**Appunti Reti**

Una rete di calcolatori è un grafo composto da vari nodi (ogni macchina connessa alla rete) questa viene appunto raffigurata attraverso un grafo di nodi che usano connessioni cablate o connessioni a radiofrequenza

Un protocollo definisce le regole di comunicazione, attraverso regole di astrazione attraverso vari livelli dal fisico al digitale

DTE = DATA TERMINAL EQUIPMENT

Reti **punto-a-punto**:

* **Caratteristiche:**
  + I DTE sono collegati tramite DCE (Data CircuitTerminating Equipment) e linee di trasmissione, queste sono collegati direttamente l’uno all’altro, è detto punto-a-punto quando sono collegati fisicamente soli 2 DTE
* **Vantaggi:** 
  + Semplicità nella gestione della rete
  + Tempi di attesa nulli
* **Svantaggi:** 
  + Costo:
    - prezzo del cavo
    - “costo” come perdita del segnale in quanto in base alla lunghezza del cavo il segnale si degrada logaritmicamente pathloss
    - in distanze molto lunghe potrebbe inficiare anche la velocità e produrre tempi di attesa allungati.
  + Non è facilmente scalabile all’aumentare dei DTE

Reti **Multipunto:**

* **Caratteristiche:**
  + Consiste nel collegare più DTE sulla stessa linea di trasmissione, venendo organizzati secondo una gerarchia master/slave, in cui un DTE master (o principale) comunica e gli slaves (o secondari) ascoltano in modo broadcast, ovvero tutti ascoltano quando un DTE comunica = tutte le comunicazioni sono indirizzate a tutti, sta ad ogni DTE vedere se accettare la comunicazione in baso ad un ID contenuto nel “messaggio” comunicato
* **Svantaggi**:
  + Possono nascere problemi di contesa per la comunicazione

Rete **Broadcast:**

* **Caratteristiche**:
  + È una rete composta da un solo bus di comunicazione a cui sono collegati più elaboratori, in cui ogni pacchetto è inviato a tutti e recepibile da ogni elaboratore. Ogni elaboratore riceve un pacchetto e ha il compito di leggere l’header e di elaborare solo quelli in cui combacia il loro indirizzo con il destinatario dell’header del pacchetto. Nel caso di un pacchetto che **deve** essere elaborato da tutti si parla di **broadcasting**. Mentre **multicast** nel caso il pacchetto sia indirizzato a più di un elaboratore ma non a tutti, e infine **unicast** nel caso sia indirizzato ad un singolo elaboratore.

**Pacchetto:**

* Composto da
  + un ***Header*** nel quale è indicato il destinatario delle informazioni
  + un ***Body*** nel quale sono contenute le informazioni da inviare

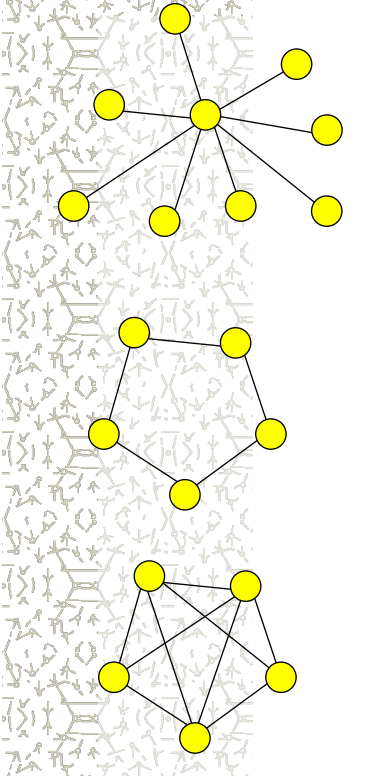
**Topologia di rete**: configurazione geometrica dei collegamenti dei componenti della rete, ne esistono vari tipi specializzati in vari campi, come:

- Massima Affidabilità (Tasso di guasti e ridondanza)

- Scalabilità (Margini di crescita)

- Alto Rendimento Complessivo (banda e latenza)

- Minimi Costi di startup ed esercizio

Topologia con **centro stella**

Svantaggio: unica connessione centrale su cui si fonda tutta la rete

n-1 collegamenti

Topologia **ad anello**

Svantaggio: latenza

n collegamenti

Topologia **a maglia**

Svantaggio: Troppi collegamenti e problemi di costo (leggi sopra)

n(n-1)/2 collegamenti

Rete **gerarchica** o **ad albero**:

* **Caratteristiche**:
  + I pacchetti viaggiano dalle foglie alla radice, attraverso i nodi intermedi, è la topologia più comune. Generalmente le foglie rappresentano nodi (elaboratori) con un compito singolo di elaborare le informazioni che vengono passate dai nodi intermedi
* **Vantaggi**:
  + Permette la scalabilità
* **Svantaggi**:
  + Il nodo principale se sovraccarico può avere l’effetto di un collo di bottiglia e rallentare in blocco tutta l’elaborazione
  + La rete dipende dal nodo principale, caduto quello, la rete smette di funzionare
    - Questo problema può essere tamponato seguendo una politica di back up periodici

Rete **a stella**:

* Caratteristiche:
  + Simile alla rete ad albero, escludendo però la disposizione funzionale dei nodi connessi alla rete
* Svantaggi:
  + Basata tutta sul nodo centrale, che una volta corrotto ferma tutta la rete, inoltre causa un bottle neck in caso venga sovraccaricato

Rete **a dorsale** (**bus condiviso**):

* Caratteristiche:
  + Adottata dalle connessioni di tipo Ethernet, analoga al concetto di Von Neumann in cui i vari nodi sono collegati da un unico bus. Implicitamente la comunicazione avviene in modo Broadcast. Non essendo gestiti secondo una gerarchia ognuno può comunicare (sono tutti peer), è necessario quindi un arbitraggio. Questo arbitraggio può avvenire con un token (poco conveniente e poco sicuro)
* Vantaggi:
  + Semplicità
* Svantaggi:
  + Problemi di prestazione dovuti al fatto che esiste un unico mezzo comunicativo
  + Interruzioni improvvisa del bus principale mette fuori uso tutta la rete
  + È difficile identificare punti di malfunzionamento in quanto non esistono punti notevoli

Rete **ad anello:**

* Caratteristiche:
  + La trasmissione avviene in un senso unidirezionale, la comunicazione circumnaviga l’anello. Si basa molto su un limite fisico reale in cui di sicuro la comunicazione arriverà al nodo destinatario, questo limite è certo nel peggiore dei casi. Utile per prevedere e adattarsi alla comunicazione
* Svantaggio:
  + È necessario un arbitraggio per la comunicazione (token)
  + Se un nodo si rompe la rete smette di funzionare

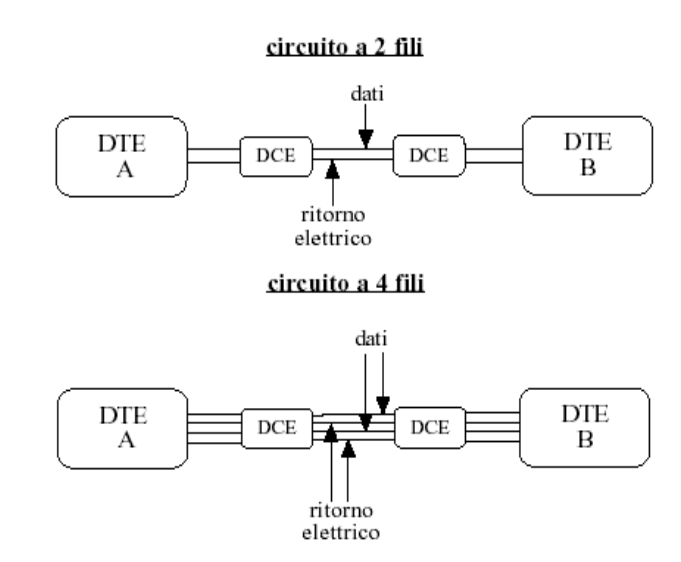
Rete **Mesh:**

* Caratteristiche:
  + Rete composta da tanti nodi interconnessi tra loro (anche attraverso diversi circuiti o mezzi: come radiofrequenze o anche cablate). Un esempio di rete mesh molto usata è il wireless
* Vantaggi:
  + Assicura delle buone prestazioni, grazie all’implementazione dl multitap (ovvero esistono percorsi multipli per arrivare da un nodo ad un altro) che permette anche alla rete di “sopravvivere” alla corruzione o alla rottura di un nodo
* Svantaggi:
  + Costo
  + Scalabilità

La comunicazione avviene secondo tre possibili modi:

* Simplex
  + La comunicazione avviene in senso monodirezionale, e un nodo trasmittente può solo trasmettere e uno ricevente può solo ricevere (es. televisione/radio), per la sua natura non permette la comunicazione tra nodi, impedendo l’invio di un feedback alla recezione di un pacchetto
* Half-Duplex
  + La comunicazione avviene in entrambi i sensi da un nodo all’altro. Questa comunicazione però è basata su una struttura request-reply. Infatti la comunicazione può avvenire una alla volta
* Full-Duplex
  + La comunicazione avviene in entrambi i sensi e può avvenire anche contemporaneamente.

