



INSTITUTO POLITÉCNICO SUPERIOR

ADMINISTRACIÓN DE REDES LOCALES

CAPA DE ENLACE DE DATOS

Rodríguez Costello, Alejandro
Córdoba, Agustín Leonel

Índice

1. Aspectos de diseño	2
1.1. Entramado y corrección de errores	3
2. Transmisión confiable: protocolos elementales	5
2.1. Protocolo simplex	5
2.2. Procolo simplex de parada y espera	6
2.3. Protocolo simplex de parada y espera en canal ruidoso	7
3. Transmisión confiable: protocolos de ventana deslizante	10
3.1. Ventana deslizante de un bit	11
3.2. Ventana deslizante con retroceso N	12
3.3. Ventana deslizante con repetición selectiva	13
4. Anexo: Introducción a Packet Tracer	15
4.1. Entorno de trabajo	15
4.2. Creación de topología de red	16

1. Aspectos de diseño

La capa de enlace de datos utiliza los servicios de la capa física para enviar y recibir bits a través de los canales de comunicación. Tiene varias funciones específicas, entre las que se incluyen:

- Proporcionar a la capa de red una interfaz de servicio bien definida.
- Manejar los errores de transmisión.
- Regular el flujo de datos para que los emisores rápidos no saturen a los receptores lentos.

Para cumplir con estas metas, la capa de enlace de datos toma los paquetes que obtiene de la capa de red y los encapsula en tramas para transmitirlos. Cada trama contiene un encabezado, un campo de carga útil (payload) para almacenar el paquete y un terminador, como se muestra en la figura de ejemplo a continuación. El manejo de las tramas es la tarea más importante de la capa de enlace de datos.

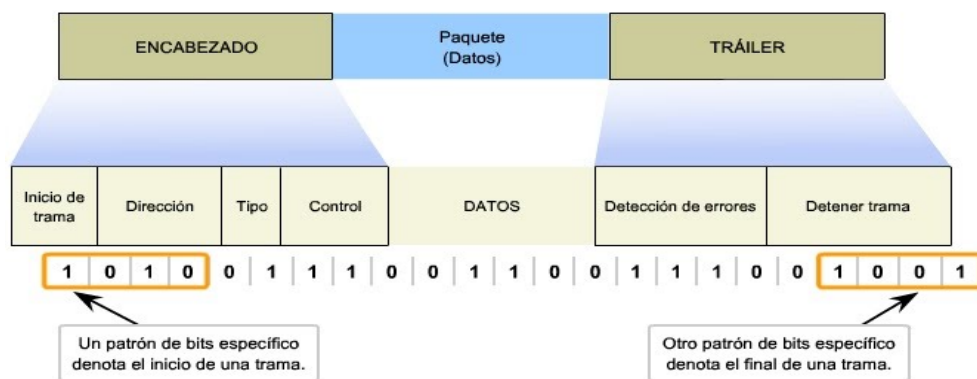


Figura 1: Conformación de trama en capa 2

La capa de enlace de datos puede diseñarse para ofrecer varios servicios. Los servicios reales ofrecidos varían de un protocolo a otro. Tres posibilidades razonables que normalmente se proporcionan son:

1. Servicio sin conexión ni confirmación de recepción.
2. Servicio sin conexión con confirmación de recepción.
3. Servicio orientado a conexión con confirmación de recepción.

El servicio sin conexión ni confirmación de recepción consiste en hacer que la máquina de origen envíe tramas independientes a la máquina de destino sin que ésta confirme la recepción. Ethernet es un buen ejemplo de una capa de enlace de datos que provee esta clase de servicio. No se establece una conexión lógica de antemano ni se libera después. Si se pierde una trama debido a ruido en la línea, en la capa de

datos no se realiza ningún intento por detectar la pérdida o recuperarse de ella. Esta clase de servicio es apropiada cuando la tasa de error es muy baja, de modo que la recuperación se deja a las capas superiores.

El siguiente paso en términos de confiabilidad es el servicio sin conexión con confirmación de recepción. Cuando se ofrece este servicio tampoco se utilizan conexiones lógicas, pero se confirma de manera individual la recepción de cada trama enviada. De esta manera, el emisor sabe si la trama llegó bien o se perdió. Si no ha llegado en un intervalo especificado, se puede enviar de nuevo. Este servicio es útil en canales no confiables.

