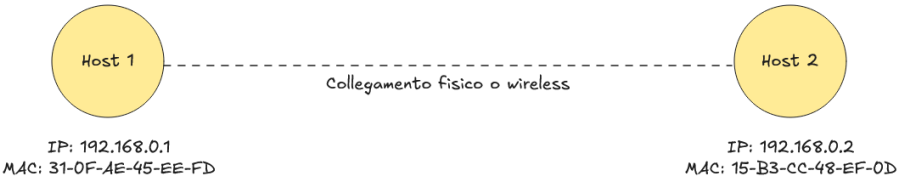
Local Area Network

All'interno di una rete LAN (Local Area Network) non possiamo operare con i protocolli di rete dato che al suo interno vi sono apparecchiature di livello collegamento e quindi sprovvisti di logica interna, dobbiamo trovare metodi alternativi a OSPF, RIP.

Indirizzo MAC

L'**indirizzo MAC** (Media Access Control) è un identificatore **univoco** assegnato a ogni **scheda di rete (NIC)**, registrato nella **ROM** del dispositivo.

- È lungo **48 bit** e rappresentato in **esadecimale** (es. `00:1A:2B:3C:4D:5E`).
- Ogni dispositivo in rete possiede sia un **indirizzo IP** (**volatile**, cambia con la configurazione di rete) che un **indirizzo MAC** (**permanente**, legato alla NIC).
- Viene usato localmente per inviare un frame da un dispositivo a un altro connesso fisicamente.



L'**IEEE** gestisce la distribuzione degli indirizzi MAC per garantire la loro **unicità globale**. Dato che i dispositivi si possono spostare fisicamente di LAN. Si usano le tabelle ARP per associare indirizzi IP con MAC, per ogni riga abbiamo `<Indirizzo IP, Indirizzo MAC, TTL>.`

Protocollo ARP

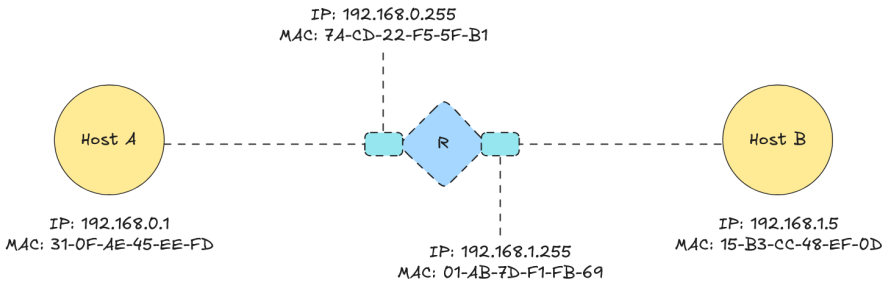
Il protocollo ARP (Address Resolution Protocol) serve per conoscere il MAC di un dispositivo dato il suo indirizzo IP, siamo nella situazione in cui A vuole inviare un messaggio a B senza conoscere il suo MAC localmente.

1. Viene inviata in broadcast una **ARP Query** la quale contiene l'indirizzo IP di destinazione ovvero B.
2. Quando B vede che la richiesta contiene il suo IP invia una **ARP Response** con il suo MAC.
3. A questo punto A aggiunge la entry nella ARP table e salva così l'informazione di B.

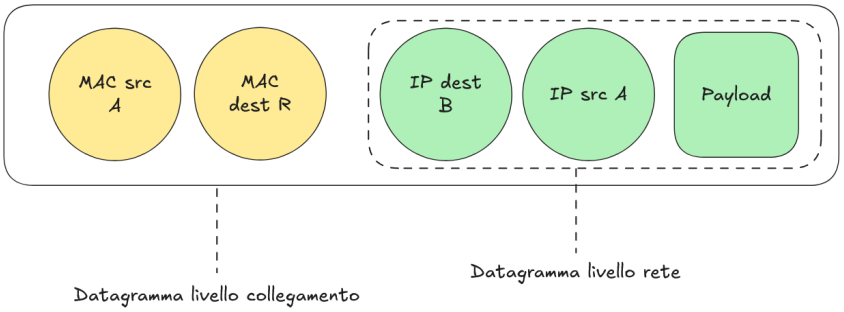
Routing verso un'altra sottorete

Nel momento in cui non si lavora più localmente ma ci si interfaccia con altre reti dobbiamo avere delle informazioni come l'IP e MAC di R e l'IP di B, queste informazioni le otteniamo così:

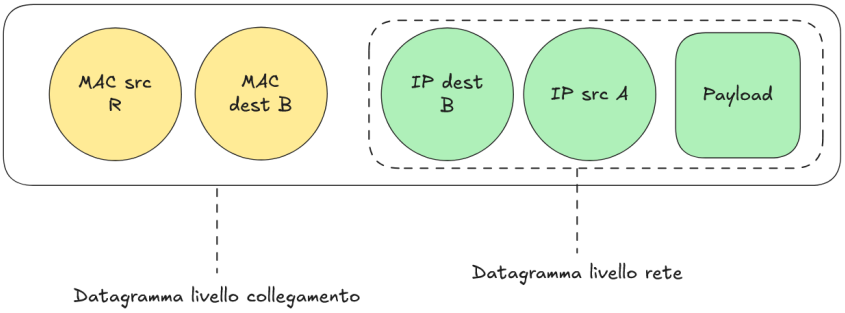
IP di R con la configurazione **DHCP, MAC di R** con una richiesta **ARP**. **IP di B** con una interrogazione **DNS**.



1. A vuole inviare un messaggio a B, crea un **datagramma IP con sorgente A e destinazione B**, poi crea un frame di livello collegamento con **MAC address di sorgente A e destinazione R**.



2. Il router riceve il frame di livello collegamento da A, **rimuove l'intestazione** verifica in quale interfaccia si trova B e crea un **nuovo frame** con **MAC** address **sorgente R e destinazione B.**



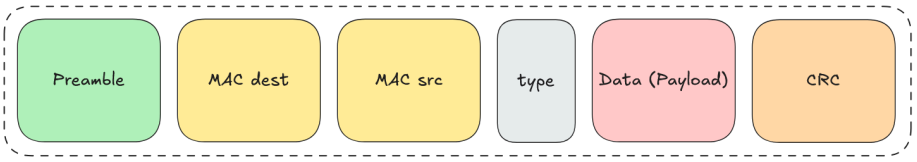
3. A questo punto B riceve il frame e può leggere il datagramma, inviandolo ai livelli superiori.

Ethernet

Tecnologia principale usata nelle reti cablate perchè è una soluzione economica, semplice e può evolvere facilmente con le tecnologie attuali. Ci sono due modalità di utilizzo:

- **Bus**: tutti i nodi sono nello stesso dominio di collisione.
- **Switched**: i nodi sono indipendenti grazie a uno switch che interrompe il dominio di collisione.

Struttura del pacchetto



- **Preamble**: Serve per sincronizzare mittente e destinatario, ha un valore di sette byte di `10101010` e un finale byte di valore `10101011`.

- **Indirizzi MAC**: indirizzi a 6 byte, se l'adattatore del NIC riceve un MAC dest uguale al suo lo accetta altrimenti viene scartato.
- **Type:** specifica il protocollo di livello rete.
- **Data:** payload con i dati dei livelli superiori.
- **CRC:** Cyclic Redundancy Check eseguito dal destinatario per verificare l'integrità del frame.

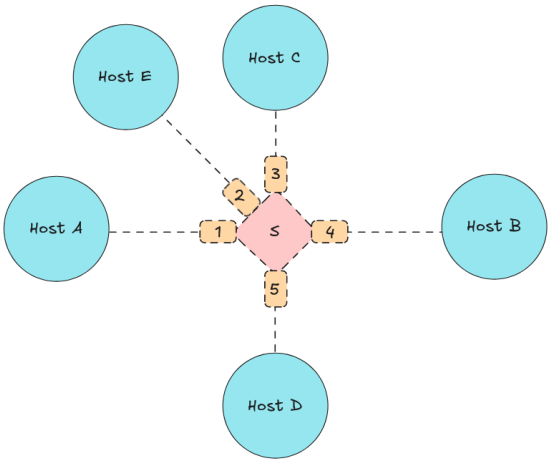
Protocollo Ethernet 802.3

Le caratteristiche di Ethernet sono:

- **Connectionless**: i NIC di mittente e destinatario non fanno handshake
- **Unreliable**: il NIC del destinatario non manda ACK e NACK, i dati sono scartati, vengono recuperati solo nel momento in cui passano a livello trasporto TCP dove avviene la ritrasmissione.
- Il protocollo usato per l'invio è unslotted **CSMA/CD** con backoff binario.
- Esistono molti standard di Ethernet

Switch

Lo switch è un apparecchio di livello collegamento che serve per inoltrare i frame. Esaminano i MAC e decidono, usano CSMA/CD per l'accesso al canale. Si dice trasparente perchè un host A non ha idea che in mezzo alla comunicazione con B c'è uno switch. Si dice anche **plug-and-play** perchè non ha bisogno di configurazioni e **self-learning** perchè impara le destinazioni durante l'utilizzo.



Algoritmo di inoltro e filtraggio

L'algoritmo che segue uno switch quando arriva un frame da una interfaccia è il seguente:

1. Registra il MAC e l'interfaccia da cui proviene (self-learning) nella switch table.
2. Cerco nella switch table il MAC del destinatario con relativa interfaccia
3. Se trova la destinazione
    a. Se il destinatario è sulla stessa porta del mittente: drop
    b. Se il destinatario è su un altra porta: inoltra
4. Altrimenti eseguo un flood

Flood: inoltro su tutte le altre interfacce tranne quella da cui arriva il frame.

Gestione di connessioni multiple

Ogni host ha una connessione dedicata con lo switch, lo switch per ogni interfaccia ha un buffer, su ogni link si usa il protocollo Ethernet quindi nessuna collisione e full duplex (bidirezionale).

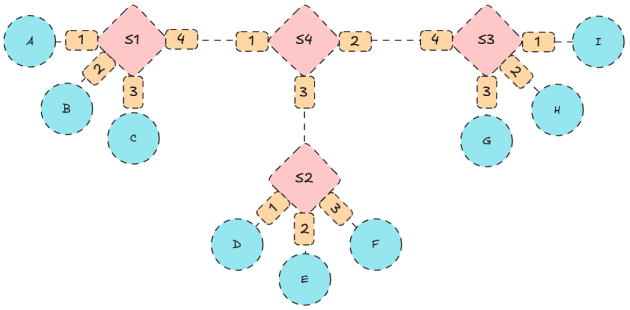
Limitazioni dello switch: A può trasmettere a B simultaneamente a D che trasmette a C ma se D provasse a trasmettere ad B ci sarebbe una collisione dato che il canale è già occupato.

Tabelle di inoltro di uno switch

Ogni switch ha una tabella per cui una entry è composta da `<Indirizzo MAC, Interfaccia, TTL>`. Uno switch apprende dove si trova un dispositivo grazie alla prima esplorazione DHCP che esegue un host appena collegato alla rete.

Switch interconnessi

La gestione dietro una rete di switch interconnessi non richiede logica aggiuntiva, gli switch che hanno collegati direttamente gli host semplicemente salvano la direzione del MAC e da quale interfaccia proviene, ad esempio S4 ha salvato nella sua switch table che B proviene dall'interfaccia 1.



Switch e Router

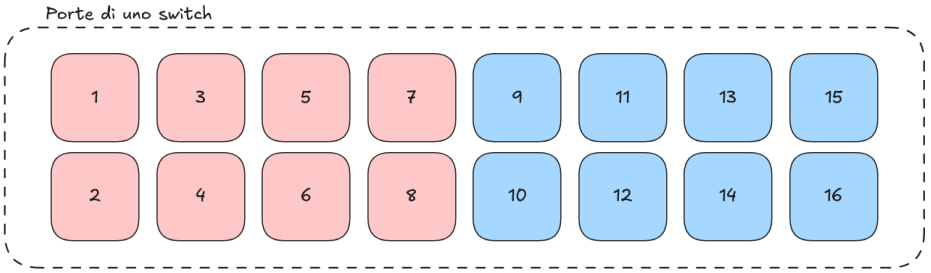
- **Entrambi sono store-and-forward**: i routers esaminano l'intestazione del pacchetto di rete e lo inoltrano grazie alle routing table. Gli switch esamino l'intestazione del frame a livello collegamento e tramite tabelle interne inoltra.
- **Entrambi hanno tabelle di inoltro**: i router creano le tabelle di inoltro tramite gli algoritmi sull'IP mentre gli switch creano le tabelle di inoltro tramite apprendimento basato sull'indirizzo MAC.