La realizzazione dell’Half-duplex avviene utilizzando il doppino telefonico (un filo di rame per i dati, e l’altro per il ritorno elettrico). Mentre il full-duplex usa quattro fili (2 rispettivamente, per inviare e ricevere dati e 2 per il loro rispettivo ritorno elettrico)

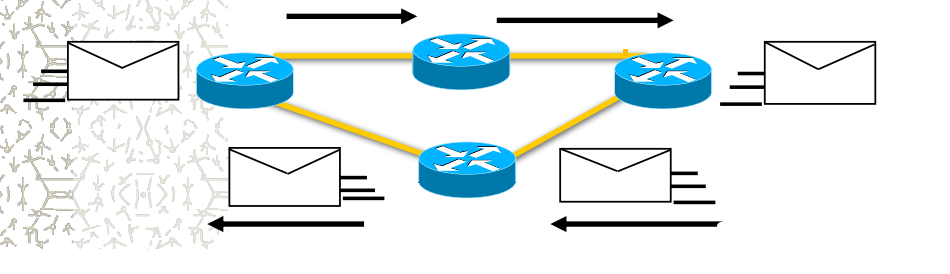


Half-Duplex

Full-Duplex

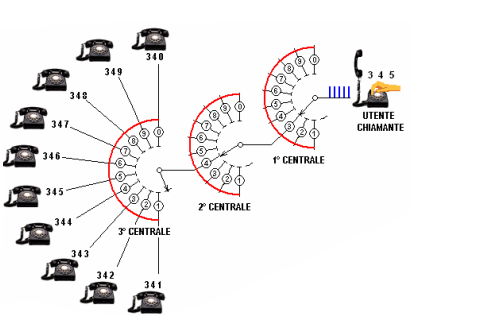
Il “viaggio” delle informazioni avviene tramite la commutazione, nel caso del doppino telefonico si parla di **commutazione di circuito**, mentre **attualmente si parla di commutazione di pacchetto**

**1 - Commutazione di pacchetto**

Si basa sulla suddivisione dell’informazione in *pacchetti* che vengono numerati ed instradati attraverso i nodi della rete (grafo). I pacchetti è molto probabile non seguano lo stesso percorso dei suoi precedenti; infatti, il ricevitore metterà in buffer tutti i pacchetti finché non potrà ricostruire il messaggio nell’ordine in cui è stato inviato. Il **vantaggio** è di una **maggior efficienza, maggiore carico e** soprattutto **è adattabile ad una rete senza che venga prefigurato il percorso dei pacchetti.** Lo **svantaggio** è quello che nell’instradamento dei pacchetti può avvenire che **dei pacchetti vengano persi** o anche **arrivino in disordine**, inoltre necessità di uno standard.

**2 – Commutazione di circuito**

A differenza di quello di pacchetto, il percorso dell’informazione avviene in un percorso predefinito. Si basa su una serie di switch (funzionano come dei multiplexer, ovvero scelgono la “strada” che il segnale percorrerà) che crea un circuito di connessione elettrica tra il mittente e il ricevitore. Per esempio una chiamata telefonica. Lo svantaggio è inefficiente nel gestire un carico maggiore, mentre sulla singola comunicazione risulta molto più efficiente della commutazione di pacchetto.



Quindi la commutazione di pacchetto risulta più facilmente scalabile, più efficiente nel sopportare ed instradare carichi più grandi. Mentre la comunicazione di circuito risulta molto più efficiente nel gestire un’unica comunicazione anche in grandi dimensioni, ma risulta molto meno efficiente nel gestire più richieste e più comunicazioni.

**3 - Caratteristiche commutazione di pacchetto**

L’elemento che switcha il percorso del pacchetto è detto DSE (Data Switch Element) e la loro connessione forma il grafo percorso dai pacchetti nella commutazione di pacchetto. Hanno il compito quindi di instradare i pacchetti.

I percorsi scelti possono anche essere ridondanti. La decisione del percorso non è mai la stessa, e viene scelta sempre la “miglior scelta possibile” che non significa sempre la più veloce o la più breve.

È necessario poi evitare dei punti di congestione (più pacchetti vengono inviati contemporaneamente allo stesso DSE).

**Protocolli**

Un protocollo non è altro che uno standard adattato per comprendere la sintassi e la semantica utilizzate dai calcolatori e dai dispositivi in rete per comunicare. Questo è formato da un insieme di norme, convenzioni e tecniche, ogni protocollo è anche caratterizzato poi dal suo livello di astrazione (1 = livello fisico a salire)

Il protocollo è necessario per il DTE per definire un percorso nella rete, generalmente questo compito viene eseguito dal routing protocol; il compito di questo protocollo è di verificare se una strada da prendere è possibile o un nodo del grafo è non funzionante. Si controlla inviando un messaggio prima della comunicazione per poter definire un percorso sicuro e funzionante.

Standard **de facto**:

* I provider impongono il loro standard agli istituti di standardizzazione

Standard **de Jure**:

* Gli istituti di standardizzazione impongono uno standard ai provider

Alcuni dei principali istituti di standardizzazione sono:

IEEE, ISO, ANSI

Tipi di reti:

* Calcolatori paralleli
  + Massive Parallel (distanza 0,1 m)
  + Multi Processor (distanza 1 m)
  + Cluster (distanza 10 m)
* Reti di calcolatori
  + LAN
  + WLAN
  + MAN
  + WAN
  + GAN (Internet)

**Reti di calcolatori:**

* LAN:
  + Velocità tra 10MB/s e 10Gb/s
  + Standard IEEE 802.3 (evoluzione di Ethernet)
  + Interconnessione tra PC e Workstation
* WAN
  + Struttura formata da un insieme di LAN interconnesse da vai gateway
  + Struttura a maglia
* MAN
  + Estensione di aree metropolitane
  + Velocità tra i 2MB/s e 10Gb/s
  + Precedentemente basate sulle WAN scalate ad aree metropolitane, ora sostituita da Ethernet su scala metropolitana
* Internetwork
  + Rete formata dall’interconnessione di reti WAN, MAN e LAN. Per permettere la connessione di reti progettualmente diverse la connessione avviene attraverso un gateway, che, oltre a instradare i pacchetti verso altri gateway, effettua operazioni per permettere la comunicazione tra gateway sul protocollo TCP/IP

Queste tre connessioni possono comunicare attraverso due metodi:

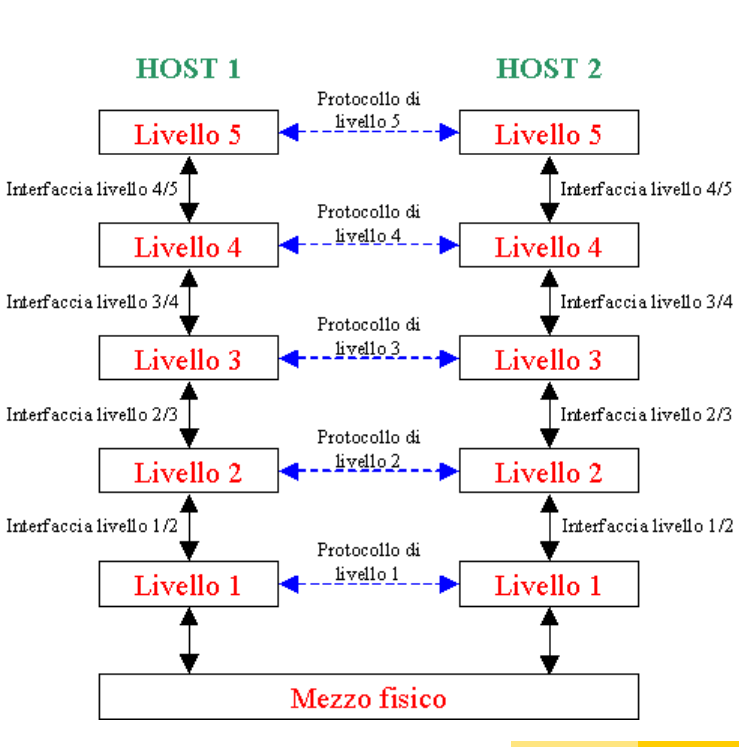
* Connection oriented mode (tipica della comunicazione di circuito)
* Connectionless mode (tipica della comunicazione di pacchetto)
  + Si inviano dati e pacchetti anche nel caso il destinatario non è in linea sperando che vengano ricevuti e letti

Nel caso della Connectionless la **congestione** avviene quando il buffer di un nodo nella rete è saturo di messaggi (pacchetti)

Generalmente è importante ricordare che per subnet si intende una ***sottorete*** con un indirizzo univoco

**ISO-OSI**

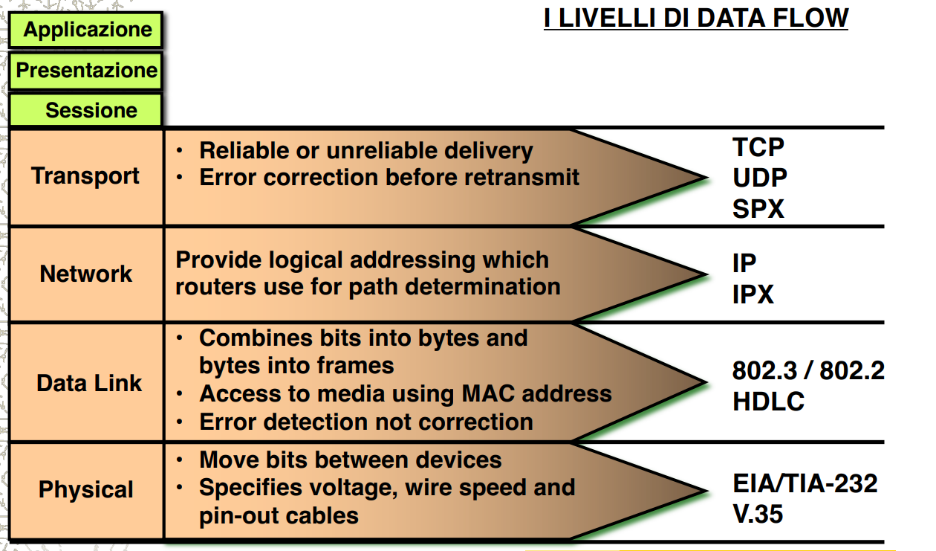
È un protocollo basato su **livello** che comunicano tra di loro, ogni livello ha un grado di astrazione sempre più alto(salendo).

Il livello più basso è il livello fisico.

