

Módulo 04

La Capa de Red

(Pt. 1)



Redes de Computadoras
Depto. de Cs. e Ing. de la Comp.
Universidad Nacional del Sur



Copyright

- Copyright © 2010-2024 A. G. Stankevicius
- Se asegura la libertad para copiar, distribuir y modificar este documento de acuerdo a los términos de la **GNU Free Documentation License**, versión 1.2 o cualquiera posterior publicada por la Free Software Foundation, sin secciones invariantes ni textos de cubierta delantera o trasera
- Una copia de esta licencia está siempre disponible en la página <http://www.gnu.org/copyleft/fdl.html>
- La versión transparente de este documento puede ser obtenida de la siguiente dirección:

<http://cs.uns.edu.ar/~ags/teaching>

Contenidos

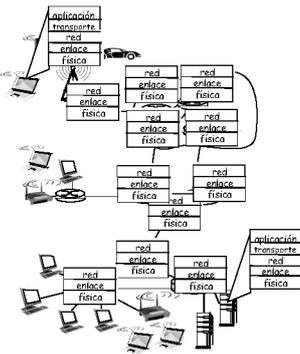
- Modelos de servicios de la capa de red
- Estructura interna de un router
- El protocolo **IP**
- **IPv4 vs. IPv6**
- Protocolos de ruteo
- Ruteo jerárquico
- Ruteo en internet
- Multicast

ISO/OSI - TCP/IP



Funciones de la capa de red

- La capa de red encauza los paquetes entregados por la capa de transporte a través de las distintas computadoras de la red
- La capa de red está implementada en todos los nodos de la red:
 - Computadoras
 - Routers



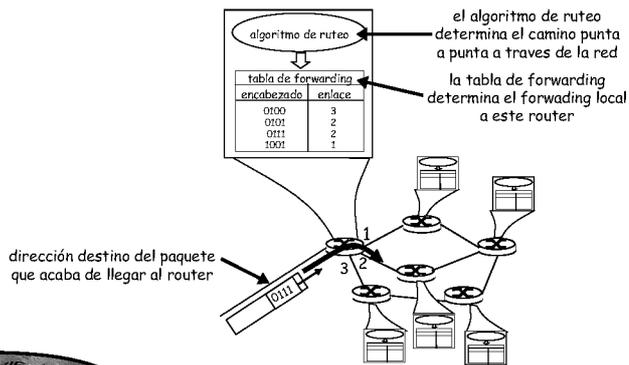
Funciones de la capa de red

- La capa de red debe cumplir dos funciones primordiales:
 - Debe determinar el camino que han de tomar los paquetes al ser conducidos desde el origen hasta el destino haciendo uso de los algoritmos de ruteo
 - Tiene que resolver correctamente el forwarding de los paquetes en cada router visitado, esto es, en cada router de la red se debe decidir por qué enlace ha de salir cada uno de los paquetes que van arribando

Otra "car analogy"

- Haciendo uso nuevamente de una analogía que involucre autos, supongamos que se quiere ir desde el campus de la **UNS** hasta el aeropuerto:
 - El ruteo se corresponde con elegir la ruta hasta el aeropuerto, por caso, salir por la Av. Cabrera hasta la rotonda del ChangoMás, de ahí tomar la Av. Newbery hasta la rotonda de Indiada y finalmente de ahí encarar hacia el aeropuerto
 - El forwarding se corresponde con la decisión en cada una de las rotondas que vamos atravesando acerca de por cuál de sus salidas seguir camino

Interacción ruteo-forwarding



Establecimiento de conexiones

- En algunas configuraciones puede hacer falta una fase de inicialización previa a comenzar con el enrutado de los paquetes
- Tanto las dos computadoras en los extremos de la conexión como los routers entre las mismas participan estableciendo una conexión virtual
- No confundir con la conexiones que se gestionan a nivel de la capa de transporte
 - ¿Por qué razón? ¿Cuál sería la principal diferencia?

¿Cómo es bien lo de ATM?

Tip: Recuerde que Tercer los diferentes modelos de servicios de la tecnología de red ATM.