Le VLAN

Con le LAN abbiamo diversi problemi legati al modo in cui funzionano:

- **Dominio di broadcast unico**: il traffico di livello collegamento in broadcast (ARP, DHCP, flooding) può creare problemi di efficienza.
- **Problemi amministrativi**: gli utenti devono essere fisicamente presenti dentro la rete, se si sposta un dispositivo dobbiamo anche cambiare lo switch al quale è attaccato.

VLAN basate sulle porte

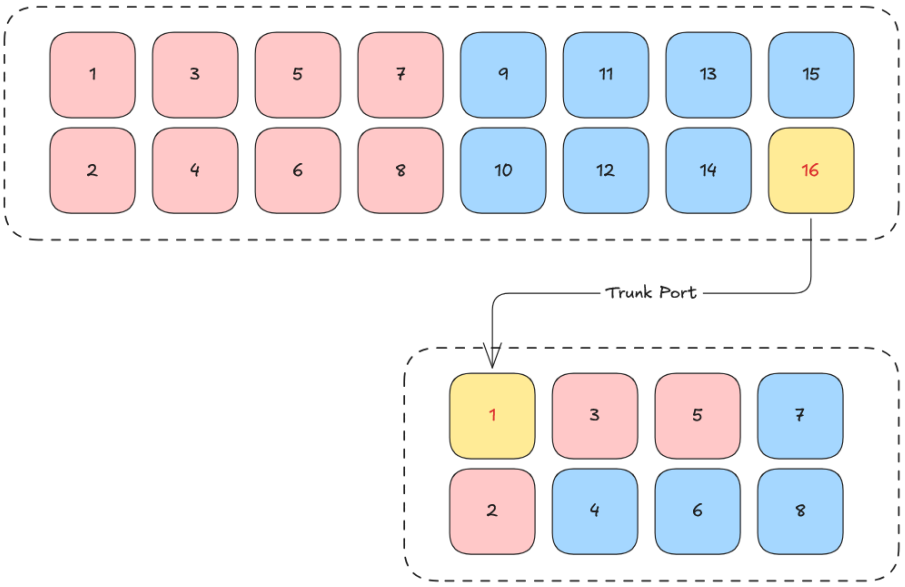


Uno switch ha molte porte che sono di un unico dominio, tuttavia con software interni possiamo impostare più domini di broadcast sullo stesso switch:

- **Traffic Isolation:** i frame della VLAN1 (1-8) non sono visti dai dispositivi della VLAN2 (9-16).
- **Dynamic Membership:** possiamo cambiare dinamicamente l'appartenenza di una porta a una determinata VLAN.
- **Forwarding tra VLANs:** per consegnare un frame della VLAN1 serve del routing, effettivamente le VLAN sono due sottoreti distinte.

VLAN con switch multipli

Le **trunk port** sono porte di uno switch che permettono di trasportare il traffico di più VLAN contemporaneamente tra switch diversi. Dobbiamo usare un protocollo diverso da 802.3 vanilla, serve qualcosa per identificare a quale VLAN appartiene il frame, usiamo 802.1q.



Protocollo 802.1Q

Il protocollo 802.1Q serve per ampliare il frame Ethernet e aggiungere due cambi fondamentali per far viaggiare i pacchetti, attraverso un software infatti si inseriscono i campi.

- **Tag Protocol Identifier**: valore sempre impostato a `0x8100.`
- **Tag Control Information**: valore che definisce l'ID della VLAN di appartenenza del frame.