Finalmente, el servicio más sofisticado que puede proveer la capa de enlace de datos a la capa de red es el servicio orientado a conexión. Con este servicio, las máquinas de origen y de destino establecen una conexión antes de transferir datos. Cada trama enviada a través de la conexión está numerada, y la capa de enlace de datos garantiza que cada trama enviada llegará a su destino. Es más, garantiza que cada trama se recibirá exactamente una vez y que todas las tramas se recibirán en el orden correcto. Así, el servicio orientado a conexión ofrece a los procesos de la capa de red el equivalente a un flujo de bits confiable.

1.1. Entramado y corrección de errores

Para proveer servicio a la capa de red, la capa de enlace de datos debe usar el servicio que la capa física le proporciona. Lo que hace la capa física es aceptar un flujo de bits puros y tratar de entregarlo al destino. Si el canal es ruidoso, como en la mayoría de los enlaces inalámbricos y en algunos alámbricos, la capa física agregará cierta redundancia a sus señales para reducir la tasa de error de bits a un nivel tolerable. Sin embargo, no se garantiza que el flujo de bits recibido por la capa de enlace de datos esté libre de errores. Algunos bits pueden tener distintos valores y la cantidad de bits recibidos puede ser menor, igual o mayor que la cantidad de bits transmitidos. Es responsabilidad de la capa de enlace de datos detectar y, de ser necesario, corregir los errores.

Es más difícil dividir el flujo de bits en tramas de lo que parece a simple vista. Un buen diseño debe facilitar a un receptor el proceso de encontrar el inicio de las nuevas tramas al tiempo que utiliza una pequeña parte del ancho de banda del canal. Para ellos existen varios métodos. A continuación se desarrollará uno a modo de ejemplo para clarificar la operatoria.

El Conteo de Bytes se vale de un campo en el encabezado para especificar el número de bytes en la trama. Cuando la capa de enlace de datos del destino ve el conteo de bytes, sabe cuántos bytes siguen y, por lo tanto, dónde concluye la trama. Esta

técnica se muestra en la figura a continuación para cuatro tramas pequeñas de ejemplo con 5, 5, 8 y 8 bytes de longitud, respectivamente. Uno de los problemas con este algoritmo es que el conteo se puede alterar debido a un error de transmisión.

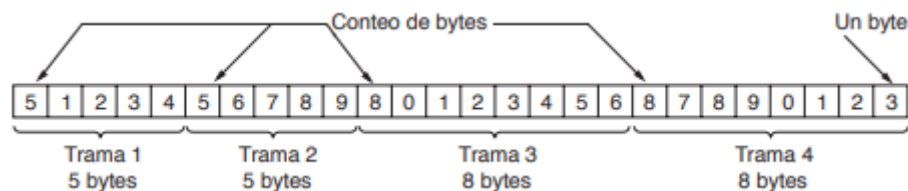


Figura 2: Flujo de bytes con conteo

Una vez resuelto el problema de marcar el inicio y el fin de cada trama, se llega al siguiente dilema: cómo asegurar que todas las tramas realmente se entreguen en el orden apropiado a la capa de red del destino (suponiendo por ahora que el receptor puede saber si una trama que recibe contiene información correcta o errónea). La manera normal de asegurar la entrega confiable de datos es proporcionar retroalimentación al emisor sobre lo que está ocurriendo en el otro lado de la línea. Por lo general, el protocolo exige que el receptor devuelva tramas de control especiales que contengan confirmaciones de recepción positivas o negativas de las tramas que llegan. Si el emisor recibe una confirmación de recepción positiva de una trama, sabe que la trama llegó de manera correcta. Por otra parte, una confirmación de recepción negativa significa que algo falló y que se debe transmitir la trama otra vez.

Una complicación adicional surge de la posibilidad de que los problemas de hardware causen la desaparición de una trama completa (por ejemplo, por una ráfaga de ruido). En este caso, el receptor no reaccionará en absoluto, ya que no tiene razón para reaccionar. De manera similar, si se pierde la trama de confirmación de recepción, el emisor no sabrá cómo proceder.

Es por todo ello que se han desarrollado dos estrategias para manejar los errores. Ambas añaden información redundante a los datos que se envían. Una es incluir suficiente información redundante para que el receptor pueda deducir cuáles debieron ser los datos transmitidos. La otra estrategia es incluir sólo suficiente redundancia para permitir que el receptor sepa que ha ocurrido un error (pero no qué error) y entonces solicite una retransmisión. La primera estrategia utiliza códigos de corrección de errores; la segunda usa códigos de detección de errores.

