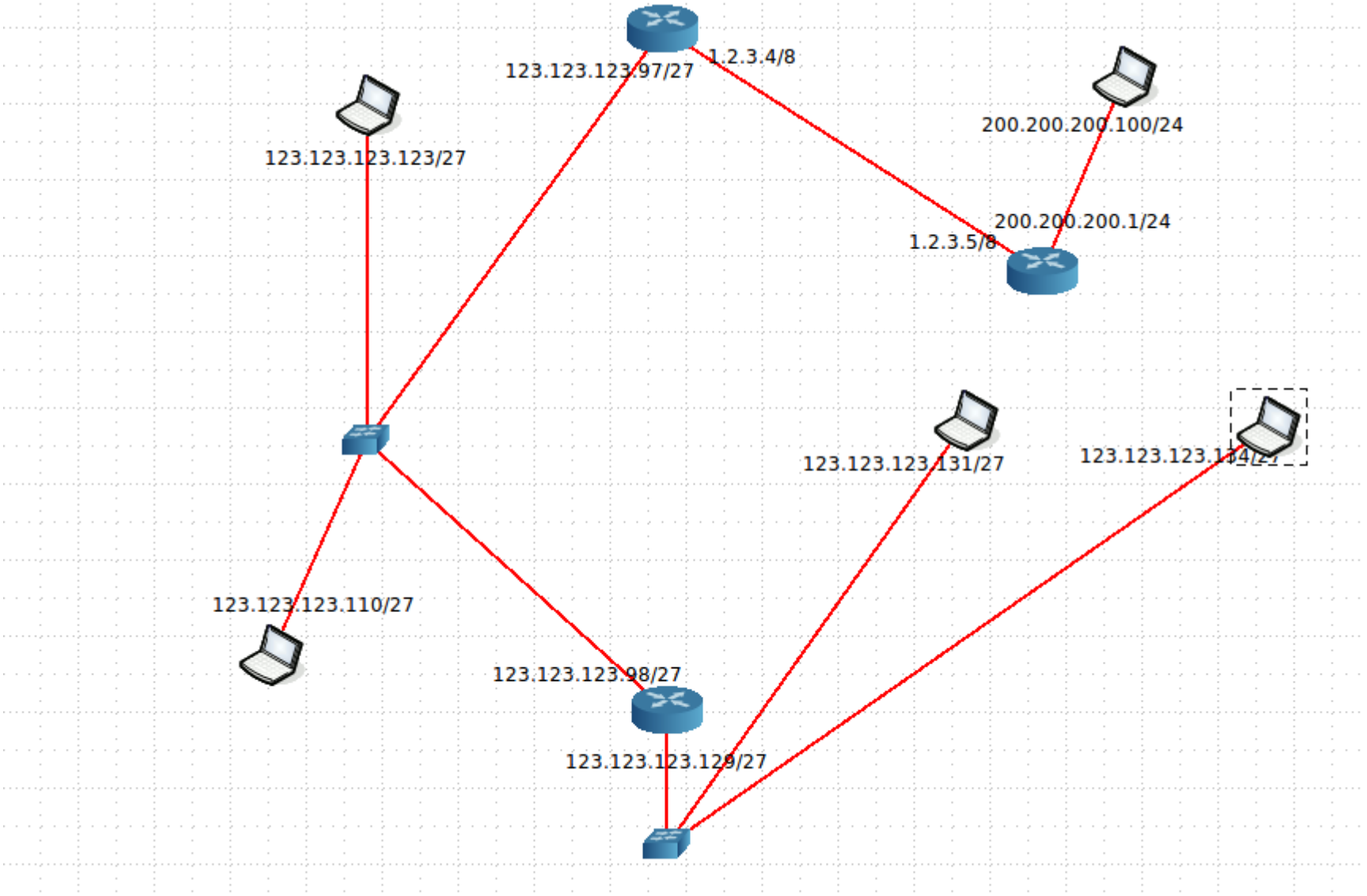
REDES DE COMPUTADORAS
2a. parte.

- Ruteo
- Protocolo ARP
- Protocolos de Transporte

Tablas de Ruteo

Protocolo ARP para convertir direcciones IP a direcciones MAC

Tablas de Ruteo



Tablas de ruteo I

- Supongamos un sistema conectado a una red local , que a su vez mediante un router se conecta con otras redes como el que se muestra en la transparencia anterior
- Para poder determinar como debe comunicarse con otros sistemas usando TCP/IP, debe conocer al menos su propia dirección IP, la red a la que está conectado y la dirección IP del router que vincula su red con otras redes.
- La red a la que pertenece la determina usando el par dirección IP (IP) / Mascara (MASK) de la siguiente manera:
  - RED = IP&MASK
  - Ejemplo IP=123.123.123.123 =0x7B7B7B7B
  - MASK= 255.255.255.224 =0xFFFFFFE0
  - RED=0x7B7B7B7B&0xFFFFFFE0=0x7B7B7B60
  - RED=123.123.123.96

Tablas de ruteo II

- Si el sistema anterior desea comunicarse con otro sistema, lo primero que hace es determinar si ambos están en la misma red, utilizando el mismo mecanismo anterior
  - IP1 = 123.123.123.123 = 0x7B7B7B7B
  - IP2 = 123.123.123.110 = 0x7B7B7B6E
  - MASK = 255.255.255.224 = 0xFFFFFFE0
  - Si IP1 & MASK= IP2 & MASK => IP1 e IP2 pertenecen a la misma red
  - 0x7B7B7B7B & 0xFFFFFFE0 = 0x7B7B7B60
  - 0x7B7B7B6E & 0xFFFFFFE0 = 0x7B7B7B60
- En este caso están en la misma red

Tablas de ruteo III

- Volvemos a utilizar el mecanismo pero para otra IP
  - IP1 = 123.123.123.123 = 0x7B7B7B7B
  - IP3 = 123.123.123.131 = 0x7B7B7B83
  - MASK = 255.255.255.224 = 0xFFFFFFE0

  - 0x7B7B7B7B & 0xFFFFFFE0 = 0x7B7B7B60
  - 0x7B7B7B83 & 0xFFFFFFE0 = 0x7B7B7B80
- En este caso NO están en la misma red

Tablas de ruteo IV

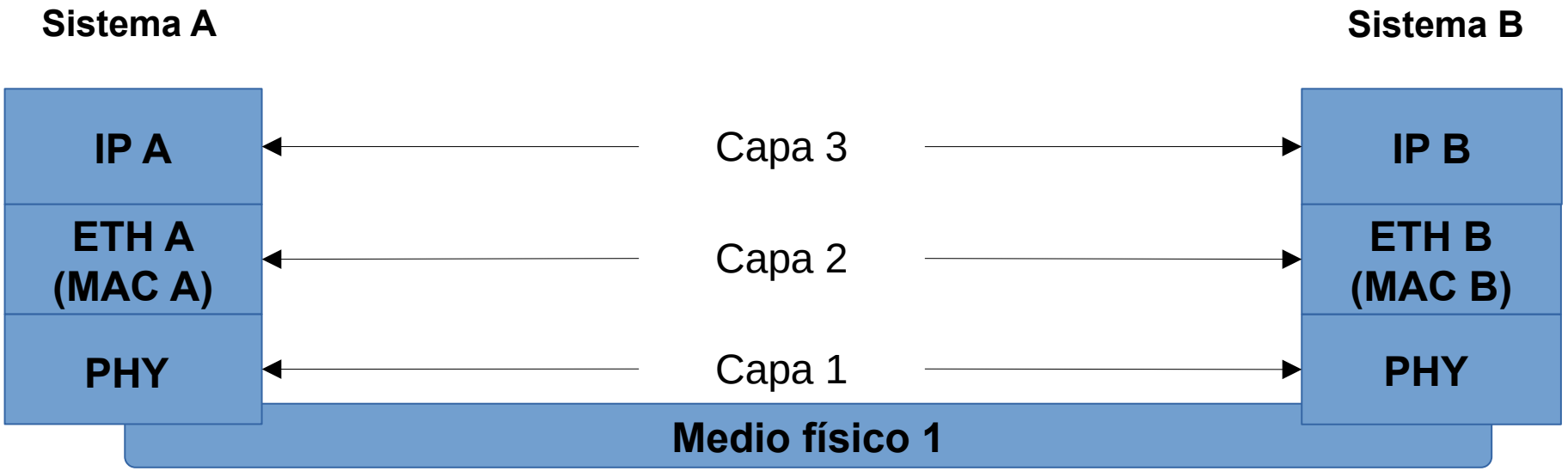
- El sistema, tendrá una tabla, llamada tabla de ruteo, que le indica que hacer en cada uno de esos casos
- Lo más común es que en la tabla se indique por que interfaz de red debe dirigirse la comunicación si está en la misma red.
- Si el destino no está en la misma red, entonces se indica la dirección del ***router***, **gateway** (GW) o ***pasarela*** por la que deben dirigirse los mensajes hacia otra red
- Puede haber más de un GW que conecte con distintas redes.
- Es habitual el uso de un ***default gateway*** (DG) que se encargara de enrutar el trafico hacia las redes que no conozca el sistema

Tablas de ruteo V

- En este caso la tabla tiene tres rutas
- La primera dirige a traves del router 123.123.123.97 todo el tráfico hacia redes desconocidas
- La segunda dirige todo el tráfico hacia la red 123.123.123.128/255.255.255.224 a través del router 123.123.123.98
- La tercera indica que el trafico hacia la red 123.123.123.96/255.255.255.224 es LOCAL y que puede accederse directamente a través de la interfaz de red.

Destino	Pasarela	Genmask	Indic	Interfaz
0.0.0.0	123.123.123.97	0.0.0.0	UG	eth0
123.123.123.128	123.123.123.98	255.255.255.224	UG	eth0
123.123.123.96	0.0.0.0	255.255.255.224	U	eth0

Direcciones físicas y direcciones IP – ARP



- Sistema A quiere enviar un paquete IP a sistema B
- IP A, con su tabla de ruteo, determina que IP B está en su misma LAN y con esa información se lo pasa a ETH A (capa 2).
- ETH A conoce su propia MAC A, pero necesita saber la dirección MAC B correspondiente a IP B para enviarle el paquete

Direcciones físicas y direcciones IP – ARP

- Se deben usar direcciones físicas (MAC) para comunicarse sobre la red local
- Las aplicaciones usan direcciones IP
- El software que implementa el protocolo de LAN necesita un mecanismo que convierta una dirección IP a una dirección de hardware equivalente
- Conocido como el “problema de resolución de dirección”

Direcciones físicas y direcciones IP – ARP

- Resolución de Direcciones
  - Se realiza en cada paso a lo largo del recorrido del paquete a través de Internet.
  - Hay dos algoritmos básicos:
    - Mapeo directo
    - Asociación dinámica
  - La elección depende del tipo de hardware

Direcciones físicas y direcciones IP – ARP

- Mapeo Directo
  - Facil de entender
  - Eficiente
  - Solo funciona si la dirección de hardware es un número pequeño
  - Tecnica: Asignar a la computadora una dirección IP que codifique la dirección de hardware.
    - P.Ej, si la dirección HW es un número de 0 a 255, asignar como ultimo octeto de la IP, la dirección de HW.