Ogni livello per comunicare non è effettivamente connesso con il corrispettivo livello, ma è “logicamente” connesso con il suo pari (in quanto ad astrazione). Mentre è fisicamente connesso con il suo livello soprastante e il suo sottostante

Un dato (PDU: Protocol Data Unit) viene incapsulato e deincapsulato

* Incapsulato
  + Al payload viene aggiunto un header ad ogni livello che scende
* Deincapsulamento
  + Dal payload viene rimosso un header e viene convertito il dato nel livello di astrazione corrispondente specificato nel payload

**Livelli del modello ISO-OSI**

TCP/UDP ->

Ethernet ->

Livelli:

1. Fisico:
   * Rappresentazione in bit come impulso elettrico
2. Data Link
   * Eleva la comunicazione “grezza” elettrica del livello fisico segnalando eventuali errori
   * Suddivide in pacchetti
3. Network
   * Come comunicare in una rete: instradamento, e gestione della congestione e del flusso
4. Transport
   * Fornisce il servizio di invio dei messaggi senza garanzia di invio
5. Sessione
   * Specifica dati differenti in base all’applicazione, gestendo i token per la comunicazione. Impostando anche la comunicazione e la sincronizzazione della comunicazione
6. Presentazione
   * Specifica il modo in cui vengono rappresentati i dati, per esempio la crittografia o la compressione dei dati
7. Applicazione
   * Interfaccia utente, es. file transfer, posta elettronica, DNS, ecc…

Tra loro i dispositivi ISO-OSI hanno dispositivi come switch e router, che implementano solamente i livelli necessari per il funzionamento. Per esempio, un router arriva fino al livello Network mentre uno switch arriva fino al livello Data Link

**1 - Livello Fisico**

Il livello fisico gestisce i segnali della propria macchina nella rete. Ha il compito quindi di permettere la comunicazione in rete utilizzando segnali (acustico, elettrico, luminoso, elettromagnetico ecc…)

Un segnale non è altro che la variazione di una grandezza fisica che trasporta informazioni (onda)

Nel caso di segnali elettrici vi sono di due tipi:

* Analogici
  + Nel variare del tempo i segnali analogici possono assumere tutti i valori tra il picco e la gola che rappresentano massimo e minimo consentiti dal canale di comunicazione
* Digitale
  + Si dice digitale un segnale che può strettamente assumere solo 2 valori, massimo o minimo, nel caso di un computer può assumere esclusivamente 0 o 1

Il segnale dev’essere codificato e può essere quindi trasformato utilizzando la **trasformata di fourier** in forma di somma di sinusoidi per studiarne frequenza ed ampiezza attraverso lo spettro dell’onda così da identificare eventuali distorsioni di fase o di ampiezza (disturbi fisici).

La trasformata di fourier di un’onda produce la sinusoide che descrive l’andamento dell’onda (detta **armonica** dell’onda). Quindi in un segnale formato da due frequenze la trasformata di fourier produrrà due spettri delle due diverse frequenze.

Conoscere la frequenza di un segnale è necessario per impostare e riconoscere i filtri (passa alto, passa basso, passa banda, annulla banda), che escludono o permettono solo determinate frequenze. I filtri vengono dettati dalla **larghezza di banda**.

Ogni canale di comunicazione possiede poi una propria **larghezza di banda**, oltre la quale (o prima) il canale non è capace di permettere la comunicazione. Infatti, il canale permette alle comunicazioni di passare solo se soltanto se si trovano nel loro **range di frequenza.**

La comunicazione avviene attraverso l’invio di **campioni** dell’onda formata dal segnale. Poi il ricevente interpola i vari campioni così da ricomporre l’onda originale (o almeno approssimata)

La frequenza di campionamento è definita attraverso il **Teorema del campionamento (teorema di Nyquist)**: Dato un segnale x(t) [*il segnale*] a banda limitata B, si può ricostruire completamente il segnale a partire da un campionamento del segnale se la frequenza di campionamento è **F >= 2B** (**Frequenza di Nyquist**).

**CRITERIO DI NYQUIST** Per via di questa relazione, sia Q la quantità di informazione associata e V un numero di livelli trasmissivi equiprobabili, **Q=log2V**. Allora, la **massima quantità di informazione trasferibile** è uguale a **2BQ**.

Le conseguenze di questo criterio sono che la massima quantità di informazione trasferibile attraverso un canale allora è direttamente proporzionale al logaritmo del numero dei livelli trasmissivi.

**CONTRIBUTO DI SHANNON** Nyquist nel suo criterio non ha considerato che in qualsiasi mezzo comunicativo è presente un rumore di fondo (che sia di causa endogena o esogena). Il problema di aumentare e creare nuove suddivisioni dei livelli della banda è che il limite per non perdere dati è dettato dall’ampiezza del rumore ma soprattutto dalla differenza tra un canale e l’altro che all’aumentare dei canali diminuisce la differenza tra un canale e l’altro, rendendo sempre più difficile riconoscere un canale dall’altro.

**Il rumore**

Il rumore è una quantità di energia non voluta presente nel segnale

Esistono vari tipi di rumore:

* Bianco
  + Ha un contributo su tutte le frequenze, facilmente risolvibile in quanto è costante ovunque
* Modo comune
* Quantizzazione
* Intermodulazione

Per il lavoro di shannon possiamo affermare che la capacità massima del segnale è misurata in bit/s, e la sua formula è:

**C = B log2(1+ )**

Dove S rappresenta la potenza del segnale ed N

Quindi per la **legge di shannon**, sì, aumentare i livelli aumenta il tasso di trasferimento però, questo aumenta anche la frequenza di eventi non previsti o voluti per via del rumore

La **trasmissione del segnale** è **analogica** se si trasmette senza curarsi del significato; quindi, la trasmissione ha l’unico compito di recapitare il segnale ed eventualmente aumentarne di intensità

Si dice **digitale** la **trasmissione** che tiene conto del contenuto dei dati per decidere se e quando amplificare il segnale, che si basa su una rigenerazione del segnale per via di un’estrazione del contenuto informativo per poter inviare il segnale rigenerato al destinatario.

Trasmissione in banda base: immesso il segnale direttamente sul canale che può causare un’attenuazione di segnale

In alcuni casi è opportuno trasmettere il segnale in un canale che utilizzi una banda a frequenza che non causi un’attenuazione di segnale

Ogni dato binario prodotto dall’onda viene codificato in bit attraverso sistemi di codifica, in cui ad ogni bit corrisponde un particolare livello di codifica. Il ricevente del segnale deve far corrispondere ad un qualsiasi livello del segnale uno specifico bit, la durata del bit e il valore di quel bit. La comunicazione tra più nodi nella rete deve quindi essere vincolata da uno standard (che a livelli superiori sarebbe un protocollo) dettato da un sistema di codifica che si basa sulla caratteristica di trovare lo spettro sempre al centro della banda

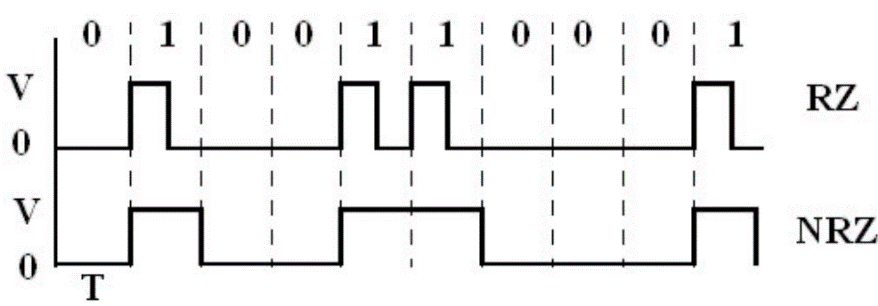
**Codifica unipolare RZ (Return to Zero)**

Ogni bit richiede per la trasmissione un tempo massimo T, in cui il bit con valore 0 ha ampiezza 0, mentre per i bit con valore 1, l’impulso dura T/2 e ha ampiezza (voltaggio) maggiore:



**Codifica unipolare NRZ (Non Return to Zero)**

A differenza della RZ la tensione nel caso di un bit pari ad 1 viene mantenuta alta per tutta la durata di T



Differenza tra RZ e NRZ

Pregi della NRZ:

* Facile da progettare
* Utilizzo efficiente della larghezza di banda (la potenza è inclusa tra 0 e la metà della capacità trasmissiva R, R/2)

Difetti:

* Componente continua (tensione alta continua)
  + La trasmissione continua su una tensione può causare uno sfasamento del ricevitore

La NRZ può essere migliorata in due modi:

* NRZ-L
  + A differenza dell’NRZ, ci sono due tensioni possibili una V+ che indica il bit 0 e una V- che indica il bit 1
  + Questa implementazione riduce la componente continua anche se non la annulla del tutto
* NRZI
  + Alternativa alla NRZ-L ma a i bit 1 hanno V+ e i bit 0 hanno V-

**Codifica Bipolar-AMI**

Nel caso della codifica Bipolar-AMI (Alternate Mark Inversion) esistono 3 possibili tensioni:

* Bit 0
  + Tensione (V) = 0
* Bit 1
  + V+ o V-

Una variazione del Bipolar AMI è la Pseudoternary che inverte i bit 1 e 0, quindi:

* Bit 0
  + V+ o V-
* Bit 1
  + V = 0

I vantaggi della Bipolar-AMI sono:

* Risolve il problema della NRZ della sequenza di bit 1, eliminando la frequenza continua per i bit 1 (ma resta quella per i bit 0) azzerando la componente continua
* A parità di transfer rate, utilizza minor banda
* Gli errori isolati possono essere individuati più semplicemente

Svantaggi della Bipolar-AMI:

* Utilizza 3 livelli, ogni simbolo potrebbe portare più informazione ma è sprecata (Log2(3) = 1,58)
* A parità di bit rate richiede circa 3dB in più della NRZ