OutGP: ATM (Asynchronous Transfer Mode) es una tecnología de red que ofrece varios modelos de servicios para satisfacer diferentes necesidades de comunicación. Aquí le damos un resumen de los principales modelos de servicios de ATM:

- Servicio de Datos Constante (CB - Constant Bit Rate)**
 - Proporciona un tipo de datos constante y predecible.
 - Adecuado para aplicaciones en tiempo real, como voz y video.
 - Se garantiza el ancho de banda para cada conexión.
- Servicio de Datos Asegurado (ABR - Available Bit Rate)**
 - Proporciona un ancho de banda variable basado en la disponibilidad de recursos de red.
 - Adecuado para aplicaciones con requisitos de ancho de banda fluctuantes.
 - Mantiene un uso eficiente de los recursos de red al adaptarse dinámicamente a las condiciones de tráfico.
- Servicio de Datos No Especificado (UBR - Unspecified Bit Rate)**
 - Ofrece un ancho de banda no garantizado y no reservado.
 - Adecuado para aplicaciones que pueden tolerar retrasos y pérdida de paquetes, como transferencia de archivos y correo electrónico.
 - Utiliza el ancho de banda disponible de manera no controlada.
- Servicio de Datos Asegurado (GFR - Guaranteed Bit Rate)**
 - Proporciona una cantidad de servicio (GFR) garantizada y predecible.
 - Adecuado para aplicaciones sensibles al retraso, como transferencia de paquetes, como videoconferencia y aplicaciones de tiempo real.
 - Utiliza mecanismos de reserva de recursos y priorización de tráfico para garantizar el rendimiento.
- Servicio de Datos sin conexión (CL - Connectionless)**
 - No requiere establecimiento de conexión antes de la transmisión de datos.
 - Adecuado para aplicaciones de comunicación en difusión, como streaming y video de área local (LAN).
 - Proporciona un servicio similar al modo de operación de la capa de red en el modelo OSI.

Existen otros algunos de los modelos de servicios más comunes en la tecnología de red ATM. Cada modelo ofrece diferentes características y beneficios para satisfacer las diversas necesidades de comunicación de las aplicaciones. La elección del modelo adecuado depende de los requisitos específicos de cada aplicación en términos de ancho de banda, retraso, pérdida de paquetes y control de congestión.

¿Con o sin conexión?

- Las redes de datagramas ofrecen una capa de red no orientada a la conexión
- En contraste, las basadas en circuitos virtuales ofrecen una orientada a la conexión
- Se asemeja a la discusión **TCP vs. UDP**, pero con las siguientes salvedades:
 - El servicio es computadora a computadora
 - No es electivo, la red provee o bien uno o el otro
 - La implementación reside en el núcleo de la red

Circuitos virtuales

- Uno de los modelos de servicio posibles se basa en hacer uso de circuitos virtuales
- Hacer uso de **CV** implica que la capa de red brindará una conexión análoga a la que se obtiene al usar un circuito telefónico
 - Se debe inicializar la llamada antes de comenzar a transmitir información
 - Cada celda de datos consigna información del **CV** al que pertenece y no del destino deseado
 - Los routers mantienen información de los **CV**

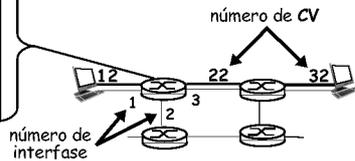
Implementación

- A nivel de implementación, un circuito virtual consiste de los siguientes elementos:
 - Un camino desde el origen hasta el destino
 - Un conjunto de números de circuito virtual, uno para cada enlace a lo largo del camino
 - Una entrada en la tabla de forwarding de los routers a lo largo del camino
- Los paquetes que pertenecen a un dado circuito virtual se identifican a través de su número
 - El número de circuito puede cambiar en cada enlace

Implementación

- El no utilizar el mismo número de circuito virtual de punta a punta hace que los routers tengan que almacenar más información
 - ¿Por qué razón se optó por no utilizar el mismo número de CV de punta a punta?

Ent.	CV	Sal.	CV
1	12	3	22
2	63	1	18
3	7	2	17
1	97	3	87



¡Este año mejoró y mucho!

You
Por qué razón ATM no mantiene el mismo número de circuito virtual de origen a destino?

