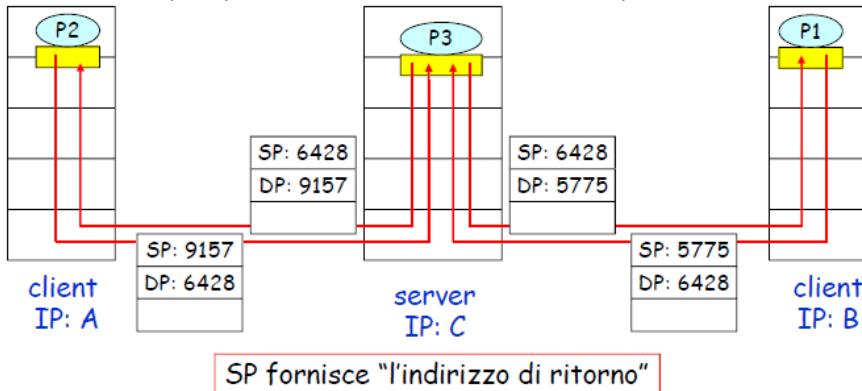


## Struttura del segmento TCP/UDP

### Demultiplexing senza connessione

- Un Host crea le socket con i numeri di porta
- Una socket UDP è identificata da 2 parametri:
  - (indirizzo IP di destinazione, numero di porta di destinazione)
- Quando l'host riceve il segmento UDP:
  - Controlla il numero della porta di destinazione nel segmento
  - Invia il segmento UDP alla socket con quel numero di porta
- Pacchetti IP con indirizzi IP di origine e/o numeri di porta di origine differenti vengono inviati alla stessa socket

- Il server C crea per il processo P3 una socket con il numero di porta 6428



#### Demultiplexing orientato alla connessione

- Una socket TCP è identificata da 4 parametri:

- Indirizzo IP di origine
- Numero di porta di origine
- Indirizzo IP di destinazione
- Numero di porta di destinazione

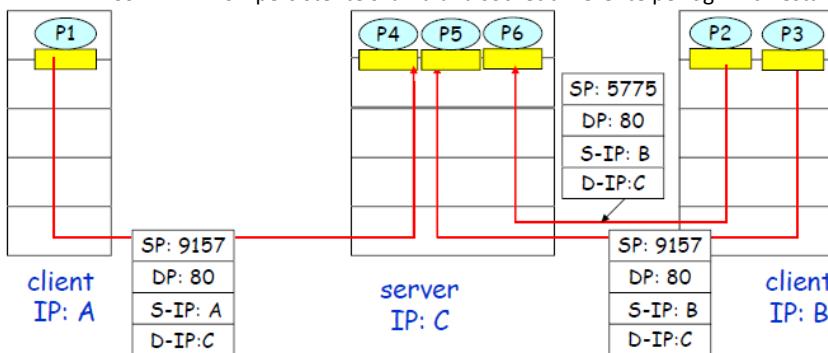
• L'host ricevente usa i quattro parametri per inviare i segmenti alla socket appropriata

• Un host server può supportare più socket TCP contemporaneamente:

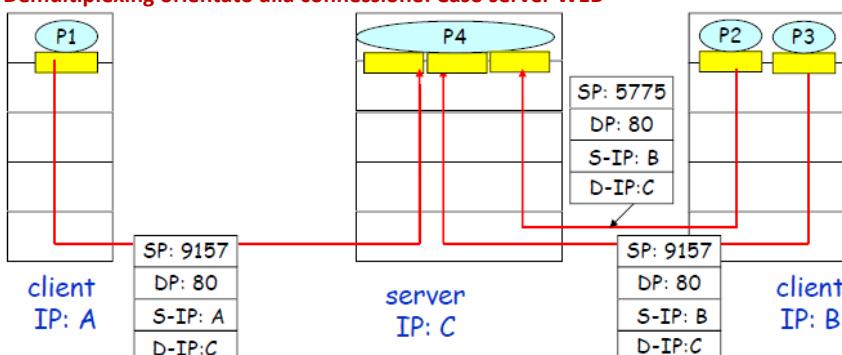
- Ogni socket è identificata dai suoi 4 parametri

• I server web hanno socket differenti per ogni connessione client:

- Con HTTP non-persistente si avrà una socket differente per ogni richiesta



#### Demultiplexing orientato alla connessione: Caso server WEB



#### Indirizzamento

- Statico:

- Le applicazioni più diffuse hanno dei numeri di porta assegnati (**well-known port numbers**)
  - Intervallo: 0 - 1023
- L'elenco dei port number è gestito dalla IANA ([www.iana.org](http://www.iana.org)) ed aggiornato in tempo reale

| Numero | Applicazione                          |
|--------|---------------------------------------|
| 7      | Echo                                  |
| 21     | FTP (File Transfer Protocol)          |
| 23     | TELNET                                |
| 25     | SMTP (Simple Mail Transport Protocol) |

| Numero | Applicazione                     |
|--------|----------------------------------|
| 37     | Time                             |
| 53     | Domain Name Server               |
| 80     | HTTP                             |
| 119    | NNTP (USENET New Transfer Prot.) |

- Dinamico:

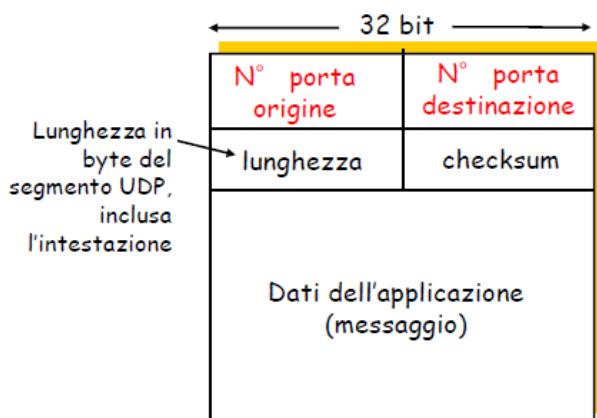
- Sono identificativi assegnati direttamente dal sistema operativo al momento dell'apertura della connessione
- Si utilizzano valori maggiori di 1023

## UDP: User Datagram Protocol [RFC 768]

- Protocollo di trasporto “semplice”
- I segmenti UDP possono essere:
  - Perduti
  - Consegnati fuori sequenza all’applicazione
- Senza connessione:
  - No handshaking tra mittente e destinatario UDP
  - Ogni segmento UDP è gestito indipendentemente dagli altri
- Senza controllo di congestione:
  - UDP può sparare dati a senza controllo

### UDP

- Utilizzato spesso nelle applicazioni multimediali:
  - Tollera piccole perdite
  - Sensibile alla frequenza
- Altri impieghi di UDP:
  - DNS
  - SNMP
- Trasferimento affidabile con UDP:
  - Aggiungere affidabilità al livello di applicazione
  - Recupero degli errori delle applicazioni



Struttura del segmento UDP

## Il protocollo TCP

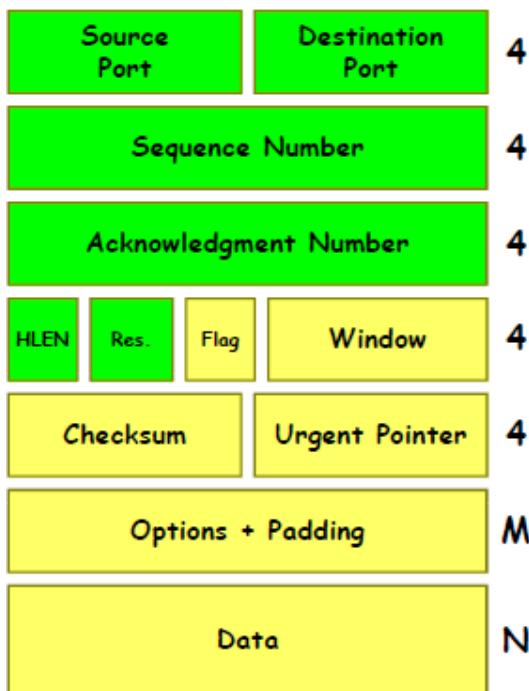
- È un protocollo con connessione (RFC 793, 1122, 1323, 2018, 2581)
- Interpreta il flusso di dati proveniente dallo strato applicativo come sequenza di ottetti
- Funzioni:
  - Indirizzamento di una specifica applicazione
  - Controllo di sequenza delle unità informative
  - Controllo e recupero di errore
  - Controllo di flusso
  - Controllo di congestione



