REDES LAN

Contenido

[Tecnología Ethernet 2](#_Toc115626221)

[Operación en el modelo OSI: capa 2 (Enlace) 2](#_Toc115626222)

[Topología original red LAN Ethernet: Bus 3](#_Toc115626223)

[Formato de la trama Ethernet / 802.3 4](#_Toc115626224)

[Diferencia entre trama IEEE 802.3 y Ethernet 5](#_Toc115626225)

[Direcciones MAC 6](#_Toc115626226)

[Transmisión 7](#_Toc115626227)

[Tiempo de transmisión 7](#_Toc115626228)

[Mecanismo CSMA/CD 7](#_Toc115626229)

[Algoritmo exponencial binario (backoff exponencial) 8](#_Toc115626230)

[Dominio de colisión y broadcast 9](#_Toc115626231)

[Interfaz full-duplex 10](#_Toc115626232)

[Formas de comunicación (x-cast) – repaso 10](#_Toc115626233)

[Dispositivos de red 11](#_Toc115626234)

[Hub 11](#_Toc115626235)

[Bridging 11](#_Toc115626236)

[Bridging Loop y Protocolo Spanning Tree (STP) 12](#_Toc115626237)

[Switch 14](#_Toc115626238)

[Modos de operación 14](#_Toc115626239)

[Loop y STP 15](#_Toc115626240)

[Virtual LANs (VLANs) – Capa 3 15](#_Toc115626241)

[VLAN Tagging o Trunking (capa 2) - Protocolo 802.1Q 15](#_Toc115626242)

[Cómo funciona 15](#_Toc115626243)

[La etiqueta 16](#_Toc115626244)

[Conexión a internet de una VLAN 16](#_Toc115626245)

[Servicio storage (ESX?) 17](#_Toc115626246)

[LACP 17](#_Toc115626247)

[Ejemplo 18](#_Toc115626248)

[Notas misc 18](#_Toc115626249)

# Tecnología Ethernet

* Es un estándar de redes de área local
* Es una red Peer to Peer donde el control está totalmente descentralizado. Todos los participantes son iguales y tienen el mismo acceso a los recursos.
* El principio de funcionamiento original se conoce como CSMA/CD (Carrier-Sense Multiple Access / Colision Detection) o Acceso múltiple por detección de portadora con detección de colisiones.
* Originalmente esta tecnología es half-duplex (ambos hablan pero solo uno a la vez).

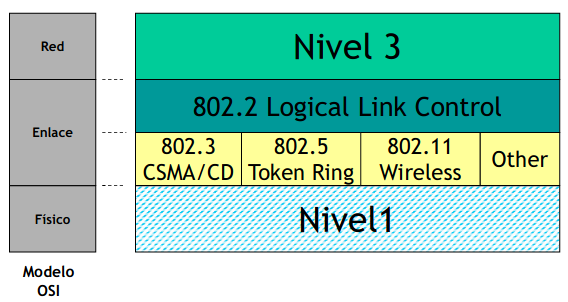
La IEEE 802.3 estandarizo el CSMA/CD que es lo que creó Ethernet (luego incorporó a Ethernet). Por lo que es lo mismo decir **protocolo 802.3 y Ethernet.**

**Evolución**

Tuvo diferentes nombres y velocidades.

El formato de trama se mantuvo siempre, más allá de los estándares y la velocidad.

# Operación en el modelo OSI: capa 2 (Enlace)

****

El protocolo 802.3 (CSMA/CD) (o Ethernet) opera en la capa de enlace (capa 2). Pero no cumple con todas las especificaciones de la misma según el modelo OSI, por lo que la IEEE estandarizo el protocolo 802.2 que viene a cumplir lo que los otros protocolos (802.3, 802.5, etc) no hacen.

A estas dos se la llaman subcapas:

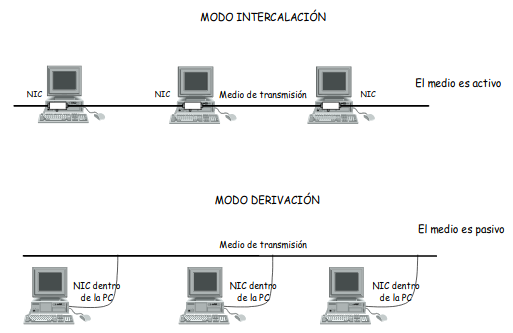
* **Subcapa superior: Control de Enlace Logico LLC (Logical Link Control):** La subcapa LLC de Ethernet se ocupa de la **comunicación entre las capas superiores y las capas inferiores.** Se implementa mediante software (no depende del hardware).
* **Subcapa inferior: Control de Acceso al medio MAC (Medium Access Control):** La MAC se implementa mediante hardware y va a depender de la tecnología. Esta subcapa tiene dos responsabilidades
  + **Encapsulación de datos**
  + **Control de acceso al medio.**

# Topología original red LAN Ethernet: Bus

Topología original: Bus. Un cable general compartido y único, al que se conectan las estaciones.

