Redes

Tema 1: Introducción

José Carlos Cabaleiro Domínguez

Escola Técnica Superior de Enxeñería

Índice

1. Introducción a las prácticas
2. Elementos de Internet
3. Servicio orientado a conexión y sin conexión
4. Tipos de redes
5. Acceso a Internet
6. Medios de transmisión
7. Arquitectura en capas

Índice

1. Introducción a las prácticas
2. Elementos de Internet
3. Servicio orientado a conexión y sin conexión
4. Tipos de redes
5. Acceso a Internet
6. Medios de transmisión
7. Arquitectura en capas

Introducción a las prácticas

Direcciones IP: IPv4

- Formato textual: cadenas de texto
  - `char ip[]="193.110.128.200";` (16 caracteres)
- Formato binario: 4 bytes o un entero de 32 bits sin signo
  - `uint8_t ip[]={193,110,128,200};` (4 bytes)
  - `uint32_t ip=3363860161u;` (entero de 32 bits sin signo)
  - $3363860161 = 193 + 110 \times 2^8 + 128 \times 2^{16} + 200 \times 2^{24}$

struct in_addr	
uint32_t s_addr	dirección IPv4

- En C existen funciones para realizar conversiones entre los dos formatos

Introducción a las prácticas

Nombres de host

- Alternativa para identificar un ordenador en Internet
- Cadenas de caracteres:
  - `char name[]="www.elmundo.es";`

struct hostent	
`char *h_name`	el nombre “oficial” del host
`char **h_aliases`	lista de alias del host
`int h_addrtype`	el tipo de dirección
`int h_length`	la longitud en bytes de cada dirección
`char **h_addr_list`	lista de IPs del host
`char *h_addr`	la primera IP del host

- Un nombre puede tener varias direcciones IP
- Un ordenador puede tener varios nombres

Introducción a las prácticas

Puertos

- Entregar los datos a la aplicación (servicio) correcta
- Cada servicio se identifica con un número de puerto
- Entero de 16 bits, `int puerto=8000` (también sirve `unsigned short int` o `uint16_t`)

struct servent	
`char *s_name`	el nombre "oficial" del servicio
`char **s_aliases`	lista de alias del servicio
`int s_port`	el número de puerto (en el orden *big-endian*)
`char *s_proto`	el nombre del protocolo que usa este servicio

Introducción a las prácticas

Sockets

- Interfaz entre la aplicación y la capa de transporte
  - los buzones es la interfaz entre las personas (aplicaciones) y el sistema postal (capa de transporte)
- Para la comunicación entre dos procesos de diferentes ordenadores se necesita:
  - La dirección IP del ordenador que ejecuta el proceso
  - El puerto que tiene asignado el proceso en el ordenador
- Los sockets (buzones) se construyen a partir de direcciones IP y puertos
- En C se identifican por el número de socket (variable `int`)

`struct sockaddr_in`	
`sa_family_t sin_family`	el tipo de socket, `AF_INET` para IPv4
`struct in_addr sin_addr`	la dirección IPv4
`uint16_t sin_port`	el número de puerto

Introducción a las prácticas

Servidores y clientes

- Servidor: proporciona un servicio (servidor web)
  - Tanto el programa como el computador
- Cliente: solicita un servicio (cliente web)
  - Tanto el programa como el computador

Sockets

- Orientados a conexión
- No orientados a conexión o sin conexión

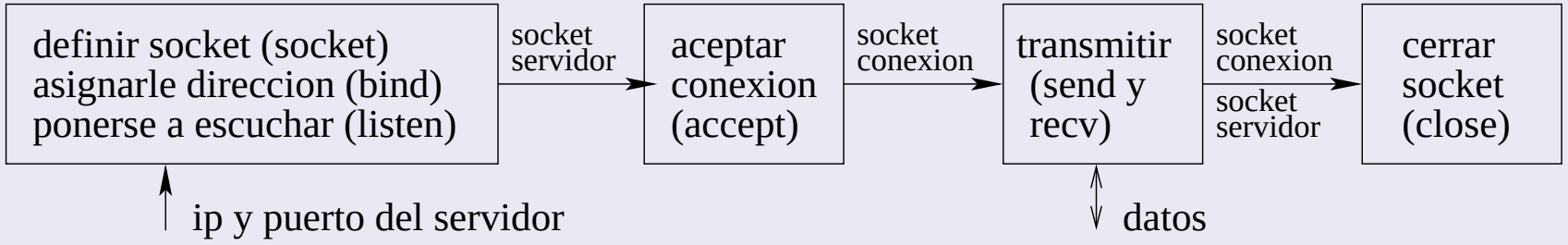
Introducción a las prácticas

Sockets orientados a conexión

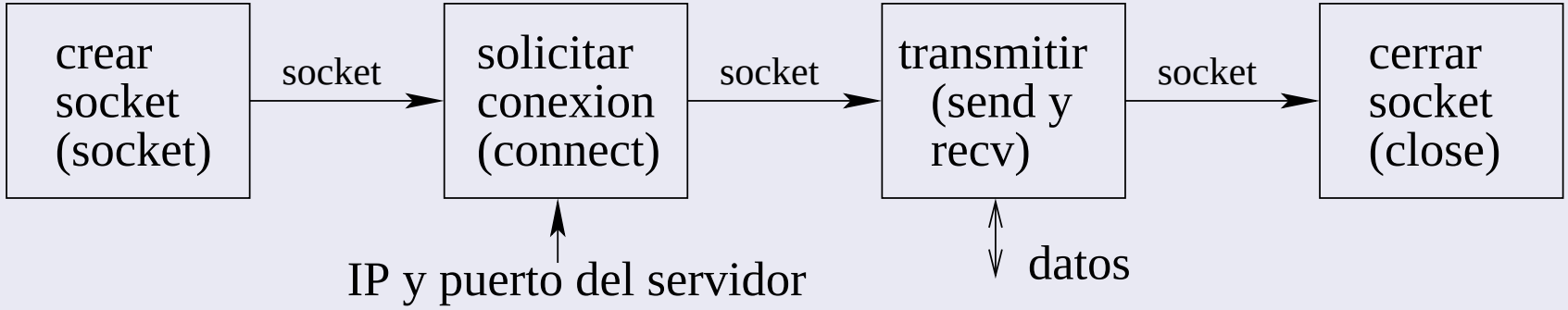
- Cliente: solicita la conexión
- Servidor: acepta la conexión
- Pasos:
  - Crear los sockets
  - Solicitar una conexión (cliente) o aceptarla (servidor)
  - Enviar y recibir datos
  - Cerrar los sockets

Introducción a las prácticas

Servidor



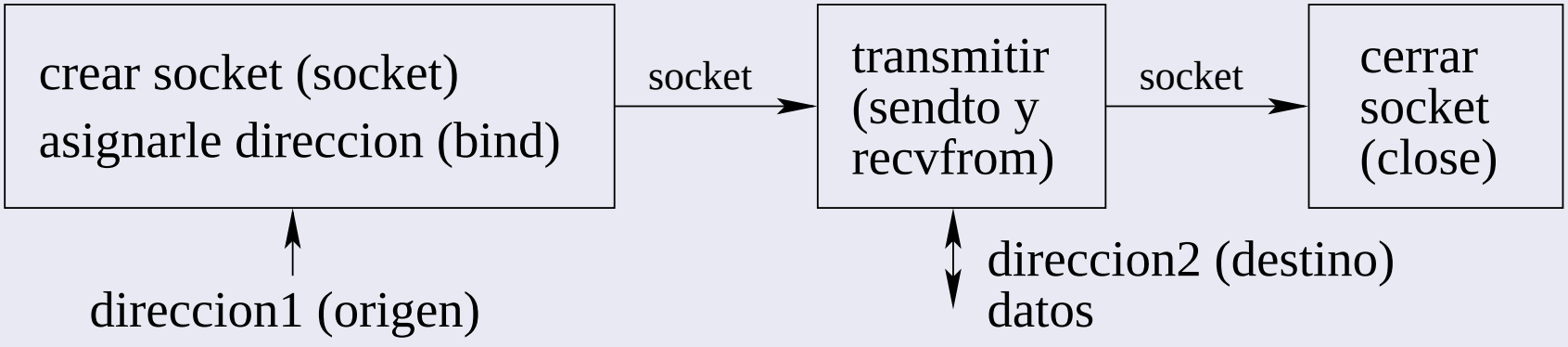
Cliente



Introducción a las prácticas

Sockets no orientados a conexión

- Se pueden enviar datos en cualquier momento
- No es necesario establecer una conexión previa
- Cliente y servidor funcionan de forma simétrica
- Pasos:
  - Crear el socket y se asigna la IP y puerto origen (destino)
  - Enviar y recibir datos indicando la IP y puerto destino (origen)
  - Cerrar los sockets



Índice

1. Introducción a las prácticas
2. Elementos de Internet
3. Servicio orientado a conexión y sin conexión
4. Tipos de redes
5. Acceso a Internet
6. Medios de transmisión
7. Arquitectura en capas

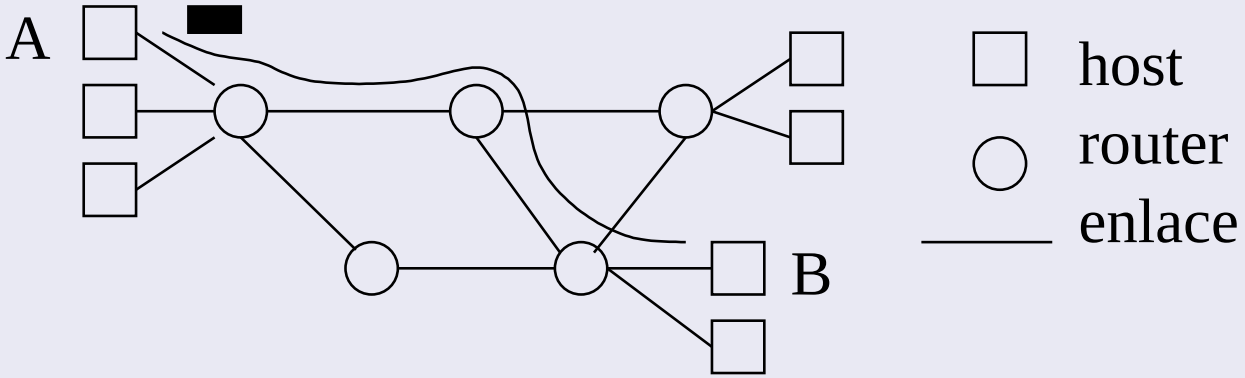
Elementos de Internet

¿Cómo funciona Internet?

Elementos de Internet

Hardware

- *Hosts*: sistemas terminales, origen y destino de las transmisiones
- *Enlaces*: medios físicos por los que se realizan las transmisiones
- *Routers*: dispositivos que interconectan enlaces



Elementos de Internet

Software: protocolos

- Básicos: TCP/IP y algunos como UDP, ICMP, etc.
- De aplicación: como HTTP, SMTP, etc.

Comercial: proveedores de servicios de Internet (ISP)

- Proveedores de baja escala (residenciales): acceso a Internet a usuarios
- Proveedores de alta escala (nacionales o internacionales):
  - Redes troncales que interconectan a los proveedores de baja escala
  - Líneas de larga distancia

Índice

1. Introducción a las prácticas
2. Elementos de Internet
3. Servicio orientado a conexión y sin conexión
4. Tipos de redes
5. Acceso a Internet
6. Medios de transmisión
7. Arquitectura en capas

Servicio orientado a conexión

Fases

- Establecimiento de la conexión
  - El cliente solicita una conexión
  - Se fijan parámetros
  - Ambos extremos se preparan para la transmisión
- Transmisión de datos
- Desconexión: terminan la transmisión y se liberan los recursos

Servicio orientado a conexión

Características

- *Segmentación*: TCP recoge datos que la aplicación escribe en el socket y forma paquetes (MSS)
- *Transferencia fiable*: el receptor envía confirmaciones (ACK). Si el emisor no recibe ACK de un paquete, se retransmite
- *Control de flujo*: permite que receptor controle la tasa de envío del emisor.
  - TCP tiene un mecanismo para que el receptor le indique al emisor la tasa de datos que puede aceptar
- *Control de congestión*: permite que la tasa de envío del emisor se ajuste a las capacidades de la red

Servicio no orientado a conexión

Servicio sin conexión

- No hay fase de establecimiento de conexión
- No hay confirmaciones: el emisor desconoce si el paquete llegó al destino
- No hay control de flujo ni control de congestión

$\Longrightarrow$ transmisión más rápida, aunque menos fiable

En Internet: UDP

- TCP: cuando se necesite fiabilidad, telnet, FTP, SMTP, HTTP, etc.
- UDP: velocidad y no importa la pérdida de paquetes, telefonía IP, videoconferencia

Índice

1. Introducción a las prácticas
2. Elementos de Internet
3. Servicio orientado a conexión y sin conexión
4. Tipos de redes
5. Acceso a Internet
6. Medios de transmisión
7. Arquitectura en capas

Tipos de redes

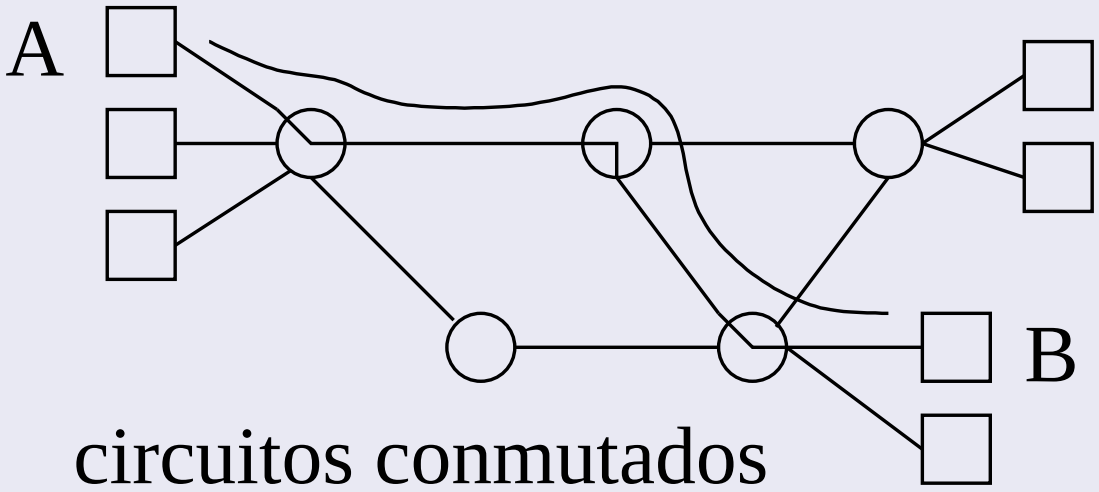
Desde el punto de vista hardware

- *Conmutación*
  - Circuitos. Pueden ser sin multiplexación, FDM, TDM
  - Paquetes. Pueden ser de datagramas, circuitos virtuales
- *Difusión*. Redes Ethernet, redes inalámbricas, etc.
  - Todos los hosts reciben las transmisiones
  - Solo el destinatario procesa la transmisión

Tipos de redes

Redes de conmutación de circuitos

- Fase de conexión en la que se reservan recursos hardware
  - No pueden ser usados por otra transmisión
  - Se establece la ruta que van a seguir los datos
  - Ranuras temporales o bandas de frecuencia
- Transmisión de datos
- Fase de desconexión: se liberan todos los recursos



Tipos de redes

Redes de conmutación de circuitos

- Sin multiplexación: solo una transmisión por cada enlace de cada vez
- Con multiplexación: se reparte la capacidad del enlace entre varias transmisiones
  - División en frecuencia (FDM): bandas de frecuencia
  - División en el tiempo (TDM): ranuras temporales

Las redes de conmutación de circuitos son derrochadoras de recursos: reservados aunque la transmisión no los use

Redes de conmutación de paquetes

Características

- No se reservan recursos para cada conexión: se comparten y se asignan bajo demanda
- Trabajan con paquetes $\Longrightarrow$ segmentación
- Los paquetes contienen una cabecera con información de control (para llegar a su destino, ACK, etc.)
- En Internet, los routers funcionan como conmutadores de paquetes (en general *store-and-forward*)
  - Reciben el paquete completo antes de reenviarlo
  - Se procesa y se almacenan en una cola
  - Si la cola está llena, se descarta el paquete

Redes de conmutación de paquetes

Retardo en redes de conmutación de paquetes

- De procesamiento (examinar la cabecera y dirigir el paquete a la salida)
- De espera en la cola, proporcional a la carga de la red
- De transmisión, proporcional al tamaño del paquete (longitud del paquete/tasa de transmisión)
- De propagación, depende del tipo y longitud del enlace (longitud del enlace/velocidad de propagación)

Retardo total

$$d_{\text{total}} = d_{\text{procesamiento}} + d_{\text{cola}} + d_{\text{transmisión}} + d_{\text{propagación}}$$

Retardo de extremo a extremo

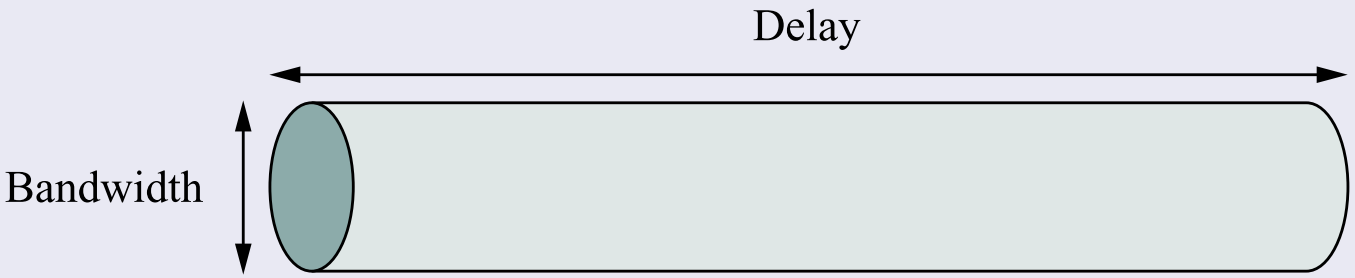
$$d = \sum_{i=1}^{\text{Nsaltos}} (d_{\text{procesamiento}_i} + d_{\text{cola}_i} + d_{\text{transmisión}_i} + d_{\text{propagación}_i})$$

Redes de conmutación de paquetes

Producto retardo × ancho de banda

- Capacidad del enlace = retardo × ancho de banda
- Máximo número de bits que podría estar en tránsito en un momento dado
- Aprovechamiento del enlace: número de bits que el emisor debe transmitir antes de que el primer bit llegue al receptor
- Si el emisor espera ACKs, se considera el RTT (2 × retardo) en lugar del retardo

Enlace visto como una tubería



- Número de bits que “caben” en el enlace

Redes de conmutación de paquetes

Segmentación

- Tiempo de transmisión más corto
- No se satura la red con mensajes grandes, se pueden intercalar otras transmisiones
- Si hay errores, solo se transmiten los paquetes con errores

Tipos de redes de conmutación de paquetes

- Datagramas: no orientadas a conexión y encaminamiento en función del destino
- Circuitos virtuales: orientadas a conexión y encaminamiento en función del número de circuito virtual

Redes de conmutación de paquetes

Redes de datagramas

- Cada paquete incluye en la cabecera la IP destino
- Reenvío: el router examina la cabecera y lo coloca en la salida más apropiada (tabla de reenvío)
- No mantienen información de estado: una secuencia de paquetes se encaminan de forma independiente

Redes de circuitos virtuales

- Se establece la conexión planificando una ruta al destino: un número de circuito virtual (CV)
- A cada paquete se le escribe el número de CV: los routers lo usan para el reenvío
- Los routers mantienen información de estado (tabla de circuitos virtuales)

Índice

1. Introducción a las prácticas
2. Elementos de Internet
3. Servicio orientado a conexión y sin conexión
4. Tipos de redes
5. Acceso a Internet
6. Medios de transmisión
7. Arquitectura en capas

Acceso a Internet

Formas

- Acceso residencial
  - Módem telefónico (56 kbps)
  - ADSL (Línea de subscripción digital asimétrica)
  - Cable HFC (híbrido fibra-coaxial)
  - FTTH (Fibra hasta el domicilio)
- Acceso empresarial y doméstico: Ethernet y WiFi
- Acceso móvil: WiFi, 3G, 4G, LTE (*Long Term Evolution*)...

Acceso a Internet

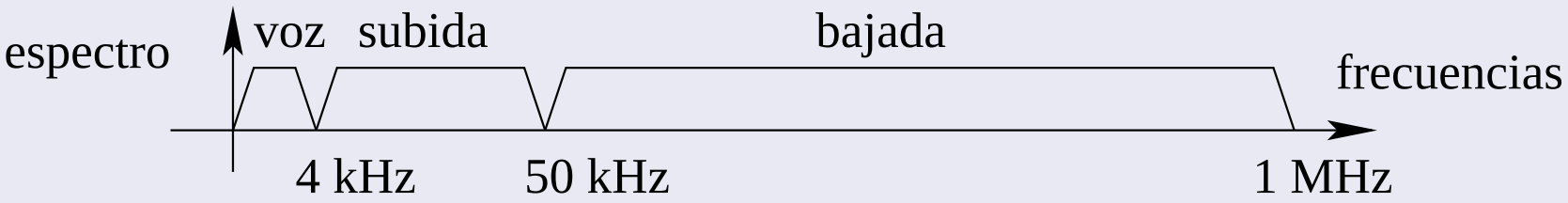
Módem

- Usa la línea telefónica como si fuese una llamada de voz
- Fases:
  - Establecimiento de la conexión: llama al número telefónico del ISP
  - Modulación: convierte la señal digital en una señal modulada
  - Demodulación: el receptor realiza la operación inversa
- Problema: ancho de banda de frecuencias muy estrecho (4 KHz) $\Longrightarrow$ velocidad de transmisión de 56 kbps

Acceso a Internet

DSL

- Aprovecha todo el ancho de banda de frecuencias del cable telefónico (1 MHz)
- FDM en tres canales independientes
  - Voz telefónica
  - Canal de subida a Internet
  - Canal de bajada a Internet



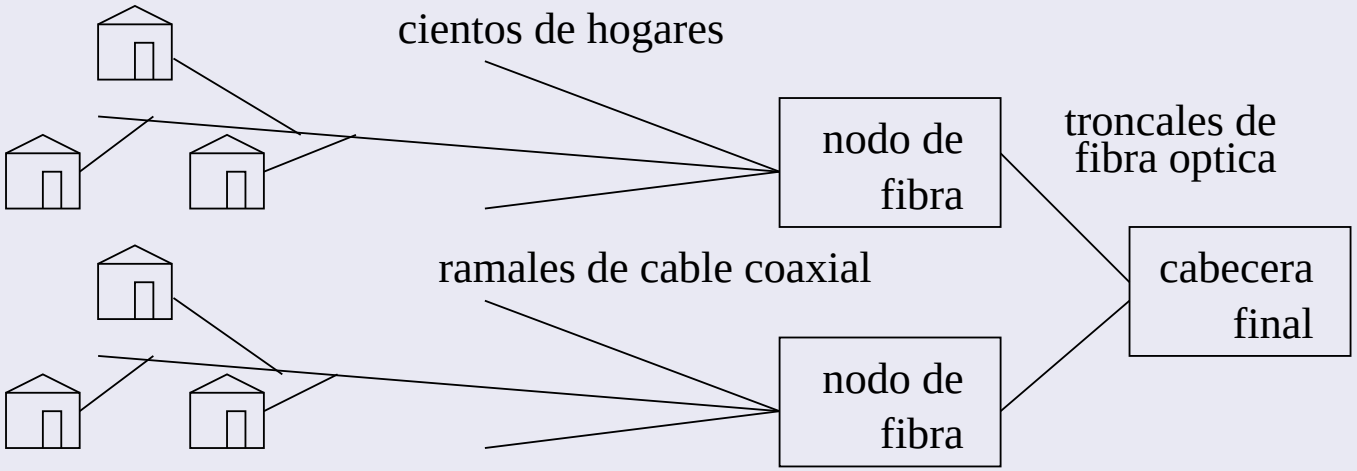
- Ancho de banda de frecuencias elevado $\Longrightarrow$ velocidad de transmisión hasta 30 Mbps (VDSL2)

Acceso a Internet

Cable HFC (híbrido fibra y coaxial)

- Cabecera final: centraliza las transmisiones de los abonados a Internet
- Líneas troncales de fibra óptica: conectan la cabecera con los nodos de fibra
- Ramales de cable coaxial: dan servicio de TV, teléfono e Internet a los usuarios

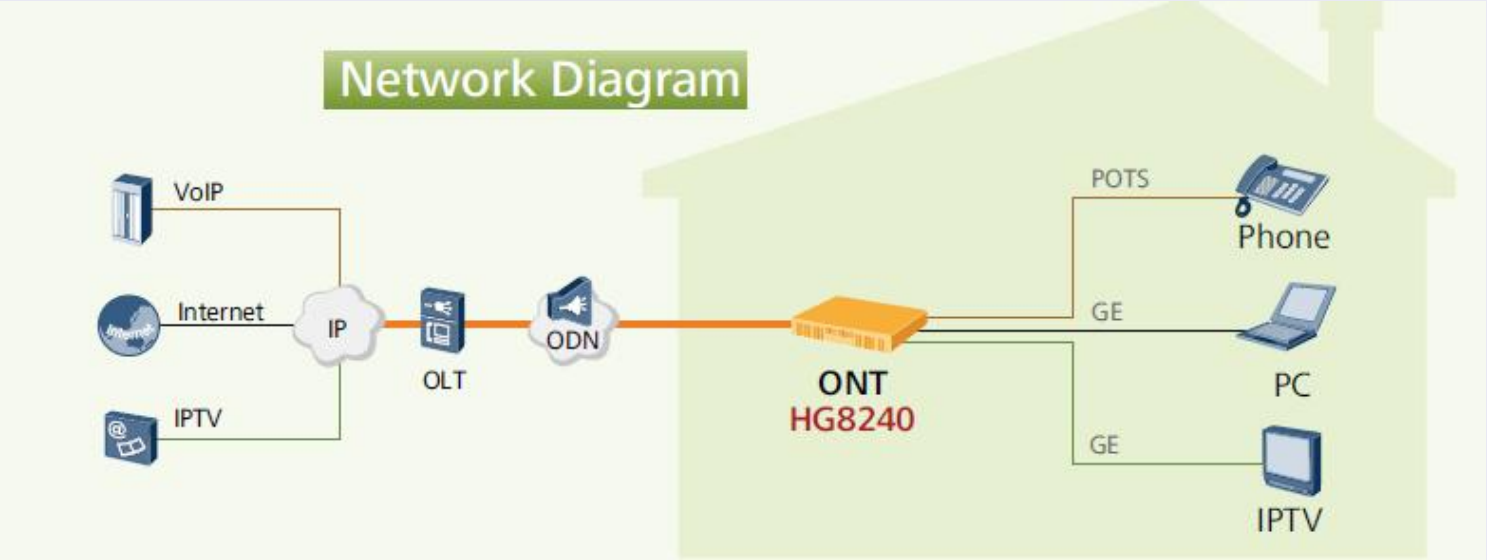
Ejemplo:



Acceso a Internet

FTTH (*Fiber To The Home*)

- Fibra para distribución de servicios avanzados: *Triple Play*
- OLT (*Optical Line Terminal*): punto final que viene del ISP
- ODN (*Optical Distribution Network*): desde el OLT a los usuarios
- ONT (*Optical Network Termination*): conversión de señales ópticas $\leftrightarrow$ eléctricas



Acceso a Internet

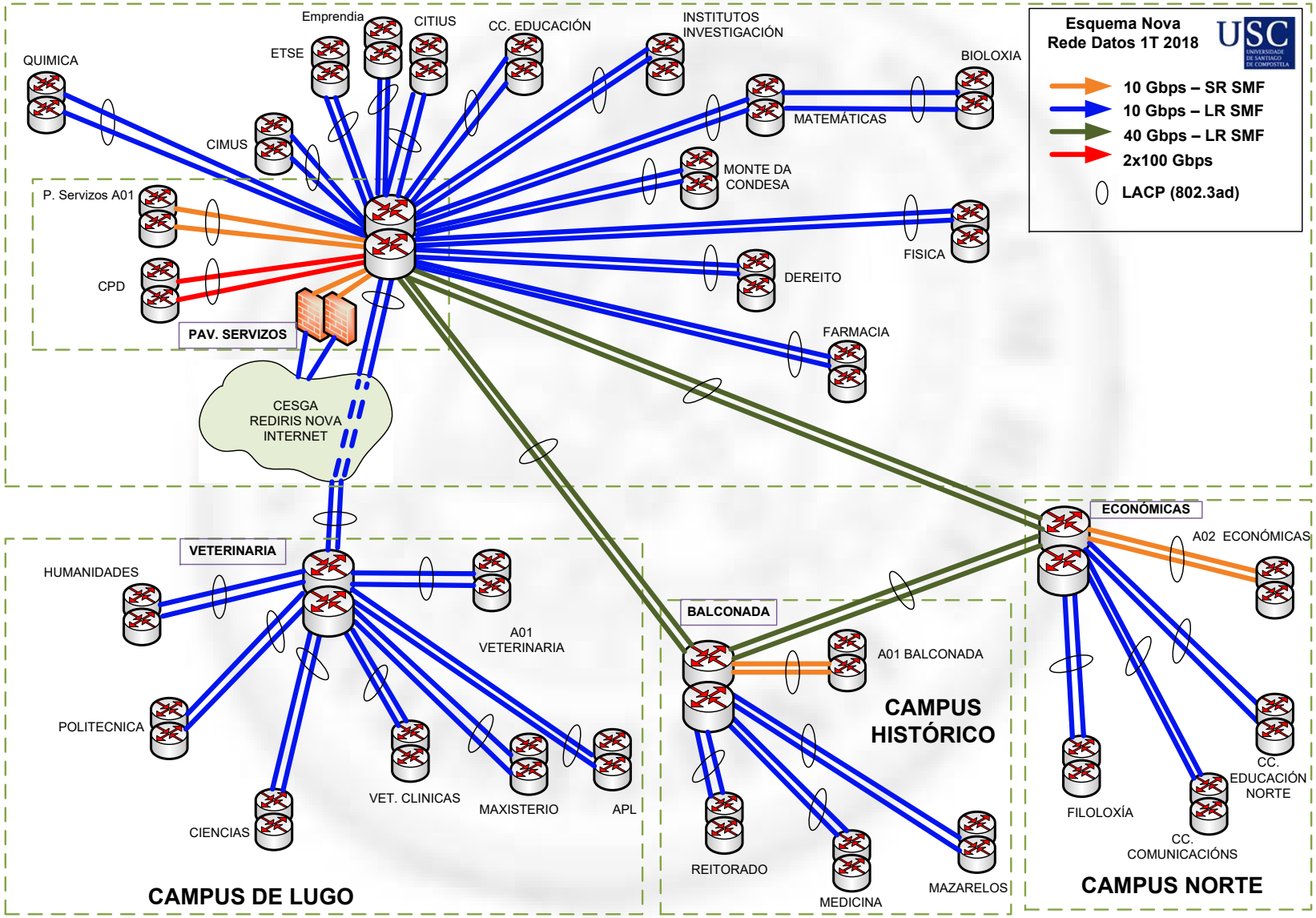
Acceso empresarial

- Mediante una LAN (tipo Ethernet) conectada a un router y a un ISP con enlace dedicado (aparte de la telefónica)

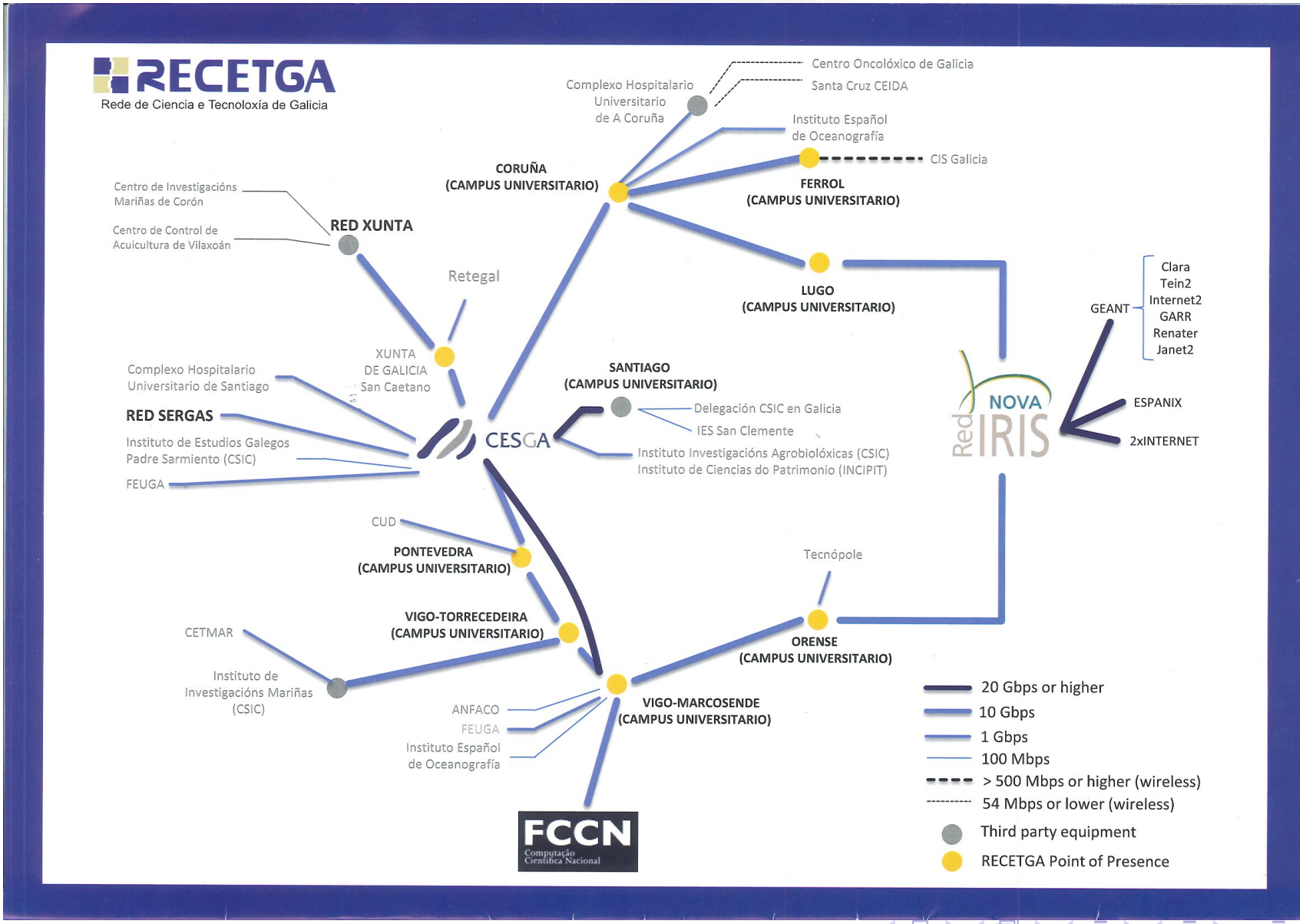
Red de la USC

- Tres nodos troncales en Santiago y uno en Lugo unidos a través de RedIRIS Nova y el CESGA
  - Nodos troncales de Santiago unidos con enlaces dobles de 40 GE
  - Enlaces dobles desde los nodos troncales a los nodos de distribución de 10 GE
  - Enlaces a 10 GE entre los nodos de distribución y los nodos de acceso (conmutadores a 100 Mbps o 1 Gbps)
- Acceso a Internet mediante un nodo en el Cesga que enlaza con RedIris (gestiona la red pública)

Red de la USC


Esquema Nova
Rede Datos 1T 2018
USC
10 Gbps – SR SMF
10 Gbps – LR SMF
40 Gbps – LR SMF
2x100 Gbps
LACP (802.3ad)
QUIMICA
ETSE
Emprendia
CITIUS
CC. EDUCACIÓN
INSTITUTOS INVESTIGACIÓN
BIOLOXIA
MATEMÁTICAS
CIMUS
MONTE DA CONDESA
P. Servizos A01
FISICA
CPD
DEREITO
PAV. SERVIZOS
FARMACIA
CESGA
REDIRIS NOVA
INTERNET
ECONÓMICAS
A02 ECONÓMICAS
VETERINARIA
HUMANIDADES
A01 VETERINARIA
BALCONADA
A01 BALCONADA
CAMPUS HISTÓRICO
POLITECNICA
VET. CLINICAS
MAXISTERIO
APL
CIENCIAS
REITORADO
MEDICINA
MAZARELOS
FILOLOXÍA
CC. EDUCACIÓN NORTE
CC. COMUNICACIÓNS
CAMPUS DE LUGO
CAMPUS NORTE

RECETGA


RECETGA
Rede de Ciencia e Tecnoloxía de Galicia
Complexo Hospitalario Universitario de A Coruña
Centro Oncolóxico de Galicia
Santa Cruz CEIDA
Instituto Español de Oceanografía
CIS Galicia
CORUÑA (CAMPUS UNIVERSITARIO)
FERROL (CAMPUS UNIVERSITARIO)
Centro de Investigacións Mariñas de Corón
Centro de Control de Acuicultura de Vilaxoán
RED XUNTA
Retegal
LUGO (CAMPUS UNIVERSITARIO)
GEANT
Clara
Tein2
Internet2
GARR
Renater
Janet2
XUNTA DE GALICIA San Caetano
Complexo Hospitalario Universitario de Santiago
RED SERGAS
Instituto de Estudios Galegos Padre Sarmiento (CSIC)
FEUGA
CESGA
SANTIAGO (CAMPUS UNIVERSITARIO)
Delegación CSIC en Galicia
IES San Clemente
Instituto Investigacións Agrobiolóxicas (CSIC)
Instituto de Ciencias do Patrimonio (INCIPIT)
Red NOVA IRIS
ESPANIX
2xINTERNET
CUD
PONTEVEDRA (CAMPUS UNIVERSITARIO)
Tecnópole
VIGO-TORRECEDEIRA (CAMPUS UNIVERSITARIO)
CETMAR
Instituto de Investigacións Mariñas (CSIC)
ORENSE (CAMPUS UNIVERSITARIO)
ANFACO
FEUGA
Instituto Español de Oceanografía
VIGO-MARCOSENDE (CAMPUS UNIVERSITARIO)
FCCN Computação Científica Nacional
20 Gbps or higher
10 Gbps
1 Gbps
100 Mbps
> 500 Mbps or higher (wireless)
54 Mbps or lower (wireless)
Third party equipment
RECETGA Point of Presence

Índice

1. Introducción a las prácticas
2. Elementos de Internet
3. Servicio orientado a conexión y sin conexión
4. Tipos de redes
5. Acceso a Internet
6. Medios de transmisión
7. Arquitectura en capas

Medios de transmisión

Medios guiados

- Cable de cobre de par trenzado
- Cable coaxial
- Fibra óptica

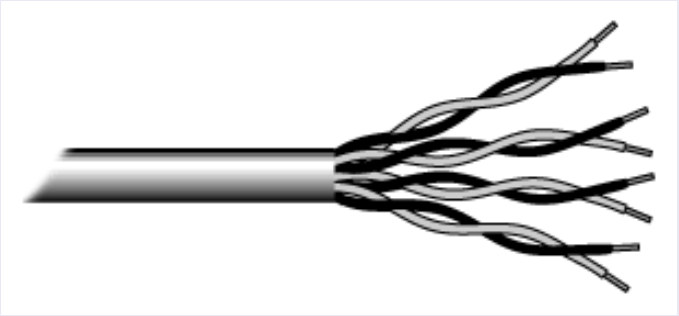
Medios no guiados

- La atmósfera y el espacio
  - Canales de radio terrestre
  - Canales de radio vía satélite

Medios de transmisión

Cable de cobre de par trenzado

- Tipo telefónico: dos hilos de cobre trenzados
- Para LAN se suele usar un cable de 4 pares trenzados

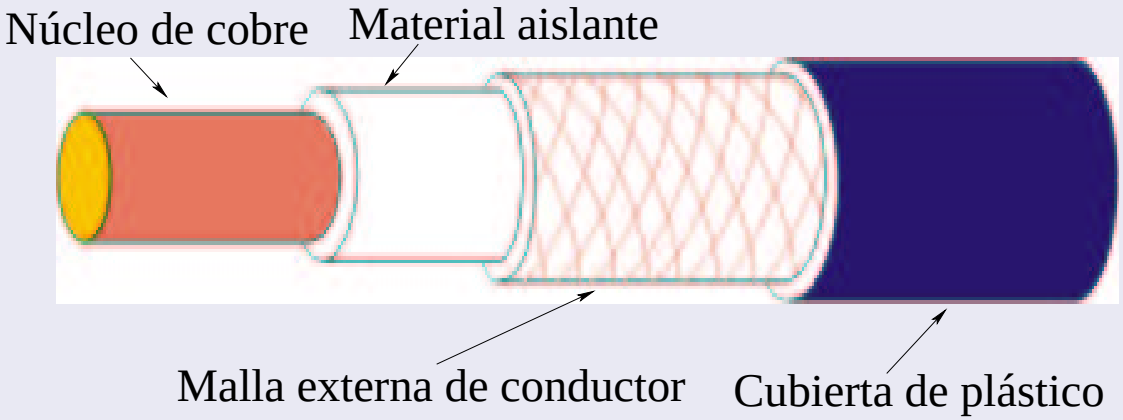


- Tipos: UTP de diferentes categorías (5 y 6a las más comunes) y STP
  - UTP categoría 6a
    - Permite velocidades de 10 Gbps hasta 100 m

Medios de transmisión

Cable coaxial

- Dos conductores concéntricos con aislante entre ellos

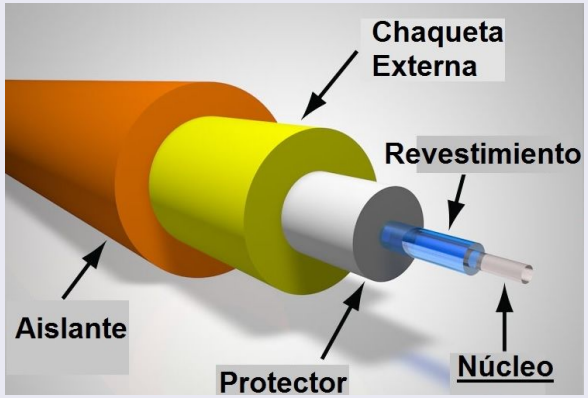


- Evita pérdidas de energía por radiación
- Dos tipos
  - 50 Ω para señales sin modular
  - 75 Ω para señales de banda ancha en las redes de HFC

Medios de transmisión

Fibra óptica

- Transmiten luz $\Longrightarrow$ evita pérdidas por radiación e interferencias



- Materiales transparentes con baja atenuación $\Longrightarrow$ 100 Km sin repetidores
- Más difíciles de instalar y dispositivos más costosos

Medios de transmisión

Fibra óptica

- Fibra multimodo
  - La luz se propaga rebotando en las paredes del núcleo
  - Utilizada para redes de conexión locales, centros de datos de edificio a edificio y para FTTH
- Fibra monomodo
  - Se propaga en línea recta $\Longrightarrow$ mayor distancia
  - Más costosa
- Designación OC-n: velocidad de enlace $n \times 51{,}8$ Mbps
  - Ejemplos: OC-3, OC-12, OC-48, OC-192, OC-768

Índice

1. Introducción a las prácticas
2. Elementos de Internet
3. Servicio orientado a conexión y sin conexión
4. Tipos de redes
5. Acceso a Internet
6. Medios de transmisión
7. Arquitectura en capas

Arquitectura en capas

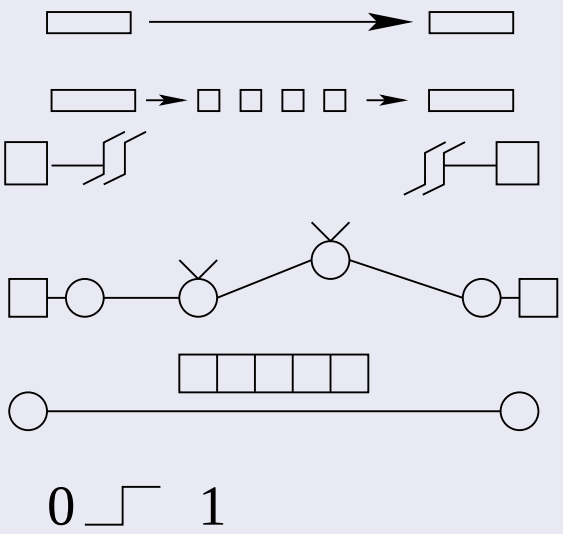
Arquitectura en capas

- Facilita el diseño de protocolos de comunicación
- Se divide la comunicación en tareas independientes $\Longrightarrow$ cada tarea en una capa
- Las capas superiores usan los servicios de las inferiores
- Modularidad: se deben respetar las especificaciones de cada capa
- Hay varias propuestas: modelo OSI, arquitectura TCP/IP, etc.

