



# Teoría de las Comunicaciones

Edición 65 oficial, pero en realidad desde que se comenzó a dictar (a.k.a Redes) es la edición 74. Este es el último cuatrimestre que se dicta.

**Dr. Claudio Enrique Righetti**

24 septiembre 2025

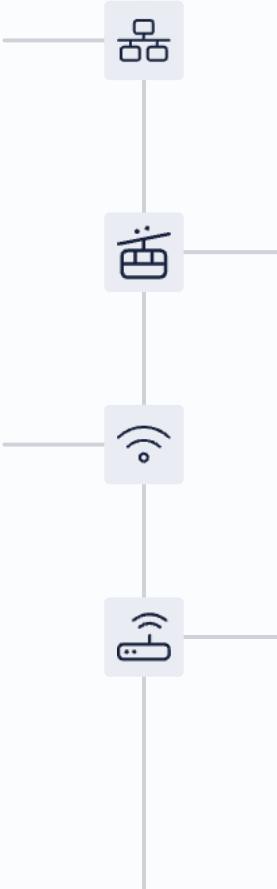
*Segundo Cuatrimestre*

**Departamento de Computación  
Facultad de Ciencias Exactas y Naturales  
Universidad de Buenos Aires  
Argentina**

# Agenda : Módulos (1-4)

## Módulo 1: Fundamentos de Redes

Historia y evolución de redes, modelos OSI y TCP/IP, señales analógicas vs. digitales, teoría de la información y teoremas de Shannon. Práctica: cálculo de entropía, codificación Huffman y análisis de capacidad de canal.



## Módulo 2: Nivel Físico y Enlace

Técnicas de codificación y modulación, medios de transmisión, protocolos de enlace y mecanismos de ventana deslizante. Práctica: implementación de protocolos nivel 2, CRC y algoritmos Stop & Wait.

## Módulo 3: Redes de Acceso y LAN

Tecnologías inalámbricas, Ethernet, switching y VLANs. Práctica: configuración de switches y análisis de tráfico con Wireshark.

## Módulo 4: Nivel de Red

Redes Orientadas a conexión ( Circuitos Virtuales ) y sin Conexión ( Datagramas). **Protocolo IP**, tablas de ruteo, algoritmos Distance Vector y Link State. Práctica: direccionamiento IP, ICMP y configuración de RIP y OSPF.

Al finalizar estos cuatro módulos, se realizará el primer parcial para evaluar la comprensión de los conceptos fundamentales de redes. Estos módulos establecen las bases necesarias para abordar temas más avanzados en la segunda parte del curso.



Nivel de Red



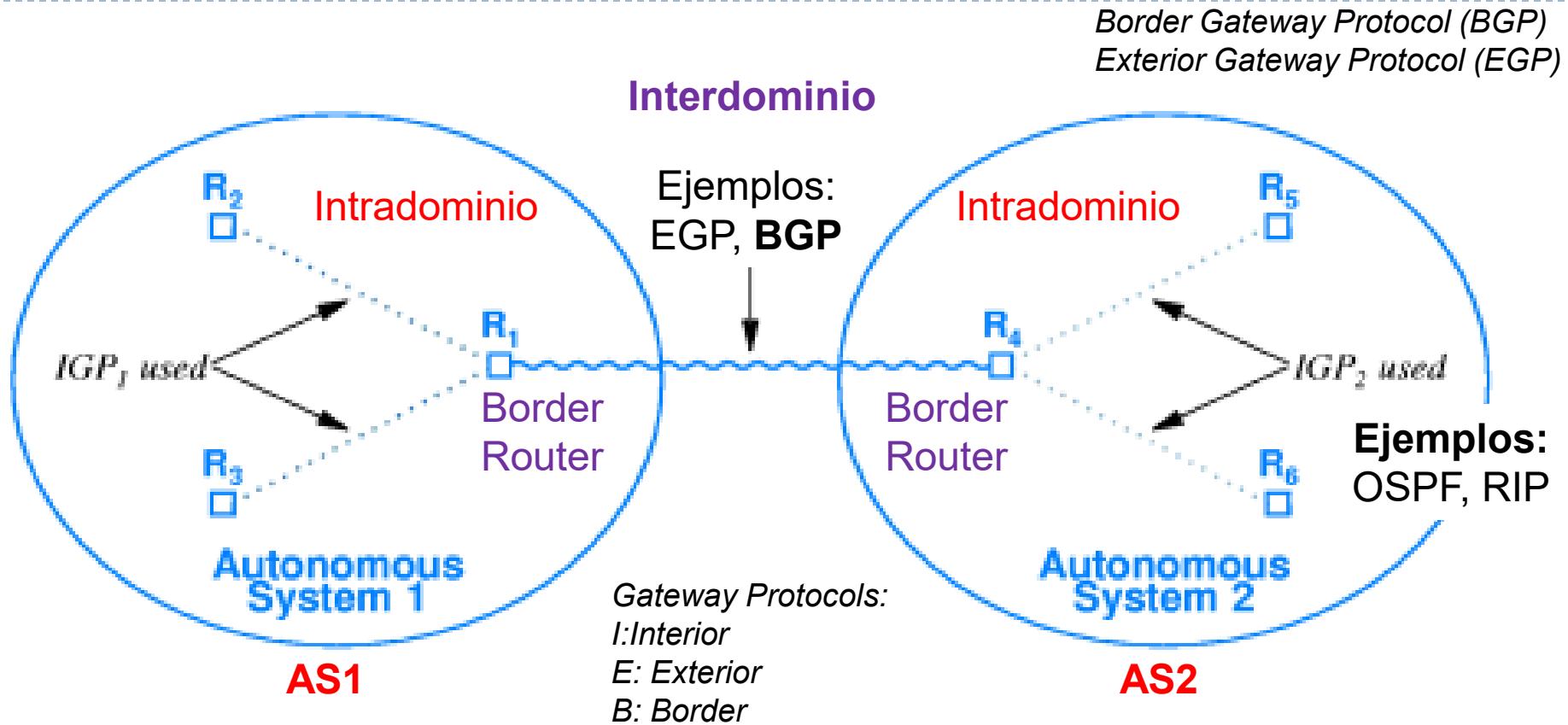
Ruteo

# Agenda

---

- ▶ Introducción: ruteo interno y externo
- ▶ Algoritmos y protocolos
- ▶ Escalabilidad

# Sistemas Autónomos (AS)



Idea clave: **autonomía y heterogeneidad**.

Los detalles de lo que pasa adentro de un AS permanece oculta para otros AS.

Los AS pueden manejar tecnologías heterogéneas.

# Introducción

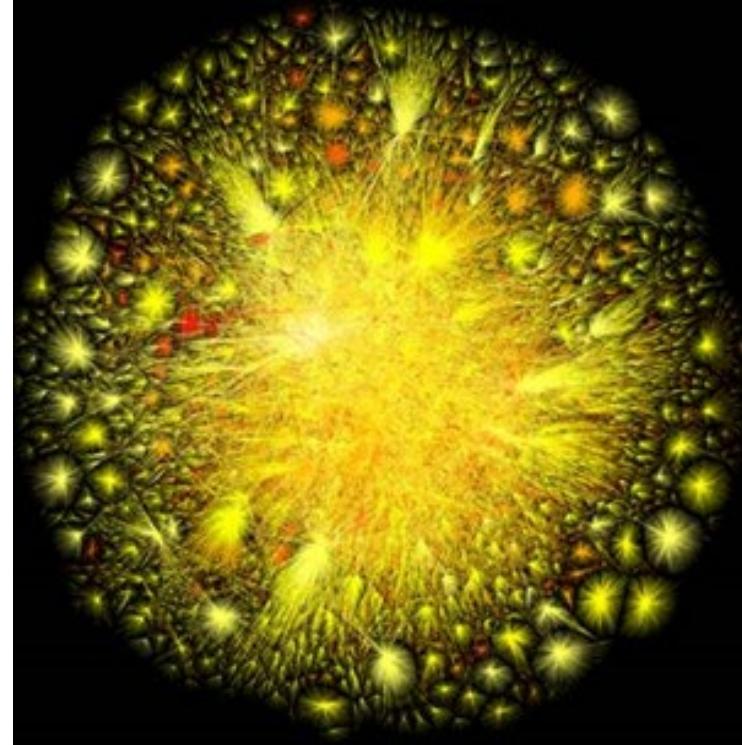
---

- ▶ Los protocolos de ruteo pueden clasificarse en:
  - ▶ Ruteo Interno (**IGP**, *Internal Gateway Protocols*)
  - ▶ Ruteo Externo (**BGP**, *Border Gateway Protocols*)
- ▶ De **Ruteo Interno**: su dominio de ruteo es **dentro de un Sistema Autónomo (AS)**. Son protocolos **intradominio**.
- ▶ De **Ruteo Externo**: aplican a **interdominios**, es decir rutean **entre distintos AS**
- ▶ Podemos decir que **Internet** es una **interconexión de muchos AS**

# Introducción

---

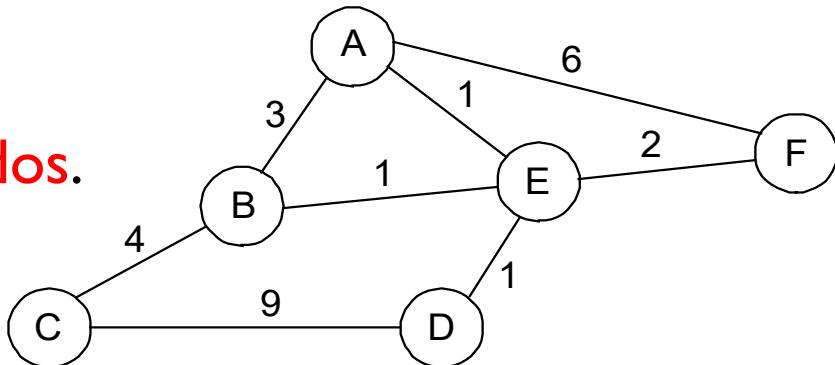
- ▶ Pensamos en un grafo o “red” de sistemas autónomos
- ▶ En Internet existen varios “planos” de redes:
  - ▶ la red (o grafo) de Routers (**IP**)
  - ▶ el grafo de Web Servers (**WWW**)
  - ▶ el grafo de Name Servers (**DNS**)
- ▶ También existen otros grafos como:
  - ▶ P2P (peer to peer)
  - ▶ CDN (content distribution network)
  - ▶ etc.



Fuente de la imagen : [www.opte.org](http://www.opte.org)

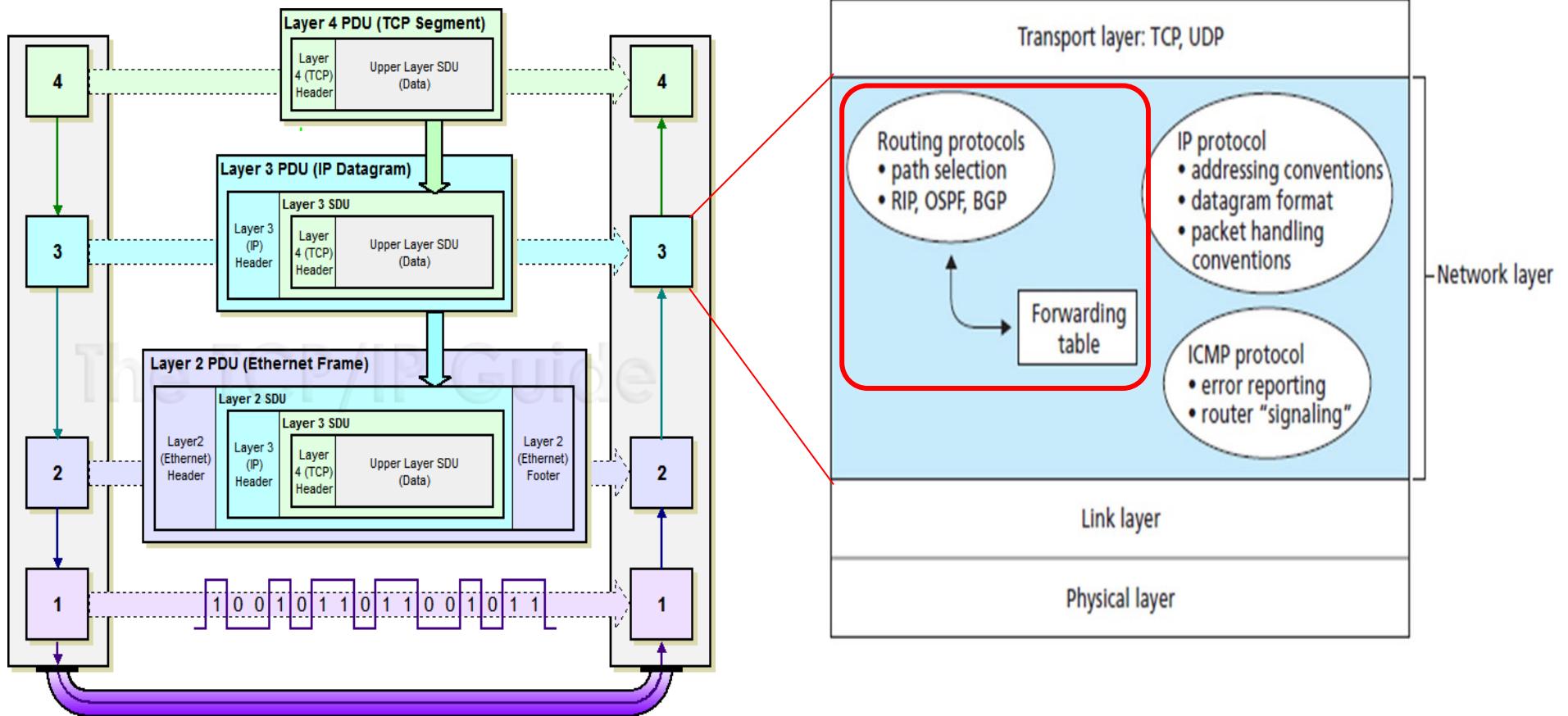
# Introducción

- ▶ Re-envío (**Forwarding**) vs. Ruteo (**Routing**)
  - ▶ **Re-envío:** proceso para seleccionar una **puerta de salida** basado en la *dirección destino* y las *tablas de ruteo*.
  - ▶ **Ruteo:** proceso mediante el cual son *construidas y actualizadas* las *tablas de ruteo*.
- ▶ El Ruteo es un problema de **grafos, optimización y algoritmos distribuidos.**
  - ▶ La red es vista como un conjunto de nodos y arcos pesados.

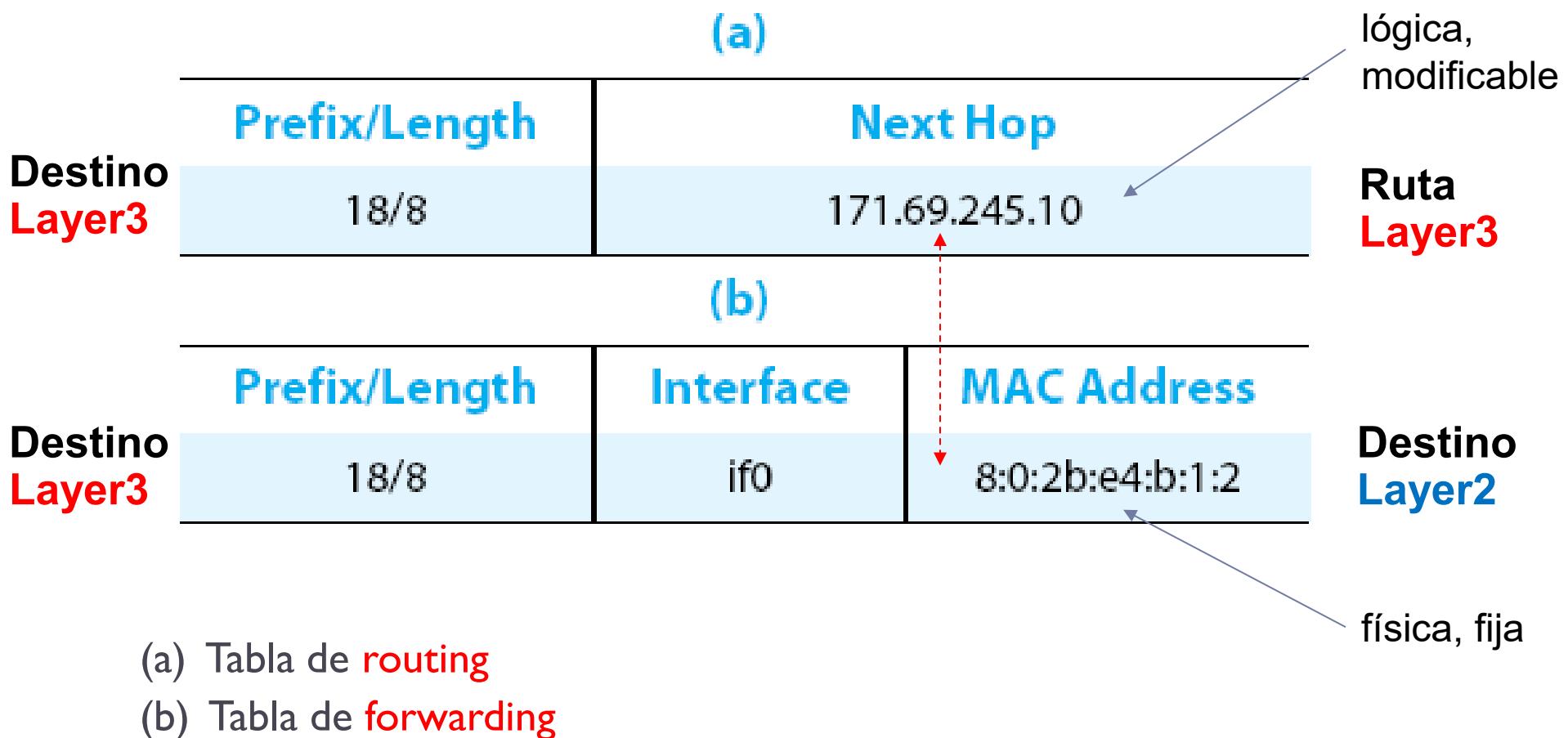


- ▶ **Problema:** Encontrar el **camino de menor costo** entre dos nodos, en **tiempos razonables**, usando **recursos mínimos**.
- ▶ Tipos de ruteo:
  - ▶ **Estático:** Configuración manual, no reactiva.
  - ▶ **Dinámico:** Configuración autónoma y adaptativa (acorde a fallas, carga de nodos, estado de enlaces, etc.)

# Ruteo en el modelo OSI



# Routing





# Protocolos de Ruteo Interno

RIP

OSPF

# Vector de distancia vs. Estado del enlace

Cada Router	DISTANCE-VECTOR	LINK-STATE
¿Qué informa?	• Toda su Tabla de Ruteo	• Solo el Estado de sus Enlaces directos
¿A quién le pasa información?	• Solo a sus vecinos	• A toda la red (inundación)
Algoritmo utilizado	• Bellman-Ford Distribuido	• Dijkstra
Datos utilizados	• Información de los vecinos	• Estado de Enlaces de cada nodo
Estructuras de Datos	• Tabla de Distancias • Tabla de Ruteo	• Tabla de Estado de Enlaces • Tabla de Ruteo
Características	• Ciclos de Ruteo  Gran variedad de Algoritmos: • Merlin-Segall • Jaffe-Moss • Esquema OP • Diffusing Comp • Cheng  • Cálculo Distribuido	• Visión Consistente de la Red • Gran uso de CPU y Memoria  • Algoritmo Básico único  • Cálculo Centralizado
Ejemplo de Protocolos de Internet	Routing Information Protocol - RIP	Open Shortest Path First - OSPF

# Protocolos de Ruteo Interno más utilizados

---

- ▶ **RIP: Routing Information Protocol**
  - ▶ usa algoritmo de **vector de distancia**
  - ▶ basado en cuenta de hops
- ▶ **OSPF: Open Shortest Path First**
  - ▶ usa algoritmo de **estado de enlaces**
  - ▶ soporta balanceo de carga y QoS (Quality of Service)
  - ▶ soporta autenticación

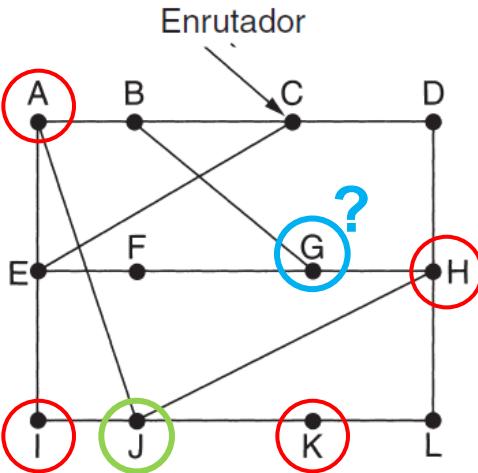
# Distance Vector

# Vector de Distancia

---

- ▶ Cada nodo mantiene una tabla hacia **todos** los nodos
  - ▶ (**Destination**, **Cost**, **NextHop**)
- ▶ Intercambia actualizaciones **solo con sus vecinos directamente conectados** del siguiente modo:
  - ▶ Periódicamente (en el orden de varios segundos)
  - ▶ Cuando su tabla cambia (se habla de una actualización gatillada)
- ▶ Cada actualización es una lista de pares:
  - ▶ (**Destination**, **Cost**)
- ▶ Se **modifica** la tabla local si se recibe una **mejor ruta**
  - ▶ Costo menor
  - ▶ Llegó desde el host próximo (“next-hop”)
- ▶ Se refrescan rutas existentes; se borran si hay time out

# Vector de Distancia (ejemplo I)



J calcula su nueva ruta a G

(a)

Nodo J

A	0	24	20	21	8	A
B	12	36	31	28	20	A
C	25	18	19	36	28	I
D	40	27	8	24	20	H
E	14	7	30	22	17	I
F	23	20	19	40	30	I
G	18	31	6	31	18	H
H	17	20	0	19	12	H
I	21	0	14	22	10	I
J	9	11	7	10	0	-
K	24	22	22	0	6	K
L	29	33	9	9	15	K

Retardo JA es de 8

Retardo JI es de 10

Retardo JH es de 12

Retardo JK es de 6

Vectores recibidos de los cuatro vecinos de J

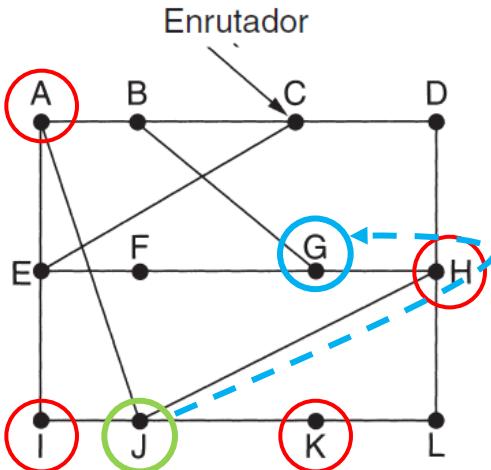
Nuevo retardo estimado desde K

Línea next hop

Nueva tabla de enrutamiento para J

(a) Subred. (b) Entrada de A, I, H, K y la nueva tabla de enrutamiento de J.

# Vector de Distancia (ejemplo I)



**J calcula su nueva ruta a G**

Retardo **J a G** a través de:  
A:24 (8+18), I:41 (31+10), **H:18** (6+12), K:37 (31+6)

(a)

Nodo A: 0, 12, 25, 40, 14, 23, 18, 17, 21, 9, 24, 29  
 Nodo I: 24, 36, 18, 27, 7, 20, 31, 20, 0, 11, 22, 33  
 Nodo H: 20, 31, 19, 8, 30, 19, 6, 0, 14, 7, 22, 9  
 Nodo K: 21, 28, 36, 24, 22, 40, 31, 19, 22, 10, 0, 9  
 Nodo J: 8, 20, 28, 20, 17, 30, 18, 12, 10, 0, 6, 15  
 Retardo JA es de 8, Retardo JI es de 10, Retardo JH es de 12, Retardo JK es de 6

Vectores recibidos de los cuatro vecinos de J

Nuevo retardo estimado desde J → Línea next hop

Desde J Hacia G conviene vía: H Costo: 6+12

Nueva tabla de enrutamiento para J

(b)

(a) Subred. (b) Entrada de A, I, H, K y la nueva tabla de enrutamiento de J.

# La tabla de ruteo (ejemplo I)

---

- ▶ Las primeras cuatro columnas de la parte (b) vectores de retardo recibidos de los vecinos del router J .
- ▶ A indica tener un retardo de 12 mseg a B, un retardo de 25 mseg a C, un retardo de 40 mseg a D, etc.
- ▶ J ha medido o estimado el retardo a sus vecinos A, I, H y K en 8, 10, 12 y 6 mseg, respectivamente
- ▶ J calcula su nueva ruta al router G. Sabe que puede llegar a A en 8 mseg, y A indica ser capaz de llegar a G en 18 mseg, por lo que J sabe que puede contar con un retardo de 26 mseg a G si reenvía a través de A los paquetes destinados a G.
- ▶ J calcula el retardo a G a través de I, H y K en 41 ( $31 + 10$ ), 18 ( $6 + 12$ ) y 37 ( $31 + 6$ ) mseg, respectivamente.
- ▶ El mejor de estos valores es el 18, por lo que escribe una entrada en su tabla de enrutamiento indicando que el retardo a G es de 18 mseg, y que la ruta que se utilizará es vía H.