Se dice topología *original* porque el BUS hecho con cable coaxil, que era inflexible, se suplantó por cable UPC. Se usa esta topología para explicar porque explica bien el funcionamiento y es fácil de interpretar.

Modos:

* **Modo derivación (tecnología original, primero):** es también llamado **coaxil grueso**, las unidades se iban “colgando” del coaxil.
* **Modo intercalación (después):** es también llamado **coaxil fino**, iba interconectando por las diferentes unidades con un conector T que permitía que el cable siga a la siguiente estación.

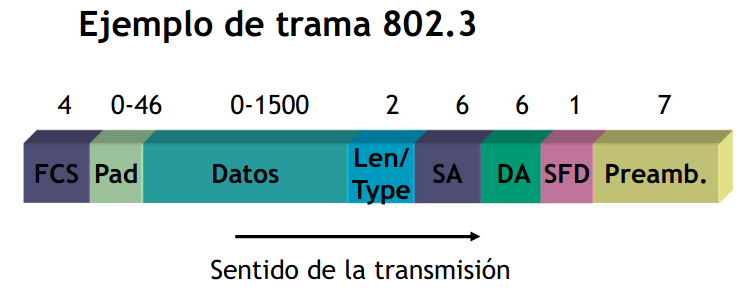
En estas transmisiones se usan códigos de banda base. Si una transmite, las otras escuchan. Punto -> multipunto.

Problemas que se deben resolver:

* Quién es el destinatario de una transmisión.
* Mec**anismo para regular la transmisión**. Controlar quién y cómo accede al medio.

Si **dos estaciones intentan transmitir simultáneamente, sus señales se superpondrán y serán erróneas**; también se puede considerar la situación en que una **estación decide transmitir continuamente durante un largo periodo de tiempo.**

# Formato de la trama Ethernet / 802.3

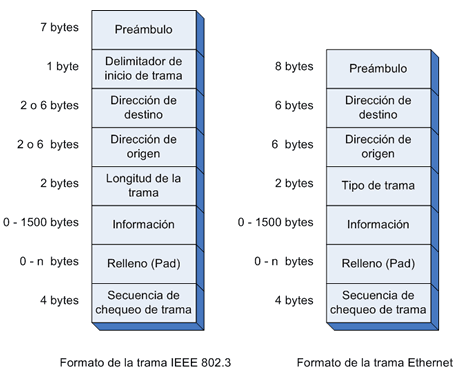


Longitud de la trama:

* **Mínima**: 64 bytes. Estos bytes aseguran que en un largo máximo de 2500m se pueda detectar la colisión antes de terminar de emitir la señal. A esto se lo llama **ventana de colisión**. Si fuera por debajo de 64bytes nunca se enteraría que su transmisión colisionó con otra.
* **Máxima** de 1518 bytes (sin incluir preámbulo ni SFD).

|  |  |  |
| --- | --- | --- |
|  | Función | Formato (en bytes!) |
| Preámbulo | Establecer el sincronismo. Una estación la transmite, todas reciben la señal, empiezan a decodificar el código Manchester y encuentran el preámbulo hasta que encuentran la SFD. | 7 bytes/octetos   Contenido: todos contienen 10101010 |
| SFD (Start of frame delimiter) | Indica que finalizó el preámbulo y el periodo de sincronización, y lo que viene a continuación es el primer campo significativo | 1 byte  Contenido: 10101011 |
| DA (Destination Address) | Dirección destino (a quién va dirigido el mensaje). Es una MAC address. | 6 bytes/48bits. MAC address |
| SA (Source Address) | Dirección origen (quién transmite el mensaje). Es una MAC address. | 6 bytes/48bits. MAC address |
| [VLAN TAG – 802.1Q only] | [Cuando se usa VLAN Tagging (802.1Q), acá se inserta la etiqueta] |  |
| Len (length) / Type | Es una de las diferencias entre Ethernet2 y 802.3.  - Len (802.3): Indica cual es la longitud del campo de datos de longitud variable que sigue a continuación.  - Type o Ethertype (Ethernet2). Indica cuál fue la capa usuaria (la capa superior) del modelo OSI que origino el mensaje, así sabe a qué capa entregárselo. Valores posibles: 0800 -> IPV4   / 0806 -> IPV6 | 2 bytes |
| Datos |  | Longitud variable. Entre 0 y 1500 bytes |
| Pad | Campo de relleno. Sirve para asegurarse que la trama tenga al menos 64bytes (necesario para detección de colisiones) | 0-46 bytes |
| FCS (Frame Check Sequence) | CRC (Verificación de redundancia cíclica). Sirve para detectar si hay un error en la transmisión de la trama. Se aplica sobre dirección destino, dirección origen y data. | 4 bytes |