Arquitectura en capas

La arquitectura TCP/IP

APLICACION	genera mensajes
TRANSPORTE	prepara los mensajes para transmitir
RED	busca rutas
ENLACE	transmite bloques de datos a traves de un enlace
FISICA	detalles a nivel de bit



La arquitectura TCP/IP

Capa de aplicación

- Se localizan los procesos que se comunican entre sí mediante mensajes
- Protocolos de aplicación: HTTP, SMTP, etc.

Capa de transporte

- Prepara los mensajes para que se puedan transmitir fuera del computador
- TCP recoge los datos de la aplicación origen y forma paquetes y, en destino, TCP comprueba que todo llega OK y reensambla el mensaje

La arquitectura TCP/IP

Capa de red

- Encargada de hacer llegar los paquetes de un host a otro
- Protocolos de encaminamiento: determinan rutas
- Elementos principales: routers

La arquitectura TCP/IP

Capa de enlace

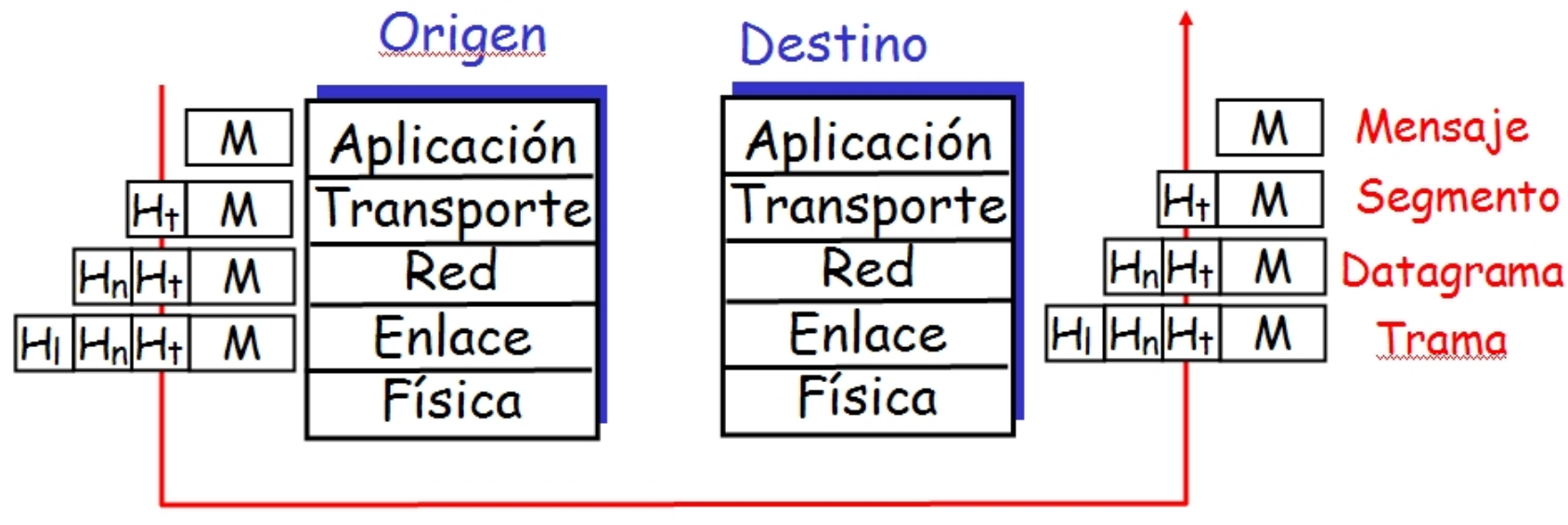
- Se encarga de los detalles de bajo nivel de la transmisión de cada paquete entre los dos extremos de un enlace
- Protocolos dependientes de la tecnología de red (difusión, circuitos virtuales, conmutación de circuitos, conmutación de paquetes, etc.)

Capa física

- Trabaja a nivel de bits: convierte bits en señales eléctricas
- Define las características físicas del medio de transmisión

Arquitectura en capas

La arquitectura TCP/IP

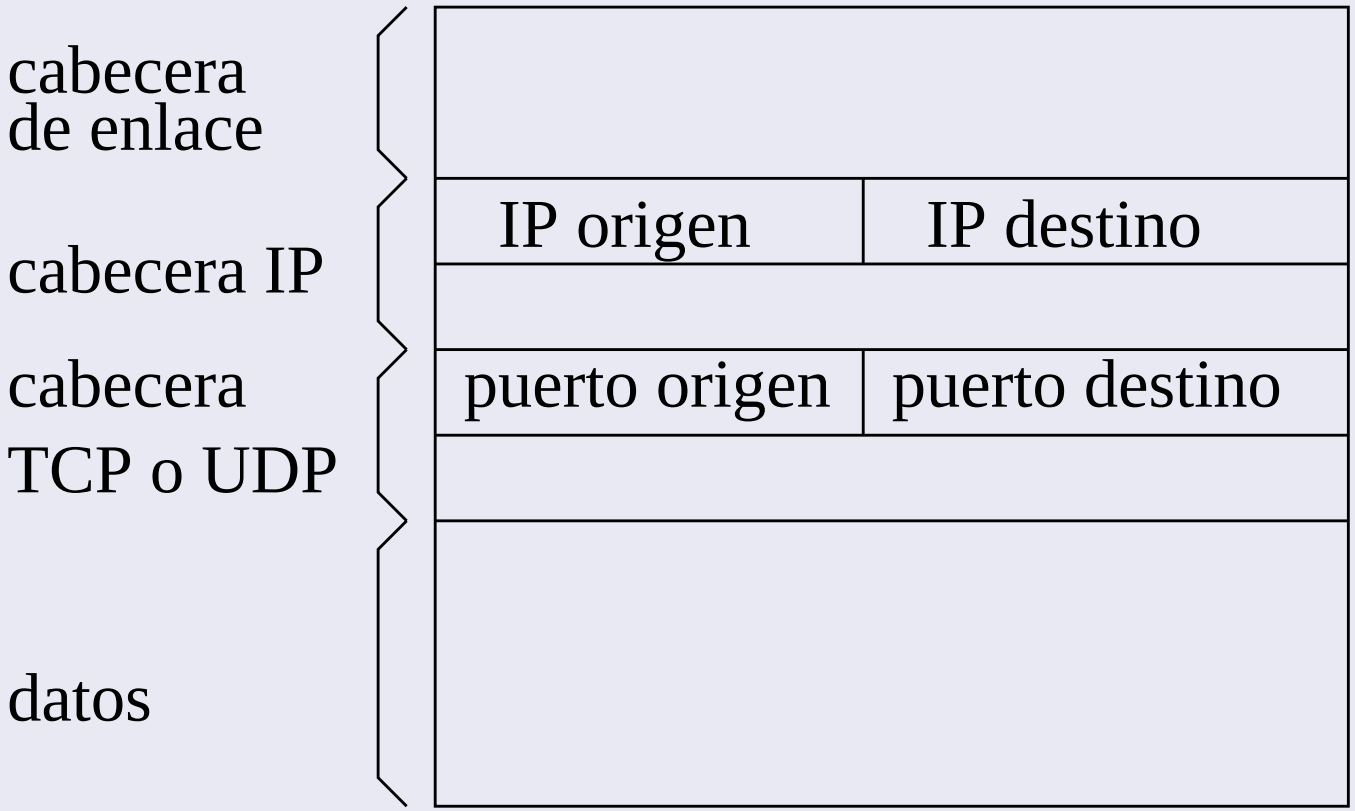


- Cabeceras: información de control que se añade al mensaje
- TCP/IP: en la capa de transporte TCP y UDP y, en la capa de red, IP

Arquitectura en capas

La arquitectura TCP/IP

Esquema de los paquetes en TCP/IP



Redes

Tema 2: Capa de aplicación

José Carlos Cabaleiro Domínguez

Escola Técnica Superior de Enxeñería

Índice

1. Introducción
2. Protocolo de transferencia de hipertexto
3. Protocolo de transferencia de archivos
4. Protocolos de correo electrónico
5. DNS: Servicio de nombres de dominio
6. Distribución de contenidos

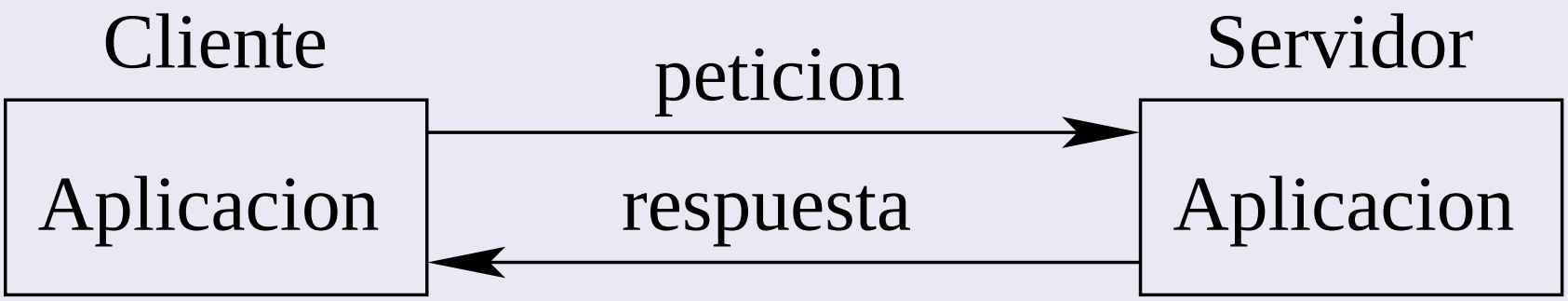
Índice

1. Introducción
2. Protocolo de transferencia de hipertexto
3. Protocolo de transferencia de archivos
4. Protocolos de correo electrónico
5. DNS: Servicio de nombres de dominio
6. Distribución de contenidos

Introducción

Capa de aplicación

- Se ocupa de la comunicación entre procesos



Capa de aplicación

Protocolos de comunicación

- Necesarios para la comprensión de los mensajes entre los procesos
- Facilitan la programación de las funciones de envío y recepción

Deben especificar

- Tipo de mensajes que se intercambian: petición, respuesta, etc.
- Reglas que especifican cuándo y cómo se envían los mensajes
- Sintaxis del mensaje: campos
- Semántica de cada campo

Capa de aplicación

Servicios que necesita la aplicación de red

- Los protocolos de la capa de aplicación usan protocolos básicos de la capa de transporte
- Transferencia fiable de datos o no fiable
  - TCP proporciona un servicio fiable y orientado a conexión
  - UDP un servicio no fiable más sencillo pero rápido y no orientado a conexión
- Ancho de banda
- Temporización

Capa de aplicación

Protocolos a tratar

TCP	HTTP (web) SMTP, POP3 e IMAP (correo) FTP (ficheros)
UDP	DNS (traducciones)

Agente de usuario

- Interfaz entre el usuario y la aplicación (navegador, gestor de correo, etc.)

Índice

1. Introducción
2. Protocolo de transferencia de hipertexto
3. Protocolo de transferencia de archivos
4. Protocolos de correo electrónico
5. DNS: Servicio de nombres de dominio
6. Distribución de contenidos

HTTP

Protocolo de transferencia de hipertexto

- Protocolo que define la comunicación entre un servidor web y un cliente web
- Usa TCP
- Es un protocolo sin estado (a veces mecanismos como cookies)
- Usa por defecto el puerto 80

HTTP

Conceptos

- Página web (o documento): consta de objetos (archivos direccionables por un URL)
- Las páginas web: archivo HTML base (contiene la distribución y, en general, el texto) y varios objetos
- Navegador: agente de usuario para la web
- Servidor web: alberga objetos
- Solicitud de una página web $\Longrightarrow$ el servidor devuelve el documento base

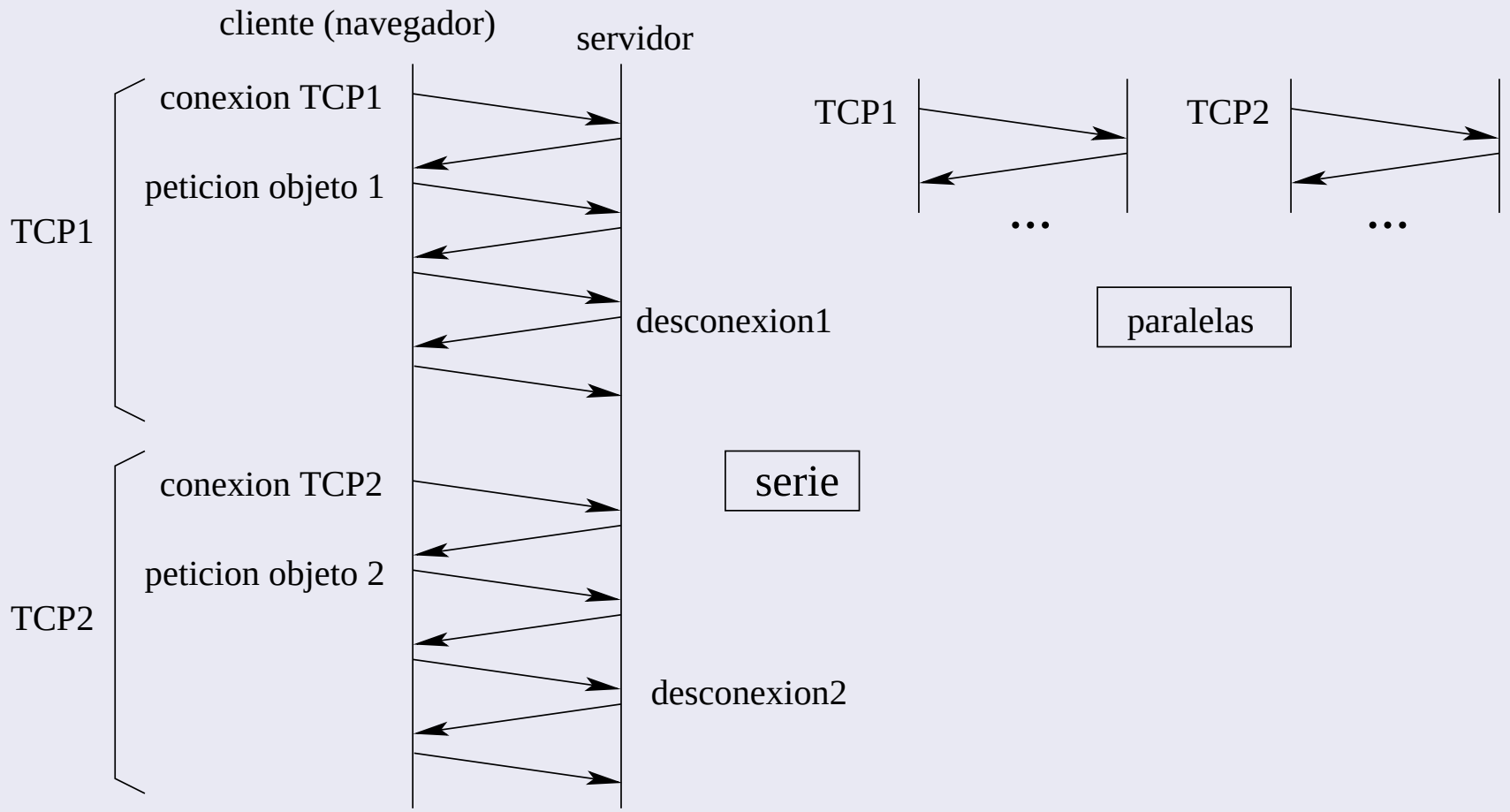
HTTP

Conexiones HTTP

- No persistentes: se usa una conexión TCP distinta para transferir cada objeto (HTTP/1.0)
  - Serie: se espera a que acabe la conexión TCP previa
  - Paralelo: se inician varias conexiones TCP a la vez
- Persistentes: se pueden transferir varios objetos (e incluso varias páginas) con la misma conexión TCP
  - Sin entubamiento: el cliente pide un nuevo objeto cuando se ha recibido el previo
  - Con entubamiento: el cliente puede hacer peticiones de varios objetos antes de recibir los anteriores (modo por defecto en HTTP/1.1)

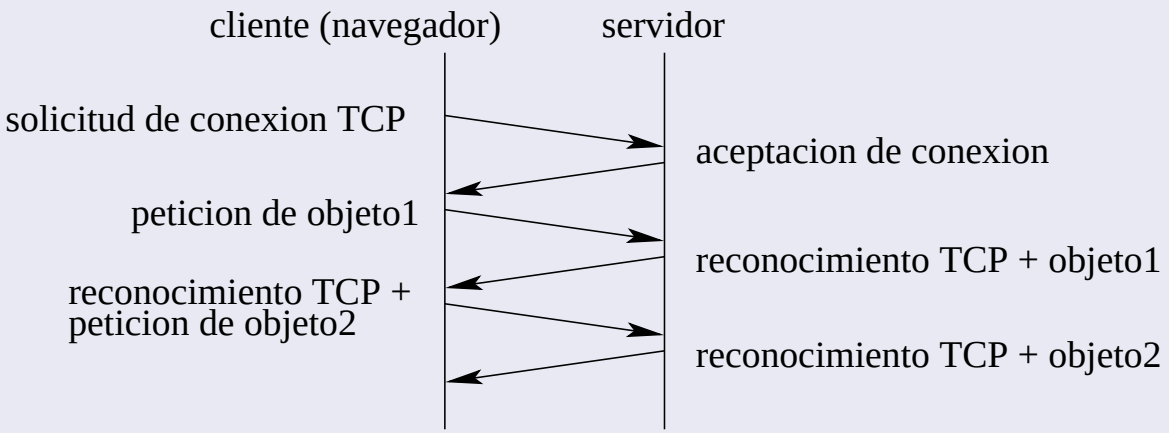
HTTP

HTTP no persistente

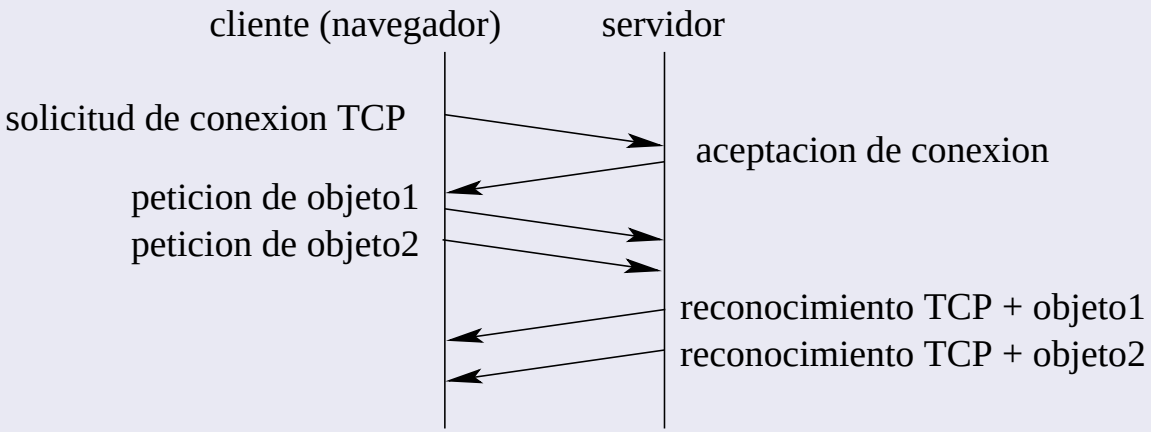


HTTP

HTTP persistente



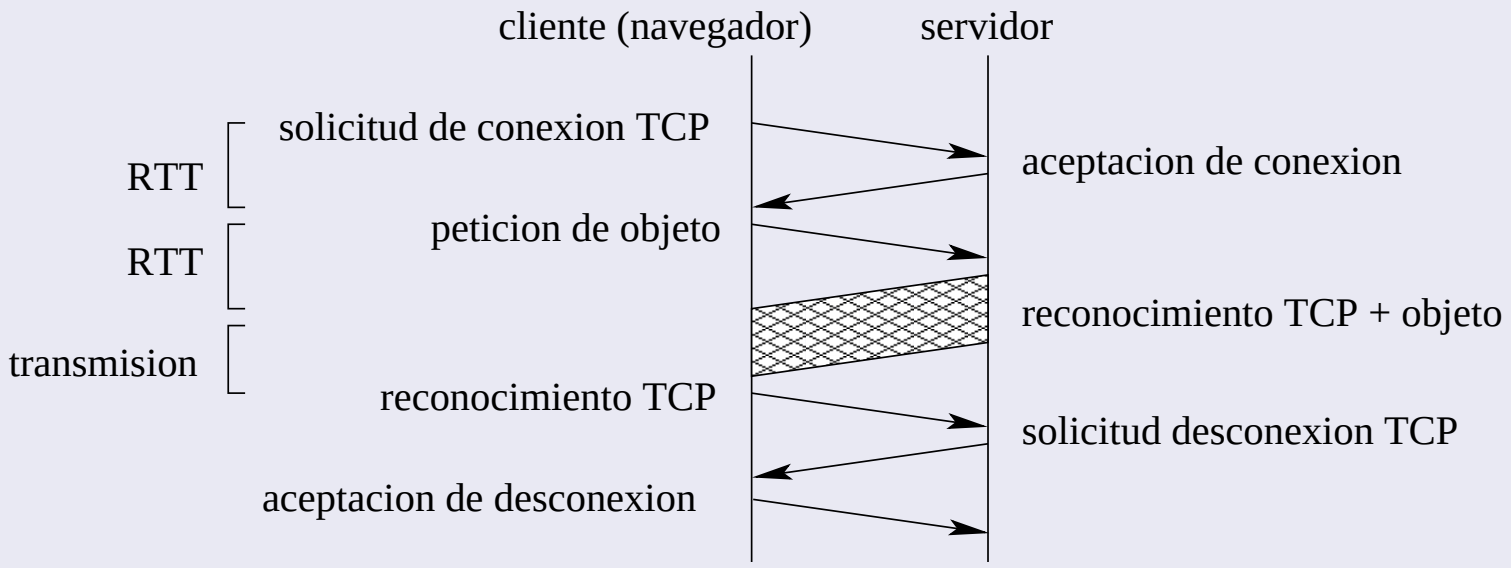
sin entubamiento



con entubamiento

Tiempo de transferencia de una página web

Conexiones no persistentes y serie



Tiempo de transferencia de una página web

Parámetros

- RTT (tiempo de ida y vuelta): tiempo necesario para que un paquete pequeño vaya del cliente al servidor y vuelva al cliente
- Tiempo de transmisión del archivo: depende del tamaño del archivo

Todos los tipos de conexiones

- El primer objeto se transfiere en $2RTT + t_{transmision}$
- Los siguientes objetos dependen del tipo de conexión

HTTP

Mensajes HTTP

Dos tipos:

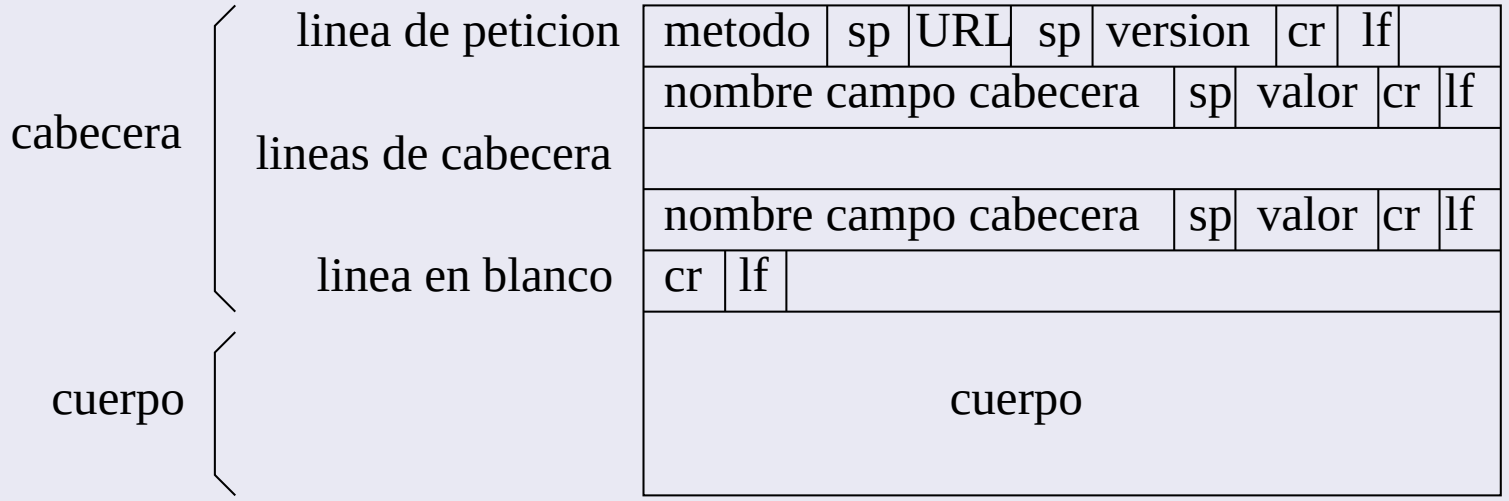
- Petición: para pedir objetos
- Respuesta: contienen los objetos

Ambos mensajes tienen:

- Cabecera: información de control, en ASCII de 7 bits
- Cuerpo: los datos (un objeto, contenido de un formulario, etc.) y son en binario

Mensajes HTTP

Mensajes de petición



Mensajes HTTP

Mensajes de petición

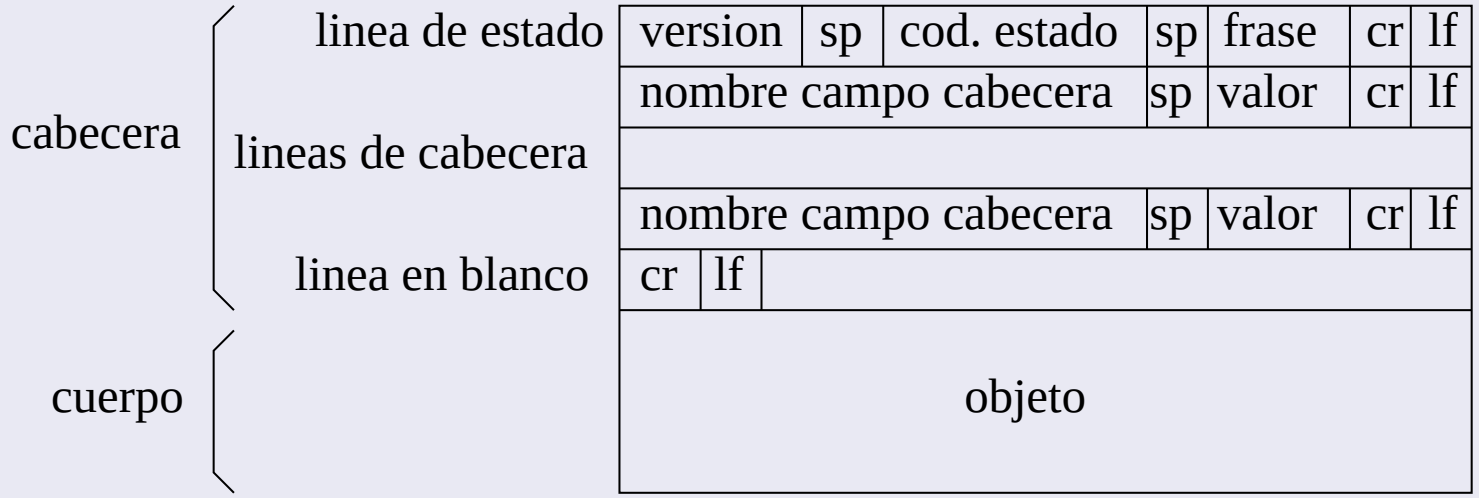
- Línea de petición: objeto solicitado
  - Método: GET (página normal) POST (formulario)
- Líneas de cabecera: opciones

Host:	nombre servidor web (puede haber varios para una IP)
Connection:	close (si queremos conexiones no persistentes)
User-agent:	mozilla/4.0 (el navegador)
Accept-languaje:	es (el idioma preferido de la página, si la hay)

- Cuerpo
  - GET: vacío
  - POST: los datos del formulario

Mensajes HTTP

Mensajes de respuesta



Mensajes HTTP

Mensajes de respuesta

- Línea de estado

200	OK (el objeto o página se sirve sin problemas)
404	Not found (la página no existe)
400	Bad Request (no se entendió el formato de la petición)

- Líneas de cabecera: opciones

Connection:	close (indica que se usan conexiones no persistentes)
Date:	fecha de envío de la página
Server:	Apache/1.3.0 (el servidor web)
Last-Modified:	cuando se creó o modificó por última vez la página
Content-Length:	tamaño en bytes del objeto, página o imagen
Content-Type:	text/html, image/gif, image/jpeg

- Cuerpo: los datos

HTTP/2

Nueva versión del protocolo HTTP

- Basado en el protocolo SPDY, desarrollado por Google (64 % más rápido)
- Especificación publicada en mayo de 2015
- Soportado por las últimas versiones de muchos navegadores

Mejoras

- Multiplexación de peticiones HTTP sobre una misma conexión TCP
- Compresión de cabeceras
- Soporte para *server push*
- Pipelining de solicitud–respuesta
- Cifrado de la información: uso obligatorio de SSL

Índice

1. Introducción
2. Protocolo de transferencia de hipertexto
3. Protocolo de transferencia de archivos
4. Protocolos de correo electrónico
5. DNS: Servicio de nombres de dominio
6. Distribución de contenidos

FTP

Protocolo de transferencia de archivos

- Define la comunicación con un servidor de archivos
- Con estado: mantiene información durante toda la sesión
- Usa dos conexiones TCP paralelas (información de control fuera de banda)
  - Conexión de control
    - Usa el puerto 21
    - Para enviar los comandos: nombre y clave de usuario, *dir*, *put*, *get* y recibir las respuestas
    - Es persistente: dura todo el tiempo de la conexión FTP
    - Usa ASCII de 7 bits
  - Conexión de datos
    - Usa el puerto 20
    - Para transmitir datos en respuesta a los comandos
    - Es no persistente: se abre y se cierra una conexión nueva por cada archivo que se transmite
    - Usa ASCII de 7 bits

FTP

Comandos y respuestas FTP

- Por la conexión de control
- Comandos: constan de 4 caracteres en mayúsculas y campos adicionales acabadas en cr+lf

USER	nombre de usuario
PASS	clave
LIST	lista de archivos
RETR	nombre de fichero (traer fichero)
STOR	nombre de fichero (almacenar fichero)

- Respuestas: un código de 3 dígitos y una frase explicativa

331	Username OK, password required
125	Data connection already open; transfer starting
425	Can't open data connection
452	Error writing file

Transmisión cifrada

- FTP, telnet y HTTP sin cifrar. SFTP, SSH y HTTPS no

Índice

1. Introducción
2. Protocolo de transferencia de hipertexto
3. Protocolo de transferencia de archivos
4. Protocolos de correo electrónico
5. DNS: Servicio de nombres de dominio
6. Distribución de contenidos

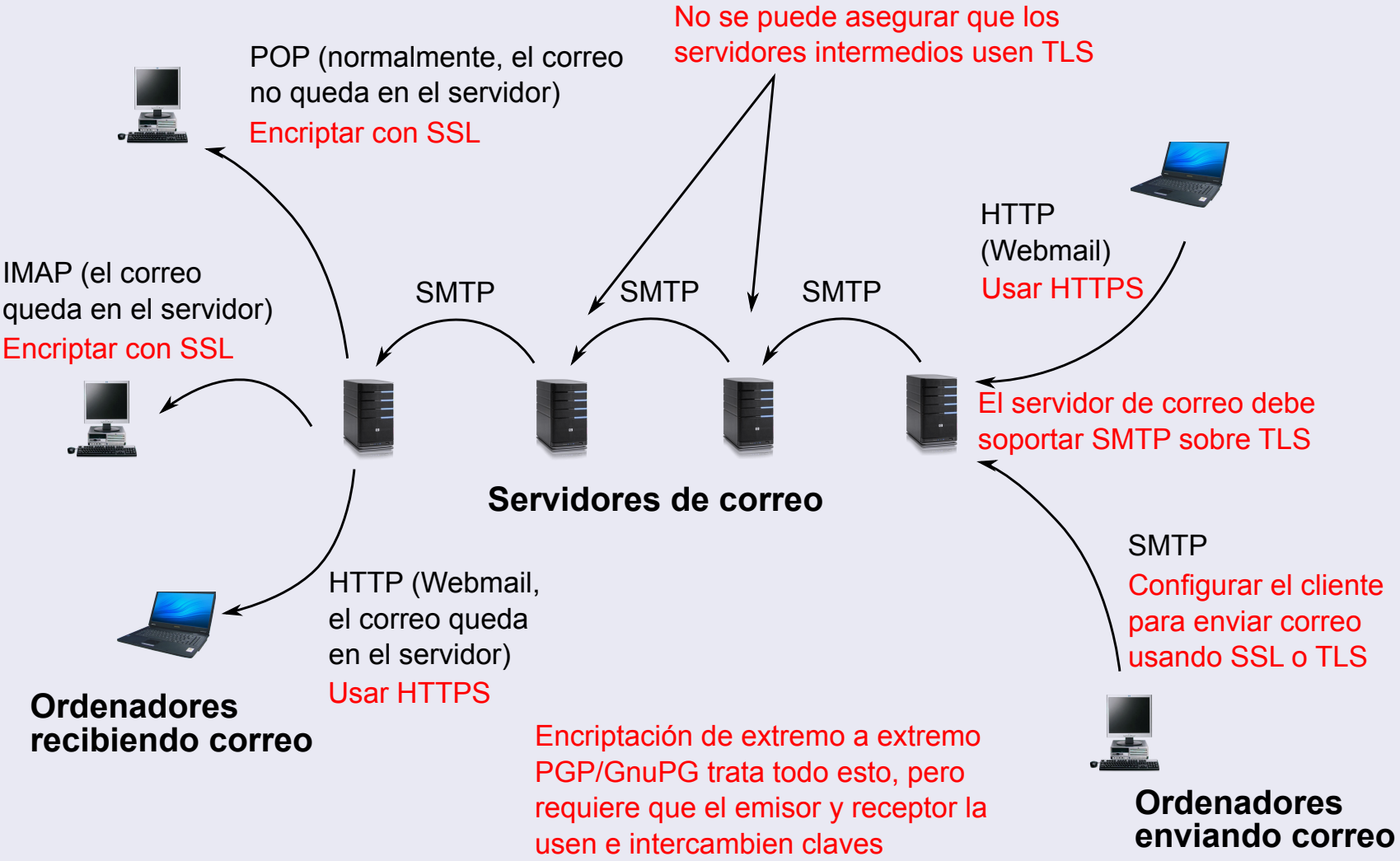
Correo electrónico

Protocolos de correo electrónico

- Para enviar correo al servidor de correo o entre servidores: SMTP, HTTP (si se usa un navegador)
- Acceso al correo por el agente de usuario: POP3, IMAP, HTTP (si se usa un navegador)

Correo electrónico

Protocolos de correo electrónico



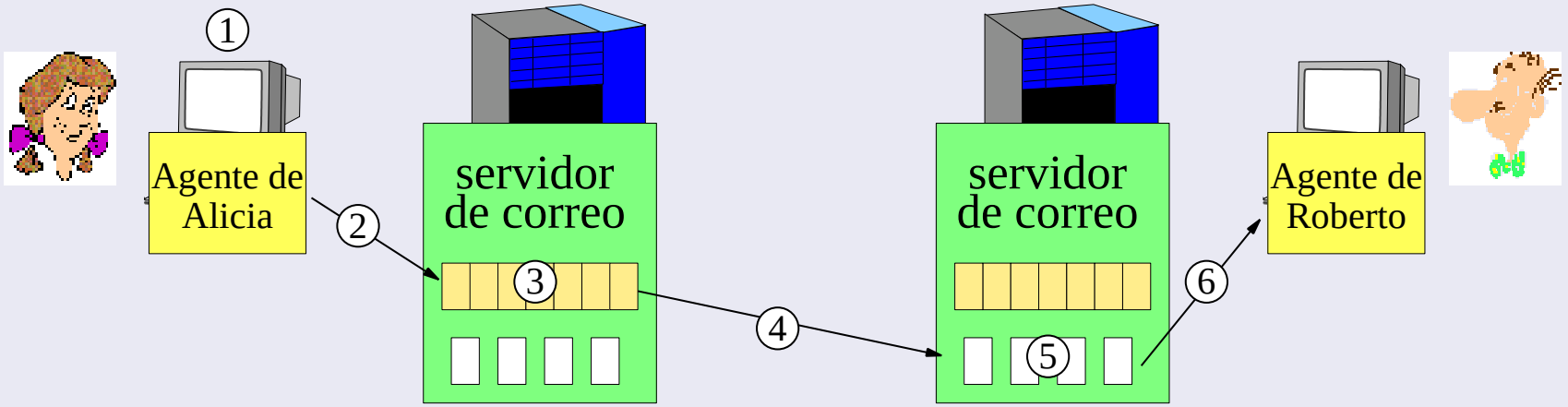
SMTP

Características del funcionamiento del correo electrónico

- Buzón de correo en el servidor
- El agente de usuario (AU) se comunica con el servidor
- Si el servidor destino está fuera de servicio, se reintenta (30 minutos)
- SMTP usa el puerto 25
- Usa conexiones TCP persistentes: mantiene abierta la conexión durante toda la sesión (varios mensajes)
- Emplea ASCII de 7 bits tanto para las cabeceras como para los cuerpos. Una imagen, el AU la transforma en ASCII

SMTP

Funcionamiento del correo electrónico



1. Alicia usa su AU para escribir un mensaje
2. El AU de Alicia envía el mensaje a su servidor de correo: se almacena en una cola de mensajes
3. El lado cliente de SMTP (servidor de Alicia) abre una conexión TCP con el servidor de Roberto
4. Tras una sincronización inicial (envío dirección destino y remite), el cliente envía el mensaje sobre la conexión TCP
5. El servidor de Roberto lo deposita en su buzón de correo
6. Roberto usa su AU para leer el mensaje

SMTP

Mensajes SMTP

Tres tipos de mensajes:

- Comandos: palabra en mayúscula seguida de los parámetros

HELO	nombre servidor
MAIL FROM:	dirección de correo del remitente
RCPT TO:	direccion de correo del destinatario
DATA	(el contenido del correo)
QUIT	

- Respuestas: código numérico seguido de una frase aclarativa

220	nombre del servidor
250	comando que se ejecutó satistactoriamente
354	envíe el correo, terminando en .
221	cierre de conexión

- Datos: el contenido de los correos: todos los objetos encapsulados en un único archivo codificado en ASCII de 7 bits (con las llamadas extensiones MIME)

SMTP

Ejemplo de transacción típica entre cliente (C) y servidor (S) de correo

```
S:  220 smtp.usc.es SMTP USC Server
C:  HELO smtp.usc.es
S:  250 smtp.usc.es
C:  MAIL FROM: pepito@usc.es
S:  250 Ok
C:  RCPT TO: juanito@yahoo.es
S:  250 Ok
C:  DATA
S:  354 Enter data, end data with <CR><LF>.<CR><LF>
C:  Hola
C:  Esto es una prueba, adios
C:  .
S:  250 Ok: message queued as C5F0B13210D
C:  quit
S:  221 Bye
```

SMTP

Protocolo inseguro

- No se pide nombre y clave de usuario: permite enviar correos a cualquiera

Comparación con HTTP

- Ambos transmiten archivos y usan conexiones TCP persistentes
- HTTP es un protocolo de demanda (el cliente demanda el archivo) y SMTP de oferta (el cliente oferta el archivo)
- SMTP requiere que la cabecera y el cuerpo sean de ASCII de 7. En HTTP el cuerpo puede contener datos binarios
- HTTP envía objetos en archivos diferentes, SMTP van encapsulados en el mismo archivo que contiene el texto
- HTTP no tiene estado y SMTP debe recordar la fase de la sesión en la que se encuentra