# Vector de Distancia (ejemplo II)

- ▶ Cada nodo arma un vector que contiene la distancia (costos) a todos los nodos y lo envía a sus vecinos

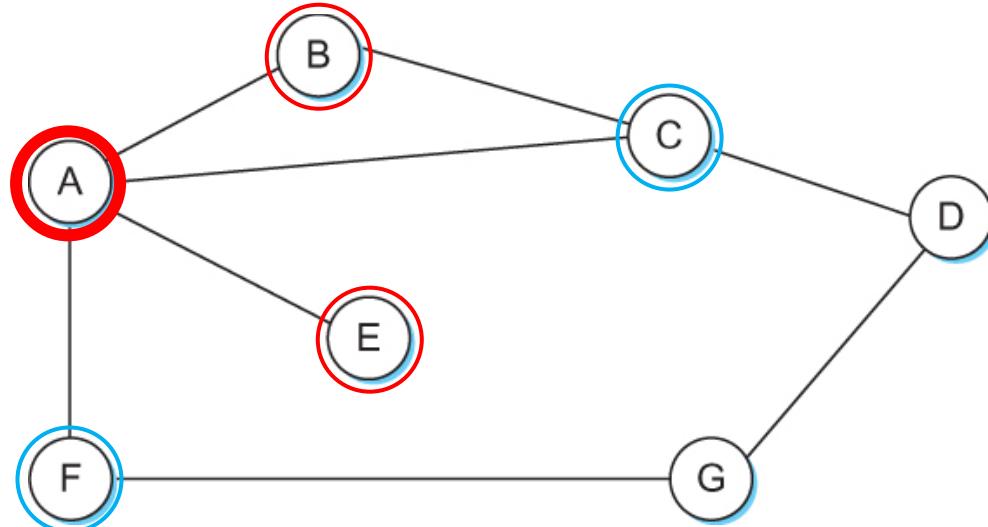


Tabla de ruteo del **Nodo A** ?

Destino	Costo	NextHop
B	1	B
C	1	C
D	2	C
E	1	E
F	1	F
G	2	F

- ▶ Suponemos acá que cada salto tiene costo igual a 1 al comenzar

# Vector de Distancia (II)

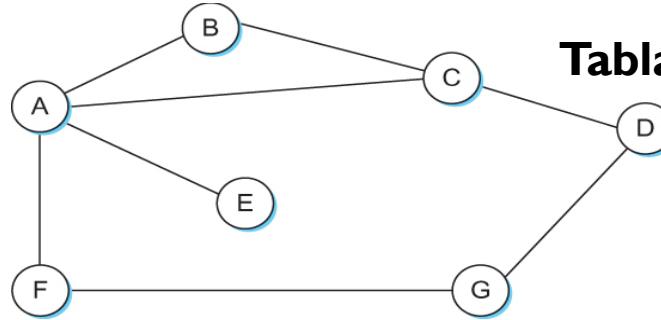
Tabla de ruteo del **Nodo A** ?

Information Stored at Node	Distance to Reach Node						
	A	B	C	D	E	F	G
A	0	1	1	$\infty$	1	1	$\infty$
B	1	0	1	$\infty$	$\infty$	$\infty$	$\infty$
C	1	1	0	1	$\infty$	$\infty$	$\infty$
D	$\infty$	$\infty$	1	0	$\infty$	$\infty$	1
E	1	$\infty$	$\infty$	$\infty$	0	$\infty$	$\infty$
F	1	$\infty$	$\infty$	$\infty$	$\infty$	0	1
G	$\infty$	$\infty$	$\infty$	1	$\infty$	1	0

Estado inicial de distancia almacenado en cada nodo.

Una visión global, ningún nodo conoce “todo el estado”.

# Vector de Distancia (II)



**Tabla de ruteo del Nodo A ?**

Destination	Cost	NextHop
B	1	B
C	1	C
D	$\infty$	—
E	1	E
F	1	F
G	$\infty$	—

# Vector de Distancia (II)

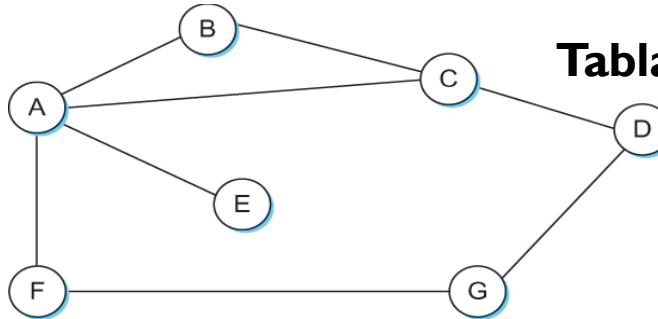


Tabla de ruteo del Nodo A ?

Destination	Cost	NextHop
B	1	B
C	1	C
D	2	C
E	1	E
F	1	F
G	2	F

Tabla de ruteo final de A

# Vector de Distancia (II)

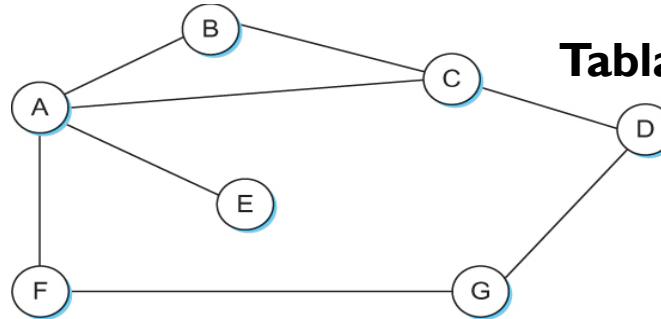


Tabla de ruteo del **Nodo A** ?

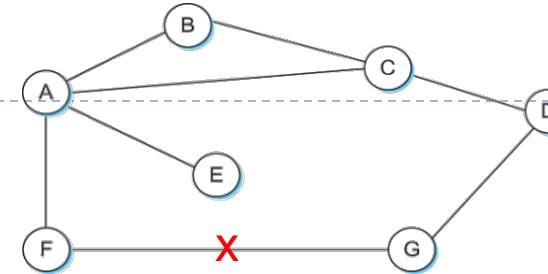
Information Stored at Node	Distance to Reach Node						
	A	B	C	D	E	F	G
A	0	1	1	2	1	1	2
B	1	0	1	2	2	2	3
C	1	1	0	1	2	2	2
D	2	2	1	0	3	2	1
E	1	2	2	3	0	2	3
F	1	2	2	2	2	0	1
G	2	3	2	1	3	1	0

Tabla final luego de una **convergencia global**

# Ciclos de Actualización

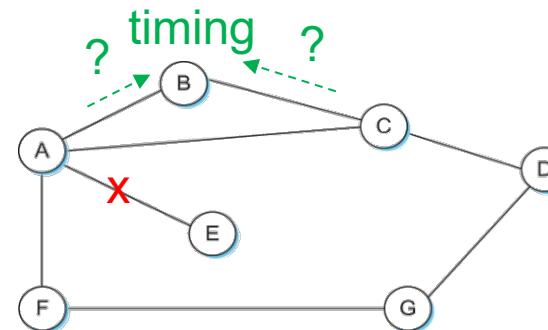
## ► Ejemplo 1 (caso feliz: estable)

- ▶ F detecta que su enlace a G falla.
- ▶ F fija distancia  $\infty$  a G y envía una actualización a A
- ▶ A fija distancia  $\infty$  a G porque A usa F para llegar a G
- ▶ A recibe actualización periódica de C con camino de 2 hops a G
- ▶ A fija distancia a G con 3 y envía actualización a F
- ▶ F decide que él puede llegar a G en 4 hops vía A

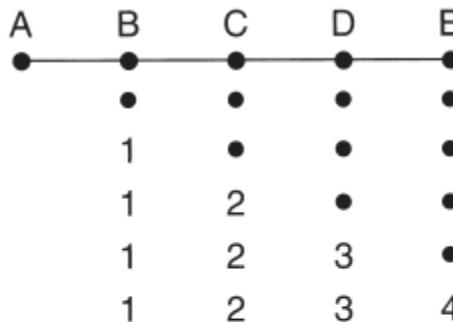


## ► Ejemplo 2 (caso no feliz: inestable, conteo a infinito)

- ▶ Enlace de A a E falla.
- ▶ A comunica a B y C distancia  $\infty$  a E (timing ?)
- ▶ B y C comunican distancia 2 a E (timing ?)
- ▶ B decide, gracias a C, que puede llegar a E en 3 hops; comunica esto a A
- ▶ A decide que puede llegar a E en 4 hops; comunica esto a C
- ▶ C decide que puede llegar a E en 5 hops; comunica esto a B ...



# Problema de conteo a Infinito



Inicialmente

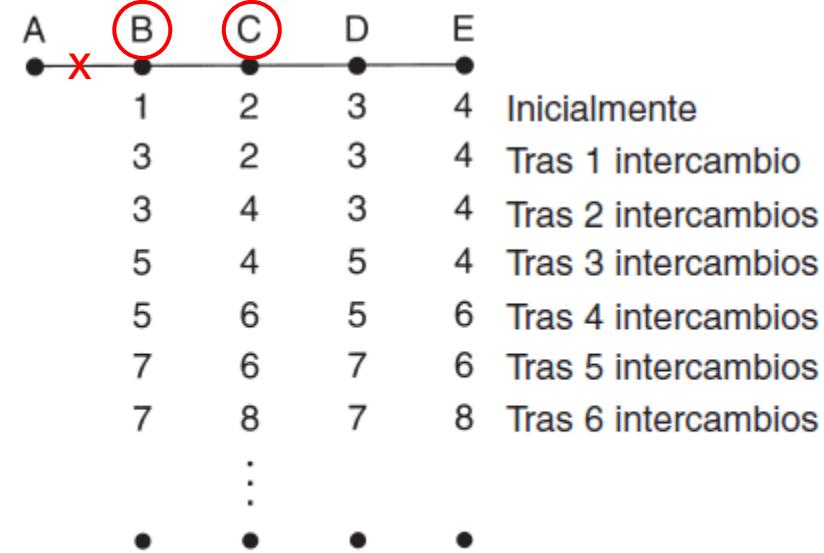
Tras 1 intercambio

Tras 2 intercambios

Tras 3 intercambios

Tras 4 intercambios

(a)



Inicialmente

Tras 1 intercambio

Tras 2 intercambios

Tras 3 intercambios

Tras 4 intercambios

Tras 5 intercambios

Tras 6 intercambios

⋮

(b)

- (a) El nodo A está seteado como infinito, luego se activa
- (b) El nodo A está activo, luego cae

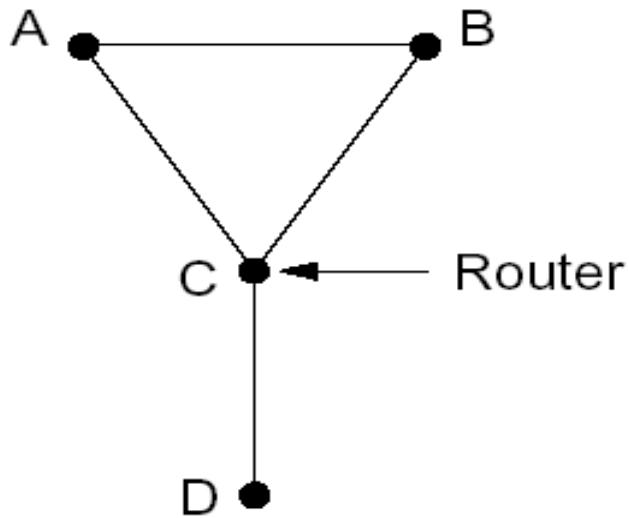
# Heurísticas para romper los ciclos

---

- ▶ Fijar **16 como infinito**, es decir cuando el costo llega a 16 se asume que no hay ruta al nodo
- ▶ Partir el horizonte (**Split horizon**): omite enviar información de distancia que fue aprendida por el nodo al cual se le envía el vector
- ▶ Partir el horizonte con reverso venenoso (**Split horizon with poison reverse**): si notifica entradas aprendidas desde el nodo al cual se envía el vector, pero a esos destinos se les pone costo infinito
- ▶ Las últimas dos técnicas sólo solucionan ciclos que involucran dos nodos
- ▶ La convergencia de este protocolo no es buena, se mejora con ruteo usando el **Estado de los Enlaces (Link-State)**

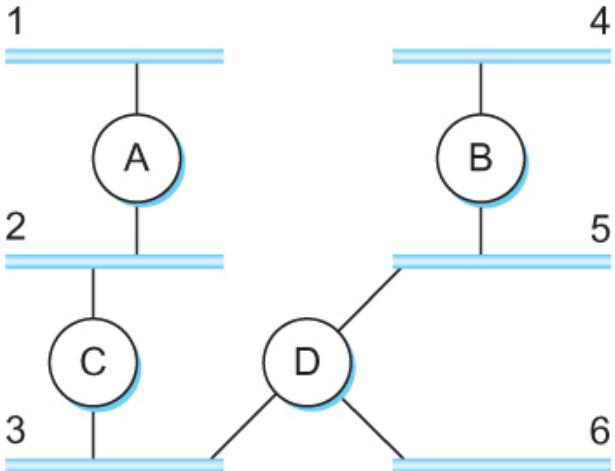
## Ejemplo con Poison Reverse:

---



- (a) Todos los costos son uno
- (b) Falla el enlace C – D
- (c) A puede llegar a D por B
- (d) ...

# Routing Information Protocol (RIP)



Red con RIP

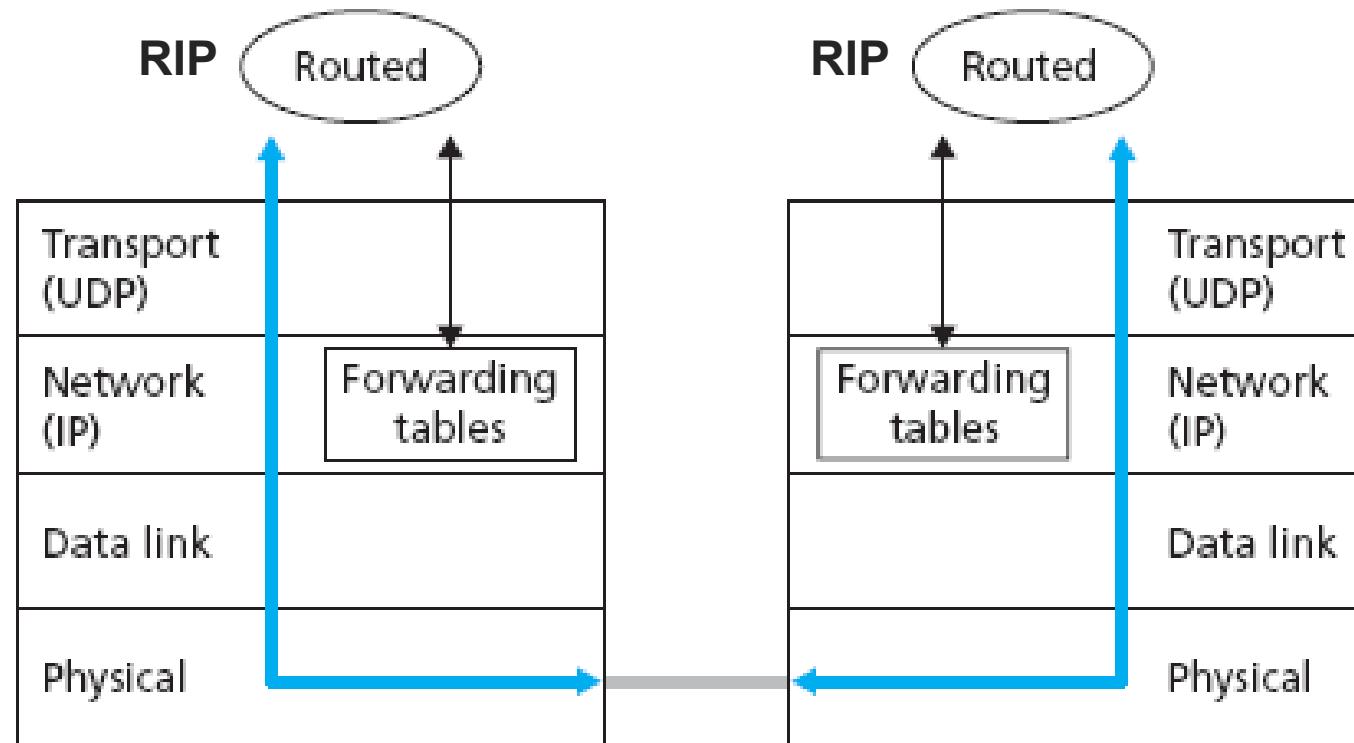
0	8	16	31
Command	Version	Must be zero	
Family of net 1	Route Tags		
Address prefix of net 1			
Mask of net 1			
Distance to net 1			
Family of net 2	Route Tags		
Address prefix of net 2			
Mask of net 2			
Distance to net 2			

Formato del paquete RIPv2

Las distancias válidas son de 1 a 15.

Distancia 16 representa infinito: limitación de RIP a redes de tamaño pequeño.

# RIP implementado como un “daemon”



**RIP** usa el protocolo de transporte **UDP** y tiene reservado el número de Puerto 520.

# RIP implementado como un “daemon”

IP

```
Internet Protocol version 4, Src: 10.1.12.2 (10.1.12.2), Dst: 255.255.255.255 (255.255.255.255)
  Version: 4
  Header length: 20 bytes
  Differentiated Services Field: 0xc0 (DSCP 0x30: Class Selector 6; ECN: 0x00: Not-ECT (Not ECN-Capable Transport))
    1100 00.. = Differentiated services codepoint: Class selector 6 (0x30)
    .... ..00 = Explicit Congestion Notification: Not-ECT (Not ECN-Capable Transport) (0x00)
  Total length: 52
  Identification: 0x0000 (0)
  Flags: 0x00
    0... .... = Reserved bit: Not set
    .0... .... = Don't fragment: Not set
    ..0. .... = More fragments: Not set
  Fragment offset: 0
  Time to Live: 2
    [Expert info (note/Sequence): "Time To Live" only 2]
      [Message: "Time To Live" only 2]
      [Severity level: note]
      [Group: Sequence]
    Protocol: UDP (17)
  Header checksum: 0xalf7 [correct]
    [Good: True]
    [Bad: False]
  Source: 10.1.12.2 (10.1.12.2)
  Destination: 255.255.255.255 (255.255.255.255)
    [Source GeoIP: Unknown]
    [Destination GeoIP: Unknown]
```

UDP

```
User Datagram Protocol, Src Port: router (520), Dst Port: router (520)
  Source port: router (520)
  Destination port: router (520)
  Length: 32
  Checksum: 0xe48a [validation disabled]
    [Good Checksum: False]
    [Bad Checksum: False]
```

RIP

```
Routing Information Protocol
  Command: Request (1)
  Version: RIPv1 (1)
  Address not specified, Metric: 16
    Address Family: unspecified (0)
    Metric: 16
```



Link State

# Algoritmo de Estado del Enlace

---

- ▶ “Link State Broadcast”: Todos los nodos tienen **la misma** información
- ▶ Topología de red y costos: **conocidos por todos los nodos**
- ▶ Cómputo del camino mínimo: Dijkstra (**forward search**)

# Algoritmo de Estado del Enlace

---

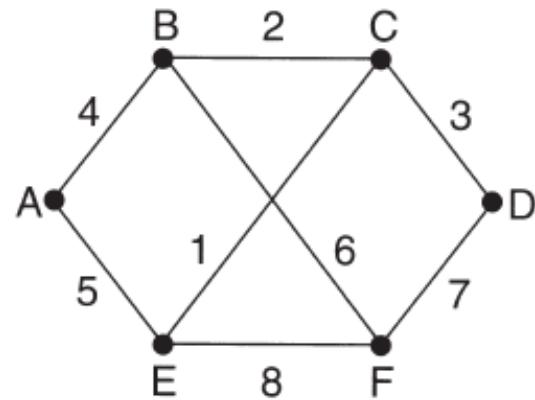
- ▶ Descubrir a sus vecinos y conocer sus direcciones de red
- ▶ Medir costo para cada uno de sus vecinos
- ▶ Construir un paquete que indique todo lo que acaba de aprender
- ▶ Enviar este paquete a todos los demás routers
- ▶ Calcular la ruta más corta a todos los nodos

# Estado del Enlace

---

- ▶ Estrategia
  - ▶ Enviar a todos los nodos (no sólo los vecinos) información solo sobre sus **enlaces directamente conectados** (no la tabla completa).
  - ▶ Se “inunda” la red con esta información.
- ▶ Paquete del estado del enlace (**Link State Packet, LSP**)
  - ▶ **ID** del router que creó el LSP
  - ▶ **Costo** del enlace a cada vecino directamente conectado
  - ▶ Número de secuencia (**SEQNO**)
  - ▶ Time-To-Live (**TTL**) para este paquete

# Paquetes de Estado del Enlace



A	B	C	D	E	F
Sec.	Sec.	Sec.	Sec.	Sec.	Sec.
Edad	Edad	Edad	Edad	Edad	Edad
B   4	A   4	B   2	C   3	A   5	B   6
E   5	C   2	D   3	F   7	C   1	D   7
	F   6	E   1		F   8	E   8

# Estado del Enlace (cont.)

---

- ▶ Inundación **Confiable**
- ▶ Cada Router:
  - ▶ **Almacena** el LSP más reciente de cada nodo
    - ▶ Decrementa TTL de cada LSP almacenado
    - ▶ Descarta cuando TTL=0
  - ▶ **LSP ACK**
  - ▶ **Reenvía** LSP a todos **excepto a quien se lo envió a él**
  - ▶ **Genera** un nuevo LSP periódicamente
    - ▶ Incrementa SEQNO
  - ▶ **Inicia** SEQNO en 0 cuando se reinicia

# Shortest Path Routing

---

- ▶ En la práctica el cálculo de la tabla de ruta se hace conforme los **LSP** van llegando
  - ▶ Se utiliza una implementación del algoritmo de Dijkstra llamada “forward search algorithm”
- ▶ Se manejan dos listas de entradas: las **tentativas** y las **confirmadas**.
- ▶ Cada una de esas listas tiene entradas:  
**(Destination, Cost, NextHop)**

# Uso del algoritmo de Dijkstra en ruteo

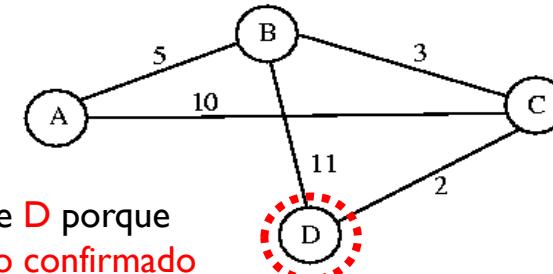
Construcción de la Tabla de Ruteo para D

Paso	Confirmado	Tentativo
1	(D,0,-)	
2	(D,0,-)	(B,11,B) (C,2,C)
3	(D,0,-) (C,2,C)	<u>(B,11,B)</u>
4	(D,0,-) (C,2,C)	<u>(B,5,C)</u> (A,12,C)
5	(D,0,-) (C,2,C) (B,5,C)	<u>(A,12,C)</u>
6	(D,0,-) (C,2,C) (B,5,C)	<u>(A,10,C)</u>
7	(D,0,-) (C,2,C) (B,5,C) (A,10,C)	

(Destination, Cost, NextHop)

## Comentario

- Miramos el LSP de D porque es **nuevo miembro confirmado**
- El LSP de D nos dice cómo llegamos a B y C y sus costos.
- El miembro de menor costo entre los tentativos es C. Se examina el LSP del nodo recién confirmado.
- Con C, se *actualizan los costos*. **Ahora se llega a B vía C** y se incorpora A vía C.
- Se mueve el de menor costo de tentativos a confirmados, que es B.
- Con B, se *actualizan los costos*. **Ahora se llega a A vía C**.
- Después de esto se mueve el menor costo de los tentativos a los confirmados. A es el único y terminamos.



# Cálculo de Ruta

---

- ▶ Algoritmo de Dijkstra para el camino más corto entre nodos
- ▶ Sean
  - ▶  $N$  denota el conjunto de nodos del grafo
  - ▶  $l(i, j)$  denota costo no negativo (peso) para el arco  $(i, j)$   
Si no hay arco, el costo es infinito.
  - ▶  $s$  denota este nodo (self)
  - ▶  $M$  denota el conjunto de nodos incorporados hasta ahora
  - ▶  $C(n)$  denota el costo del camino de  $s$  hasta  $n$

```
 $M = \{s\}$ 
for each  $n$  in  $N - \{s\}$ 
   $C(n) = l(s, n)$ 
while ( $N \neq M$ )
   $M = M$  unión  $\{w\}$  tal que
     $C(w)$  es el mínimo
    para todo  $w$  en  $(N - M)$ 
  for each  $n$  in  $(N - M)$ 
     $C(n) = \text{MIN}(C(n), C(w) + l(w, n))$ 
```

# Comparación entre algoritmos: Vector de Distancia y Estado de Enlaces.

---

- ▶ En el algoritmo **Vector de Distancia** cada nodo transmite a sus vecinos todo lo que sabe respecto de toda la red (distancia a todos los nodos).
  - ▶ Puede ser inestable. **Conteo a infinito**.
- ▶ En el algoritmo **Estado de Enlaces** cada nodo transmite a toda la red lo que sabe de sus vecinos (solo el estado de sus vecinos)
  - ▶ Es **estable**, no genera demasiado tráfico y responde rápido a cambios de topología.
    - ▶ Su problema es la cantidad de información que debe ser almacenada en los nodos (un LSP por nodo)

# Métricas

---

- ▶ Métrica ARPANET "original"
  - ▶ Mide el número de paquetes encolados hacia cada enlace
  - ▶ No toma en cuenta latencia ni ancho de banda
- ▶ Métrica ARPANET "nueva"
  - ▶ Marca cada paquete entrante con su tiempo de llegada (**arrival time, AT**)
  - ▶ Graba tiempo de salida (**departure time, DT**)
  - ▶ Cuando llega el ACK del enlace de datos, calcula:  
**Delay = (DT - AT) + Transmit + Latency**  
Transmit y Latency son parámetros estáticos del enlace.
  - ▶ Si hay timeout: reset DT a tiempo de salida para retransmisión
  - ▶ **Costo del enlace = retardo promediado sobre algún período**
- ▶ Mejora fina (ver "revised ARPANET routing metric [1]")
  - ▶ Reducir el "rango dinámico" para el costo
    - ▶ Moderación de la diferencia entre el peor y el mejor costo.
  - ▶ En lugar del retardo se emplea la utilización del enlace

# OSPF (Open Shortest Path First)

---

- ▶ Open=“Abierto”: código disponible públicamente
- ▶ Utiliza un algoritmo de Estado de Enlaces:
  - ▶ Distribución de paquetes del estado de enlaces.
  - ▶ Mapa de la topología en cada nodo.
  - ▶ Cómputo de la ruta usando el algoritmo de Dijkstra.
- ▶ El anuncio OSPF lleva una entrada por cada router vecino.
  