## Diferencia entre trama IEEE 802.3 y Ethernet

****

* **Preámbulo**. Difiere en la longitud. .
  + En Ethernet tiene una longitud de 8 bytes. No hay SFD (start of frame delimiter)
  + En IEEE 802.3 la longitud del mismo es de 7 bytes, en este último el octavo byte se convierte en el comienzo del delimitador de la trama.
* **LEN / Type. El uso de ese campo difiere entre los dos protocolos.** 
  + **LEN (802.3)**
  + **Ethertype (Ethernet**). Especificar el protocolo que es transportado en la trama. Esto posibilita que muchos protocolos puedan ser transportados en la trama.
  + Type (802.3). El campo tipo fue **reemplazado en el estándar IEEE 802.3 por un campo longitud de trama**, el cual es utilizado para indicar el número de bytes que se encuentran en el campo de datos.
* C**ampos de dirección**, tanto de destino como de origen.
  + El formato de **IEEE 802.3 permite el uso tanto de direcciones de 2 como de 6 bytes**
  + **El estándar Ethernet permite solo direcciones de 6 bytes.**
* La cuarta diferencia es que en la **capa de enlace la 802.3 tiene 2 subcapas LLC y MAC, la Ethernet no tiene esas dos capas.**

## Direcciones MAC

**MAC Address:** Dirección asociada a la interfaz de red. Cada dispositivo con placa de red (cableada o wireless) tiene una MAC address única de fábrica. La IEEE asigna el espacio de direcciones.



Está compuesta por 6 bytes (48 bits), subdivisibles en 2 campos de 3 bytes cada uno:

* OUI (Organizationally Unique Identifier) o identificador único del fabricante (primeros 3).
* DUI (Device Unique Identifier) o identificador único del dispositivo (de red).

Las MAC se suelen representar en hexadecimal, usando 1 carácter Hexa por cada 4 bits. Por ende, son de 12 letras.

0🡪 0000, F (15) 🡪 1111

**Tipos de direcciones MAC:**

* Dirección de Broadcast (48 unos). Únicamente puede ser destino. Indica que el mensaje está destinado para todos los nodos de la red.
* Dirección multicast: El mensaje va dirigido a un grupo de estaciones.
* Dirección unicast: Destinado a una única estación.

**Direcciones especiales**

* 0xFFFF.FFFF.FFFF: broadcast
* 0x0000.5e00.0000 - 0x0000.5eff.ffff : Multicast



# Transmisión

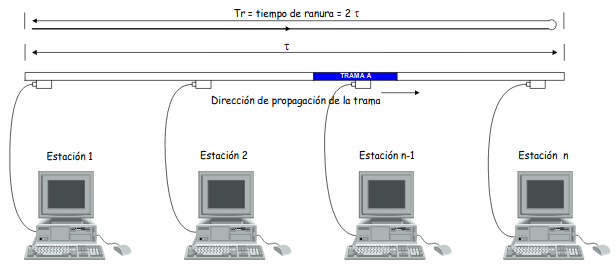
## Tiempo de transmisión

* Velocidad: 10Mbps
* Trama mínima: 64bytes
* Longitud máxima de red: 2500m
* Slot: 512 bits/10Mbps = 51,2 useg.

La **probabilidad de colisión aumenta a medida que el medio es más largo**. Por ello se estableció la limitación de la **longitud máxima de la red** (cable coaxil) de 500m, pero con la utilización de repetidores (máximo 4) se aumentó a 5x500m = 2500m.

Se establece la transmisión de una **trama mínima** de 64bytes. Estos 64bytes aseguran que en un largo máximo de 2500m el emisor pueda detectar la colisión antes de terminar de emitir la señal. A esto se lo llama **ventana de colisión**. Si fuera por debajo de 64bytes nunca se enteraría que su transmisión colisionó con otra.

**Slot**: Ranura de espera (**tiempo que se tarda en enviar 64bytes**). A una velocidad de 10Mbps 512 bits/10Mbps = 51,2 useg.

****

## Mecanismo CSMA/CD

#### 1. Envío

Escucha el medio para ver si está libre.

* No está libre: espera.
* Está libre: (no hay nadie transmitiendo), empieza a trasmitir el preámbulo y el resto de la trama.

¿Cómo se da cuenta que no hay nadie transmitiendo? Porque no hay señal (Manchester tiene señales en todos los intervalos, y no es posible que no se detecte una transmisión).

#### 1a. Transmisión exitosa

La trama se propaga por el medio y llega a todas las terminales, ahí finaliza la transmisión. En Ethernet no existe la confirmación, se asume exitosa la transmisión si no hay colisión.