### Segmento TCP

- Source Port e Destination Port (16 bit ciascuno)
- Sequence Number (32 bit):
  - Numero d’ordine del primo byte di dati contenuto nel campo dati
- Acknowledgment Number (ACKNUM) (32 bit):
  - Contiene un valore valido se il bit ACK del campo Flag è uguale a 1
  - Contiene il numero di sequenza del prossimo byte che l’entità ricevente si aspetta di ricevere
- HLEN (4 bit):
  - Contiene il numero di parole di 32 bit
  - Contenute nell’intestazione del segmento
  - L’intestazione del segmento non supera i 60 ottetti ed è sempre un multiplo di 32
- Reserved (6 bit):
  - Riservato per usi futuri

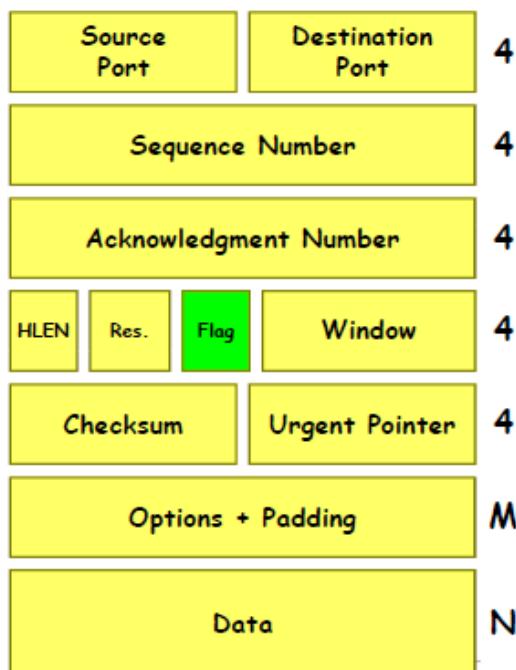
## Byte



- **Flag (6 bit):**

- URG: è uguale a uno quando il campo "Urgent Pointer" contiene un valore significativo
- ACK: è uguale a uno quando il campo "Acknowledgement Number" contiene un valore valido
- PSH: è uguale a uno quando l'applicazione indica che i dati vengano consegnati all'applicazione ricevente prescindendo dal riempimento dei buffer di ricezione
- RST: è uguale a uno in caso di richiesta di re-inizializzazione della connessione
- SYN: è uguale a uno solo nel primo segmento inviato durante la fase di sincronizzazione fra le entità TCP
- FIN: è uguale a uno quando la sorgente ha esaurito i dati da trasmettere

## Byte



- **Window (16 bit):**

- Larghezza della finestra misurata in ottetti
- è il numero di ottetti che, ad iniziare dal valore di ACK Number, l'emittitore del segmento autorizza a trasmettere

- **Checksum (16 bit):**

- Protegge l'intero segmento più alcuni campi dell'header IP (pseudo header)

- **Urgent Pointer (16 bit):**

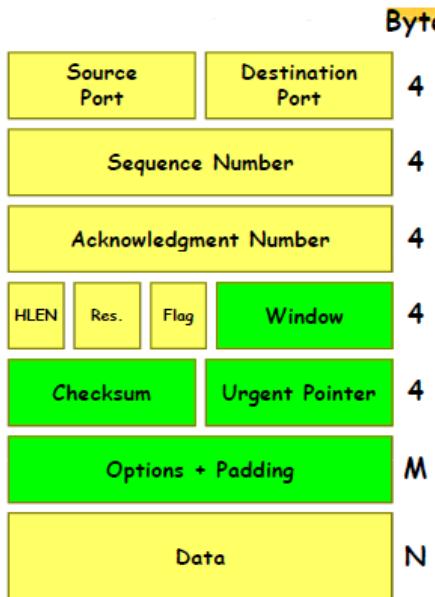
- Contiene il numero di sequenza dell'ultimo byte dei dati che devono essere consegnati urgentemente al processo ricevente

- **Options (di lunghezza variabile):**

- Sono presenti solo raramente

- **Padding (di lunghezza variabile):**

- Impone che l'intestazione abbia una lunghezza multipla di 32 bit



### Gestione della connessione

- Il protocollo TCP è un protocollo del tipo con connessione

- Nella fase di instaurazione della connessione le due entità TCP remote si sincronizzano scambiandosi:

- Gli identificatori dei socket (port, IP address)
- Il proprio numero di sequenza iniziale, che rappresenta il numero a partire dal quale tutti gli ottetti emessi saranno sequenzialmente numerati
- Il valore iniziale della finestra di ricezione

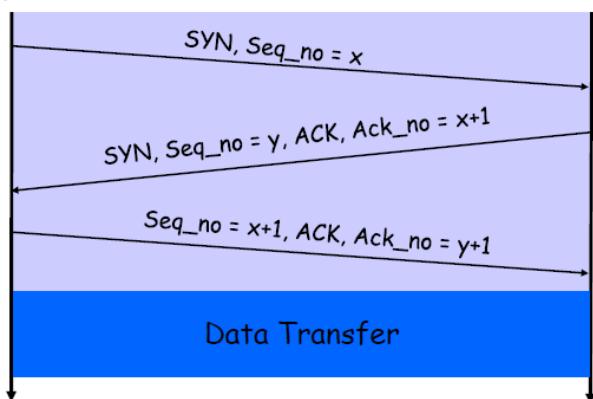
- Three Way Handshake:

- Passo 1: L'host A invia un segmento SYN all'host B:
  - Specifica il numero di sequenza iniziale nessun dato
- Passo 2: L'host B riceve SYN e risponde con un segmento SYN ACK
  - L'host B alloca i buffer
  - Specifica il numero di sequenza iniziale del server
- Passo 3: L'host A riceve un segmento SYN ACK e risponde con un segmento ACK, che può contenere dati

### Three-way handshake

Host A

Host B

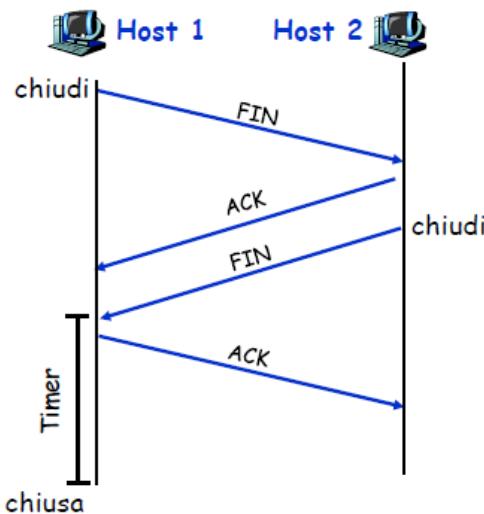


### Maximum Segment Size (MSS)

- Quando l'entità TCP emittente invia la prima TCP-PDU (SYN) può inserire l'informazione relativa alla massima dimensione del campo dei dati di utente di una TCP-PDU (**Maximum Segment Size - MSS**)
- L'entità ricevente risponde comunicando la propria MSS
- Nel caso di uno scambio bidirezionale, la dimensione della MSS è scelta in modo indipendente nei due versi e può quindi essere diversa nelle due direzioni