Direcciones físicas y direcciones IP – ARP

- Asociación Dinámica (Dynamic Binding)
  - Se necesita cuando las direcciones de Hardware son grandes o no tienen una relación directa con la cantidad de máquinas en la LAN (por ej. Ethernet)
  - Permite al sistema A encontrar la dirección de hardware del sistema B bajo estas condiciones:
    - A comienza con la dirección IP de B
    - A sabe que B está en la red local
  - La tecnica es enviar una interrogación a todas las maquinas en la LAN, y obtener una respuesta
  - Esto funciona solamente en una LAN a la vez, NO CRUZA LOS LÍMITES DE SUBREDES.

Direcciones físicas y direcciones IP – ARP

- Address Resolution Protocol (ARP)
  - Estándar para la resolución dinámica de direcciones en Internet
  - Requiere la posibilidad de realizar un broadcast de Hardware en la LAN.
  - Idea IMPORTANTE:
    - ARP se usa solamente para obtener direcciones físicas dentro de una red de área local (LAN), nunca a traves de múltiples subredes.

Direcciones físicas y direcciones IP – ARP

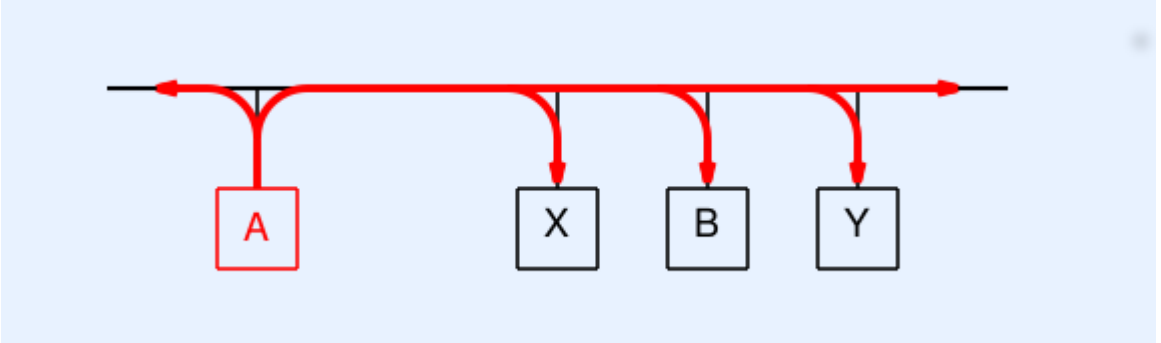
- Funcionamiento de ARP
  - La máquina A, realiza un requerimiento de ARP a toda la red (broadcast) con la dirección IP de la máquina B.
  - Todas las maquinas en la LAN reciben el pedido
  - La máquina B, que conoce su propio número IP, responde con su dirección física.
  - La Máquina A agrega la información de direcciones IP y Física de la máquina B a una tabla.
  - La máquina A, envía ahora los paquetes destinados a la dirección IP de la máquina B, encapsulados en un paquete físico con la dirección de hardware de B.

Direcciones físicas y direcciones IP – ARP

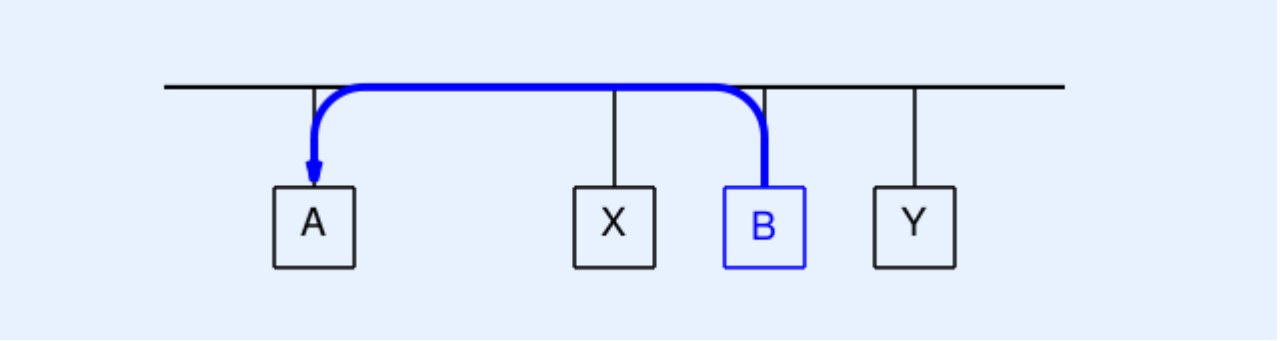
- Funcionamiento de ARP
  - La máquina A, realiza un requerimiento de ARP a toda la red (broadcast) con la dirección IP de la máquina B.
  - Todas las maquinas en la LAN reciben el pedido
  - La máquina B, que conoce su propio número IP, responde con su dirección física.
  - La Máquina A agrega la información de direcciones IP y Física de la máquina B a una tabla.
  - La máquina A, envía ahora los paquetes destinados a la dirección IP de la máquina B, encapsulados en un paquete físico con la dirección de hardware de B.

Direcciones físicas y direcciones IP – ARP

Ejemplo de mensajes de requerimiento y respuesta en ARP



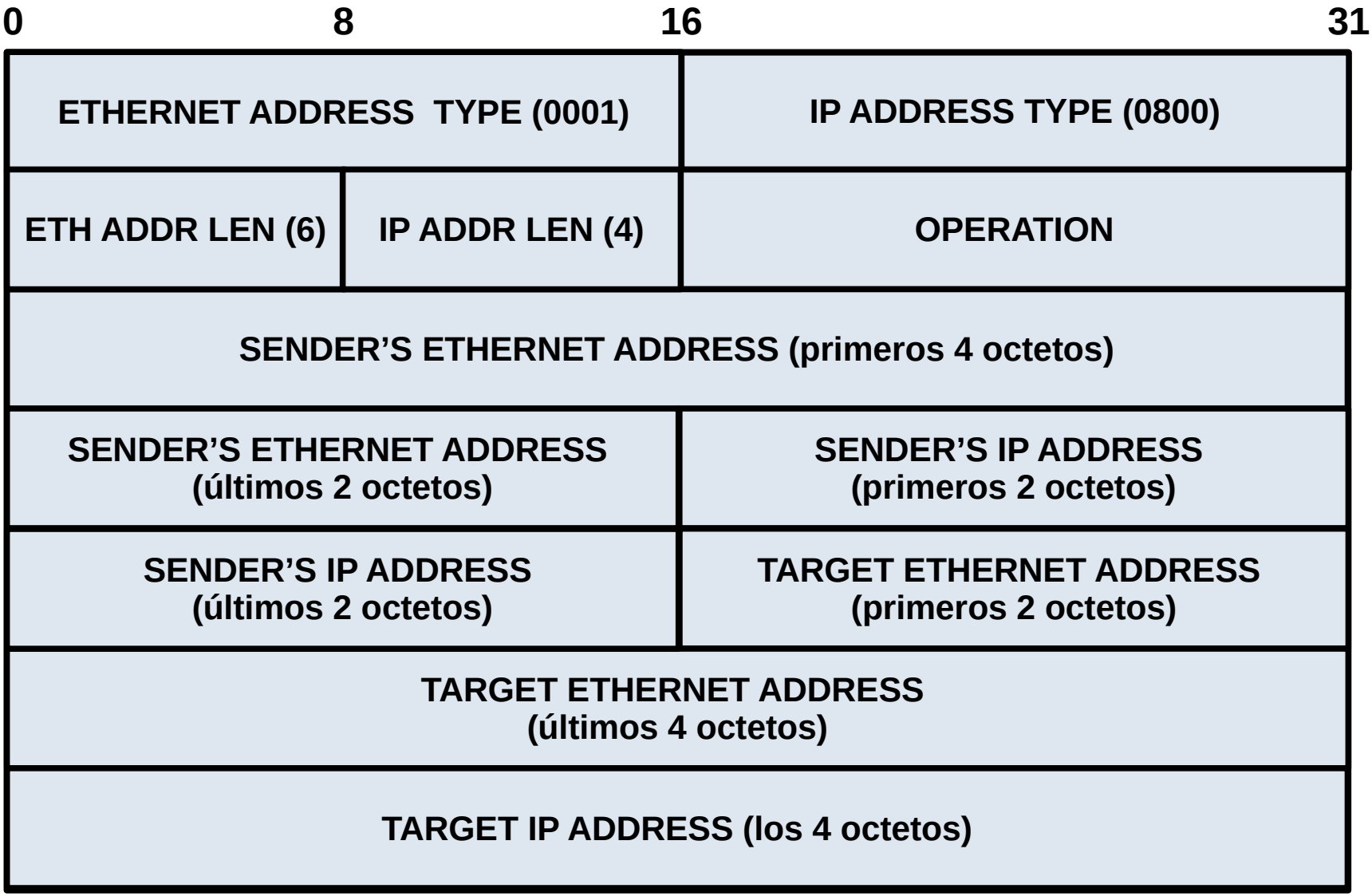
Mensaje dirigido a la dirección de broadcast, requiriendo la dirección física de B (solo transita dentro de la misma subred)



B responde al requerimiento

Direcciones físicas y direcciones IP – ARP

Formato de paquete ARP cuando se usa con Ethernet



Direcciones físicas y direcciones IP – ARP

- Observaciones acerca del formato del paquete
  - Es general: esto es que puede ser usado con:
  - Direcciones de Hardware arbitrarias (no sólo Ethernet)
  - Protocolos de red arbitrarios (No sólo IP)
  - Posee campos de longitud variable (depende del tipo de direcciones)
  - Los campos que indican longitud, permiten el procesamiento del paquete por computadoras que no “entiendan” los diferentes tipos de direcciones

Direcciones físicas y direcciones IP – ARP

- Retención de la asociación
  - No es razonable enviar requerimientos de ARP para cada paquete.
  - Solución:
    - Mantener una tabla de asociaciones
  - Efecto
    - Usar ARP una vez, armar una tabla con las asociaciones encontradas y despues enviar muchos paquetes, usando esta tabla

Direcciones físicas y direcciones IP – ARP

- Caché de ARP
  - La tabla de asociaciones ARP es una caché
  - Las asociaciones encontradas tienen un time-out y cuando expiran son removidas
  - Evita asociaciones inválidas
  - Timeout típico: 20 minutos

Direcciones físicas y direcciones IP – ARP

- Algoritmo para procesar requerimientos ARP
  - Extraer el par de direcciones de origen (IP_A, HW_A) y actualizar la tabla de ARP local si existe.
  - Si este es un pedido y está dirigido a mí:
    - Agregar el par del que envía a mi tabla de ARP si no está desde antes.
    - Reemplazar la dirección de broadcast con mi dirección de hardware
    - Intercambiar los campos dirección de origen y destino
    - Establecer la operación en el paquete ARP a “respuesta”
    - Enviar la respuesta a la máquina de origen

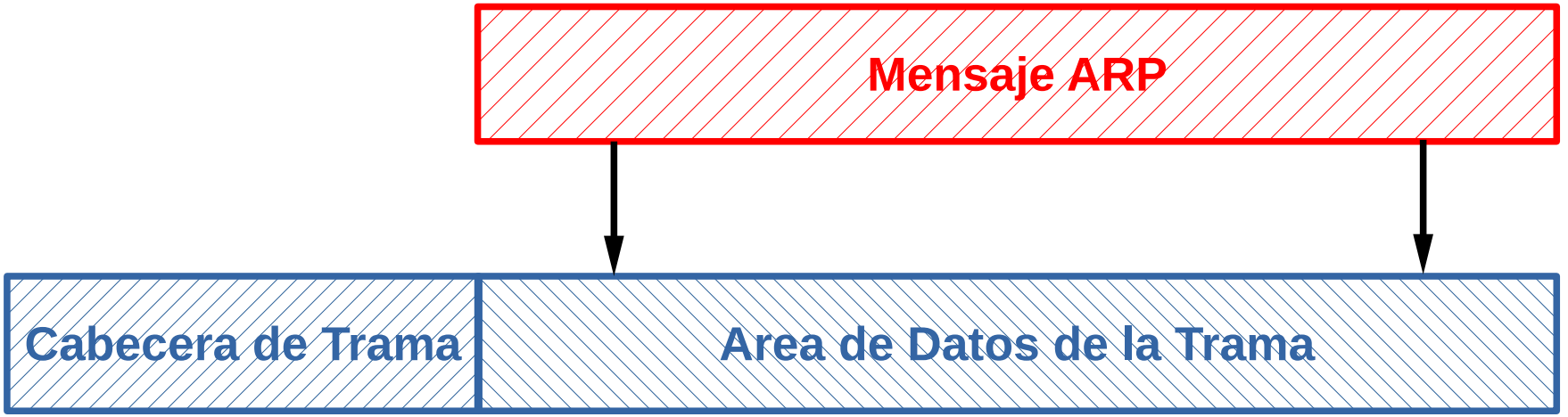
Direcciones físicas y direcciones IP – ARP

- Características del Algoritmo
  - Características del Algoritmo
  - Si A hace la operación ARP con B, B retiene la información de A
    - Esto se hace porque probablemente B envíe un paquete a A dentro de poco tiempo.
  - Si A hace la operación ARP con B, otras máquinas no retienen la información de A.
    - Esto se hace para no llenar las tablas de ARP innecesariamente.