#### 1b. En caso de colisión

Una colisión se da cuando dos o más estaciones comienzan a transmitir al mismo tiempo (ambas encontraron el medio libre y al mismo tiempo comenzaron). Las señales se interfieren mutuamente y por eso se vuelve imposible detectar qué se transmite. El resultado de una colisión es que ambas estaciones van a tener que transmitir en otro momento.

**Detección de colisiones:** Las estaciones detectan las colisiones porque al mismo tiempo que transmite, escuchan lo que hay en el medio, si lo que escuchan en el medio no es lo que están transmitiendo, detectan la colisión.

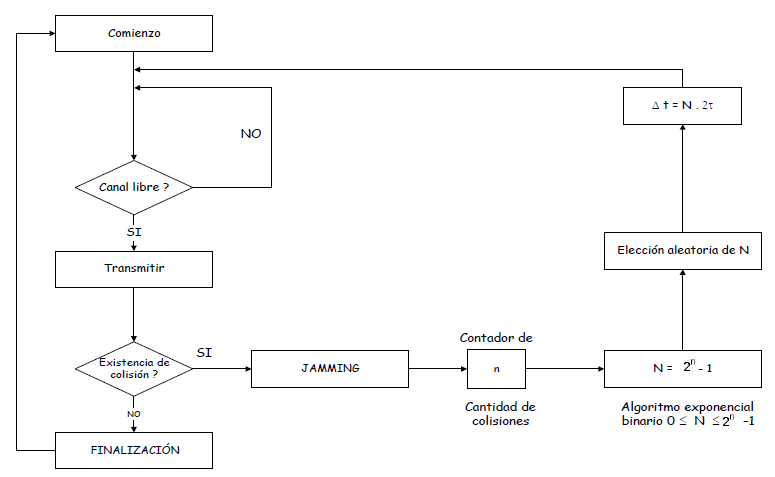
Al **detectar una colisión**, la estación pone en funcionamiento el Algoritmo exponencial binario (backoff exponencial), que lo hace es ordenar las transmisiones en el tiempo y buscar que no vuelvan a colisionar al retransmitir.

#### 2. Recepción

Todos reciben el mensaje.

1. Chequeo del CRC. Recibieron el mensaje correcto? Si no, se descarta. (Si hubo una colisión, el jamming hace que el CRC sea incorrecto y se descarte siempre)
2. Si es correcto, se revisa la dirección de destino. Si es esa estación o un broadcast, o un multicast en el que participa, se toma. Si no, se ignora.

### Algoritmo exponencial binario (backoff exponencial)



Backoff exponencial es un algoritmo que **se utiliza para espaciar retransmisiones** repetidas del mismo bloque de datos de manera multiplicativa, a menudo como parte de la **evitación de congestión de red.**

Si no se detecta colisión, se asume que el mensaje fue entregado exitosamente (no hay confirmación de error en este protocolo)

**Si hay colisión:**

* **Continúa transmitiendo con Jamming.** Esto es un código que no da CRC correcto, utilizado para sostener la transmisión una vez que se detectó la colisión para que las demás estaciones también la detecten.
* **Agrega una unidad al contador de colisiones** (n minúscula, que es una variable global). El algoritmo exponencial alimenta a una variable N, que se elige aleatoriamente entre 0 y 2^n -1. N representa la cantidad de slots que se debe esperar antes de intentar retransmitir. Este tiempo aleatorio previene una nueva colisión. A medida que crece el contador de colisiones, aumenta exponencialmente el tiempo entre transmisiones consecutivas.

Luego de 10 colisiones consecutivas, se setea el máximo de ranuras a 1023 para bajar la probabilidad de colisión.

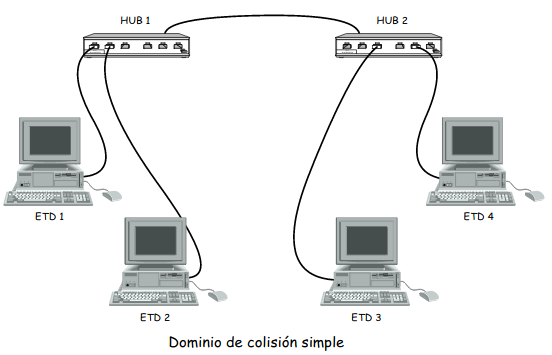
Luego de 16 colisiones consecutivas, la subcapa MAC aborta la misión. Se determina que la red no está accesible.

**Cuando logra transmitir, se reduce de a 1 el contador de intentos.**

N = (1/ probabilidad de colision) -1.

# Dominio de colisión y broadcast

* **Dominio de colisión:** segmento de una red donde es posible que las tramas puedan colisionar con otras.
* **Dominios de Broadcast:** Segmento de la red que involucra a todos los dispositivos que recibirán frames de Broadcast provenientes de cualquier dispositivo del conjunto. Los **routers son usados para dividir estos dominios**, cada uno de sus puertos pertenece a un dominio de broadcast.