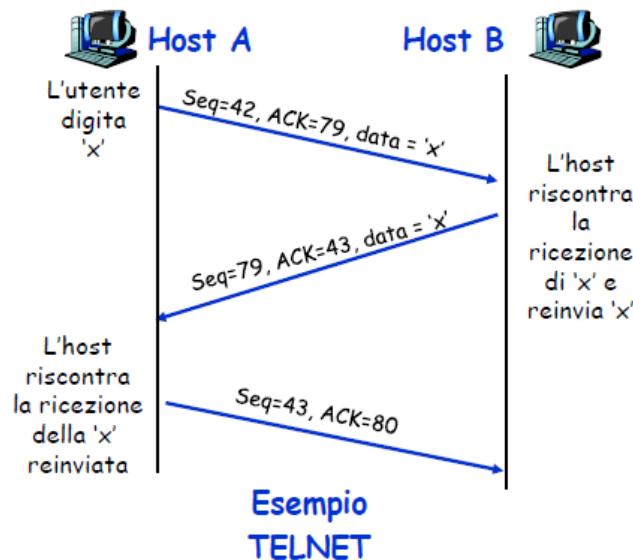
### Chiusura di una connessione

- L'Host 1 invia un segmento di controllo FIN al server
- L'Host 2 riceve il segmento FIN, risponde con un ACK
- L'Host 2 chiude la connessione e invia un FIN
- L'Host 1 riceve FIN e risponde con un ACK
- Viene attivato un timer:
  - Si risponde con un ACK ai FIN ricevuti
- L'Host 2 riceve un ACK:
  - La connessione viene chiusa



#### Numeri di sequenza e ACK

- **Numeri di sequenza:**
  - "Numero" del primo byte del segmento nel flusso di byte
- **ACK:**
  - Numero di sequenza del prossimo byte atteso dall'altro lato
  - ACK cumulativo
- **La gestione dei segmenti fuori sequenza non è specificata dallo standard:**
  - Dipende dall'implementazione (Es. Scarto, Memorizzazione)



#### Controllo d'errore

- In TCP il controllo d'errore è basato sull'impiego di:
  - Una codifica a rivelazione d'errore che:
    - è effettuata dall'entità TCP emittente e il cui risultato è inserito nell'intestazione del segmento (Checksum)
    - è utilizzata dall'entità TCP ricevente per la rivelazione di eventuali errori
- Riscontri positivi (ACK), che possono essere inoltrati dall'entità TCP ricevente con segmenti vuoti (senza dati) ovvero in modalità "piggybacking"
- Retransmission Timeout (RTO):
  - È un temporizzatore adattativo attivato dall'entità emittente
  - È attivato nel momento in cui un segmento viene inoltrato su una connessione uscente (**il timer è associato all'ultimo segmento non riscontrato**)
  - È disattivato nel momento in cui viene ricevuto un ACK relativo al segmento corrispondente e quando tale ricezione avviene prima che l'RTO si esaurisca

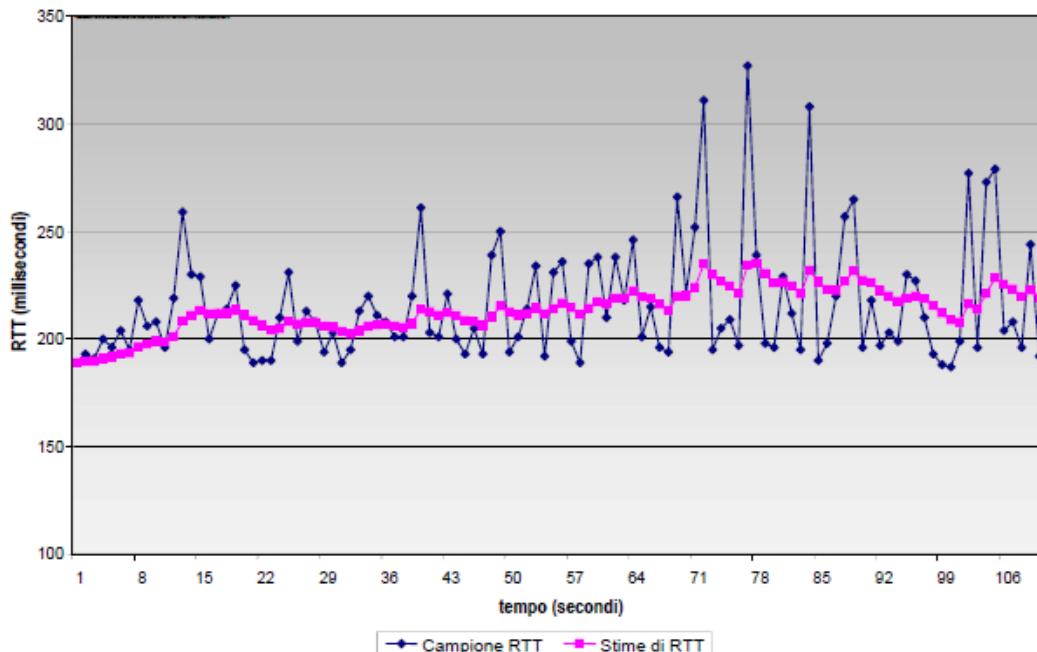
#### Riscontri

- L'entità TCP ricevente può emettere i riscontri (ACK) secondo due modalità:
  - **Immediata**, appena vengono accettati i dati, emette immediatamente un segmento vuoto (senza dati) che contiene l'appropriato numero di riscontro
  - **Cumulativa**, appena vengono accettati i dati, tiene memoria della necessità di inviare un riscontro, ma aspetta un segmento in uscita nel quale inserirlo:
    - Per evitare lunghi ritardi, attiva un timer di finestra
    - Se il tempo di questo timer si esaurisce prima che venga inviato un riscontro, emette un segmento vuoto che contiene l'appropriato numero di riscontro

### Round Trip Time (RTT) e timeout

- Come impostare il valore del timeout di TCP?
- Più grande di RTT:
  - Ma RTT varia
- Troppo piccolo: timeout prematuro:
  - Ritrasmissioni non necessarie
- Troppo grande: reazione lenta alla perdita dei segmenti
- Come stimare RTT?
- **SampleRTT:** tempo misurato dalla trasmissione di un segmento fino alla ricezione dell'ACK relativo:
  - Ignora le ritrasmissioni
- SampleRTT varia, quindi occorre una stima "smoothed" di RTT:
  - Media di più misure recenti, non semplicemente il valore corrente di SampleRTT
- **EstimatedRTT = (1 -  $\alpha$ )\*EstimatedRTT +  $\alpha$  \*SampleRTT**
- Media mobile esponenziale ponderata
- L'influenza dei vecchi campioni decresce esponenzialmente
- Valore tipico:  $\alpha = 0,125$

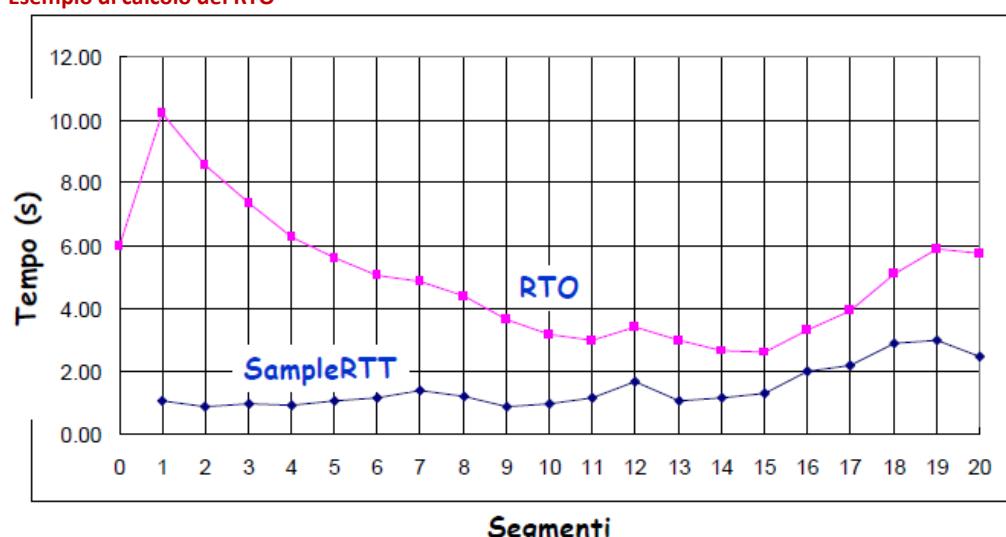
### Esempio di stima di RTT



### Determinazione del Timeout

- EstimatedRTT più un "margin di sicurezza":
  - Grande variazione di EstimatedRTT -> margine di sicurezza maggiore
- **Stima della deviazione standard dell'EstimatedRTT:**
  - $DevRTT = (1 - \beta) * DevRTT + \beta * |SampleRTT - EstimatedRTT|$
  - $\beta = 0,25$
- **Valore Retransmission TimeOut (RTO):**  
 $RTO = EstimatedRTT + 4 * DevRTT$