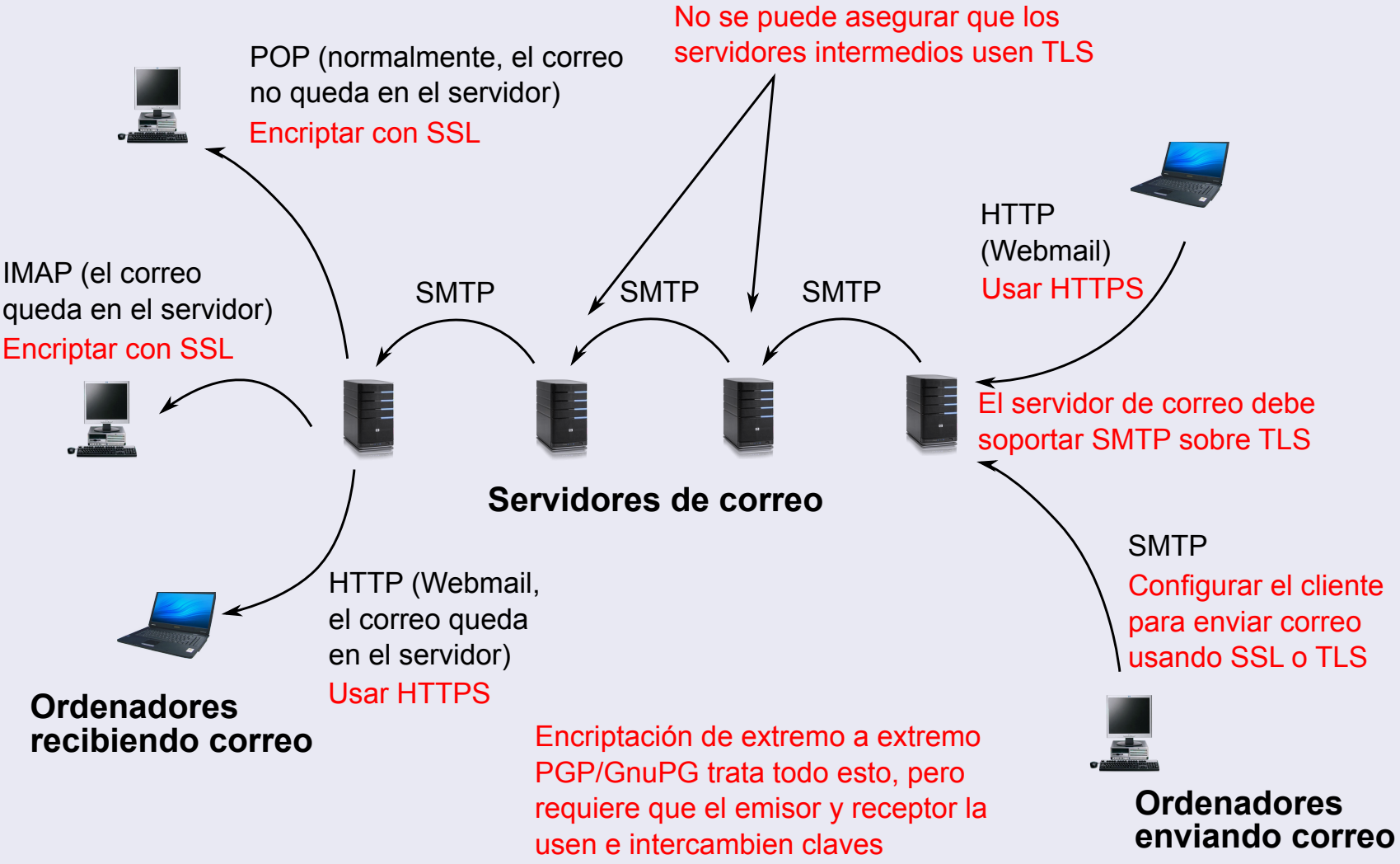
Correo electrónico

Protocolos de acceso al correo

- Los agentes de usuarios para transferir el correo del buzón destino al ordenador local
- No se usa SMTP porque es un protocolo de oferta $\Longrightarrow$ ordenador local siempre encendido esperando correos
- Se usan protocolos que permiten descargar los correos: POP3, IMAP y HTTP

Correo electrónico

Protocolos de correo electrónico



POP3: Protocolo de oficina postal, versión 3

POP3

- Protocolo sencillo para acceso al correo
  - Descargar y borrar del servidor
  - Descargar y mantener en el servidor
- Conexión TCP con el puerto 110 del servidor

Protocolo simple: tres fases

- Autorización: autenticación del usuario
- Transacción
  - Recuperar los mensajes
  - Marcar/desmarcar para borrado
  - Estadísticas de correo
- Actualización: borrado de los mensajes marcados
  - Termina la sesión POP3

POP3: Protocolo de oficina postal, versión 3

Mensajes POP3

Tres tipos de mensajes:

- Comandos: palabra de 4 caracteres seguida de los parámetros

user	nombre del usuario
pass	palabra clave
list	
retr	número de correo (traer correo)
dele	número de correo (borrar correo)
quit	

- Respuestas: básicamente de dos tipos

+OK frase explicativa
−ERR frase explicativa

- Datos: datos de la respuesta
  - La lista de mensajes, los contenidos de los correos, etc.
  - El correo completo en un único mensaje (SMTP generaba un mensaje por cada línea de cabecera)

POP3: Protocolo de oficina postal, versión 3

Otras características

- Usa conexiones persistentes: se usa la misma conexión TCP mientras dure la sesión en la que pueden descargarse varios correos
- POP3 se puede usar para descargar y borrar (retr+dele) los correos del servidor o para descargar y mantener en el servidor (retr)
- Necesita cuenta con contraseña
- No mantiene información de estado entre sesiones POP3
- Definido en el RFC 1939

IMAP: Protocolo de acceso a mensajes de Internet

Inconvenientes de POP3

- Acceso al correo en múltiples dispositivos
- No permite carpetas en el servidor

IMAP

- Más funcionalidades que POP3 $\Longrightarrow$ más complejo
- El servidor asocia cada mensaje con una carpeta
  - Cuando llega un mensaje va a la carpeta INBOX
  - Permite leer, borrar, mover a otra carpeta...
- Comandos para crear carpetas, realizar búsquedas, descargar partes del mensaje...
- Mantiene información de estado entre sesiones
- Definido en el RFC 3501

Índice

1. Introducción
2. Protocolo de transferencia de hipertexto
3. Protocolo de transferencia de archivos
4. Protocolos de correo electrónico
5. DNS: Servicio de nombres de dominio
6. Distribución de contenidos

DNS

Servicio de nombres de dominio

- Para traducir nombres de hosts $\rightleftarrows$ direcciones IP

Componentes

- Numerosos servidores de nombres distribuidos por Internet
- La base de datos del DNS está distribuida de forma jerárquica
- Un protocolo que permite a los hosts pedir traducciones a los servidores y que los servidores se intercambien datos entre ellos

Protocolo sin conexión

- DNS usa UDP, el puerto 53 y no tiene estado

DNS

Servicios proporcionados

- Traducir nombres de hosts $\rightleftarrows$ direcciones IP y obtención de alias
- Informa de los servidores autorizados para un dominio
- Alias de servidores de correo: permite simplificar las direcciones
- Distribución de la carga
  - Se asigna varias IPs a un nombre de host
  - Servidores espejo: con el mismo contenido
  - Devuelve de forma cíclica una IP del conjunto de las asignadas

DNS

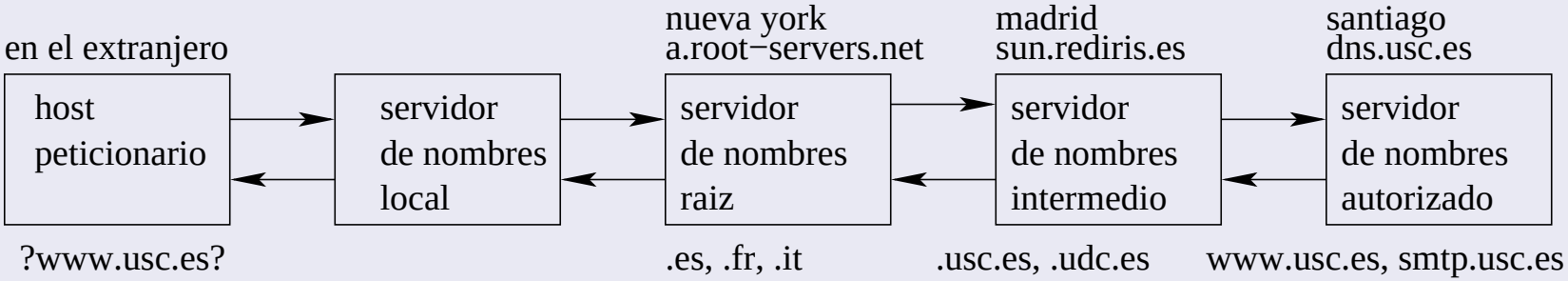
Tipos de servidores

- Servidores locales: atienden las consultas de los hosts
- Servidores autorizados o autoritativos
  - Para que un host sea accesible en Internet, debe estar registrado en un servidor autorizado
  - Normalmente pertenecen al ISP
  - Cada host debe estar en dos servidores autorizados por fiabilidad
  - Muchos servidores autorizados se comportan como locales
- Servidores raíz: sobre 400 en Internet, gestionados por 13 organizaciones
  - Información de los dominios de primer nivel (TLD)
- Servidores intermedios o TLD: información sobre los niveles intermedios

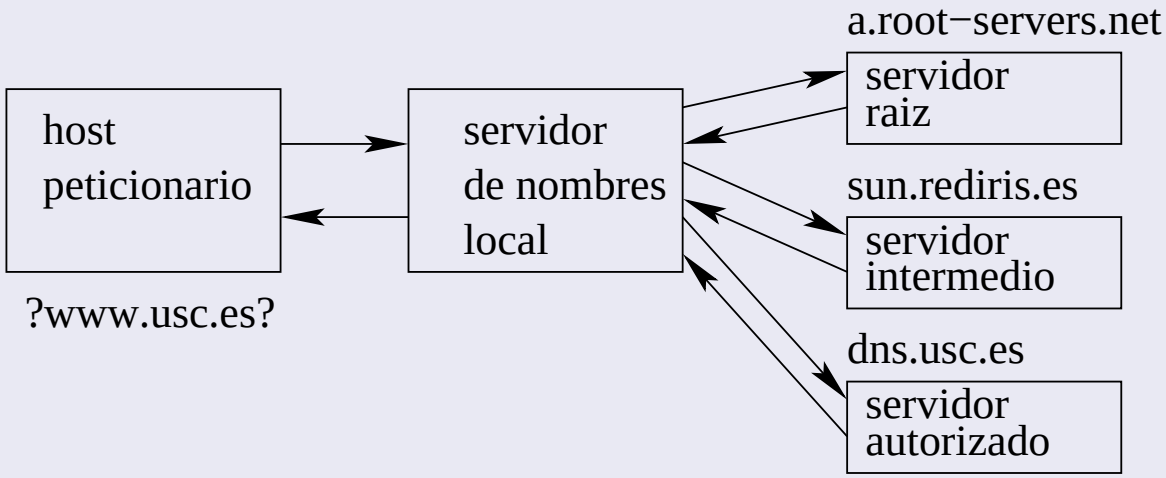
DNS

Tipos de consultas

- Consultas recursivas: cada servidor DNS interroga al siguiente



- Consultas iterativas: el servidor DNS local contacta con todos los servidores



DNS

Caché de DNS

- Los servidores almacenan copias locales de las correspondencias que obtienen
- Se borran después de cierto tiempo sin usar (dos días)
- En todos los niveles de la jerarquía, incluso en los hosts locales

DNS

Mensajes DNS

Consultas y respuestas:

- Cabecera con información de control
  - Identificación: 16 bits que identifican la consulta (y su correspondiente respuesta)
  - Señales: 4 bits que indican si consulta o respuesta, tipo de consulta, etc.
  - Tamaño de los campos del cuerpo
- Cuerpo: 4 campos
  - Cuestiones: una o varias preguntas. Nombre o dirección a traducir, etc
  - Respuestas: una o varias respuestas. Pareja host/IP
  - Servidores autorizados. Permite hacer una cadena de consultas
  - Información adicional

DNS

Mensajes DNS

cabecera	identificacion	senales
	num. cuestiones	num. respuestas
	num. s. autorizados	num. inf. adicional
cuerpo	cuestiones	
	respuestas	
	servidores autorizados	
	informacion adicional	

Ejemplo:

1	1
2	2

www.usc.es?

www.usc.es −> 193.144.74.224

usc.es −> dns.usc.es, dns2.usc.es

direcciones IP de los servidores autorizados

DNS

Mensajes DNS

Esos campos contienen registros de recursos: 4-tupla

Tipo	Nombre	Valor	TTL
A	nombre de host	dirección IP	
NS	dominio	servidor autorizado para el dominio	
CNAME	alias	nombre de host	
MX	alias de correo	servidor de correo	
AAAA	nombre de host	dirección IPv6	

TTL: tiempo durante el cual es válida la respuesta

Índice

1. Introducción
2. Protocolo de transferencia de hipertexto
3. Protocolo de transferencia de archivos
4. Protocolos de correo electrónico
5. DNS: Servicio de nombres de dominio
6. Distribución de contenidos

Distribución de contenidos

Proporcionar contenidos

El acceso a servidores centralizados puede ser lento:

- El camino de los mensajes es lento o está congestionado
- El servidor está sobrecargado

$\Longrightarrow$ se distribuyen (y duplican) los contenidos en distintas zonas y se dirigen las peticiones al servidor de menor tiempo de respuesta

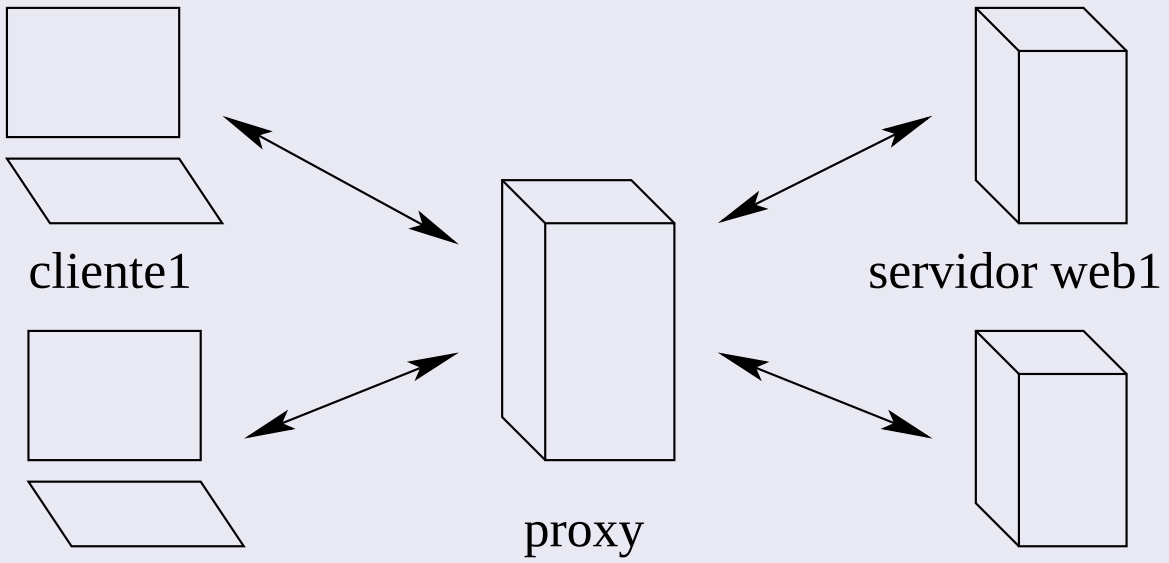
Métodos

Método de distribución	Quién proporciona la infraestructura
Caché web (o servidor proxy)	ISP
Redes de distribución de contenidos	distribuidor de contenidos
Redes P2P (de igual a igual)	usuario

Distribución de contenidos

Caché web o servidor proxy

- Un servidor intermedio (proxy) por el que pasan todas las peticiones web de los hosts de una red
- Mantiene copias de los contenidos durante cierto tiempo
- El usuario debe configurar el navegador
- Permite un esquema jerárquico
- Proporcionado por el ISP



Distribución de contenidos

Redes de distribución de contenidos (CDN)

- Empresas que poseen *centros de host de Internet* (compañía CDN) alquilan su infraestructura
- La CDN replica los contenidos de sus clientes en los servidores CDN y los mantiene actualizados
- La CDN proporciona un mecanismo para que el contenido sea entregado por el servidor CDN que pueda hacerlo más rápidamente
- Acceso a los contenidos mediante dos técnicas:
  - Redirección de objetos
  - Balanceo de las peticiones usando el DNS

Distribución de contenidos

Redes P2P

- Todos los usuarios son a la vez servidores y clientes usando una aplicación
- Se necesita un nodo de arranque
- Se necesita construir un directorio

Tipo de directorio	Ejemplo
centralizado	napster
no centralizado	kaaza
inundación de consultas	gnutella

- BitTorrent
  - Cada archivo se comparte en una red P2P propia
  - Se necesita un *tracker* para cada red
  - No se necesita construir un directorio. La información de cada archivo va en los .torrent

Distribución de contenidos

Aplicaciones P2P

- Escalabilidad de las arquitecturas P2P
- BitTorrent
- Tablas hash distribuidas (DHT)
- Telefonía sobre Internet: Skype

Flujos de vídeo y redes de distribución de contenido

- Vídeo por Internet
- Flujos de vídeo HTTP y tecnología de flujos dinámicos adaptativos sobre HTTP (DASH)
- Redes de distribución de contenidos (CDN)
- Casos de estudio: Netflix, YouTube y Kankan

Redes

Tema 3: Capa de transporte

José Carlos Cabaleiro Domínguez

Escola Técnica Superior de Enxeñería

Índice

1. Introducción
2. UDP: protocolo de datagramas de usuario
3. Fundamentos de la transmisión fiable
4. TCP: protocolo de control de transmisión
5. Control de congestión

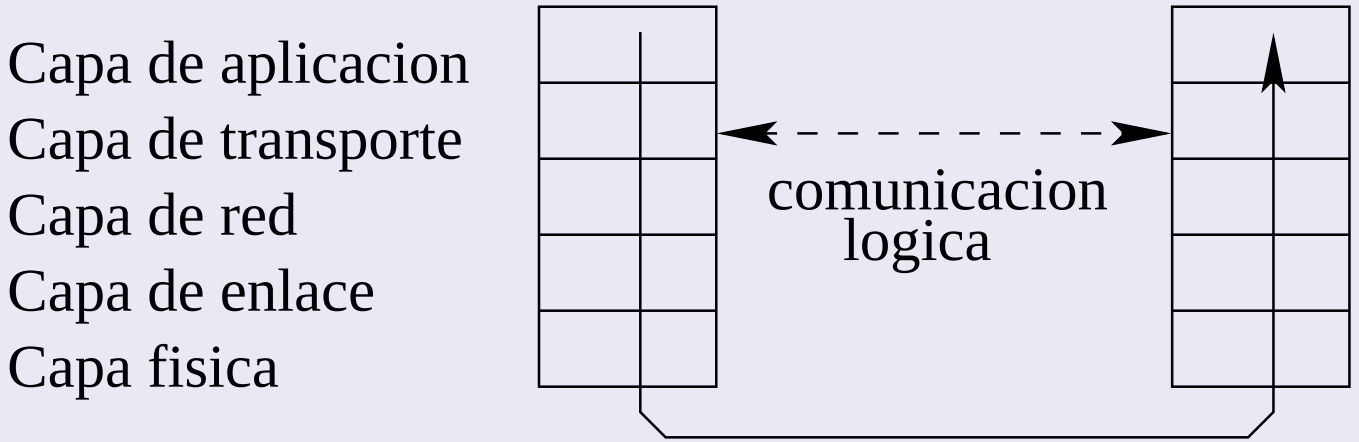
Índice

1. Introducción
2. UDP: protocolo de datagramas de usuario
3. Fundamentos de la transmisión fiable
4. TCP: protocolo de control de transmisión
5. Control de congestión

Introducción

Capa de transporte

- Prepara los mensajes de las aplicaciones para ser transmitidos
- En destino, recupera los mensajes y los entrega a las aplicaciones
- Solo está implementada en los sistemas finales
- Proporciona una comunicación lógica entre procesos



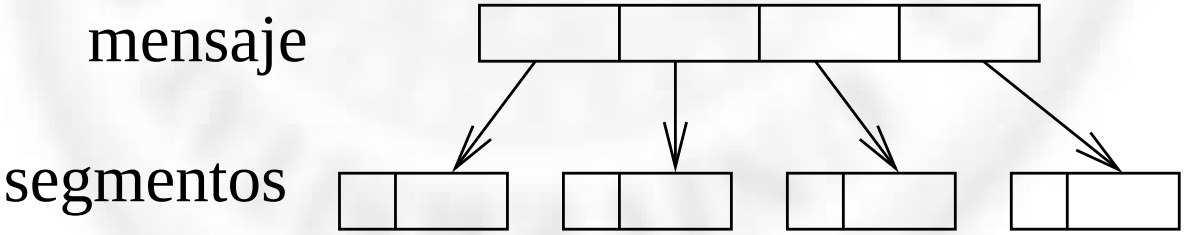
Capa de transporte

En TCP/IP

- Prepara los mensajes para transmitirlos por un canal no fiable (red de datagramas, IP, servicio de mejor esfuerzo)
- Protocolos de transporte TCP y UDP

TCP en origen

- Fragmentar los mensajes en segmentos
- Añadirles la cabecera de la capa de transporte



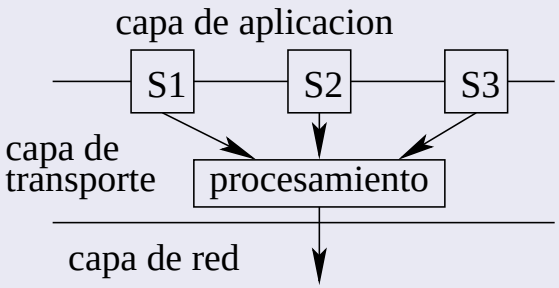
UDP en origen

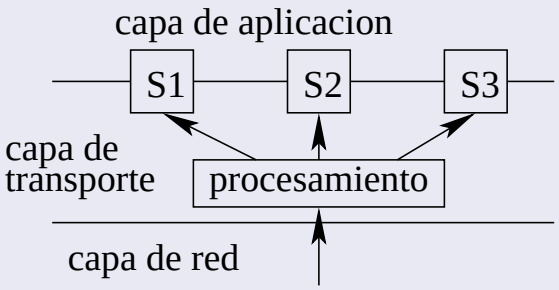
- Añadirles la cabecera de la capa de transporte

Multiplexación y demultiplexación

Interacción entre las aplicaciones y la capa de transporte

- Los mensajes pasan de la capa de aplicación a la capa de transporte a través de un socket
  - Los procesos escriben y leen del socket
  - La capa de transporte recoge los mensajes del socket y los traslada al socket destino
- Multiplexación: recorrer todos los sockets abiertos, procesar los mensajes y enviarlos a la capa de red
- Demultiplexación: recoger los segmentos que llegan de la capa de red, reconstruir los mensajes y colocarlos en los sockets destino

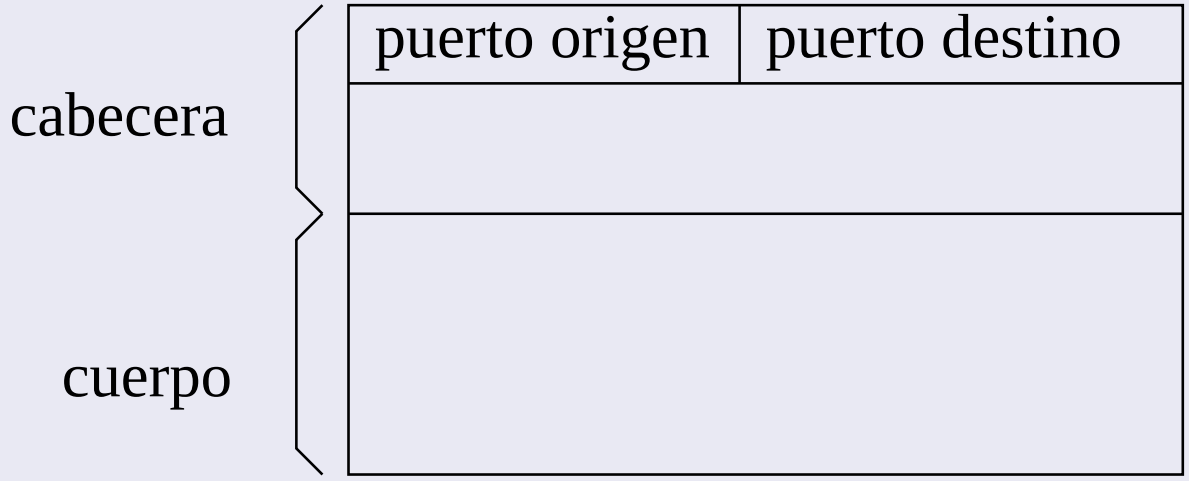




Multiplexación y demultiplexación

Interacción entre las aplicaciones y la capa de transporte

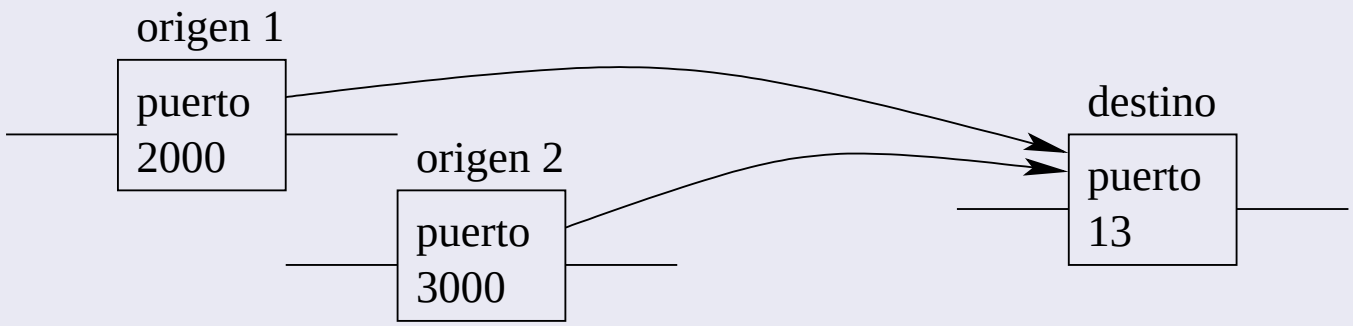
- Identificación del socket destino: a través de los números de puerto de la cabecera del segmento
  - Enteros de 16 bits
  - De 0–1023 usados por el administrador para los servicios bien conocidos



Multiplexación y demultiplexación

Multiplexación con sockets sin conexión (en UDP)

- Identificación del socket: la pareja dirección IP destino y puerto destino
- Segmentos de distintos hosts que se entreguen en el mismo puerto son recogidos por el mismo proceso
- El puerto origen se usa para que el proceso sepa a quién responder

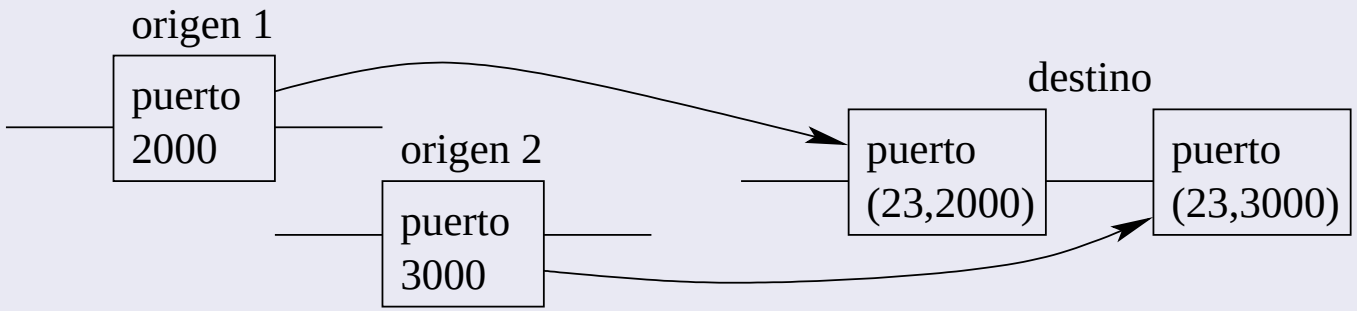


Multiplexación y demultiplexación

Multiplexación con sockets orientados a conexión (en TCP)

- Identificación del socket: la tupla de direcciones IP origen y destino y puertos origen y destino
- Segmentos de distintos hosts (o distintos puertos origen) que se entreguen en el mismo puerto van a sockets distintos

$\Longrightarrow$ varias conexiones al mismo puerto puedan ser atendidas por procesos o hilos diferentes

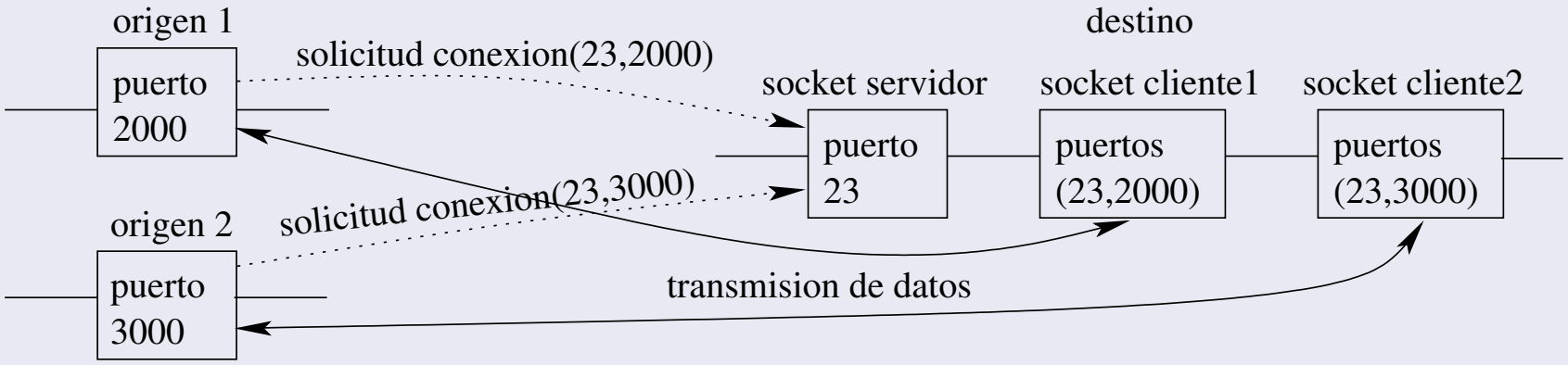


Multiplexación con sockets orientados a conexión TCP

Conexiones TCP

Dos tipos de sockets:

- Un socket de servidor: espera conexiones de clientes
- Varios sockets de conexión: se encargan de la transmisión de datos



Índice

1. Introducción
2. UDP: protocolo de datagramas de usuario
3. Fundamentos de la transmisión fiable
4. TCP: protocolo de control de transmisión
5. Control de congestión

Transporte no orientado a conexión: UDP

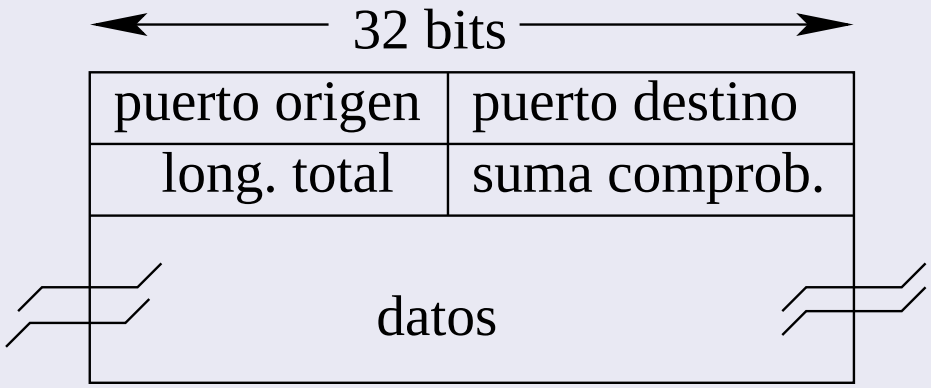
Protocolo de datagramas de usuario

- Protocolo de transporte simple y poco sofisticado
- Hace casi lo mínimo que debería hacer un protocolo de transporte
  - Multiplexación/demultiplexación
  - Mecanismo de comprobación de errores
- Descrito en el RFC 768

Transporte no orientado a conexión: UDP

Protocolo de datagramas de usuario

- En origen, le añade una cabecera al mensaje para formar un segmento
- En destino, comprueba si el paquete llegó sin errores
  - Sin errores: entrega el paquete al socket
  - Con errores: en general se descarta
- Cabecera de 4 campos (8 bytes):
  - Puertos origen y destino
  - Longitud total del segmento (cabecera + datos) en bytes
  - Suma de comprobación incluyendo la pseudo-cabecera



Transporte no orientado a conexión: UDP

Características

- Sin conexión
  - No hay acuerdo previo entre emisor y receptor $\Longrightarrow$ no hay retardo en el establecimiento de la conexión
  - No hay retransmisiones en caso de errores
- Sin estado: cada segmento se envía con independencia de los demás
  - No hay segmentación (no hay información para reconstruir los mensajes)
  - Procesamiento más rápido y con menos recursos (más clientes)
- Cabeceras más pequeñas
- Más control de la aplicación
- Para aplicaciones que prefieren velocidad frente a fiabilidad: transmisión de audio, vídeo, DNS...

Transporte no orientado a conexión: UDP

Aplicaciones que usan TCP

- Correo electrónico (SMTP)
- Web (HTTP)
- Acceso a terminales remotos (telnet, SSH)
- Transferencia de archivos (FTP, SFTP)

Aplicaciones que normalmente usan UDP

- Traducción de nombres (DNS)
- Protocolos de encaminamiento (RIP)
- Administración de red (SNMP)
- Servidor de archivos remoto (NFS)

Aplicaciones que usan TCP o UDP

- Flujos multimedia
- Telefonía por Internet

Cada vez se usa más TCP en aplicaciones multimedia, ¿?

- Aplicaciones multimedia: toleran pérdidas y responden mal a mecanismos de control de congestión
- Hay organizaciones que bloquean el tráfico UDP, porque no tienen mecanismos de control de congestión
  - Desbordamiento de paquetes en los routers
  - Estrangulamiento del tráfico TCP, que sí tiene mecanismos de control de congestión

Índice

1. Introducción
2. UDP: protocolo de datagramas de usuario
3. Fundamentos de la transmisión fiable
4. TCP: protocolo de control de transmisión
5. Control de congestión

Transmisión fiable

Fundamentos

- Se basa en los protocolos de retransmisión: que retransmiten los paquetes con errores
- Protocolos ARQ *Automatic Repeat reQuest* (solicitud automática de repetición)
  - Parar y esperar: el emisor envía un paquete y espera la confirmación del receptor
  - Ventana deslizante: se pueden enviar varios paquetes antes de recibir las confirmaciones
    - Retroceder N
    - Repetición selectiva
- Se consideran enlaces bidireccionales (*full duplex*)
- Numeración de los paquetes 0, 1, ..., $2^n - 1$, n número de bits para numerarlos

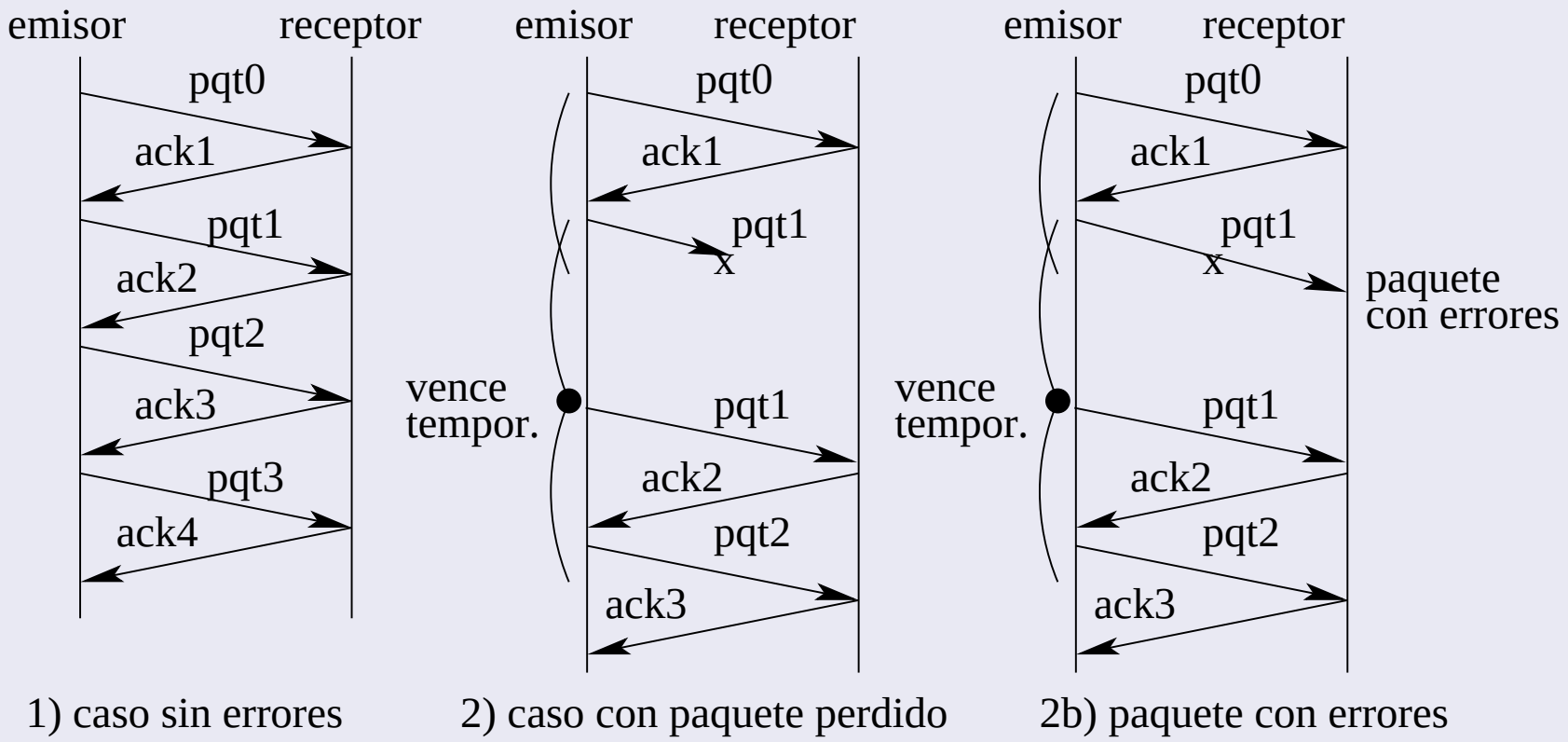
Protocolo ARQ parar y esperar

Casos

- Sin errores: el receptor devuelve un ACK con el número del siguiente paquete
- Pérdida de paquetes: el receptor no devuelve el ACK (timeout en el emisor) y el emisor retransmite el paquete
- Paquete con errores: similar al anterior
- Pérdida de un ACK: el emisor retransmite el paquete (timeout). El receptor recibe un duplicado, lo descarta y devuelve el ACK
- Timeout: paquetes y ACKs llegan con retraso $\Longrightarrow$ paquetes y ACKs duplicados
  - Paquetes duplicados: como antes
  - ACKs duplicados: se ignoran