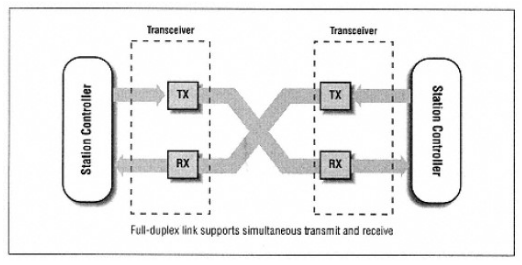


*En la imagen: los 2 hubs están unidos por cable de crossover que hacen que operen como un gran bus único. Por ende, hay un único dominio de colisión.*

# Interfaz full-duplex

Para obtener una comunicación full-duplex dos estaciones deben estar conectadas punto a punto con un vínculo full duplex.

* Doble cable, con conexión cruzada (de trasmisión de A a recepción de B, y viceversa): Caminos de transmisión y recepción independientes que operan en simultáneo
* Dos estaciones conectadas punto a punto con un vínculo full dúplex
* No hay contención -> se elimina el CSMA/CD. El canal de transmisión de una estación está vinculado al de la recepción de la otra, por lo que puede transmitir cuando quiera, incluso al mismo tiempo. Deja de ser necesario el CSMA/CD.
* La velocidad de transmisión no se divide entre las estaciones, es mucho mayor que con CSMA/CD



# Formas de comunicación (x-cast) – repaso

* Comunicación unicast. Se envía a un único destinatario definido.
* Broadcast: Comunicación de 1 origen a TODOS
* Multicast: mensaje dirigdo a un grupo de destinatarios, que se identifica por direcciones MAC grupales

# Dispositivos de red

(Ver comparativa en archivo de Modelo OSI)

## Hub

Dispositivo de capa 1 (física) del modelo OSI, solo entiende de señales eléctricas. **Combina las señales eléctricas de todos los puertos como si fuera un bus.** Cuando una estación transmite, todas las demás escuchan. Es half-duplex. Su funcionamiento es similar a un cable por lo que **solo cuenta con un dominio de colisión. Aunque parezca topología en estrella, opera como un bus.**

Las estaciones conectadas al HUB compiten por la capacidad del hubo y por eso pueden colisionar entre sí. Si el HUB es de 10mb, hay un único bus de 10mbps que va a ser utilizado de a uno a la vez, entre todos los que estén conectados.

Si conecto varios hubs entre sí (usando cable cruzado), se extiende el bus único, y se mantiene el único dominio de colisión. Cuantos más hubs hay, más retardo hay en la regeneración de la señal y mayor probabilidad de colisión. Cuantas más estaciones estén conectadas al BUS, mayor es la contención y menor el ancho de banda.

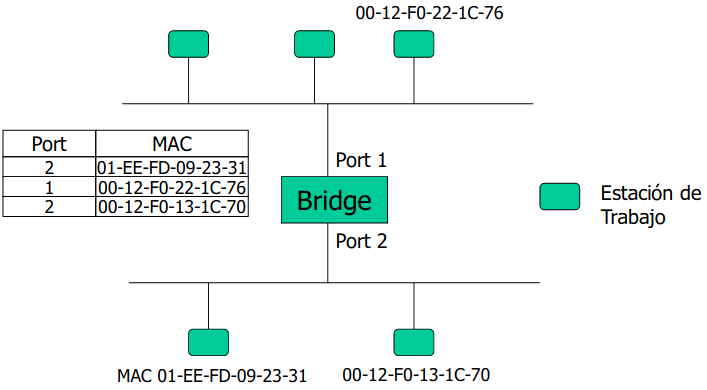
## Bridging

Sirve para mejorar el rendimiento/eficiencia.

Conecta dos o más BUS (redes LAN). Cada BUS tiene conectadas estaciones que compiten por la capacidad (velocidad de transmisión) de ese BUS (el Bridge cuenta como una estación mas). Por lo que se obtienen dos dominios de colisión y uno de Broadcasts.

El Bridge define si la transmisión se transmite del otro lado (según el destinatario), y si no, lo filtra. Los broadcasts siempre se emiten a los dos lados. El bridge es transparente para las estaciones, que no saben que existe el bridge.

Como se ve en la siguiente imagen, si conecta un HUB, a un mismo puerto adjunta varias MACs.