- ▶ Anuncios distribuidos a **todos los nodos** pero **dentro de un AS** (mediante inundación)
  - ▶ Transportados por mensajes en OSPF directamente sobre IP
- ▶ **OSPF jerárquico:** para grandes dominios (escala)

# OSPF

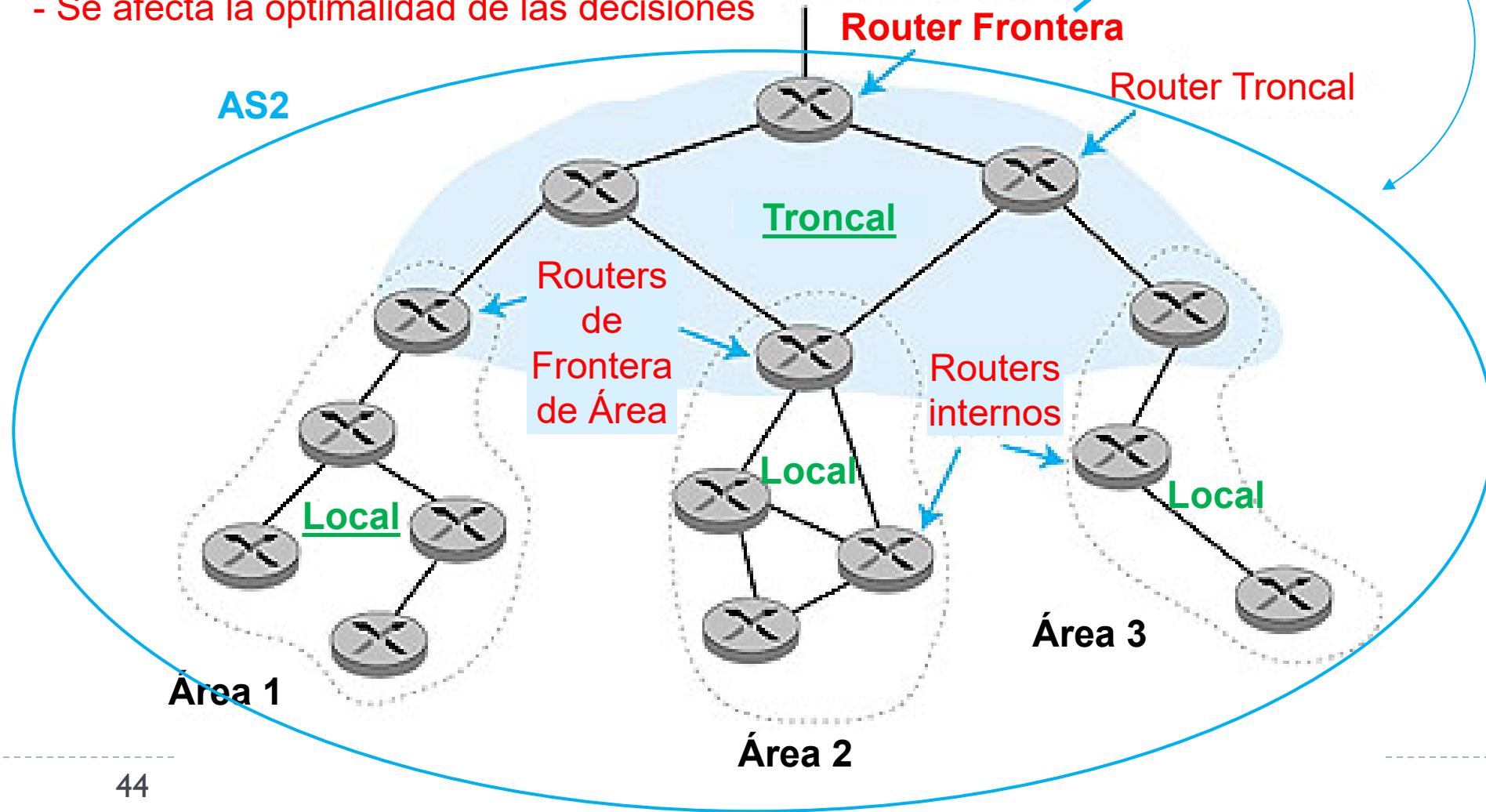
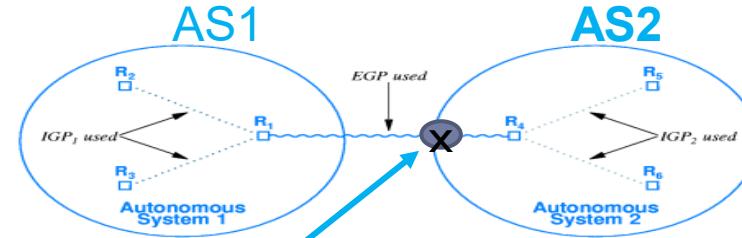
---

- ▶ **Seguridad:** todos los mensajes OSPF están autenticados (para prevenir intrusiones malignas).
  - ▶ Múltiples caminos permitidos con el mismo costo (c.f. sólo un camino en RIP).
- 
- ▶ Para cada enlace, varias métricas de coste para distintos ToS (Type of Service)
    - ▶ Por ejemplo, costo del enlace satélite “rebajado” para un mayor rendimiento
    - ▶ Costo alto para tiempo real
  - ▶ Soporte integrado uni- y multi-difusión (Unicast, Multicast):
    - ▶ La multidifusión OSPF (MOSPF) usa la misma base de datos topológica que OSPF.

# OSPF jerárquico

Intercambio de **escalabilidad** por optimalidad

- Jerarquizar implica ocultar información
- Se afecta la optimalidad de las decisiones



# OSPF jerárquico

---

- ▶ **Dos niveles** de jerarquía: área Local, área Troncal.
  - ▶ Anuncios de estado de enlace: sólo en el área.
  - ▶ Cada nodo detalla la topología del área
    - ▶ sólo conoce la dirección (el camino más corto) a las **redes de otras Áreas**.
- ▶ **Routers Frontera**: conectan con **otros AS**.
- ▶ **Routers Troncales**: ejecutan ruteo de **OSPF** limitados al área troncal.
- ▶ **Routers de Frontera de Área**: “resumen” de las distancias a las redes del mismo área, anuncian a otros routers de Frontera de Área.

# Tipos de Mensaje en OSPF



Tipo de mensaje	Descripción
Hello	Descubre quiénes son los vecinos
Link state <b>update</b>	Proporciona los costos del emisor a sus vecinos
Link state ack	Confirma la recepción de la actualización del estado del enlace
Database description	Anuncia qué actualizaciones tiene el emisor
Link state <b>request</b>	Solicita información del socio

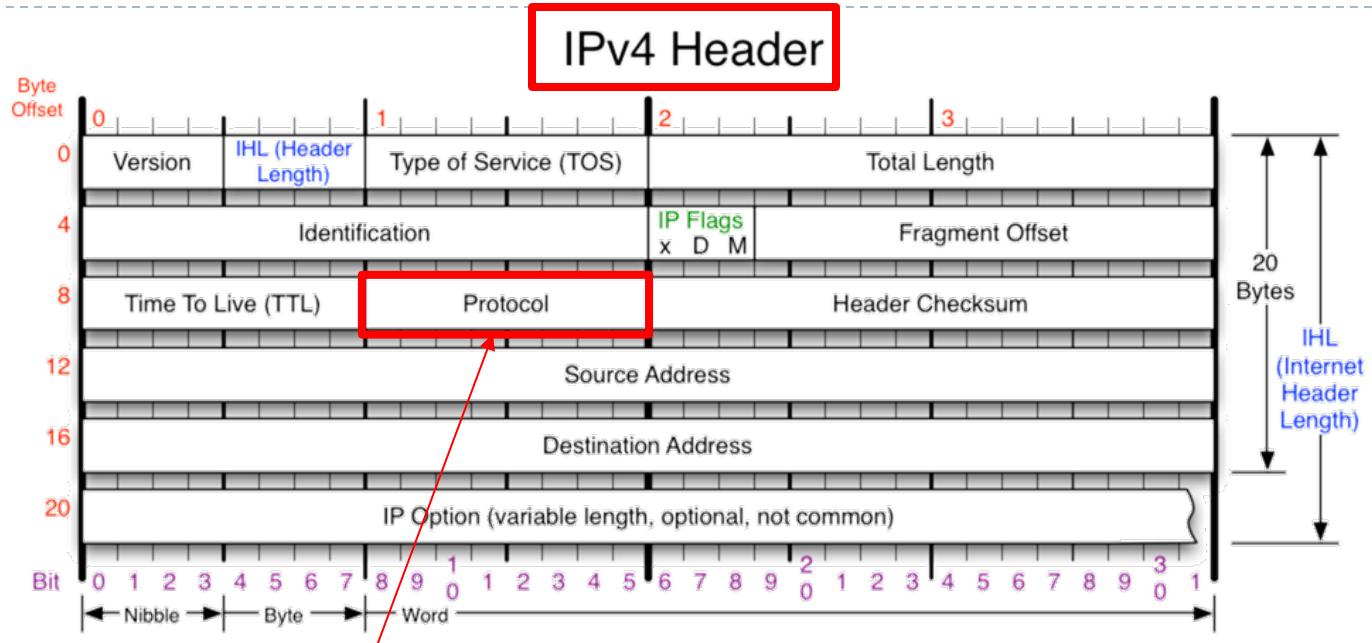
Confiabilidad: Se confirman con ACKs los *Database Description*, *LS request* y *LS update*

# Tipos de Mensaje en OSPF

---

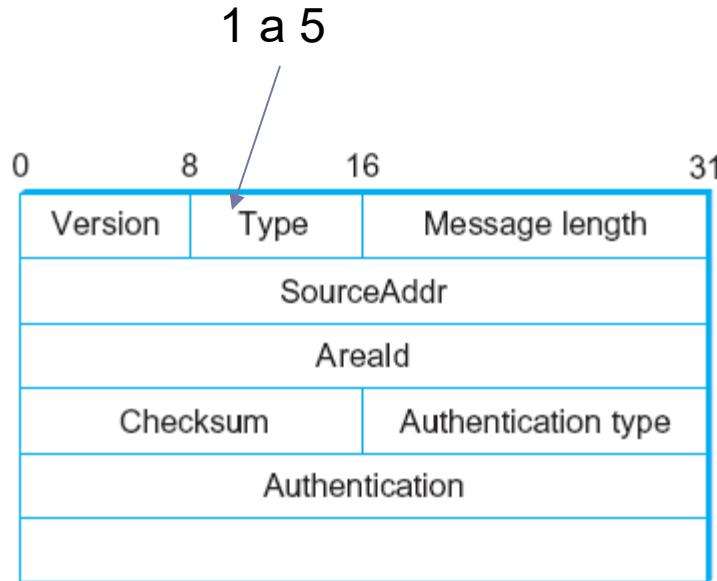
- ▶ Cada router inunda periódicamente con mensajes LINK STATE UPDATE a cada uno de routers adyacentes. Este mensaje da su estado y proporciona los costos usados en la base de datos topológica.
- ▶ **Para hacerlos confiables, se confirma la recepción de los mensajes de inundación.** Cada mensaje tiene un número de secuencia para que un enrutador pueda ver si un LINK STATE UPDATE entrante es más viejo o más nuevo que el que tiene actualmente. Los routers también envían estos mensajes cuando una línea su costo cambia.
- ▶ Los mensajes DATABASE DESCRIPTION dan los números de secuencia de todas las entradas de estado del enlace poseídas por el emisor actualmente. Comparando sus propios valores con los del emisor, el receptor puede determinar quién tiene los valores más recientes. Estos mensajes se usan cuando se activa una línea.
- ▶ Cualquier socio puede pedir información del estado del enlace al otro usando los mensajes LINK STATE REQUEST.
- ▶ El resultado de este algoritmo es que cada par de enrutadores adyacentes hace una verificación para ver quién tiene los datos más recientes, y de esta manera se difunde la nueva información a lo largo del área.

# Open Shortest Path First (OSPF)



Version	Protocol	Fragment Offset	IP Flags
Version of IP Protocol. 4 and 6 are valid. This diagram represents version 4 structure only.	IP Protocol ID. Including (but not limited to): 1 ICMP    17 UDP    57 SKIP 2 IGMP    47 GRE    88 EIGRP 6 TCP    50 ESP    89 OSPF 9 IGRP    51 AH    115 L2TP	Fragment offset from start of IP datagram. Measured in 8 byte (2 words, 64 bits) increments. If IP datagram is fragmented, fragment size (Total Length) must be a multiple of 8 bytes.	x 0x80 reserved (evil bit) D 0x40 Do Not Fragment M 0x20 More Fragments follow
Header Length	Total Length	Header Checksum	RFC 791
Number of 32-bit words in TCP header, minimum value of 5. Multiply by 4 to get byte count.	Total length of IP datagram, or IP fragment if fragmented. Measured in Bytes.	Checksum of entire IP header	Please refer to RFC 791 for the complete Internet Protocol (IP) Specification.

# Open Shortest Path First (OSPF)



Formato header **OSPF**

LS Age	Options	Type=1
Link-state ID		
Advertising router		
LS sequence number		
LS checksum	Length	
0	Flags	0
		Number of links
Link ID		
Link data		
Link type	Num_TOS	Metric
Optional TOS information		
More links		

Formato de “**OSPF Link State Advertisement**” (LSA)

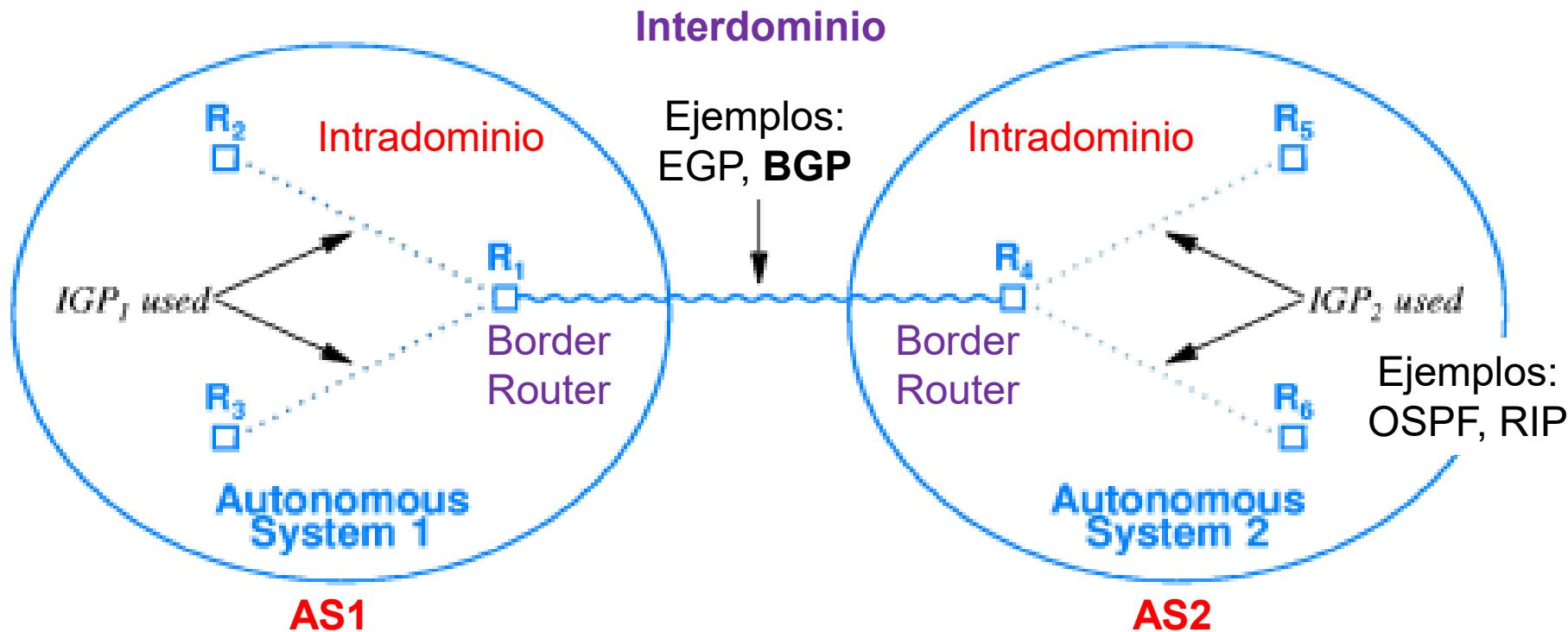


**BGP**



Ruteo Interdominio

# Sistemas Autónomos (AS) - Repaso



Idea clave: **autonomía**.

Los detalles de lo que pasa adentro de un AS permanece oculta para otros AS.

# EGP: Exterior Gateway Protocol

---

## ▶ Generalidades

- ▶ Diseñado para una Internet estructurada como árbol
- ▶ Se preocupa de *alcanzar* los nodos, *no optimiza* rutas

## ▶ Mensajes del Protocolo

### ▶ Adquisición de vecinos:

- ▶ Un router requiere que otro sea su par
- ▶ Pares intercambian información de alcance

### ▶ Alcance de vecinos:

- ▶ Un router periódicamente prueba si el otro es aún alcanzable
- ▶ Intercambia mensajes HELLO/ACK

### ▶ Actualización de rutas:

- ▶ Nodos pares intercambian periódicamente sus tablas de ruteo  
**(vector de distancia)**

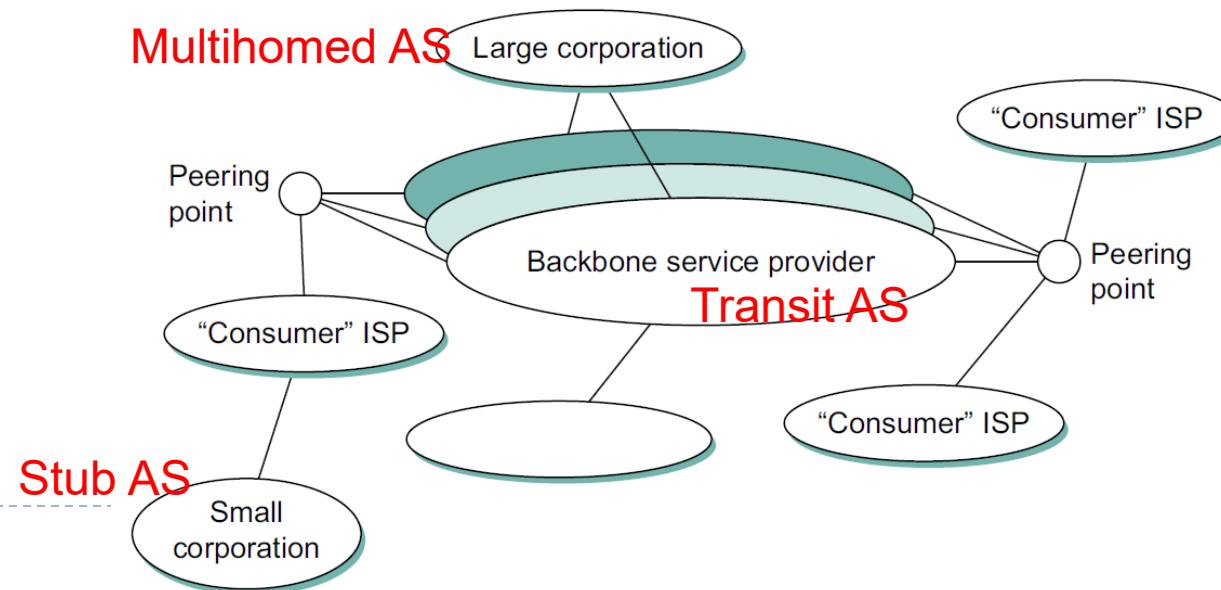
# BGP: Exterior Gateway Protocol

---

- ▶ Generalidades
  - ▶ Internet organizado como Sistemas Autónomos
    - ▶ Cada uno bajo el control de **una única entidad administrativa**
    - ▶ Otra forma más (!) de agregar jerárquicamente información de enrutamiento **para alcanzar gran escala (fundamental en Internet)**
  - ▶ El problema del ruteo se divide en dos
    - ▶ **Adentro o Afuera** de un Sistema Autónomo
    - ▶ BGP se ocupa de ruteo Afuera de los sistemas autónomos
      - ▶ **Ruteo interdominio**
      - ▶ **Cada AS puede ejecutar su propio ruteo intradominio**
  - ▶ BGP resuelve el problema de evitar que diferentes AS necesiten compartir información de alcanzabilidad entre ellos, conociendo:
    - ▶ Cuales **rangos de direcciones IP** pueden alcanzarse en cada AS
    - ▶ Por cual **ruta** se puede llegar **de un AS a otro AS**

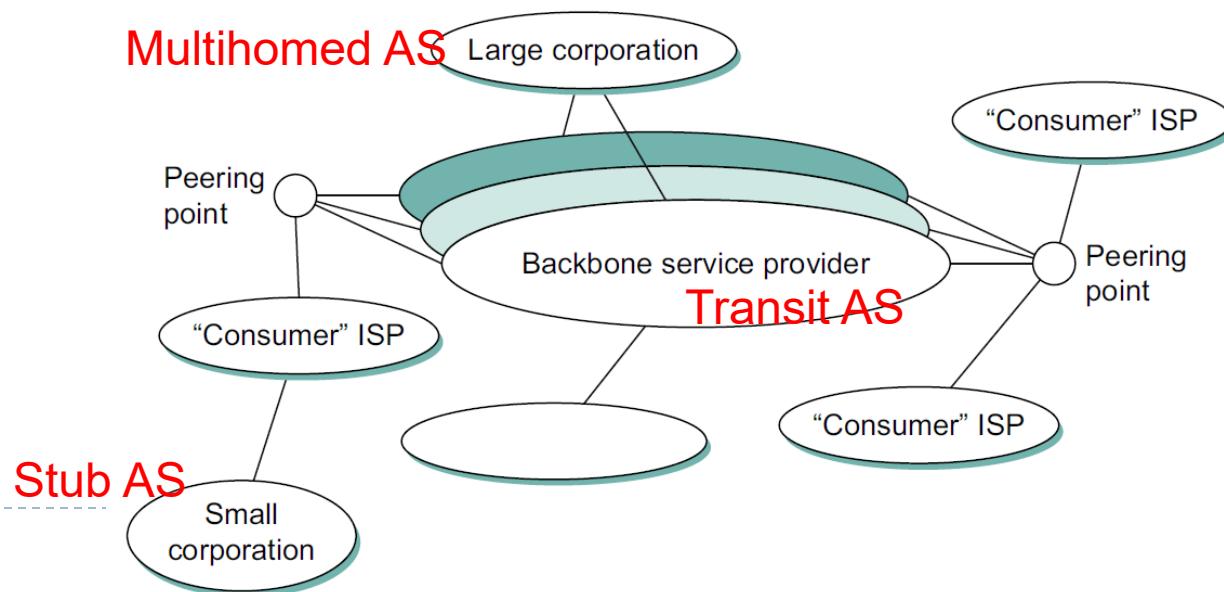
# BGP: Border Gateway Protocol

- ▶ BGP no requiere jerarquía de árbol
- ▶ Tipos de AS:
  - ▶ **Stub AS**: tiene una única conexión a otro AS
    - ▶ transporta sólo tráfico local
  - ▶ **Multihomed AS**: tiene conexiones a más de un AS
    - ▶ no transporta tráfico en transito
  - ▶ **Transit AS**: tiene conexiones a más de un AS
    - ▶ transporta ambos, tráfico local y en tránsito



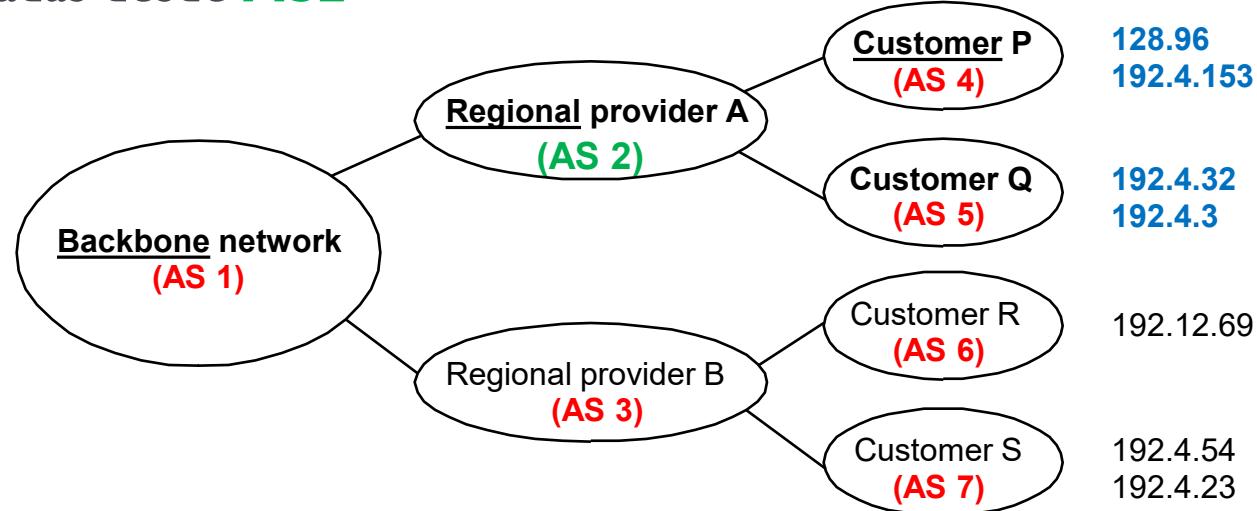
# BGP: Border Gateway Protocol

- ▶ Cada AS tiene:
  - ▶ Uno o más Routers de Borde
  - ▶ Al menos un “Portavoz“ BGP que publica:
    - ▶ Redes locales
    - ▶ Otras redes alcanzables (sólo el Transit AS)
    - ▶ Información de rutas (paths AS)



# Ejemplo BGP

- ▶ El Portavoz para AS2 publica alcanzabilidad a P y Q
  - ▶ Redes 128.96, 192.4.153, 192.4.32, y 192.4.3, pueden ser directamente alcanzadas desde AS2