2. Transmisión confiable: protocolos elementales

Para comenzar, vamos a suponer que las capas física, de enlace de datos y de red son procesos independientes que se comunican pasando mensajes de un lado a otro. Para ello, el proceso de la capa física y una parte del proceso de la capa de enlace se ejecutan en hardware dedicado, conocido como NIC (Tarjeta de Interfaz de Red), y el resto del proceso de la capa de enlace y el proceso de la capa de red se ejecutan en la CPU principal como parte del sistema operativo, en donde el software para el proceso de la capa de enlace a menudo toma la forma de un controlador de dispositivo.

Además, otro supuesto que haremos es que la máquina A desea mandar un flujo considerable de datos a la máquina B mediante el uso de un servicio confiable orientado a conexión. Asimismo, A tiene un suministro infinito de datos listos para ser enviados y nunca tiene que esperar a que éstos se produzcan, sino que cuando la capa de enlace de datos de A los solicita, la capa de red siempre es capaz de proporcionarlos de inmediato.

Cuando la capa de enlace de datos acepta un paquete, lo encapsula en una trama agregándole un encabezado y un terminador de enlace de datos. Por lo tanto, una trama consiste en un paquete incrustado, con cierta información de control (en el encabezado) y una suma de verificación (en el terminador), como hemos visto en la sección anterior. A continuación la trama se transmite a la capa de enlace de datos de la otra máquina. En un principio, el receptor no tiene nada que hacer. Sólo está esperando a que ocurra algo.

Cabe destacar además, que el canal no siempre es confiable y en ocasiones pierde tramas completas. Para poder recuperarse, la capa de enlace de datos emisora debe iniciar un temporizador o reloj interno cada vez que envía una trama. Si no obtiene respuesta después de cierto intervalo predeterminado, el reloj expira y la capa de enlace de datos recibe una señal de interrupción. Dicho todo esto, pasaremos a estudiar los protocolos.

2.1. Protocolo simplex

Como ejemplo inicial se considerará un protocolo que es lo más sencillo posible, por la posibilidad de que algo salga mal. Los datos son transmitidos en una sola dirección; las capas de red tanto del emisor como del receptor siempre están listas. Hay un espacio infinito de búfer disponible. Y lo mejor de todo, el canal de comunicación entre las capas de enlace de datos nunca daña ni pierde las tramas. Este protocolo es completamente irreal y es simplemente para mostrar la estructura básica de funcionamiento.

El emisor está en un ciclo *while* infinito que sólo envía datos a la línea tan rápido como puede. El cuerpo del ciclo consiste en tres acciones: obtener un paquete de la (siempre dispuesta) capa de red, construir una trama de salida y enviar la trama a su destino. El receptor también es sencillo. Al principio espera que algo ocurra, siendo la única posibilidad la llegada de una trama sin daños. En algún momento llega la trama y la parte de los datos se pasa a la capa de red y la capa de enlace de datos se retira para esperar la siguiente trama.

Este protocolo es irreal ya que no maneja el control de flujo ni la corrección de errores. Su procesamiento se asemeja al de un servicio sin conexión ni confirmación de recepción que depende de las capas más altas para resolver estos problemas.