### Esempio di calcolo del RTO



### Exponential RTO Backoff

- Determina il valore di RTO associato ad un segmento riemesso:
  - è consigliabile variare il valore di RTO sui segmenti riemessi perché l'esaurimento dell'RTO è dovuto a congestione in rete
  - è consigliabile variare il valore di RTO delle sorgenti che sono coinvolte nella congestione per evitare riemissioni contemporanee
- Una sorgente TCP aumenta il valore di RTO per ogni riemannessione (exponential backoff process) (normalmente q=2)

$$RTO_{i+1} = q \cdot RTO_i$$

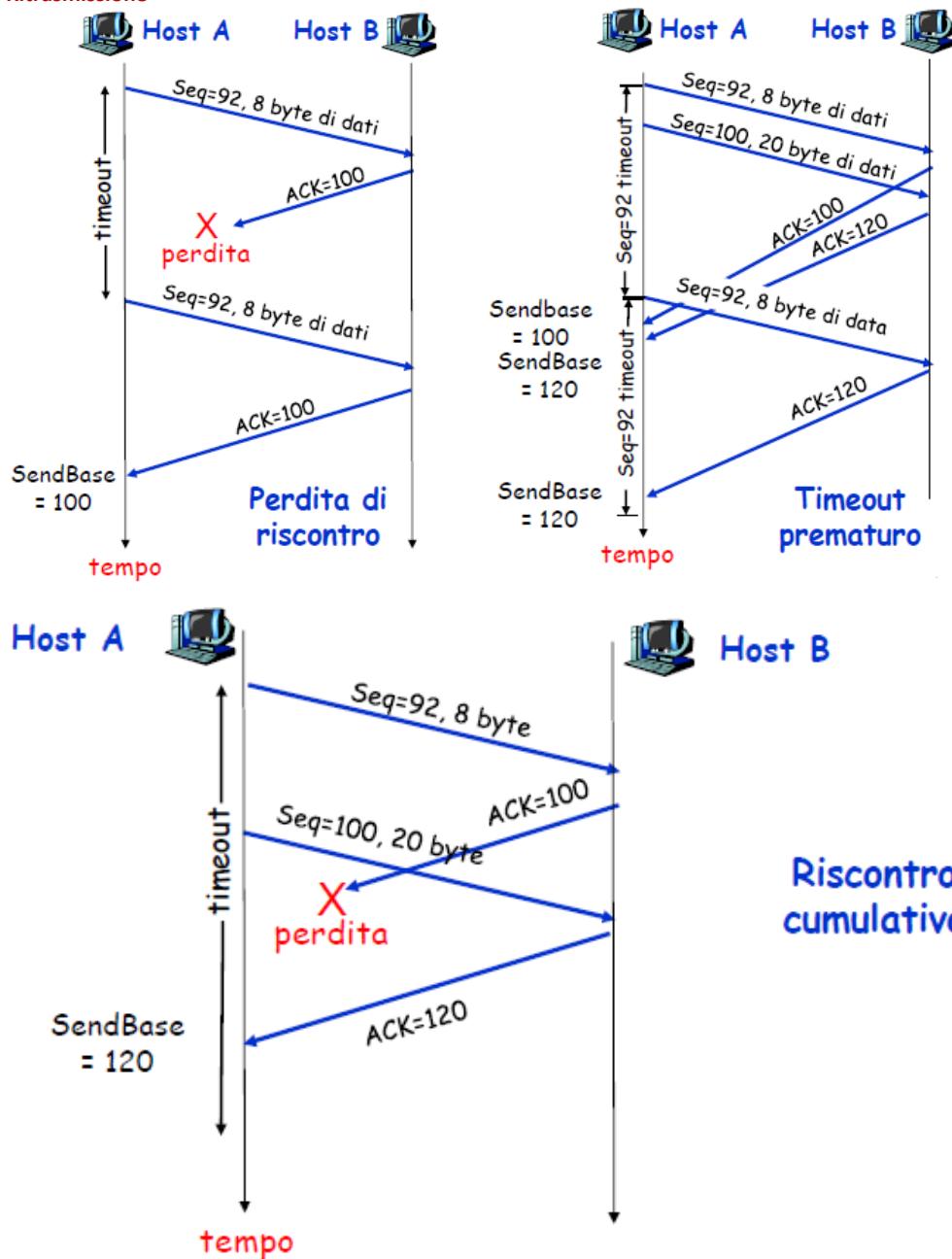
### Algoritmo di Karn

- L'entità TCP ricevente non distingue se il riscontro si riferisce:
  - Alla prima emissione del segmento (RTO troppo elevato con perdita di efficienza e inutili ritardi)
  - Alla riemannessione del segmento (RTO troppo breve e quindi riemannessioni eccessive e nuovi errori di misura).
- L'algoritmo di Karn stabilisce di:
  - Non considerare il RTT dei segmenti riemessi
  - Usare come RTO il valore dato dalla procedura di exponential backoff
  - Ricalcolare il nuovo valore di RTO solo al momento della ricezione di un ACK di un segmento **non riemesso**

### Controllo d'errore

- TCP ha lo scopo di offrire un servizio di trasferimento dati affidabile utilizzando il servizio inaffidabile offerto dallo strato di rete (IP)
- Si utilizzano solo segmenti ACK
- Un solo timeout di ritrasmissione
- Le ritrasmissioni sono avviate da:
  - Esauremento del timeout
  - ACK duplicati

### Ritrasmissione

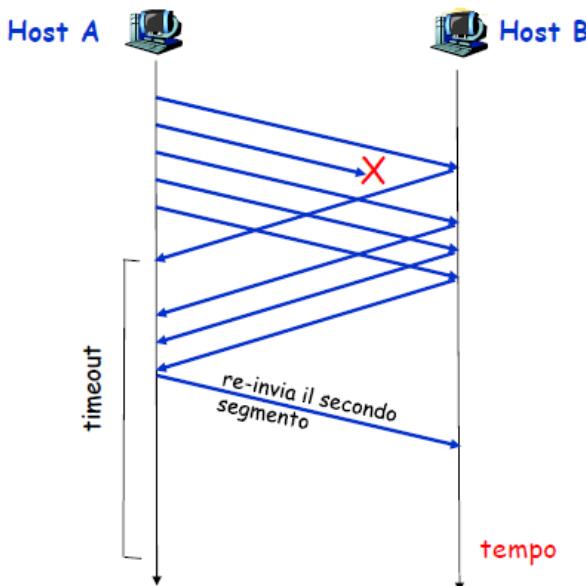


### Generazione di ACK [RFC 1122, RFC 2581]

| Evento presso il ricevente                                                                                             | Azione del ricevente                                                                                         |
|------------------------------------------------------------------------------------------------------------------------|--------------------------------------------------------------------------------------------------------------|
| Arrivo ordinato di un segmento. Tutti i dati fino al numero di sequenza atteso sono già stati riscontrati              | ACK ritardato. Attende fino a 500 ms l'arrivo del prossimo segmento. Se il segmento non arriva, invia un ACK |
| Arrivo ordinato di un segmento. Un altro segmento è in attesa di trasmissione dell'ACK                                 | Invia immediatamente un singolo ACK cumulativo, riscontrando entrambi i segmenti                             |
| Arrivo non ordinato di un segmento con numero di sequenza superiore a quello atteso Viene rilevato un "fuori sequenza" | Invia immediatamente un ACK duplicato, indicando il numero di sequenza del prossimo byte atteso              |
| Arrivo di un segmento che ripristina parzialmente o completamente il "fuori sequenza"                                  | Invia immediatamente un ACK, ammesso che il segmento sia sequenza con l'ultimo segmento riscontrato          |