**Transparent Bridge**

* Operan en capa 2 y utilizan las direcciones MAC para encaminar las tramas.
* Aprenden automáticamente la ubicación de los hosts. El bridge va “aprendiendo” quien está de cada lado de sus interfaces (es decir, que estaciones hay en cada BUS), hasta que aprende de qué lado esta cada estación, va pasando las tramas de un lado al otro. La ubicación de cada estación se convierte en dato dinámico y efímero (porque se pueden mover, y el bridge es transparente para la red).   
  Si alguien envía un mensaje con destinatario de una MAC que no está en el directorio de direcciones del bridge, el bridge está obligado a transmitir el mensaje a todos (podría haber aparecido una nueva estación). También los broadcasts van de un lado al otro.
* Las tramas soportan dos procesos: filtering y forwarding (a la trama o se la filtra, o se la pasa, dependiendo si el destino está en el mismo BUS o en el otro).
* Transparente significa que ninguna otra estación conoce de su existencia. Las estaciones se comunican como si estuvieran conectadas a la misma red y no existiera el Bridge.

Si conecto una estación extra directamente a un puerto el bridge por full dúplex, tendría 100% de velocidad de transmisión para él solo, mientras que las de los otros puertos compiten por esa velocidad.

**Translating Bridge**

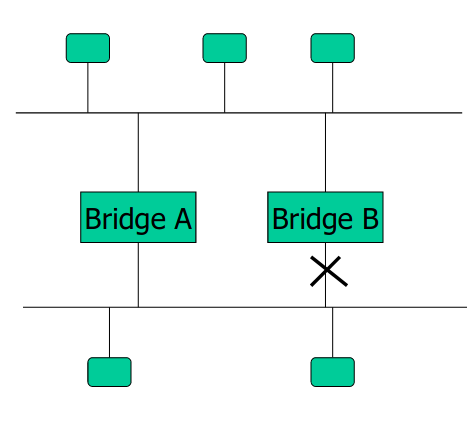
* Realiza además conversión de protocolo y velocidad.

### Bridging Loop y Protocolo Spanning Tree (STP)

Debido a que un Bridge se conecta a electricidad, puedo querer tener dos Bridge en caso de que uno de los dos deje de funcionar. Sin embargo, mientras ambos están en funcionamiento, podría surgir un bridging loop. Esto sucede cuando se **envía una trama de Broadcast, los dos Bridges lo toman y lo empiezan a enviar sin parar** (porque también le llega ese mensaje del otro Bridge y lo vuelve a enviar).

Los bridging loops **se producen por el desconocimiento de la existencia de otros bridges en la red**.

**Qué pasa:**

* Los dos bridges tratan de mandar el mismo mensaje de un bus al otro, y compiten por el espacio de transmisión
* Si uno transmite un mensaje del otro lado del bridge, de A a B, el bridge 2 lo recibe del lado de B e interpreta que A está de ese lado porque tiene la MAC de origen.

#### Protocolo Spanning Tree

Este protocolo **impide que se generen bucles** por **enlaces redundantes.** Genera **BPDU** (Bridging PDU) que **hace que los Bridges se conozcan entre si** (los bridge envían estos BPDU, si no los envía significa que se murió).

* Descubre loops y desactiva vínculos redundantes (rompe bucles).
* En caso que un link se desconecte, se dispara nuevamente el STA, para activar el link desconectado por el STP.
* Todos los bridges (o switches) en una red participan del proceso de elección del root (entre todos indican cual es la raíz del árbol).
* Se envían BPDU cada 2 segundos.
* Todo switch tiene un Bridge ID (8 bytes) compuesto de:
  + Bridge priority (se puede cambiar a mano también).
  + MAC address (6 bytes).
* La prioridad menor se designa root. Si tiene el mismo, desempata por MAC address más baja.
* Cuando cambia el estado de un port, se envían notificaciones de cambio de topología (TCN) y comienza nuevamente el cálculo del árbol.

**Estados del puerto de un bridge**

El puerto de un Switch cuando se conecta un dispositivo, para no generar un bucle, pasa por los siguientes estados, en orden:

* **Blocking.** Inicialmente se encuentra en Blocking ya que desconoce si el dispositivo conectado enviá BPDU o no.
* **Listening:** escuchando el trafico de la red, para saber si lo que acabaron de conectar es un Switch o no.
* **Learning** para aprender MAC addresses. SI un puerto no pasa a Forwarding (bloqueado por spanning tree), igualmente recibe BPDU.
* **Forwarding** que seria activado (no pasa a este estado si se formaría un bucle). Todos los puertos del root están en Forwarding.

## Switch

Dispositivo de red usados para **dividir los segmentos de colisión**, cada puerto es un segmento diferente de colisión. Sin embargo, genera un único dominio de broadcast, debido a que cuando envía un mensaje, todas las demás estaciones lo van a recibir.

Es un dispositivo de **capa 2.**

**Es como un bridge, pero cada estación está conectada directamente a** un puerto del Switch, con conexión **full-duplex** (**divide los dominios de colisión**).

**Se puede combinar switches con bridges. El switch va a recordar que en un único puerto hay varias MAC (por el bridges). En el puerto del bridge puede haber colisiones porque es un puerto compartido.**

Cuando se enviá un unicast las otras estaciones nunca se enteran de la trama debido a que el Switch solo la envía al destinatario.