- ▶ El Portavoz para Backbone publica que:
  - ▶ Las redes 128.96, 192.4.153, 192.4.32 y 192.4.3 pueden ser alcanzadas a lo largo de la ruta (AS1,AS2)
- ▶ Un Portavoz puede cancelar una ruta publicada antes

# BGP y los Network Access Points (NAP)

**CABASE: Cámara Argentina de Internet**

- ▶ “Los **NAPs** (**Network Access Points** por sus siglas en inglés) o **también conocidos como IXs (Internet eXchanges)** son componentes fundamentales de la Red Internet.
- ▶ A través de un **NAP**, se produce el intercambio de tráfico entre las redes de diversas entidades (operadores, proveedores de acceso, organismos de gobierno, entidades académicas, etc.)
- ▶ Los **NAPs** son **puntos neurálgicos de la Red** se han construido en todo el mundo bajo distintos esquemas institucionales, topológicos y operacionales.
- ▶ La mayoría de **NAPs** persigue idénticos objetivos: eficientizar el ruteo de Internet, mejorando la calidad de servicio y minimizar los costos de interconexión.
- ▶ Todos los **NAPs CABASE** siguen el modelo cooperativo. Todos los miembros de los NAPs CABASE son socios de la **Cámara Argentina de Internet** que tienen como objetivo mejorar la calidad en las comunicaciones y reducir costos.” [I]

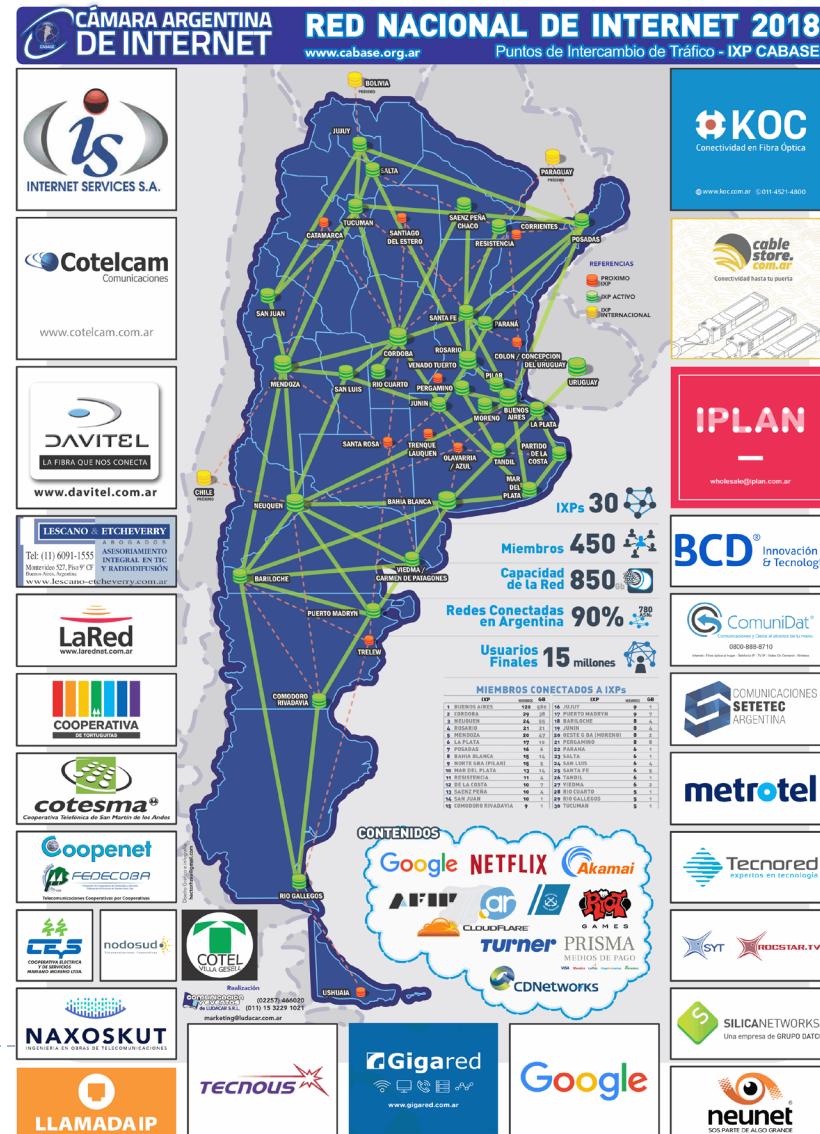
# BGP y los Network Access Points (NAP)

<http://www.cabase.org.ar/wp-content/uploads/2015/07/Poster-Cabase15-06-2015.pdf>



# BGP y los Network Access Points (NAP)

Evolución a 2018



# Bibliografía Básica

---

- ▶ Computer Networks, (Fifth Edition):A Systems Approach. Larry L. Peterson, Bruce S. Davie 2011
  - ▶ Routing – págs. 240-466
- ▶ Computer Networks (5th Edition) Andrew S.Tanenbaum, David J.Wetherall 2010
  - ▶ Routing Algorithms - págs. 392-404

# Algunas Referencias

---

- ▶ **RIPV1**
  - ▶ RFC-1058 - Charles Hedrick, June 1988
- ▶ **RIPV2**
  - ▶ RFC 1388 (Jan. 1993), RFC 1723 (Nov. 1994), RFC 2453 (Nov. 1998),,,  
Gary Malkin
- ▶ **OSPFV1**
  - ▶ RFC 1131 J. Moy, Oct. 1989
- ▶ **OSPFV2**
  - ▶ RFC 1247 (July 1991), RFC 1583 (March 1994), RFC 2178 (July 1997),  
RFC 2328 (April 1998), J. Moy



**Nivel Físico**



**Fundamentos**

# Marco de Referencia



## 3º Teorema de Comunicación Confiable Codificación de Canal (error bajo control)

$$\frac{H(\mathcal{S})}{T_s} \leq \frac{C}{T_c}$$

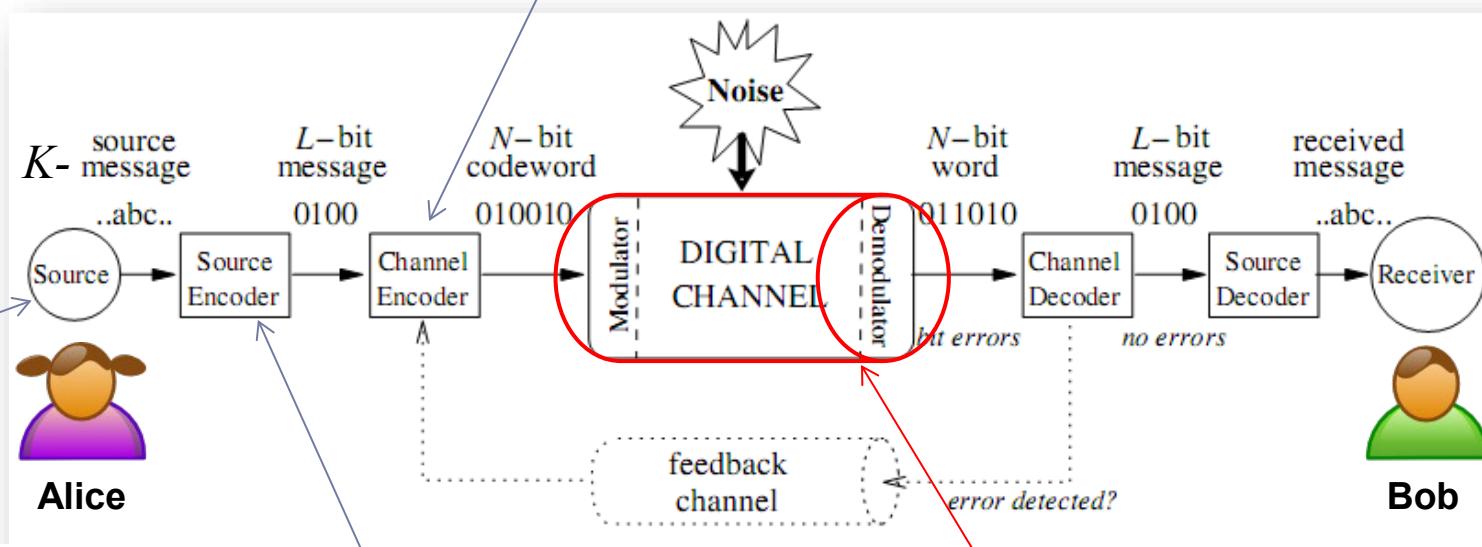
### 1º Conceptos Básicos

$$I = \log_2\left(\frac{1}{p_k}\right)$$

$$H = \sum_{k=0}^{K-1} p_k \cdot \log_2\left(\frac{1}{p_k}\right)$$

Información y Entropía  
 $0 \leq H(S) \leq \log_2 K$

### 2º Teorema de Codificación de Fuente Comunicación Eficiente



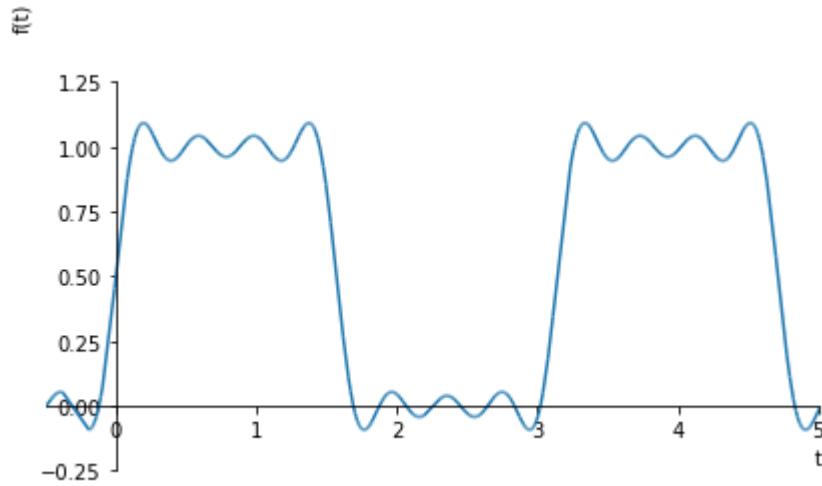
$$\bar{L} \geq H(\mathcal{S})$$

$$C = B \cdot \log_2\left(1 + \frac{S}{N}\right)$$

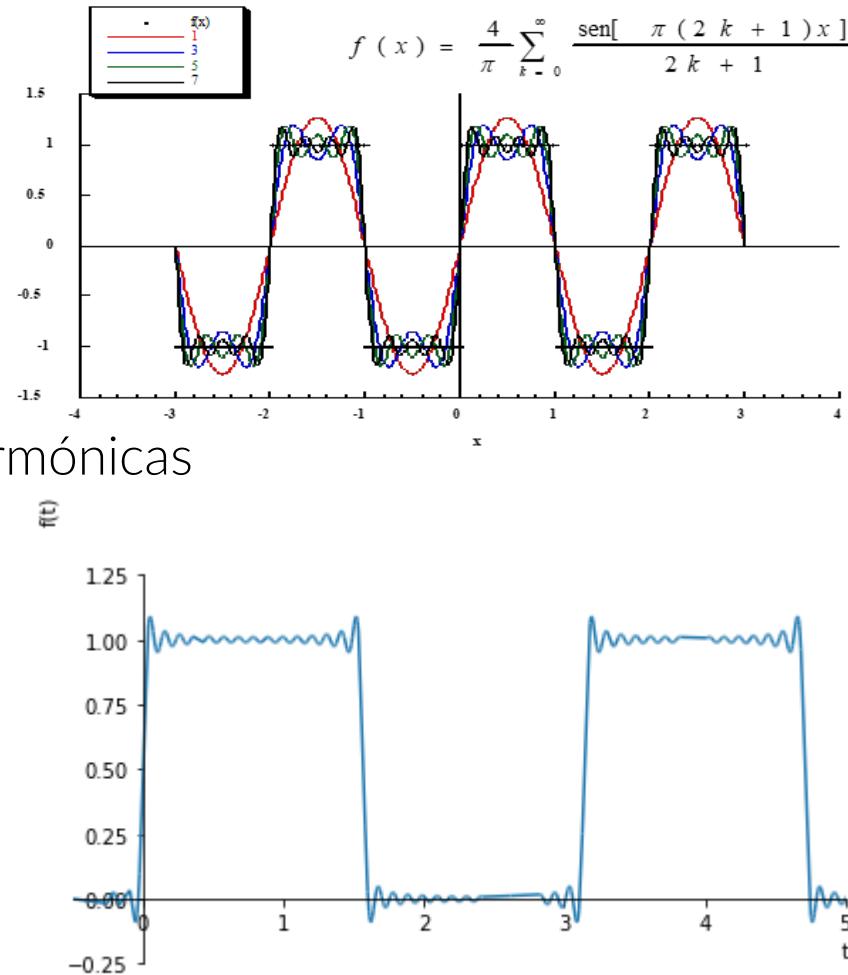
### 4º Teorema de Capacidad de Información Compromiso Ancho de banda vs. Relación Señal a Ruido

# Serie de Fourier

- 9 armónicas



29 armónicas

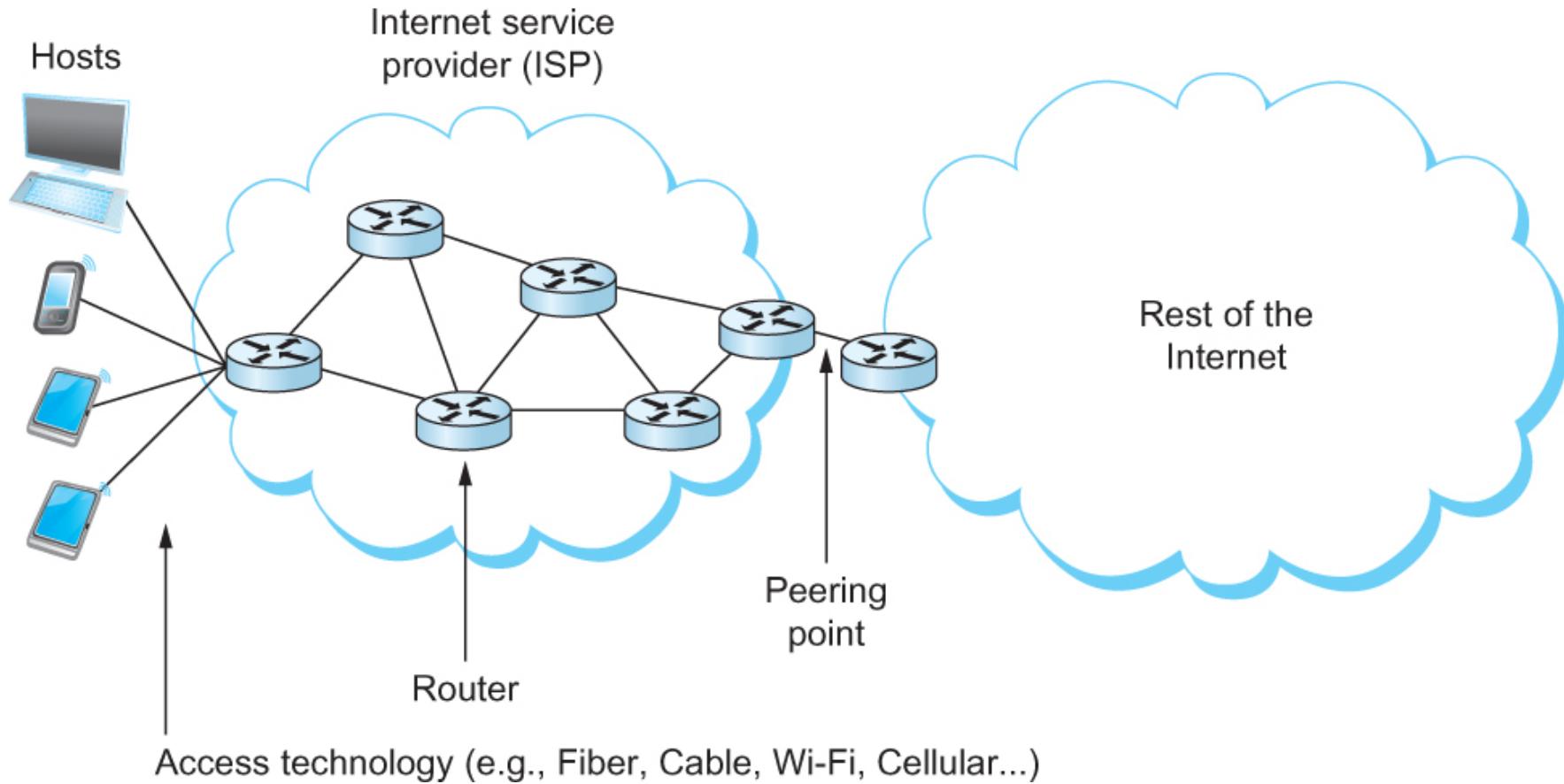


# 1 Parte Agenda

---

- Medios de Transmisión: guiados y no guiados
  - El dominio de la frecuencia
  - La red telefónica
  - Conversión analógico - digital
  - Modulación (“Modulación” Digital / Portadora Analógica)
  - Codificación (“Codificación” Digital / Portadora Digital)
  - Capacidad de Volumen de un Canal
-

# Para “conectarnos”: Medios de transmisión





## Los medios de transmisión



Guiados y no guiados

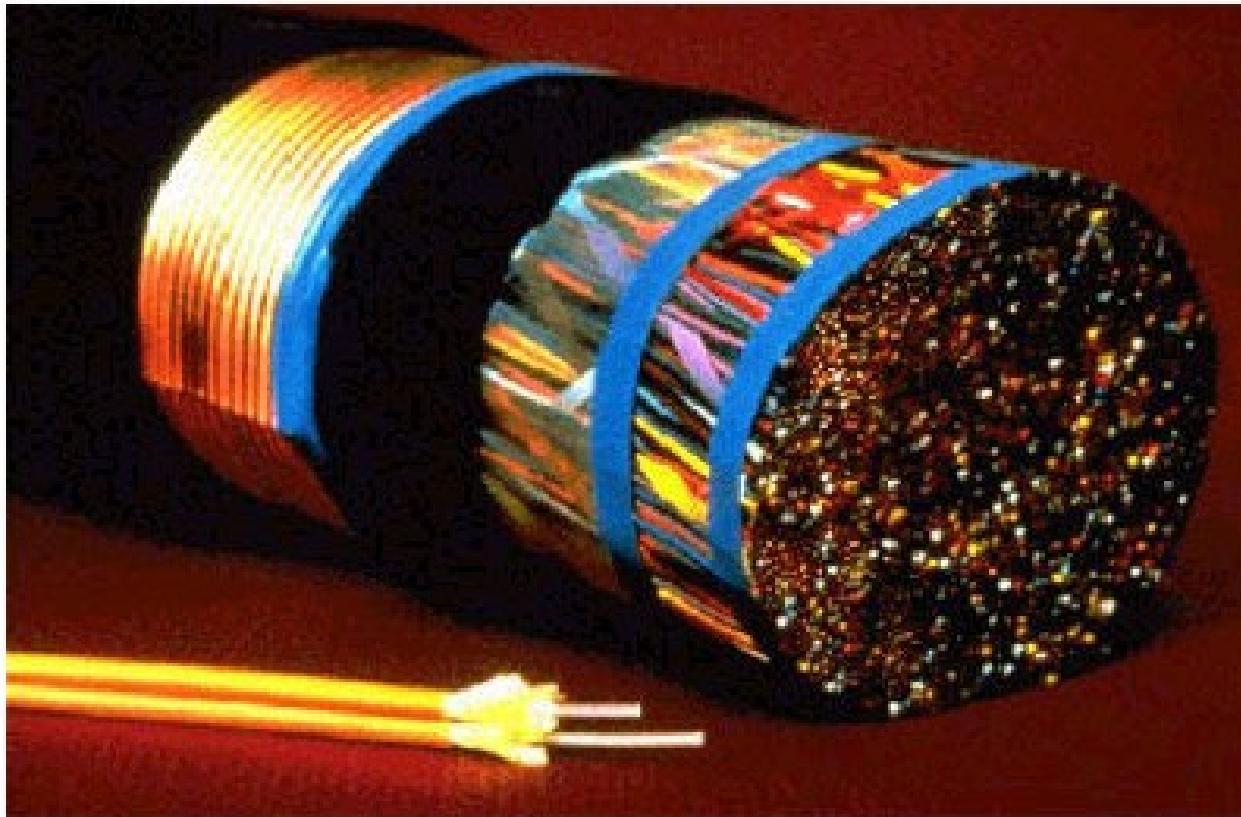
# Medios de transmisión por guía de onda

---

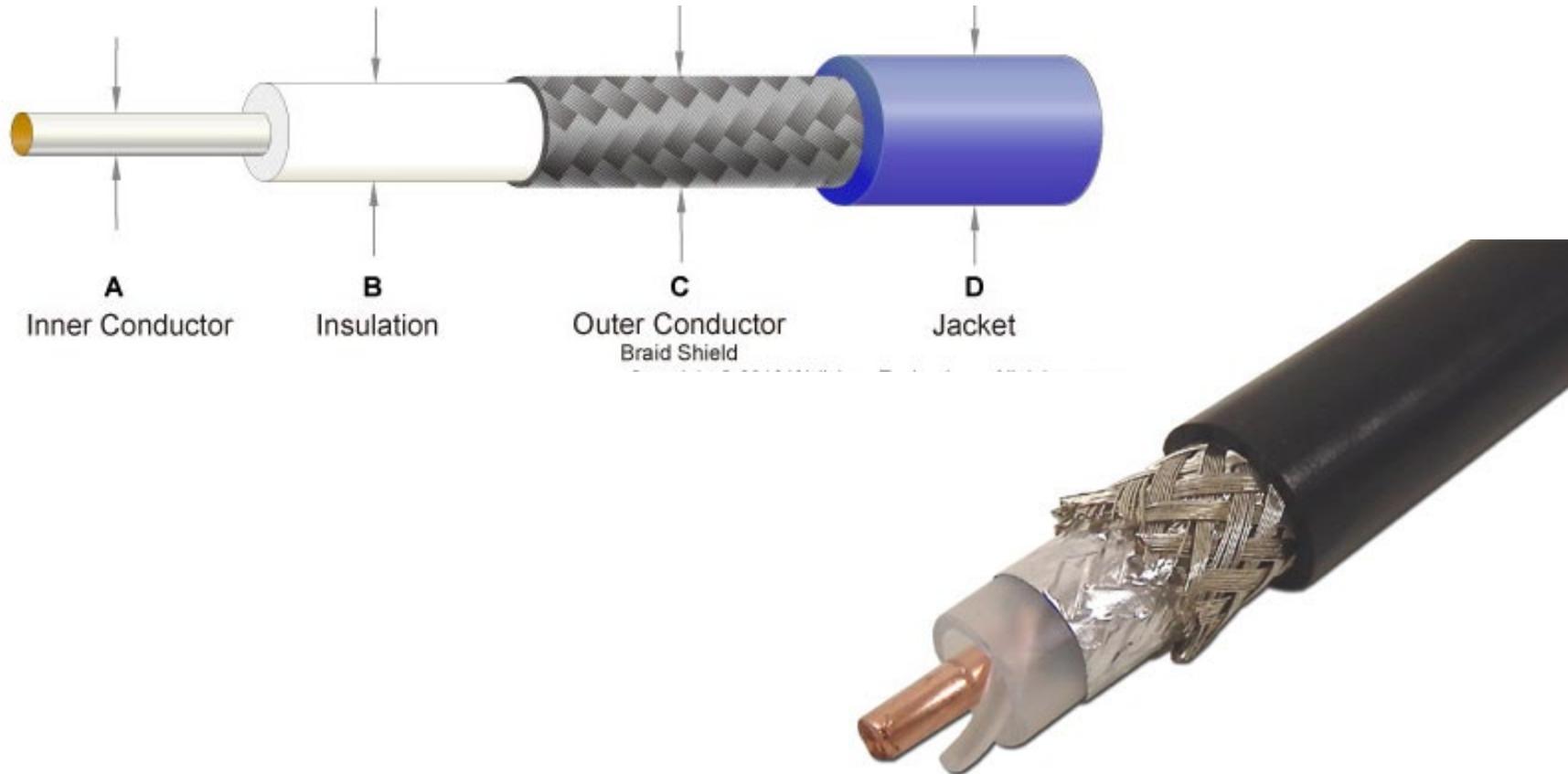
- Par trenzado de cobre
- Coaxial
- Red Eléctrica (Power Line)
- Fibra óptica
  - Monomodo
  - Multimodo

# Par de cobre

---



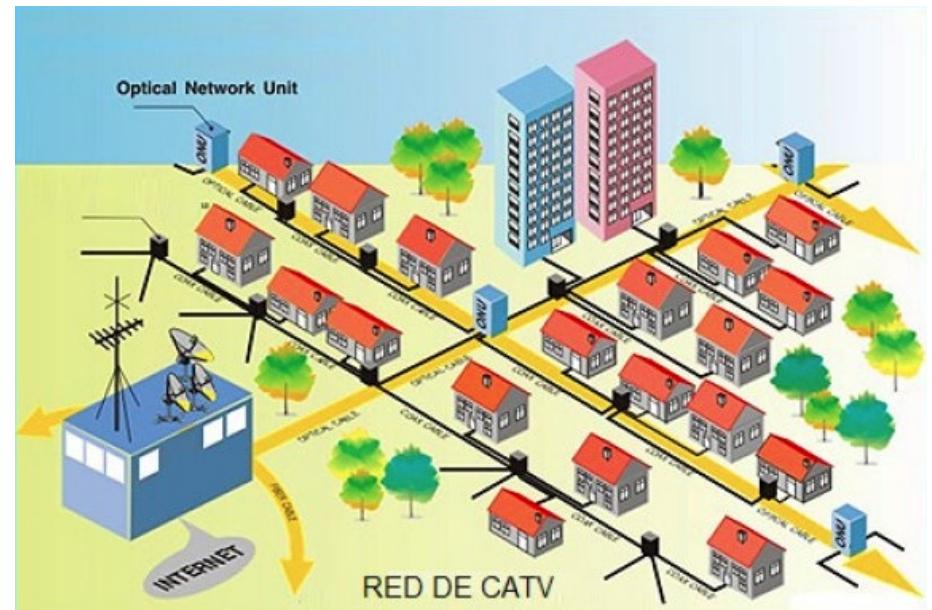
# Cable coaxial



# Coaxial: Redes CATV

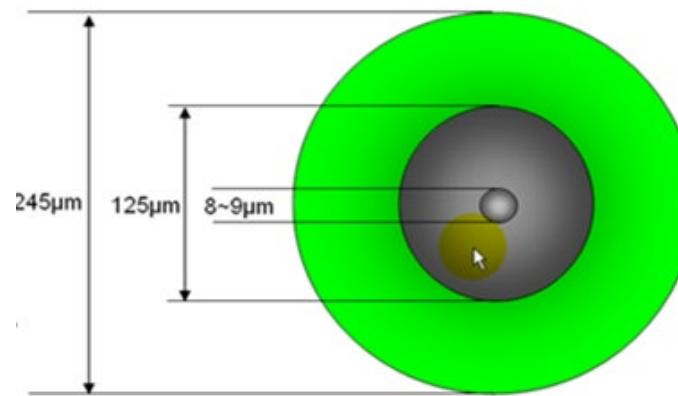
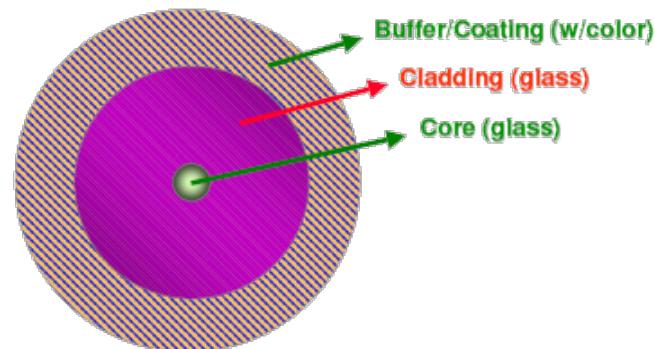
- ▶ Historia:
  - ▶ Las redes CATV (Community Antenna TeleVision) nacieron (1948) para resolver problemas de recepción en zonas de mala cobertura.
  - ▶ La antena (centro emisor) se ubicaba en sitio elevado con buena recepción. La señal se enviaba a los usuarios hacia abajo (downstream).
- ▶ Actualmente:
  - ▶ **CATV = Cable TV**
  - ▶ Cable coaxial de  $75 \Omega$
  - ▶ Amplificadores cada 0,5-1 Km. Hasta 50 en cascada.