Direcciones físicas y direcciones IP – ARP

- Propósito conceptual de ARP
  - Aislar las direcciones de hardware a bajo nivel
  - Permitir a las aplicaciones utilizar las direcciones IP
- Encapsulación de ARP
  - Los mensaje ARP viajan en la porción de datos de la trama que usa la LAN (por ej. Ethernet).
  - Se dice que el mensaje ARP está encapsulado (en una trama ethernet por ejemplo)

Ejemplo de encapsulación de ARP



Direcciones físicas y direcciones IP – ARP

- Encapsulación Ethernet
  - El mensaje ARP se ubica en el area de datos de la trama Ethernet.
  - El área de datos se rellena con ceros si el mensaje ARP es mas corto que la trama mínima de ethernet
  - El tipo de mensaje Ethernet que se usa para ARP es 0x0806

Direcciones físicas y direcciones IP – ARP

- Reverse Address Resolution Protocol (RARP)
  - Realiza la operación inversa a ARP: Mapea direcciones Ethernet a direcciones IP.
  - Usa el mismo formato de mensajes que ARP.
  - Se utiliza para asignarle direcciones IP a maquinas que no la tienen configurada.
  - La computadora envía su dirección Ethernet
  - El servidor RARP contesta con la dirección IP que se asigna a la computadora.
  - Casi no se usa más, fue reemplazado por DHCP

Direcciones físicas y direcciones IP – ARP

- RESUMEN
  - Las direcciones IP de los sistemas son independientes de las direcciones de hardware.
  - Las aplicaciones necesitan usar direcciones IP
  - El hardware de red (Capa 2) sólo interpreta direcciones de hardware.
  - Es necesario convertir o mapear las direcciones IP a direcciones de de Hardware para realizar las comunicaciones
  - Hay dos tipos de mapeo:
    - Mapeo directo
    - Asociación dinámica

Como se dirige un paquete IP a través de un router

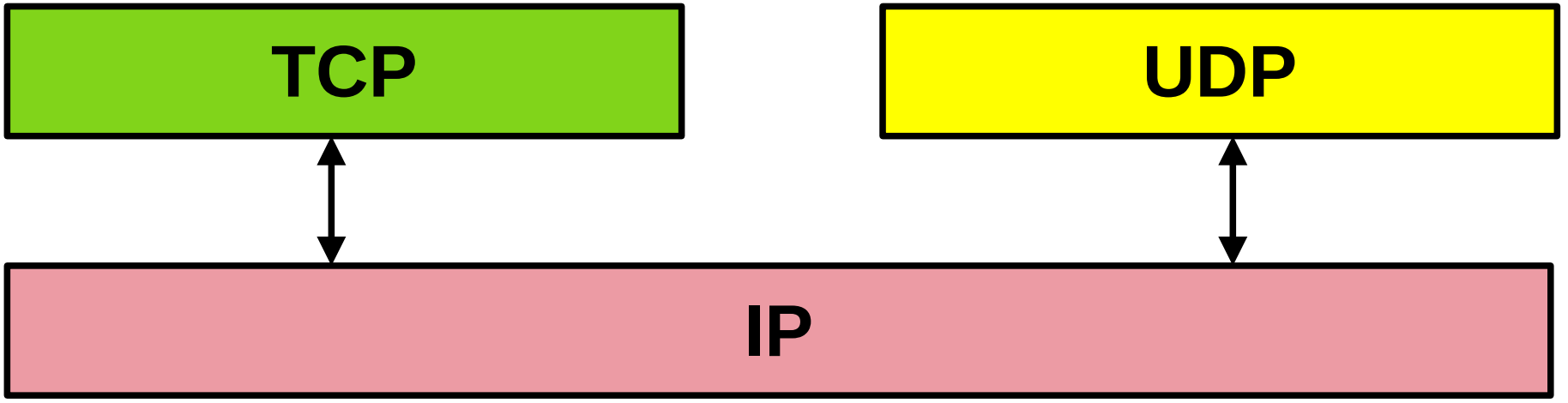
- Si el sistema determina que debe dirigir el paquete IP a través de un gateway o router, debe hacer lo siguiente:
  - Determinar la dirección Física del router a través de ARP
  - Encapsular el paquete IP en una trama Ethernet, con la direción de destino física (MAC) del router.
- El router recibirá el paquete IP encapsulado a través de la interfaz ethernet
- Cuando lo desencapsule, examinará la dirección IP de destino y hará lo siguiente:
  - Determinará si pertenece a una red directamente conectada a sus interfaces de red, en ese caso lo encapsulará en ethernet y lo enviará a la dirección destino directamente.
  - En caso contrario, revisará sus tablas para ver si tiene una ruta para esa dirección IP a través de otro router, si no la tiene lo enviará al ***default gateway,*** encapsulado en una trama ethernet con la dirección física que averigua utilizando ARP.

Protocolos de Transporte

- User Datagram Protocol (UDP)
  - Formato
  - Header
  - Pseudo Header

Protocolos de Transporte

- En TCP/IP se han definido dos protocolos de transporte:
- UDP – User Datagram Protocol
  - Protocolo de Datagramas de Usuario.
- TCP – Transmission Control Protocol
  - Protocolo de Control de Transmisión.



Necesidad de Protocolos de Transporte

- La dirección IP sólo especifica un Sistema o Computadora
- Se necesita una manera de especificar a qué aplicación o proceso está dirigida la comunicación.
- Además se debe tener en cuenta que:
  - Las aplicaciones o procesos pueden crearse o cerrarse rápidamente
  - Cada Sistema Operativo, usa su propio medio de identificación

Necesidad de Protocolos de Transporte

- Teiendo en cuenta lo anterior, TCP/IP presenta su propia especificación.
- Se define un punto de destino abstracto conocido como *número de puerto del protocolo*. (*protocol port number*)
  - Es un entero positivo de 16 bits
- Cada Sistema Operativo determina como asociar un numero de puerto del protocolo a una aplicación o proceso específico.

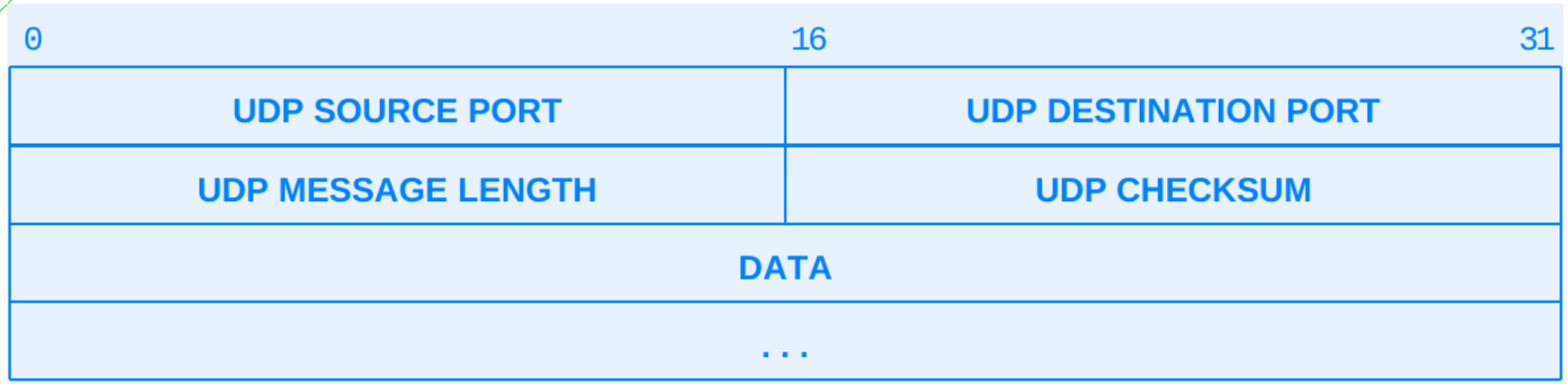
Protocolo UDP

- Es un protocolo de capa de transporte (capa 4)
- Provee un *servicio sin conexión* a los programas de aplicación.
  - Capacidad de enviar y recibir mensajes (datagramas).
  - Permite que multiples aplicaciones se comuniquen concurrentemente en la misma máquina.
  - Sigue la misma semántica de *mejor esfuerzo* que IP
    - Los mensajes pueden perderse, retrasarse o llegar duplicados
    - Los mensajes pueden llegar fuera de orden
- Las Aplicaciones deben tomar responsabilidad completa por los errores.

Protocolo UDP

- El Protocolo de Datagramas de Usuario (UDP) provee un servicio de entrega sin conexión y no confiable para transportar mensajes (datagramas) entre sistemas.
- Usa IP como contenedor para llevar los mensajes, pero suma la capacidad de distinguir entre múltiples destinos dentro de una computadora dada.

Formato del Mensaje UDP

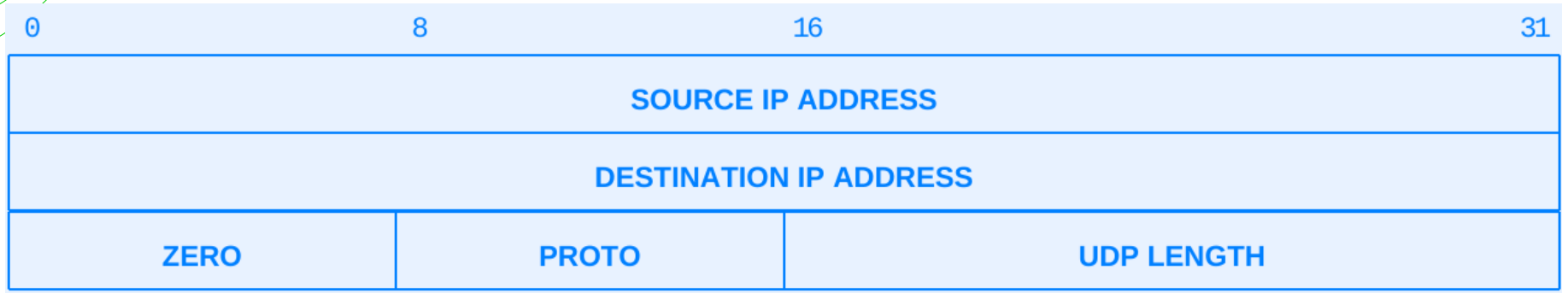


- Se definen dos campos UDP PORT
  - Origen (Identifica la aplicación en la computadora que **originó** el mensaje)
  - Destino (Identifica la aplicación en la computadora que **recibirá** el mensaje)
  - Cada uno ocupa 16 bits, por lo que el rango está entre 0 y 65535
  - Las direcciones IP de Origen y Destino NO está explicitamente indicada en la cabecera UDP.
- La longitud del mensaje UDP también se define como de 16 bits. (hasta 65535)
- Se aprecia que la cabecera es mínima.