Viene utilizzata quindi per le linee punto-punto (ISDN)

**Codifica Manchester**

La codifica manchester per ogni bit utilizza mezzo periodo per rappresentare il bit (T/2) e:

* Bit 0
  + Si ha V+ per la prima metà del periodo e V- per la seconda
* Bit 1
  + Si ha V+ per la seconda metà del periodo e V- per la prima

Differential Manchester è precisamente l’opposto della Manchester normale

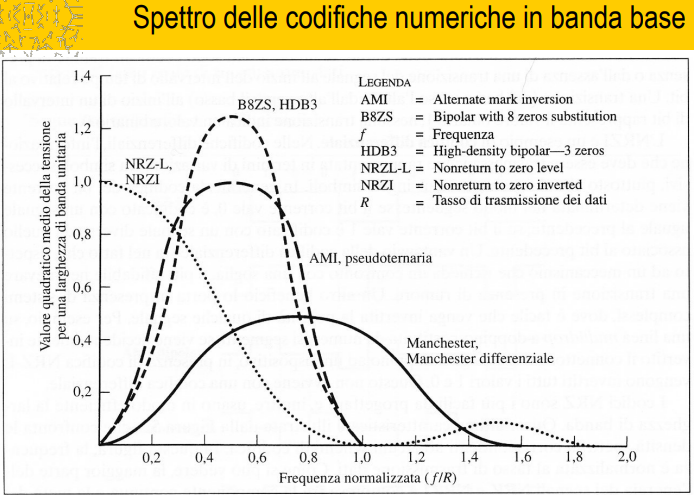
Vantaggi:

* Sincrizzazione
  + Ogni bit possiede una transizione in mezzo così da poter permettere la sincronizzazione del ricevitore
* Mancanza della componente continua
* Facile rilevazione dell’errore

Svantaggi:

* Richiede frequenza doppia per la rappresentazione di un singolo bit: 1 bit richiede 2 baud, quindi una banda doppia

È utilizzata soprattuto nello standard 802.3 (ethernet), 802.5 (token ring), sul cavo coassiale e sul doppino



**Modulazione**

La modulazione è un processo che produce un segnale *modulato* all’interno del passa banda di un mezzo di comunicazione

Vantaggi:

* Per la trasmissione spesso sono preferiti diversi intervalli di frequenza
* Permette la trasmissione di più comunicazioni differenti sullo stesso mezzo, trasferendo in **zone** differenti **comunicazioni** differenti

**Tecniche di modulazione**

* **Tecnica ASK (amplitude shift keying)**
  + Partendo da un segnale numerico (tipo NRZ) è possibile modificare l’ampiezza del segnale moltiplicando il segnale originale per un segnale modulatore che produce un’ampiezza sinusoidale
* **Tecnica FSK (Frequency Shift Keying)**
  + Dato un segnale numerico, questo viene modulato in frequenza moltiplicandolo con un segnale modulatore con 2 frequenze per i due stati del segnale numerico
* **Tecnica PSK (Phase Shift Keying)**
  + Dato un segnale numerico, ad un determinato valore del segnale viene assegnato un determinato valore di fase, per esempio il cambio di valore in bit 1 corrisponde ad un cambio di fase
* **Tecnica BPSK (BiPhase Shift Keying)**
  + Ai valori 0 ed 1 vengono associate due fasi differenti sulla stessa frequenza, precisamente 0° e 180°
  + Questo produce una maggior efficienza in un canale permettendo ad ogni simbolo trasporti più bit
* **Termine QPSK (Quadrature Phase Shift Keying)**
  + Vengono utilizzati 4 angoli di fase, permettendo di trasmettere due bit:
    - 00 a 45°
    - 01 a 135°
    - 10 a 225°
    - 11 a 315°
  + Il numero di bit non può essere aumentato all’infinito per la presenza sempre maggiore di distorsione e rotazione del segnale (***phase jitter***)

**Tecniche di Multiplazione**

Ovvero consentire una comunicazione multipla su un solo canale trasmissivo

* TDM (a divisione di tempo)
  + Il canale viene diviso in quanti di tempo e in ognuno avviene una comunicazione (anche diverse)
* SDM (a divisione di spazio)
  + I dati vengono inviati attraverso mezzi diversi fisicamente
* FDM e WDM (a divisione per frequenza)
  + Il mezzo viene suddiviso per bande di frequenza, ognuna permette di differenziare i dati trasmessi
* CDM (a divisione per codifica)
  + La differenziazione avviene in base alla codifica di ogni dato trasmesso

**TDM**

Meno efficiente ma ottimizza l’uso dei mezzi trasmissivi, in quanto il mezzo comunicativo viene suddiviso **frame** (trame insieme di slots) che a loro volta sono composti di **slot** (quanti di tempo)

**Trasmissione wireless**

Trasmissione di informazioni via aerea attraverso l’uso di onde elettromagnetiche.

La trasmissione si basa sulla presenza di un trasmittente e di un ricevente (ognuno con la propria antenna). L’antenna può captare ed emanare una determinata ***banda***determinata dalla frequenza usata dall’onda elettromagnetica.

**Un’antenna** è composta da:

* Feader
  + Rappresenta l’antenna vera e propria che ha il compito di trasdurre
* Circolatore
  + Elemento che ha il compito di switchare da onda segnale da inviare e onda segnale da ricevere
* Trasmettitore
  + Elemento che produce un segnale onda che attraverso il circolatore arriverà al feader
* Ricevitore
  + Elemento che riceve un segnale onda per via del circolatore (dopo che è stato captato dal feader)

Immagine che contiene diagramma

Descrizione generata automaticamenteIn un’onda elettromagnetica la sua frequenza implica la sua natura e la sua quantità di energia trasmessa e di energia trasportabile (banda):

Nell’ambito della telecomunicazione, le frequenze per la comunicazione vengono assegnati da enti nazionali.

E su queste bande (in italia) è preposto il controllo da parte della polizia postale

**Tipologie di trasmissione radio**:

* DSSS (Direct Sequence Spectrum)
  + Ogni bit viene trasmesso come una sequenza ridondante di valori
  + Preferito per la trasmissione e ricezione di segnali deboli, consente l’interoperabilità delle reti wireless attuali
* Spread Spectrum
  + Il trasmettitore esegue dei salti di frequenza per consentire la comunicazione continua senza interruzioni causate da frequenze che creano interferenza
  + Tecnologia militare per evitare segnali di jamming
* Trasmissione via ponte radio
  + Si hanno delle torri trasmittenti che formano una comunicazione punto-punto, anche se è necessaria visibilità tra un’antenna e l’altra
  + Si utilizzano antenne più piccole per sfruttare una frequenza maggiore (ad una frequenza maggiore coincidono gittate minori) mentre per distanze maggiori vengono utilizzate frequenze più basse

La comunicazione satellitare si basa appunto su una flotta di satelliti, distinti per altezza dal suolo:

* LEO
* MEO
* GEO

Ogni satellite utilizza una frequenza diversa, quella utilizzata da un satellite più esterno utilizza frequenze più basse, ma richiedono meno satelliti per coprire tutta la terra, mentre più sono vicini e più necessitano altri satelliti ma hanno una frequenza maggiore

**2 – Livello Data-Link**

Il livello data-link ha generalmente il compito di **verificare la correttezza** della trasmissione dei dati attraverso il livello fisico, **suddivide i dati in pacchetti** e soprattutto ha il compito di **risolvere l’addressing**, ovvero di specificare il destinatario del pacchetto, soprattutto nel caso non ci sia una comunicazione diretta col destinatario.

Il livello data-link:

* Riceve i pacchetti dal livello superiore, ha il compito di risolverle in frame (**framing**)
* Garantisce la mutua esclusione dal mezzo di comunicazione (**accesso multiplo/multiplazione**)
  + Fa rispettare la regola di half-duplex o di full-duplex
* Controlla e gestisce gli errori di trasmissione (**controllo di errore**)
  + Rilevazione e correzione o rilevazione e scarto
* Regola il flusso della trasmissione fra sorgente e destinatario (**controllo di flusso**)
  + Evita che il nodo trasmittente saturi il nodo ricevente

Nel livello data-link non è necessario conoscere le caratteristiche del livello fisico (tipo riconoscere se si tratta di fibra ottica o doppino telefonico) bensì utilizza delle API per la trasmissione

Il livello data-link fornisce ai livelli superiori:

* Trasmissione senza riscontro e senza connessione
* Trasmissione affidabile senza connessione
* Trasmissione affidabile con connessione

Il frame (informazione) viene individuato da un numero univoco. Questo per permetterne l’identificazione all’invio e alla recezione.

**Controllo di parità**

Nel frame esiste un bit detto *di parità* che indica il numero di 1 nel payload e rileva esclusivamente un numero dispari di errori. Una possibile soluzione a questo problema (rilevazione di soli numeri dispari di errori) è dividere in k gruppi detti ***parole sorgente*** di n bit ciascuno. Queste parole sorgente vengono trasformate in parole codice di dimensione k+r, dove r è una quantità di spazio intera qualsiasi tale da poter individuare un errore.

**Parità bidimensionale**

Le parole di k bit vengono organizzate una sotto l’altra (formando una matrice con i loro bit), aggiungendo ad ognuna il bit di parità (rispetto alle righe e alle colonne). Ottenendo quindi 2\*k+1 bit di parità totali:

es:

10101|1

11110|0

--------+---

01011|1

In internet il sistema di checksum rileva gli errori ma viene usata esclusivamente a livello di trasporto:

* Mittente
  + I dati vengono trattati come interi da 16 bit e sommati
  + Checksum
    - Complemento ad 1 della somma
  + Il complemento ad 1 della somma viene messo come intestatario

**Campi di Galois**

È un campo finito con q elementi in cui sono definite due operazioni aritmetiche (addizione e moltiplicazione), ovvero, che godono di proprietà commutativa e associativa. Si indica con GF(q)

Generalmente q dev’essere sempre primo o potenza di numeri primi

L’addizione e la moltiplicazione vengono calcolate utilizzando metodi aritmetici classici (somma e moltiplicazione) ma con l’applicazione del mod(q) sul risultato

**CRC (Cyclic Redundance Check-Sum)**

CRC =

Una sequenza di N bit si può rappresentare attraverso un polinomio a coefficienti binari con grado N-1:

1011 si rappresenta come -> x3 + x1 + 1

**CRC Polinomiale**

La tecnica consiste nel considerare i dati di m bit come un polinomio di grado m-1.