<https://calcable.org/learn/history-of-cable/>

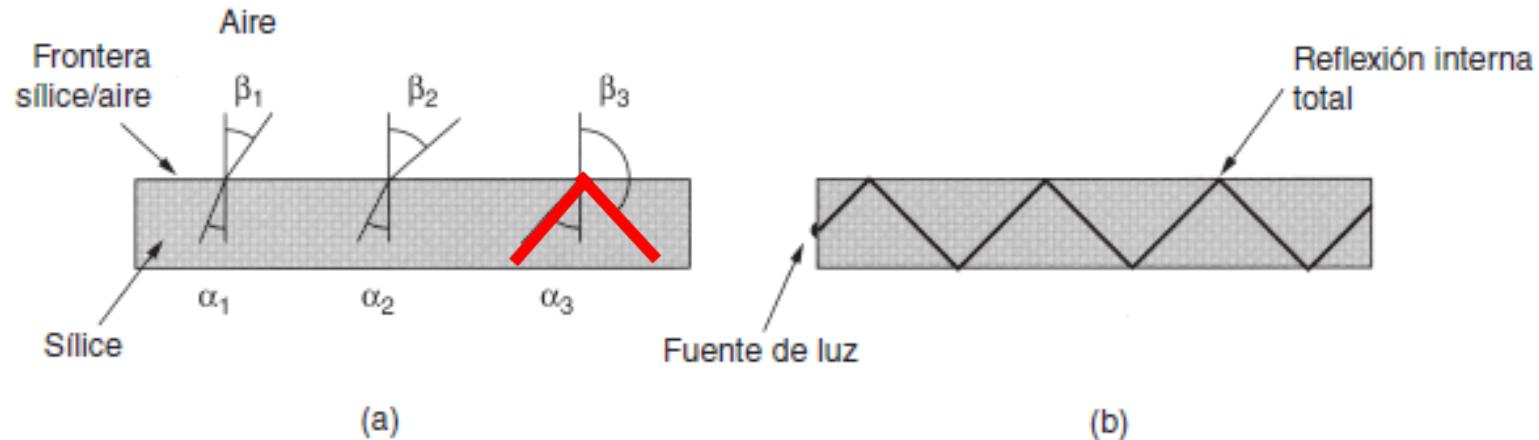


# Fibra Óptica

- La función principal de las fibras ópticas (FO) es la de guiar las ondas de luz con un mínimo de atenuación y distorsión.
- Las FO están compuestas de vidrio solidificado con un alto grado de pureza en capas llamadas núcleo (core), revestimiento (cladding) y cubierta (buffer).
- La luz se propaga únicamente por el núcleo, con una velocidad de propagación de aproximadamente hasta  $2/3$  de la velocidad de la luz en el vacío.
- $c \approx 2/3 \times c_0$



# Fibra óptica: Reflexión total interna



- (a) Tres ejemplos de un rayo de luz procedente del interior de una fibra de sílice que incide sobre la frontera de la sílice y el aire con diferentes ángulos
- (b) Luz atrapada por reflexión interna total

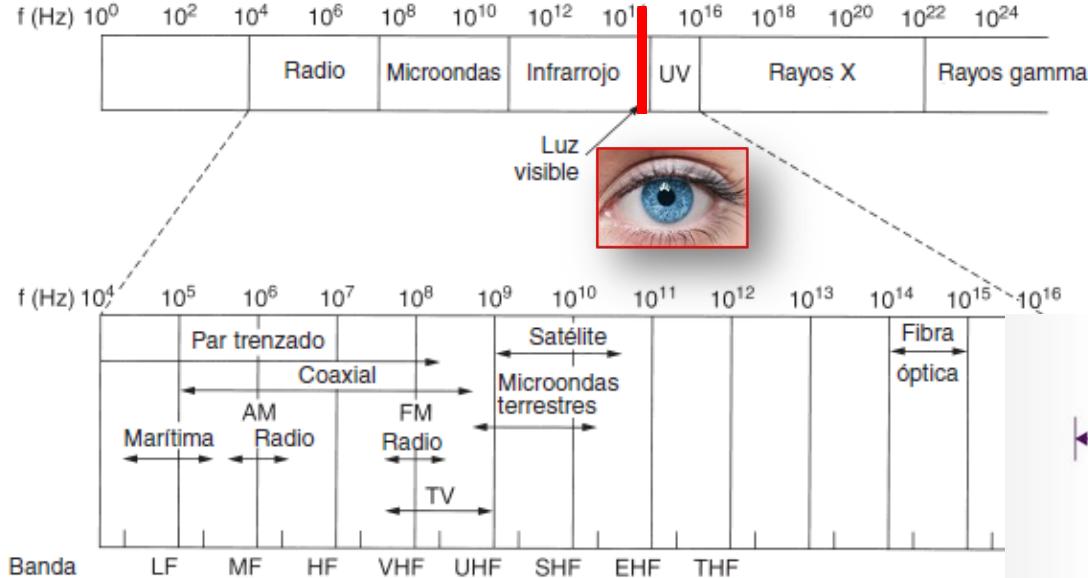
# Medios de transmisión sin guía de onda (“Wireless”)

---

## “El espectro electromagnético”

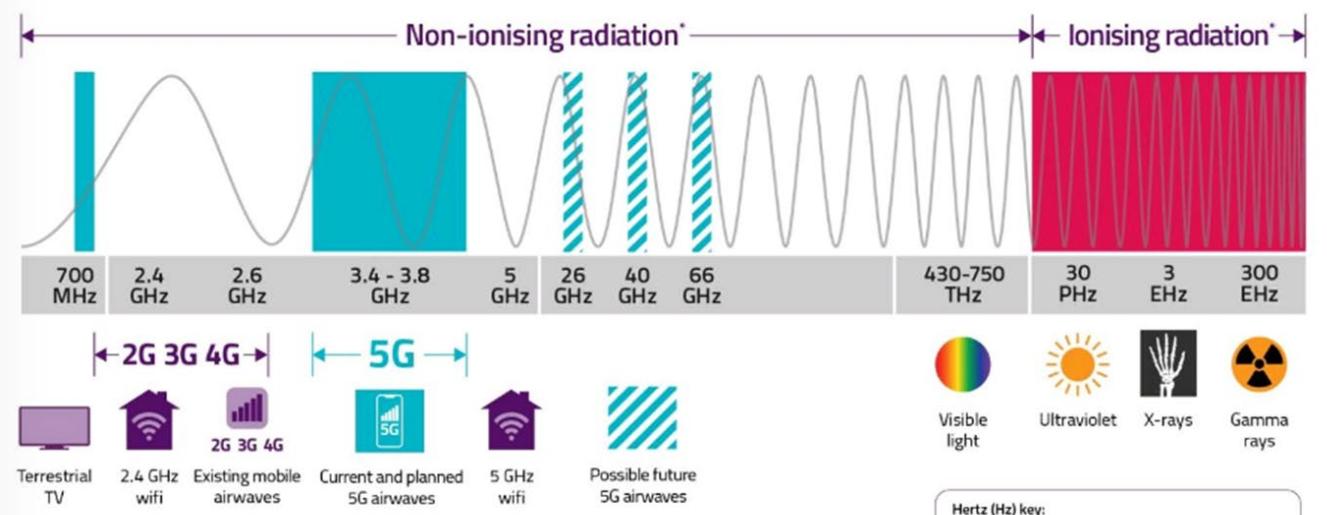
- Transmisión por radio
- Transmisión por microondas
- Transmisión por ondas infrarrojas
- Transmisión por láser
- Trasmisión Satelital
- Li-Fi

# El espectro electromagnético

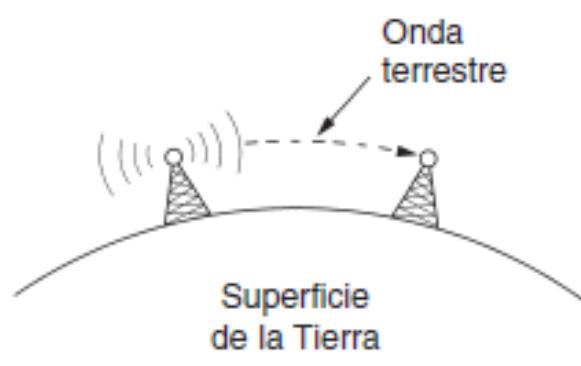


## Tecnologías

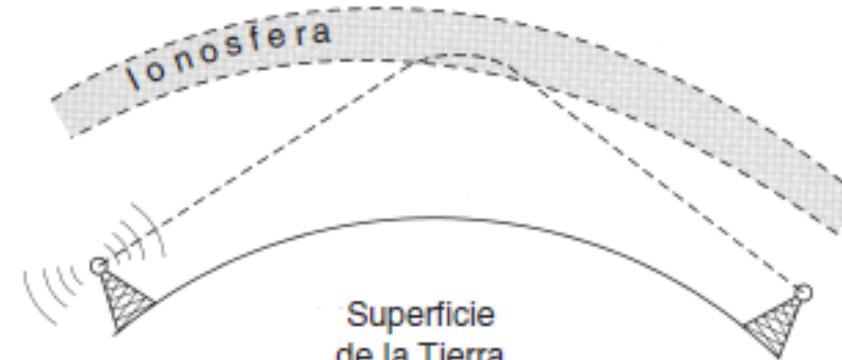
### The Electromagnetic Spectrum



# Radio



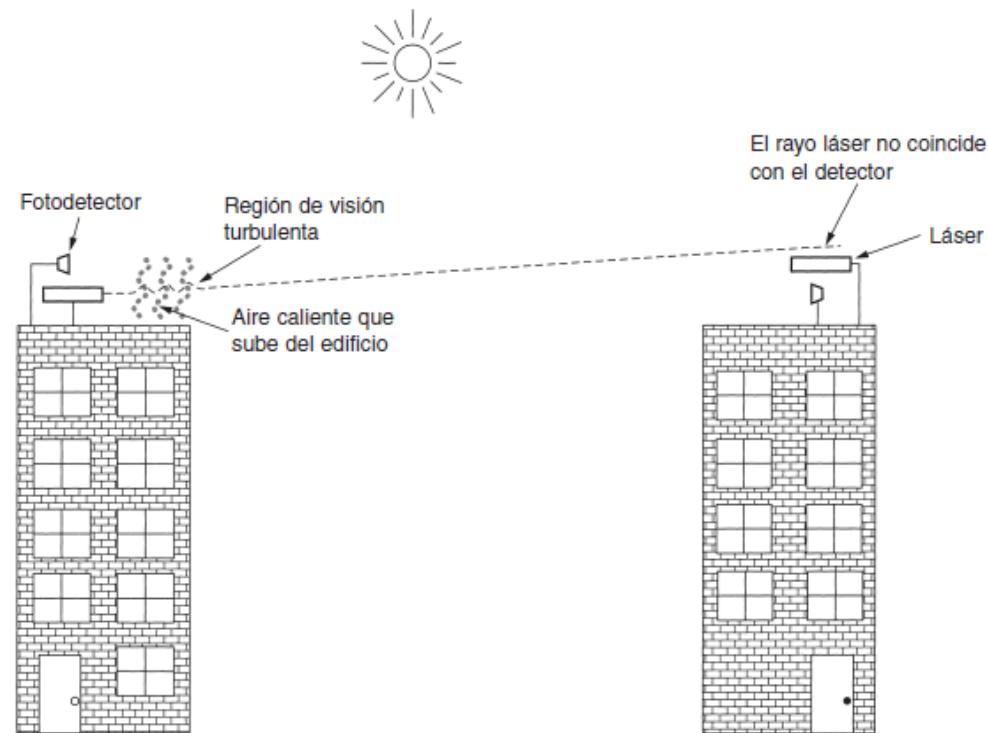
(a)



(b)

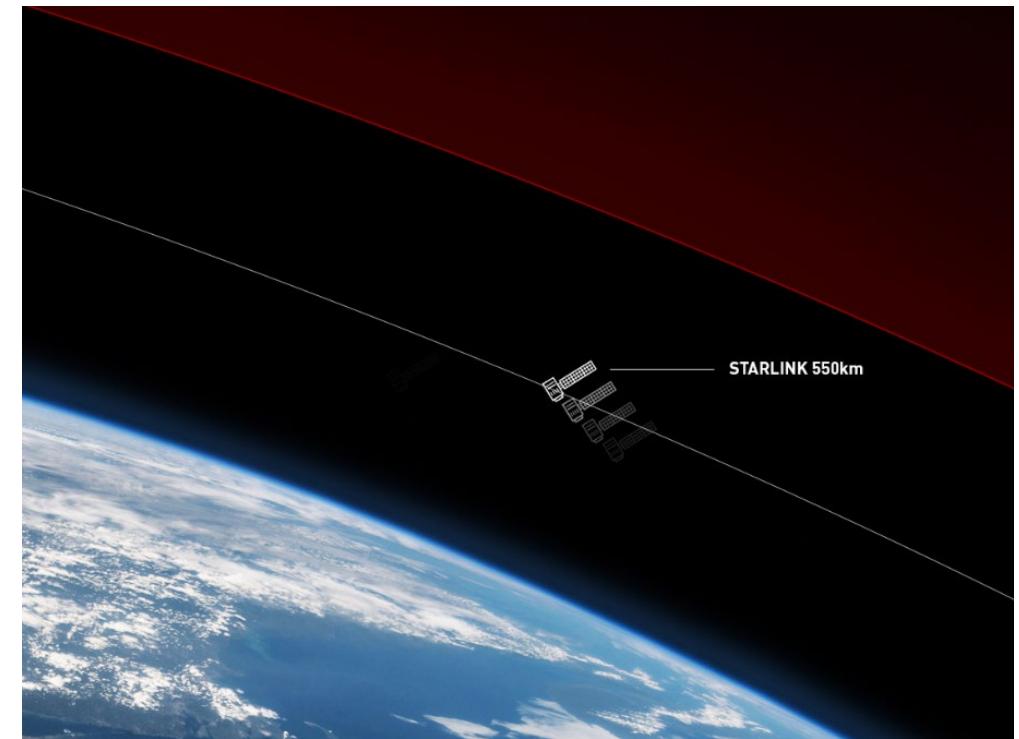
- (a) En las bandas VLF, LF y MF, las ondas de radio siguen la curvatura de la Tierra.
- (b) En la banda HF las ondas rebotan en la ionosfera

# Láser



**Figura 2-14.** Las corrientes de convección pueden interferir los sistemas de comunicación por láser. Aquí se ilustra un sistema bidireccional con dos láseres.

# Satélites



<https://www.starlink.com/>

# Li-Fi (Lo que eso de Morse, hecho luz)

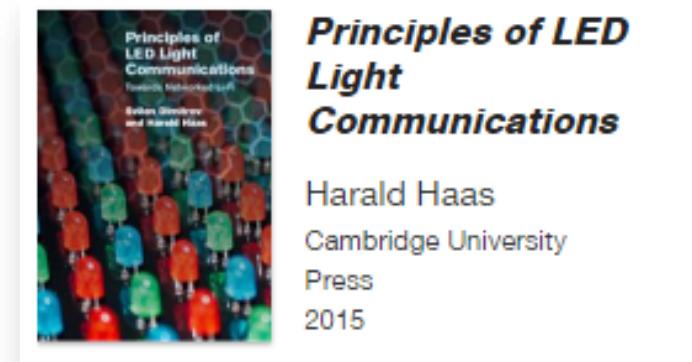


“Quisiera demostrar por primera vez en público que es posible transmitir un vídeo desde una bombilla LED comercial estándar a un panel solar con una computadora portátil que actúa como receptor. No hay Wi-Fi involucrado, es solo luz.

Y pueden preguntarse, ¿cuál es la razón? Y la razón es que habrá una extensión masiva de Internet para cerrar la brecha digital, y para permitir lo que llamamos “El Internet de las cosas” -decenas de miles de millones de dispositivos conectados a Internet-.

En mi opinión, una extensión de Internet solo puede funcionar si es casi energéticamente neutral. Lo que significa usar la infraestructura existente tanto como sea posible. Y aquí es donde la célula solar y el LED intervienen. Demostré por primera vez, en TED en 2011, la Li-Fi o fidelidad de luz. La Li-Fi utiliza LEDs comerciales para transmitir datos increíblemente rápido, y también en una manera segura y protegida. Los datos se transportan por la luz, codificados en cambios sutiles del brillo. Si miramos a nuestro alrededor, tenemos muchos LEDs que nos rodean, así que hay una rica infraestructura de transmisores Li-Fi que nos rodea” Harald Hass.

[https://www.ted.com/talks/harald\\_haas\\_a\\_breakthrough\\_new\\_kind\\_of\\_wireless\\_internet](https://www.ted.com/talks/harald_haas_a_breakthrough_new_kind_of_wireless_internet)





# Red Telefónica

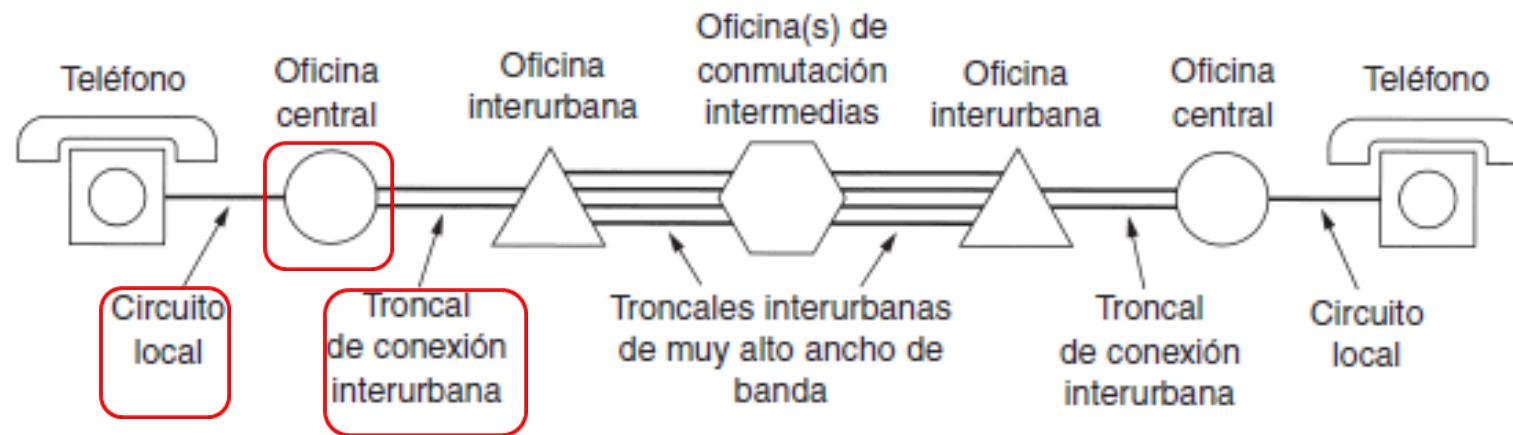
Fundamentos de la red de telefonía fija  
mediante commutación de circuitos

# Estructura del sistema telefónico

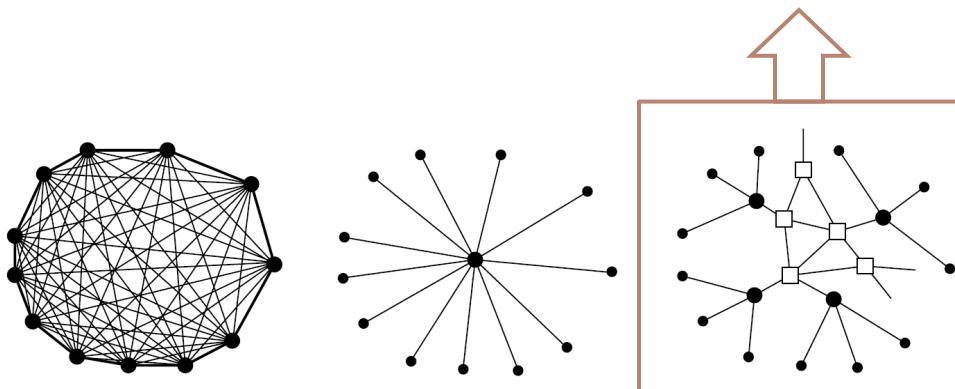
---

- ▶ **PSTN** (Public Switched Telephone Network)
- ▶ **Objetivo:** Transmitir la **voz humana** en una forma **más o menos reconocible**.
- ▶ El sistema telefónico tradicional se encuentra organizado en una jerarquía multinivel altamente redundante
- ▶ **Componentes:**
  - ▶ Local loops (pares trenzados, señalización analógica)
  - ▶ Troncales (fibra óptica o microondas, digital)
  - ▶ Oficinas de conmutación

# Red telefónica

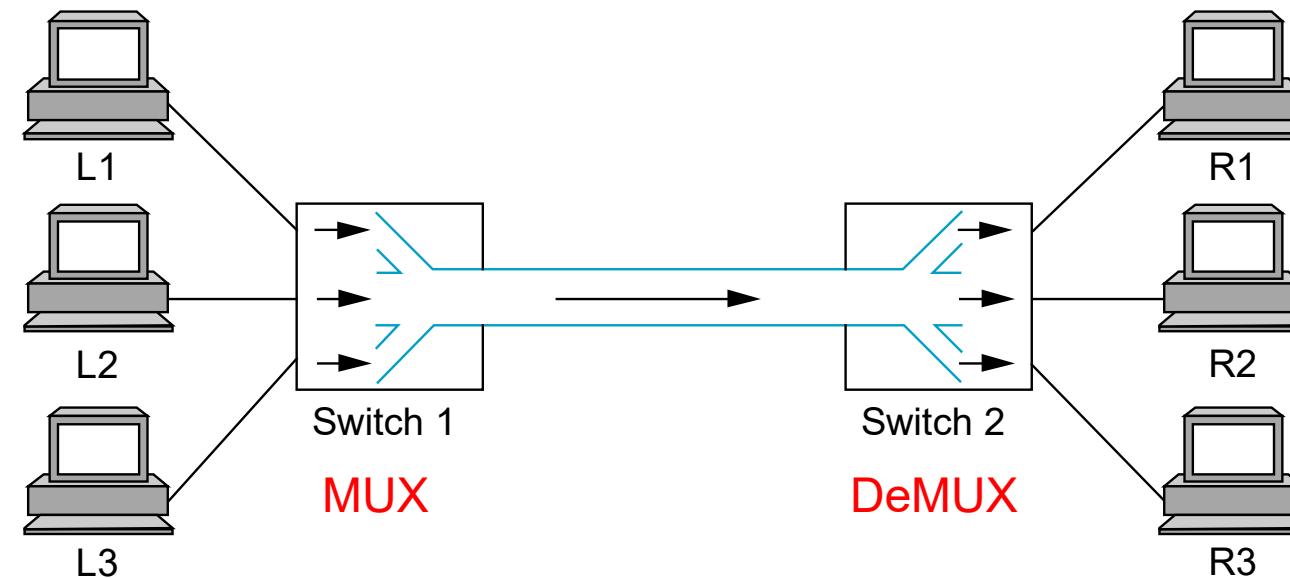


Ruta típica de un circuito para una llamada de media distancia.



# Multiplexación

- ▶ Debido a consideraciones económicas, las compañías telefónicas han desarrollado políticas elaboradas para **multiplexar** varias conversaciones sobre un único troncal físico.



# Troncales y multiplexación

---

- ▶ **TDM** (Time Division Multiplexing)
  - ▶ Los usuarios toman turnos (en “round robin”) obteniendo periódicamente cada uno el **ancho de banda completo** por un **período de tiempo acotado**
- ▶ **FDM** (Frequency Division Multiplexing)
  - ▶ El espectro de frecuencias es subdividido en canales de **ancho de banda acotado**, que es usado a **tiempo completo y exclusivo** por cada usuario.