ChatGPT
ATM (Modo de Transferencia Asíncrona) utiliza circuitos virtuales (CV) para establecer conexiones entre nodos en una red. Sin embargo, a diferencia de algunas otras tecnologías de conmutación de circuitos, como Frame Relay, ATM no mantiene necesariamente el mismo número de circuito virtual de origen a destino. Hay varias razones por las cuales esto puede ocurrir:

- 1. Conmutación de celdas:** ATM conmuta celdas individuales en lugar de tramas completas, lo que significa que una única conexión puede compartir recursos de red y, por lo tanto, puede ser transmitida a través de diferentes rutas y circuitos virtuales en la red. Esto permite una mejor utilización de los recursos de red y una mayor eficiencia en el transporte de datos.
- 2. Conmutación de rutas dinámica:** En ATM, la selección de ruta y la conmutación de celdas pueden ser dinámicas y dependen de factores como la congestión de la red, la calidad de servicio (QoS) requerida y las condiciones de tráfico. Esto significa que una conexión puede cambiar de circuito virtual durante su transmisión si se determina que otra ruta ofrece un mejor rendimiento.
- 3. Conexiones multipunto:** En algunas configuraciones de red ATM, una conexión puede tener múltiples puntos de salida o destino, lo que significa que puede ser entregada a través de diferentes circuitos virtuales según la topología de la red y la disponibilidad de recursos.
- 4. Flexibilidad y escalabilidad:** Permitir que una conexión utilice diferentes circuitos virtuales en diferentes partes de la red proporciona flexibilidad y escalabilidad, lo que permite adaptarse a cambios en la topología de la red, la carga de tráfico y los requisitos de calidad de servicio.

En resumen, ATM no mantiene necesariamente el mismo número de circuito virtual de origen a destino debido a su naturaleza de conmutación de celdas, la conmutación de rutas dinámica, las conexiones multipunto y la necesidad de flexibilidad y escalabilidad en las redes. Esto permite una mejor utilización de los recursos de red y una entrega eficiente de datos en entornos de red ATM.

Protocolo de señalización

- El protocolo de señalización es el encargado de establecer y liberar los CV
- Por caso, es utilizado en **ATM, Frame Relay, X.25**

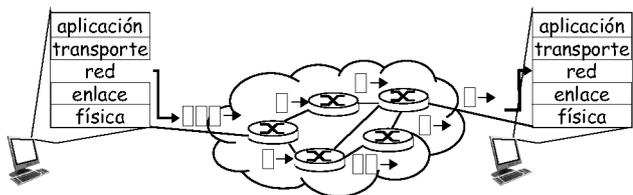


Datagramas

- El otro modelo de servicio posible se basa en la utilización de datagramas
- No requiere de una inicialización de la conexión antes de comenzar a transmitir información
- Los routers no cuentan con información acerca de las conexiones punta-a-punta (esto es, a nivel de capa de red no existe el concepto de conexión)
- Los paquetes son ruteados inspeccionando la información contenida en cada datagrama acerca del destino hacia dónde se dirige

Modelo de internet

- A diferencia del modelo de servicio basado en circuitos virtuales, los datagramas de una misma conexión pueden no transitar el mismo camino



Implementación

- La tabla de forwarding de los routers no puede contar con información acerca de qué enlace tomar para cada una de las 2^{32} direcciones IP

Rango de Direcciones Destino	Enlace de Salida
desde 11001000 00010111 00010000 00000000 hasta 11001000 00010111 00010111 11111111	1
desde 11001000 00010111 00011000 00000000 hasta 11001000 00010111 00011000 11111111	2
desde 11001000 00010111 00011001 00000000 hasta 11001000 00010111 00011111 11111111	3
caso contrario	4

Implementación

- En el mundo real, los rangos de la tabla de forwarding no suelen quedar tan prolijos
 - La tabla se consulta buscando cuál es el prefijo de mayor longitud con el cual coincide la dirección destino del datagrama

Rango de Direcciones Destino	Enlace de Salida
11001000 00010111 00010*** *****	1
11001000 00010111 00011000 *****	2
11001000 00010111 00011*** *****	3
caso contrario	4

Datagramas vs. CV

- En internet:
 - Se intercambian datos entre computadoras, por lo que usualmente no hay requerimientos tan estrictos de temporizado
 - Los sistemas en la frontera de la red son inteligentes, se adaptan a los cambios, pueden implementar control de errores, etc.
 - La complejidad está en la frontera, no en el núcleo
 - Está compuesta de muchos tipos distintos de enlaces, por lo que brindar un servicio homogéneo no es necesariamente trivial

Datagramas vs. CV

En ATM:

- Evolucionó a partir del sistema telefónico
- Se diseñó para entablar conversaciones entre humanos, por lo que se deben poder asegurar estrictos niveles de retardo y de confiabilidad
- Es necesario que la red asegure una determinada calidad de servicio
- Los sistemas en la frontera de la red carecen de inteligencia (por caso, un teléfono fijo o un fax)
- La complejidad se aloja en el núcleo de la red

Pulgar arriba estimado bot...