### Fast retransmit

- Il periodo di timeout spesso è relativamente lungo:
  - Elevato ritardo prima di ritrasmettere il pacchetto perduto
- L'entità TCP emittente può rivelare precocemente i segmenti perduti tramite l'analisi degli ACK duplicati:
  - L'entità TCP emittente spesso invia molti segmenti
  - Se un segmento viene smarrito, è probabile che ci saranno molti ACK duplicati
- Se l'entità TCP emittente riceve **3 ACK duplicati** per lo stesso dato, suppone che il segmento che segue il dato riscontrato sia andato perduto:
  - Ritrasmissione rapida
  - Si ritrasmette il segmento prima che scada il timer

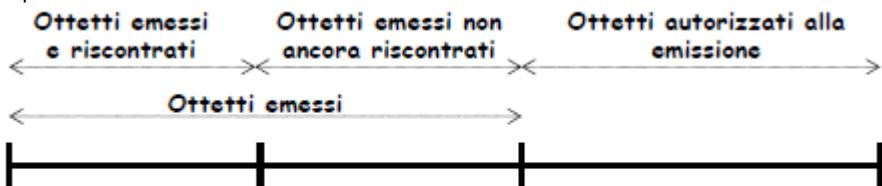


### Controllo di flusso

- Il controllo di flusso ha lo scopo di limitare il ritmo di emissione dei dati da parte di un host per evitare la saturazione della capacità del buffer di ricezione
- TCP utilizza un controllo di flusso basato su una **finestra scorrevole di larghezza variabile**:
  - Lo scorrimento e la larghezza della finestra sono controllati dall'entità TCP ricevente
- Il controllo di flusso opera a livello di ottetti (byte)**:
  - Gli ottetti sono numerati in sequenza a partire dal numero scelto durante il 3-way handshaking (procedura di instaurazione della connessione)
- La procedura di controllo di flusso TCP utilizza i seguenti parametri:
  - SN (Sequence Number)**:
    - SN si riferisce al primo ottetto contenuto nel segmento
  - AckN (Acknowledgement Number)**:
    - AN si riferisce al prossimo ottetto che l'entità ricevente aspetta di ricevere
  - RecWindow (Window)**:
    - WD esprime il numero massimo di ottetti che l'entità emittente può trasmettere consecutivamente senza ricevere riscontro per alcuno di questi
- Un riscontro (**AckN = X e RecWindow = W**) significa che:
  - Sono riscontrati tutti gli ottetti ricevuti fino a quello numerato con X-1
  - L'entità TCP emittente è autorizzata a trasmettere fino a ulteriori W ottetti, ovvero fino all'ottetto numerato con X+W-1

### Controllo della Finestra

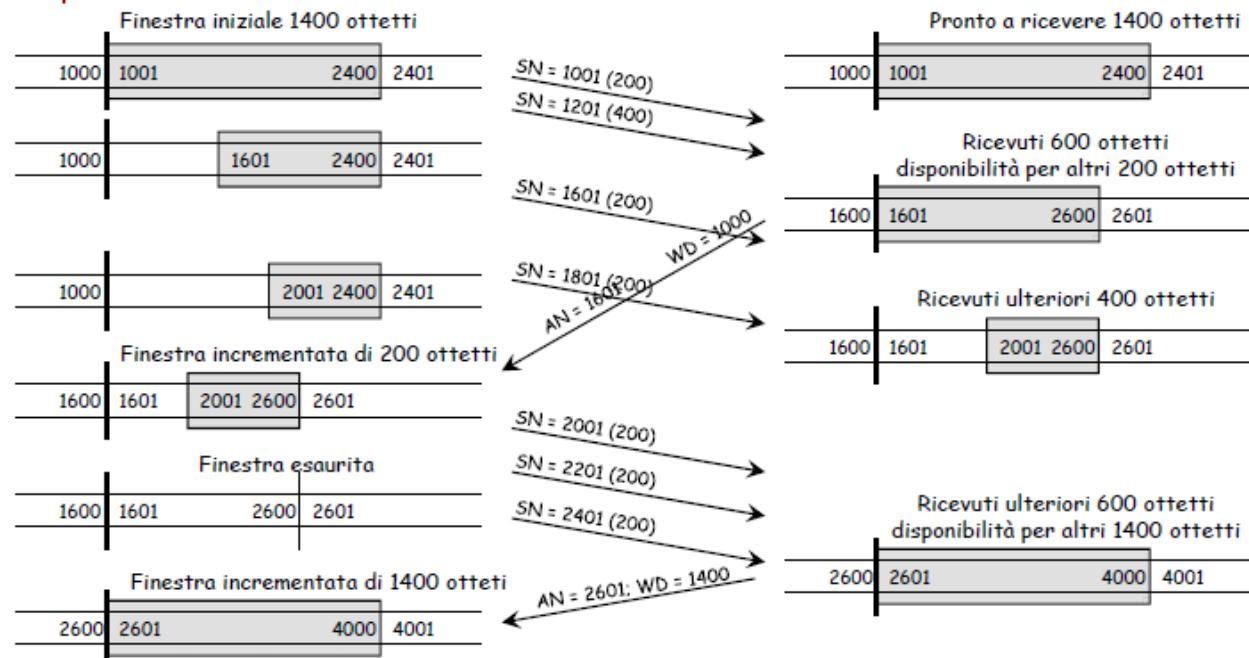
- Puntatori per il controllo a finestra lato emittente:



- Puntatori per il controllo a finestra lato ricevente:



### Esempio



### Throughput di una connessione TCP

- Il throughput (TH) di una connessione TCP, nell'ipotesi di overhead nullo e di assenza di ritrasmissioni, è dato da:

$$TH = \begin{cases} 1 & \text{se } W \geq \lceil 2\alpha + 1 \rceil \\ \frac{W}{2\alpha + 1} & \text{se } W < \lceil 2\alpha + 1 \rceil \end{cases}$$

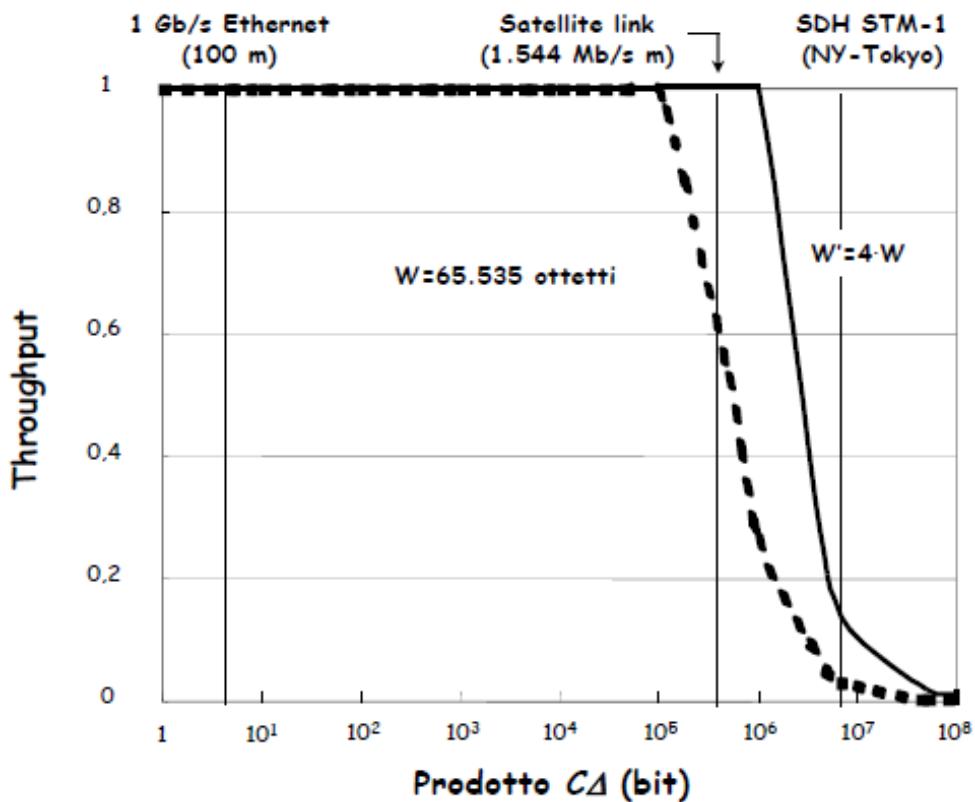
- Dove:

- C è il ritmo binario della connessione
- W è la larghezza della finestra
- Δ è il ritardo di propagazione sulla connessione
- $\alpha = C \Delta / 8$  (rapporto tra ritardo di propagazione e tempo di trasmissione di un ottetto)

- Se si suppone  $2\alpha \gg 1$ , risulta allora:

$$TH = \begin{cases} 1 & \text{se } W \geq C\Delta / 4 \\ \frac{4W}{C \cdot \Delta} & \text{se } W < C\Delta / 4 \end{cases}$$

- In funzione della larghezza della finestra W (in ottetti) e del prodotto banda ritardo C Δ (in bit)



### Principi del controllo di congestione

- **Definizione:**
  - "Troppe sorgenti trasmettono troppi dati, a una rate che la rete non è in grado di gestire"
- **Il controllo di congestione è diverso dal controllo di flusso:**
  - Il controllo di congestione riguarda la rete
  - Il controllo di flusso riguarda il ricevente
- **Sintomi:**
  - Pacchetti persi (overflow nei buffer dei router)
  - Elevati ritardi (accodamento nei buffer dei router)

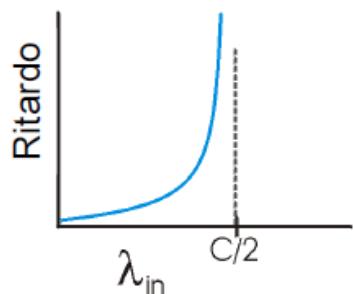
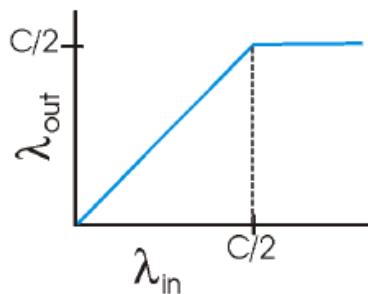
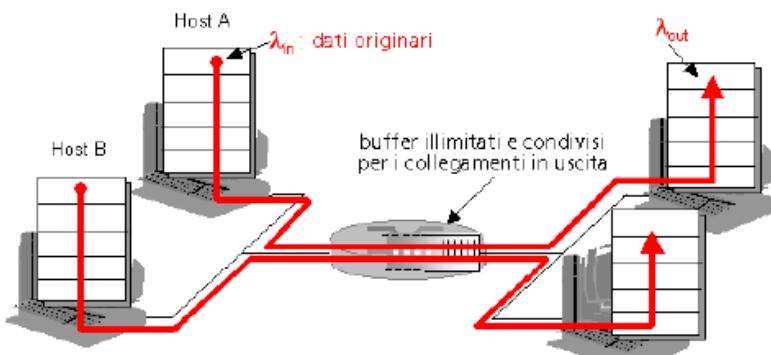
### Esempio: scenario 1

- Due mittenti, due destinatari
- Buffer illimitati
- Nessuna ritrasmissione

**Due mittenti,  
due destinatari**

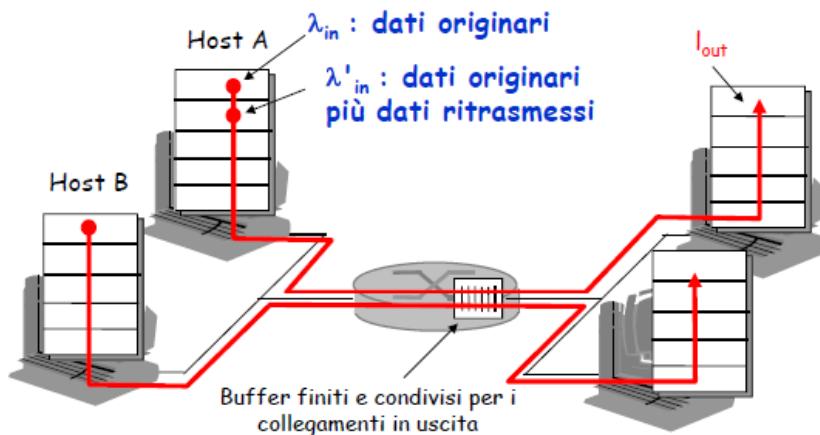
**Buffer illimitati**

**Nessuna  
ritrasmissione**

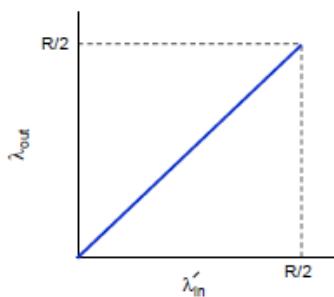


### Esempio: scenario 2

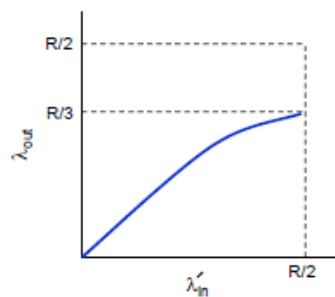
- Buffer finiti
- Il mittente ritrasmette i pacchetti perduti



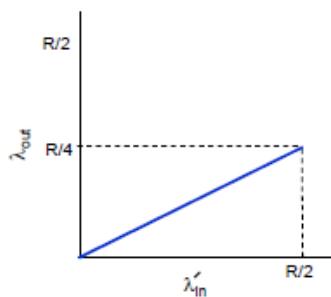
**a - Assenza congestione**



**b - Ritrasmissioni solo per perdita per congestione**



**c - Ritrasmissioni anche per ritardi eccessivi**



- $\lambda_{in} = \lambda_{out} = \text{goodput}$

- Rate di arrivo di pacchetti utili a destinazione

- Caso a: tutti i pacchetti arrivano a destinazione ( $\lambda'_{in} = \lambda_{in} = \lambda_{out}$ )

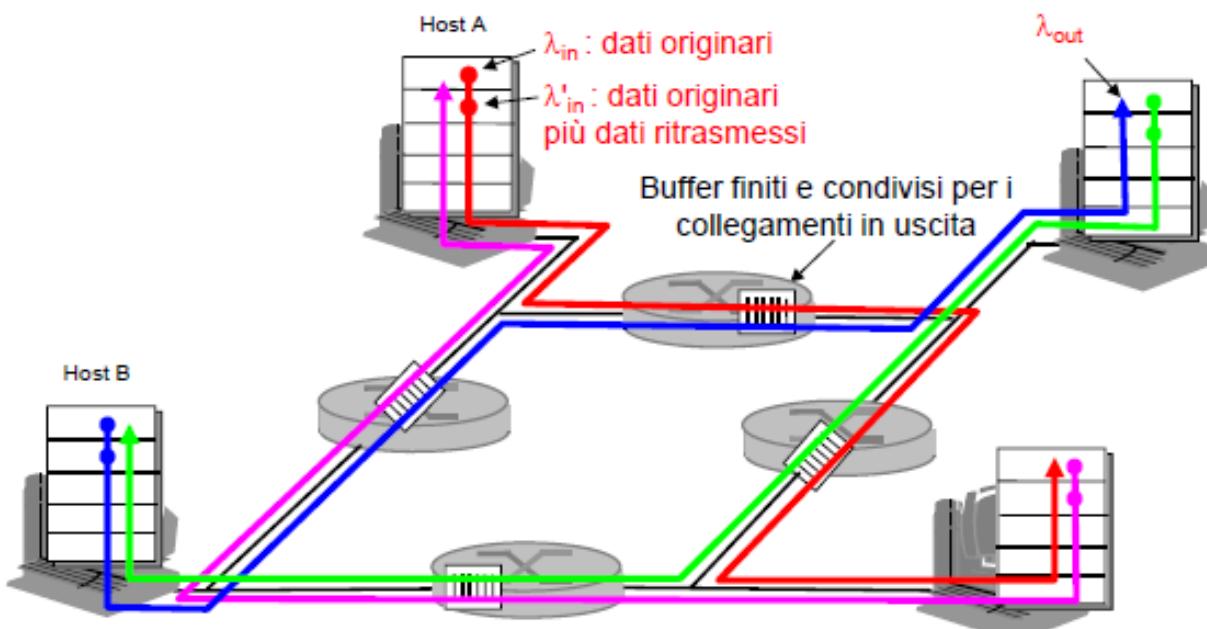
- Caso b: le ritrasmissioni rendono  $\lambda'_{in}$  maggiore di  $\lambda_{out}$  quindi il goodput