# FDM y TDM

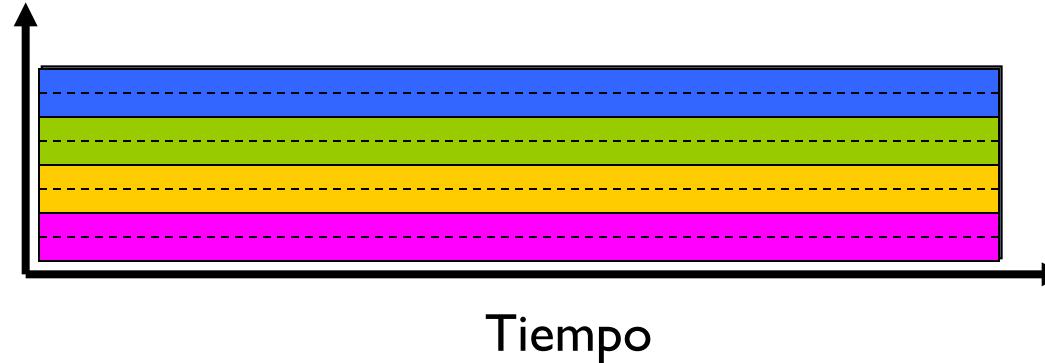
Ejemplo:

**FDM**

4 usuarios

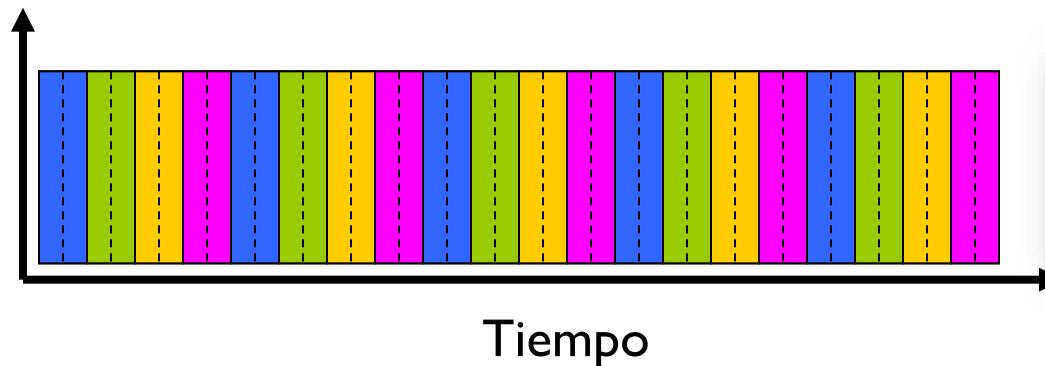


Frecuencia



**TDM**

Frecuencia

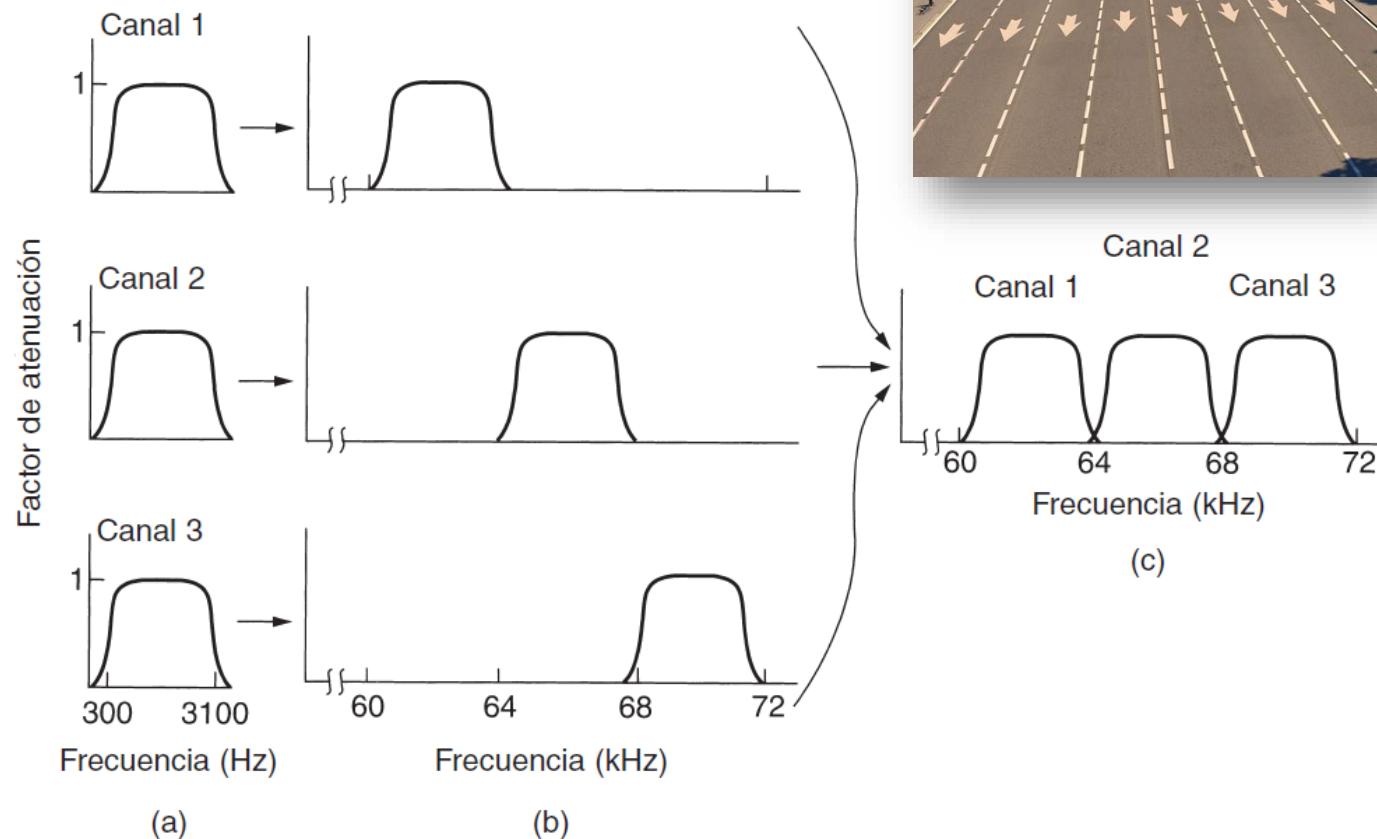


# FDM y TDM

---

- ▶ Ejemplo: difusión de **radio AM**
- ▶ Espectro reservado ~ 1 Mhz (500-1500 kHz)
- ▶ Diferentes frecuencias reservadas a diferentes **canales lógicos (emisoras)**. Cada una opera en una porción del espectro => **FDM**
- ▶ Cada estación tiene dos subcanales lógicos: música y avisos comerciales.
- ▶ Los dos alternan en la misma frecuencia, primero una ráfaga de música y luego una ráfaga de avisos y así siguiendo => **TDM**

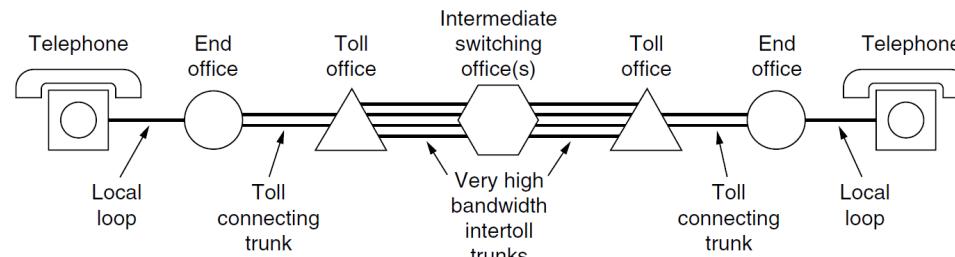
# FDM



**Figura 2-31.** Multiplexión por división de frecuencia. (a) Los anchos de banda originales. (b) Incremento de frecuencia de los anchos de banda. (c) El canal multiplexado.

# TDM

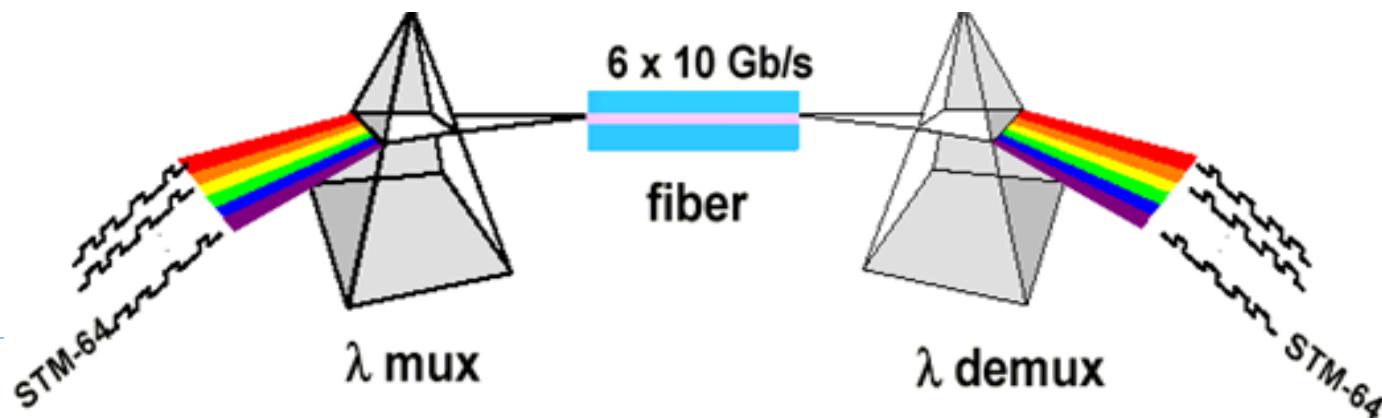
- ▶ Aunque **FDM** se utiliza todavía sobre cables de cobre o canales de microondas, requiere **circuitería analógica no trivial**.
- ▶ En contraste **TDM** puede ser manejado **enteramente por electrónica digital**, y se ha vuelto de más amplio uso en años recientes.
- ▶ **TDM** solo puede ser utilizado para datos digitales
- ▶ En telefonía:
  - ▶ Como el “local loop” produce señales analógicas, es necesario realizar una **conversión analógico/digital** en la “end office”, donde todos los “local loops” individuales se combinan sobre los “trunks” (troncales).
  - ▶ **Cómo se digitalizan múltiples señales de voz analógicas y se combinan sobre un único troncal digital ?**
  - ▶ **Conversión Analógico-Digital**



# Multiplexación por Longitud de Onda (WDM)

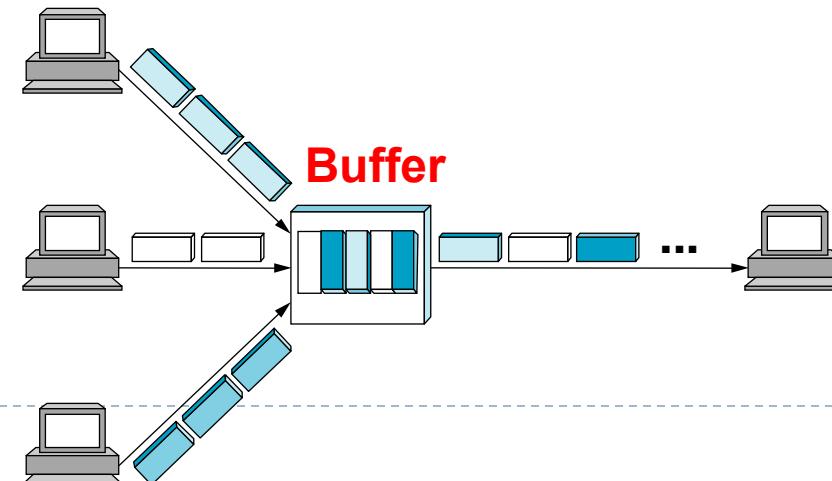


- ▶ La capacidad de una FO se puede incrementar transmitiendo **diversas** longitudes de onda por una **única** fibra.
- ▶ Esta técnica bien conocida de Multiplexación por división de frecuencia (FDM - Frequency Division Multiplexing) aplicada en los sistemas ópticos se denomina **Multiplexación por División de Longitud de Onda**.
- ▶ (WDM - Wavelength Division Multiplexing).



# Multiplexación Estadística (redes de conmutación de paquetes)

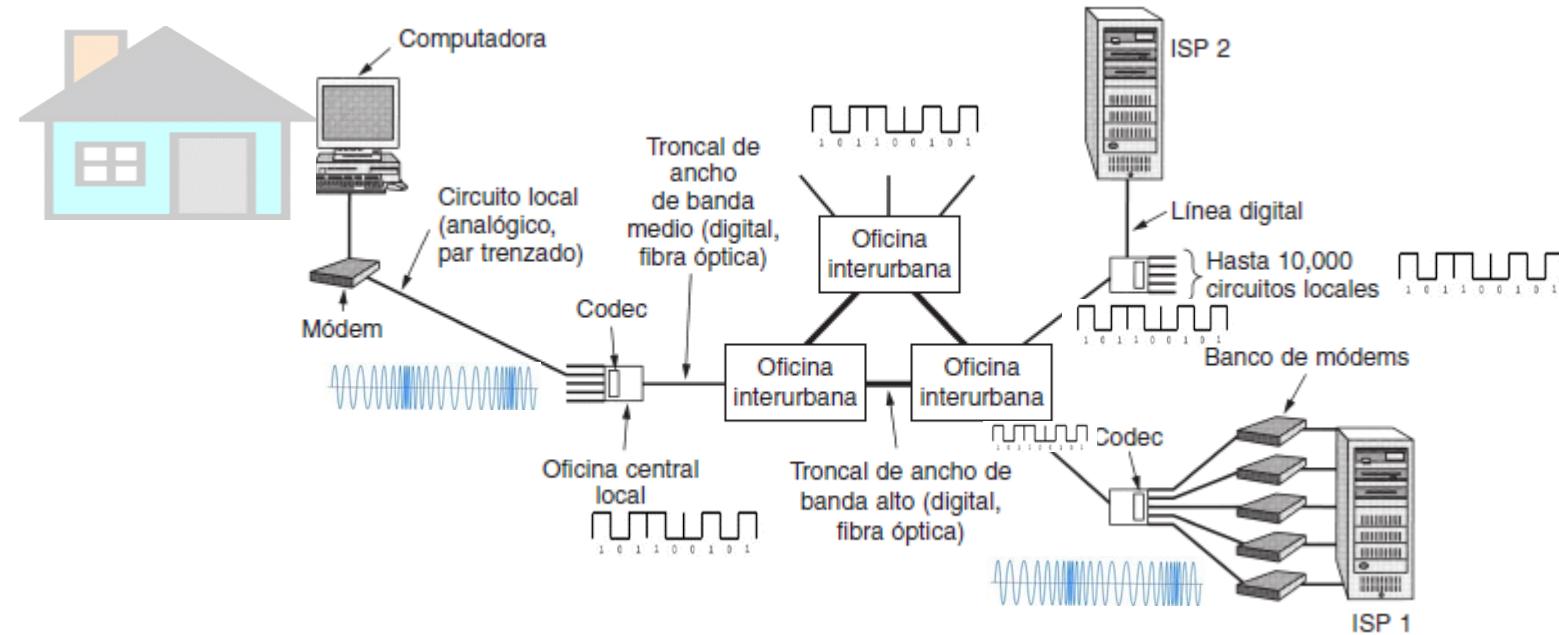
- ▶ División del tiempo “bajo demanda”
- ▶ Paquetes de diferentes fuentes “comparten” el enlace, a tiempos distintos
- ▶ Se “encolan” los paquetes que “compiten” por el enlace cuando el mismo no está disponible
- ▶ Cuando hay “overflow” decimos que tenemos **congestión**



# Conversión Analógico - Digital

Teorema del Muestreo – Codificación  
Modulación PCM

# Conversión analógico/digital



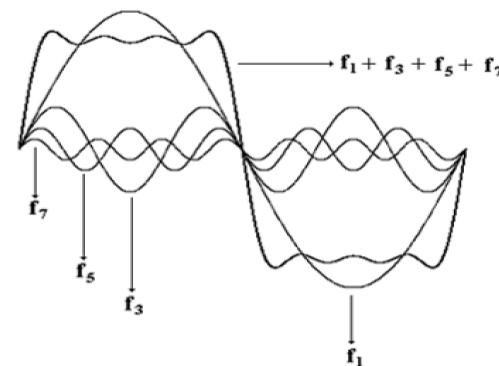
Uso de transmisión analógica y digital para una llamada de computadora a computadora. Los módems y los codecs realizan la conversión.

# Teorema de Muestreo

- ▶ Vimos que las señales periódicas se pueden descomponer como una **sumatoria de componentes en senos y cosenos**.
- ▶ Cada una de una amplitud, frecuencia y fase diferentes (Desarrollo de una señal periódica en su **Serie Infinita de Fourier**)

$$f(t) = \frac{1}{2}a_0 + \sum_{n=1}^{\infty} [a_n \cos(n\omega_0 t) + b_n \sin(n\omega_0 t)]$$

Ejemplo onda cuadrada, fundamental + 3 armónicas:



- ▶ Si a dichas sinusoides las “muestreamos”, el caso más crítico de muestreo será aquella de **mayor frecuencia** (frecuencia máxima  $f_{max}$ , corresponde al período mínimo  $T_{min}=1/f_{max}$ ), siendo la señal  $f(t)=A \sin(2\pi f_{max} t + \phi)$

A: amplitud, t: tiempo y  $\phi$ : fase

# Muestreo

---

- ▶ El **Teorema de Muestreo formulado por Nyquist (1924) dice:**
  - ▶ Si queremos **reconstruir** una señal de componente frecuencial máxima  $f_m$  debemos muestrearla según  $f_s > 2 * f_m$  llamada **frecuencia de sampling** (también de “muestreo”, o de “modulación”)
  - ▶ **Ejemplo 1:** los CD de audio almacenan muestras producidas por muestreos a 44.100 veces por segundo. Por tanto pueden reproducir música conteniendo frecuencias de hasta ~22 KHz
  - ▶ **Ejemplo 2:** Si la voz humana con calidad “**suficiente para telefonía**” se estima en el espectro de 0 a 4 KHz, para poder muestrear y recuperar la señal requeriríamos 8000 muestras por segundo.
  - ▶ Cuantos dígitos binarios uso para codificar cada muestra ? 4 , 7, 8, 16 ...? Lo veremos luego en PCM.



# Modulación



Modems

# Recordando los Principios básicos:

---

- ▶ Señal analógica vs. señal digital
  - ▶ La señal analógica utiliza una magnitud con una variación continua, abarcando infinitos niveles posibles.
  - ▶ La señal digital emplea un número finito de valores discretos posibles, predefinidos.
- ▶ Módem vs. Códec
  - ▶ **Módem** (**MOD**ulador-**DEM**odulador): convierte (“traduce”) de **digital a analógico** (y viceversa).
  - ▶ **Códec** (**CO**dificador-**DEC**odificador): convierte (“traduce”) de **analógico a digital** (y viceversa)
  - ▶ No son lo mismo !

# Modulación

---

- ▶ Proceso de variación de **cierta característica** de una señal sin mensaje, llamada **portadora**, de acuerdo con una **señal mensaje**, llamada **moduladora**
  
- ▶ Tipos
  - ▶ Moduladora Analógica/Portadora Analógica
  - ▶ Moduladora Digital/Portadora Analógica
  - ▶ Moduladora Analógica/Portadora Digital
  - ▶ Moduladora Digital/Portadora Digital

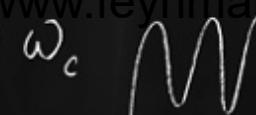
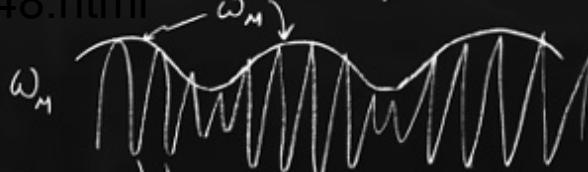
# Batido y Modulación

Summary

If two oscillations of different frequency are added ( $A_1 e^{i\omega_1 t} + A_2 e^{i\omega_2 t}$ )

the result is equivalent to a wave of freq.  $\omega_c = \frac{\omega_1 + \omega_2}{2}$  whose amplitude changes

in time at frequency  $\omega_m = \frac{\omega_1 - \omega_2}{2}$ :  
(intensity oscillates at frequency  $\omega_1 - \omega_2$ )

Conversely a carrier  $\omega_c$   modulated by  $\omega_m$  

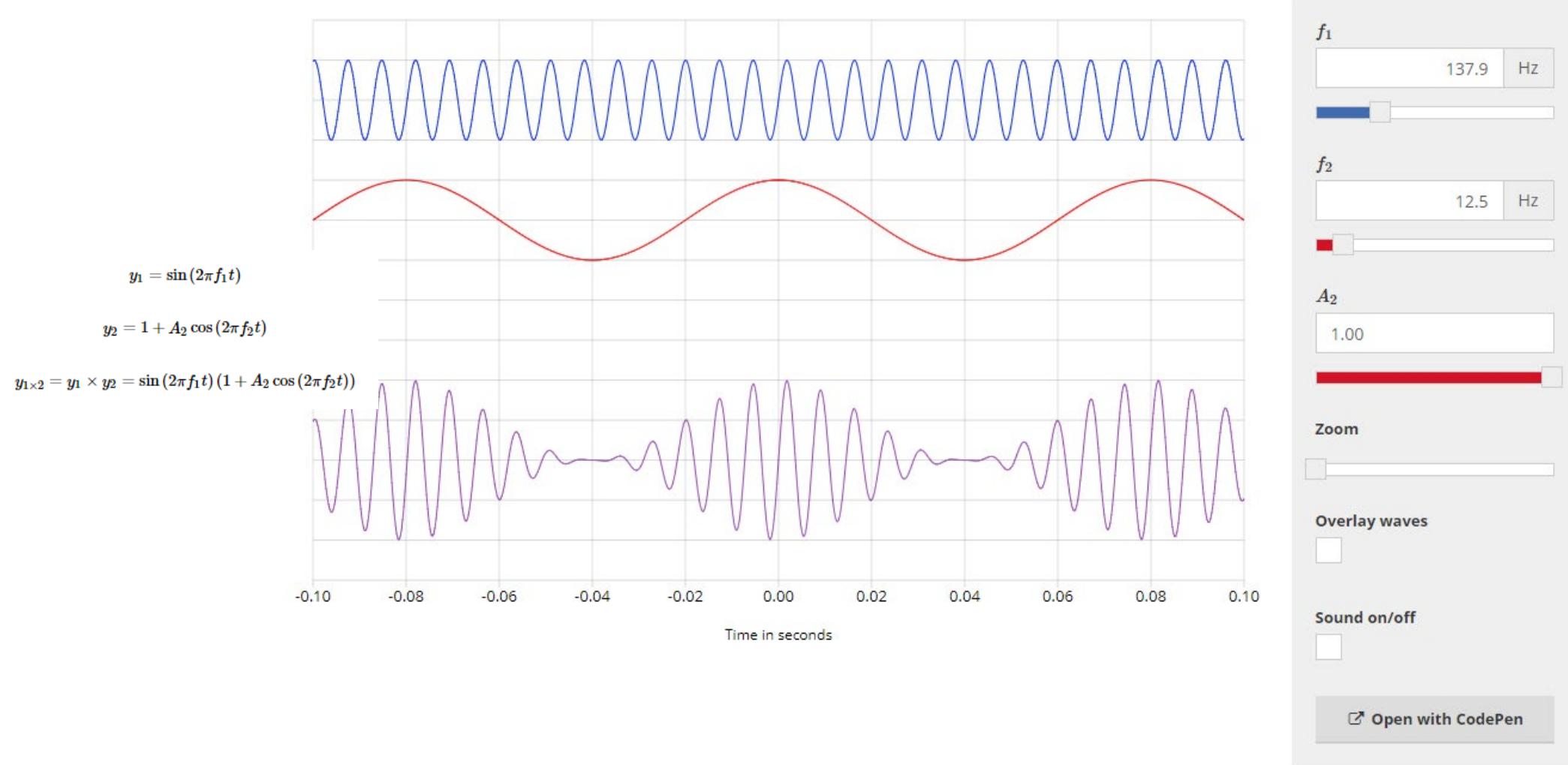
is equivalent to 3 waves: carrier at  $\omega_c$  plus two "side bands" at  $\omega_c + \omega_m$  and  $\omega_c - \omega_m$ .