Trasmettitore e ricevitore si accordano sull’uso di un unico polinomio generatore G(x) di grado r. Questo polinomio generatore serve per poter identificare un nuovo messaggio ricevuto e inviato.

Difatti ad ogni messaggio viene aggiungo, in coda, una frequenza di controllo, che rende il polinomio associato al messaggio (payload dei dati e CRC) sia divisibile per G(x)

In ricezione quindi il polinomio generatore G(x) si usa come dividendo per i dati ricevuti.

**Controllo di flusso**

Il controllo di flusso è necessario in quanto il mittente può avere una capacità di trasmissione maggiore rispetto a quella di ricezione del ricevente. Questo può saturare il buffer di arrivo, causando la perdita delle nuove informazioni inviate dal mittente. È un problema anche di affidabilità in quanto si incombe in una perdita di informazioni.

Questa problematica è causata dall’implementazione stessa del data-link layer. In quanto questo presenta due procedure usate per comunicare con il livello superiore:

* from-network-layer()
* to-network-layer()

Inoltre, presenta altre due procedure analoghe per la comunicazione col livello fisico.

Logicamente il data-link layer ha una procedura di wait for event nel quale si blocca finché non è presente un evento da processare. All’arrivo di un segnale (ricezione), questo preleva i dati dal livello fisico, li processa e li invia allo strato di rete (network layer).

Durante la processazione dei dati in arrivo (tra le chiamate from-physical-layer() e to-network-layer), il data-link mette in un buffer tutti i dati in arrivo dal livello fisico.

Una soluzione alla saturazione di questo buffer, viene inserito un ***collo di bottiglia*** formato da un meccanismo di valutazione dei tempi di risposta del ricevente e da un conseguente ritardo di attesa. Il problema però sorge per via dei tempi di ricezione variabili e non costanti.

Generalmente possiamo scegliere tra due alternative:

* Sovrastimare basandosi sul caso peggiore
  + Più affidabilità ma meno efficienza
* Sottostimare basandosi sul caso medio
  + Meno affidabilità ma più efficienza

L’ottimale avviene in seguito ad un’attenta e complessa analisi del ritardo del ricevente.

Il **frame data-link** possiede un **header**, una coda (**trailer**) aggiunti al payload passato dal livello di rete.

**Header:**

* Tipo pacchetto (**type**) (data, ack, nack, …)
* Numero sequenza del pacchetto (**seq**)
* Numero di riscontro (**ack**)
  + Utile per ***riscontrare***l’arrivo di un determinato frame
    - Nel caso sia un NAK (negative ack) il messaggio dev’essere rinviato poiché il ricevente ha ricevuto un messaggio corrotto oppure errato

**Affidabilità dei dati**

Queste tecniche servono per ridurre l’inaffidabilità della trasmissione dei dati. Difatti, pur prendendo i dati da un livello inaffidabile, vengono applicate queste tecniche affinché i livelli superiori (in ricezione) rilevino l’informazione come proveniente da una fonte affidabile. Quindi, più in alto ci si trova nell’architettura ISO-OSI e più sarà affidabile l’informazione. A differenza dei livelli bassi in cui il canale inaffidabile risiede nella sua natura e non può essere eliminato del tutto.

Rtd\_send() -> invio dei dati affidabile (dal livello superiore al livello data-link)

UDT\_send() -> invio dei dati inaffidabile attraverso un canale inaffidabile (data-link a livello fisico)

**ACK, NAK, STOP-AND-WAIT (RDT2.0)**

L’ack e il nak potrebbero essere a loro volta corrotti nell’invio dal ricevente al mittente. Perciò è necessario aggiungere cheksum (in un pacchetto), poiché ci si basa (originariamente) su un’architettura stop-and-wait, e poiché il segnale di avvio viene perso ci si ritrova in un deadlock (entrambi ricevente e mittente bloccati in attesa di un segnale che è stato perso/corrotto).

**RDT3.0**

Quest’architettura può essere implementata aggiungendo un time-to-wait, in cui si aspetta un tempo t (generalmente il doppio del roundtrip time), oltre il quale viene rinviato il messaggio in mancanza di un ACK come risposta (mittente). Il problema di questo sistema è che può causare la presenza di duplicati

Il tempo di trasmissione in RDT3.0 è uguale a:

T= L/R (L= lunghezza pacchetto in bit, R=tasso trasmissivo)

Mentre U indica la frazione di tempo in cui il mittente è impegnato nella trasmissione:

U=(L/R)/(RTT + L/R)

Pur utilizzando un Sistema RDT3.0 per permettere una comunicazione più efficiente, si fa pipelining nella comunicazione, inviando gruppi di pacchetti così da ricevere poi gruppi di ACK relativi ai pacchetti. Gli algoritmi che implementano il pipelining sono basati su due forme generiche:

* Go-Back-N
* Ripetizione selettiva

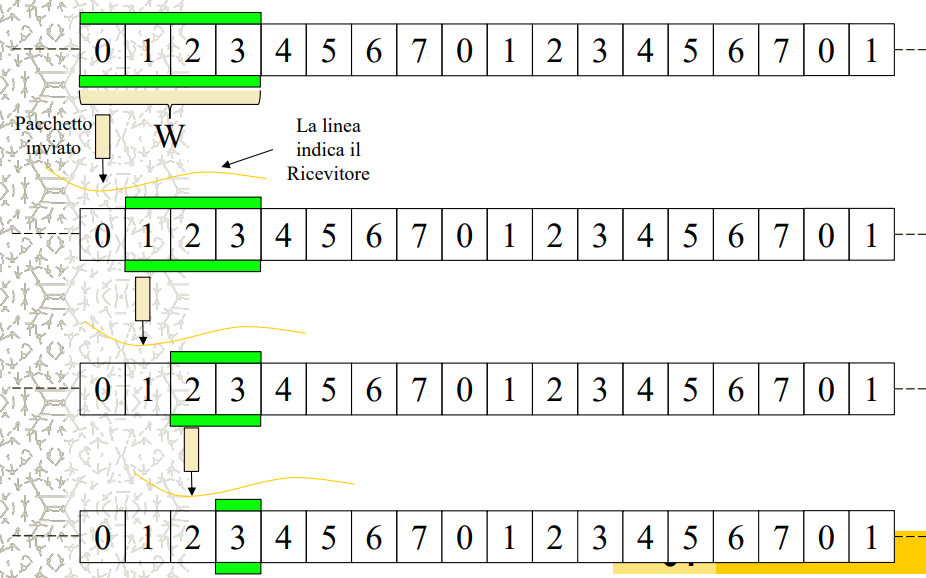
**Go-Back-N**

Immagine che contiene testo

Descrizione generata automaticamenteÈ detto “a finestra scorrevole”, ovvero, i frame vengono numerati (ogni frame ha il suo frame di sequenza). Questi numeri di sequenza vengono incorporati nel frame stesso. Questo frame viene inviato “scorrendo” la sequenza inviando n frame (dipendentemente dalla dimensione della finestra), e ad ogni ack in arrivo al mittente questa finestra scorre in avanti (tanti quanti sono gli ack ricevuti) così da inviare i bit successivi:

La dimensione n della finestra dev’essere:

* N abbastanza piccolo da essere efficiente
* N abbastanza grande per non creare ambiguità nel frame

La finestra (in invio) ha dimensione variabile che ha come massima dimensione n: 

(in questo caso è il limite superiore a muoversi)

Immagine che contiene diagramma

Descrizione generata automaticamenteLe dimensioni delle finestre si allargano e restringono in base ai pacchetti inviati e i loro corrrispettivi ack:

**Bridging vs Switching**

Bridging:

1. Software-based
2. Un solo spaninng tree per bridge
3. Numero inferiore di porte

Switching:

1. Ha componenti Hardware-based
2. Multipli spanning tree associati ad ogni bridge
3. Numero molto più grande di porte

A livello Data-Link vengono utilizzati insieme per un uso più efficiente e scalabile di una rete in quanto si possono mettere (tramite un bridge/switch) in comunicazione più sottoreti che utilizzano protocolli diversi. In queste reti composte da switch e bridge è possibile effettuare il **backward learning**:

* Ovvero, ogni nodo (macchina) collegata alla rete viene memorizzata e individuata in seguito al suo primo invio di una trama sulla rete, in quanto al boot una rete non conosce tutti i nodi connessi; infatti, le trame che vengono passate vengono trasmesse in multicast/broadcast se non è noto ancora il nodo destinatario (eccetto che per il nodo mittente che diventa noto e viene escluso per evitare effetto **loopback**. Ogni nodo quindi diventa noto in base al suo MAC e alla porta usata, permettendo così allo switch di evitare collisioni sulla stessa porta.

**Switching**

Esistono due tipi principali di switching:

* Cut-through switching
  + Il frame viene subito inviato sulla porta corretta
* Store-and-forward
  + Permette di filtrare il traffico, attraverso un controllo del CRC del frame prima dell’invio, questo controllo se rileva un errore scarta il suddetto frame
* Port-Based Switching
  + Ad ogni porta è associato un unico indirizzo Ethernet
* Segmented-Based switching
  + Ad ogni porta vengono associati più indirizzi Ethernet (es. collegata ad un Hub)

**Topologie con Ridondanza**

Le topologia con ridondanza eliminano i single points of failure, ma in compenso può causare una ripetitività delle trame e dei dati durante le comunicazioni broadcast, per esempio:

un server host manda un frame in broadcast al primo switch, che lo invia al secondo switch ridondante, che a sua volta lo rimanderà al primo e così via. Questo è detto **broadcast storm**.