Checksum y Pseudo – Header

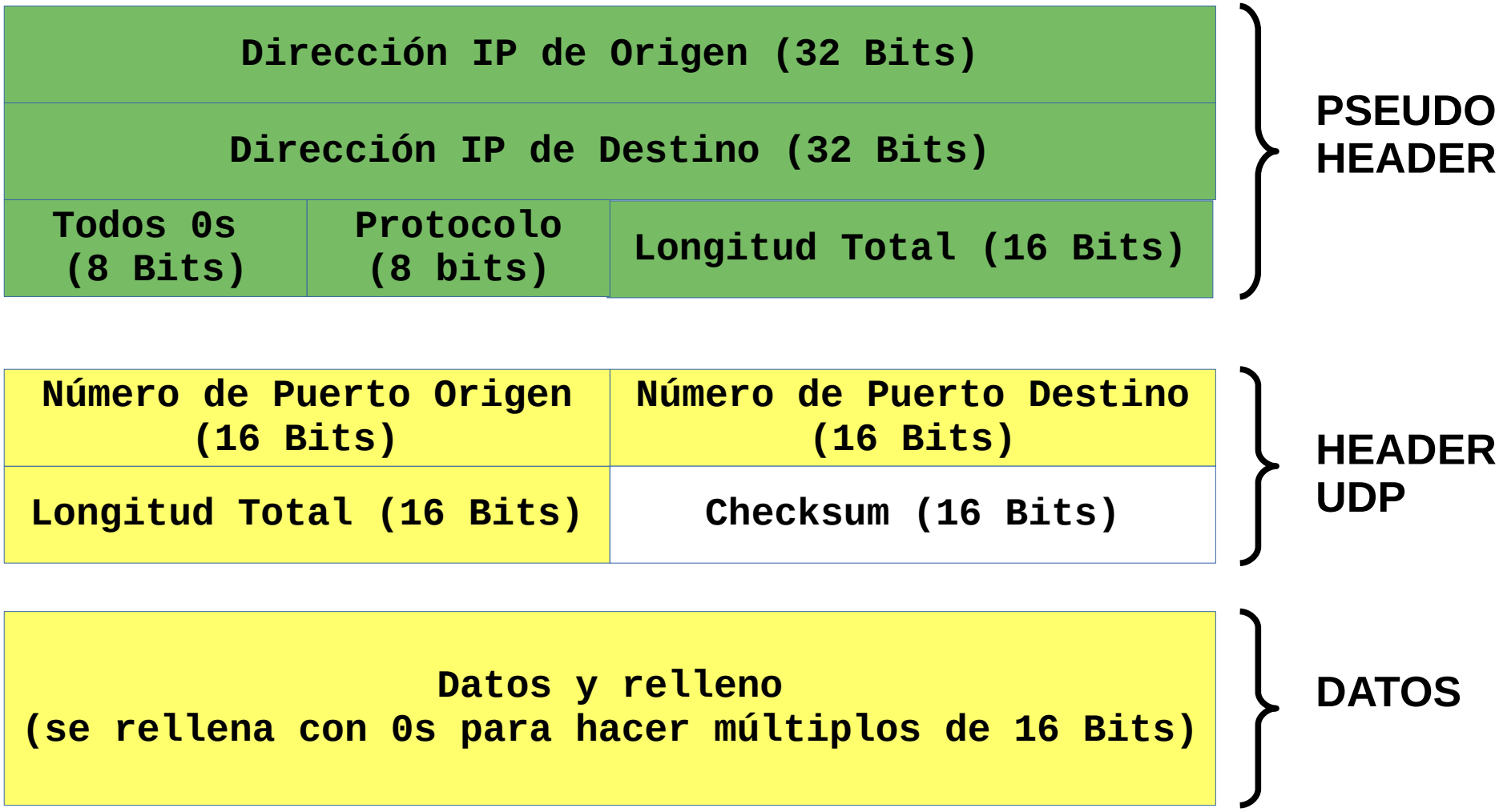
- Si el campo UDP CHECKSUM contiene ceros, el receptor no verifica el checksum
- Para el cálculo del Checksum se agrega temporariamente una seudo-cabecera (Pseudo Header) que contiene información extraida de la cabecera IP.
- Esta seudo cabecera no se envía por la red, sino que se agrega solamente para realizar el cálculo del checksum.
- Sirve para garantizar que llegue al destino correcto

Checksum y Pseudo – Header

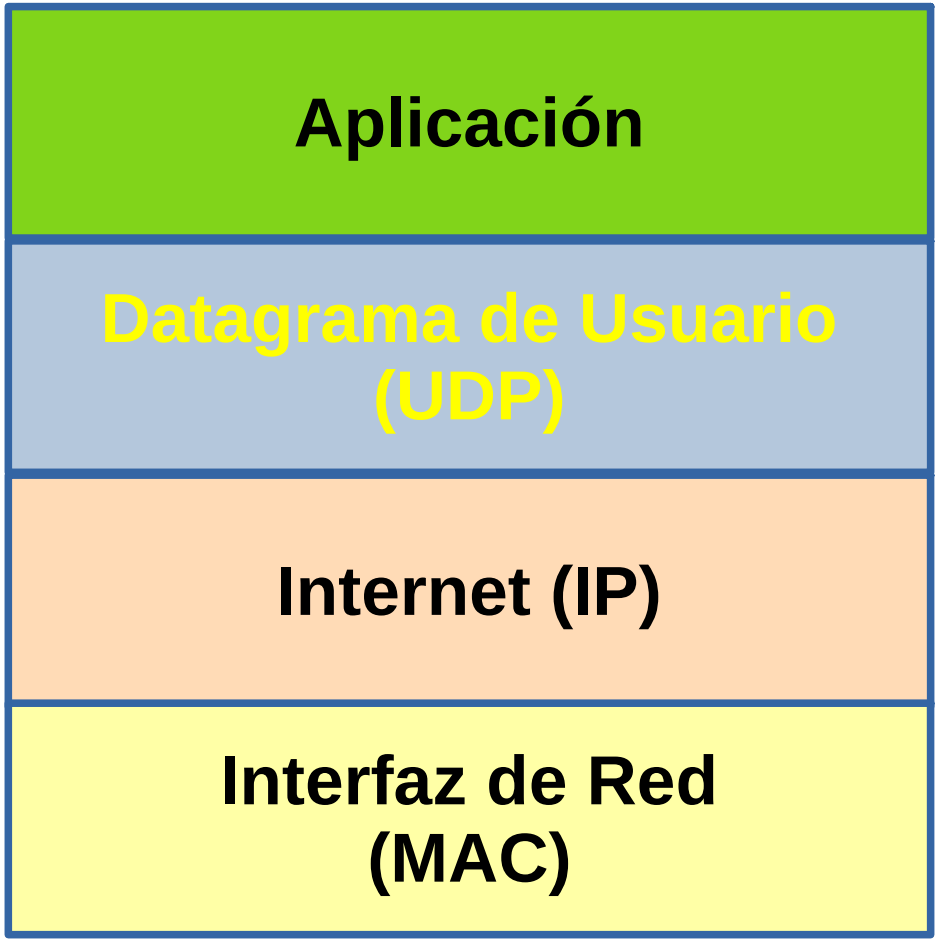


- SOURCE IP ADDRESS y DESTINATION IP ADDRESS, especifican las direcciones IP de la computadora Origen y Destino respectivamente
- PROTO se copia del campo TYPE de la cabecera del datagrama IP (17 para UDP).