¿No?
Una vez, me comparas conmutar datagramas por un lado con el uso de circuitos virtuales por el otro?

¡Claro!
Por supuesto, puedo comparar la conmutación de datagramas y el uso de circuitos virtuales en el contexto de las redes de comunicaciones:

1. Conmutación de Datagramas:

- En la conmutación de datagramas, los datos se transmiten como unidades independientes llamadas datagramas.
- Cada datagrama contiene la dirección de origen y la dirección de destino, lo que permite que se envíe de manera independiente a través de la red.
- Cada datagrama se transmite de acuerdo con la tabla de enrutamiento de la red, que se actualiza dinámicamente para reflejar las condiciones de la red y los cambios en la topología.
- No se establece una ruta predeterminada para los datagramas antes de su transmisión, lo que significa que pueden seguir diferentes rutas y experimentar diferentes retardos en la red.

2. Circuitos Virtuales:

- En el uso de circuitos virtuales, se establece una ruta predeterminada antes de la transmisión de datos entre los nodos de la red.
- Antes de la transmisión de datos, se establece un circuito virtual entre el origen y el destino, que incluye varios enlaces y nodos intermedios en la red.
- Una vez establecido, todos los datos transmitidos entre el origen y el destino siguen la misma ruta a través de la red, evitando el circuito virtual previamente configurado.
- Los circuitos virtuales pueden ser de dos tipos: conmutación de circuitos virtuales permanentes (PVC) y conmutación de circuitos virtuales conmutados (SVC). Los PVCs tienen una ruta establecida de antemano y son utilizados de manera constante, mientras que los SVCs se establecen dinámicamente según la demanda y finalizan cuando ya no se necesitan.

Comparación:

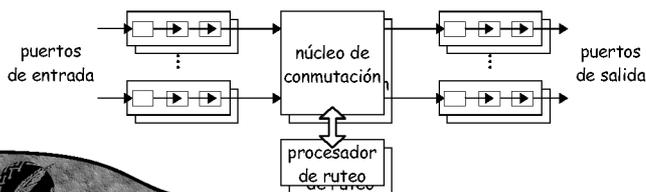
- La conmutación de datagramas es más adecuada para redes donde la capacidad y la carga de tráfico son variables, ya que cada paquete se envía de manera independiente y puede seguir diferentes rutas.
- Los circuitos virtuales son más adecuados para aplicaciones que requieren una comunicación predecible y constante, ya que una vez establecido, el circuito virtual garantiza una ruta fija y dedicada para todos los datos transmitidos entre el origen y el destino.
- La conmutación de datagramas es más eficiente en términos de uso de recursos de red, ya que no se requiere asignación de recursos antes de la transmisión de datos. Sin embargo, puede llevar a una mayor variabilidad en los tiempos de entrega y la calidad de servicio.
- Los circuitos virtuales son menos eficientes en términos de uso de recursos, ya que requieren la asignación de recursos antes de la transmisión de datos. Sin embargo, proporcionan una comunicación más predecible y constante.

En resumen, la elección entre la conmutación de datagramas y el uso de circuitos virtuales depende de los requisitos específicos de la aplicación, incluida la predictibilidad de la comunicación, la variabilidad de la carga de tráfico y la eficiencia en el uso de los recursos de red.

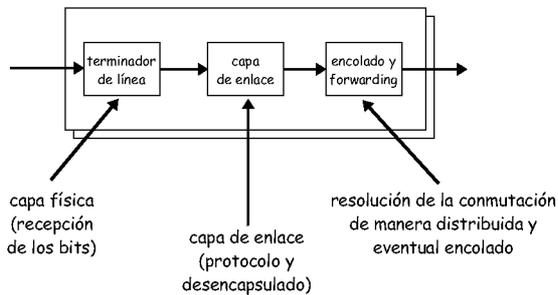
Estructura de un router

Si dirigimos nuestra atención a qué hay dentro de un router, se identifican dos grandes tareas:

- Debe ejecutar y cumplir con el protocolo de ruteo elegido (puede ser más de uno)
- Debe llevar adelante el rol central del router, esto es, resolver el forwarding de los datagramas recibidos