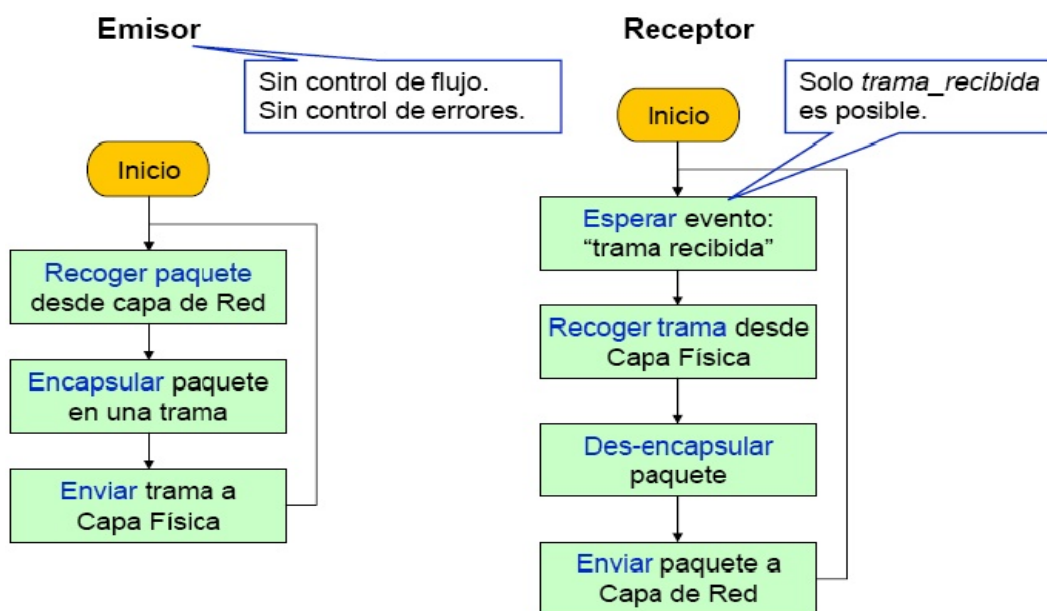


Figura 3: Protocolo simplex sin restricciones

2.2. Procolo simplex de parada y espera

Ahora deberemos lidiar con el problema principal de evitar que el emisor sature al receptor enviando tramas a una mayor velocidad de la que este último puede procesarlas. Esta situación puede ocurrir con facilidad en la práctica, por lo que es de extrema importancia evitarla. Sin embargo, aún existe el supuesto de que el canal está libre de errores y el tráfico de datos sigue siendo simplex.

Una solución general es hacer que el receptor proporcione retroalimentación al emisor. Tras haber pasado un paquete a su capa de red, el receptor regresa al emisor una pequeña trama ficticia que, de hecho, autoriza al emisor para que transmita la

siguiente trama. Después de enviar una trama, el protocolo exige que el emisor espere hasta que llegue la pequeña trama ficticia (es decir, la confirmación de recepción). Este retraso es un ejemplo simple de un protocolo de control de flujo. Los protocolos de este tipo se denominan de parada y espera.

Al igual que en el protocolo simplex, el emisor comienza obteniendo un paquete de la capa de red, lo usa para construir una trama y enviarla a su destino. Sólo que ahora, el emisor debe esperar hasta que llegue una trama de confirmación de recepción antes de reiniciar el ciclo y obtener el siguiente paquete de la capa de red. La capa de enlace de datos emisora ni siquiera necesita inspeccionar la trama entrante, ya que sólo hay una posibilidad. La trama entrante siempre es de confirmación de recepción y se la denomina ACK.

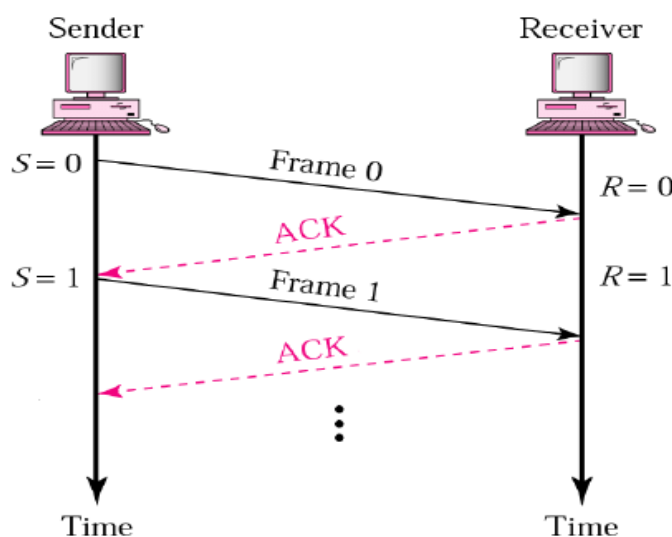


Figura 4: Protocolo simplex de parada y espera

2.3. Protocolo simplex de parada y espera en canal ruidoso

Finalmente, consideremos la situación normal de un canal de comunicación que comete errores. Las tramas pueden llegar dañadas o se pueden perder por completo. Sin embargo, suponemos que si una trama se daña en tránsito, el hardware del receptor detectará esto cuando calcule la suma de verificación.

A primera vista puede parecer que funcionaría una variación del protocolo simplex de parada y espera: agregar un temporizador. El emisor podría enviar una trama, pero el receptor sólo enviaría una trama de confirmación de recepción si los datos llegaran correctamente. Si llegara una trama dañada al receptor, se desecharía. Después de un tiempo el temporizador del emisor expiraría y éste enviaría la trama otra vez. Este proceso se repetiría hasta que la trama por fin llegara intacta. Pero este proceso tendría un error fatal, ya que el receptor no sabría si lo que está viendo

es una trama recibida por primera vez o una retransmisión. La forma evidente de lograr esto es hacer que el emisor ponga un número de secuencia en el encabezado de cada trama que envía. A continuación, el receptor puede verificar el número de secuencia de cada trama que llega para ver si es una trama nueva o un duplicado que debe descartarse.