Il broadcast storm avviene per via della statelessness dello switch (non si ricorda di aver mandato già lo stesso pacchetto)

Inoltre sia in questa rete un router Y, questo in un invio broadcast da parte dell’host riceverà due volte lo stesso frame (la prima volta come unicast dall’host, e la seconda attraverso i due switch ( host -> sw1 -> sw2 -> Y)

**La soluzione** è rappresentata dallo **Spanning-Tree Protocol**

* Ha come caratteristica quella di disabilitare determinate porte dei bridge
* L’algoritmo funziona così:

1. Seleziona un root bridge
2. Determina il percorso meno costoso da ciascun bridge/rete a root bridge
3. Disabilita i link inutilizzati

Ogni bridge nella rete poi possiede esclusivamente un unico root port (ovvero il collegamento verso la radice), ed una designated port per segmento

Per lo scambio di dati tra bridge viene utilizzato il BPDU (Bridge Protocol Data Unit), di default l’invio avviene ogni 2 secondi

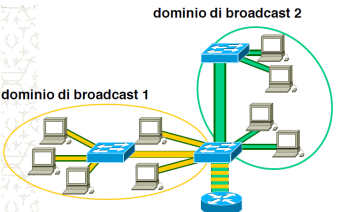
Inoltre, ogni Bridge è identificato da un BridgeID, in cui quello con ID più piccolo è detto Bridge Root

**Virual LANs**

VLAN = dominio di broadcast = Rete Logica (subnet)

Data una lan, si vuole definire un dominio logico che consente flessibilità e scalabilità di questa.

Una VLAN è una rete LAN virtuale, esponendo e rendendo raggiungibile solo un segmento di rete, mentre la rete all’interno è frammentata in domini di broadcast accessibili attraverso lo switch (parte esposta) a cui poi è collegato il router



La rete VLAN in immagine è composta da entrambi i domini di broadcast, dal bridge centrale (Rettangolo) e dal router (cerchio)

Ad ogni VLAN logica corrisponde un Bridge fisico

Inoltre ogni VLAN può attraversare diversi Switch, collegate da un **Trunk**, ovvero un mezzo di comunicazione veloce che permette, appunto, di collegare due VLAN uguali in switch diversi

Le VLAN possono dividersi in Statiche e Dinamiche:

* Statiche
  + Si può identificare la VLAN in base alla linea di arrivo, in quanto ogni linea appartiene esclusivamente ad una ed una sola VLAN
  + Se due switch sono connessi tra loro per il trasferimento di dati di due VLAN, questi presentano anche due linee di connessione, ciascuna appartenente ad una VLAN
  + **PRO**
    - Le VLAN Statiche presentano una prestazione migliore sulla singola connessione, in quanto esiste un collegamento diretto tra mittente e destinatario
  + **CONTRO**
    - Il cablaggio diventa complicato quando bisogna scambiare informazioni tra due VLAN attraverso un router in quanto si necessitano due trasmissioni su porte diverse
* Dinamiche
  + La VLAN viene identificata in base al MAC Address di provenienza e non di invio in quanto non si potrebbe sapere a quale VLAN assegnare i frame Broadcast
  + In questo caso possono esserci lo scambio di informazioni tra due VLAN contemporaneamente
  + Utilizzano la **trunk** per una comunicazione promiscua, evitando di utilizzare tante porte quante sono le VLAN, ma si basa invece sulla multiplazione attraverso l’ASIC

Durante il **trunking** il destinatario viene aggiunto nel payload del frame, che poi viene rimosso alla recezione di tale frame (dallo switch)

**VLAN mittente** –manda il frame🡪 **Switch mittente** –incapsula il **frame** con **CRC** (in coda)e indirizzo **MAC**(testa) 🡪 **Switch destinatario** –**deincapsula**, rimuove **CRC** e **MAC**, reincapsula solo il **frame** 🡪 **VLAN destinatario**

**Switch Livello 3**

Lo switch di livello 3 è uno switch Ethernet che commuta pacchetti in base al loro indirizzo MAC ma anche al loro indirizzo IP (indirizzo di rete)

Le reti possono essere categorizzate per estensione (LAN,MAN,GAN)

Per WLAN si intende Wireless LAN, che rappresenta una rete lan senza necessità di un cablaggio effettivo, questo produce vari vantaggi:

* **Mobilità**
  + Gli utenti possono connettersi alla rete senza la necessità di doversi collegare fisicamente ad una presa
* **Velocità e semplicità**
  + Rende più facile l’accesso, l’espansione e la scalabilità
* **Facilità di installazione**
* **Costi minori**
  + La manutenzione è molto più semplice in quanto non ci si basa su molti dispositivi fisici fallaci
* **Facilmente scalabile**
  + La scalabilità deriva dalla possibilità di aumentare la portata grazie all’aggiunta di access point

Nella WLAN la trasmissione fisica avviene omnidirezionalmente, ovvero, l’antenna del router/access point emettono onde verso tutte le direzioni, in una propagazione ellittica (in cui l’antenna è uno dei due fuochi), inoltre esistono varie bande di frequenza, senza la necessità di registrazione e allocazione dette **ISM**, per l’utilizzo di queste bande sono obbligatorie caratteristiche precise:

* Potenza massima di trasmissione
* Utilizzo dello spread spectrum come tecnica trasmissiva

Le bande utilizzate poi per le trasmissioni wireless sono 2,4GHz e 5GHz

Le **Spread Spectrum** sono tecniche di trasmissione (FHSS o DSSS), si basa sullo “spalmare” la trasmissione sulle varie frequenze disponibili per evitare un jamming, queste sono regolate dallo standard IEEE 802.11

Topologie dello standard 802.11:

1. Independent Basic Service Set
   1. È una BSS che presenta un componente chiamato Access Point (AP) che ha il compito di relay nel BSS
   2. La BSS è una comunicazione di tipo cellulare
2. Extended Service Set
   1. Estende la mobilità delle stazioni ad un raggio arbitrario, è composto da un insieme di BSS composta da più Access Point
   2. L’ESS nasconde le caratteristiche della rete al di fuori del modello ISO

Lo spread spectrum è utile per non utilizzare tutta la banda, abbassando di conseguenza anche le interferenze (aumentando quindi la resistenza in ambienti disturbati)

Divisione di spettro SST:

1. FH (frequency hopping) SS (
   1. Si possiedono più canali all’interno della banda (49 canali da 1MHz), tra le quali la trasmissione salta ogni tot di tempo determinato (solitamente meno di 400ms)
   2. **VANTAGGI**
      1. **Più sicura**
   3. **SVANTAGGI**
      1. **Molto limitata in banda**
      2. **Non utilizzabile nel WIFI**
2. DSSS
   1. Si utilizza la tecnica chipping, ovvero, ogni bit della trasmissione viene suddivisa in una serie di bit ridondanti, questo bit viene trasmesso per m intervalli di tempo generando uno **pseudorandom number (PN)**
   2. La banda viene suddivisa in 14 canali da 5MHz
   3. **VANTAGGI**
      1. **Più immune ai rumori**
      2. **Possibilità di arrivare a 11Mbps**
   4. **SVANTAGGI**
      1. **Maggior spreco di banda**

Protocollo di accesso al messo wireless **CSMA/CA**:

* Rileva se un canale è libero per un tempo **DIFS** (Distributed Inter Frame Space), e nel caso trasmette (un intero frame)
* Se il canale è occupato (non riceve ACK), aspetta un tempo casuale (backoff time) e poi riprova
* Il frame risulta ricevuto se non sono avvenute collisioni

In CSMA/CA non rileva le collisioni, in quanto questa richiede l’abilità di spedire e ricevere contemporaneamente (full duplex), ma il CSMA/CA lavora sull’half duplex

Anche fosse full duplex, ci sarebbero comunque 3 tipi di problemi:

1. Stazione nascosta
   1. Siano A, B, C tre stazioni in cui B è raggiungibile da A e da C ma A e C non si possono raggiungere; quindi, per CSMA/CA di A e C rilevano B come libera e quindi provano a trasmettere entrambi causando interferenza in B perché non rilevano l’altra stazione trasmittente
2. Stazione Esposta
3. Fading
   1. I contributi delle stazioni A e C abbastanza deboli da non rilevare B occupato ma abbastanza potenti da interferire con B

Questo problema può essere evitato inviando un **RTS** (request to send) ovvero una richiesta di “permesso” di poter trasmettere per un tempo determinato, a cui poi la stazione ricevente risponderà con un **CTS** (clear to send), con cui la stazione conferma la sua disponibilità

L’algoritmo di backoff esponenziale è la risoluzione ai contenziosi del canale, il suo funzionamento è quello di generare un numero random n compreso tra 0 ed m, questo tempo viene dato alla stazione come tempo di wait dopo il quale potrà riprovare ad inviare il messaggio

Queste migliorie nel protocollo formano il **CSMA/CAW** insieme ad ottimizzazioni utili per la trasmissione wireless

**Teoria delle code**

Distribuzione esponenziale (o Markoviana)

Con rho si indica il fattore di utilizzazione dei serventi, ovvero il rapporto tra la frequenza media degli arrivi e la velocità del servizio per il numero di serventi

La teoria delle code studia i casi in equilibrio e non transitori

**3 – Livello Network**

Il livello di rete ha il compito di nascondere l’infrastruttura di rete (al livello 4 suo superiore) quando ci si interfaccia con le altre reti