Protocolo ARQ parar y esperar

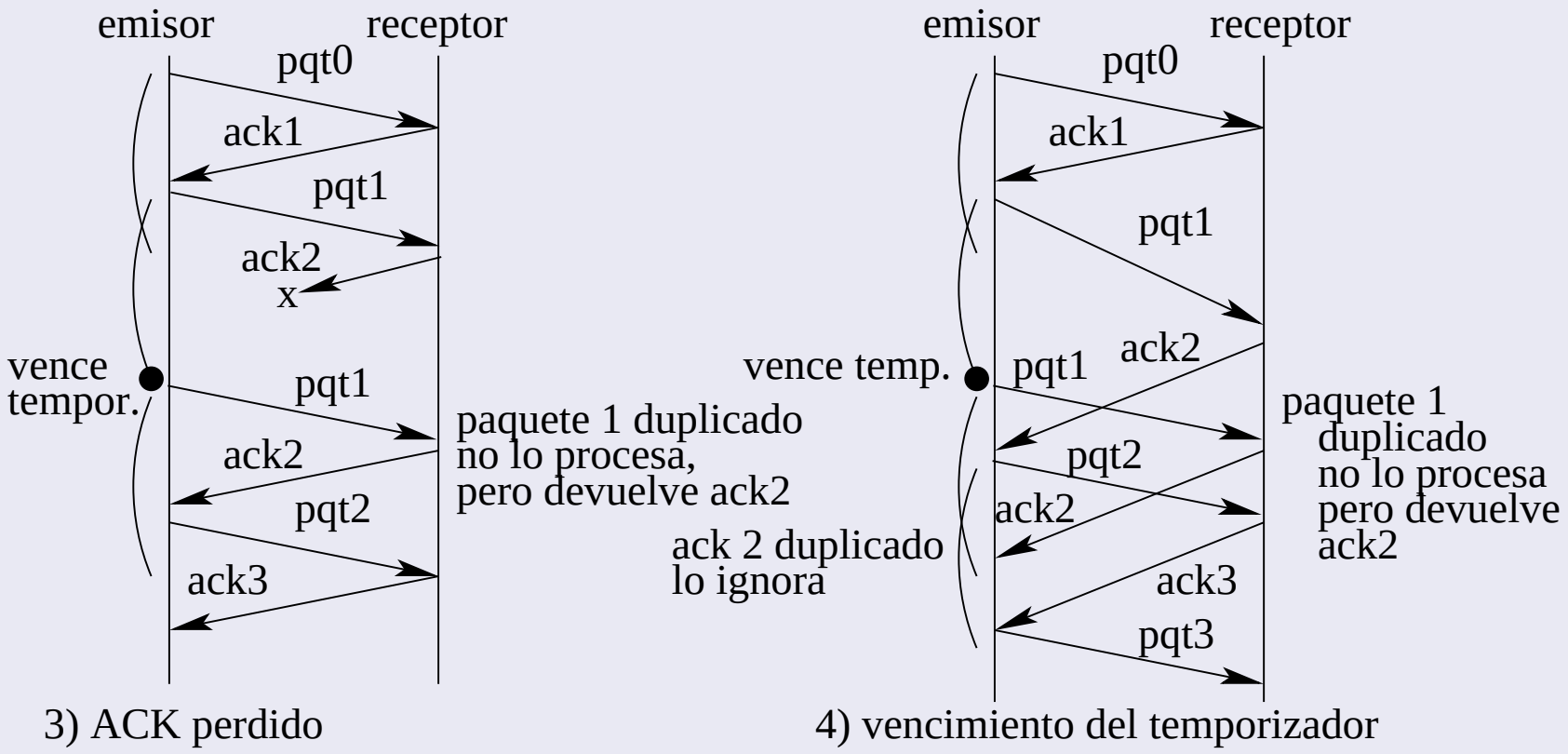
Casos



1) caso sin errores 2) caso con paquete perdido 2b) paquete con errores

Protocolo ARQ parar y esperar

Casos



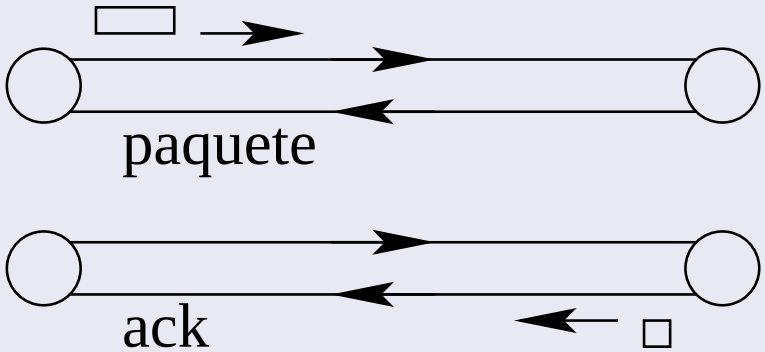
Protocolo ARQ parar y esperar

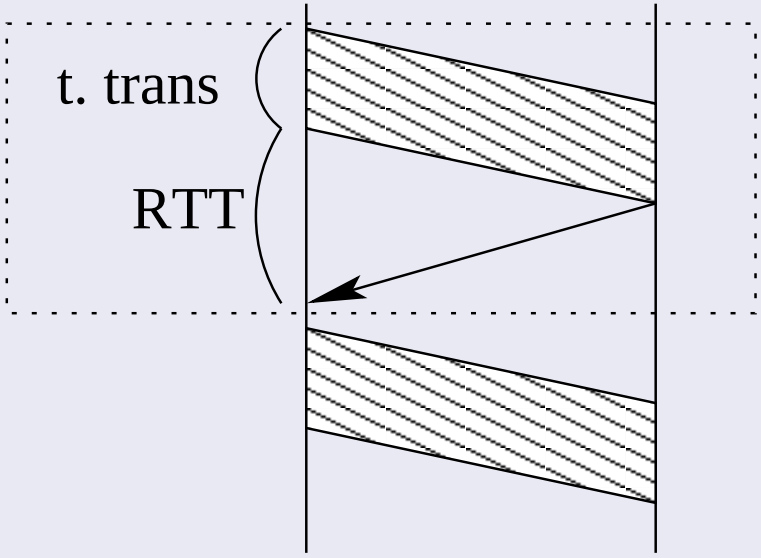
Variantes

- NAK: indica la recepción de un paquete con errores $\Longrightarrow$ no se espera al timeout
- Dos ACKs iguales equivalen a un NAK: la recepción de un paquete con errores se devuelve el ACK del último paquete correcto
  - Si vence el temporizador, se envían continuamente duplicados de paquetes
- Tres ACKs iguales equivalen a un NAK: resuelve inconvenientes de la anterior
  - TCP lo usa con alguna variación

Protocolo ARQ parar y esperar

Inconveniente principal: poca utilización del enlace





Utilización del enlace por parte del emisor: $U = \dfrac{t_{trans}}{RTT + t_{trans}}$

Ejemplo: paquetes de 4.000 bits, por un enlace a 1 Gbps de 1.000 km con una velocidad de propagación de 200.000 km/s

Protocolo ARQ de ventana deslizante

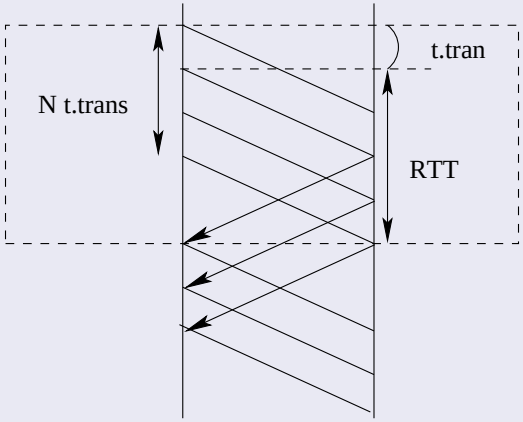
Entubamiento

- El emisor envía un determinado número N de paquetes antes de recibir las confirmaciones

Utilización del enlace

- Tiempo útil en el emisor: $N \cdot t_{trans}$
- Tiempo total: $t_{trans} + RTT$
- $U = 1$ cuando $N \cdot t_{trans} \geq t_{trans} + RTT$

$$\Longrightarrow N \geq 1 + \frac{RTT}{t_{trans}}$$



Protocolo ARQ de ventana deslizante

Requerimientos

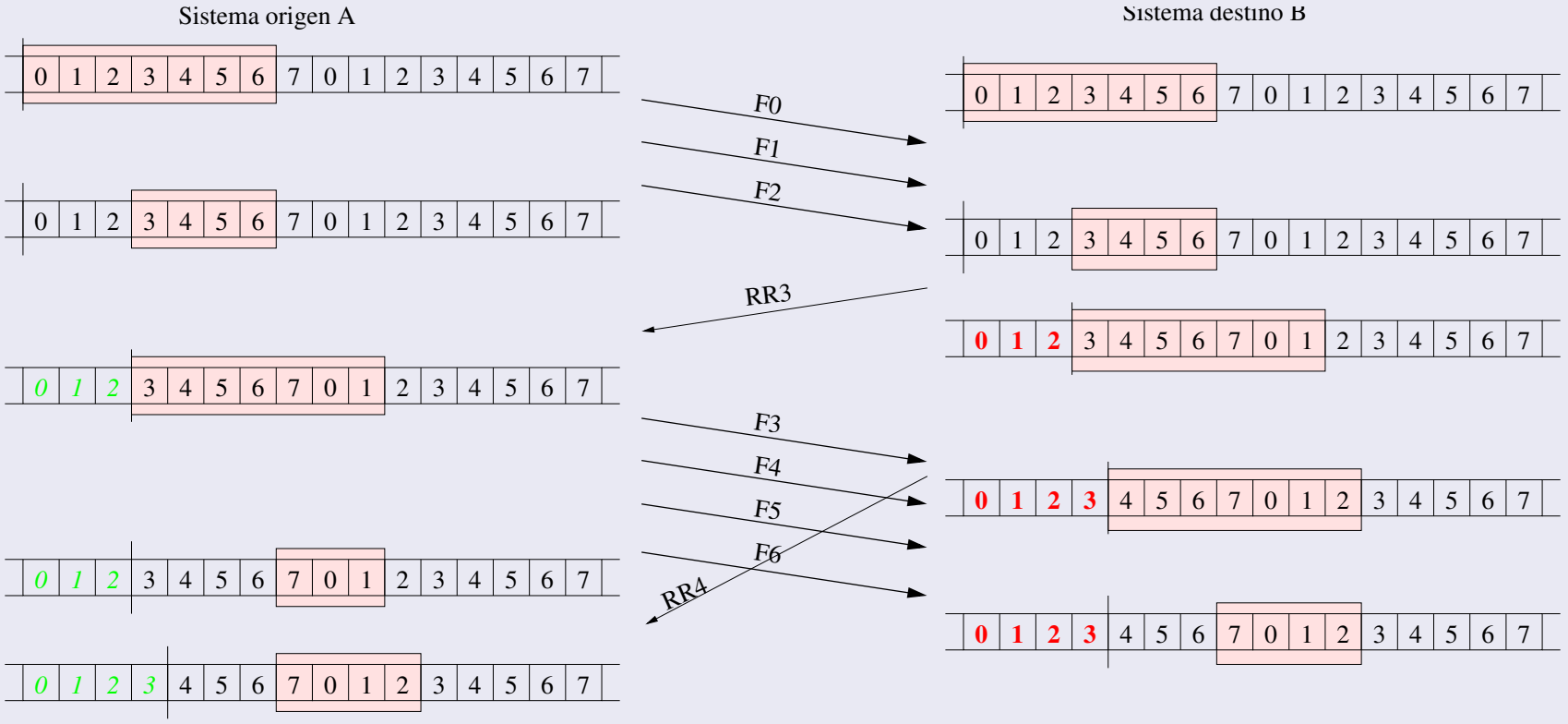
- El rango de números de secuencia debe abarcar al menos el doble del tamaño de la ventana emisora
- Emisor y receptor deben almacenar más de un paquete

Ventanas

- Ventana emisora: es el conjunto de N paquetes que el emisor puede enviar o están pendientes de confirmación
- Ventana receptora: es el conjunto de N paquetes que el receptor puede aceptar o está procesando
- Estas ventanas se van desplazando a medida que el emisor (receptor) reciba (devuelva) las confirmaciones

Protocolo ARQ de ventana deslizante

Ejemplo



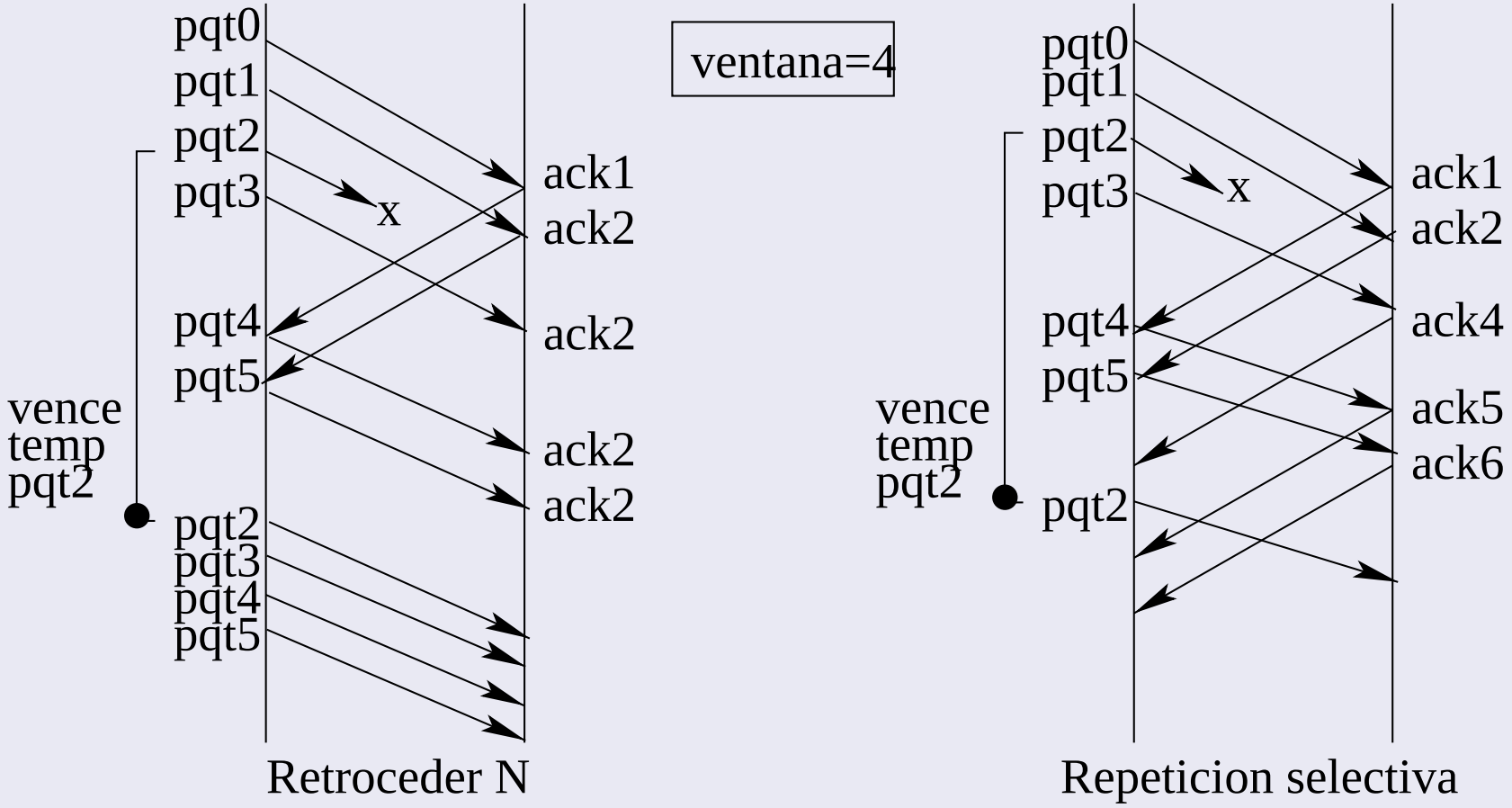
Protocolo ARQ de ventana deslizante

Tipos

- Retroceder *N* (*Go Back N*): el receptor sólo acepta paquetes en orden
  - Si un paquete llega con errores o no llega, se descartan los siguientes
  - Si expira el temporizador de un paquete, se retransmiten también los siguientes
  - ACKs acumulativos: un ACK implica un ACK a todos los paquetes previos
- Repetición selectiva: el receptor acepta paquetes fuera de orden
  - Solo se retransmiten los paquetes erróneos o que no llegan
  - Hay que enviar los ACKs para cada uno de lo paquetes recibidos

Protocolo ARQ de ventana deslizante

Retroceder *N* y repetición selectiva



Índice

1. Introducción
2. UDP: protocolo de datagramas de usuario
3. Fundamentos de la transmisión fiable
4. TCP: protocolo de control de transmisión
5. Control de congestión

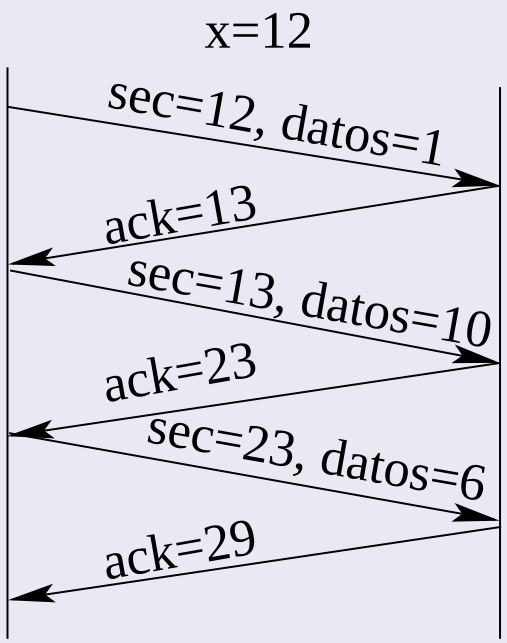
TCP: protocolo de control de transmisión

Transmisión fiable

- Aplicación de los principios de transmisión fiable

Números de secuencia

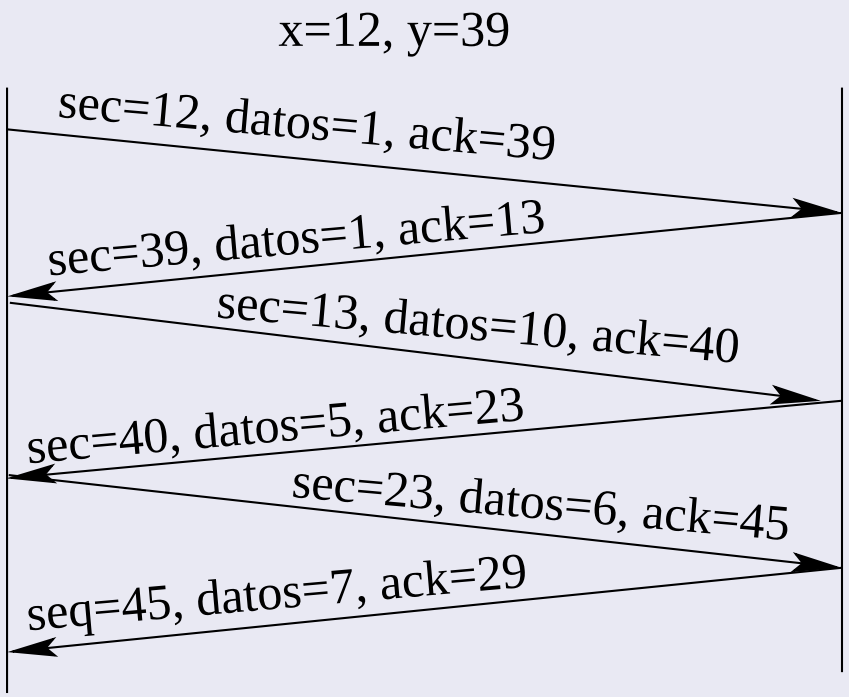
- Números de 32 bits que identifican bytes (no segmentos)
  - Se empieza por x (aleatorio)
  - Se incrementa en $x + nbytes$
  - Los ACKs indican el siguiente byte que se desea recibir



TCP: protocolo de control de transmisión

Superposición (*piggybacking*)

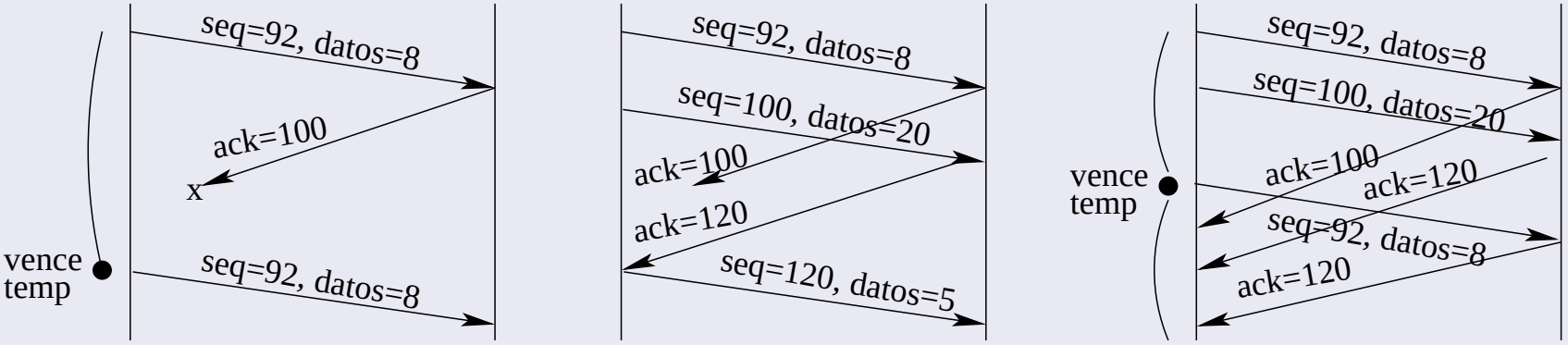
- En un mismo segmento ACK, se transmiten también datos
- Tiempo de espera máximo: si no tiene datos, transmite sólo el ACK



TCP: protocolo de control de transmisión

Transferencia fiable de datos

- El emisor usa temporizadores para la retransmisión
- Se usa ventana deslizante y ACKs acumulativos
- Se recomienda usar un temporizador único: cuando llega un ACK se reinicia
- Si un ACK no llega a tiempo, se retransmite solo ese segmento no confirmado, se reinicia el temporizador y se espera un ACK



perdida de ACK — ACK acumulativos — solo se retransmite uno

TCP: protocolo de control de transmisión

Estimación del tiempo de espera

$$\begin{aligned} \text{Temporizador} &= \text{EstimacionRTT} + 4\text{DevRTT} \\ \text{EstimacionRTT} &= (1-\alpha)\text{EstimacionRTT} + \alpha\text{MuestraRTT} \\ \text{DevRTT} &= (1-\beta)\text{DevRTT} + \beta|\text{MuestraRTT} - \text{EstimacionRTT}| \end{aligned}$$

- DevRTT es una medida de la variación del RTT y, en general, $\alpha = 0,125$ y $\beta = 0,25$
- La estimación del RTT es en base a segmentos transmitidos y confirmados (no retransmitidos)

Duplicación del tiempo de espera

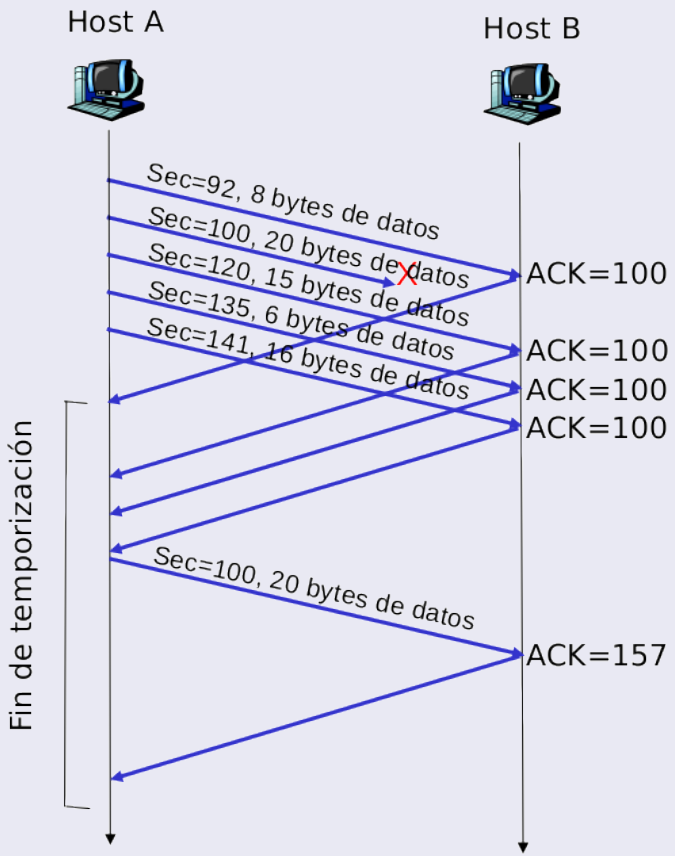
- Cuando se produce la expiración del temporizador, el emisor duplica el tiempo

TCP: protocolo de control de transmisión

Retransmisión rápida

Retransmisión de un paquete antes de que expire el temporizador

- 3 ACKs duplicados se interpretan como un NAK (el 4º)
- Se recibe un segmento con mayor número del esperado $\Longrightarrow$ el receptor envía un ACK duplicado
- Se repite la situación $\Longrightarrow$ el receptor envía otro ACK repetido $\Longrightarrow$ el emisor recibe 3 ACKs repetidos y realiza una *retransmisión rápida*



TCP: protocolo de control de transmisión

Pérdidas en TCP

- Expiración del temporizador: problema grave, se pierden los segmentos y los ACKs
- 3 ACKs duplicados: problema leve, al menos llegan los ACKs $\Longrightarrow$ *retransmisión rápida*

ARQ intermedio entre retroceder *N* y repetición selectiva

- Retroceder *N*: ACKs acumulativos
- Repetición selectiva: aceptan segmentos fuera de orden y se retransmiten solo los necesarios

TCP: protocolo de control de transmisión

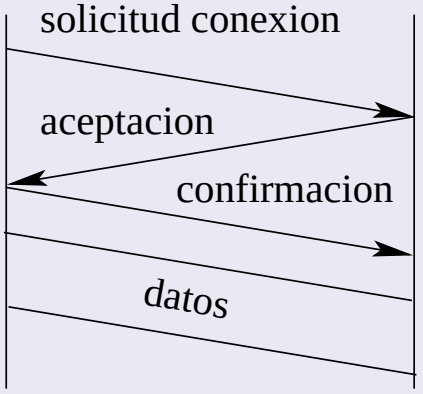
Control de flujo

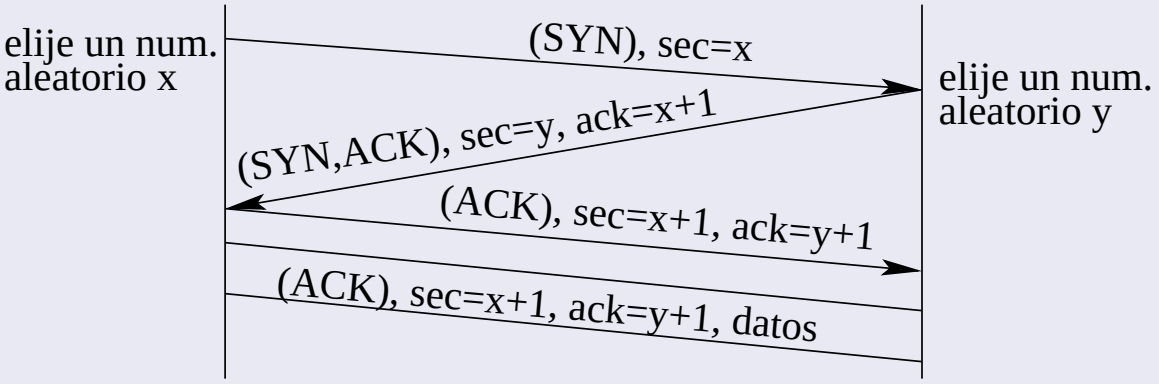
Mecanismo que permite al receptor indicar al emisor el ritmo al que puede recibir datos

- En el momento de la conexión, el receptor indica el tamaño de su ventana de recepción
- El emisor fija su ventana de envío a este valor
- Para ello existe un campo en la cabecera TCP
- El tamaño de ventana se puede modificar en cada transmisión

TCP: protocolo de control de transmisión

Conexión: acuerdo en tres fases

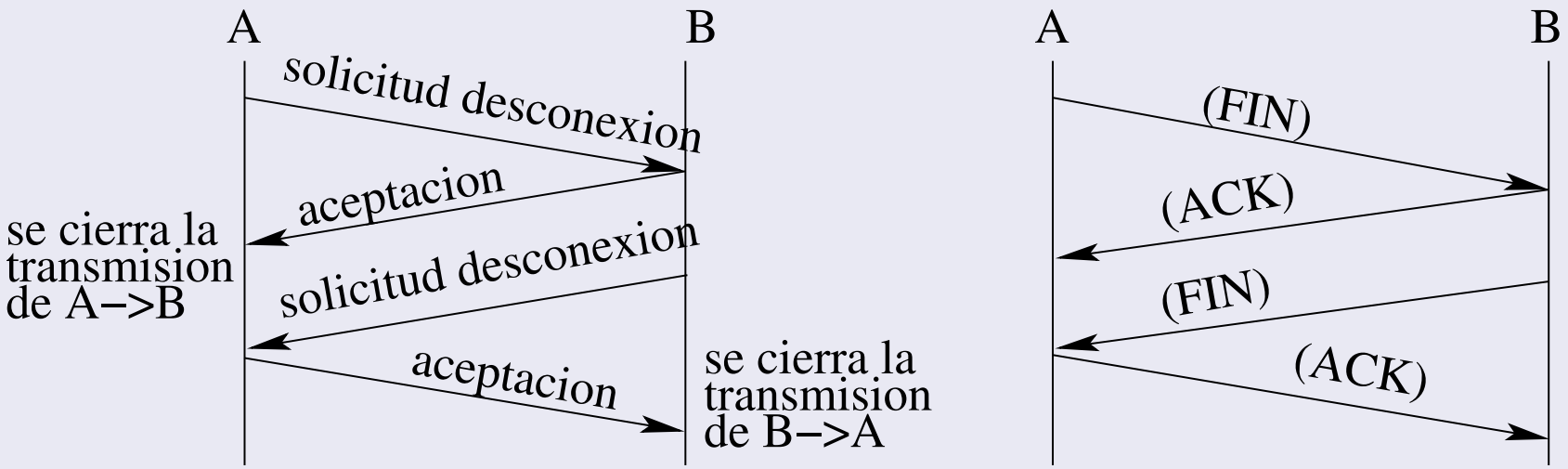




- SYN = 1, cuando se envía por primera vez x o y
- ACK = 1, cuando se confirma un segmento
- En la tercera fase ya se pueden enviar datos

TCP: protocolo de control de transmisión

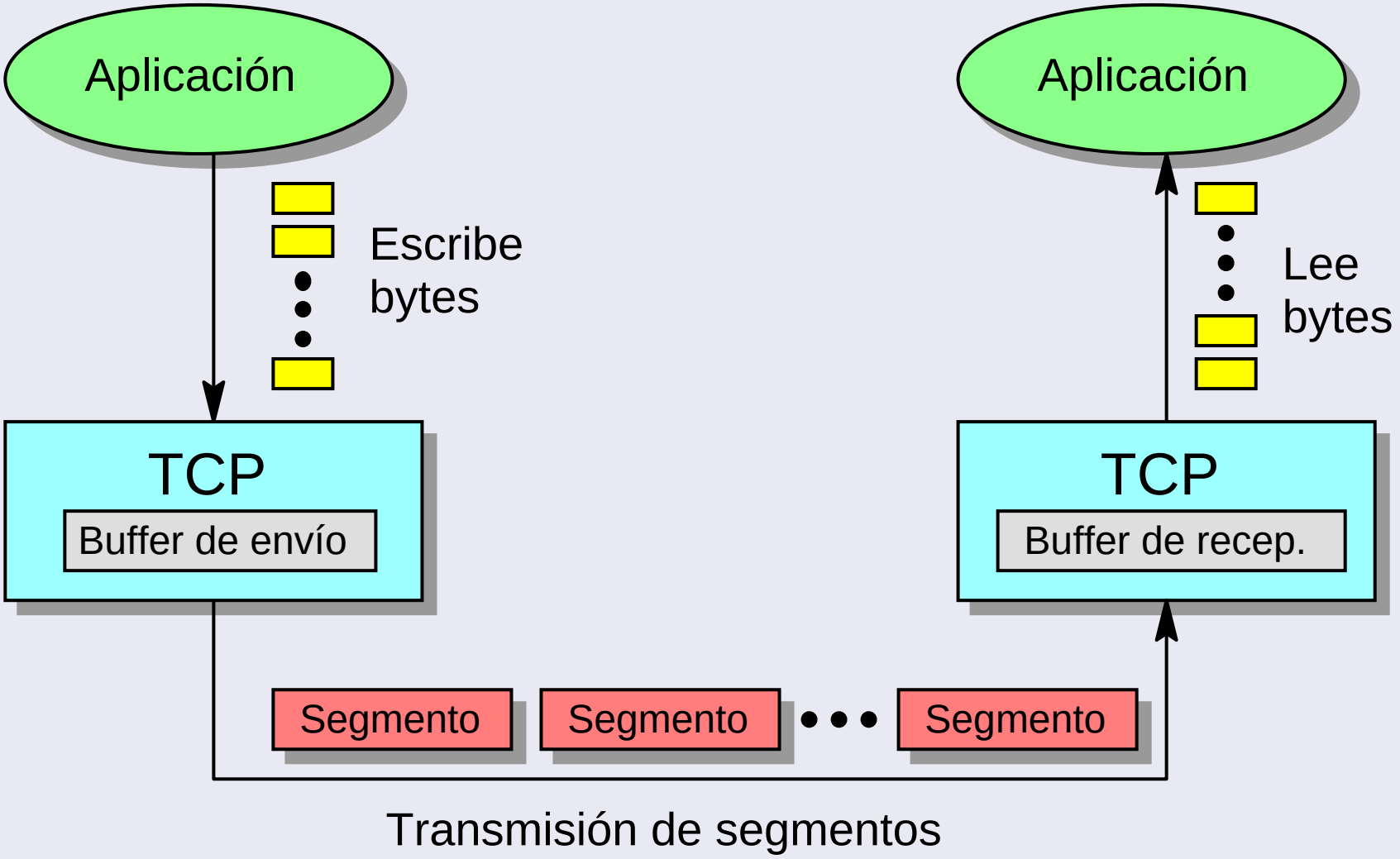
Desconexión



- En dos fases: cada una para desconectar la transmisión en un sentido
- Se podría desconectar en un sentido y seguir transmitiendo en el otro
- FIN = 1, solicitud de desconexión
- ACK = 1, aceptación

TCP: protocolo de control de transmisión

Transmisión de datos



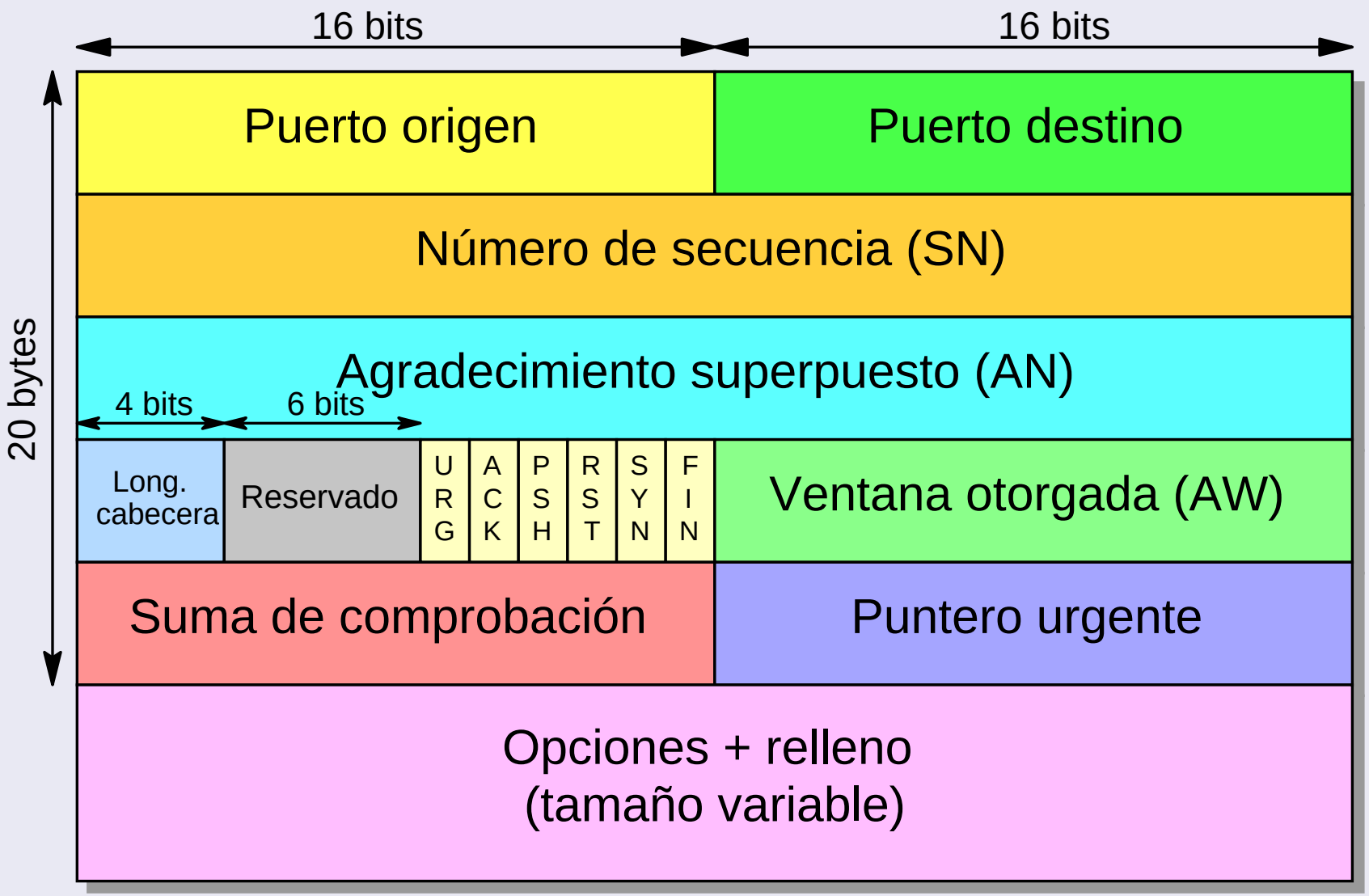
TCP: protocolo de control de transmisión

Transmisión de datos

- Una vez establecida la conexión empieza la transmisión
- La aplicación va pasando los datos a la capa de transporte (escribiendo en el socket) que los va acumulando
- Mecanismos para disparar la transmisión de un segmento:
  - Segmentación: cuando el número de bytes supere al MSS (*maximun segment size*)
  - Cuando la aplicación fuerza el envío (*push*)
  - Cuando un temporizador llega a 0
- TCP genera el segmento y se lo pasa a IP

TCP: protocolo de control de transmisión

Estructura de la cabecera TCP



Índice

1. Introducción
2. UDP: protocolo de datagramas de usuario
3. Fundamentos de la transmisión fiable
4. TCP: protocolo de control de transmisión
5. Control de congestión

Control de congestión

Asignación de recursos

- Los recursos de la red se deben repartir entre las diferentes peticiones

Congestión

- Demasiados paquetes en la red producen retardos en las transmisiones y pérdida de muchos paquetes
- Usualmente por el desbordamiento de la memoria de los routers

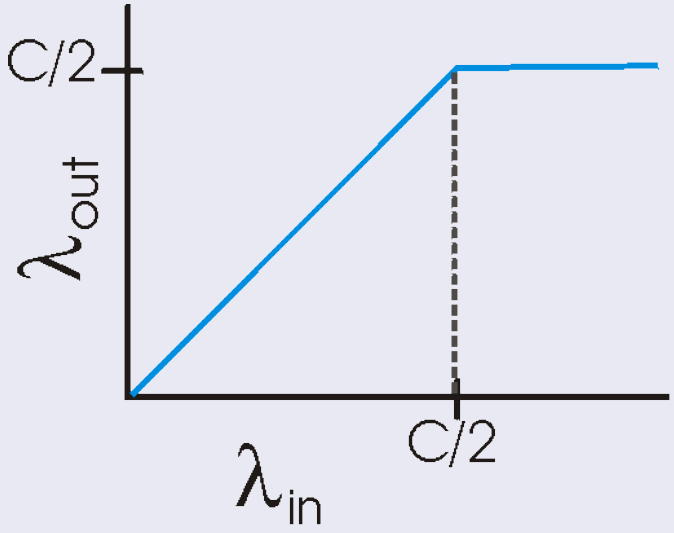
Control de congestión

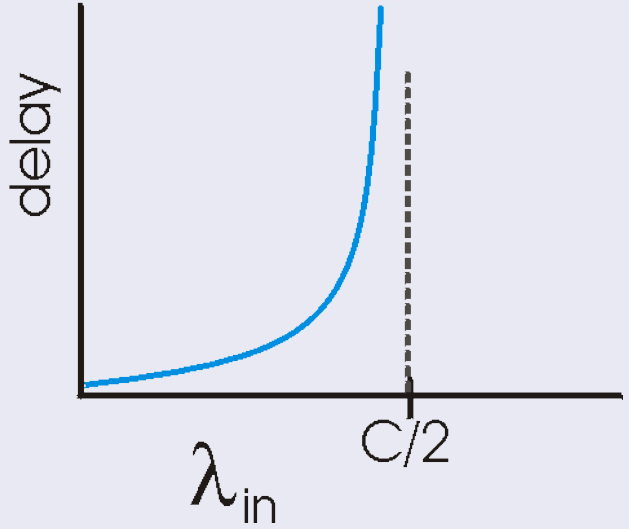
- Esfuerzos realizados por los elementos de la red para prevenir o responder a situaciones de congestión
  - Pre-reservar recursos para evitar congestión
  - Dejar que la congestión ocurra y resolverla entonces

Control de congestión

Origen de la congestión

- Escenario 1: dos emisores, un router con capacidad infinita y enlace compartido de velocidad C
  - Tasa de transmisión entre 0 y C/2: todo se recibe OK y con retardo finito
  - Tasa de transmisión mayor que C/2: el enlace no puede proporcionar paquetes a esa velocidad ⇒ paquetes en la cola del router y aumenta el retardo

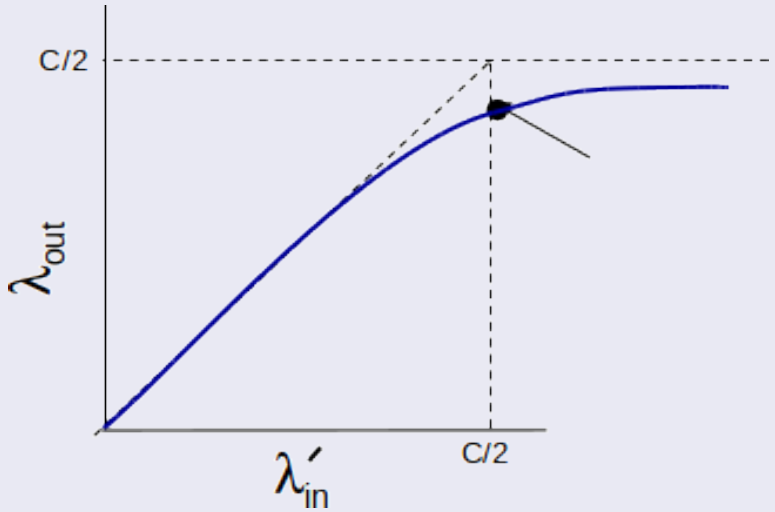




Control de congestión

Origen de la congestión

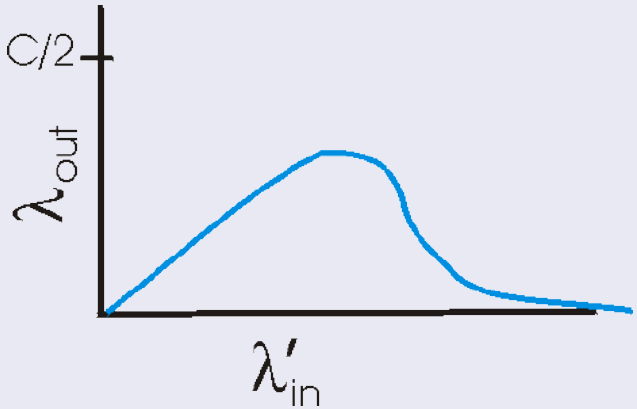
- Escenario 2: dos emisores, un router con memoria finita y enlace compartido de velocidad C
  - λ'_{in}: carga ofrecida, contiene datos originales y retransmitidos
  - Tasa de transmisión entre 0 y C/2: todo se recibe OK y con retardo finito
  - Tasa de transmisión mayor que C/2: la tasa entregada disminuye porque algunos son duplicados