Checksum y Pseudo – Header



Posición de UDP en la capa de protocolos



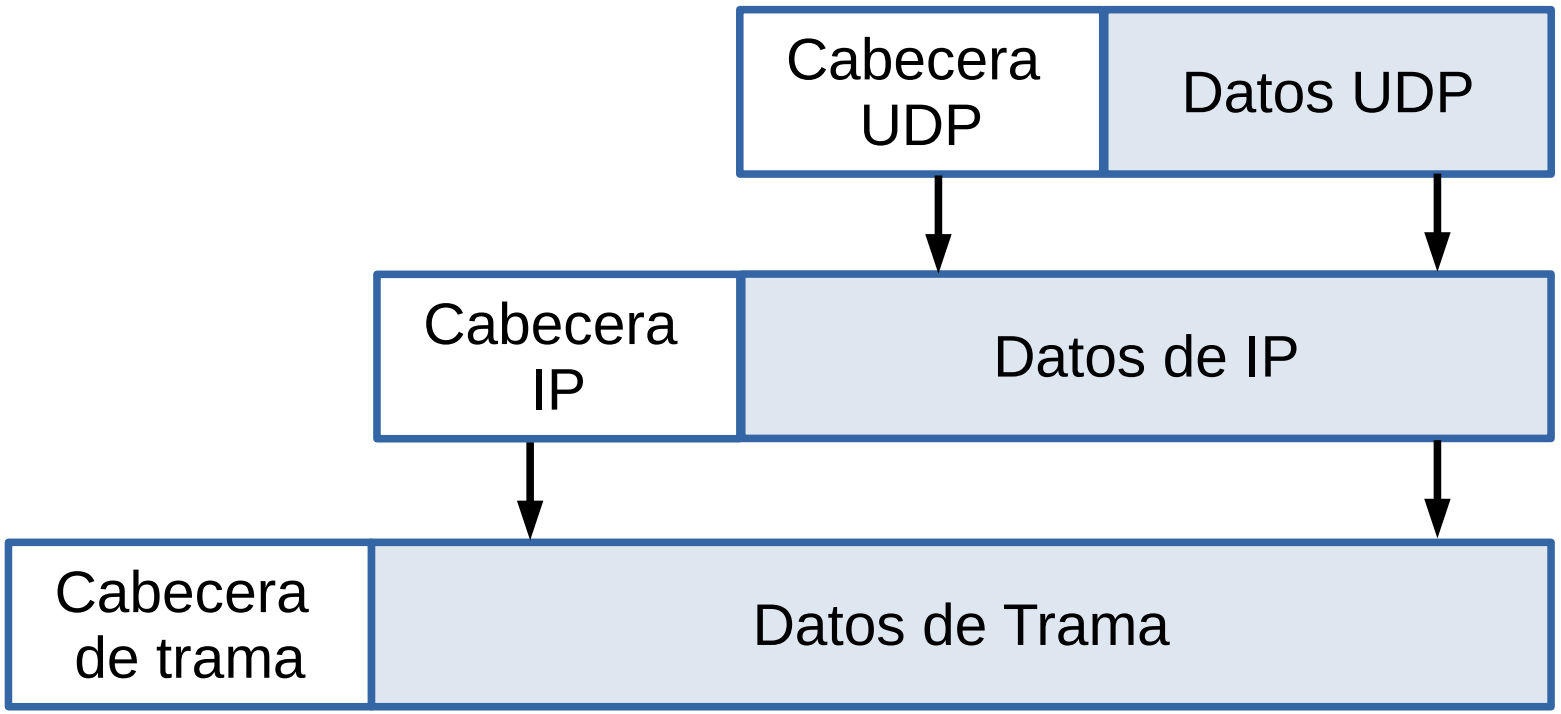
Capa de aplicación

Capa 4

Capa 3

Capas 1 y 2

Encapsulación de UDP

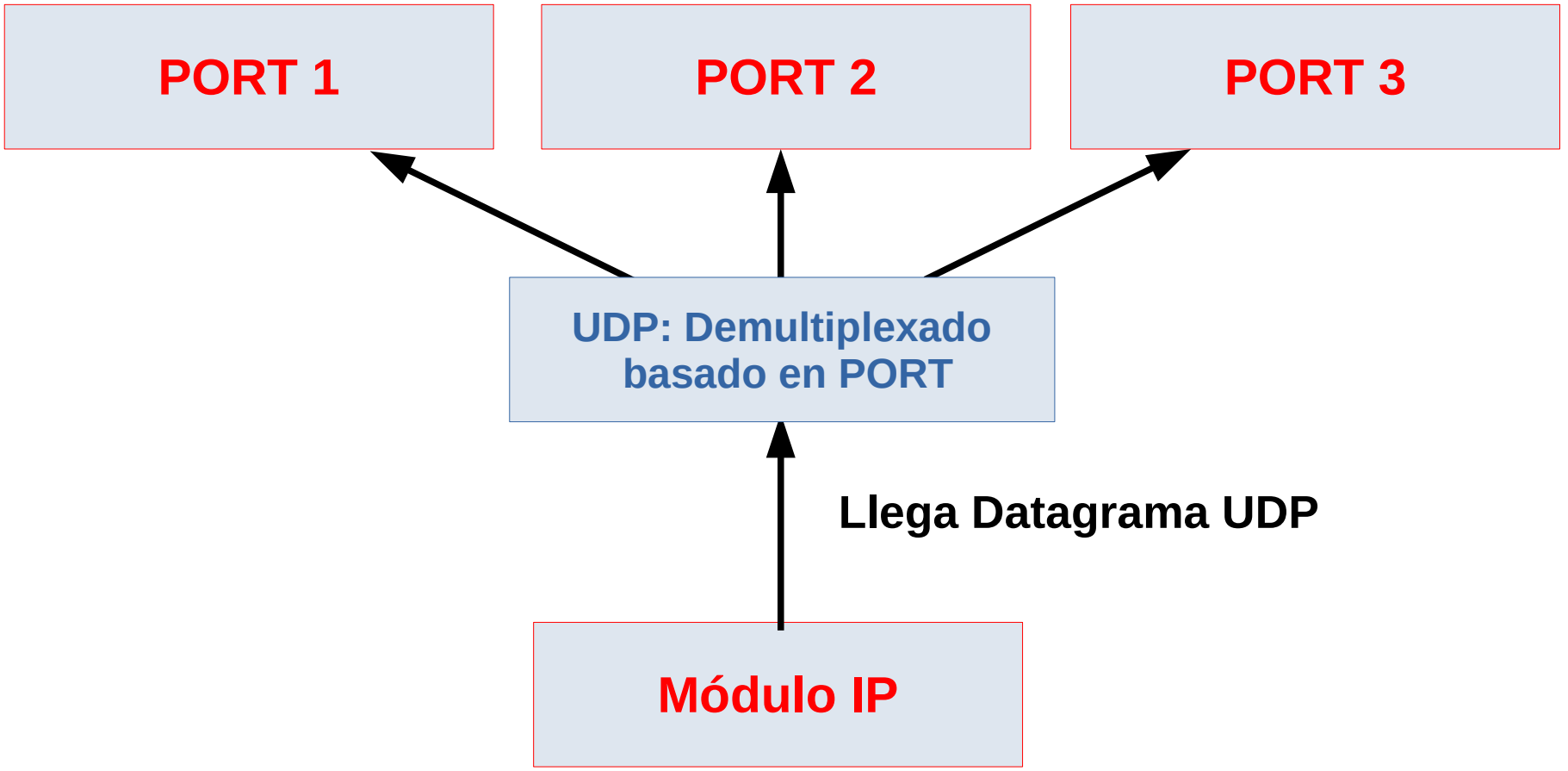


División de tareas entre IP y UDP

La capa IP es la responsable de transferir datos entre un par de sistemas en una internet, mientras que la capa UDP es la responsable de diferenciar entre múltiples fuentes o destinos dentro de cada sistema

- La cabecera IP solamente identifica sistemas (computadoras)
- La cabecera UDP solo identifica programas de aplicación.

Demultiplexado basado en Protocol Port Number (UDP)



Asignación de Números de Puerto UDP

- Los números entre 0 y 1023 se reservan para servicios específicos
  - Se los llama puertos bien conocidos (well-known ports)
  - Se interpretan de la misma manera en todo Internet
  - Son utilizados por el software de servidores estándar.
- Los números mayores que 1023 no están reservados.
  - Están disponibles para cualquier programa de aplicación.
  - Habitualmente son usados por el software de los clientes.
  - Tambien puede usarse en servidores no estándar

Ejemplos de Números de Puerto UDP asignados

Número	Palabra Clave	Servicio UNIX	Descripción
0	-	-	Reservado
7	ECHO	echo	Contesta lo mismo que recibe
9	DISCARD	discard	Descarta lo que recibe
13	DAYTIME	daytime	Fecha y hora
19	CHARGEN	chargen	Generador de caracteres
53	DOMAIN	nameserver	Domain name Server
67	BOOTPS	bootps	Servidor BOOTP o DHCP
68	BOOTPC	bootpc	Cliente BOOTP o DHCP
123	NTP	ntp	Network Time Protocol
161	-	snmp	Simple Network Management Protocol
162	-	snmp-trap	SNMP-Traps (auxiliar)
514	-	syslog	Registro del sistema

Resumen

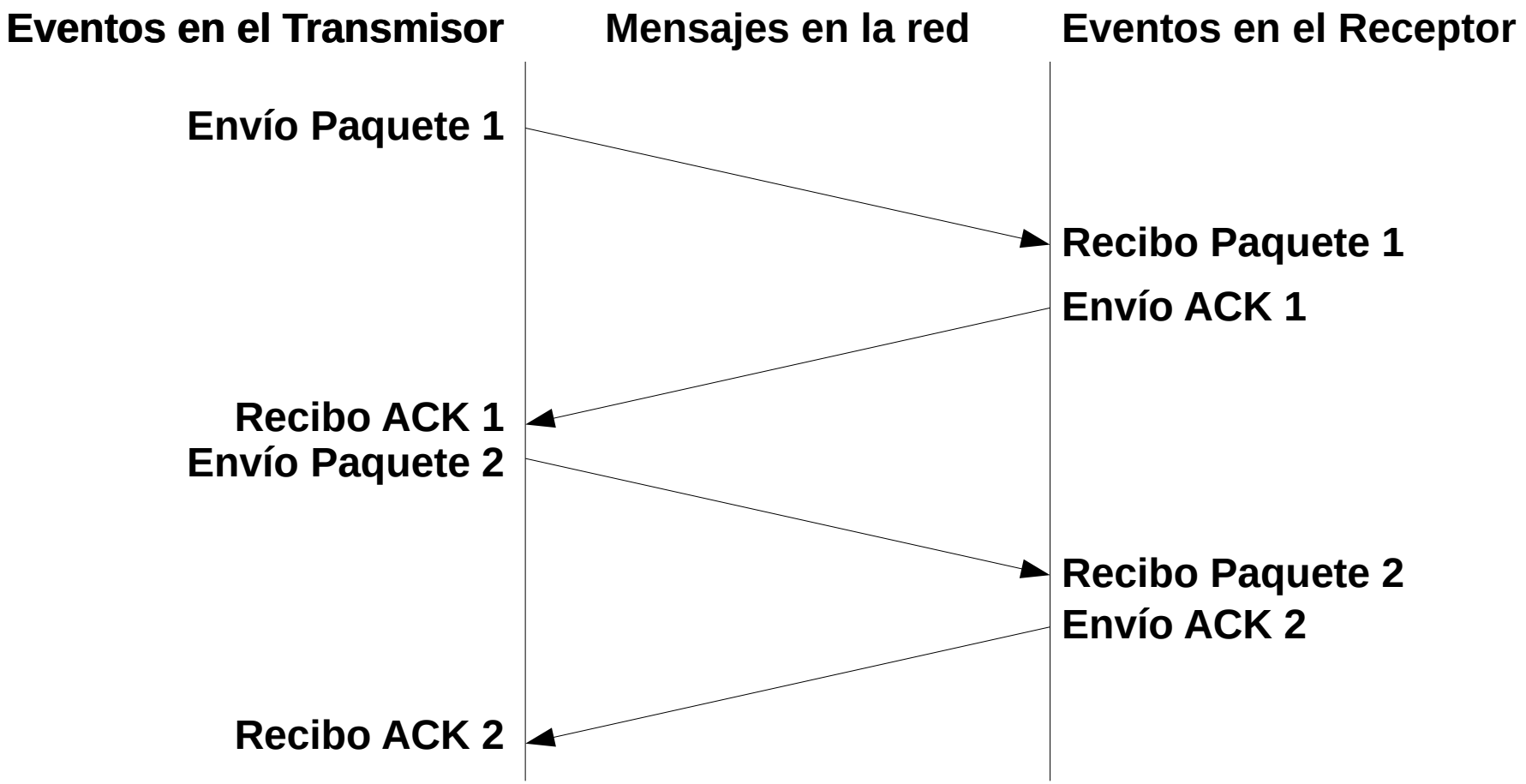
- UDP provee un servicio de mensajes (datagramas) basado en mejor esfuerzo y sin conexión.
- Los mensajes UDP se encapsulan en un datagrama IP
- IP identifica la computadora destino; UDP identifica la aplicación en la computadora destino
- UDP usa una abstracción llamada Número de Puerto del Protocolo.(protocol port numbers)
- Los números de puerto del protocolo inferiores a 1024 están reservados para protocolos de aplicación estándar (Well Known Ports)

Servicio de Transporte Confiable de Flujos de Datos
Transmission Control Protocol – TCP