Los Switch tienen una **tabla** (**CAMP**) que **marcan que MAC address se conecta a cada uno de sus puertos.** Esta tabla tiene un tamaño máximo (cantidad de MAC address que puede recordar, idem el Bridge). Si se supera el límite, agenda la nueva dirección en la última posición (es decir, olvida la entrada más vieja para insertar la nueva).

Un switch con estaciones en cada puerto no tiene dominio de colisión. Pero si en uno de los puertos hay un hub, sí hay dominio de colisiones.

### Modos de operación

**Ingresa una trama** a un Switch por un puerto, el Switch **analiza la trama**, **lee la dirección MAC destino y lo conmuta** (busca en su tabla la dirección MAC y empieza a escribir la trama en el buffer de salida del puerto asociado). En el switch hay un procesador que se ocupa de la conmutación de los mensajes.

Dependiendo su forma de construcción tienen **diferentes modos de operación**:

* **Cut through:** al leer los primeros 6 bytes de la dirección MAC destino, el dispositivo comienza el proceso de conmutación. Lo que **puede suceder es que se produzca una colisión** (con solo 6 bytes puede suceder porque estoy dentro de la ventana de colisión) y se tenga que reenviar la trama. Más rápido y menos eficiente.
* **Store and Forward:** espera a que la trama se reciba completamente, calcular el CRC para verificar que se haya recibido correctamente y recién después leer la dirección MAC destino y conmutarla. Espera todos los bytes de la trama. Evita problemas de colisiones. Más eficiente pero más lento.
* **Fragment free:** Intermedio entre los dos anteriores. Esperamos los primeros 64bytes (ventana de colisión) para asegurarse de que no va a colisionar la trama. Bastante rápido y bastante eficiente.

**Switch non-blocking:** Son aquellos que no bloquean la transmisión de paquetes cuando está transmitiendo otros paquetes. En los switch non-blocking la banda ancha interna puede manejar todas las bandas anchas de los puertos, operando a máxima capacidad.

Cuando hay un switch, no se puede hacer sniffing de toda la red. En un hub sí se puede, porque todo el tráfico está disponible para todos.

### Loop y STP

Sucede lo mismo que en los Bridges.

Cada Switch se aprende las MAC address de los otros Switch.

# Virtual LANs (VLANs) – Capa 3

* Divide dominios de broadcast
* Aísla las redes
* Se crean en un mismo switch con facilidad de VLANs (es decir, las debe soportar).
* Se requiere de un router (dispositivo de nivel 3) para interconectar las VLANs (es decir, perdemos visibilidad en nivel 2). Sin él, no es posible que se comuniquen entre sí los puertos de diferentes VLAN. El router se conecta con ambas redes (con 1 puerto de cada red) y permite el paso de una red a la otra

Existen VLAN donde se definen las VLAN:

* Por puerto. Se configura a que VLAN corresponde cada puerto del switch.
* Por MAC address: Cada VLAN registrara MAC addresses pertenecientes a ella, por lo que independientemente del puerto al que conectemos el dispositivo, si tiene la misma MAC address, sera reconocido por la misma VLAN. Si se conecta un dispositivo con una address no reconocida por ninguna VLAN, será como que no exista.

## VLAN Tagging o Trunking (capa 2) - Protocolo 802.1Q

**Problema que soluciona: VLANs que no entran en un único switch.**

**Sin este protocolo**, para aumentar la cantidad de puertos disponibles para una VLAN, tendría que conectar uno de los puertos de la VLAN (del switch original) a otro switch.

Esto es problemático porque:

* El nuevo switch estaría restringido entero a esa VLAN
* Esto hay que repetirlo para cada VLAN.

### Cómo funciona

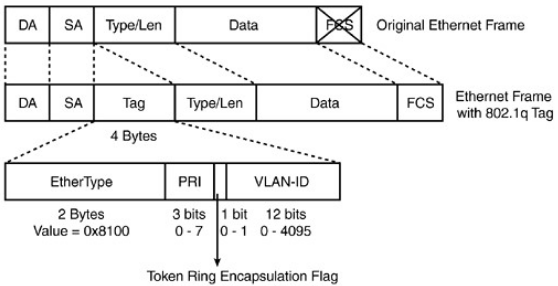
**Para usarlo, ambos switch deben soportar** el protocolo IEEE 802.1Q.

Se conectan ambos switches por medio de un puerto, que es el puerto trunk, y se configura el 802.1Q.

Qué hace el trunk?