Control de congestión

Origen de la congestión

- Escenario 3: varios emisores, routers con memoria finita y varios enlaces
  - Tasa de transmisión pequeña: todo se recibe OK y con retardo finito



  - Tasa de transmisión elevada: los buffers de los routers se llenan y la tasa entregada disminuye
    - En el límite tiende a cero

Necesidad de mecanismos de control de congestión

En TCP/IP el control de congestión recae principalmente en TCP

Control de congestión en TCP

Mecanismo en TCP

- El emisor adecúa su tasa de envío en función de la congestión que percibe
- Considera que hay congestión si:
  - Expira un temporizador
  - Se reciben 3 ACKs duplicados

Procedimiento

- En el emisor se definen: la *ventana de congestión* y el RTT
- El RTT se estima periódicamente (tiempo desde el envío de un segmento hasta que llega su ACK)
- Se actualiza la ventana de congestión según el caso

$$\text{tasa de envío} = \frac{\text{ventana de congestión}}{RTT} \text{ bytes/segundo}$$

Control de congestión en TCP

Mecanismos para actualizar la ventana de congestión

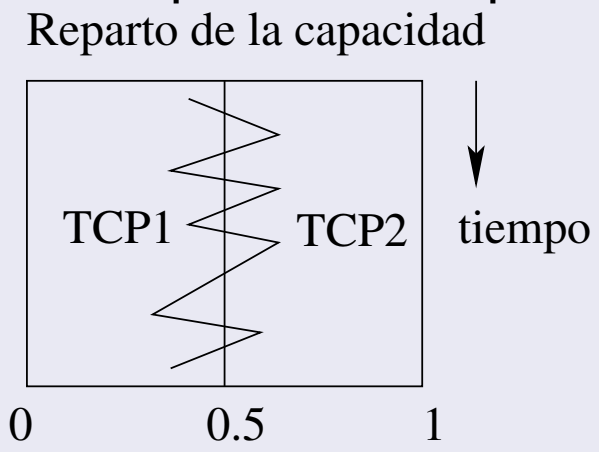
- Inicio lento: determina la capacidad inicial de la red
- Incremento aditivo/decremento multiplicativo (AIMD): situación normal
- Recuperación rápida: evita la fase de inicio lento cuando hay congestión

Control de congestión en TCP

Imparcialidad

Dos aplicaciones compartiendo un enlace de capacidad limitada

- Dos conexiones TCP se reparten la capacidad del enlace



- Una conexión TCP y una UDP, la UDP acaparará la mayor parte de la capacidad
- Si una aplicación usa una conexión TCP y la otra 9 conexiones TCP paralelas, la capacidad será 1/10, 9/10

Redes

Tema 4: Capa de red

José Carlos Cabaleiro Domínguez

Escola Técnica Superior de Enxeñería

Índice

1. Introducción
2. Redes de conmutación de paquetes
3. Algoritmos de encaminamiento
4. Encaminamiento en Internet
5. Protocolo de Internet
6. ICMP: Protocolo de mensajes de control de Internet
7. DHCP: Protocolo de configuración dinámica de hosts
8. NAT: Traducción de direcciones de red

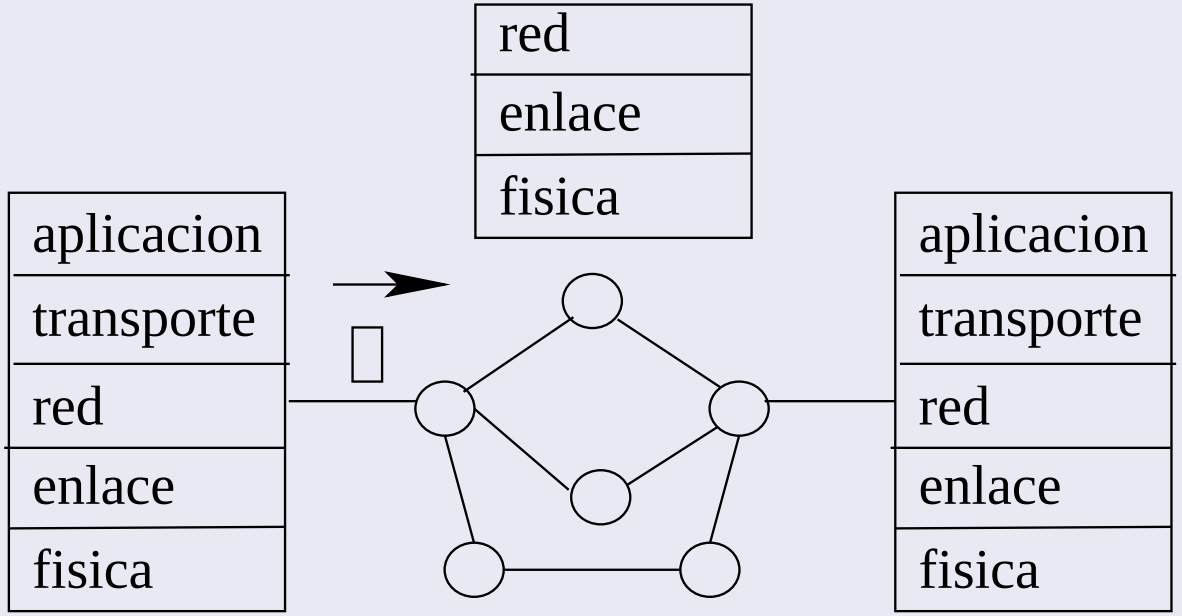
Índice

1. Introducción
2. Redes de conmutación de paquetes
3. Algoritmos de encaminamiento
4. Encaminamiento en Internet
5. Protocolo de Internet
6. ICMP: Protocolo de mensajes de control de Internet
7. DHCP: Protocolo de configuración dinámica de hosts
8. NAT: Traducción de direcciones de red

Introducción

Capa de red

- Se encarga de llevar los paquetes que le pasa la capa de transporte del host origen al host destino
- Implementada tanto en los sistemas finales como en los routers



Introducción

Conceptos

- Reenvío (*forwarding*)
  - Cuando llega un paquete a un router, el router hace pasar el paquete a la interfaz de salida apropiada
- Encaminamiento o rutado (*routing*)
  - Determinar la ruta que debe seguir un paquete que se envía desde un emisor a un receptor
  - Algoritmos de encaminamiento
- Tablas de reenvío
  - Tablas que almacenan la información necesaria para el reenvío de paquetes
    - Asigna el valor del campo de la cabecera a la interfaz de salida apropiada
    - Los algoritmos de encaminamiento determinan los valores de las tablas de reenvío

Índice

1. Introducción
2. Redes de conmutación de paquetes
3. Algoritmos de encaminamiento
4. Encaminamiento en Internet
5. Protocolo de Internet
6. ICMP: Protocolo de mensajes de control de Internet
7. DHCP: Protocolo de configuración dinámica de hosts
8. NAT: Traducción de direcciones de red

Redes de conmutación de paquetes

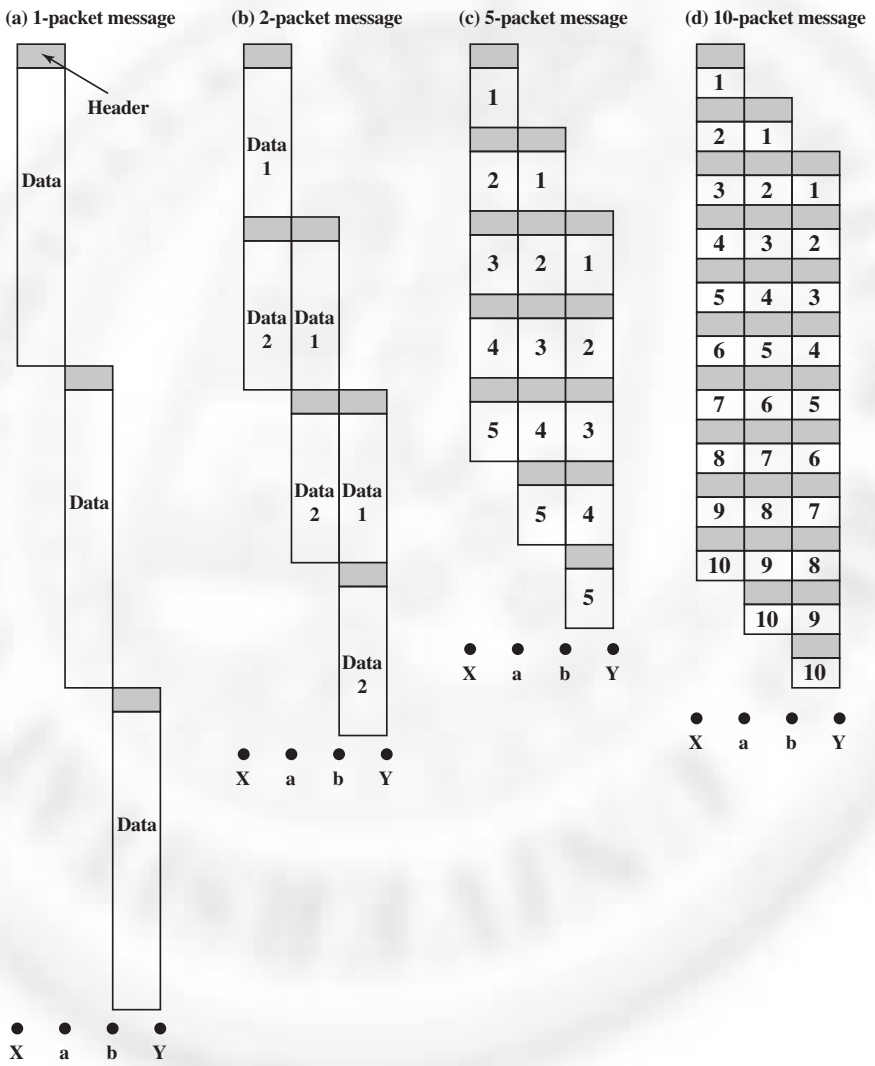


Figure 10.2 Effect of Packet Size on Transmission Time

Redes de conmutación de paquetes

Redes de datagramas

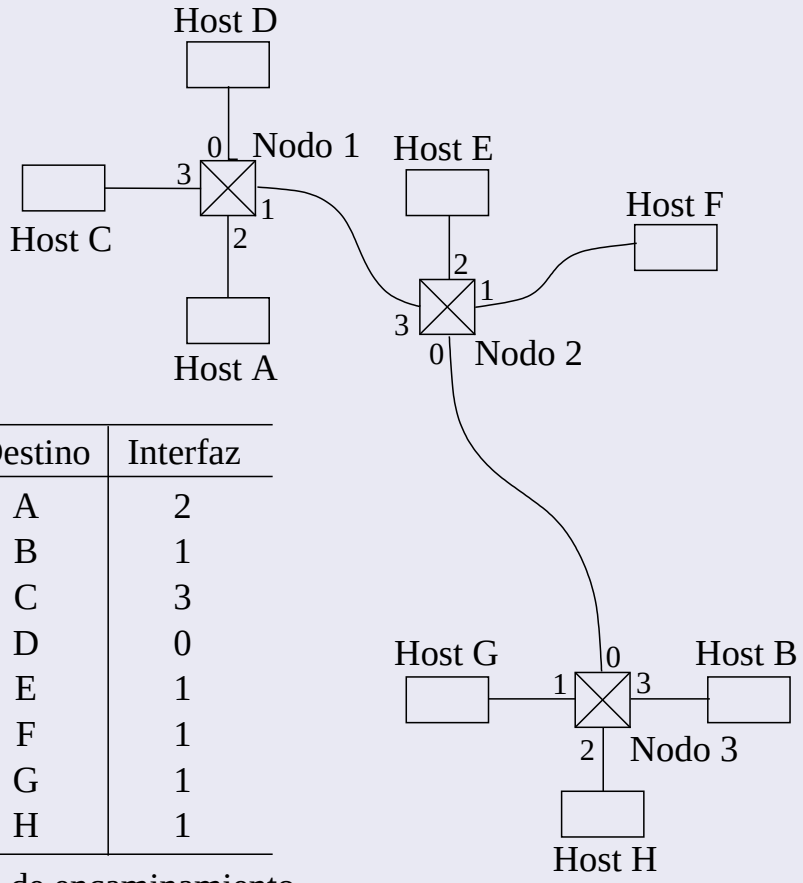
- Cada paquete incluye en la cabecera la IP destino
- Reenvío: el router examina la cabecera y lo coloca en la salida más apropiada (tabla de reenvío)
- No mantienen información de estado: una secuencia de paquetes se encaminan de forma independiente

Redes de circuitos virtuales

- Se establece la conexión planificando una ruta al destino: un circuito virtual (CV)
- A cada paquete se le escribe el identificador de CV: los routers lo usan para el reenvío
- Los routers mantienen información de estado (tabla de circuitos virtuales)

Redes de datagramas

Ejemplo



Destino	Interfaz
A	3
B	0
C	3
D	3
E	2
F	1
G	0
H	0

Tabla de encaminamiento del nodo 2

Destino	Interfaz
A	2
B	1
C	3
D	0
E	1
F	1
G	1
H	1

Tabla de encaminamiento del nodo 1

Destino	Interfaz
A	0
B	3
C	0
D	0
E	0
F	0
G	1
H	2

Tabla de encaminamiento del nodo 3

Introducción

Capa de red en Internet: IP

- Tipo datagrama: se encaminan los paquetes en función de la dirección destino que contienen
- Sin estado: cada paquete se trata de forma independiente
- Red no fiable: servicio de mejor esfuerzo.
  - No se garantiza la entrega de paquetes (ni el orden)
  - Ni la entrega en un tiempo determinado
  - Entrega el mayor número de paquetes, aunque algunos se pierdan
- Permite la interconexión de redes de diferentes tecnologías (interred)

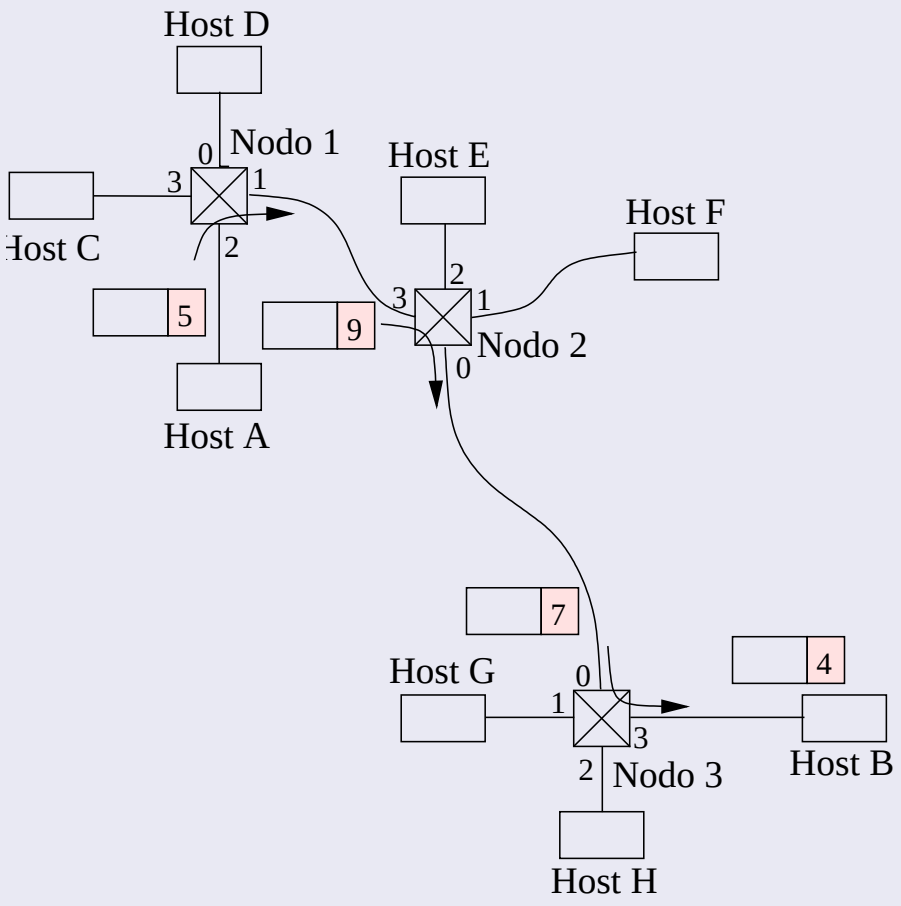
Redes de circuitos virtuales

Tabla de circuitos virtuales

- Cada nodo mantiene una tabla de encaminamiento con la siguiente información:
  - Interfaz de entrada del circuito virtual
  - Identificador del circuito virtual **VCI**
  - Interfaz de salida por la que los paquetes de ese circuito virtual dejan el nodo
  - Identificador de salida del circuito virtual
- Un paquete que llega por una interfaz con un VCI, se coloca en la interfaz indicada en la tabla con el nuevo VCI

Redes de circuitos virtuales

Ejemplo



Nodo 1			
Entrada		Salida	
Interfaz	VCI	Interfaz	VCI
2	5	1	9
.	.	.	.
.	.	.	.
.	.	.	.

Nodo 2			
Entrada		Salida	
Interfaz	VCI	Interfaz	VCI
3	9	0	7
.	.	.	.
.	.	.	.
.	.	.	.

Nodo 3			
Entrada		Salida	
Interfaz	VCI	Interfaz	VCI
0	7	3	4
.	.	.	.
.	.	.	.
.	.	.	.

Redes de circuitos virtuales

Construcción de la tabla de VC

- A envía una *Petición de llamada* dirigida a B
- Esta llega al nodo 1 por la interfaz 2, que debe saber como reenviarla para que llegue a B (similar a datagramas)
- El nodo 1 decide marcar esta petición con un VCI de 5 (aleatorio), y la envía por la interfaz 1
- 2 lo recibe, lo marca con VCI 9 y lo coloca en la interfaz 0; 3 lo marca con VCI 7 y lo coloca en 3; y B lo marca con VCI 4, que identificará los paquetes de A
- B devuelve una *Llamada aceptada* con VCI 4 al nodo 3 por el interfaz 3
- El nodo 3 puede completar su entrada en la tabla (VCI salida = 4); lo mismo los nodos 2 y 1
- 1 manda el ACK a A que lo recibe con VCI 5
  - $\Rightarrow$ A marca el resto de paquetes a B con VCI 5

Índice

1. Introducción
2. Redes de conmutación de paquetes
3. Algoritmos de encaminamiento
4. Encaminamiento en Internet
5. Protocolo de Internet
6. ICMP: Protocolo de mensajes de control de Internet
7. DHCP: Protocolo de configuración dinámica de hosts
8. NAT: Traducción de direcciones de red

Algoritmos de encaminamiento

Conceptos

- Algoritmo de encaminamiento o rutado: el encargado de encontrar el camino mínimo entre el origen y destino
  - Cada host está conectado a un router (*router por defecto*)
  - El problema se limita a encontrar el camino mínimo entre routers
- Equivalente a encontrar el camino mínimo en un grafo
  - Routers: nodos del grafo
  - Enlaces: aristas del grafo
    - Se les asigna un peso (coste)
    - Coste: distancia, velocidad, carga del enlace, coste económico, etc.

Algoritmos de encaminamiento

Clasificación

- Globales. De estado de los enlaces
  - Cada nodo dispone de toda la información sobre la red: todos los nodos y el coste de todos los enlaces
  - A partir de esta información, cada nodo puede calcular su tabla de encaminamiento (o de rutas)
- Descentralizados. De vector de distancias
  - El cálculo de los caminos mínimos se hace en colaboración de todos los nodos
  - Los nodos intercambian información sólo con sus vecinos
  - Solo conocen la distancia a los demás nodos y por dónde empezar

Algoritmos de encaminamiento

Clasificación

- Estáticos o dinámicos
  - Estáticos: solo cambian cuando cambia la topología de red o se modifican manualmente parámetros
  - Dinámicos: se ejecutan periódicamente de forma automática. Los usados actualmente en Internet
- Sensibles o insensibles a la carga
  - Sensibles: el coste de los enlaces varía dinámicamente
    - Pueden provocar que los mensajes queden atrapados en un ciclo
  - En Internet son insensibles a la carga

Encaminamiento de estado de los enlaces

Algoritmo de Dijkstra (1959)

- Busca el camino más corto entre dos vértices de un grafo pesado
- Variante del algoritmo, *forward search*:
  - El nodo `N` quiere calcular su tabla de routing a partir de los LSP (*link state packet*) que ha recibido
  - Cada nodo tiene 2 listas: `Confirmado` y `Provisional`
  - Cada elemento de las listas indican el coste para alcanzar un nodo y el siguiente salto
    - `(M,5,L)`, indica que de `N` se alcanza `M` a coste 5 a través de L
  - Inicializa la tabla `Confirmado` con una entrada para `N` con coste 0 (`(N,0,-)`)
  - Sigue el siguiente algoritmo:

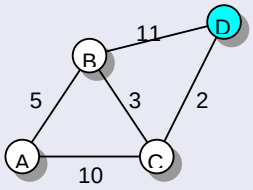
Encaminamiento de estado de los enlaces

Algoritmo de Dijkstra

1. Para el último nodo añadido a `Confirmado` (nodo `S`) examina su LSP
2. Para cada vecino (`V`) de `S`, calcula el coste (`Coste`) para alcanzar `V` como la suma del coste de `N` a `S` y de `S` a `V`
   - Si `V` no está en ninguna lista, añádelo a la lista `Provisional` de la forma `(V,Cost,SigSalto)`
   - Si `V` está en `Provisional`, y `Coste` es menor que el indicado, reemplázalo por `(V,Cost,SigSalto)`
3. Si `Provisional` está vacía, acaba; si no pasa la entrada de `Provisional` con menor coste a `Confirmado`
4. Vuelve al paso 1

Encaminamiento de estado de los enlaces

Algoritmo de Dijkstra



Destino	Coste	SigSalto
A	10	C
B	5	C
C	2	C

Tabla de routing del nodo D

Paso	Confirmado	Provisional	Comentarios
1	(D,0,−)		D es el único elemento inicial de **Confirmado**
2	(D,0,−)	(B,11,B) (C,2,C)	El LSP de D dice que puede alcanzar B a coste 11, y C a coste 2. Lo pone en **Provisional**.
3	(D,0,−) (C,2,C)	(B,11,B)	Pasa el miembro de **Provisional** con menor coste (C) a **Confirmado**, y examina su LSP.
4	(D,0,−) (C,2,C)	(B,5,C) (A,12,C)	El coste de alcanzar B a través de C es 5, asi que reemplaza (B,11,B) por (B,5,C). El LSP de C indica qu puede alcanzar A con coste 10+ 2 a través de C.
5	(D,0,−) (C,2,C) (B,5,C)	(A,12,C)	Pasa el miembro de **Provisional** con menor coste (B) a **Confirmado**, y examina su LSP.
6	(D,0,−) (C,2,C) (B,5,C)	(A,10,C)	El LSP de B dice que puede alcanzar A a coste 5, así que cambia (A,12,C) por (A,10,C) (el coste D−B es 5 a través de C)
7	(D,0,−) (C,2,C) (B,5,C) (A,10,C)		Pasa el miembro de **Provisional** con menor coste (A) a **Confirmado**, y ya está.

Encaminamiento de estado de los enlaces

Complejidad algorítmica del algoritmo de Dijkstra

- Complejidad cuadrática, $O(n^2)$

Ventajas

- Se estabiliza rápidamente
- No genera mucho tráfico
- Responde rápidamente a cambios en la topología o fallos de nodos

Inconvenientes

- La cantidad de información (LSPs) almacenada en cada nodo puede ser bastante grande $\Rightarrow$
  - Problemas de escalabilidad

Encaminamiento de vector de distancias

Algoritmo descentralizado

- Todos los nodos colaboran

Funcionamiento

- Inicialmente, los nodos solo conocen el coste a sus vecinos
- Iterativamente, los nodos comunican a sus vecinos todo lo que saben
- Los nodos computan distancias a nuevos nodos o actualizan las que tienen con menores valores
- Las actualizaciones continúan hasta que converge
- La información que un nodo z comunica a sus vecinos son las distancias $d_{z,i}$
- Se almacena el vecino que envió dicha información (siguiente salto)

Encaminamiento de vector de distancias

Funcionamiento

Sea el nodo x con un vecino z, cuyo enlace tiene coste $c_{x,z}$, y que z envía $d_{z,y}$ $\Rightarrow$

$D_{x,y}(z) =$	$c_{x,z}$ $+$	$d_{z,y}$
distancia de x a y a través de la salida z	coste del enlace que une x con z	distancia de z a y

Tabla de distancias del nodo x

		distancia por	
		z	z'
destino	y	$D_{x,y}(z)$	$D_{x,y}(z')$
	y'	$D_{x,y'}(z)$	$D_{x,y'}(z')$

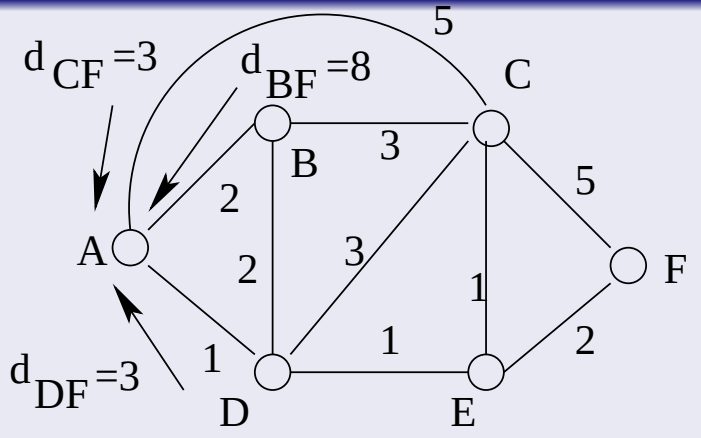
Encaminamiento de vector de distancias

Ejemplo

- Nodos B, C y D envían a A la distancia a F
- A calcula las distancias

$$D_{A,F}(B) = c_{A,B} + d_{B,F} = 2 + 8 = 10$$
$$D_{A,F}(C) = c_{A,C} + d_{C,F} = 5 + 3 = 8$$
$$D_{A,F}(D) = c_{A,D} + d_{D,F} = 1 + 3 = 4$$

- La tabla de distancias para A



		distancia por B	C	D
destino	...	...	...	...
	F	10	8	4

Encaminamiento de vector de distancias

Ejemplo

- A comunica a sus vecinos la distancia a F, $d_{A,F} = \min_z D_{A,F}(z) = 4$
- Después de ciertas iteraciones, converge

		distancia por		
		B	C	D
destino	B	2	8	3
	C	5	5	3
	D	4	7	1
	E	5	6	2
	F	7	8	4

Tabla de distancias de A

destino	salida
B	B
D	D
C	D
E	D
F	D

Tabla de rutas de A

Encaminamiento de vector de distancias

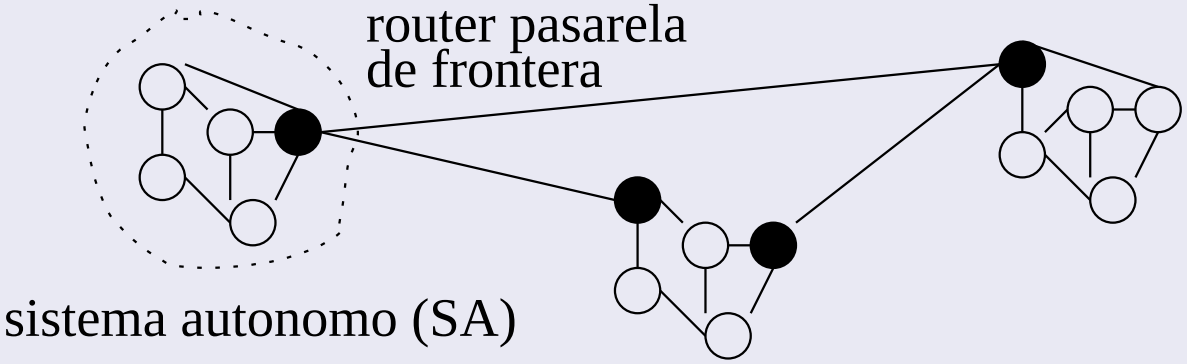
Características

- Intercambio de actualizaciones con los vecinos
  - Periódicamente
  - Cuando un cambio en su tabla o en el coste de un enlace
    - Disminución del coste de un enlace: actualización rápida de las tablas
    - Aumento del coste de un enlace: algunos problemas. Se resuelve con diversas técnicas: horizonte dividido e inverso envenenado
- Iterativo: puede necesitar muchas iteraciones. Peor que EE.
- Menos robusto que EE: si un nodo calcula mal sus distancias, todos usarán esos valores incorrectos

Encaminamiento jerárquico

Sistemas autónomos (SA)

- Regiones en las que se dividen las redes grandes como Internet
- Operados por empresas u organismos
- Los routers solo conocen en encaminamiento en su región
- *Routers pasarela frontera*: centralizan el tráfico de salida del SA



- Dos niveles de encaminamiento:
  - Intradominio: cada SA puede elegir algoritmo
  - Interdominio: común para todos los SA

Índice

1. Introducción
2. Redes de conmutación de paquetes
3. Algoritmos de encaminamiento
4. Encaminamiento en Internet
5. Protocolo de Internet
6. ICMP: Protocolo de mensajes de control de Internet
7. DHCP: Protocolo de configuración dinámica de hosts
8. NAT: Traducción de direcciones de red

Encaminamiento en Internet

Protocolos usados en Internet

- Intradominio o internos al SA (IGP): RIP y OSPF
- Interdominio o entre SAs: BGP

Categoría	Protocolo	Tipo	Protocolos transporte/red
intra-autónomo	RIP OSPF	VD EE	UDP/IP (puerto 520) propio/IP (puerto 89)
inter-autónomo	BGP	VD	TCP/IP (puerto 179)

- Protocolos de la capa de aplicación

RIP (*Routing Information Protocol*)

Protocolo de información de encaminamiento

- Encaminamiento intradominio basado en vector de distancias
- Considera que el coste de los enlaces es 1 y distancia máxima 15
- Mensajes RIP solo a los nodos vecinos
  - Mensajes de petición RIP: solicitan información
  - Mensajes de respuesta RIP: lista de hasta 25 redes internas al SA
    - También de forma automática cada 30 s. a sus vecinos. Si no se recibe respuesta en 180 s., se considera caído

OSPF (*Open Shortest Path First*)

Protocolo abierto de primero el camino más corto

- Abierto: algoritmo libre
- Encaminamiento intradominio basado en estado de los enlaces
- Más avanzado que RIP, pensado para reemplazarlo
- Mensajes OSPF se difunden a todos los nodos
  - Cuando se producen cambios
  - Periódicamente, al menos cada 30 minutos
  - Cada router obtiene información completa del SA
- Mensajes HELLO a cada vecino, para comprobar que el enlace está OK
- Posibilidad de interrogar a un vecino para obtener la toda la información
- El coste de los mensajes los pone el administrador

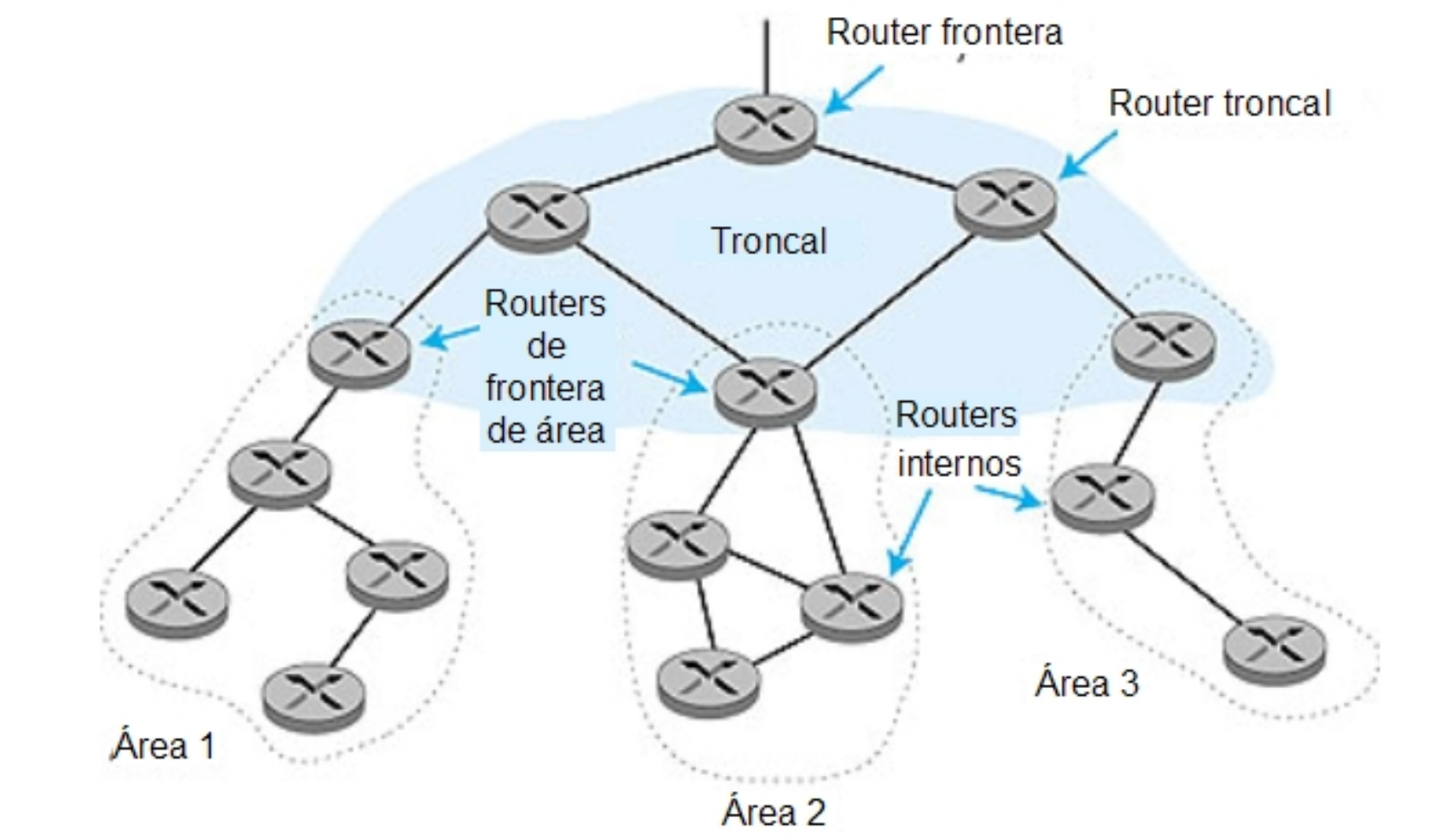
OSPF (*Open Shortest Path First*)

Protocolo abierto de primero el camino más corto

- Seguridad: implementa un protocolo de transporte propio
  - Con todos los mensajes autentificados
  - Solo considera los routers autenticados
- Múltiples caminos de mismo coste: permite repartir el tráfico
- Soporte de jerarquía: permite subdividir los SA en áreas
  - Cada área ejecuta OSPF sobre los routers de ese área
  - Las áreas se comunican entre sí mediante los routers de frontera de área
    - Paquetes con destino fuera del área se encaminan al router de frontera de área
    - Interconectados entre sí en un área troncal
  - En el SA existe un router de frontera para salir fuera del SA

OSPF (*Open Shortest Path First*)

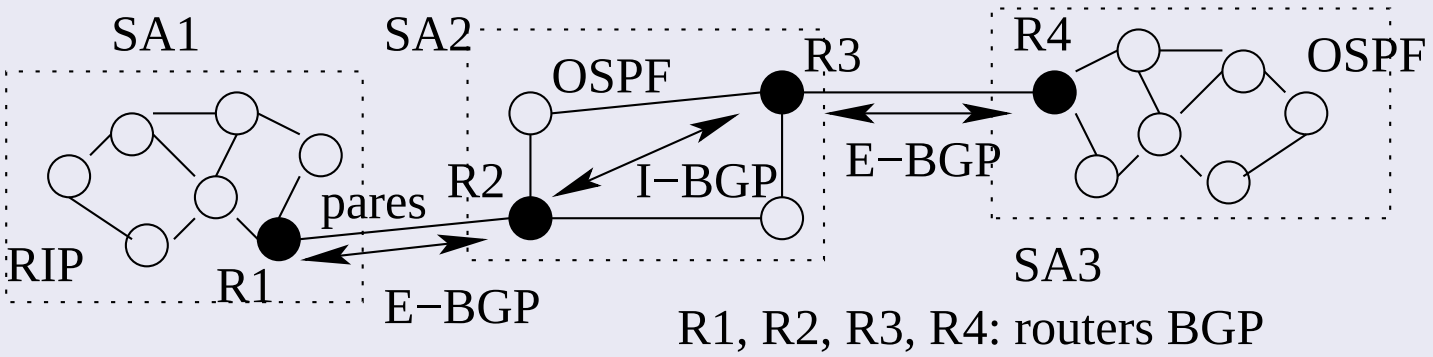
OSFP jerárquico



BGP (*Border Gateway Protocol*)

Protocolo de pasarela de frontera

- Protocolo interdominio estándar en Internet (BGP4)
- Comunica routers pasarela de frontera



- Cada SA puede usar el protocolo intradominio que desee
- Dos tipos de comunicaciones:
  - Entre routers BGP vecinos, *pares* BGP. Se usa E-BGP
  - Entre routers del mismo SA, considerados vecinos lógicos. Se usa I-BGP

BGP (*Border Gateway Protocol*)

Protocolo de pasarela de frontera

- Protocolo similar al de vector de distancias, se intercambian rutas completas (vector de rutas)
- Cada SA se identifica por un número de sistema autónomo único
- Los administradores pueden decidir las políticas de encaminamiento

red destino	métrica
x	4

Ejemplo de mensaje en RIP

red destino	ruta
x	SA1/SA2/SA3/S4

Ejemplo de mensaje en BGP

Índice

1. Introducción
2. Redes de conmutación de paquetes
3. Algoritmos de encaminamiento
4. Encaminamiento en Internet
5. Protocolo de Internet
6. ICMP: Protocolo de mensajes de control de Internet
7. DHCP: Protocolo de configuración dinámica de hosts
8. NAT: Traducción de direcciones de red

IP (*Internet Protocol*)

Protocolo de Internet

- Protocolo basado en datagramas: servicio sin conexión
- La fiabilidad recae en capas superiores (TCP)
- Diseñado para interconectar redes diferentes

Componentes de la capa de red en Internet

- Protocolo de red: IP
  - Define el formato de las direcciones
  - Formato de los datagramas
  - Acciones de los routers en base a los campos de los datagramas
- Protocolo de encaminamiento
- Protocolo de mensajes de control de Internet (ICMP)

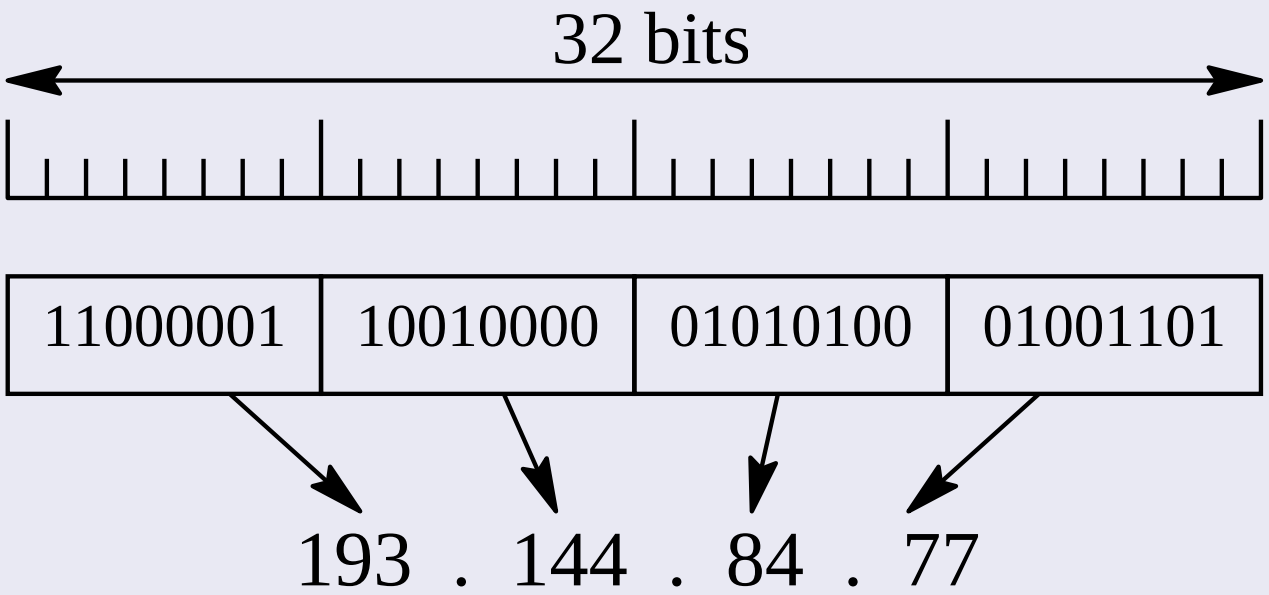
IP (*Internet Protocol*)

Direccionamiento IPv4

- Los nodos en una red tienen una dirección IP por interfaz
- Interfaz: la unión de un host o router con un enlace

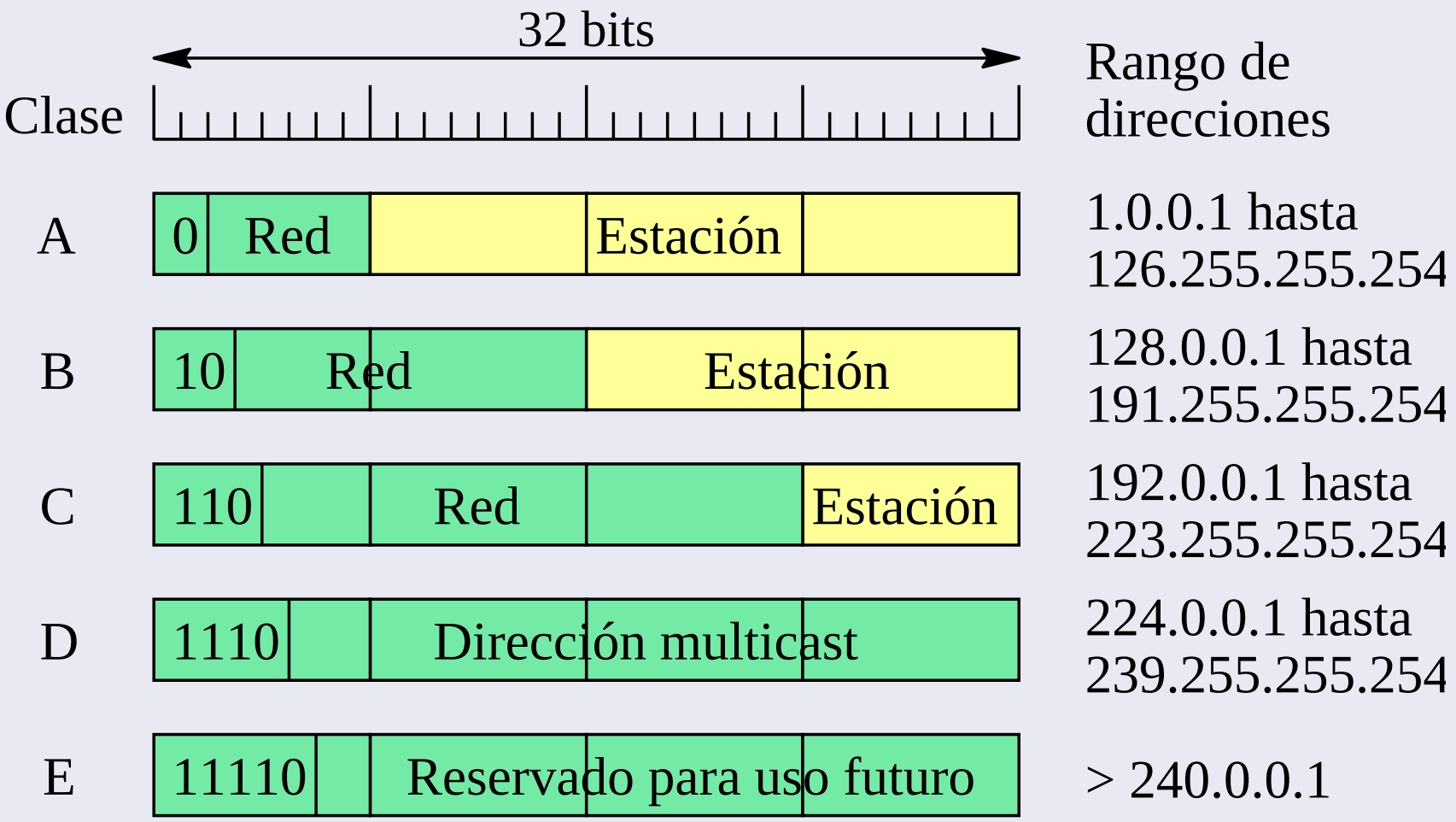
Codificación de la dirección IP

- En IPv4, cada dirección IP se codifica mediante 4 bytes, generalmente escritos en notación decimal



IP (*Internet Protocol*)

Direccionamiento IPv4



Direccionamiento IPv4

Proporción redes-máquinas

- Restricción: un campo de red o de host no puede estar todo a 1s o todo a 0s
- Según esto tenemos:

  Clase A: 126 ($2^7 - 2$) redes con $\approx$ 16 millones de estaciones cada una ($2^{24} - 2 = 16777214$)

  Clase B: 16384 (2^{14}) redes con 65534 ($2^{16} - 2$) estaciones cada una

  Clase C: $\approx$ 2 millones de redes ($2^{21} = 2097152$) con 254 estaciones cada una ($2^8 - 2$)

Direccionamiento IPv4

Direcciones especiales reservadas

- Identificación de redes: el nº de red y el resto a 0
- Ejemplos:
  - Clase A → 10.0.0.0
  - Clase B → 172.16.0.0
  - Clase C → 193.144.84.0
- Dirección de broadcast: el nº de red y el resto a 1
- Ejemplos:
  - Clase A → 10.255.255.255
  - Clase B → 172.16.255.255
  - Clase C → 193.144.84.255

Direccionamiento IPv4

Direcciones especiales reservadas

- Algunas direcciones reservadas por el IANA:
  - 0.0.0.0 → *Esta red*. Para arrancar sistemas sin disco (protocolo DHCP, también encaminamiento por defecto)
  - 127.0.0.0–127.255.255.255 → la propia estación (dirección de *loopback*), se suele usar la 127.0.0.1
  - 240.0.0.0–255.255.255.254 → reservadas para uso futuro
  - 255.255.255.255 → difusión a toda la red (protocolo DHCP)
- Descritas en el RFC 3330

Direccionamiento IPv4

Subredes

- Problema: el número de estaciones en una red puede ser demasiado grande $\Rightarrow$ dificultades de administración
- Solución: dividir la red en subredes, que se gestionen de forma independiente pero que actúen como una sola de cara al exterior

Máscara de subred

- Utilizamos parte del campo estación para indicar la subred
- Empleamos *máscaras* para delimitar la subred
- Formato de máscara: 32 bits de los que los n más significativos están a 1 y los $32 - n$ restantes a 0
- Ejemplo: máscara de 27 bits, se denota como sufijo $/27$
  $255.255.255.224 \equiv 11111111.11111111.11111111.11100000$

Direccionamiento IPv4

Ejemplo de máscara

- Dirección clase C 193.168.17.0/27 (o máscara 255.255.255.224)
  - Los 24 primeros bits indican la red (192.168.17)
  - Los 3 siguientes la subred
  - Los 5 últimos la posición de la estación en la subred
  - Tenemos $2^3 = 8$ subredes, con $2^5 - 2 = 30$ estaciones por subred
  - En total, podemos direccionar $8 \times 30 = 240$ estaciones (254 en clase C sin máscara)

Direccionamiento IPv4

Ejemplo de máscara

- Dirección 193.168.17.0/27 (clase C)

Nº de subred	Dir. base	Dir. broadcast
0	193.168.17.0	193.168.17.31
1	193.168.17.32	193.168.17.63
2	193.168.17.64	193.168.17.95
3	193.168.17.96	193.168.17.127
4	193.168.17.128	193.168.17.159
5	193.168.17.160	193.168.17.191
6	193.168.17.192	193.168.17.223
7	193.168.17.224	193.168.17.255

Direccionamiento IPv4

Ejemplo de subredes

- Dirección 193.168.17.133/27
  - ¿A qué subred pertenece?