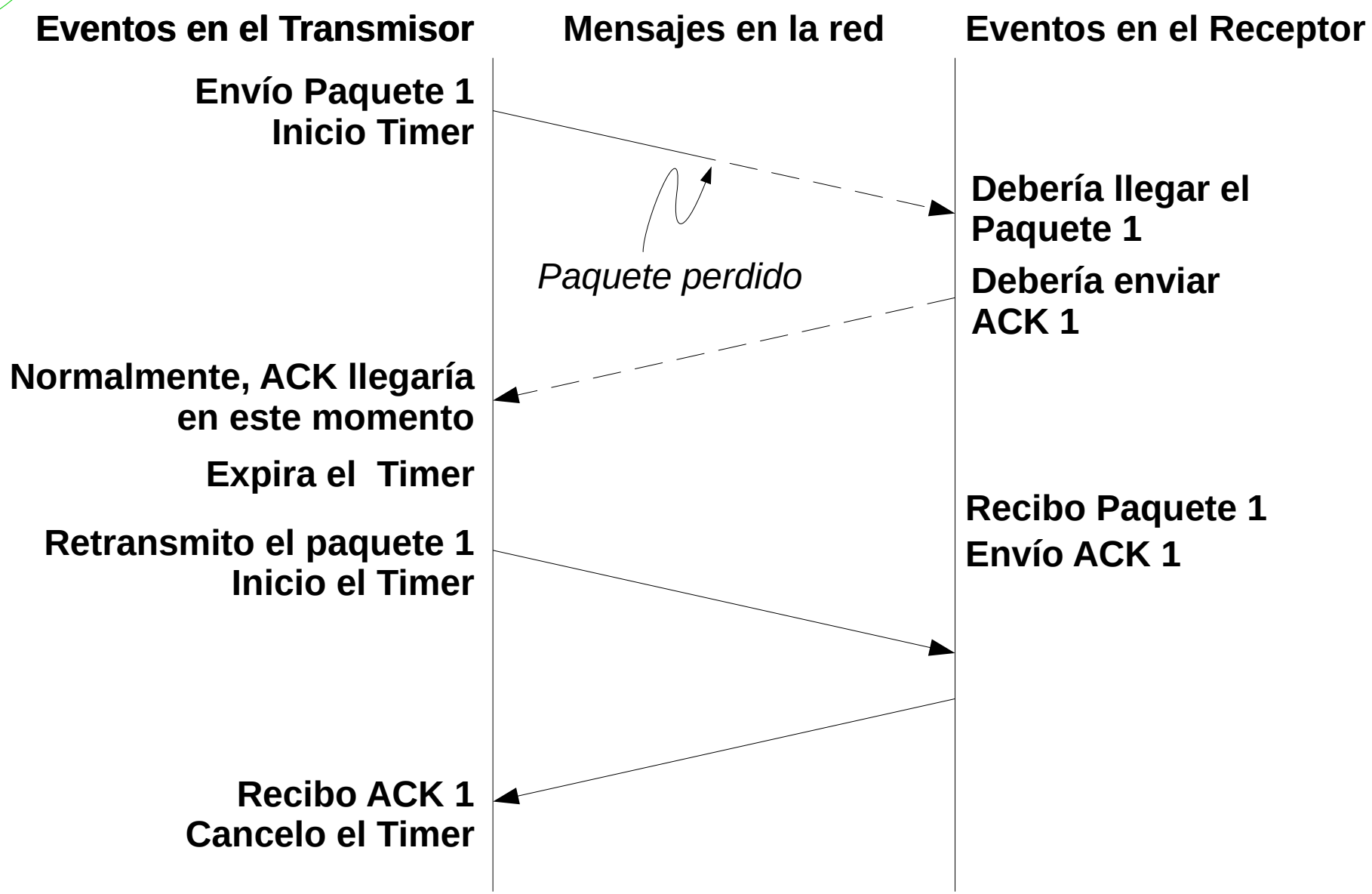
Características de TCP

- Flujo confiable:
  - Basado en conexión virtual
  - Reconocimiento Positivo con Retransmisión (Positive Acknowledgement & Retransmission – PAR)
- Optimización del uso de la red
  - Ventana Deslizante
  - Determinación del tiempo de Ida y Vuelta (Round Trip Time – RTT)
  - Time-Out = Función de RTT.
  - Control de Flujo
  - Control de Congestión

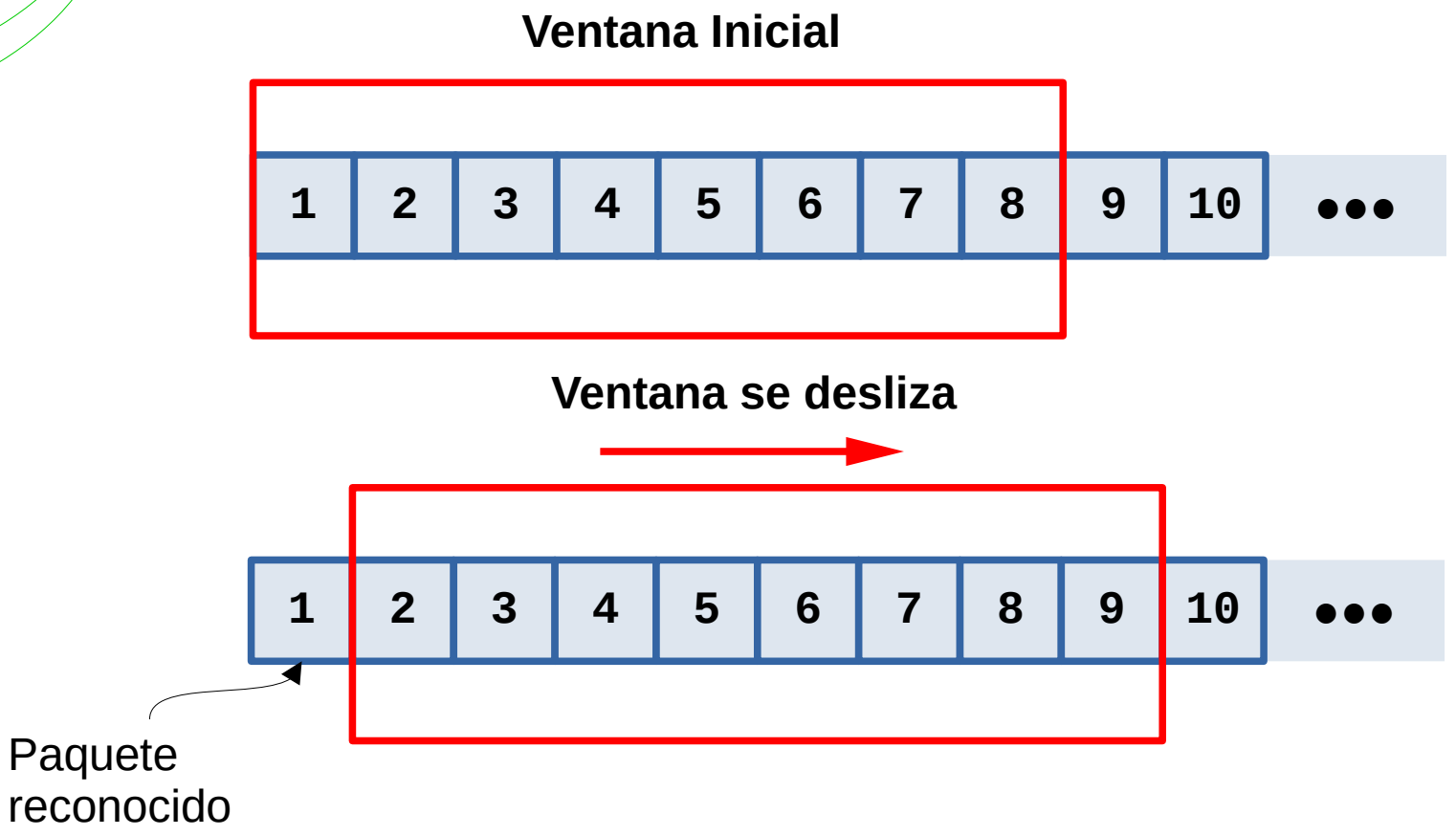
Reconocimiento Positivo con Retransmisión



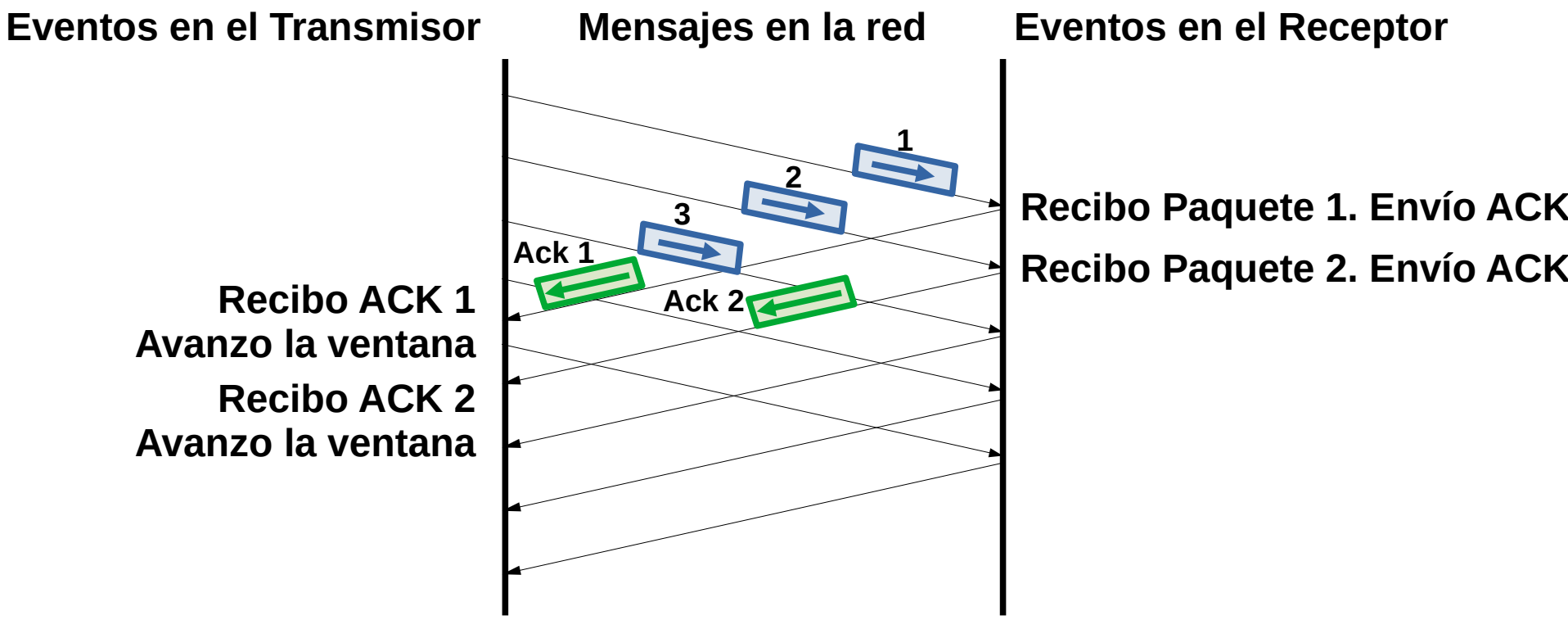
Reconocimiento Positivo con Retransmisión (falla)