A wave in space  $e^{i(\omega t - kx)}$  has  $\omega$  depending on  $k$ , so phase velocity (velocity

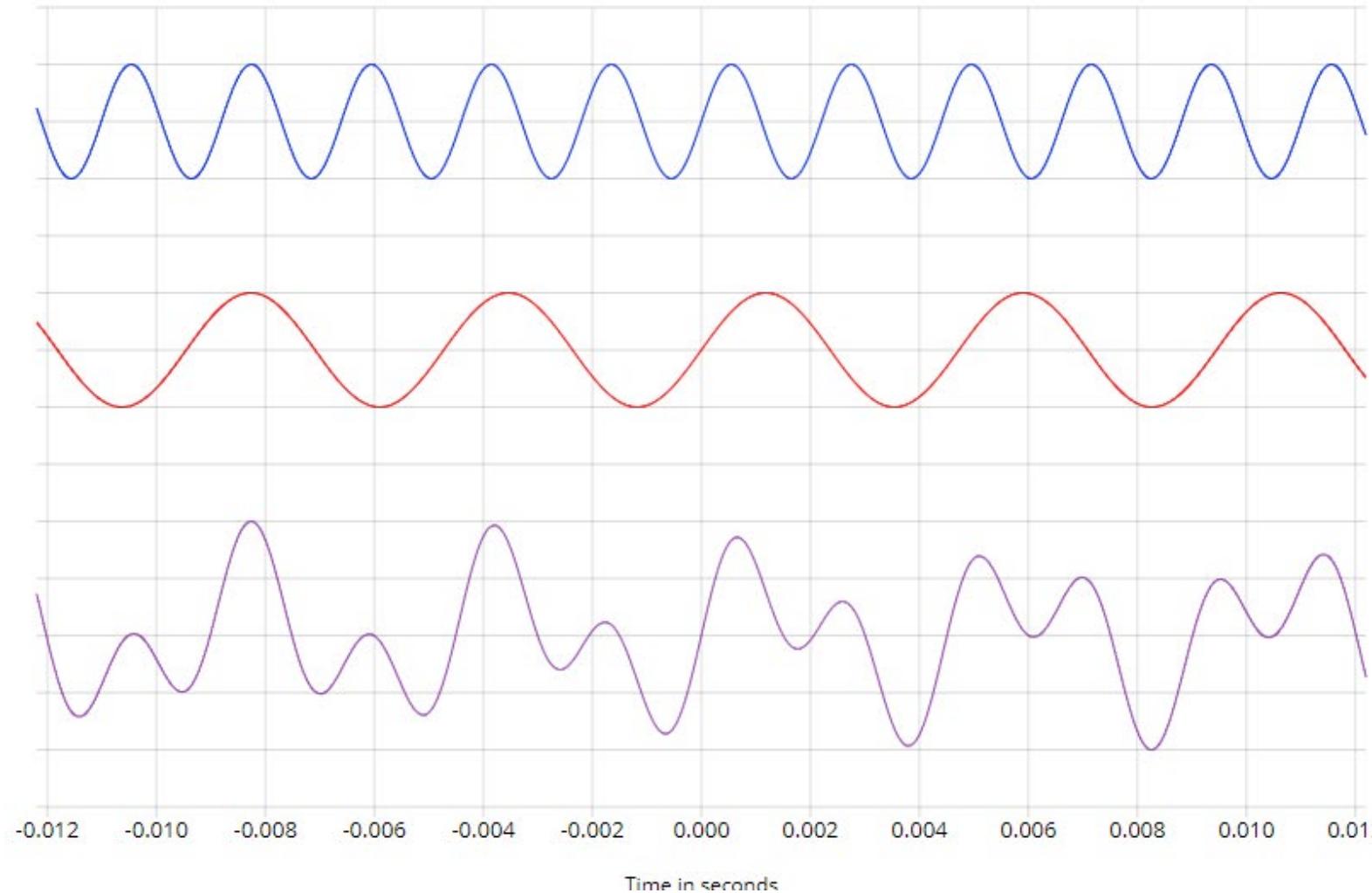
of nodes) =  $N_{ph} = \omega/k$  — but the speed of propagation of slow modulations (signal velocity)

or group vel) =  $N_{gr} = d\omega/dk$ . For light in vac, sound wave etc for which  $\omega = kc$ ,  $N_{ph} = N_{gr} = c$ .

<https://academo.org/demos/amplitude-modulation/>



<https://academo.org/demos/wave-interference-beat-frequency/>



$f_1$   Hz

$f_2$   Hz

Zoom

Overlay waves

Sound on/off

# M. Analógica/P. Analógica

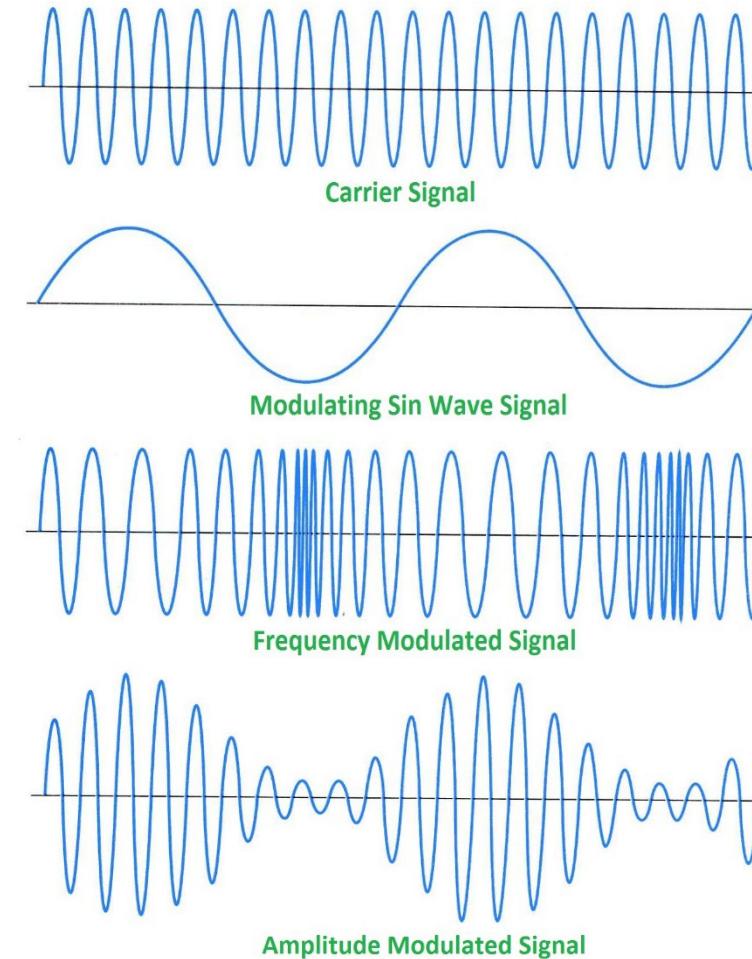
---

**Portadora**

**Señal modulante**

Onda modulada en **frecuencia**

Onda modulada en **amplitud**



# M. Digital/P. Analógica

---

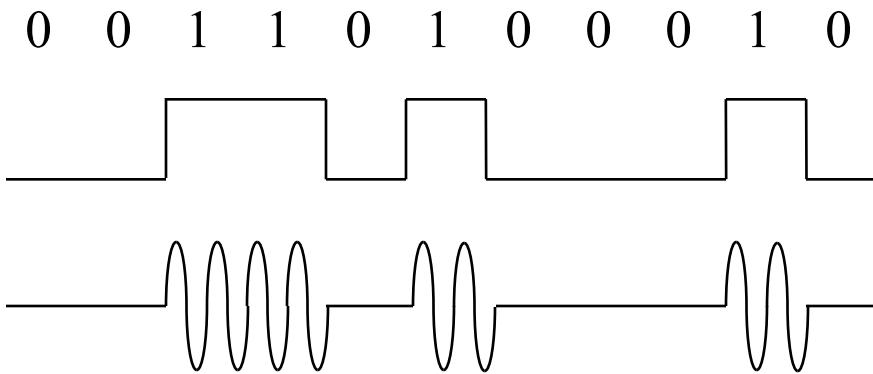
- ▶ La situación más conocida es la transmisión de datos digitales a través de la red de telefonía
- ▶ Diseñada para transmitir señales analógicas en el rango de frecuencias de la voz (300-3400 Hz)
- ▶ Técnicas
  - ▶ Desplazamiento de Amplitud (ASK)
  - ▶ Desplazamiento de Frecuencia (FSK)
  - ▶ Desplazamiento de Fase (PSK)
  - ▶ **Mixtas**

# M. Digital/P. Analógica

---

**ASK**, los valores binarios se representan mediante dos amplitudes diferentes de la portadora

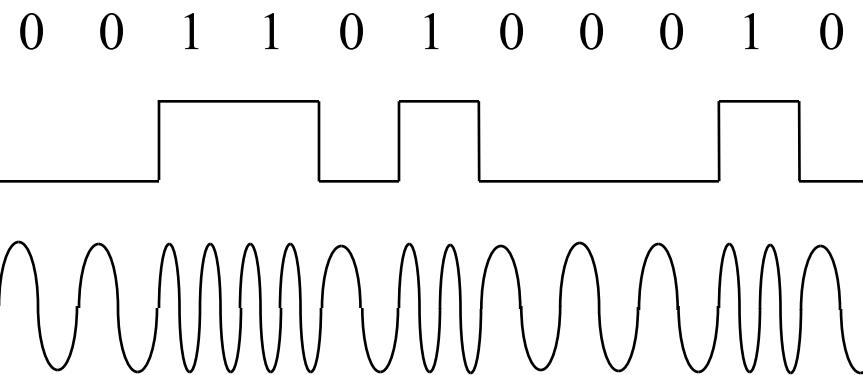
$$S(t) = \begin{cases} A \cdot \cos(2\pi f_c t) \\ 0 \end{cases}$$



# M. Digital/P. Analógica

**FSK**, los valores binarios se representan mediante dos frecuencias diferentes de la portadora

$$S(t) = \begin{cases} A \cdot \cos(2\pi f_1 t) \\ A \cdot \cos(2\pi f_2 t) \end{cases}$$

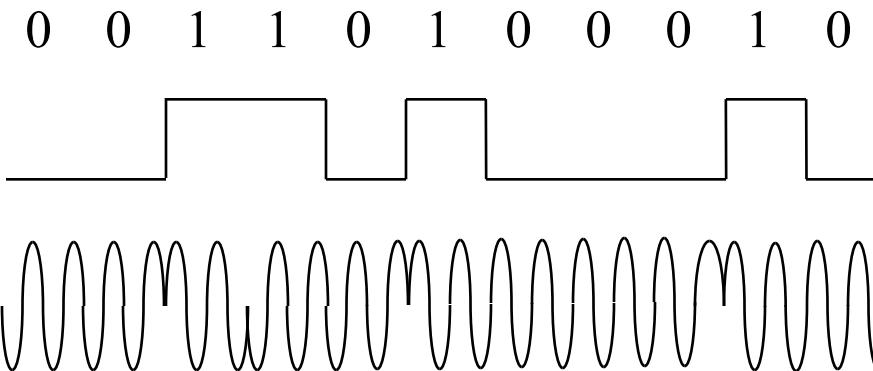


# M. Digital/P. Analógica

---

**PSK**, los valores binarios se representan mediante dos fases diferentes de la portadora

$$S(t) = \begin{cases} A \cdot \cos(2\pi f_c t + \pi) \\ A \cdot \cos(2\pi f_c t + 0) \end{cases}$$



# Velocidad de Modulación

---

- ▶ La **Velocidad de Modulación** se define como el número de cambios de señal por unidad de tiempo, y se expresa en baudios (símbolos/segundo)
- ▶ La **Velocidad de Transmisión** equivale a la velocidad de modulación multiplicado por el número de bits  $N$  representados por cada símbolo, expresada en bits/segundo:

$$V_t = V_m \cdot N$$

# M. Digital/P. Analógica ( cont.)

- ▶ Modulación **Multinivel**
- ▶ Se consigue una **utilización más eficaz del ancho de banda** si cada elemento de la señal transmitida representa **más de un bit**.

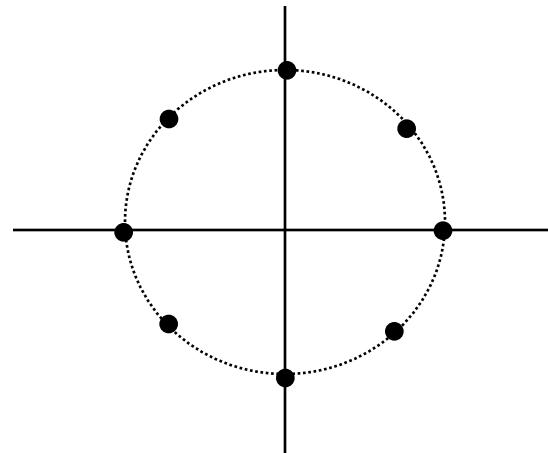
$$S(t) = \begin{cases} A \cdot \cos(2\pi f_c t + 1/4 \pi) & 11 \\ A \cdot \cos(2\pi f_c t + 3/4 \pi) & 10 \\ A \cdot \cos(2\pi f_c t + 5/4 \pi) & 00 \\ A \cdot \cos(2\pi f_c t + 7/4 \pi) & 01 \end{cases}$$

- ▶ Este esquema se puede ampliar, ya que se pueden transmitir 2, 3, 4, etc. bits por señal transmitida
- ▶ Aumentando:
  - ▶ el número de fases distintas (PSK)
  - ▶ y además el número de amplitudes para cada fase (ASK-PSK)

# M. Digital/P. Analógica

---

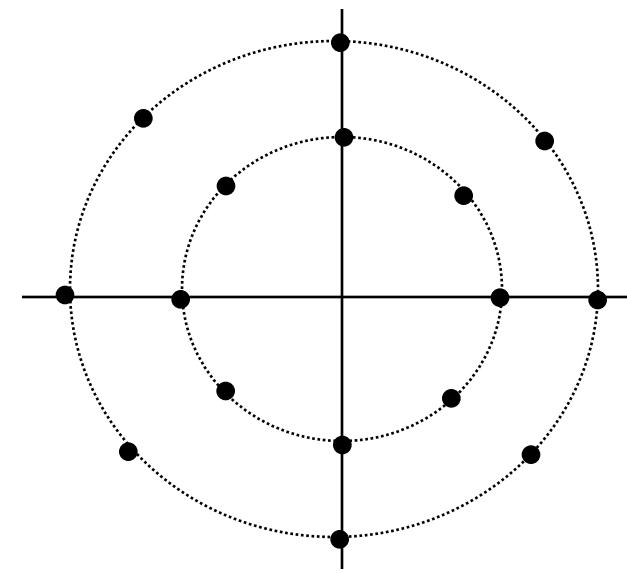
PSK



8 símbolos  
 $N = 3$

ASK-PSK

16 símbolos  
 $N = 4$



# M. Analógica/P. Digital

---

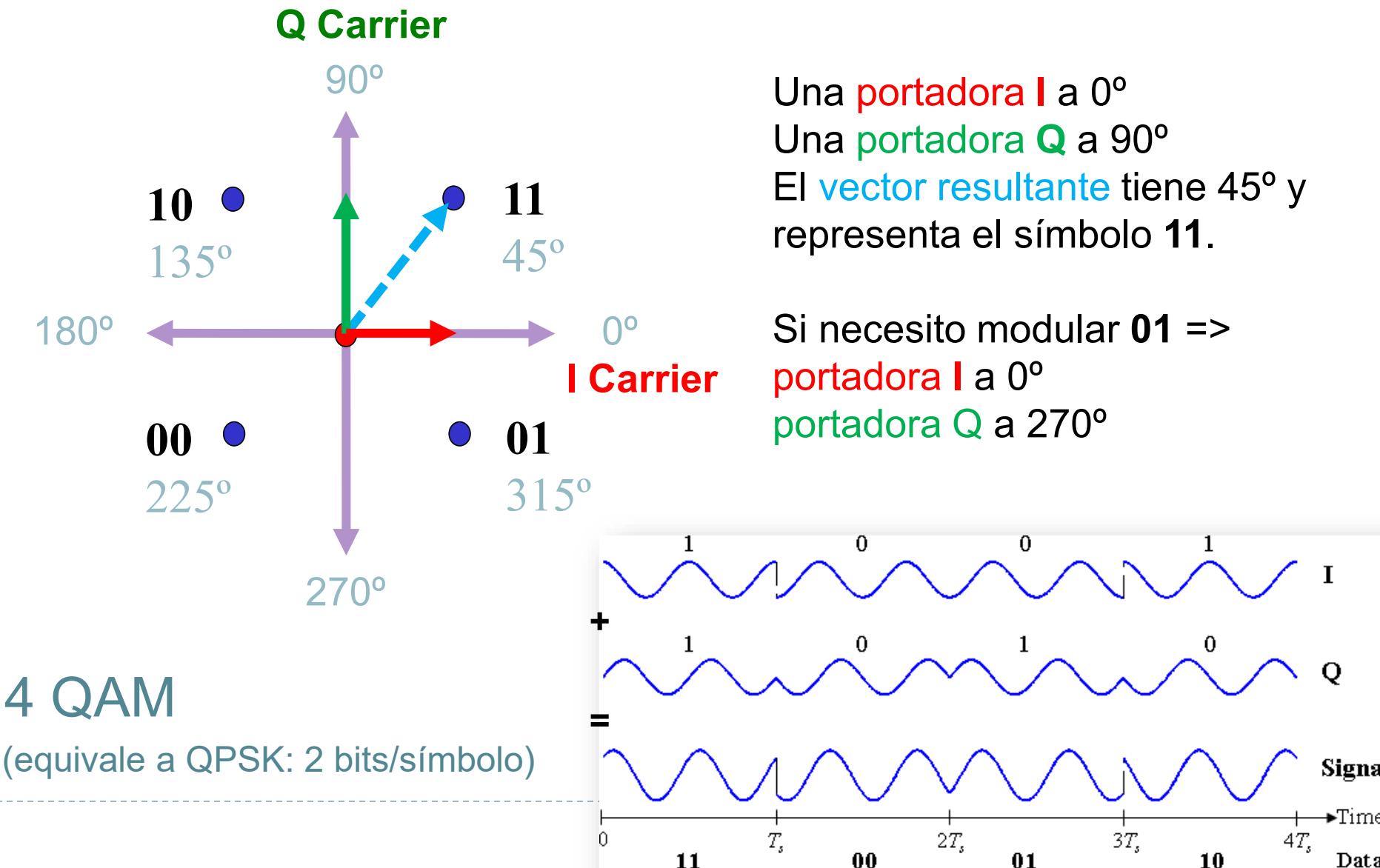
- ▶ Vimos que el proceso de conversión de señales analógicas en digitales se denomina **digitalización**.
  - ▶ Los dispositivos que lo llevan a cabo se llaman CODECS
- 
- ▶ Métodos
    - ▶ Modulación por Impulsos Codificados (MIC, o PCM)
    - ▶ Modulación Delta (DM)
      - ▶ Codifica sólo las diferencias. Para ciertas señales es más eficiente que PCM.

# M. Digital/P. Digital

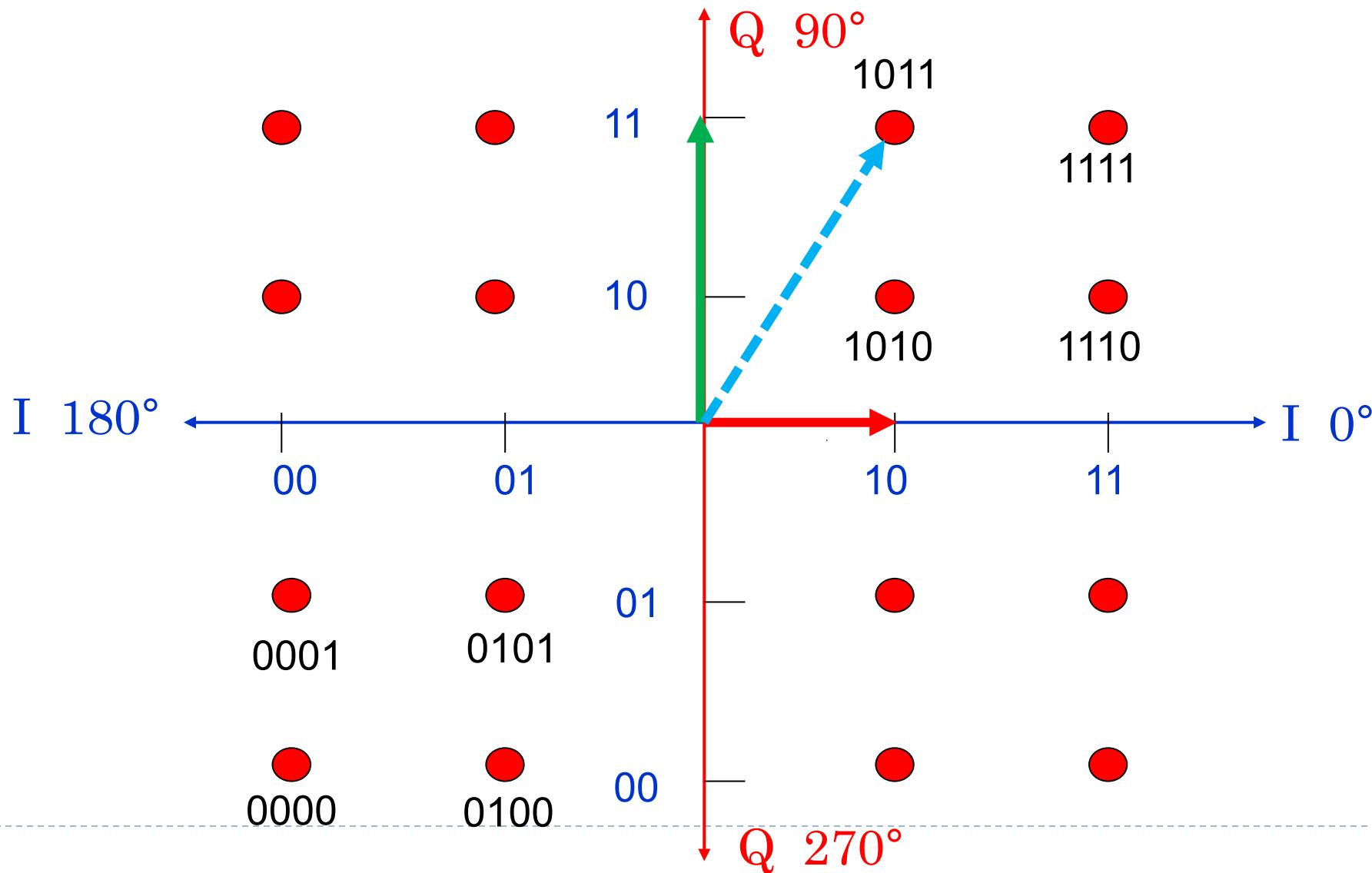
---

- ▶ Los datos binarios se transmiten codificando cada bit de datos en cada elemento de señal

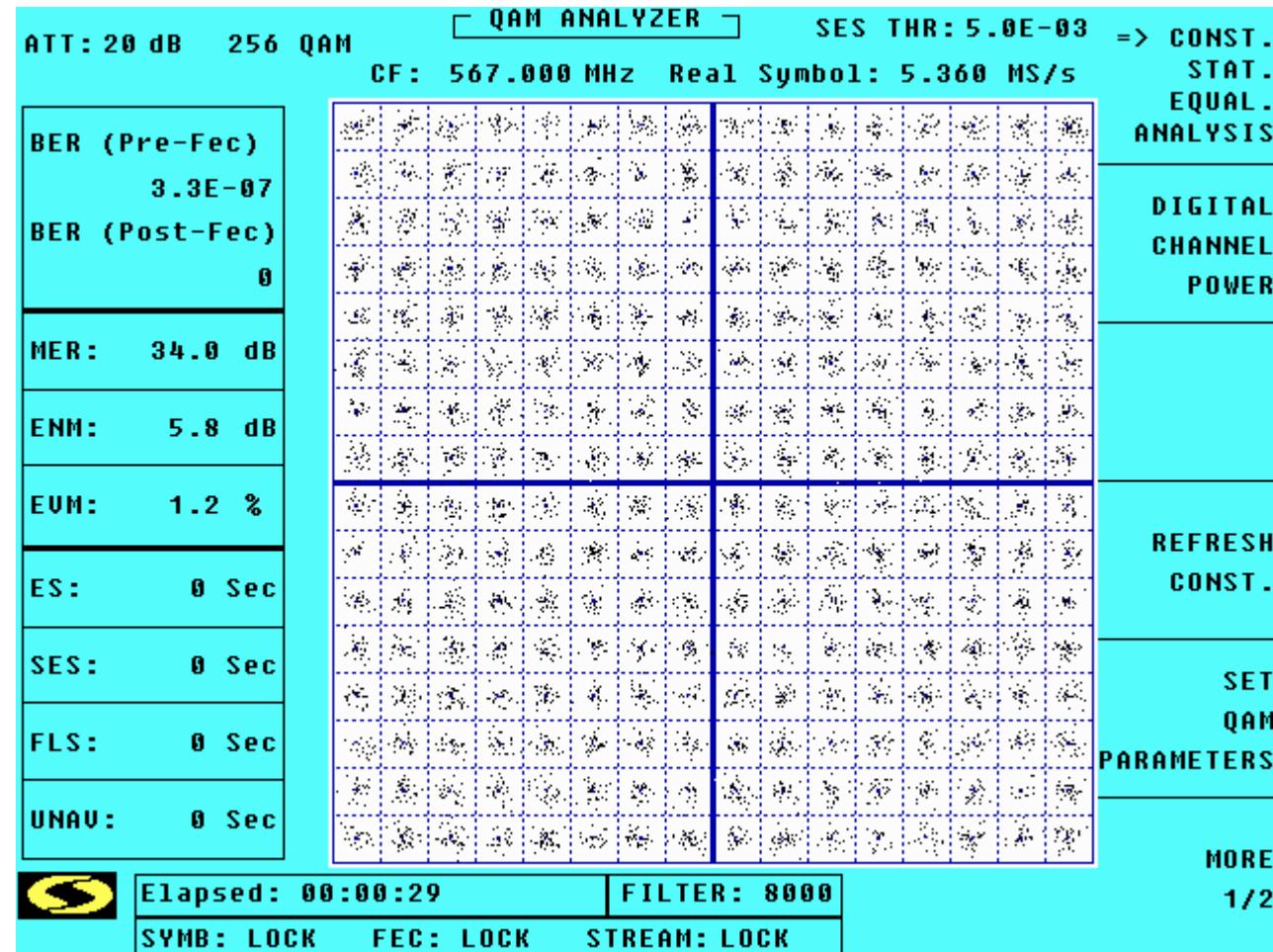
# Vectores y Modulación: QAM



# Vectores y Modulación: 16 QAM



# Modulación 256 QAM

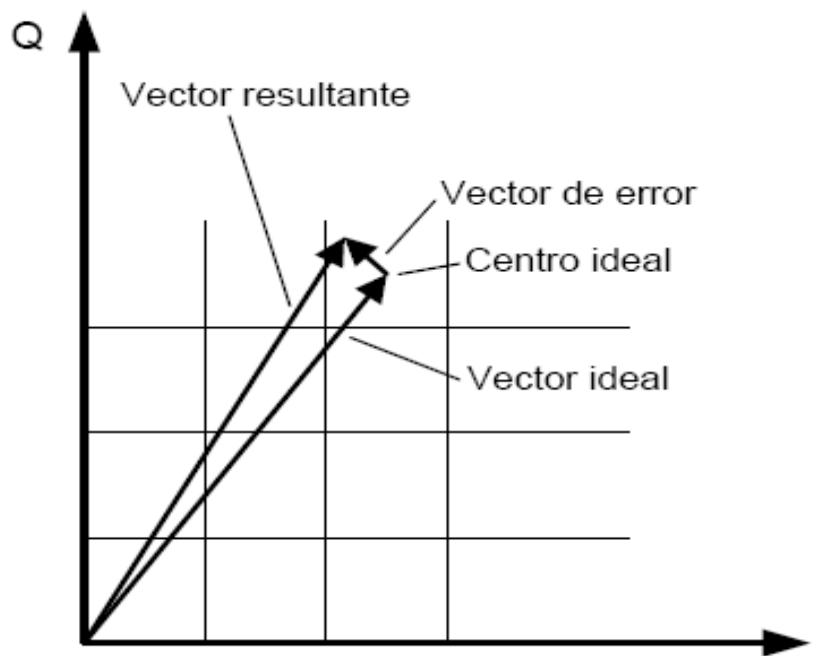


8 Bits por símbolo

# MER : Modulation Error Ratio

- ▶ MER se expresa en dB.