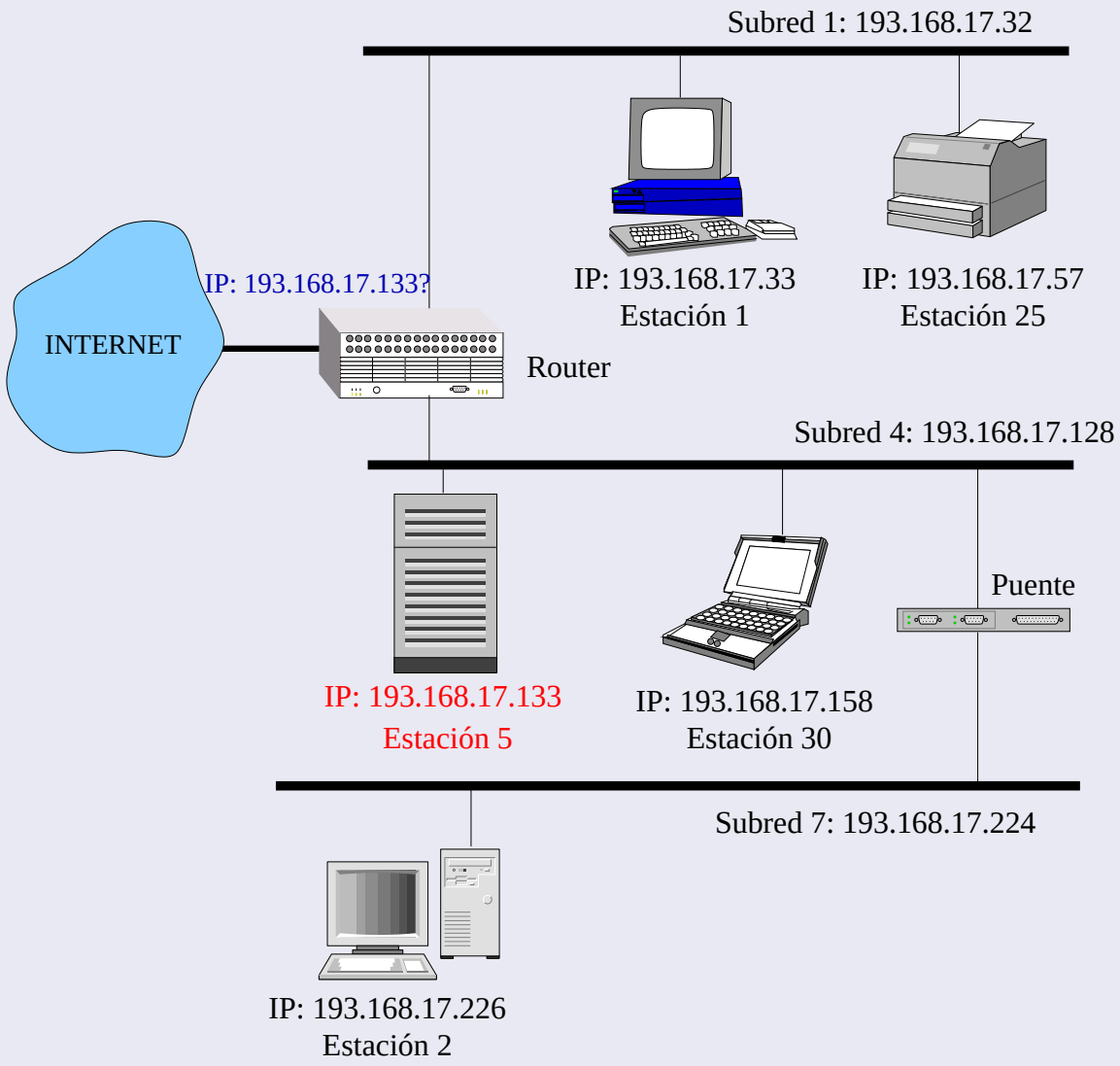
$$
\begin{array}{rcl}
193.168.17.133 & \longrightarrow & 11000001.10101000.00010001.10000101 \\
255.255.255.224 & \longrightarrow & 11111111.11111111.11111111.11100000 \\
\hline
 & & \underbrace{11000001.10101000.00010001}_{\text{Red}}.\underbrace{10}_{\text{Subred 4}}000000
\end{array}
\otimes
$$

  - ¿Qué posición ocupa en la subred?

$$
\begin{array}{rcl}
193.168.17.133 & \longrightarrow & 11000001.10101000.00010001.10000101 \\
255.255.255.224 & \longrightarrow & 00000000.00000000.00000000.00011111 \\
\hline
 & & 00000000.00000000.00000000.00\underbrace{001}_{\text{Estación 5}}01
\end{array}
\otimes
$$

Direccionamiento IPv4

Esquema de red con subredes



Direccionamiento IPv4

Redes sin clase

- En 1993, se suprimen las clases
- Direcciones CIDR (*Classless Inter-Domain Routing*)
  - Sufijo $/s \Rightarrow s$ bits para indicar la red y $32 - s$ para indicar la estación ($2^{(32-s)} - 2$ estaciones)
- Ejemplos: 193.168.17.0/18, 130.15.0.0/8
- También se conocen como *superredes*
- Ejemplo: 193.168.173.253/18
  - Nº de red: 11000001.10101000.10000000.00000000 = 193.168.128.0
  - Broadcast: 11000001.10101000.10111111.11111111 = 193.168.191.255
  - Nº estación: 11000001.10101000.10101101.11111101 = estación nº 11773
  - Nº total de estaciones: $2^{14} - 2 = 16382$

Direccionamiento IPv4

Subredes en redes sin clase

- Las redes con clase serían: A/8, B/16 y C/24
- Se pueden establecer también subredes

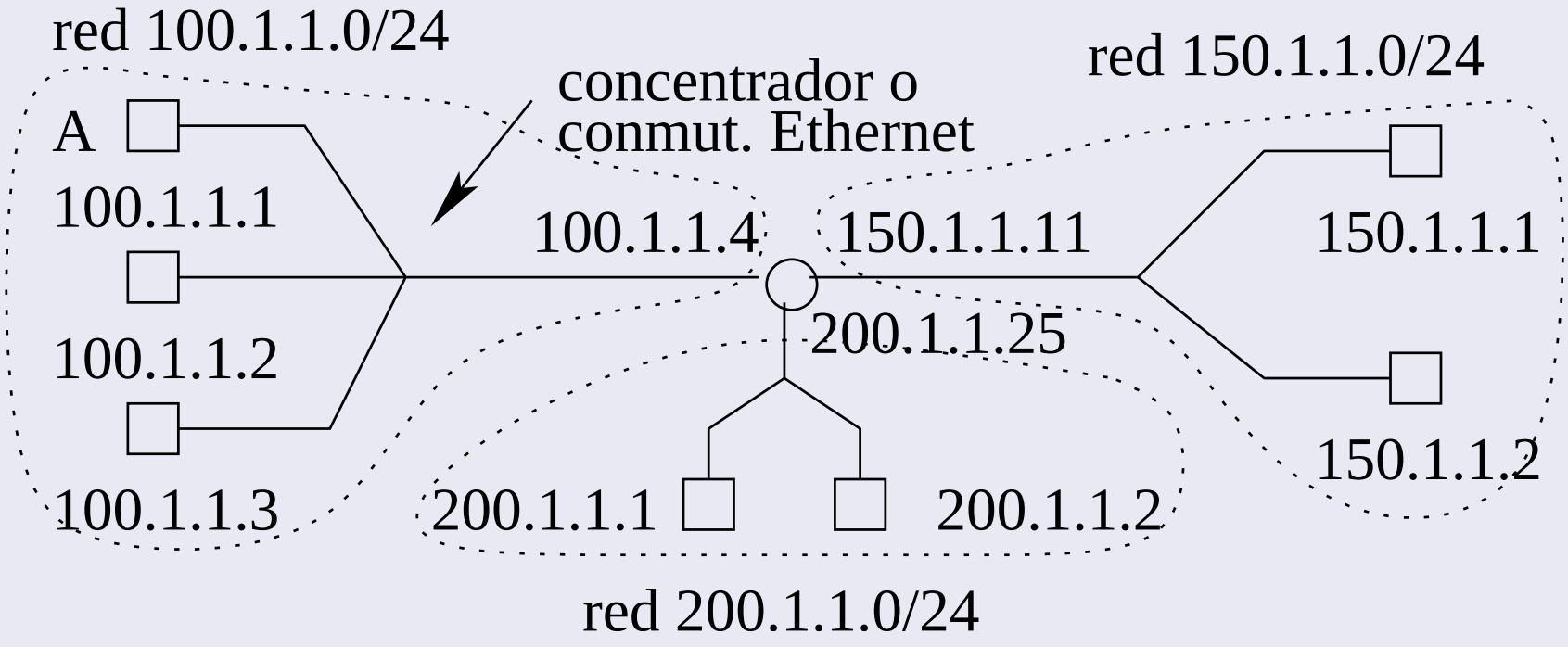
193.144.48.0/20 en dos subredes				
193	144	0011-0-000	0	subred 193.144.48.0/21
193	144	0011-1-000	0	subred 193.144.56.0/21

193.144.48.0/20 en 8 subredes				
193	144	0011-000-0	0	subred 193.144.48.0/23
193	144	0011-001-0	0	subred 193.144.50.0/23
...				
193	144	0011-111-0	0	subred 193.144.62.0/23

Direccionamiento IPv4

Ejemplo de red: 3 redes conectadas a un router

- Cada interfaz tiene una dirección IP



Direccionamiento IPv4

Ejemplo de red: 3 redes conectadas a un router

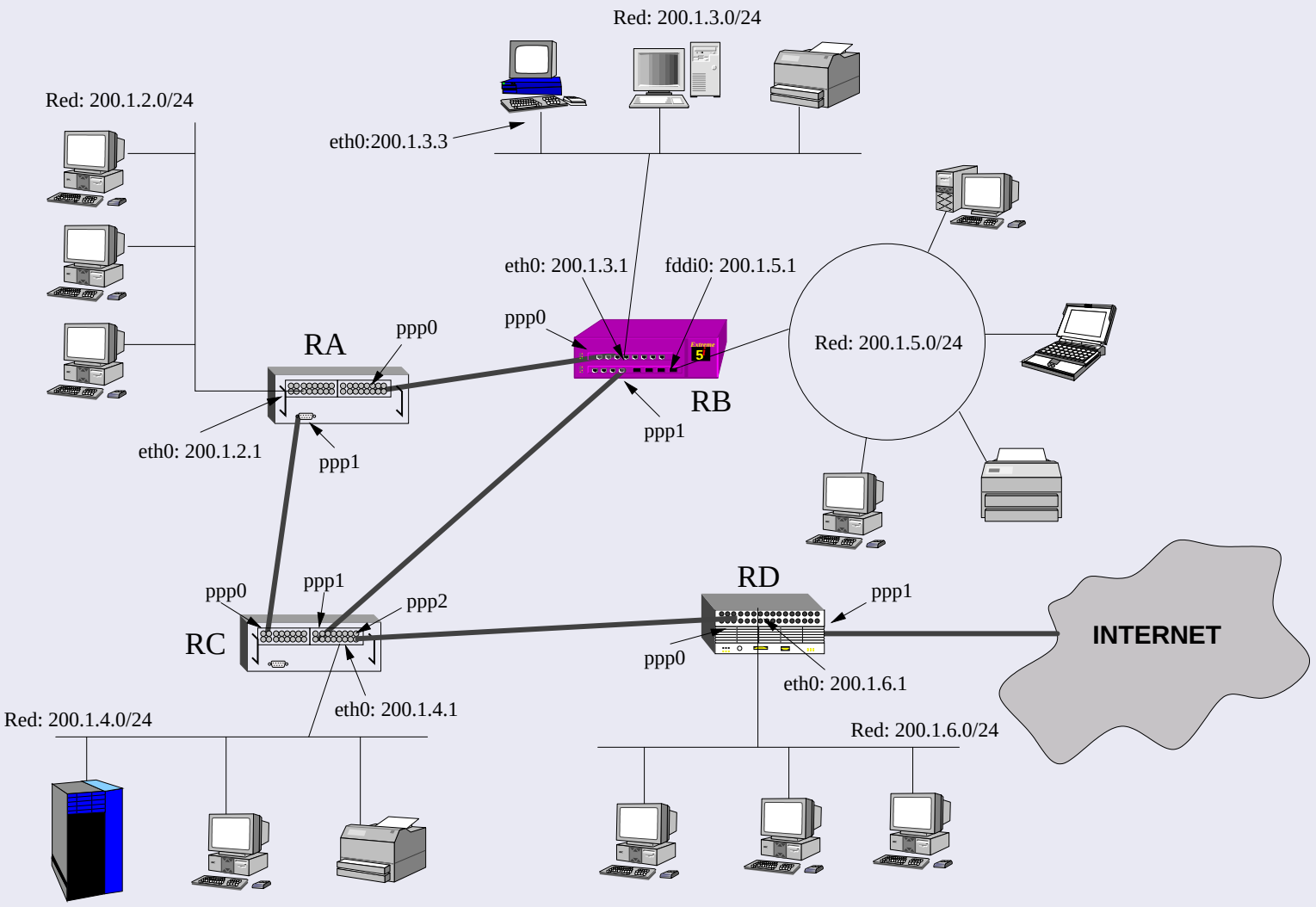
- Cada interfaz tiene una dirección IP

Tabla de rutas del host A			
destino	interfaz	gateway	métrica
100.1.1.0/24	eth0 (=100.1.1.1)	*	0
150.1.1.0/24	eth0 (=100.1.1.1)	100.1.1.4	1
200.1.1.0/24	eth0 (=100.1.1.1)	100.1.1.4	1

Tabla de rutas del router			
destino	interfaz	gateway	métrica
100.1.1.0/24	eth0 (=100.1.1.4)	*	0
150.1.1.0/24	eth1 (=150.1.1.11)	*	0
200.1.1.0/24	eth2 (=200.1.1.25)	*	0

Direccionamiento IPv4

Ejemplo de red



Direccionamiento IPv4

Ejemplo de red: tablas de reenvío de los routers

ROUTER A

Red destino	Gateway	Interfaz
200.1.2.0	200.1.2.1	eth0
200.1.3.0	*	ppp0
200.1.4.0	*	ppp1
200.1.5.0	*	ppp0
200.1.6.0	*	ppp1
0.0.0.0	*	ppp1

ROUTER B

Red destino	Gateway	Interfaz
200.1.2.0	*	ppp0
200.1.3.0	200.1.3.1	eth0
200.1.4.0	*	ppp1
200.1.5.0	200.1.5.1	fddi0
200.1.6.0	*	ppp1
0.0.0.0	*	ppp1

ROUTER C

Red destino	Gateway	Interfaz
200.1.2.0	*	ppp0
200.1.3.0	*	ppp1
200.1.4.0	200.1.4.1	eth0
200.1.5.0	*	ppp1
200.1.6.0	*	ppp2
0.0.0.0	*	ppp2

ROUTER D

Red destino	Gateway	Interfaz
200.1.2.0	*	ppp0
200.1.3.0	*	ppp0
200.1.4.0	*	ppp0
200.1.5.0	*	ppp0
200.1.6.0	200.1.6.1	eth0
0.0.0.0	*	ppp1

Direccionamiento IPv4

Ejemplo de red: tablas de reenvío de un host

Sistema 200.1.3.3

Red destino	Gateway	Interfaz
127.0.0.1	*	lo
200.1.3.0	*	eth0
0.0.0.0	200.1.3.1	eth0

- Consulta de las tablas de rutas en linux: `route [-n]`

Direccionamiento IPv4

Tabla de reenvío de los routers

Tabla de reenvío de un router			
destino	interfaz	gateway	métrica
...	...	...	...
194.24.0.0/21	int_i	*	i saltos
194.24.8.0/22	int_j	*	j saltos
194.24.16.0/20	int_k	*	k saltos
...	...	...	...
0.0.0.0 (por defecto)	int_x	*	x saltos

- Coincidencia del prefijo más largo
- Agregación de rutas

Direccionamiento IPv4

Reenvío de un paquete: coincidencia del prefijo más largo

- ¿Qué ocurre cuando le llega un paquete a 194.24.17.4?
  - Coincidencia de prefijo: se realiza un AND con cada máscara hasta que se produzca la coincidencia de prefijo
    - Con 194.24.0.0/21

      194. 24.00010001.00000100 = 194.24.17.4
      255.255.11111000.00000000 = 255.255.248.0
      194. 24.00010000.00000000 = **194.24.16.0**

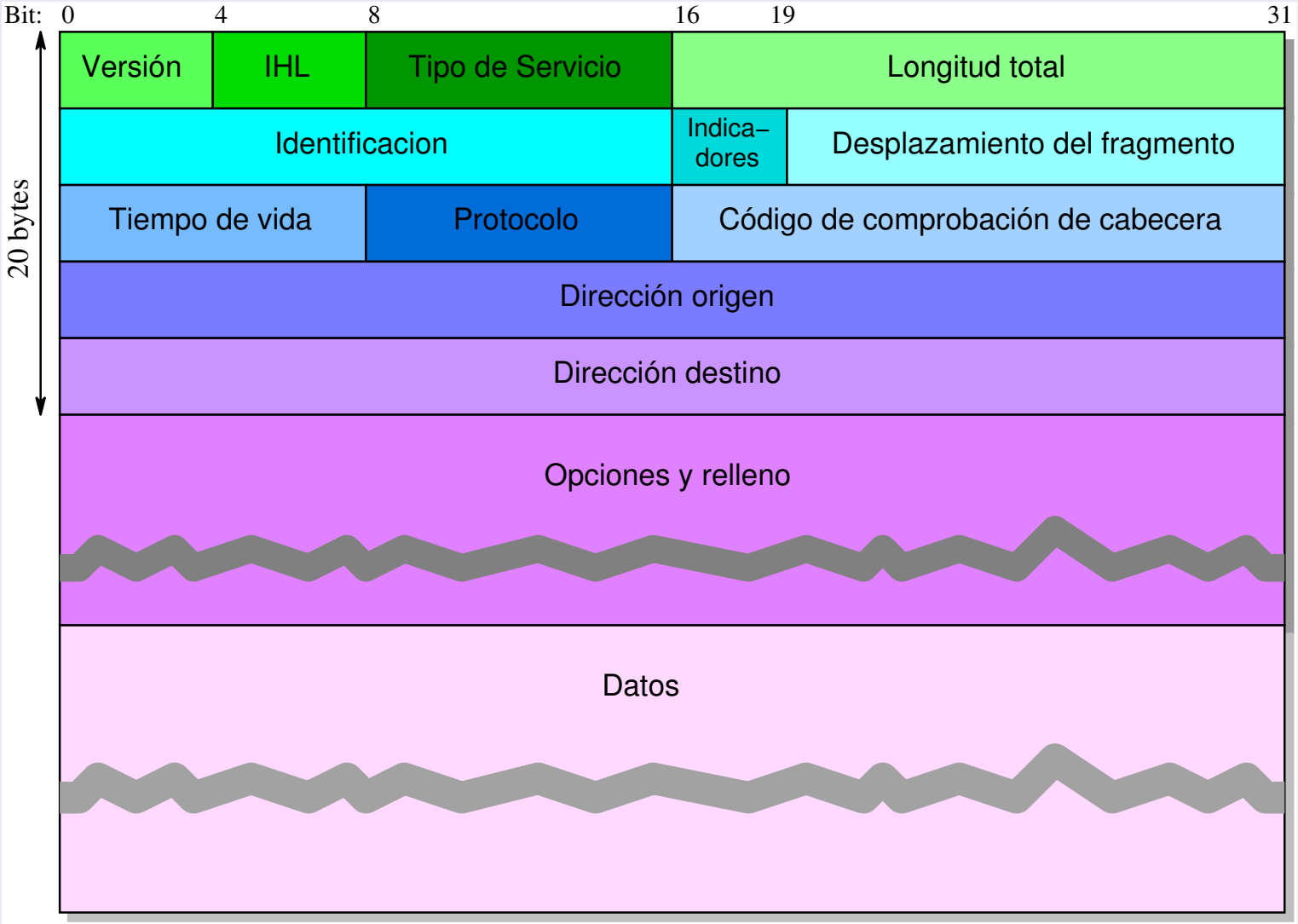
      No coincide con la dirección base de la red (194.24.0.0)
    - Con 194.24.16.0/20

      194. 24.00010001.00000100 = 194.24.17.4
      255.255.11110000.00000000 = 255.255.240.0
      194. 24.00010000.00000000 = **194.24.16.0**

      Sí coincide con la dirección base de la red (194.24.16.0)
      $\Longrightarrow$ se envía por la interfaz correspondiente
    - Si hubiese otra coincidencia con un prefijo más largo (máscara más grande), se reenviaría por la interfaz asociada a esa entrada

Datagrama IP

Formato del datagrama IP

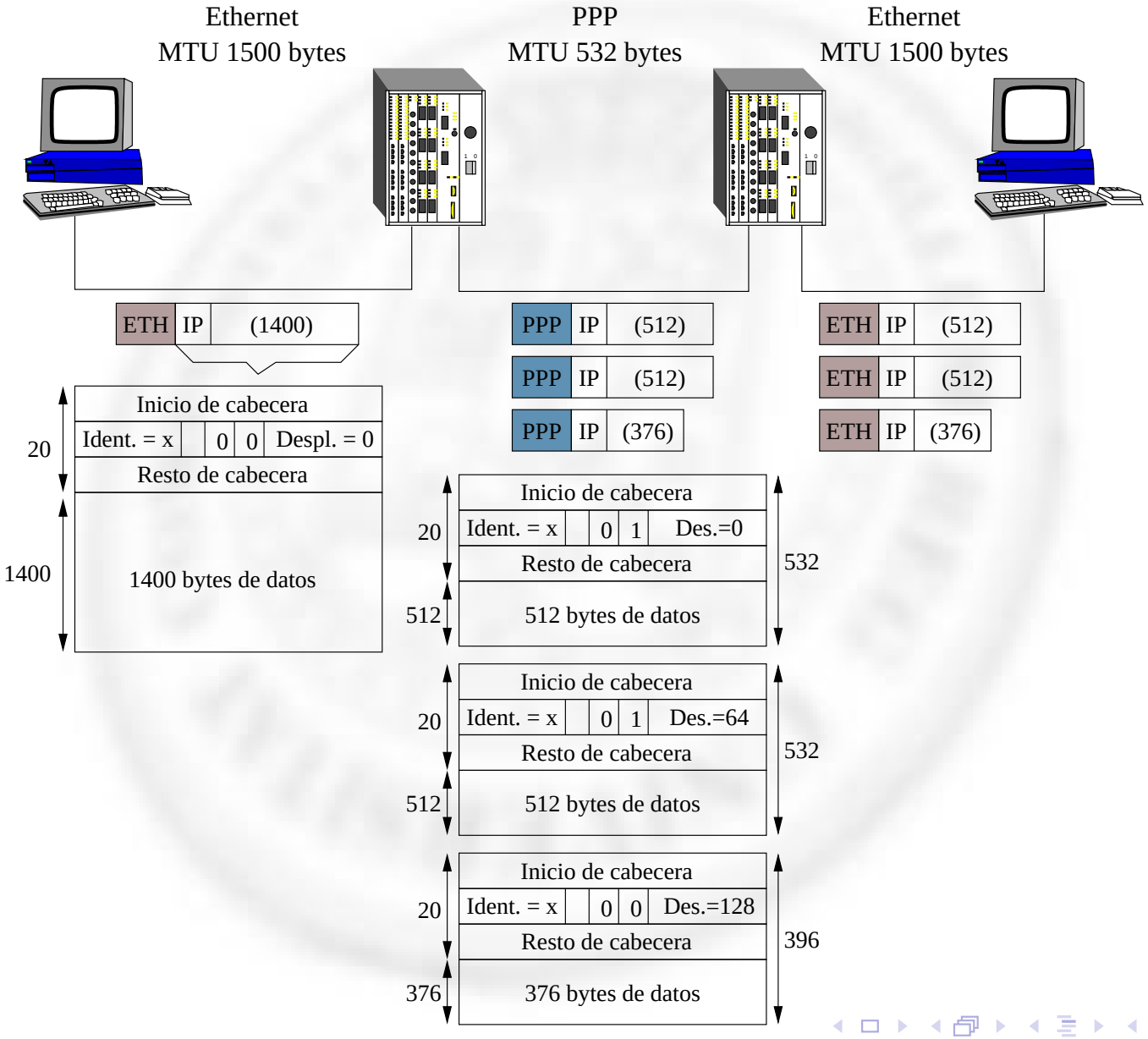


Datagrama IP

Fragmentación

- Las redes pueden especificar un tamaño máximo para los paquetes ⇒
  - Necesidad de fragmentar los datagramas en unidades más pequeñas
  - MTU: *Maximun transmission unit*, tamaño máximo de datagrama IP que puede enviarse en una trama
- Dos niveles de numeración: nº de datagrama y desplazamiento dentro del mismo
  - Uso de un bit MF *Más fragmentos* (1 en todos menos el último)
  - Bit NF indica que no se fragmenta
- El reensamblado se realiza en el sistema destino
- Se intenta evitar la fragmentación, haciendo que TCP y UDP generen segmentos pequeños (536 o 1460 bytes)

Datagrama IP: fragmentación



IPv6

IP versión 6

- La capacidad de direccionamiento del IP actual (v.4) pronto estará en el límite
- Necesidad de simplificar el protocolo, para que los encaminadores sean más eficientes
- Proporcionar mayor seguridad
- Surge IPv6 como respuesta (inicialmente IPng)

Características de IPv6

- Direcciones de 128 bits $\Rightarrow 3{,}4 \times 10^{38}$ direcciones
- No existen clases
- Permite envío multicast
- Servicios en tiempo real
- Servicios de autenticación y seguridad

IPv6

Direcciones IPv6

- Representación con 8 campos de 16 bits en hexadecimal:
  `47CD:1234:4422:AC02:0022:1234:A456:0124`
- Forma compacta si hay cadenas de ceros:
  `47CD:0000:0000:0000:0000:0000:A456:0124`
  $\Leftrightarrow$ `47CD::A456:0124`
- Direcciones IPv4 escritas como IPv6
  `::FFFF:193.144.84.77`
  - Facilitan la transición IPv4 $\rightarrow$ IPv6
- Uso de máscaras para la parte de red y de host

dirección:	`3ffe:ffff:100:1:2:3:4:5/48`
máscara:	`ffff:ffff:ffff:0000:0000:0000:0000:0000`
red:	`3ffe:ffff:0100:0000:0000:0000:0000:0000`

- Direcciones *unicast*, *multicast* (comienzan por `ff`) y *anycast* (a cualquier host de una red)

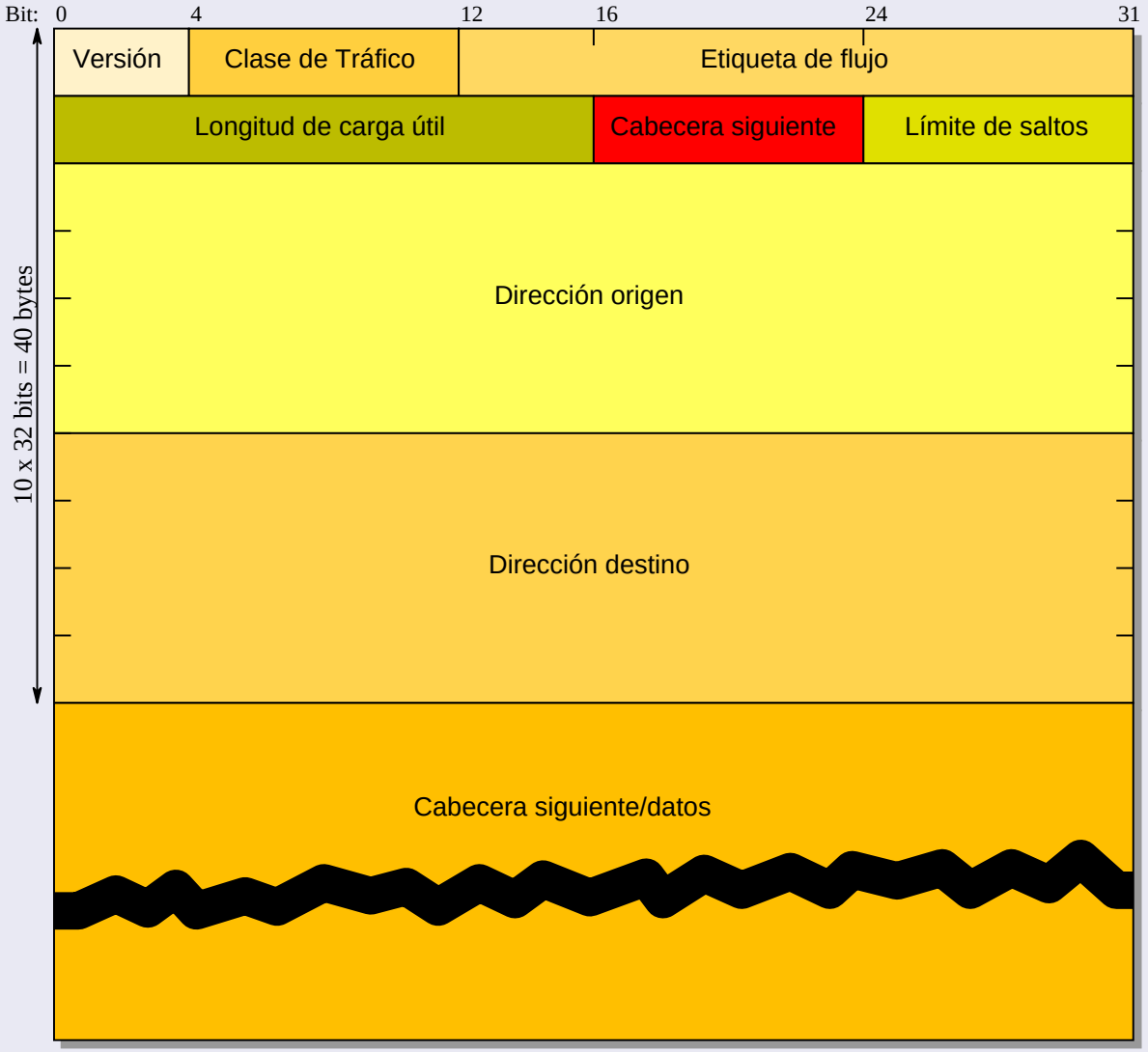
IPv6

Cabecera IPv6

- 40 bytes (desaparece la longitud de cabecera), en 8 campos (menos que IPv4)
- Campos eliminados:
  - Opciones: se indican en cabeceras adicionales
  - Fragmentación: no se fragmentan. Si un datagrama es muy grande, se devuelve un mensaje ICMP
  - Numeración de los paquetes: no se numeran
  - Suma de comprobación: se ahorra procesamiento en los routers
    - Queda en la capa de transporte y posiblemente en la de enlace
- Clase de tráfico y Etiqueta de flujo (8 y 20 bits): relacionados con la QoS

IPv6

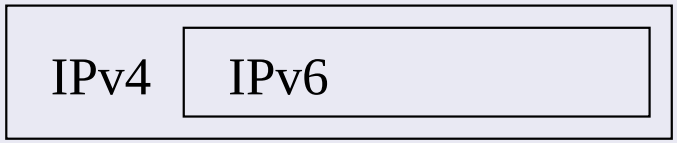
Formato del datagrama IPv6



IPv6

Transición de IPv4 a IPv6

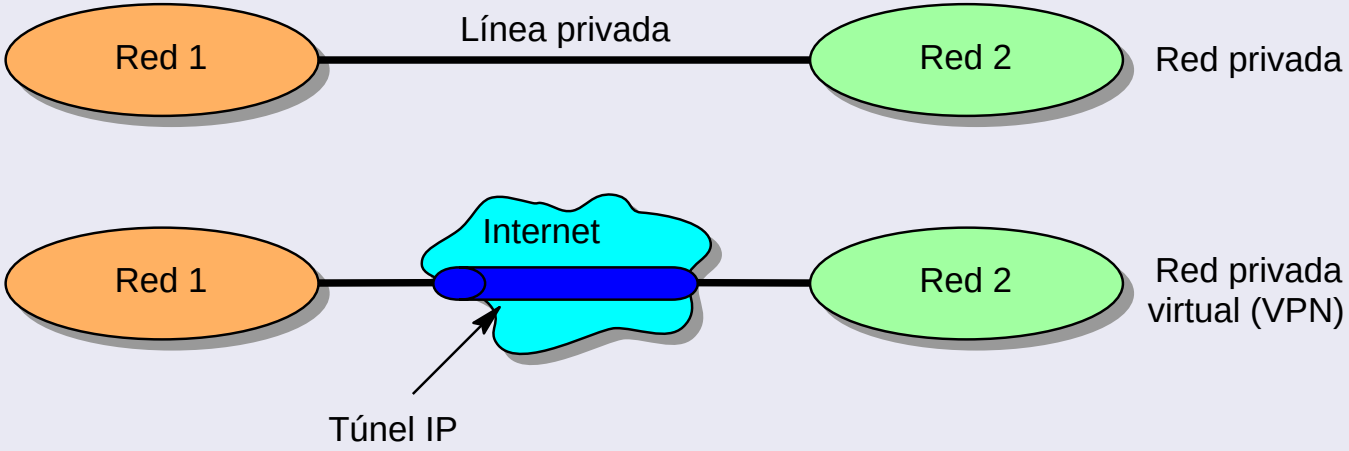
- Debe ser gradual y coexistirán las dos
- Uso de túneles (*tunneling*)
  - Cuando los paquetes IPv6 deban pasar por routers que no soportan IPv6
  - Encapsular los paquetes IPv6 dentro de paquetes IPv4
    - Añadir cabeceras IPv4 con las direcciones de los routers que no soportan IPv6
    - Eliminar dichas cabeceras al llegar a zonas con routers IPv6



VPN: *Virtual Private Network*

Redes privadas virtuales

- Red de una organización que utiliza la red pública para comunicarse de forma segura
- Usan sistemas de encriptación y autenticación
  - **IPsec**, protocolo de seguridad de IP para soportar comunicaciones seguras e implementar VPNs
  - También se pueden implementar mediante túneles IP



VPN: *Virtual Private Network*

Túneles IP

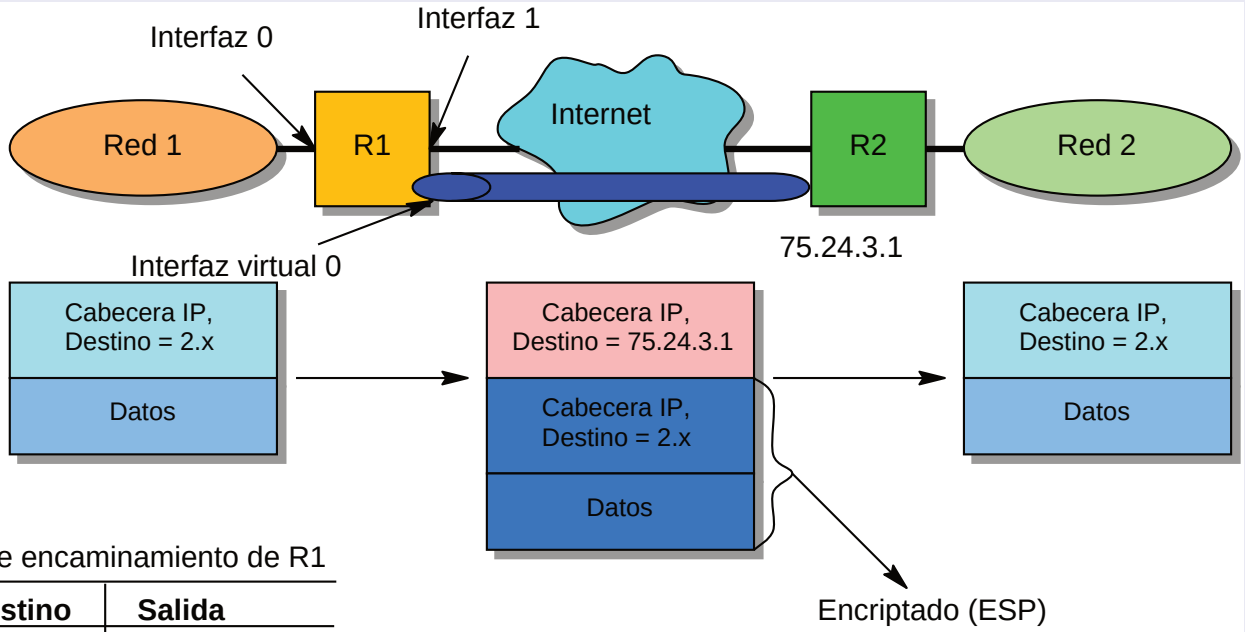


Tabla de encaminamiento de R1

Red destino	Salida
1	Interfaz 0
2	Interfaz virtual 0
Por defecto	Interfaz 1

- R1 cifra el datagrama con la clave pública de R2 y le añada una cabecera con destino R2
- R2 lo descifra con su clave privada y lo envía al destino