Ventana Deslizante

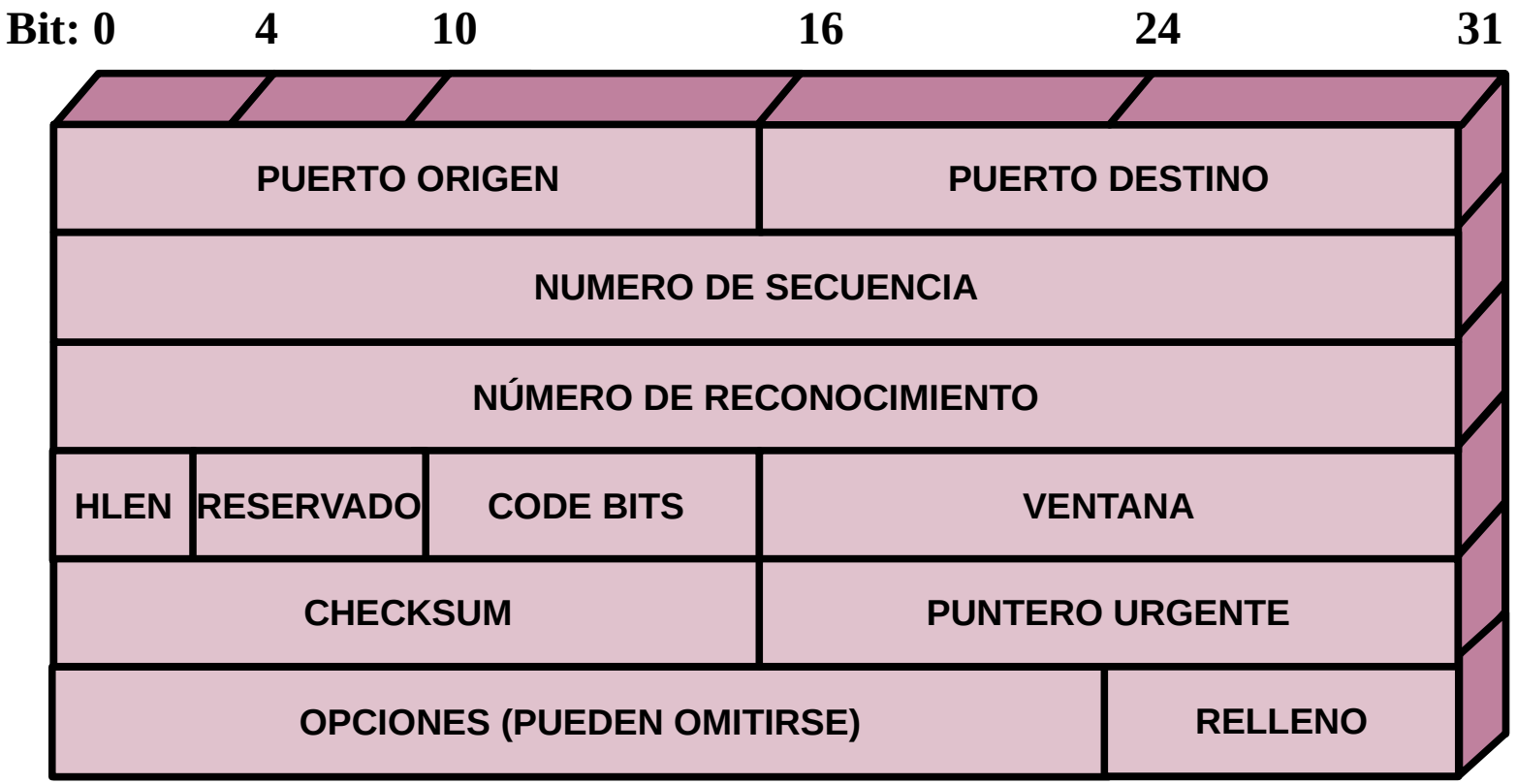


- Tamaño de la ventana fijo
- Cuando llega un reconocimiento, la ventana avanza



- Se puede observar que aumenta la utilización de la red.

Cabecera TCP



Características de TCP

- Inicio de Conexión confiable:
  - Inicio en tres pasos
- Flujos bidireccionales e independientes
  - Cierre de conexión independiente en cada sentido
- Canal de datos Urgentes
  - Se pueden insertar datos urgentes en el flujo de datos principal
- TCP se controla mediante una máquina de estados

Control de Flujo

- El RECEPTOR controla el flujo de datos que recibe.
  - Utiliza el “aviso de ventana” (advertised window) para comunicarle al transmisor cuantos datos adicionales puede recibir
  - Si el aviso de ventana es 0, el transmisor no debe transmitir más datos que los que tiene en la ventana en ese momento
  - El receptor puede habilitar la recepción enviando un ACK con el nuevo tamaño de ventana que admite.

Control de Congestión

- Está a cargo del TRANSMISOR
  - Utiliza los timeouts o la recepción de ACK duplicados para detectar la congestión en la red.
  - Usa un mecanismo llamado “decrecimiento multiplicativo y comienzo lento” para recuperarse de la congestión.

Decrecimiento multiplicativo:

- Se basa en utilizar para la transmisión un tamaño de ventana, llamado “ventana de congestión”
- Cuando no hay congestión , la ventana de congestión es igual a la ventana que comunicó el receptor en el “aviso de ventana”
- Siempre se utiliza la ventana menor entre la “ventana de congestión” y el “aviso de ventana” recibido
- Cuando detecta congestión, reduce la ventana de congestión a la mitad por cada retransmisión necesaria.
- Esto se repite hasta llegar a un “Maximum Segment Size” (MSS)
- MSS depende de la configuración de la red, suele usarse alrededor de 1500 bytes

Comienzo lento

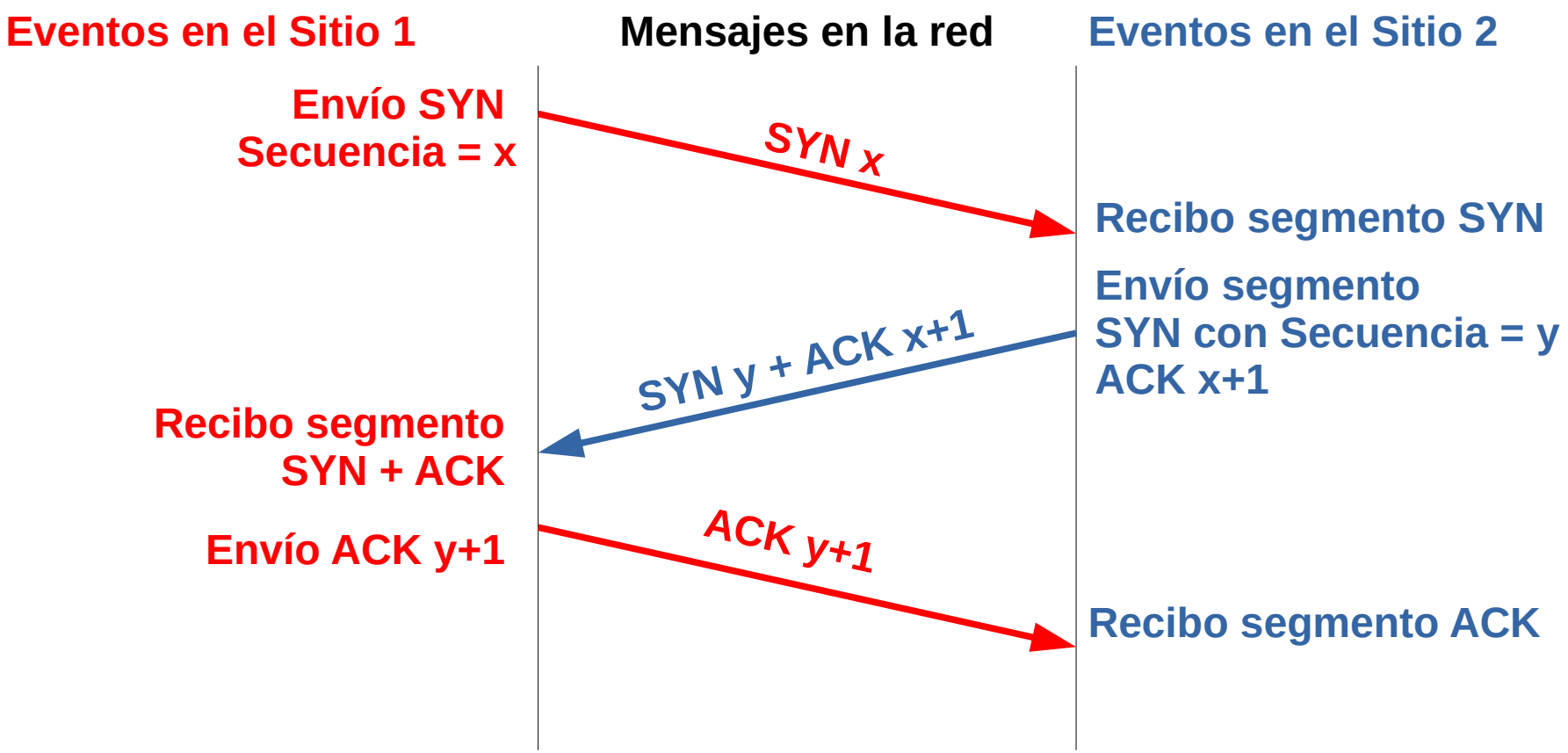
- Al inicio de la transmisión o luego de una congestión se comienza transmitiendo un MSS
- Se incrementa la “ventana de congestión” en un MSS por cada ACK que se reciba sin necesidad de retransmisión
- Se sigue aumentando hasta que la “ventana de congestión” llega a un valor umbral, a partir de ahí se incrementa mas lentamente: un MSS cada vez que se reciben los ACK de una ventana completa sin necesidad de retransmitir.
- Se prosigue hasta que

  “ventana de congestión” = “aviso de ventana”

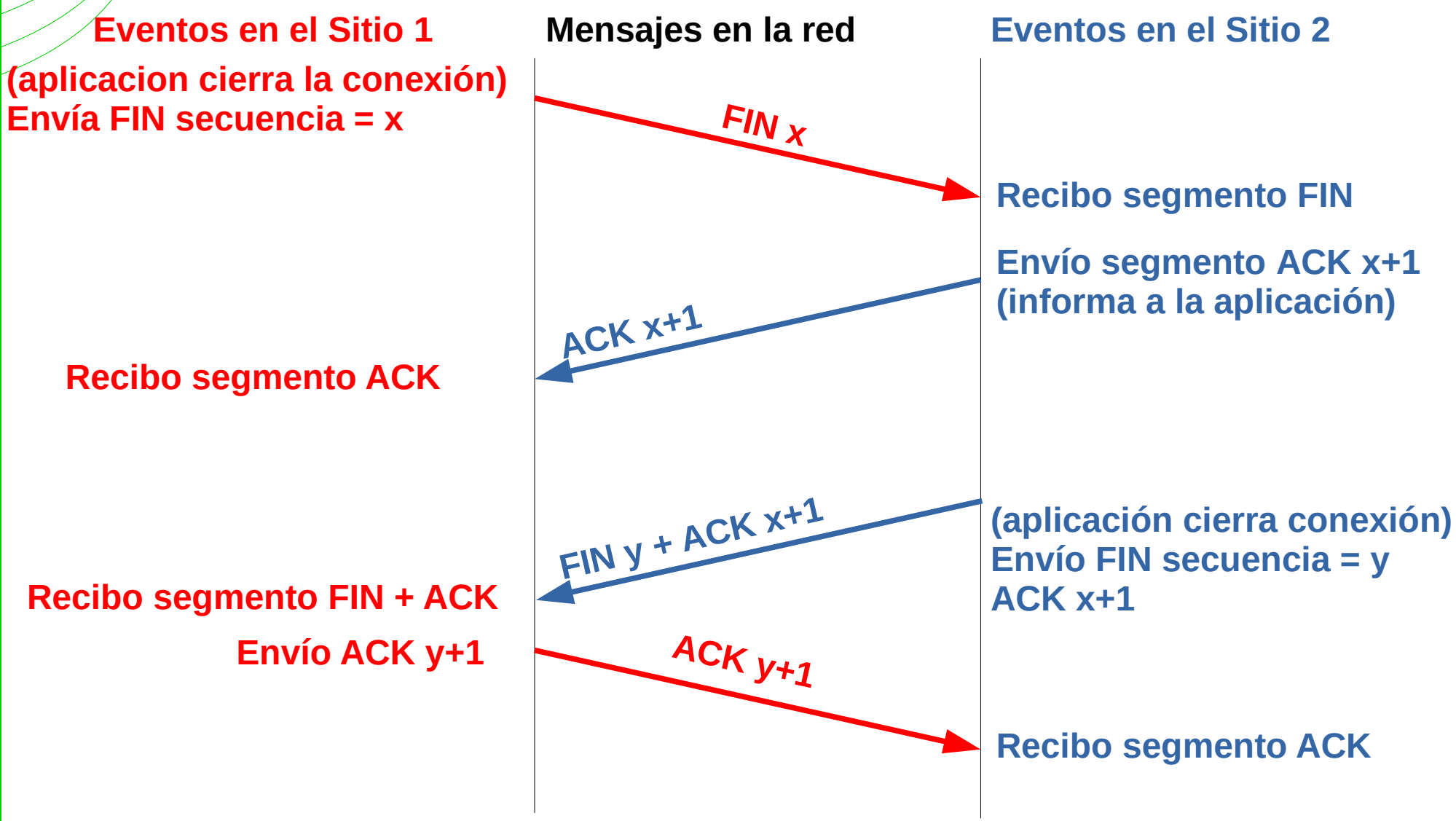
Algoritmos de Control de Congestión

- El control de congestión descripto es básico.
- No hace falta que transmisor y receptor acuerden como realizarlo.
- El transmisor puede implementar otros algoritmos de control. (existen más de 10).
- Siguen actualizandose para adecuarse a las distintas tecnologías de red.