Puertos de entrada

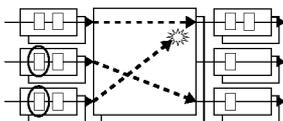


Conmutación descentralizada

- La multiplicidad de puertos de entrada permite descentralizar la toma de decisión respecto al forwarding de datagramas:
 - Para un dado datagrama, se consulta en la tabla de forwarding qué puerto de salida tomar
 - Idealmente se desea poder tomar la decisión a la misma velocidad que llegan los nuevos datagramas
 - En caso contrario, se producirá el encolado en el propio puerto de entrada

Encolado

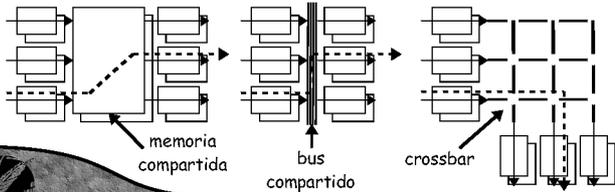
- El encolado se produce toda vez que el núcleo de conmutación no alcanza a distribuir a tiempo el tráfico recibido por los puertos de entrada
 - El datagrama en la primer posición de la cola puede bloquear a los datagramas que vienen detrás
 - Si el buffer se va saturando se producirán demoras y eventualmente pérdidas de datagramas



Núcleo de conmutación

● En la actualidad existen varias alternativas a la hora de implementar el núcleo de conmutación:

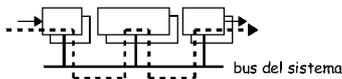
- Hacer uso de una memoria compartida
- Organizarlo en torno a un bus compartido
- Organizarlo en torno a un crossbar



Memoria compartida

● Routers de primera generación (hoy legacy):

- Un único **CPU** se encarga de copiar los datagramas de los puertos de entrada a los puertos de salida
- La velocidad de conmutación está limitada por el ancho de banda de la memoria (pues se requieren dos accesos para conmutar cada datagrama)



● Los puertos de entrada de los más modernos son capaces acceder directamente a memoria

Bus compartido

● Otra alternativa a la hora de diseñar un router es organizarlo en torno a un bus compartido:

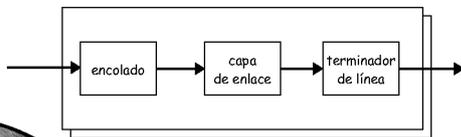
- Los datagramas se trasladan del buffer de entrada al buffer de salida haciendo uso de un bus
- La contención en el acceso al bus aumenta el tiempo de encolado en los distintos puertos de entrada
- Por ende, la velocidad de conmutación estará limitada por el ancho de banda del bus
- Adecuado incluso hoy en día para routers hogareños y también tipo **PyME**

Crossbar

- El crossbar surge como una extensión natural del bus compartido
 - Su principal objetivo es aliviar el problema de la contención en el acceso al medio compartido
 - Esta tecnología se empezó a utilizar en las viejas supercomputadoras para interconectar de manera eficiente a sus múltiples procesadores
- Las versiones más modernas fragmentan los datagramas en celdas de tamaño fijo, las que son conmutadas a través del núcleo

Puertos de salida

- Los puertos de salida son en cierta forma la imagen especular de los puertos de entrada
 - Si la cantidad de datagramas conmutados supera la capacidad de transmisión del puerto de salida se apela al encolado o al descarte de datagramas
 - La disciplina de despacho (scheduling discipline) dictamina qué datagrama transmitir a continuación



Pérdida de datagramas

- El nivel de ocupación de los distintos buffers afectan directamente el desempeño del router
 - Un mayor nivel de ocupación implica un mayor retardo de encolado
- Sólo existen dos puntos en los que se puede tener que descartar un datagrama válido:
 - Al llegar un nuevo datagrama a un puerto de entrada que tenga su buffer lleno
 - Al conmutar un nuevo datagrama hacia un puerto de salida que tenga su buffer lleno

Requerimientos de buffering

- Resulta crucial determinar la cantidad óptima de almacenamiento intermedio del router:
 - Un buffer insuficiente conduce a pérdidas
 - Un buffer excesivo incrementa el costo de router
- La **RFC 3439** especifica como regla del pulgar multiplicar el **RTT** promedio de la red por la capacidad del enlace
 - Por ejemplo, asumido un **RTT** de 250ms necesitamos 2.5Gbits de buffer para un enlace de 10Gb/s
 - Actualmente se está revisando esta recomendación

¿Preguntas?