Índice

1. Introducción
2. Redes de conmutación de paquetes
3. Algoritmos de encaminamiento
4. Encaminamiento en Internet
5. Protocolo de Internet
6. ICMP: Protocolo de mensajes de control de Internet
7. DHCP: Protocolo de configuración dinámica de hosts
8. NAT: Traducción de direcciones de red

ICMP: *Internet Control Messages Protocol*

Protocolo de mensajes de control de Internet

- Se usa para que hosts y routers puedan informarse sobre errores o el estado de la red
- Funciona sobre IP, pero no se garantiza su entrega
- Se encapsula en un datagrama IP

Mensajes ICMP

- Tipo y código del mensaje
- Los 8 primeros bytes del datagrama que causó el mensaje

ICMP: *Internet Control Messages Protocol*

Tipos de mensajes ICMP

- *Destino inalcanzable*: lo envía un nodo a la estación origen cuando no puede alcanzar el destino o cuando el datagrama no puede fragmentarse y no puede atravesar una red (tipo ICMP 3)
- *Tiempo excedido*: cuando un nodo destruye un datagrama porque su contador llegó a 0 lo manda a la estación origen (tipo ICMP 11)
- *Ralentizar fuente*: para limitar el número de datagramas que las estaciones introducen en la red y evitar la congestión (tipo ICMP 4)
- *Solicitud de eco* y *Respuesta de eco*: se utilizan para ver si un destino es alcanzable y se encuentra activo (uso en `ping`) (tipos ICMP 8 y 0)

ICMP: *Internet Control Messages Protocol*

Tipos de mensajes ICMP

- *Problema de parámetro*: indica que se ha detectado un valor ilegal en un campo de la cabecera (tipo ICMP 12)
- *Redirigir*: se utiliza cuando un nodo se da cuenta de que hay un mejor camino para enviar el datagrama
- *Marca de tiempo* y *Respuesta a marca de tiempo*: para medir el retardo de la red
- *Petición de máscara de dirección* y *Respuesta de máscara de dirección*: empleadas cuando se usan subredes, permiten a un computador conocer la máscara de subred

Índice

1. Introducción
2. Redes de conmutación de paquetes
3. Algoritmos de encaminamiento
4. Encaminamiento en Internet
5. Protocolo de Internet
6. ICMP: Protocolo de mensajes de control de Internet
7. DHCP: Protocolo de configuración dinámica de hosts
8. NAT: Traducción de direcciones de red

DHCP: *Dynamic Host Configuration Protocol*

Protocolo de configuración dinámica de hosts

- Asignación de direcciones IP a los hosts:
  - Estáticamente: el administrador del equipo
  - Dinámicamente: con el protocolo DHCP
    - Cada vez que se inicia, solicita al servidor una IP temporal
    - Usado por los ISPs, cuando no tienen direcciones para todos sus abonados
    - También usado en las redes inalámbricas
- Permite obtener información para un host como:
  - su dirección IP
  - el *gateway* por defecto
  - servidores DNS disponibles

DHCP: *Dynamic Host Configuration Protocol*

Pasos del protocolo DHCP

- Descubrimiento de un servidor DHCP: mensaje `DHCPDISCOVER`
  - IP destino 255.255.255.255 (broadcast)
  - IP origen 0.0.0.0
- Ofrecimiento del servicio DHCP: respuesta del servidor con una IP, máscara de red, etc. y un tiempo de concesión
- Petición DHCP: si hay varias ofertas, el cliente solicita una
- ACK DHCP: el servidor confirma la solicitud

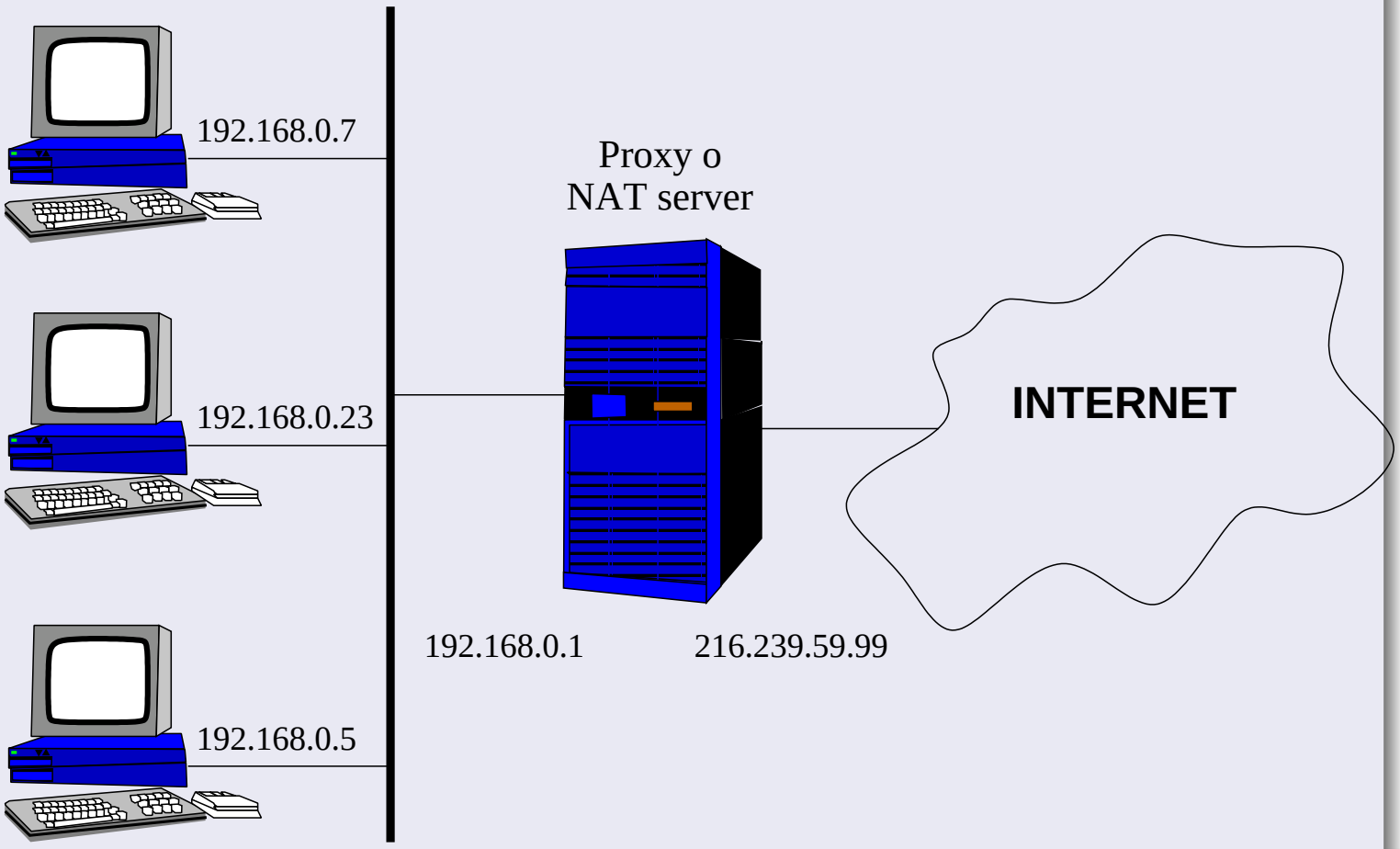
Índice

1. Introducción
2. Redes de conmutación de paquetes
3. Algoritmos de encaminamiento
4. Encaminamiento en Internet
5. Protocolo de Internet
6. ICMP: Protocolo de mensajes de control de Internet
7. DHCP: Protocolo de configuración dinámica de hosts
8. NAT: Traducción de direcciones de red

NAT: *Network Address Translation*

Traducción de direcciones de red

- Permite usar la misma IP válida en varios ordenadores



NAT: *Network Address Translation*

Direcciones sin conexión a Internet

- Direcciones especiales para uso en redes privadas:
  - Comunicaciones internas de una empresa
  - Conexión de varias estaciones usando una única IP
  - Clase A: 10.0.0.0/8 (10.0.0.0–10.255.255.255)
  - Clase B: 172.16.0.0/12 (172.16.0.0–172.31.255.255)
  - Clase C: 192.168.0.0/16 (192.168.0.0–192.168.255.255)
- Los routers de Internet ignoran los paquetes con estas IPs
- Descritas en el RFC 1918

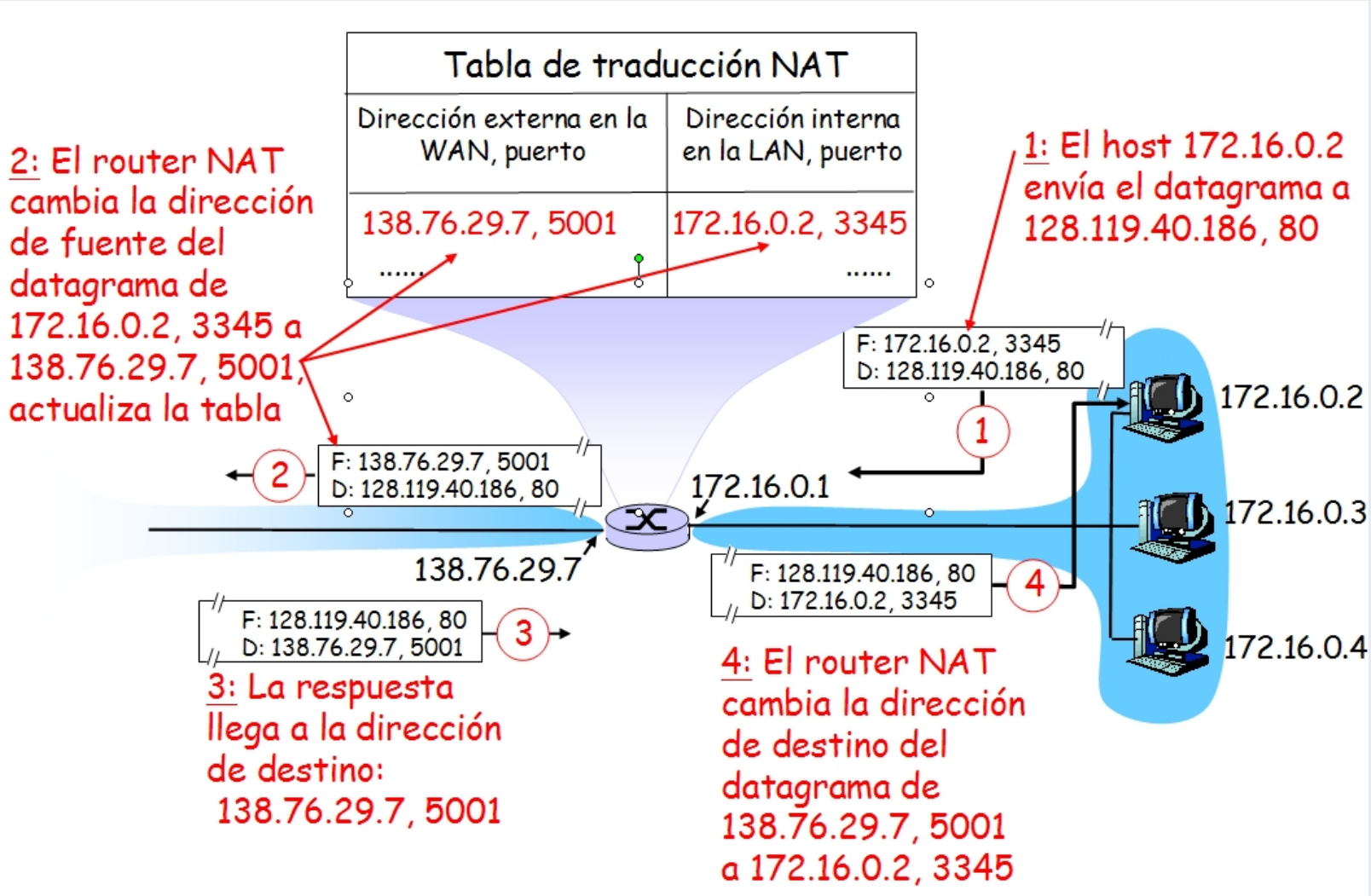
NAT: *Network Address Translation*

Servidor NAT

- Necesita dos interfaces y dos IPs
  - Una para la IP válida para conectar al exterior
  - Otra con una IP privada para conectar a la red interna
- Los ordenadores de la red privada tendrán como gateway la IP privada del servidor NAT
- Encamina de forma transparente los paquetes entre la red interna y la red externa
- Cambia la IP privada y puerto origen de los paquetes internos por la IP del servidor NAT y un puerto libre
- En una tabla almacena IP origen, puerto origen y puerto usado
  - Por el puerto usado sabe a qué máquina enviar la respuesta
- Puede combinarse fácilmente con filtrado de paquetes

NAT: *Network Address Translation*

Funcionamiento



Redes

Tema 5: Capa de enlace

José Carlos Cabaleiro Domínguez

Escola Técnica Superior de Enxeñería

Índice

1. Introducción
2. Modelo IEEE 802
3. Direcciones MAC Ethernet
4. Ethernet
5. Redes inalámbricas
6. Redes ATM
7. Ejemplo completo

Índice

1. Introducción
2. Modelo IEEE 802
3. Direcciones MAC Ethernet
4. Ethernet
5. Redes inalámbricas
6. Redes ATM
7. Ejemplo completo

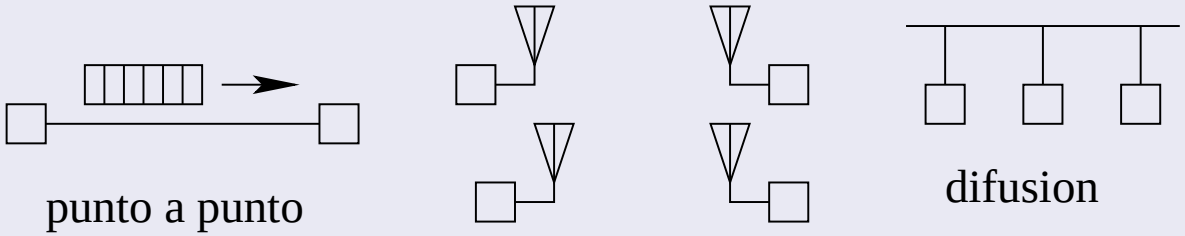
Introducción

Capa de enlace

- Se encarga de transmitir bloques de bits de un lado a otro de un enlace
- Determina el acceso al medio en redes de difusión
- Tramas o marcos: PDU de la capa de enlace

Tipos de enlace

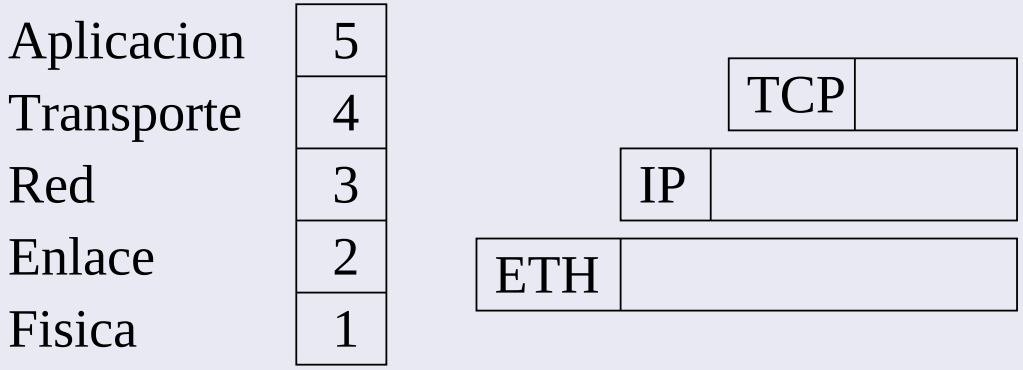
- *Punto a punto*: un emisor y un receptor a ambos extremos del enlace
- *Difusión*: medio de transmisión compartido por varios emisores



Introducción

Capa de enlace

- Implementada en la tarjeta de red (*adaptador*)
- Nodo: host o router al que se conecta el adaptador
- La capa de enlace añade su propia cabecera



Protocolos de la capa de enlace

- Definen el formato de las tramas
- Definen las acciones de los nodos cuando envían o reciben tramas

Introducción

Servicios posibles de un protocolo de capa de enlace

- Entramado o delimitado de tramas: encapsulado de datagramas
- Acceso al enlace: protocolo MAC (control de acceso al medio)
- Entrega fiable: confirmaciones y retransmisiones
- Control de flujo: limitar el envío de tramas
- Detección de errores: más sofisticada que en capas superiores
- Corrección de errores: paridad, checksums y CRC
- Half-duplex y full-duplex: transmisión en uno o ambos sentidos a la vez

Índice

1. Introducción
2. Modelo IEEE 802
3. Direcciones MAC Ethernet
4. Ethernet
5. Redes inalámbricas
6. Redes ATM
7. Ejemplo completo

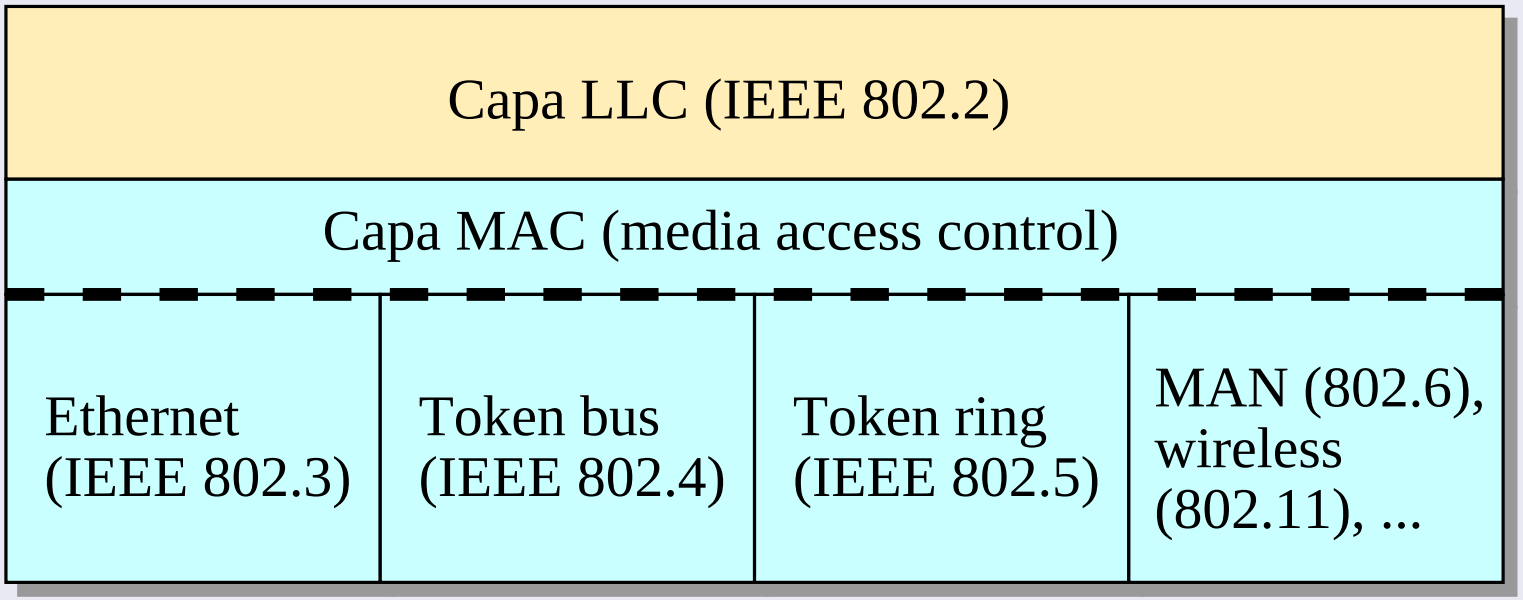
Modelo IEEE 802

Modelo IEEE 802

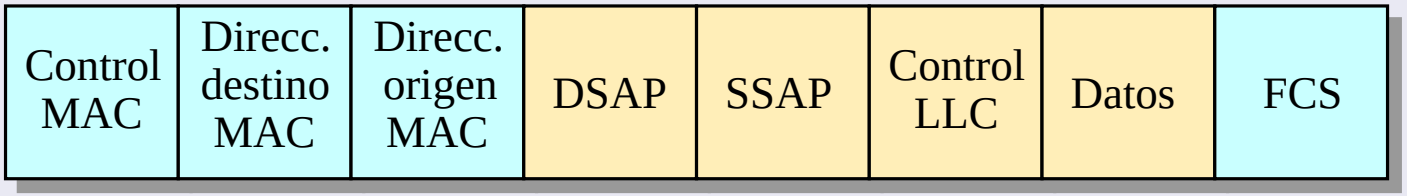
- Principales tipos de LANs definidas por el modelo IEEE 802
  - LANs de difusión: trabajan a nivel de capa de enlace
- Establece un modelo para la capa de enlace en LANs
- Capa de enlace dividida en 2 subcapas:
  - Control de enlace lógico (LLC, *Logical Link Control*)
  - Control de acceso al medio (MAC, *Media Access Control*)
- División en dos subcapas:
  - La lógica necesaria para la gestión de acceso a un medio compartido no está en la capa de enlace de datos tradicional
  - Se pueden ofrecer varias opciones MAC para el mismo LLC

Modelo IEEE 802

Capa de enlace



Capa de enlace en LAN



Formato de trama

Capa LLC

Capa LLC

- Interfaz con las capas superiores
- Control de errores y de flujo

Tipos de servicio

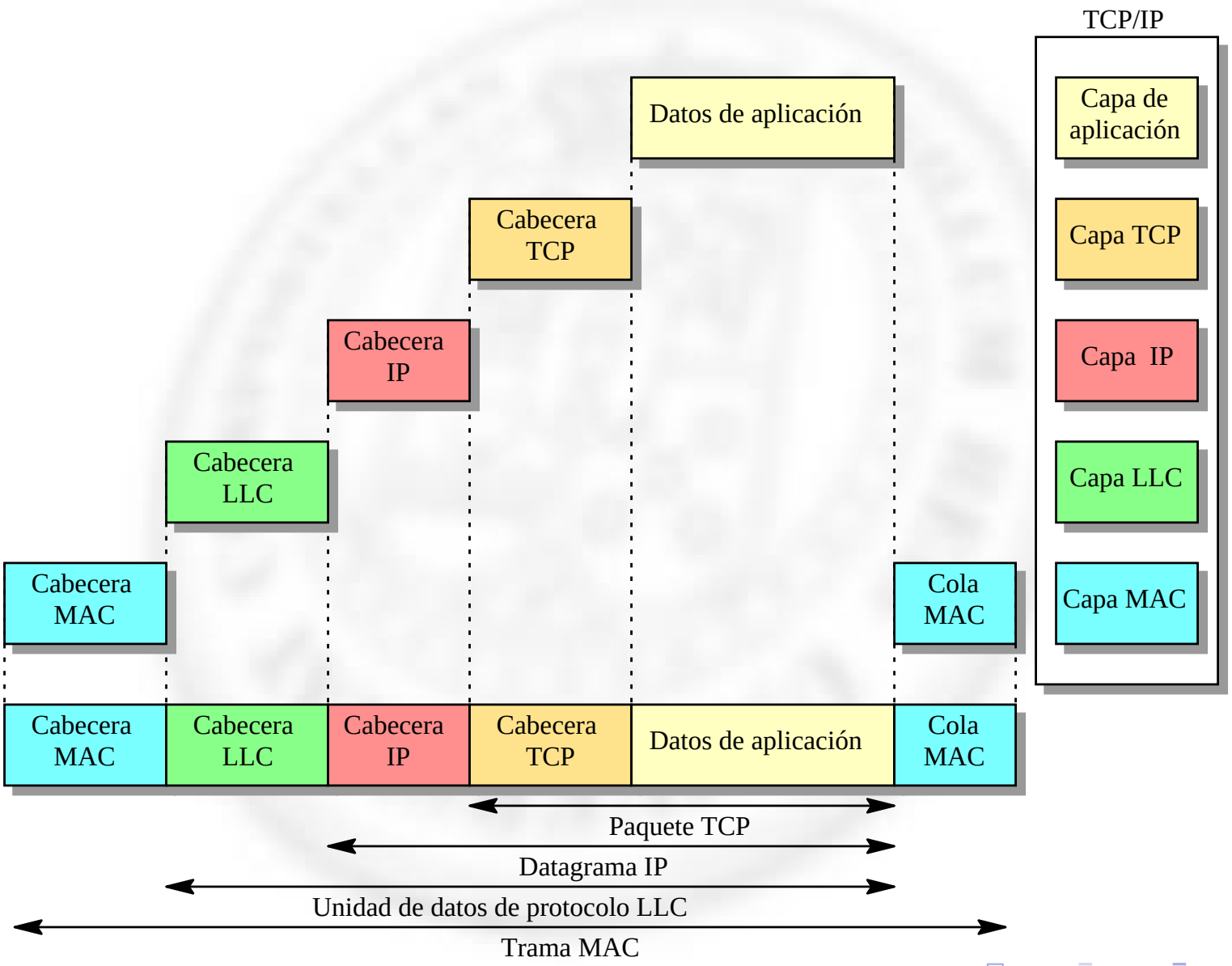
- Sin conexión ni confirmaciones
  - No incluye mecanismos de control de flujo ni errores. No garantiza la recepción de los datos
  - Control de recepción en capas superiores
- Sin conexión con confirmaciones
  - Se confirman las tramas, pero no hay conexión
- Con conexión y confirmaciones
  - Se establece una conexión lógica y hay control de flujo y errores

Capa MAC

Capa MAC

- Ensamblaje de datos en tramas con campos de dirección y de detección de errores
- Desensamblaje de tramas:
  - Reconocimiento de dirección
  - Detección de errores
- Control de acceso al medio de transmisión:
  - No se encuentra en la capa 2 de control de enlace de datos tradicionales
- Para un mismo LLC, están disponibles varios MAC

Protocolos LAN



Índice

1. Introducción
2. Modelo IEEE 802
3. Direcciones MAC Ethernet
4. Ethernet
5. Redes inalámbricas
6. Redes ATM
7. Ejemplo completo

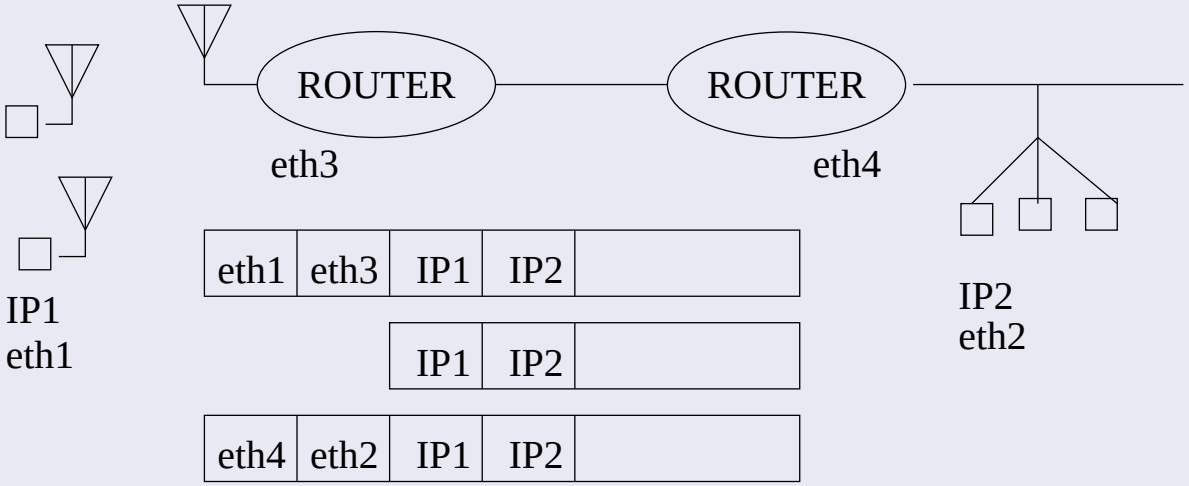
Direcciones MAC

Direcciones de los adaptadores

- La arquitectura TCP/IP considera dos direcciones:
  - Una dirección MAC, que tiene sentido en el enlace o LAN
  - Una dirección IP, que tiene sentido en Internet

Direcciones MAC

- En la LAN, los adaptadores usan las direcciones MAC
- Fuera de la LAN, se eliminan las cabeceras MAC y el paquete viaja usando las direcciones IP



Direcciones MAC

Direcciones Ethernet

- Todos los nodos Ethernet (802.3) tienen una dirección única que los identifica (dir. MAC, o *MAC address*)
- La dirección la proporciona el adaptador Ethernet y suele estar fijada en una memoria ROM
- Las direcciones constan de 6 bytes, expresados en hexadecimal: `00:08:74:4A:BA:4B`
- Para asegurar que no se repitan direcciones cada fabricante tiene un código único para el comienzo de la dirección, ejemplos:

`08:00:20`	$\rightarrow$	Sun
`08:00:5A`	$\rightarrow$	IBM
`00:20:18`	$\rightarrow$	Realtek
`00:80:9F`	$\rightarrow$	Alcatel

Direcciones MAC

Direcciones Ethernet especiales

- Dos direcciones Ethernet especiales:
  - *Broadcast*: todos los bits a 1 (`FF:FF:FF:FF:FF:FF`), trama dirigida a todos los nodos de la red
  - *Multicast*: bit menos significativo del primer byte a 1, trama dirigida a un grupo de nodos de la red
- Un nodo acepta todas las tramas en las que la dirección destino sea:
  - Su propia dirección Ethernet (*unicast*)
  - Todos los bits a 1 (*broadcast*)
  - El LSB del primer byte a 1 (*multicast*)
  - Cualquier valor si está en modo promiscuo (ej. `ifconfig eth0 promisc`)

Direcciones MAC

ARP (*Address Resolution Protocol*)

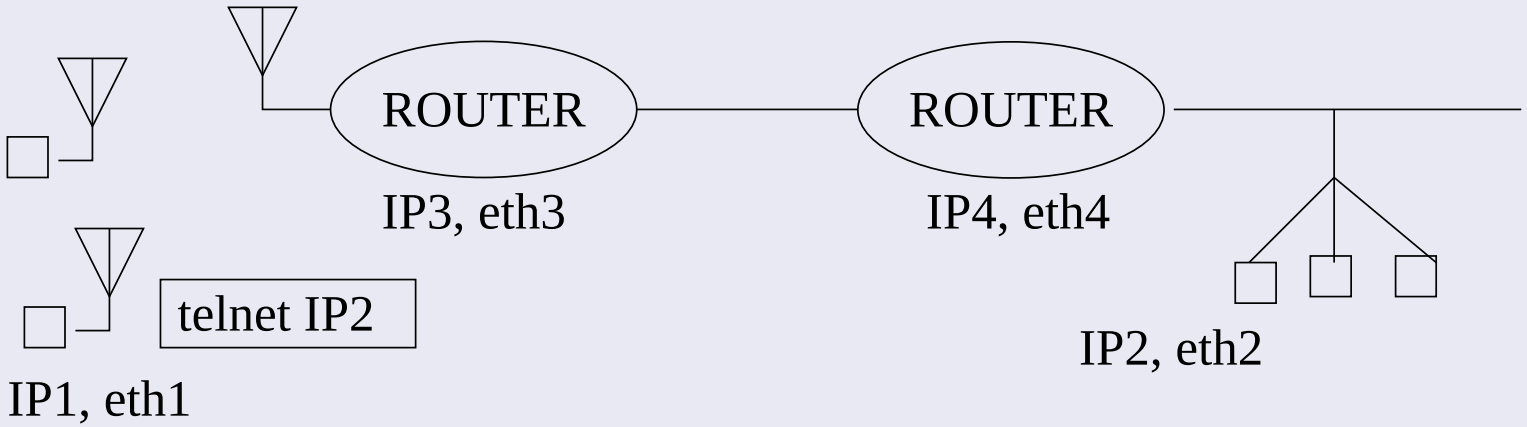
- Las tarjetas de red manejan direcciones MAC
- El software de los nodos trabajan con direcciones IP

$\Longrightarrow$ se necesita obtener direcciones MAC a partir de las IPs

- El Protocolo ARP mantiene una tabla (caché ARP) con correspondencias direcciones IP/direcciones MAC
- Cuando ARP recibe una IP, busca en la tabla
  - Si la encuentra, devuelve la MAC correspondiente
  - Si no está en la tabla
    - ARP emite una trama broadcast indicando esa IP
    - El adaptador al que corresponda esa IP responde con su dirección MAC
    - La respuesta se almacena en la cache del peticionario
    - Se procede al envío de la trama
- Las entradas se eliminan a los 15 minutos

ARP

Ejemplo



1) Se consulta en la tabla de rutas la gateway

2) Se pregunta la Ethernet de IP3 (si no esta en la tabla ARP)
Se anade a la tabla ARP

3) Se hace la transmision

eth1	eth3	IP1	IP2	

tabla rutas del host IP1

destino	gateway
default	IP3

consulta:	ARP	?IP3?	
respuesta:	ARP	IP3	eth3

tabla ARP

IP	Ethernet	Tiempo
IP3	eth3	9:40

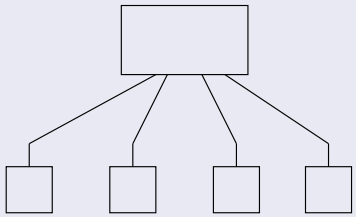
Índice

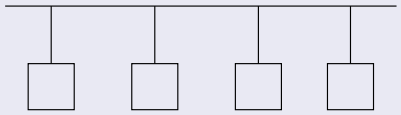
1. Introducción
2. Modelo IEEE 802
3. Direcciones MAC Ethernet
4. Ethernet
5. Redes inalámbricas
6. Redes ATM
7. Ejemplo completo

Ethernet

Características

- La red Ethernet es el tipo de LAN más sencilla y más común
- Servicio no fiable
- Red de difusión
  - Topología bus: cable coaxial puentea todos los adaptadores. Obsoleta
  - Topología estrella: par trenzado conecta cada adaptador con el centro

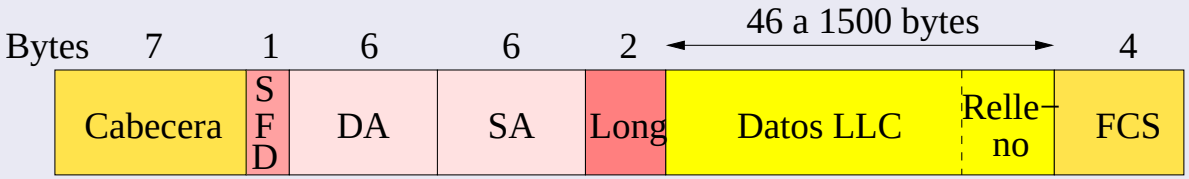




- El protocolo Ethernet funciona sobre cable coaxial, par trenzado y fibra óptica
- Muchas velocidades: 10 Mbps, 100 Mbps, 1 y 10 Gbps

Trama MAC Ethernet

Formato



SFD = Delimitador de comienzo de trama (Start of Frame Delimiter)
DA = Dirección destino (Destination Address)
SA = Dirección origen (Source Address)

- Cabecera: 7 bytes 10101010 cada uno, sincronización
- SFD: byte 10101011, indica el comienzo real de la trama
- Longitud del campo de datos (2 bytes)
  - En Ethernet DIX, campo de *Tipo*, indica el protocolo de red usado (IP o ARP)
- Relleno: para que la trama tenga un tamaño mínimo
  - Tamaño mínimo (sin cabecera ni SFD): 64 bytes = 512 bits
  - Tamaño máximo: 1518 bytes = 12144 bits
- FCS (*Frame Check Sequence*): código CRC de 4 bytes

Ethernet

Control de acceso al medio

- Red de difusión $\Longrightarrow$ protocolo MAC para decidir quién transmite
- Ethernet usa CSMA/CD, acceso múltiple por detección de portadora con detección de colisión
- Para recibir: todos los adaptadores escuchan continuamente el cable
- Para transmitir:
  - El adaptador escucha el medio
    - Si está libre transmite
    - Si está ocupado, espera hasta que quede libre + un pequeño intervalo de seguridad
- Se pueden producir colisiones
- Colisión: en el medio coinciden dos señales de datos

Ethernet

Detección de colisiones

- El nodo emisor escucha el cable mientras transmite
  - Tiempo de vulnerabilidad: un nodo ocupa el medio y su transmisión tarda un tiempo en alcanzar otros nodos
  - Durante ese tiempo, otros nodos ven el medio libre $\Longrightarrow$ pueden transmitir
  - Los datos en el medio se alteran $\Longrightarrow$ error
  - Condición: $t_{\text{trama}} > 2t_{\text{prop}}$. Implica un tamaño de trama mínimo o una longitud de enlace máxima

Respuesta a las colisiones

- Cuando un nodo detecta una colisión
  - Acaba de transmitir la cabecera de la trama
  - Emite una secuencia de 32 bits (*jamming sequence*)
  - Detiene la transmisión
  - Usa el algoritmo de espera exponencial binaria

Ethernet

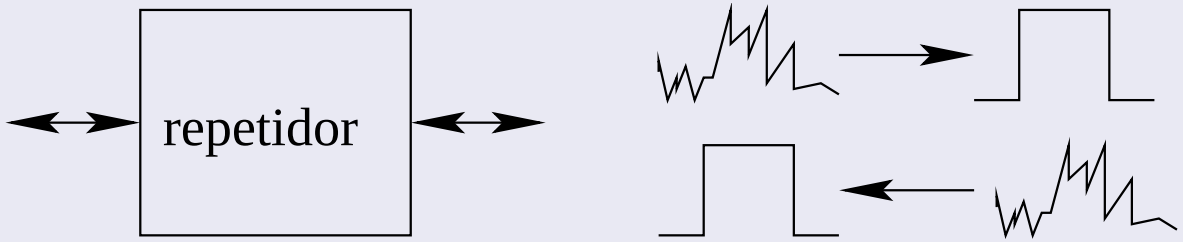
Espera exponencial binaria (*exponential backoff*)

- Divide el tiempo en ranuras discretas de longitud $T = 2t_{\text{prop_max}}$ ($T = 51{,}2\,\mu s$ a 10 Mbps)
- Las estaciones esperan un tiempo 0 o T antes de volver a intentar la transmisión
- Si se detecta una nueva colisión, selecciona aleatoriamente entre 0, T, $2T$ o $3T$
- En general, el tiempo de espera se elige aleatoriamente entre 0 y $(2^n - 1)T$, $n = 1, 2, \ldots, 10$
- A partir de 10 colisiones, el tiempo se escoge entre 0 y $1023T$
- Después de 16 colisiones seguidas, el controlador desiste e informa del fallo
- Capas superiores se encargan de recuperar el fallo

Tecnologías Ethernet

Repetidores

- Dispositivo de capa 1 (física) que trabaja sobre bits individuales
- Tiene dos o más interfaces
- Copia bits que llegan por una interfaz en el resto de las interfaces (excepto por donde llegó)
  - Reconstruye el pulso de tensión

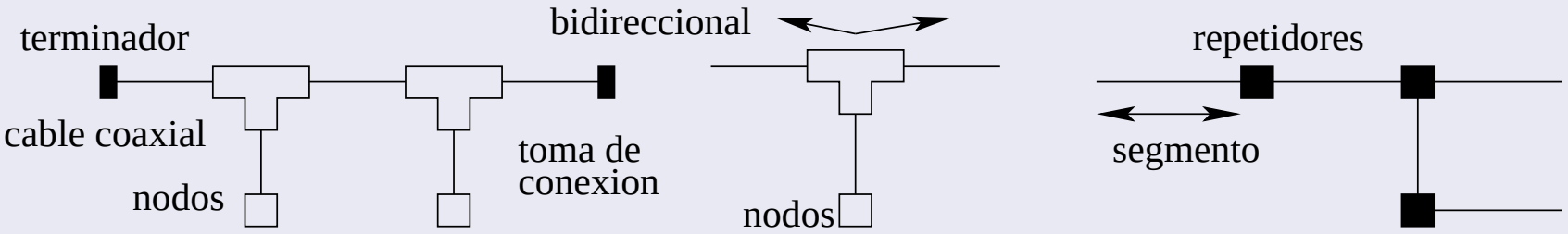


- Para transmisiones a largas distancias

Tecnologías Ethernet

Topología bus (obsoleta)

- Bus de cable coaxial usando conectores T
- Terminadores en los extremos
- No se permite más de 4 repetidores
- Se limita el número de adaptadores en cada segmento

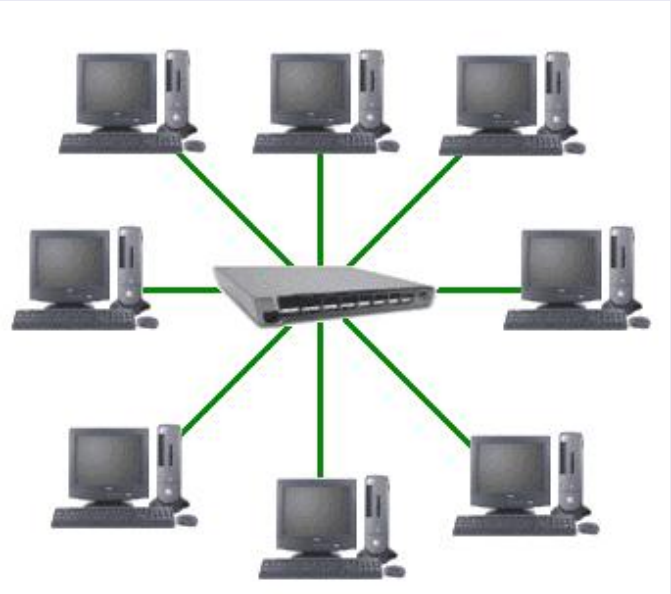


- Nomenclatura: <Mbps><transmisión><centenas m>. Ejemplos:
  - *10base2*: 10 Mbps, banda de base y segmento de 200 m
  - *10base5*: 10 Mbps, banda de base y segmento de 500 m
  - *10broad36*: 10 Mbps, banda ancha con modulación y segmento de 3600 m