* Genera una vinculación por la que pasa el tráfico de todas las VLAN conectadas al switch. Así el nuevo switch puedo tener puertos pertenecientes a cualquiera de las VLAN del switch original.
* El dominio de broadcast del primer switch se extiende y pasa a tener miembros en otro switch.
* Ninguna estación conoce el vínculo entre los switches.
* Para detectar a qué VLAN pertenece un paquete de datos, se inserta una **etiqueta (tag) en las tramas** que circulan por la vinculación (cable que vincula los trunk). La etiqueta la pone el puerto de trunk de uno de los switches, y la quita el puerto del otro switch.   
  Por ej, cuando un dispositivo manda un broadcast, se le agrega una etiqueta que dice “pertenece a la VLAN 1” y se lo pasa por el trunk al switch. El switch 2 al recibir la trama, le quita la etiqueta y se la pasa a los puertos de la VLAN correspondiente.

### La etiqueta



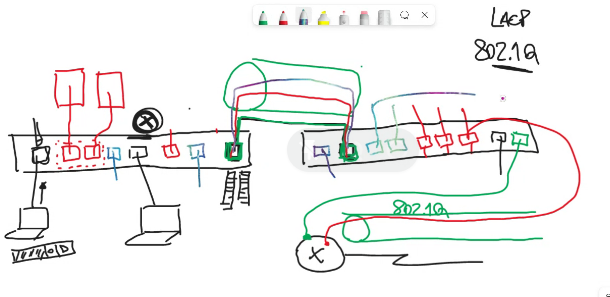
La etiqueta se pone en el puerto trunk de un switch y se saca en el del otro.

* Dónde va? Entre los campos de Source Address y Type/Len.
* Tamaño: 4 bytes
* Partes:
  + Ethertype (2bytes) con valor 0x8100, que indica que se trata de una trama 802.1Q
  + PRI: prioridad (3 bits). Como Ethernet no tenía implementadas prioridades, se aprovechó la creación del tag del protocolo 802.1Q y se implementó el **protocolo 802.1p** con esos tres bits de prioridad (PRI) que me permiten definir 8 niveles de prioridad. E**l switch maneja de manera diferenciada las tramas según su prioridad**. Esto requiere que el switch implemente en cada puerto buffers (colas) diferentes para trafico prioritario y el que no lo es.
  + Token ring encapsulation flag (1 bit): indica si la MAC viene en formato canónico (0) o encapsulado (1) en valor no canónico.
  + VLAN-ID (12 bits). 12 bits me permite un rango de 0 a 4095, pero ambos extremos están reservados, así que se pueden asignar VLANs de 1 a 4094 (4094 valores)

### Conexión a internet de una VLAN

Cuando quiero proveer de internet a alguna VLAN **debo conectar alguno de sus puertos a un router.** En el caso de que varias VLAN de un switch quieran conectarse a internet, debo utilizar un puerto de cada una para la conexión con el router.

Una alternativa a esto es: **defino un puerto del switch como trunking** (por ende, no pertenece a ninguna VLAN) **y conecto dicho puerto al router.** Para ello el **router debe soportar el protocolo 802.1Q.** Así, todas las estaciones de todas las VLANs llegan a internet.



Para ello, **en el router defino interfaces virtuales** (o lógicas) que se comunican con la VLAN correspondiente, así cuando el router responda, lo hace a través de la interfaz virtual correspondiente. Las interfaces físicas en los router son los puertos (conectores físicos), en este caso utilizo un único que lo defino como trunking y lo divido en una interfaz virtual para cada VLAN.

**Mientras yo no defina la interfaz virtual para una VLAN x, esa VLAN no tiene internet.**

### Servicio storage (ESX?)

Es lo mismo que el caso anterior, para evitar el uso de muchos puertos, defino un puerto trunk (prot 802.1Q) en el switch y en el servidor (de almacenamiento). El servidor debe soportar 802.1Q. Las VMs en el servidor de storage entran en diferentes VLANs cada una.

### LACP

El protocolo LACP me permite definir dos o más puertos físicos como un mismo puerto lógico, por lo que no se genera el bucle, ni es bloqueado por el spanning tree.

Esto aumenta la capacidad de intercambio de tramas, y previene el cuello de botella que podría ocurrir sobre la velocidad si todas las conexiones de una VLAN se manejan con un único puerto físico.

(Tanto los switches como el router deben soportar este protocolo)

## Ejemplo

**Mirando una trama de ejemplo**

* **Encabezado Ethernet 2: (no incluye preámbulo y SFD) dirección destino DA, origen SA y tipo (es Ethernet, entonces no es length)**
  + **Tipo: son 0x800 significa que es ipv4**
  + **DA y SA** 
    - **tienen 3 bytes dedicados al fabricante y 3 dedicados al dispositivo.**
    - **Individual address: mira un bit determinado a ver si es 0. Si es 1, sería broadcast (dirección de grupo)**
    - **Globally unique address: mira un bit que si es 0 significa que es una dirección asignada por IEEE y el fabricante, si no, es manipulada.**

## Notas misc

* Formato 802.3 tiene siempre asociado 802.2 asociado
* En la misma red se transmiten msjs tanto eth2 como 802.3