Inicio en tres pasos



Cierre de conexión



Datos Urgentes

- En el flujo de datos normal se pueden insertar “Datos Urgentes”
- Cuando el segmento se marca como URGENTE
  - El primer byte del segmento corresponde al primer byte de los datos urgentes.
  - El campo Urgent Pointer de la cabecera TCP contiene un puntero al último byte de los datos urgentes.
- Puede ser retransmitido
- El protocolo envía los datos urgentes a la aplicación apenas los recibe.

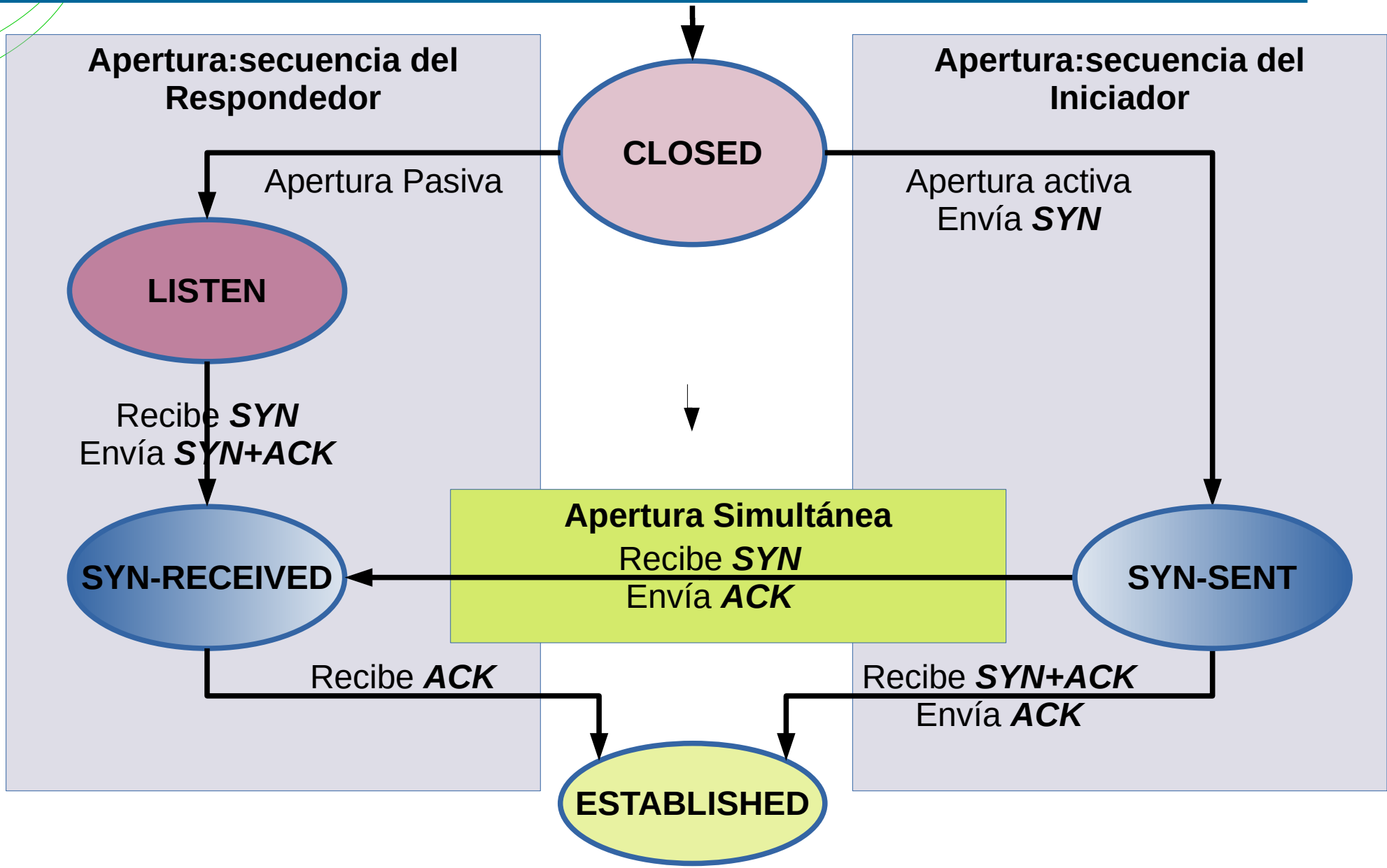
Máquina de Estados de TCP

- El funcionamiento (inicio, terminación, manejo de errores, manejo de datos urgentes, etc) de una conexión TCP puede describirse con una Máquina de Estados Finitos (FSM).
- Los eventos que provocan transiciones de estados consisten en la recepción de segmentos con ciertos Flags establecidos.
- En cualquier segmento, se pueden establecer uno o más Flags, esto permite:
  - Utilizar un mismo segmento para enviar datos y a su vez reconocer (ACK) la recepción de otros
  - Enviar un número de secuencia con SYN y a su vez reconocer la recepción de otra con ACK.
  - Si en un segmento se establece más de un flag se denota con “+” por ejemplo **SYN + ACK**

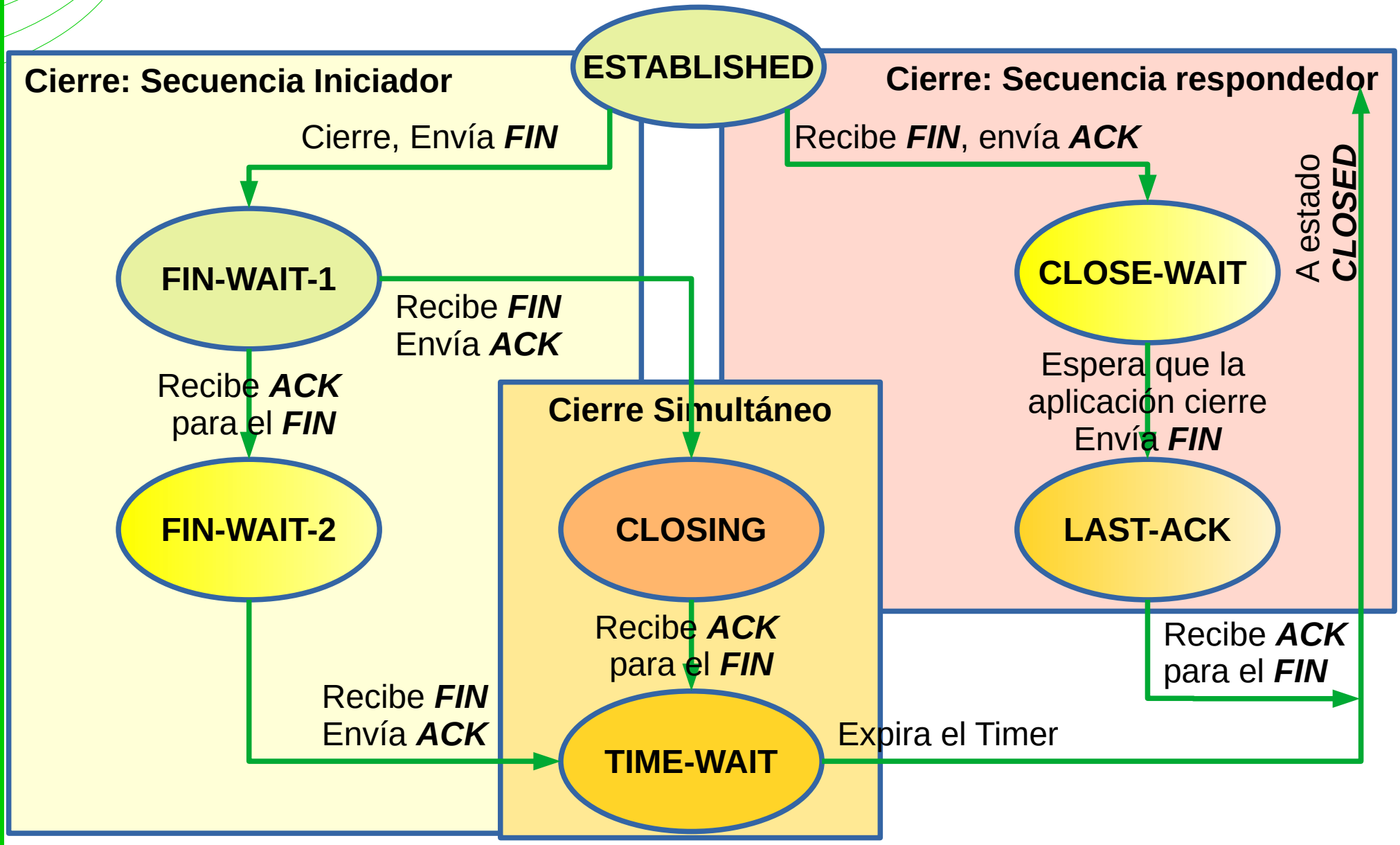
Máquina de Estados de TCP

- Describe las transiciones de Estados durante una conexión TCP (Simplificada)
  - Solo Inicio y terminación de la conexión
  - No se describen
    - Manejo de errores
    - Datos urgentes
- Debe distinguirse entre quien inicia la conexión (Iniciador) y quien actua como servidor (Respondedor)

Máquina de Estados Finitos TCP – Conexión



Máquina de Estados Finitos TCP – Desconexión



Comparación entre UDP y TCP

UDP	TCP
Intercambia paquetes llamados ***datagramas*** entre aplicaciones e IP	Intercambia paquetes llamados ***segmentos*** entre aplicaciones e IP
No confiable	Confiable
Checksum opcional	Checksum obligatorio
Sin conexión	Orientado a conexión
Datos como registros	Orientado a flujos de datps
Más adecuada para LAN	Útil para LAN y WAN
Sin control de flujo	Control de flujo
1 a 1, 1 a muchos, o muchos a 1	1 a 1
Permite unicast, multicast o broadcast	Sólo Unicast

Preguntas