**Nomeclatura**:

* Host (end-node)
  + Stazione su cui opera lo strato di trasporto (livello 4)
* Pacchetto
  + Insieme di dati+header, unità di trasmissione del livello 3
* Router
  + Stazione intermedia per l’invio e la recezione dei pacchetti

Nella topologia di rete, tra una stazione e l’altra sono presenti più nodi e quindi più percorsi possibili. I nodi possono anche implementare tecnologie diverse

Lo strato di rete avrà i compiti di:

* Determinare tra i possibili tragitti, qual è il migliore e quale tragitto sarà da seguire
  + Può richiedere di conoscere la topologia
* Reagire a modifiche della topologia
  + Questa in caso di congestioni nei nodi, oppure, modifiche/aggiunte/rimozioni di nodi

Inizialmente OSI prevedeva un servizio **connection oriented** che voleva imitare la commutazione di circuito (simile al servizio telefonico)

Ora si utilizza un servizio **connection less:**

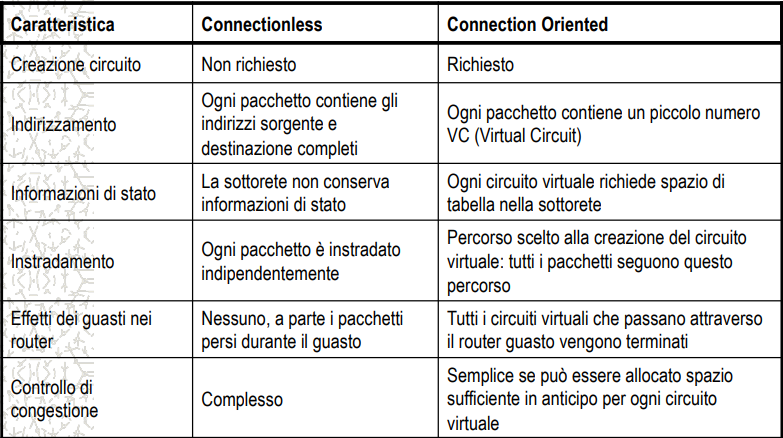
* Presenta un’inaffidabilità sulla connessione in quanto ci si basa principalmente sullo strato di trasporto
* Non presenta il concetto di stato
  + Non esistono meccanismi di gestione dello stato nel caso del fallimento di un router

A differenza del livello inferiore, non è possibile applicare backward learning visto al livello precedente in quanto si creerebbe una tempesta di broadcast per ogni nodo nella rete

Ogni pacchetto, infatti, presenta un mittente e un destinatario **NON** indicati (solo) con l’indirizzo MAC.

Bensì si utilizza un indirizzo **IP** che contiene informazioni sulla rete verso cui instradare i pacchetti.

Inoltre, per la variabilità prima definita del livello di rete, questo indirizzo IP ha anche la caratteristica di cambiare

Differenze tra **Connectionless** e **Connection Oriented**:

**Caratteristiche strato rete:**

* La funzione principale è l’instradamento (routing)
  + Questo permette di scegliere e inviare il pacchetto attraverso il percorso più adatto

**Algoritmi di Routing**

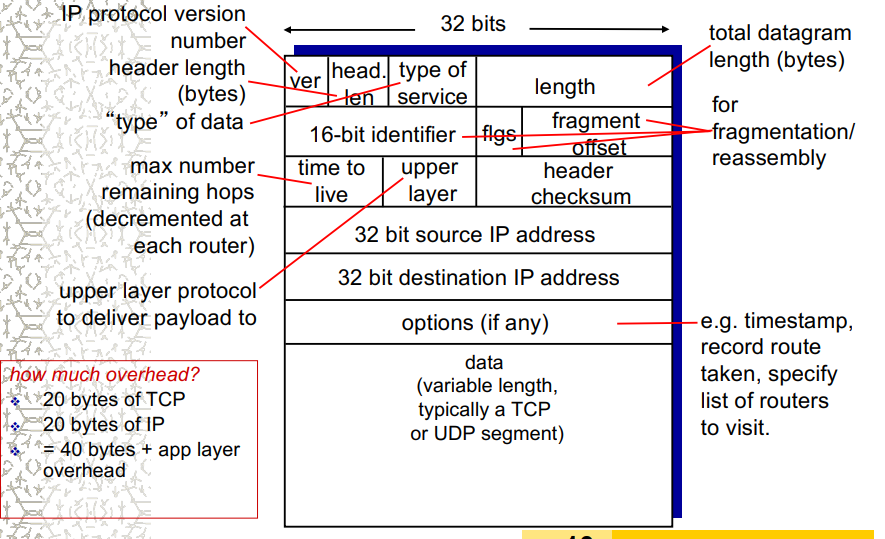
Deve presentare queste caratteristiche:

* Correttezza
* Semplicità
* Robustezza
  + Riuscire a resistere a modifiche alla topologia
* Stabile
  + Converge verso l’equilibrio
* Imparzialità
* Ottimizzato

**Internet Protocol (IP)**

È definito negli RFC 791 e 1122.

**Pacchetto IP**

Composto da multipli di 32 Bit, il suo datagramma è:

I pacchetti possono comporre un’informazione molto più grande per via della frammentazione (ogni pacchetto max 65kB), infatti questi vengono ricomposti grazie al loro id nell’header del pacchetto

L’indirizzo IP è univoco e presenta due livelli:

1. Indirizzo di rete
2. Indirizzo di host

Un **indirizzo IP** è costituito da 32 bit e composto da 4 numeri compresi tra 0 e 255

Di cui i primi due numeri rappresentano la rete e gli altri 2 l’host (classe b)

**CLASSI**

* **Classe A:**
  + Il primo numero compreso tra 0 e 127 (il primo bit è 0)
  + Il primo byte invece è utilizzato per identificare la rete, gli altri 3 l’host
  + Sono possibili 125 reti (27 - 3). Le reti 0, 10 e 127(interfaccia loop) sono riservate
  + Può contenere 224 host possibili
* **Casse B:**
  + Il primo numero va da 128 a 191, ovvero i loro primi bit sono 10
  + I primi due numeri identificano l’indirizzo di rete, mentre gli altri per l’host
* **Classe C:**
  + Indirizzi che iniziano per 110, ovvero quelli compresi tra 192 e 223
  + 3 byte per la rete e 1 byte per l’host
* **Classe D:**
  + Indirizzamento multicast, iniziano per 1110
* **Classe E:**
  + Indirizzi sperimentali di ricerca, iniziano per 1111

Sono però riservati alcuni indirizzi speciali:

* 10.0.0.0 indica la rete 10
* 255.255.255.255 indirizzo di **broadcast**
* 127.0.0.0 è dedicata all’interfaccia di **loopback**

Inserire tutti 1 in binario (255) nella sezione degli host in un indirizzo, indica un broadcast nella rete specificata nel campo rete

**VLSM**

Subnetting a lunghezza variabile -> spreca meno bit per l’indirizzamento nella rete, invece della lunghezza fissa si vede quanti sono gli host collegati in una subnetwork piuttosto che in tutta la rete

* Per i collegamenti punto-punto si usa la netmask /30 lasciando solo 2 bit per l’host id
* Per le reti con 14 < numero di host < 30 si usa /27
* Per 6< numero di host <= 14 si usa /28

Per evitare un sovraffollamento di entry nel router per le subnet, queste vengono aggregate tramite un elemento centrale che conosce il subnetting (ovvero ha le entry per poter reindirizzare alle subnet) a cui poi è collegato l’esterno, evitando che questo si riempia di entry per ricordare tutta la subnet

Per non causare spreco nell’uso di una classe con host id, invece di utilizzare un’altra classe con un numero maggiore (da classe C = 254 indirizzi, ad una classe B per esempio). È possibile aggregare più indirizzi ip **contigui** (ovvero uno dopo l’altro, tipo x.x.144.x contiguo con x.x.145.x e x.x.146.x). Questa tecnica è detta **supernetting**

Il routing quindi è un processo molto più complesso dello switching, in quanto necessità anche un’altra informazione oltre all’indirizzo. È necessaria la maschera di rete per conoscere la suddivisione in subnet o nel caso la parte comune di indirizzi contigui.

Dato un indirizzo IP verso cui instradare un pacchetto e delle tabelle di routing, è preferibile indirizzare il pacchetto verso quella più lunga perché così potrà arrivare a ritroso alle altre finché non farà **prefix matching**

**IPv6**

**Obiettivi**:

* Virtualmente illimitato il numero di indirizzi
* Ridurre il tempo di elaborazione nei router
* Sicurezza
  + Alla creazione dell’IPv4 non era molto aperto l’uso di internet = meno minacce
* Supporto per pacchetti di grosse dimensioni
  + Al tempo della creazione dell’IPv4 non erano grandi
* Permettere future evoluzioni del protocollo
* Gestire i protocolli abbinati al protocollo di rete

**Caratteristiche**: Notazione caratterizzata da 8 gruppi di 4 cifre **esadecimali** separate da “:”

**Ottimizzazioni** per la rappresentazione:

* Omettere gruppi di zeri ad inizio gruppo
  + 0030 –diventa-> :30:
* Omettere gruppi di zeri consecutivi, rappresentandoli con più “:”
  + Es. 8000:0000:0030: … --diventa-> 8000::30: …

**Pacchetto IPv6**

* **Header** a dimensione fissa di 40 bytes
  + Campo version
    - 6
  + Campo traffic class
    - Nel caso serva un instradamento particolare
      * Tipo prioritario, tipo voce, tipo video stream
  + Campo flow label
    - Usato per identificare i pacchetti provenienti dallo stesso flusso di trasporto
  + Campo payload lenght
    - Dimensione pacchetto senza header
  + Campo next header
    - Protocollo di trasporto
    - Possibili header condivisi o completati da altri header di altri pacchetti successivi o precedenti
  + Campo hop limit
  + Campo TTL (Time To Live)
    - Significa quanti **salti** può effettuare il pacchetto prima di essere eliminato
  + **NON c’è più** il campo **checksum**

All’header è possibile aggiungere un’estensione (**Estention Header**), questi sono i tipi che possono implementare le omonime opzioni:

* Opzioni hop-by-hp
* Opzioni di destinazioni
* Opzioni di routing
* Opzioni di frammentazione
* Autenticazione
* Cifratura

IPv4 e IPv6 per un po’ devono ancora essere necessariamente esistenti contemporaneamente. Da ciò è necessario permettere la comunicazione tra i due tipi di IP.