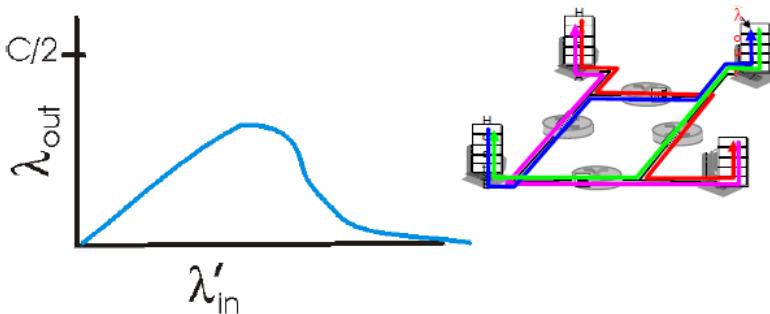
$\lambda_{out}$  diminuisce ( $\lambda'_{in} > \lambda_{in} = \lambda_{out}$ )

- Caso c: la ritrasmissione dei pacchetti ritardati aumenta ancora  $\lambda'_{in}$  ed il goodput diminuisce ulteriormente ( $\lambda'_{in} > \lambda_{in} = \lambda_{out}$ )

### Esempio: scenario 3

- Quattro mittenti
- Percorsi multihop
- Timeout/ritrasmissione





- Un altro “costo” della congestione:
- Quando il pacchetto viene scartato, la capacità trasmissiva utilizzata sui collegamenti di upstream per instradare il pacchetto risulta sprecata

### Approcci al controllo della congestione

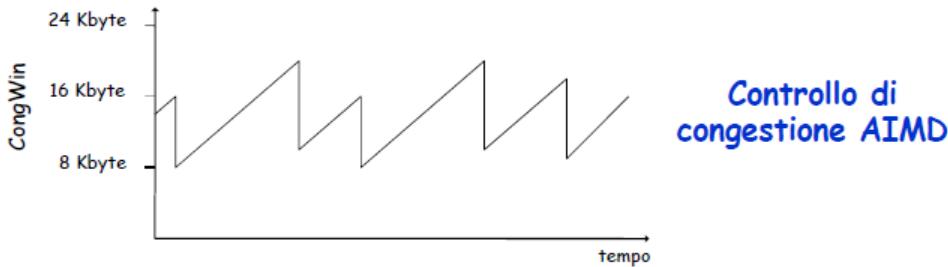
- Controllo di congestione punto-punto
- Nessun supporto esplicito dalla rete
- La congestione è dedotta osservando le perdite e i ritardi nei sistemi terminali
- Metodo adottato da TCP
- Controllo di congestione assistito dalla rete
- I router forniscono un feedback ai sistemi terminali:
  - Un singolo bit per indicare la congestione (SNA, DECbit, TCP/IP ECN, ATM)
  - Comunicare in modo esplicito al mittente la frequenza trasmissiva

### Controllo di congestione TCP

- Il protocollo TCP utilizza i seguenti meccanismi:
  - L'esaurimento dell'RTO come un sintomo di congestione
  - **La finestra di congestione (Congestion Window - Congwin)**
    - La finestra di congestione si affianca alla finestra di ricezione operante nel controllo di flusso e impone una limitazione addizionale alla quantità di traffico che un host può inviare in una connessione
  - **La soglia (Threshold):**
    - Il valore della soglia è pari alla metà del valore della Congwin al momento in cui viene rilevata una perdita
    - All'inizio della connessione (slow start) la soglia è posta uguale a  $\infty$
- L'entità emittente determina nel tempo il valore della finestra disponibile (**Available Window - Awdn**):
  - Awdn = numero di segmenti di lunghezza massima (MSS) che possono essere inviati senza riscontro
- **Il valore di Awdn:**
  - Non deve superare il minimo tra le larghezze **Congwin** della finestra di congestione e **RecWindow** della finestra di ricezione
  - $$\text{Awdn} \equiv \min \{\text{Congwin}, \text{RecWindow}\}$$
  - Congwin ed RecWindow sono quantità espresse in numero di segmenti MSS
  - RecWindow è la larghezza comunicata nell'ultimo ACK ricevuto e ottenuta dall'entità TCP emittente dividendo il numero contenuto nel campo Window di questo ACK per il numero di ottetti che compongono una MSS

### Additive-Increase Multiplicative-Decrease

- Aumenta il valore di CongWin (sondando la rete) fino a quando non si verifica una perdita
- **Incremento additivo:**
  - Aumenta CongWin di 1 MSS a ogni RTT in assenza di eventi di perdita
- **Decremento Moltiplicativo:**
  - Riduce a metà CongWin dopo un evento di perdita



### Controllo di congestione TCP

- Approssimativamente il rate di emissione dei segmenti è dato da:

$$\text{Frequenza d'invio} = \frac{\text{CongWin}}{\text{RTT}} \text{ byte/sec}$$

- CongWin è una funzione dinamica della congestione percepita
- Il mittente percepisce la congestione se:
  - Esaurimento timeout
  - Ricezione di 3 ACK duplicati
- Il mittente TCP riduce la frequenza d'invio (CongWin) dopo un evento di perdita

## Fasi della procedura

- Per evitare la congestione, l'emettitore TCP segue una procedura ciclica in cui ogni ciclo è composto da due fasi:
  - **Slow Start:**
    - Incremento esponenziale della Congwin
  - **Congestion Avoidance:**
    - Incremento lineare della Congwin

## Slow start

- Quando si stabilisce una connessione:

- CongWin = 1 MSS
- Soglia =  $\infty$
- Esempio: MSS = 500 byte, RTT = 200 msec
- Frequenza iniziale = 20 kbps

- La larghezza di banda disponibile potrebbe essere >> MSS/RTT:

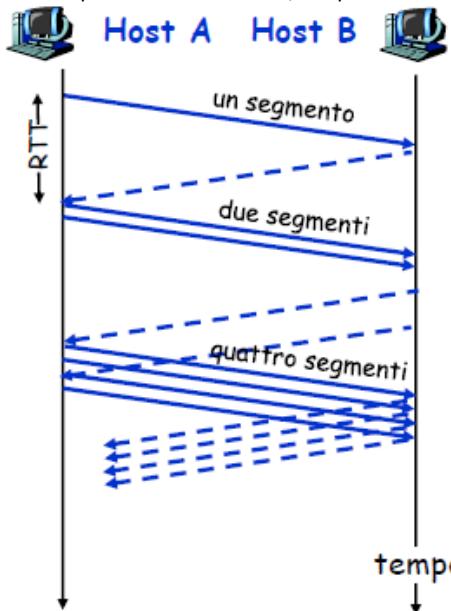
- Consente di raggiungere rapidamente una frequenza d'invio significativa

- Quando inizia la connessione, la frequenza aumenta in modo esponenziale, fino a quando non si verifica un evento di perdita
- Quando si verifica un evento di perdita si pone:

- CongWin(new) = 1 MSS
- Soglia = CongWin(old)/2

- Quando inizia la connessione, la frequenza aumenta in modo esponenziale, fino a quando non si verifica un evento di perdita:
  - Raddoppia CongWin a ogni RTT
  - Ciò avviene incrementando CongWin per ogni ACK ricevuto

- La frequenza iniziale è lenta, ma poi cresce in modo esponenziale



## Congestion Avoidance

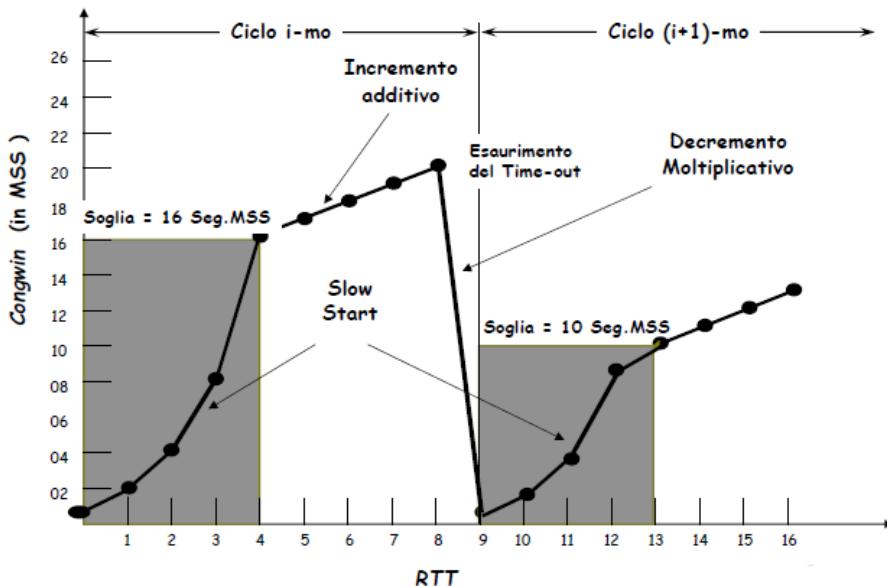
- Se l'aumento che si ha nella fase Slow Start raggiunge e supera il valore di soglia, e cioè se  $Congwin \geq Soglia$ , l'incremento di Congwin diventa lineare al crescere di RTT:
  - Se  $Congwin = w$  e se  $w \geq Soglia$ , dopo l'arrivo di  $w$  riscontri consecutivi, la larghezza Cwdn viene incrementata di 1 MSS in ciascun RTT in cui si registra l'arrivo di un intero gruppo di riscontri dei contenuti della finestra di congestione
- Questo incremento lineare continua finché i riscontri arrivano prima dei loro rispettivi RTO
- Questo aumento ha un limite superiore corrispondente al raggiungimento di uno stato di saturazione su uno dei collegamenti lungo il percorso o in uno dei nodi attraversati
- Nell'ipotesi che **Congwin < Recwindow**, il limite superiore dell'aumento della Congwin è determinato dal verificarsi di un evento di perdita di un segmento e di un conseguente raggiungimento del relativo RTO

## Decremento moltiplicativo

- Quando si verifica l'esaurimento di un RTO (evento di perdita di un segmento), inizia un nuovo ciclo
- Le operazioni effettuate sono le seguenti:
  - Il valore Soglia viene impostato a metà del valore attuale di Congwin ed è quindi ridotto esponenzialmente rispetto a quello massimo raggiunto al termine della prima fase
  - Il valore successivo di Congwin viene portato ad 1 MSS
  - L'incremento lineare continua finché i riscontri arrivano prima dei loro rispettivi RTO

## Procedura complessiva

- In conclusione, se si trascura la fase di slow start, una entità TCP emittente:
  - Incrementa Cwdn di 1 Seg.MSS per ogni RTT quando il suo percorso di rete non è congestionato
  - Diminuisce Cwdn di un fattore 2 per ogni RTT quando il percorso è congestionato
- Per questo motivo questa procedura di controllo di congestione è usualmente indicata come algoritmo di incremento additivo e di decremento moltiplicativo (**AIMD**, Additive-Increase, Multiplicative-Decrease)



- La procedura è illustrata nella figura seguente in cui:

- Il valore Soglia iniziale è uguale a 16 MSS
- Durante la fase di slow start, la Soglia è raggiunta all'istante 4
- Il valore di Congwin cresce poi linearmente, finché non si verifica una perdita (istante 8) e quando Congwin = 20 MSS
- Il valore Soglia è allora ridotto a 0,5 Congwin = 10 MSS e la finestra di congestione è successivamente posta a 1 MSS
- La fase di slow start ricomincia poi all'istante 9 e ha termine all'istante 13, quando Congwin ha raggiunto il valore 10 MSS
- Da quest'ultimo valore ricomincia l'incremento additivo di Congwin che avrà termine quando si verificherà una nuova perdita

### Fast recovery

- Dopo 3 ACK duplicati:
  - CongWin è ridotto a metà
  - La finestra poi cresce linearmente
- Dopo un evento di timeout:
  - CongWin è impostata a 1 MSS
  - La finestra cresce in modo esponenziale fino a un valore di soglia, poi cresce linearmente
- Spiegazione:
  - 3 ACK duplicati indicano la capacità della rete di consegnare qualche segmento
  - Un timeout prima di 3 ACK duplicati è "più allarmante"

### Riassunto

- Quando CongWin è sotto la soglia, il mittente è nella fase di **slow start**; la finestra cresce in modo esponenziale
- Quando CongWin è sopra la soglia, il mittente è nella fase di **congestion avoidance**; la finestra cresce in modo lineare
- Quando si verificano **tre ACK duplicati**, il valore di Soglia viene impostato a CongWin/2 e CongWin viene impostata al valore di Soglia
- Quando **scade il timeout**, il valore di Soglia viene impostato a CongWin/2 e CongWin è impostata a 1 MSS

### Controllo di congestione del mittente TCP

| Stato                     | Evento                                                    | Azione del mittente TCP                                                                               | Commenti                                                                                    |
|---------------------------|-----------------------------------------------------------|-------------------------------------------------------------------------------------------------------|---------------------------------------------------------------------------------------------|
| Slow Start (SS)           | Ricezione di ACK per dati precedentemente non riscontrati | $CongWin = CongWin + MSS$ ,<br>If ( $CongWin > Threshold$ ) imposta lo stato a "Congestion Avoidance" | CongWin raddoppia a ogni RTT                                                                |
| Congestion Avoidance (CA) | Ricezione di ACK per dati precedentemente non riscontrati | $CongWin = CongWin + MSS * (MSS/CongWin)$                                                             | Incremento additivo:<br>CongWin aumenta di 1 MSS a ogni RTT                                 |
| SS o CA                   | Rilevato un evento di perdita da tre ACK duplicati        | $Threshold = CongWin/2$ ,<br>$CongWin = Threshold$ , imposta lo stato a "Congestion Avoidance"        | Ripristino rapido con il decremento moltiplicativo.<br>CongWin non sarà mai minore di 1 MSS |
| SS o CA                   | Timeout                                                   | $Threshold = CongWin/2$ ,<br>$CongWin = 1 MSS$ , imposta lo stato a "Slow Start"                      | Entra nello stato "Slow Start"                                                              |
| SS o CA                   | ACK duplicato                                             | Incrementa il conteggio degli ACK duplicati per il segmento in corso di riscontro                     | CongWin e Threshold non variano                                                             |

### Throughput TCP

- Qual è il throughput medio di TCP in funzione della dimensione della finestra e di RTT?
  - Ignoriamo le fasi di slow start
  - Sia  $W$  la dimensione della finestra quando si verifica una perdita
- Quando la finestra è  $W$ , si ha:  
$$\text{TH1} = W / \text{RTT}$$
- Subito dopo la perdita, la finestra si riduce a  $W/2$ , quindi:  
$$\text{TH2} = W/2 * \text{RTT}$$
- Poiché l'aumento è lineare:  
$$\text{TH} = (\text{TH1} + \text{TH2})/2 = 0,75 W/\text{RTT}$$