Tecnologías Ethernet

Topología estrella

- Estrella con un centro (hub o conmutador) con par trenzado o fibra óptica
- Cada nodo usa un par trenzado o fibra de entrada y otra de salida (en 10 y 100 Mbps)
- Distancia limitada, 100 m en el caso de par trenzado
- A partir de 1000 Mbps se usan 4 pares trenzados
- Funcionamiento del protocolo equivalente al de un bus
- Nomenclatura: T par trenzado y F, S, L y E fibra óptica



Tecnologías Ethernet

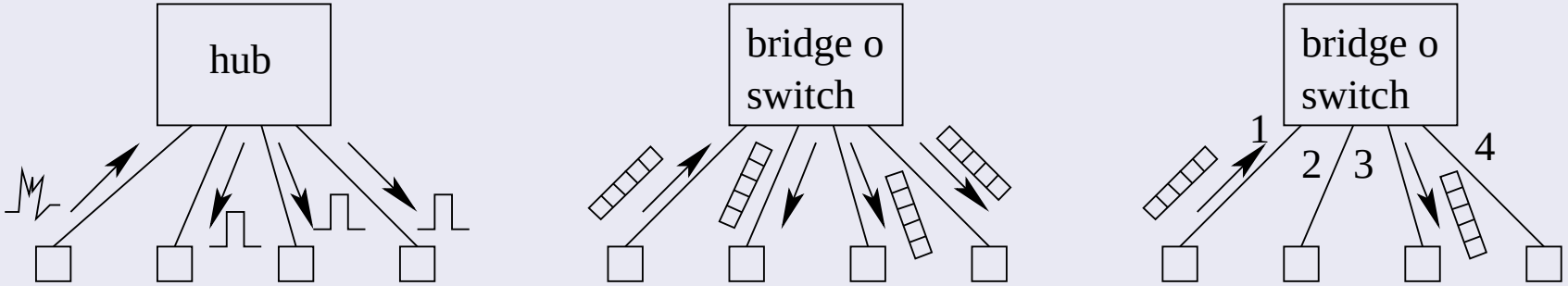
Topología estrella: ejemplos

- *10base-T*: 10 Mbps, banda de base y longitud de 100 m
- *100base-TX* (fast Ethernet de par trenzado): 100 Mbps, banda de base y longitud de 100 m
- *100base-FX* (fast Ethernet de fibra óptica): 100 Mbps, banda de base y longitud de 400 m
- *1000base-T* (Gigabit Ethernet de par trenzado)
- *1000base-SX* y *1000base-LX* (Gigabit Ethernet de fibra óptica)
- *10Gbase-S*, *10Gbase-L* y *10Gbase-E* (10 Gigabit Ethernet de fibra óptica)

Ethernet

Hubs (concentradores)

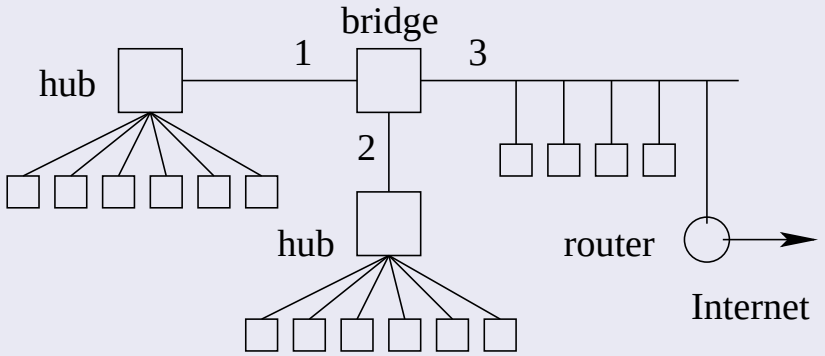
- Dispositivos de capa 1 (capa física) que trabajan a nivel de bits individuales
- Hoy en día están obsoletos
- Difunden los bits: regenera el bit y lo envía por todas las interfaces excepto por la que llegó
- Si llegan a la vez por distintos interfaces, el hub informa a los adaptadores de que hubo colisión

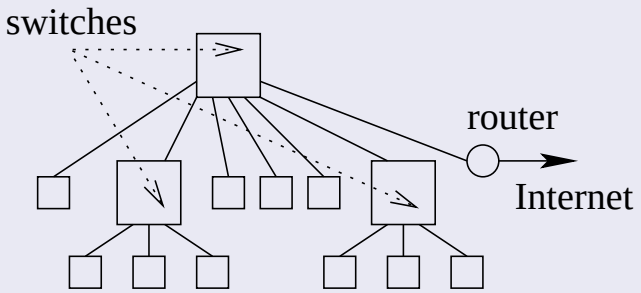


Ethernet

Bridges (puentes) y switches (conmutadores)

- Dispositivos de capa 2 que trabajan a nivel de tramas Ethernet
  - Procesan los distintos campos de las tramas Ethernet, extraen la dirección destino, determinan si tienen errores usando el CRC, etc.
- Disponen de colas en las interfaces de salida
- Los puentes tienen pocas interfaces y los conmutadores decenas
- Los switches han dejado obsoletos a los puentes





Conmutadores y puentes Ethernet

Autoaprendizaje

- Aprenden la localización de los adaptadores
- Tabla de reenvío: entradas para algunos adaptadores
  - Dirección Ethernet, Interfaz, Instante de creación

1. Al principio, la tabla está vacía. Utilizan difusión
2. Aprendizaje hacia atrás: examinan las tramas que llegan
   - La interfaz de llegada indica la localización del adaptador
   - La dirección origen indica la identidad del adaptador

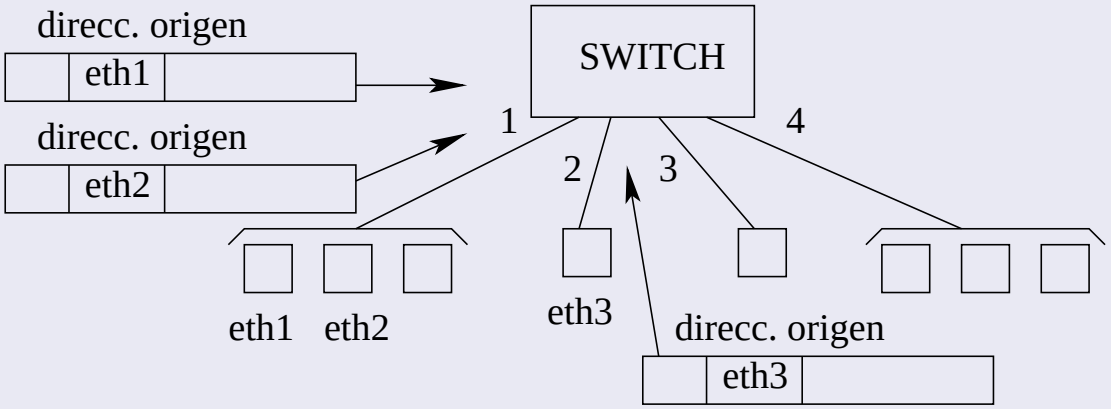


tabla del switch

host	interf.	tiempo
eth1	1	9:20
eth2	1	9:30
eth3	2	9:55

3. Se eliminan las entradas de más de unos minutos

Conmutadores y puentes Ethernet

Aislamiento del tráfico y filtrado

- Adaptador destino en la tabla ⇒ reenvían las tramas sólo por la interfaz indicada
- El resto de los adaptadores no verán esa transmisión ⇒ tráfico aislado ⇒ evita colisiones
- Interfaz origen coincide con el destino ⇒ el conmutador descarta la trama ⇒ filtrado

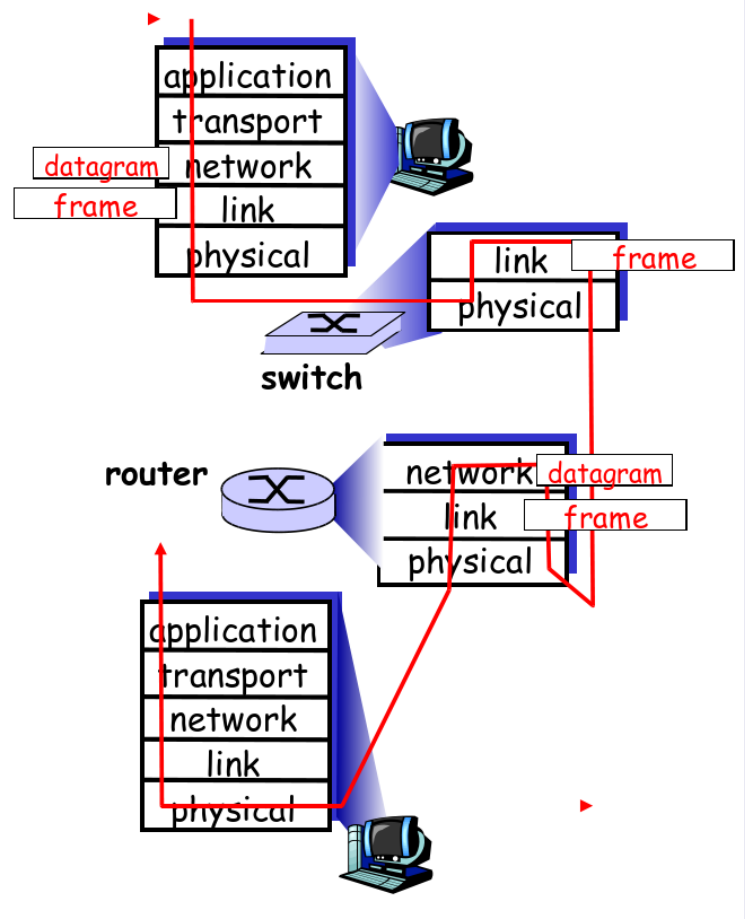
Tasa de transmisión agregada

- Múltiples transmisiones simultáneas, siempre que las interfaces origen y destino sean distintas
- Muchas interfaces ⇒ tasa de transmisión agregada elevada ⇒ diseño de prestaciones elevadas

Conmutadores Ethernet frente a routers

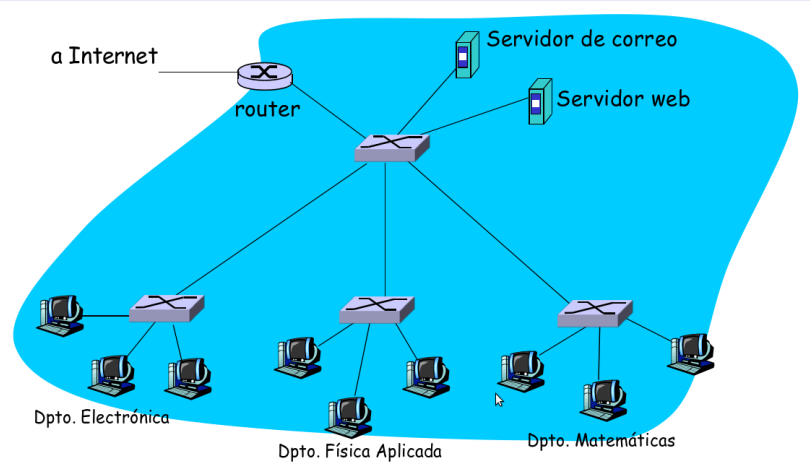
Conmutadores y routers

- Ambos son dispositivos de almacenamiento y reenvío
- Conmutadores:
  - Trabajan con cabeceras Ethernet (capa de enlace)
  - Mantienen las tablas de conmutación, implementan filtrado y algoritmos de aprendizaje
- Routers:
  - Trabajan con cabeceras IP (capa de red)
  - Mantienen las tablas de rutas, implementan algoritmos de encaminamiento



Conmutadores Ethernet

Ejemplo de red institucional



- Inconvenientes:
  - Si un usuario se cambia físicamente de dpto. y desea seguir conectado al anterior
  - Dominio de broadcast único
    - Tramas de mensajes ARP o DHCP
  - Uso ineficiente de los conmutadores: cada uno solo usa unos pocos puertos

Redes de área local virtuales, VLANs

- Se pueden abordar esos problemas con un conmutador compatible con VLANs, que soporte el estándar IEEE 802.1Q (se añaden unos campos en la cabecera)

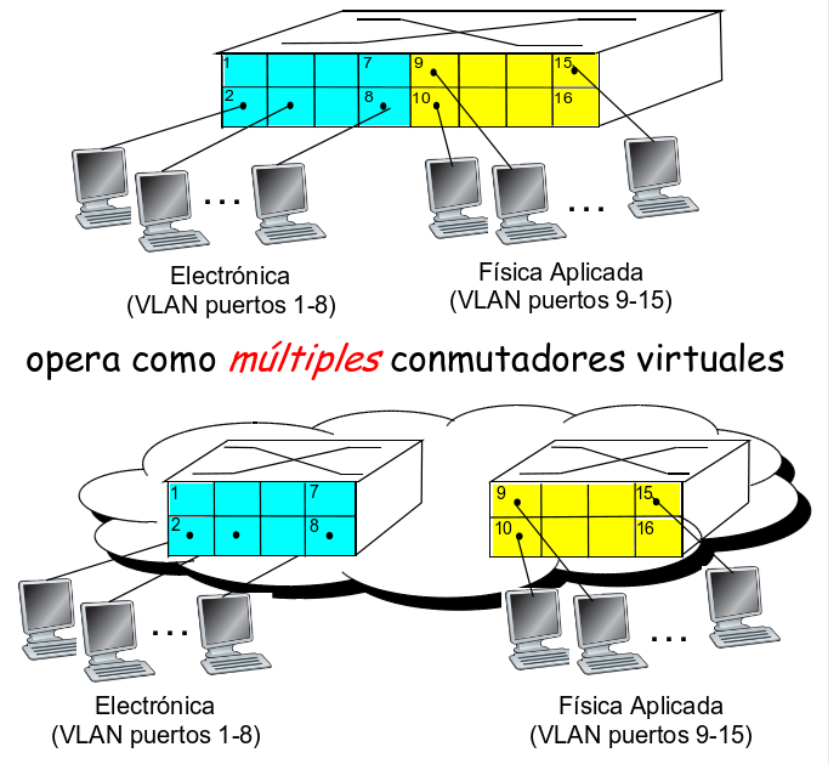
VLANs

VLANs

- Los conmutadores que soportan VLAN permiten definir múltiples LANs *virtuales* sobre una única red física

VLANs basadas en puertos

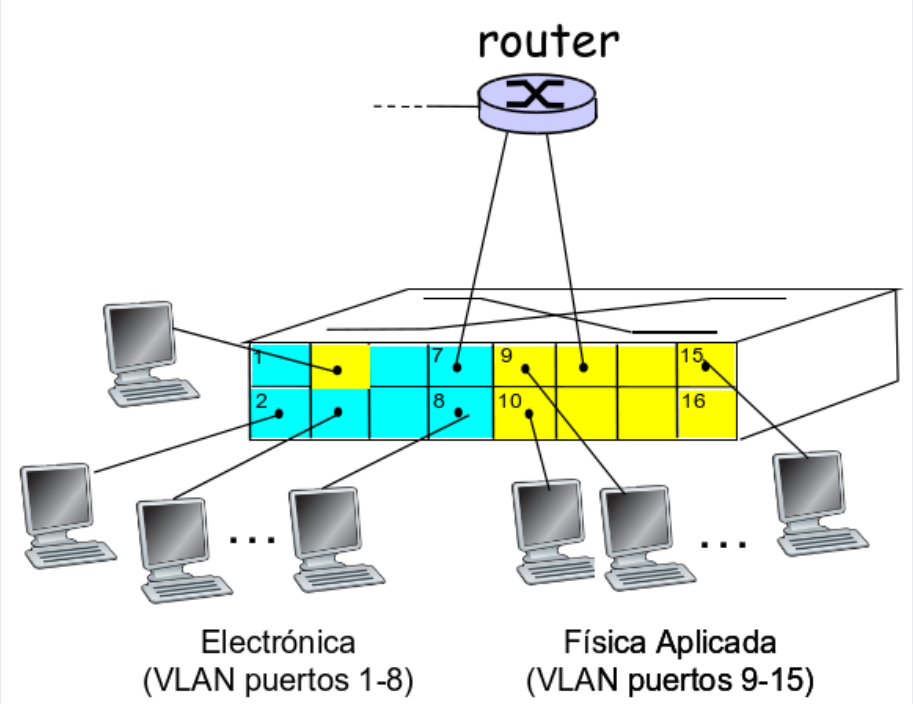
- Se dividen los puertos del conmutador en grupos
- Cada grupo una VLAN
  - Se mantiene una tabla puertos - VLAN
  - Solo se entregan tramas entre puertos de la misma VLAN
- Cualquier cambio ⇒ reconfiguración software



VLANs

VLANs basadas en puertos

- Aislamiento del tráfico
  - Solo se entregan tramas entre puertos de la misma VLAN
  - Se pueden definir VLANs por MAC
- Pertenencia dinámica
  - Asignación dinámica de puertos a VLANs
- Reenvío entre VLANs mediante encaminamiento
  - En la práctica, se combinan routers y switches



Índice

1. Introducción
2. Modelo IEEE 802
3. Direcciones MAC Ethernet
4. Ethernet
5. Redes inalámbricas
6. Redes ATM
7. Ejemplo completo

Redes inalámbricas (WLAN, *Wireless LAN*)

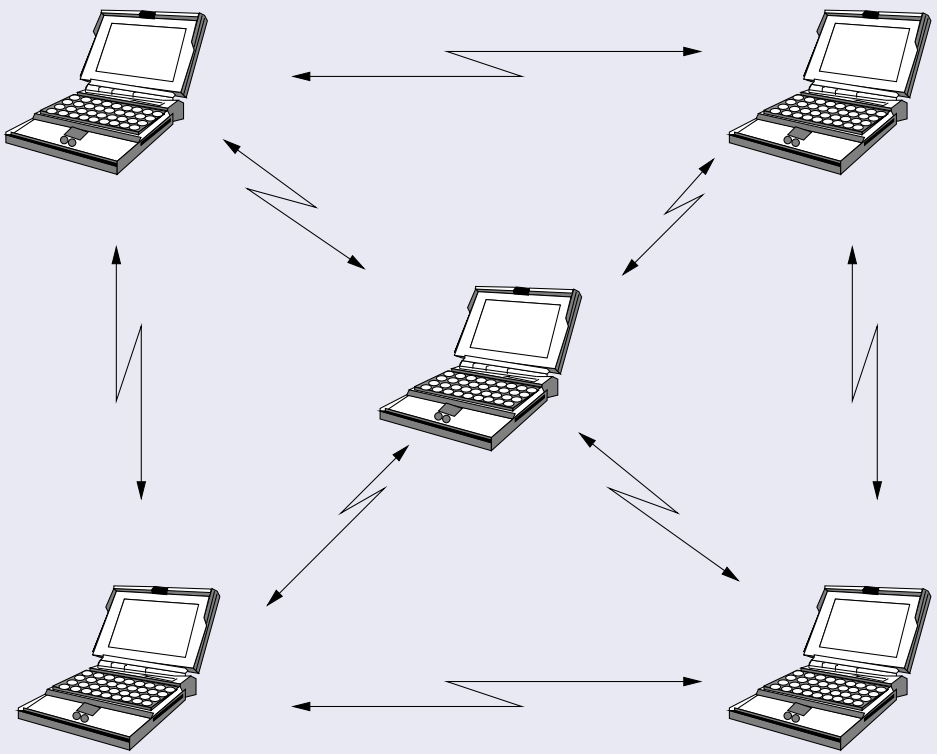
Especificación IEEE 802.11

- 11b (en 2.4 GHz, DSSS, hasta 11 Mbps)
  - Interferencias, velocidad baja
  - Menor absorción, mayor alcance (120–460 m exterior, 30–90 m interior)
- 11a (en 5 GHz, OFDM, hasta 54 Mbps)
  - Menos interferencias, pero mayor absorción y menor alcance (30–300 m exterior, 12–90 m interior)
  - En España, frecuencia reservada para uso militar
- 11g (en 2.4 GHz, OFDM/DSSS, hasta 54 Mbps)
  - Misma velocidad que 11a con mayor alcance
  - Coexistencia con 11b (WiFi)
- 11n (bandas 2,4 Ghz y 5 Ghz, hasta 600 Mbps teóricos)
  - Es la actual, en casi todos los productos
- 11ac nuevo estándar (hasta 1 Gbps teórico)

Configuraciones WLAN

Redes simples (*ad-hoc*)

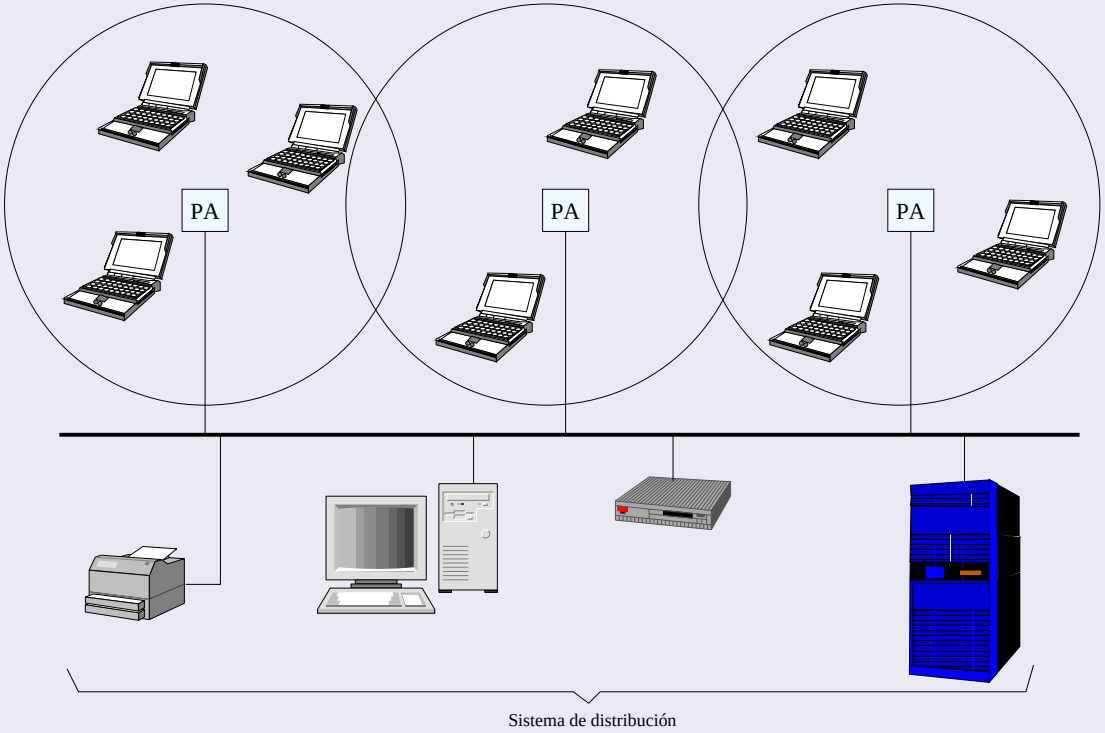
- Conexiones de igual a igual
- Permiten comunicar 2 estaciones siempre que están en su radio de alcance



Configuraciones WLAN

Redes distribuidas (*managed*), con estación base central

- LAN troncal cableada (*distribution system*) que conecta los servidores y los puntos de acceso (AP, *access point*)
- Cada AP da servicio a un número de estaciones móviles, distribuyendo el espacio en celdas



Redes inalámbricas

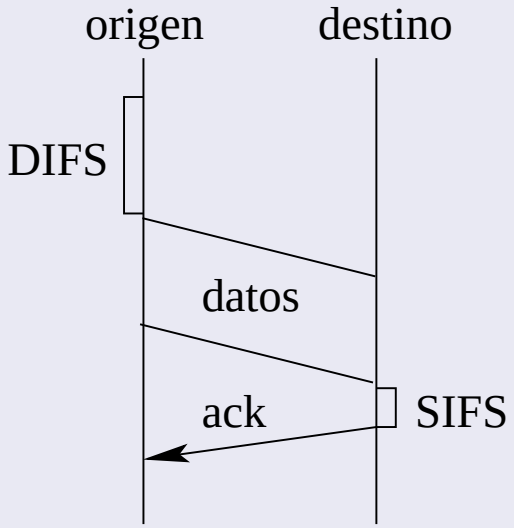
Protocolo de acceso al medio

- Protocolo MACA (*Multiple Access with Collision Avoidance*). También denominado CSMA/CA
- Un host que quiera transmitir sondea el medio
  - Si está libre, espera un intervalo de seguridad grande (DIFS, *Distributed Inter Frame Space*)
  - Si continúa libre, transmite
  - Si está ocupado, continúa escuchando hasta que quede libre. Espera un intervalo y, si sigue libre, emite
  - Si sigue ocupado, utiliza un algoritmo de espera exponencial binaria
- No tiene detección de colisiones. Se usan los ACKs
  - Entre la recepción de la trama y el envío del ACK se espera un intervalo corto (SIFS)
  - Todos los hosts deben esperar a la transmisión del ACK

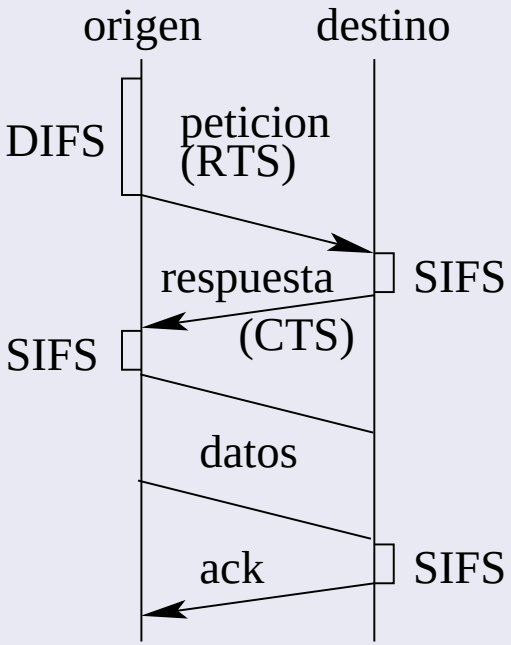
Redes inalámbricas

Protocolo de acceso al medio

- Uso de tramas de control para asegurar la transmisión
  - Se envía primero una trama de petición de envío (RTS)
  - El destino responde con una trama de reserva del canal (CTS)
  - Los demás hosts deben esperar a que la transmisión se complete



envios normales



envios con peticion previa

Índice

1. Introducción
2. Modelo IEEE 802
3. Direcciones MAC Ethernet
4. Ethernet
5. Redes inalámbricas
6. Redes ATM
7. Ejemplo completo

Redes ATM

Modo de transferencia asíncrono

- Tipo de red con la que trabajaban (trabajan) las compañías telefónicas
- Diseñada para operar a alta velocidad
  - Pueden transmitir datos, voz y vídeo
  - Los conmutadores pueden operar a velocidades de terabits por segundo
- El modelo ATM cubre las tres capas inferiores
  - Capa física, de enlace y de red
- Se integra en la arquitectura TCP/IP
- Se usaba en redes telefónicas y en las troncales de Internet

Redes ATM

Tipos de servicio

- CBR (*Constant Bit Rate*)
  - Se reserva y garantiza una cierta tasa de transmisión
  - Los retardos y las pérdidas bajo ciertos límites garantizados
  - Adecuado para transmitir audio y vídeo
- ABR (*Available Bit Rate*)
  - La tasa de transmisión varía en función de los recursos disponibles, aunque se garantiza un mínimo
  - No se garantiza un mínimo en las pérdidas o el retardo
- UBR (*Unespecified Bit Rate*)
  - Solo se transmiten paquetes cuando el resto de los servicios de la red dejan recursos
- VBR (*Variable Bit Rate*) para aplicaciones en tiempo real (VBR-rt) o no en tiempo real (VBR-nrt)

Redes ATM

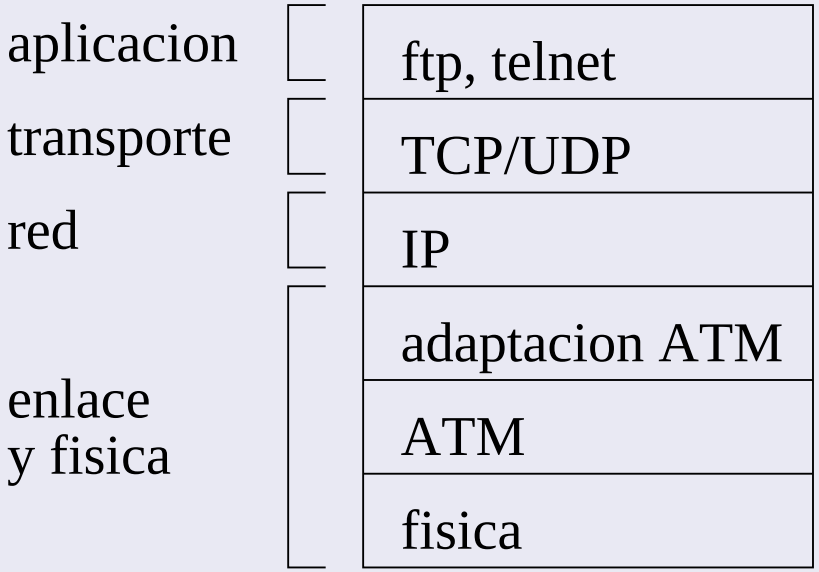
Características

- Paquetes muy pequeños y sencillos (celdas), para garantizar su conmutación a altas velocidades
  - 53 bytes: 5 bytes de cabecera y 48 de datos
- Red de circuitos virtuales (canales virtuales) orientada a conexión
  - Antes de la transmisión, hay una solicitud de conexión
  - Se planifica la ruta
  - Las celdas llevan el número de canal virtual
    - El conmutador ATM consulta en la tabla de canales virtuales y selecciona la línea de salida
    - Todas las celdas siguen el mismo camino $\Rightarrow$ llegan en orden
  - Al finalizar, hay una fase de desconexión en la que se eliminan los canales virtuales
- No hay ACKs ni retransmisiones, pero las celdas tienen control de errores de la cabecera

Redes ATM

Modelo de capas en ATM

- ATM puede funcionar sobre cualquier capa física
- La capa de adaptación a ATM (AAL) permite que otros protocolos usen la red ATM

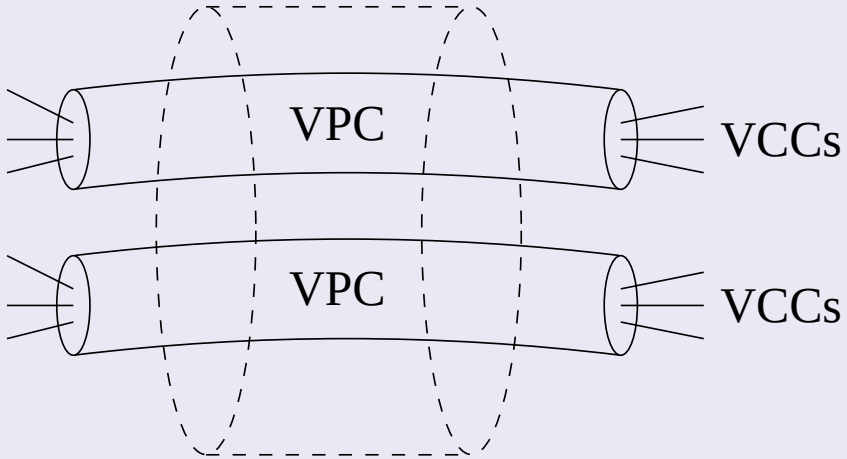


- Diferentes AAL dependiendo del tipo de servicio
  - TCP/IP: a la entrada de la ATM se fragmentan los datagramas para que quepan en las celdas y se reensamblan a la salida
  - Audio y vídeo: se agrupan los datos hasta rellenar una celda

Redes ATM

Identificador de circuito virtual

- Define dos niveles de conexión:
  - VCC (*Virtual Channel Connection*): Canal Virtual
    - Es un circuito virtual
  - VPC (*Virtual Path Connection*): Camino Virtual
    - Conjunto de VCCs con los mismos extremos
    - Facilitan la gestión de los VCCs



Medio de transmisión

Redes ATM

Estructura de las celdas

4	8	16	3	1	8	384 (48 bytes)
GFC	VPI	VCI	Tipo	CLP	HEC (CRC-8)	Carga útil

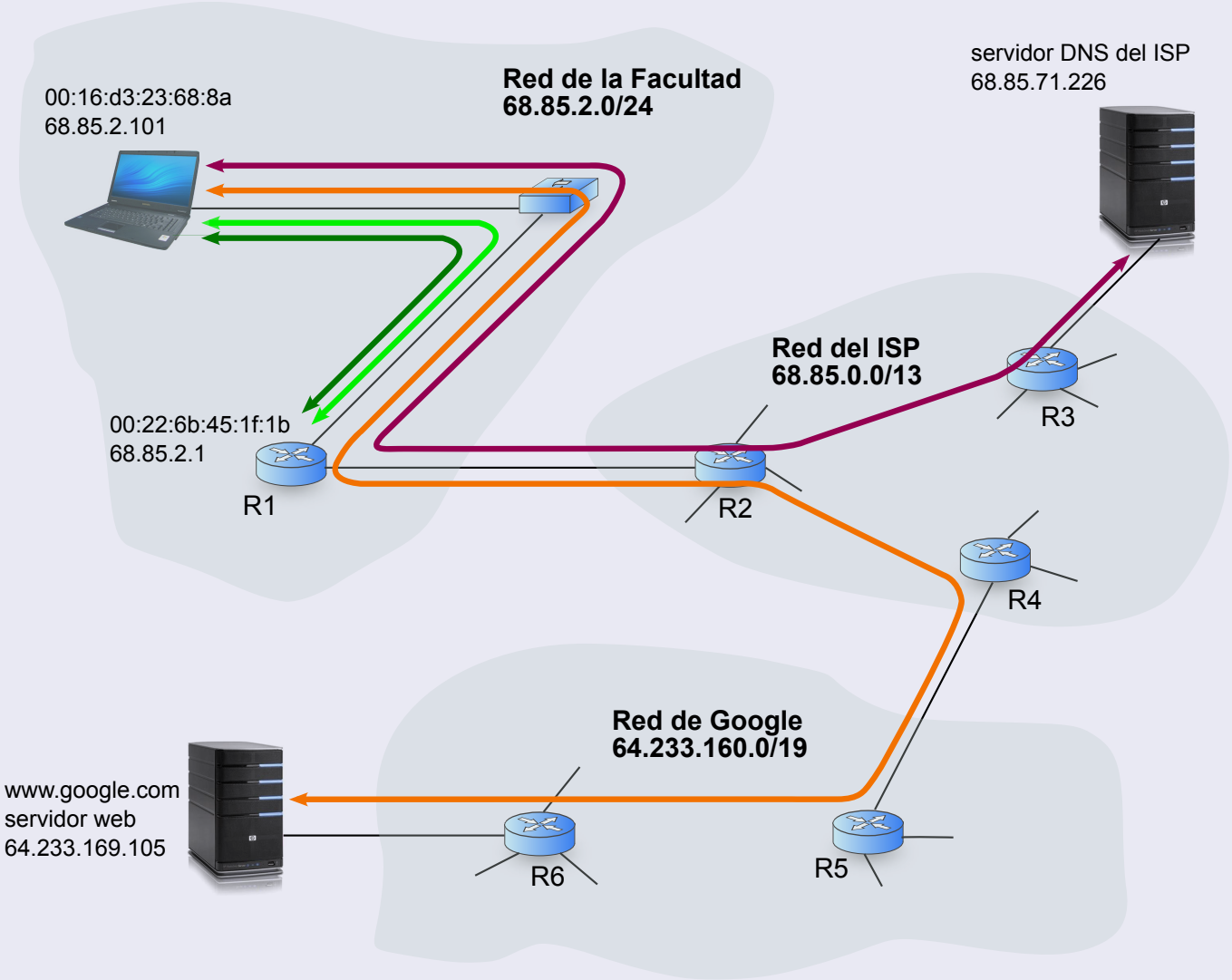
- Dos formatos: interfaz usuario-red e interfaz red-red
- Control de flujo genérico: para la QoS
  - Solo en la interfaz usuario-red
- Identificador de canal virtual, VPI y VCI
- Tipo de carga útil
  - Si la celda es de datos o de control
  - También indica si se detecta congestión
- Bit de prioridad de la celda, CLP
- Byte de control de error de la cabecera

Índice

1. Introducción
2. Modelo IEEE 802
3. Direcciones MAC Ethernet
4. Ethernet
5. Redes inalámbricas
6. Redes ATM
7. Ejemplo completo

Ejemplo completo: acceso a una página web

Esquema de la red



Acceso a una página web

Inicio: obtención de IP,...

- Mensaje de solicitud DHCP
  - En un segmento UDP, puerto destino 67, puerto origen 68
  - En un datagrama IP, IP destino 255.255.255.255, IP origen 0.0.0.0
  - En una trama Ethernet, MAC destino ff:ff:ff:ff:ff:ff, MAC origen 00:16:d3:23:68:8a (P)
    - 1ª trama que emite el portátil al conmutador Ethernet, que la reenvía por todos sus puertos
  - El router R1 (servidor de DHCP) recibe la trama, extrae el datagrama con IP destino la de broadcast y se la pasa a las capas superiores (se demultiplexa, se extrae el segmento UDP y se obtiene el mensaje DHCP)

Acceso a una página web

Inicio: obtención de IP,...

- El servidor DHCP crea un mensaje `ACK DHCP` con la IP asignada, la del DNS, la de la puerta de enlace por defecto y la máscara
  - En un segmento UDP, en un datagrama IP
  - En una trama Ethernet, MAC destino 00:16:d3:23:68:8a (P), MAC origen 00:22:6b:45:1f:1b (R1)
    - Se envía al conmutador Ethernet, que ya sabe cómo reenviarla al portátil
  - El portátil recibe la trama, extrae el datagrama, extrae el segmento UDP y obtiene el mensaje `ACK DHCP`

Acceso a una página web

Inicio: DNS, ARP

- El portátil intenta crear el socket TCP para la solicitud HTTP, pero necesita la IP
- Mensaje de consulta de DNS “www.google.com”
  - En un segmento UDP, puerto destino 53
  - En un datagrama IP, IP destino 68.87.71.226 (DNS), IP origen 68.85.2.101 (P)
  - En una trama Ethernet al router, MAC destino ? $\Rightarrow$ ARP
- Consulta ARP con la IP del router por defecto (68.85.2.1)
  - En una trama Ethernet con MAC destino ff:ff:ff:ff:ff:ff, que el conmutador entrega a todos
  - El router R1 recibe la trama con la petición ARP con su propia IP

Acceso a una página web

Inicio: DNS, ARP

- El router R1 prepara la respuesta ARP: la IP 68.85.2.1 se corresponde con la MAC 00:22:6b:45:1f:1b
  - En una trama Ethernet con MAC destino 00:16:d3:23:68:8a (el portátil)
  - El portátil recibe la trama y obtiene la MAC del router por defecto
- El portátil completa la trama con el mensaje de consulta de DNS
  - En un segmento UDP, puerto destino 53
  - En un datagrama IP, IP destino 68.87.71.226 (DNS), IP origen 68.85.2.101 (P)
  - En una trama Ethernet con MAC destino 00:22:6b:45:1f:1b (R1)

Acceso a una página web

Inicio: DNS

- El router R1 recibe la trama y extrae el datagrama con la consulta DNS con destino 68.87.71.226 y determina según su tabla que debe reenviarlo al router R2 de la red del ISP
  - En una trama de capa de enlace según el tipo de enlace existente entre esos dos routers
- R2 recibe la trama y extrae el datagrama con IP destino 68.87.71.226 y determina la interfaz por la que debe reenviarlo
  - Esta tabla se rellenó con el protocolo intradominio del ISP (RIP, OSPF) y con BGP
- El datagrama con la consulta al DNS llega al servidor DNS
  - Extrae la consulta y busca www.google.com en su base de datos
  - Encuentra el registro que contiene (www.google.com, 64.233.169.105, A)

Acceso a una página web

Inicio: DNS

- El servidor genera un mensaje con la correspondencia obtenido
  - En un segmento UDP, que inserta en un datagrama con destino el portátil (68.85.2.101)
  - Este datagrama se reenvía a través de la red del ISP hasta el router R1
  - Desde R1 se reenvía al portátil a través del conmutador Ethernet
  - El portátil extrae la dirección IP de www.google.com

Acceso a una página web

Interacción cliente servidor: TCP y HTTP

- El portátil puede crear el socket TCP que usará para el mensaje `GET` de HTTP
- Un segmento TCP `SYN` con puerto destino 80 (acuerdo en 3 fases)
  - En un datagrama con destino www.google.com (64.233.169.105)
  - En una trama Ethernet con MAC destino 00:22:6b:45:1f:1b (R1, router por defecto) al conmutador
- Los routers R1, los del ISP y los de Google reenvían el datagrama

Acceso a una página web

Interacción cliente servidor: TCP y HTTP

- El host www.google.con recibe el datagrama con el segmento TCP `SYN`
  - Se extrae el segmento TCP `SYN`
  - Se demultiplexa y se entrega al socket de acogida en el puerto 80
  - Se crea el socket de conexión TCP entre el servidor HTTP de Google y el portátil
  - Se genera un segmento TCP `SYNACK`
    - En un datagrama con IP destino la del portátil
    - En una trama de enlace según la conexión de www.google.com con su router de primer salto
  - El datagrama con el segmento `SYNACK` se envía a través de las redes y se recibe en la tarjeta Ethernet del portátil
    - Se extrae el datagrama, el segmento TCP y se demultiplexa al socket creado para iniciar la conexión

Acceso a una página web

Interacción cliente servidor: TCP y HTTP

- El navegador del portátil genera el mensaje HTTP `GET` con el URL que desea
  - Este mensaje se escribe en el socket, es decir, pasa a ser la carga útil de un segmento TCP
  - El segmento se incluye en un datagrama
  - Se entrega a www.google.com
- El servidor HTTP en www.google.com lee el mensaje HTTP `GET` del socket de conexión
- Crea un mensaje HTTP de respuesta, insertando la página web solicitada en el cuerpo del mensaje y lo envía a través del socket
- El datagrama con la respuesta se envía al portátil
- El navegador lee la respuesta del socket, extrae el código HTML de la página y la muestra