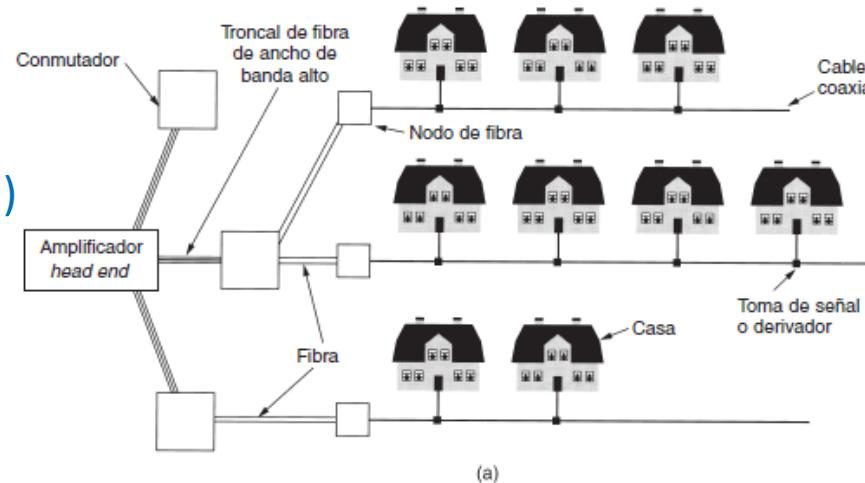
$$10 \log \frac{\text{RMS error magnitude}}{\text{average symbol magnitude}}$$



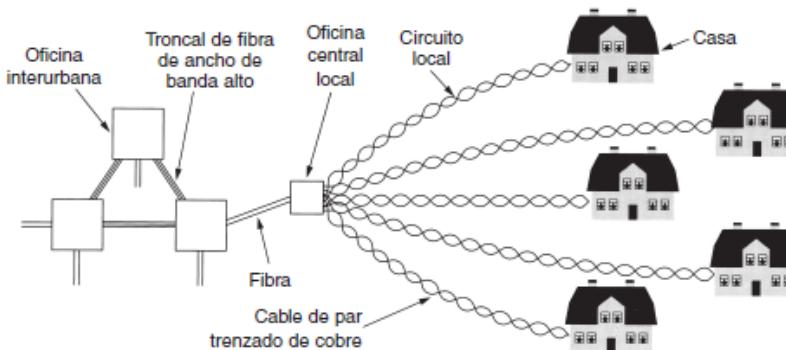
# Tecnologías de acceso

# TV por Cable y Telefonía Fija

Red HFC ( Hybrid Fiber Coaxial)

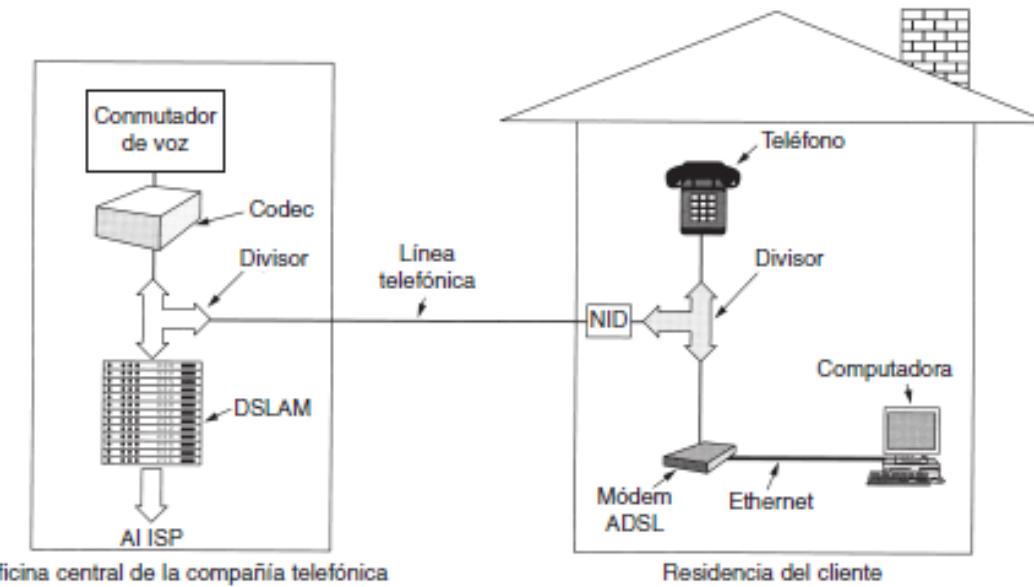


(a)



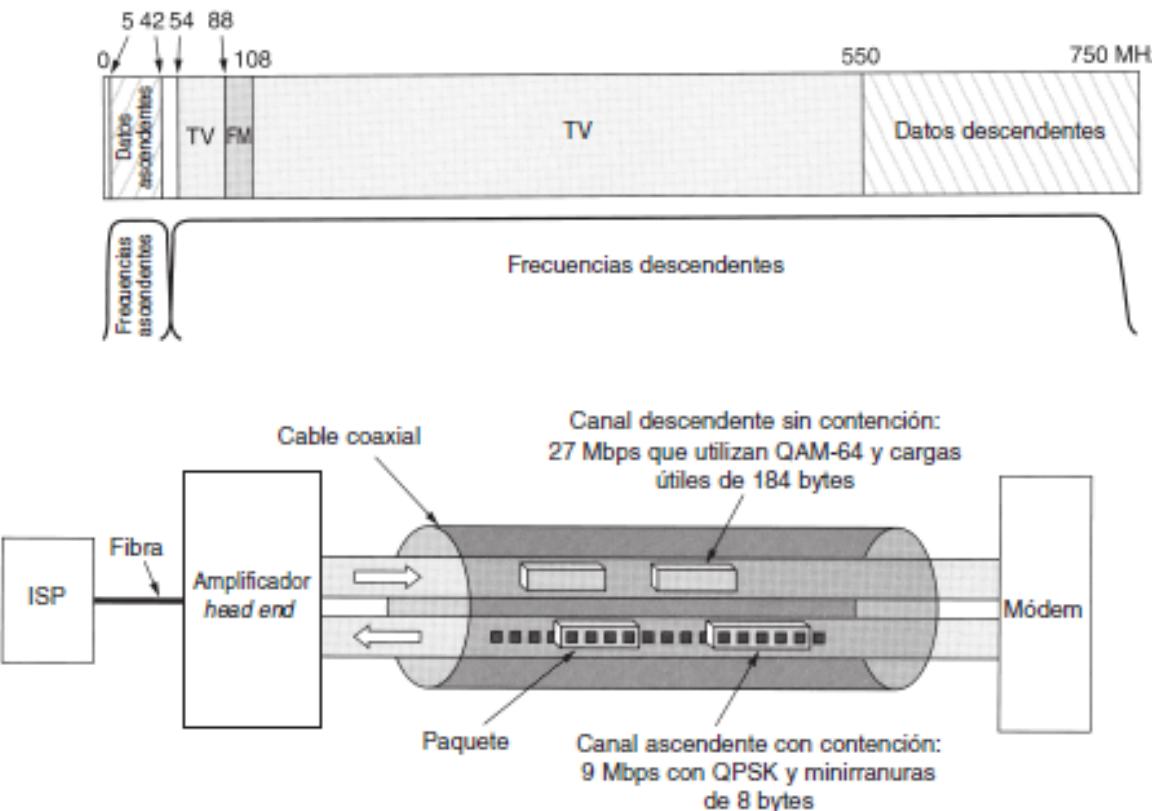
(b)

# ADSL

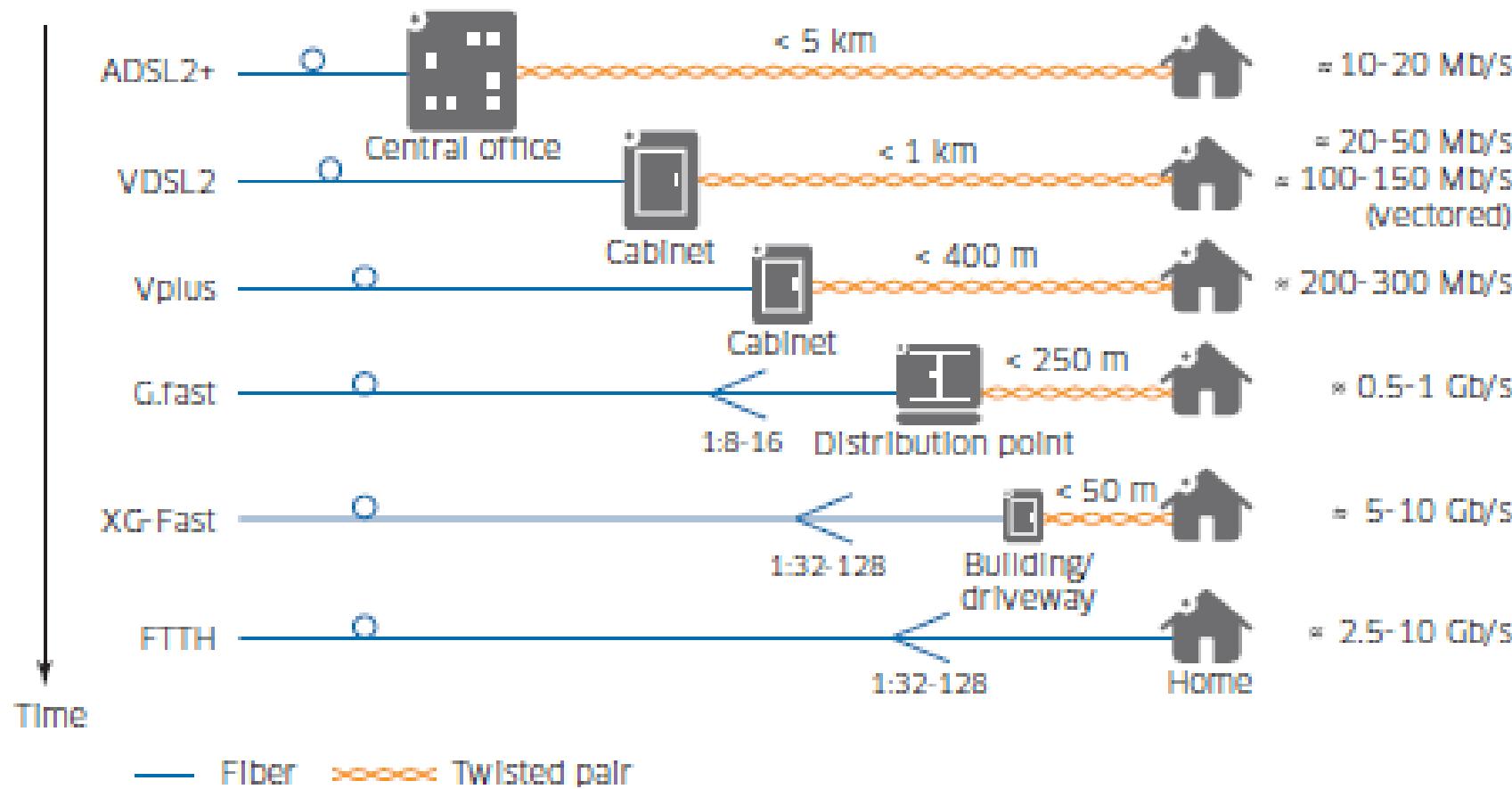


**DSLAM** Digital Subscriber Line Access Multiplexer (Multiplexor de acceso de línea de abonado digital)

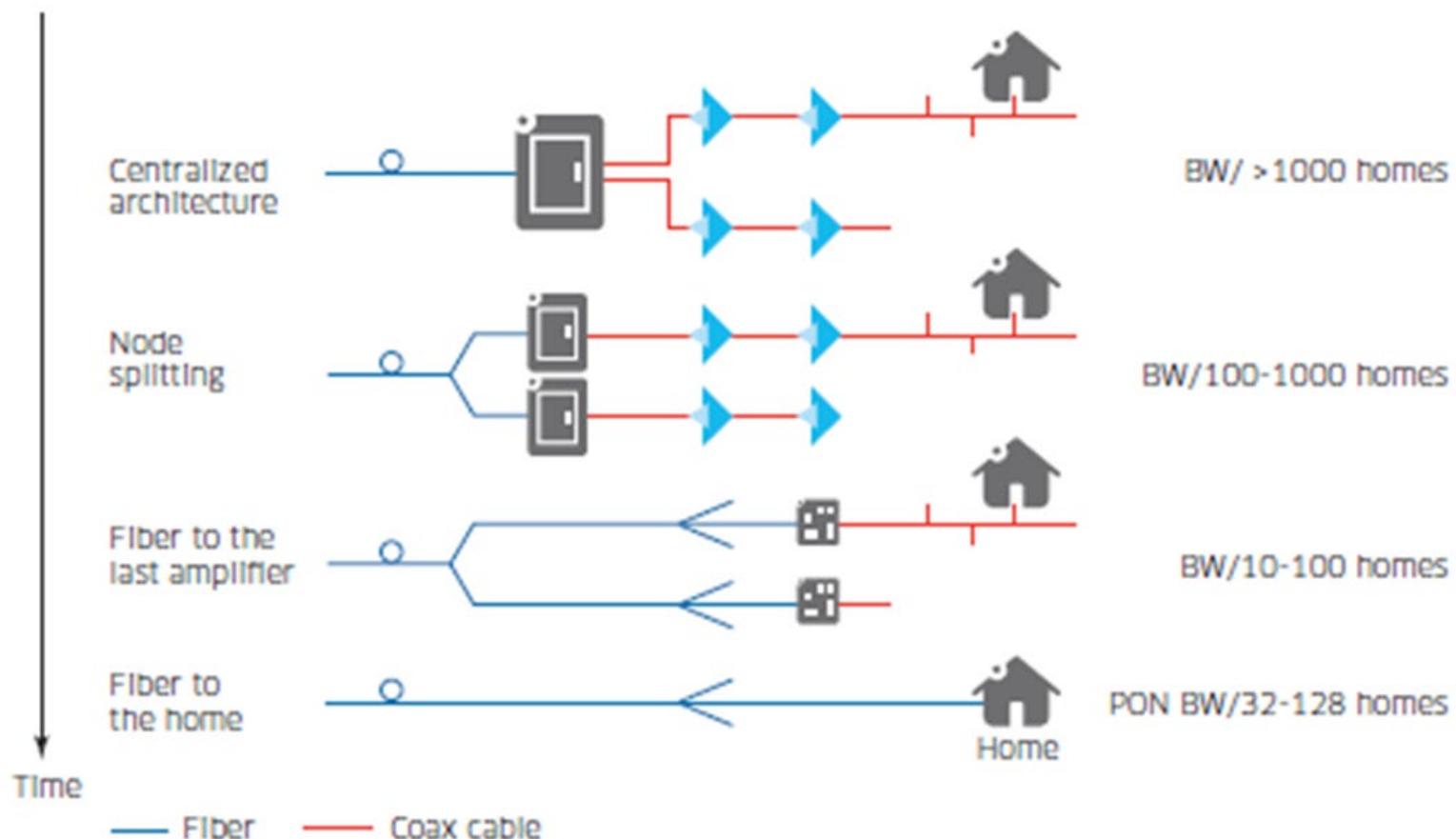
# DOCSIS : Tanenbaum , 4 edición 2003



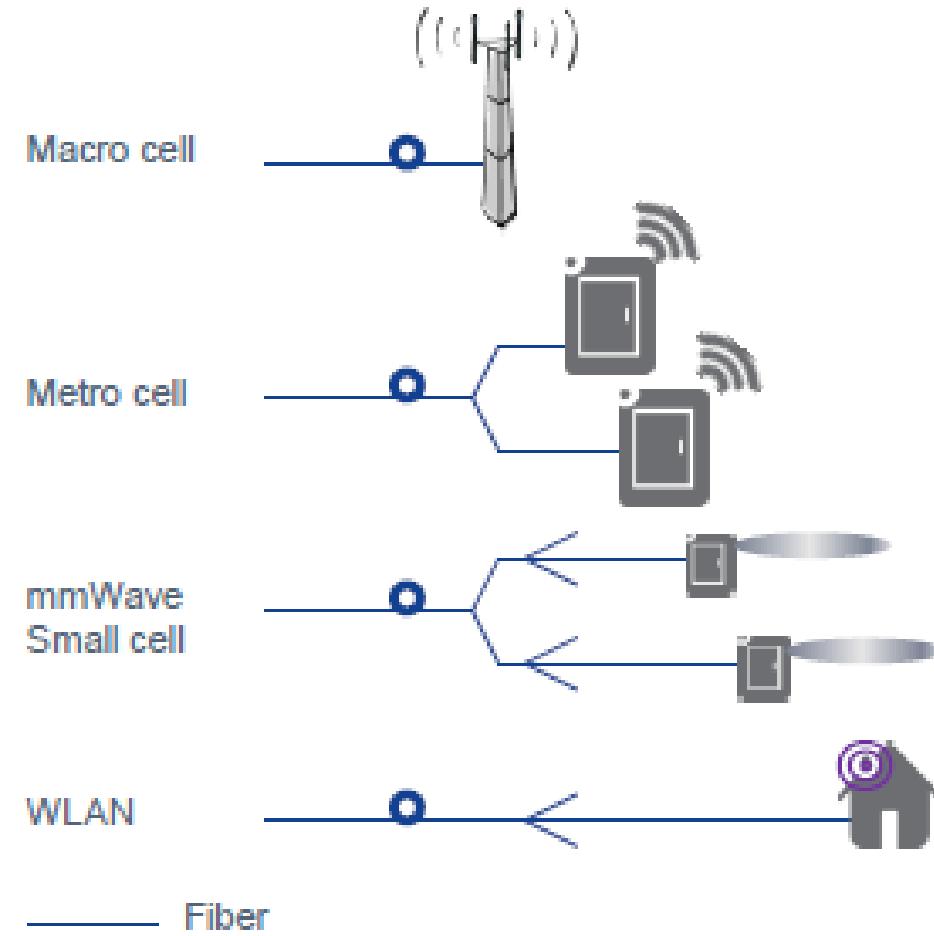
# Evolución DSL



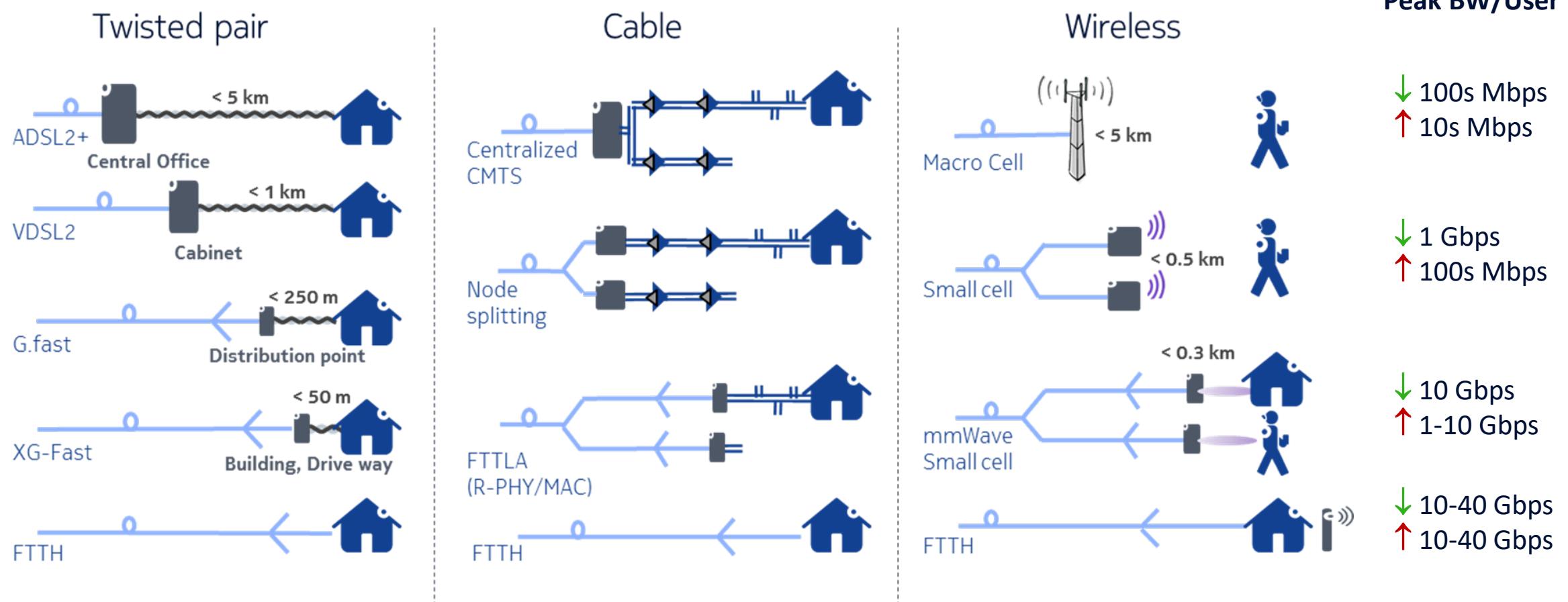
# Evolución DOCSIS



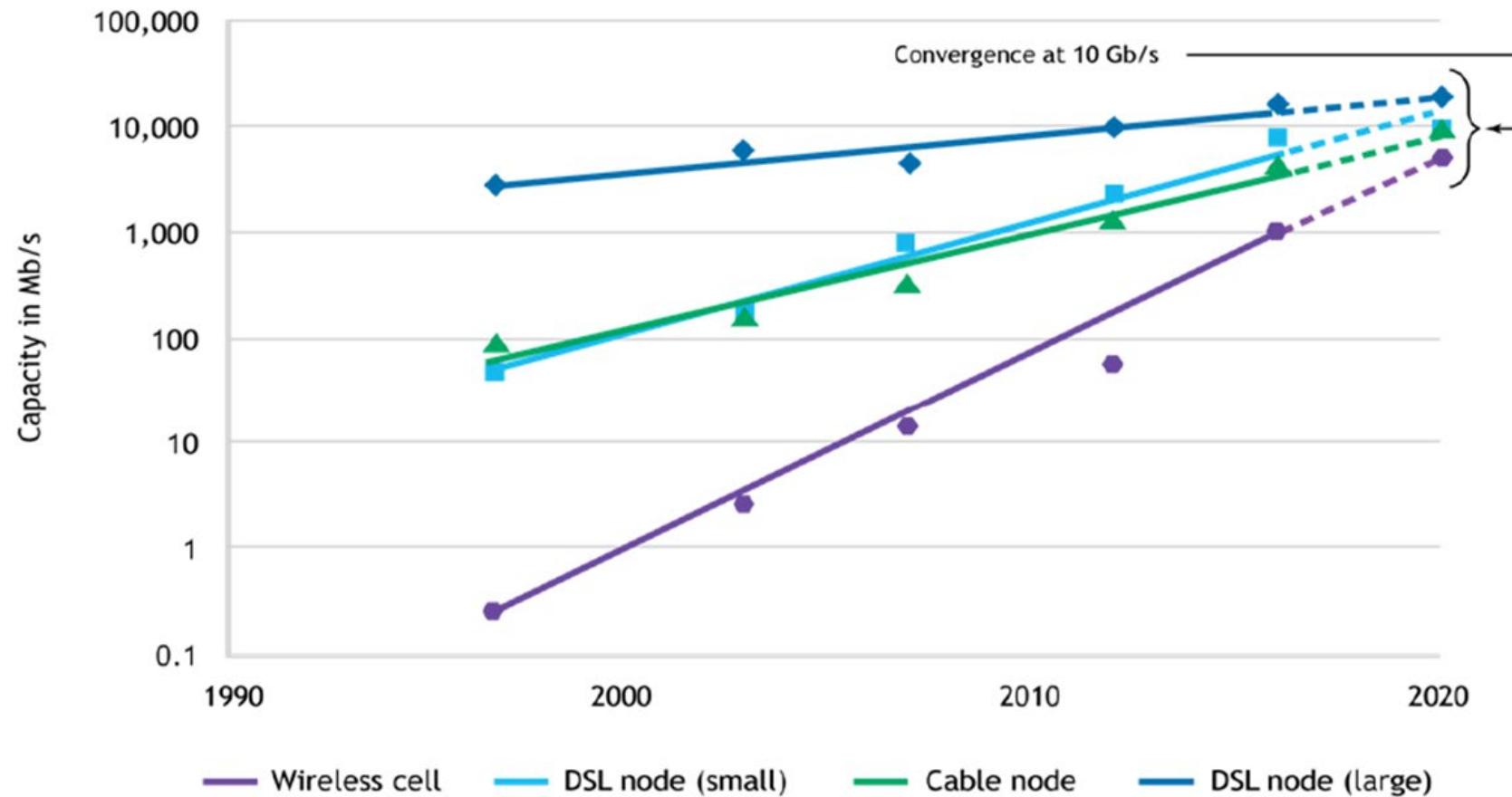
# Evolución Wireless



# Resumiendo



# Evolución de las tecnologías de acceso



Fuente : Marcus K. Weldon. 2015. The Future X Network: A Bell Labs Perspective. CRC Press, Inc., Boca Raton, FL, USA.