Questo avviene attraverso un processo **tunnel**

IPv6 compatibile con IPv4:



IPv4 mapped:

Immagine che contiene diagramma

Descrizione generata automaticamente

I più diffusi sono questi ultimi, e permettono la comunicazione diretta tra IPv4 e IPv6

Ovviamente per la comunicazione effettiva tra questi due sono necessari nodi di rete **dual stack** ovvero che riescono ad interpretare entrambi gli indirizzi.

**Tunneling**

Immagine che contiene diagramma

Descrizione generata automaticamente

Tra i nodi A e B si parla in IPv6, poi da B ad E si passa attraverso un tunnel IPv4 (nodi C e D)

**DNS**

Il DNS è necessario per la comunicazione tra IPv4 ed IPv6 in quanto questo servizio mappa (**risolve**) l’indirizzo IPv6 in IPv4.

**Strategie di Instradamento**

**Statico:**

* Il router si basa su informazioni note a priori, queste non vengono modificate e il percorso viene scelto collegandosi manualmente al router.
* Si specificano una route di default per l’instradamento di tutti i pacchetti che non possiedono una route specifica

**Dinamico**

* Non vengono definite manualmente le route
* Si costruisce una tabella che contiene tutte le informazioni per l’instradamento dei pacchetti
  + La tabella viene costruita con una tecnologia comune ai vari router, che comunicano e si scambiano informazioni necessarie
* La route viene quindi scelta dinamicamente, in base ad una **metrica**, ovvero un insieme di algoritmi che permette di scegliere il percorso migliore (in quanto è un problema di ottimizzazione)
  + Una **metrica** basa i propri algoritmi su diverse variabili
    - Larghezza di banda
    - Numero di salti
    - Protocollo utilizzato
    - …
* La **distanza amministrativa** è un’informazione aggregata (sfruttata dalle metriche) che definisce quanto è **distante** (o costoso) un router con cui comunicare, logicamente, più questa distanza è breve e più è sicura la comunicazione

**Algoritmi di routing**:

* **Non adattativi**
  + Usano criteri fissi di instradamento. Questo li rende statici e dererministici
* **Adattativi**
  + Utilizzano un’analisi delle variabili aleatorie presenti in ogni istante per la definizione del percorso migliore

**Algoritmi non adattativi**

* **Fixed Directory Routing**
  + L’algoritmo si basa sull’uso di una tabella di instradamento (riempita degli altri router dal gestore) assegnata ad ogni nodo. L’algoritmo, quindi consulta la tabella per comunicare con il più vicino al nodo considerato. Uno degli svantaggi è che in caso di un guasto va riconfigurata la tabella dal gestore
* **Basati su Flooding**
  + Come il backward learning, instrada il pacchetto su tutte le porte, eccetto quella di arrivo. Massimizzando la probabilità di arrivo di questa informazione pur causando un alto volume sulla rete

**Algoritmi adattativi**

* **Routing Centralizzato**
  + Un RCC (routing control center), conosce tutta la rete. Calcola, distribuisce e gestisce tutte le tebelle di instradamento per ogni router, è applicabile ad una rete piccola poiché è difficile raccogliere tutte le informazioni in un solo punto. Gli algoritmi di calcolo per le route migliori sono complessi, e in una rete grossa verrebbe sopraffatto dalle informazioni
* **Routing Isolato**
  + È la “soluzione” alla scalabilità, infatti, ogni router si occupa di calcolare le proprie tabelle di instradamento
* **Routing Distribuito**
  + Fonde le due soluzioni sopra, infatti, non si ha un solo RCC, ma esiste una divisione **settoriale** (esistono rcc settoriali che dirigono una sottorete), infatti, ogni nodo poi contribuisce all’aggiornamento delle tabelle di instradamento. Questo porta ad una “sincronizzazione”

**Algoritmi di Flooding**

Ogni pacchetto viene inoltrato su tutte le linee di uscita meno quella che ha inviato il pacchetto, per evitare la duplicazione eccessiva di questi pacchetti si può:

* Dotare i pacchetti di un contatore che quando raggiunge lo 0 vengono eliminati
* I router possono tenere traccia dei messaggi ricevuti e inviati, evitando di duplicare messaggi già elaborati
* Si può utilizzare il **flooding selettivo**, ovvero, si fa flooding solo su una “cerchia ristretta” di destinatari

Spesso si utilizzano come benchmark perché i pacchetti seguono il percorso a loro più breve possibile

Ovviamente questo algoritmo è inefficiente in quanto utilizza tutte le linee della rete (includendo molte inutili e dispendiose)

**Algoritmo di Dijkstra**

Una rete può essere rappresentata con un grafo. E, allora per raggiungere un nodo x partendo da un nodo y, si può utilizzare **l’algoritmo di dijkstra** per trovare il costo minimo degli archi (pesati) da percorrere. Una delle limitazioni dell’algoritmo di dijkstra è quello di avere il peso di ogni arco maggiore di 0.

**Algoritmo di Bellman-Fort** (detto **distance vector**)

Il funzionamento di questo algoritmo adattivo, ad ogni nodo del percorso si assegna un costo ai nodi a lui vicino, assegnando valori +x in base all’influenza sul costo totale. A differenza di dijkstra non utilizza una tecnica greedy per comporre una soluzione totale, questo processa tutti gli archi e lo fa |V|-1 volte, quindi avendo una complessità O(|V|\*|E|).

* Quindi ad ogni **arco** (**linea**) viene assegnata una **distanza** valutata in base alla **metrica** (considerata poi come costo), questo si basa su un **costo massimo** da rispettare per essere “**raggiungibile**” e quindi anche **preferibile**
* Le informazioni vengono scambiate poi con tutti i router adiacenti
* Ogni nodo è raggiungibile indirettamente tramite i suoi vicini
* In base ai vicini poi viene costruita una tabella di routing

Questo algoritmo a differenza di Dijkstra può elaborare e risolvere costi negativi sugli archi pesati

Uno dei problemi di questo algortimo è quello di poter avere cicli infiniti nel caso della rottura di un percorso necessario, le soluzioni possono essere:

* **Split Horizon**
* **Poison Route**
* **Poison Reverse**
* **Hold-Down Timers (il migliore)**
  + È il migliore in quanto non si basa sulla conoscenza di un nodo certamente non funzionante; infatti, si basa su un tempo di attesa massimo. Oltre il quale indica il percorso come invalido (questo potrebbe anche essere funzionante ma molto lento)

**Algoritmi di Routing link state**

Si basano sull’invio di pacchettti contenenti le informazioni sul costo dei percorsi. La propagazione di questi si basa sulla tecnica di flooding

I passi fondamentali sono:

* Individuare i vicini
* Misurare il costo per raggiungere i vicini
* Costruzione e integrazione dei costi nei pacchetti
* Invio dei pacchetti a tutti i router vicini

Questo algoritmo non è scalabile per ordini molto grandi, in quanto poi la dimensione della tabella aumenterebbe esponenzialmente. Infatti, veniva utilizzato per reti piccole come arpanet agli inizi o anche in reti domestiche ai giorni d’oggi.

La soluzione a questo è il **routing gerarchico**, ovvero, un routing in cui ogni router “gestore” conosce le informazioni di tutta la sua rete e le informazioni necessarie per raggiungere il prossimo router “gestore” che nasconde la sua infrastruttura e le sue informazioni in una gerarchia superiore visibile solo a lui

**OSPF**

È un algoritmo (gerarchico) di instradamento molto reattivo, che si basa sulle informazioni nell’header dei pacchetti basando l’instradamento sul tipo di instradamento

La struttura gerarchica di OSPF è basato su un router di confine che rappresenta il nodo radice di un albero i cui nodi e foglie rappresentano un **sistema autonomo** (collegando tutto alla radice attraverso dei nodi **backbone**), e proprio questo **router di confine** serve per l’interconnessione con altri sistemi autonomi (**peering**)

**Peering**

Nel **peering** esistono i ruoli di provvider-customer(gerarchico) e peer-peer (equo). Il peering ha bisogno di essere regolato da organizzazioni (telecom ecc) per concedere una sicurezza di comunicazione (l’highjacking è uno degli attacchi hacker possibili in cui ci si finge un peer autorevole o si sniffa una connessione tra due peer facendola passare per un terzo peer intermedio malevolo)

L’organizzazione tra AS (sistemi autonomi) è basata su:

* Unico algoritmo di instradamento,
* Utilizzando percorsi statici
  + Nascondendo l’interno dell’AS alla rete, che viene poi gestito utilizzano un altro algoritmo
* Exterior Gateway Protocol
  + Usano il BGP (border gateway protocol)
  + Protocollo per le informazioni di routing tra i router di confine degli AS

Tutti i router interni ad un AS conoscono giusto le regole di instradamento e le informazioni necessarie giusto per raggiungere tutto l’AS e il proprio Gateway (router di frontiera/confine). Mentre ogni router di gateway possiede anche le informazioni per il peering con gli altri router di confine. Riducendo esponenzialmente la complessità dell’algoritmo

**4 – Livello Trasporto**

Ha lo scopo di permettere la comunicazione e il trasferimento dei dati end to end, **nascondendo** **completamente** i livelli inferiori (tipo di rete, topologia, tecnologia, complessità ecc) a quello suo superiore.

* Il superiore in OSI è il livello di sessione
* In TCP/IP è il livello applicazione