Este difiere de sus antecesores en cuanto a que tanto el emisor como el receptor tienen una variable cuyo valor se recuerda mientras la capa de enlace de datos está en estado de espera. Después de transmitir una trama, el emisor inicia el temporizador. Si ya estaba en operación, se restablece para conceder otro intervalo completo de temporización. Hay que elegir dicho intervalo de modo que haya suficiente tiempo para que la trama llegue al receptor, éste la procese en el peor caso y la confirmación de recepción se propague de vuelta al emisor. Sólo hasta que haya transcurrido ese intervalo podremos suponer con seguridad que se ha perdido la trama transmitida o su confirmación de recepción, y se debe enviar un duplicado.

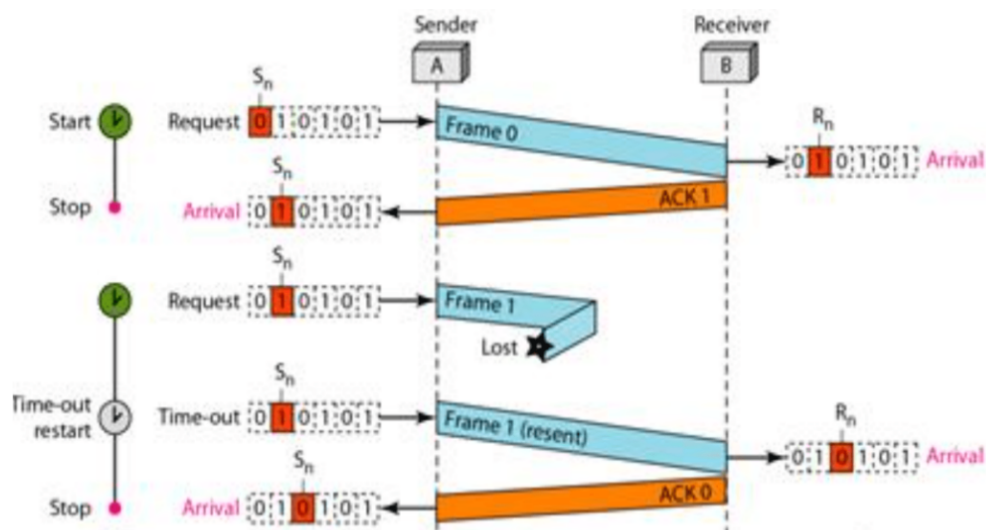


Figura 5: Protocolo simplex de parada y espera en canal ruidoso

Después de transmitir una trama e iniciar el temporizador, el emisor espera a que ocurra algo interesante. Sólo hay tres posibilidades: que una trama de confirmación de recepción llegue sin daño, que llegue una trama de confirmación de recepción dañada o que expire el temporizador. Si entra una confirmación de recepción válida, el emisor obtiene el siguiente paquete de su capa de red y lo coloca en el búfer, sobrescribiendo el paquete anterior. También incrementa el número de secuencia. Si llega una trama dañada o expira el temporizador, no cambia ni el búfer ni el número de secuencia de modo que se pueda enviar un duplicado. En todos los casos se envía a continuación el contenido del búfer (ya sea el siguiente paquete o un duplicado). Cuando llega una trama válida al receptor, su número de secuencia se verifica para

saber si es un duplicado. Si no lo es, se acepta, se pasa a la capa de red y se genera una confirmación de recepción. Los duplicados y las tramas dañadas no se pasan a la capa de red, pero hacen que se confirme la recepción de la última trama que se recibió correctamente para avisar al emisor de modo que avance a la siguiente trama o retransmita la trama dañada.

Como proposición al alumno, la cátedra recomienda tratar de implementar las tres variantes vistas en lenguaje C o similar, por ejemplo, Python. Para ello se deben crear pequeñas rutinas que implementen tanto el emisor como el receptor y una rutina principal que coordine cada protocolo.

3. Transmisión confiable: protocolos de ventana deslizante

En los protocolos anteriores, las tramas de datos se transmitían en una sola dirección. En la mayoría de las situaciones prácticas existe la necesidad de transmitir datos en ambas direcciones. Una manera de lograr una transmisión de datos como esta es tener dos instancias de uno de los protocolos anteriores, cada uno de los cuales debe usar un enlace separado para el tráfico de datos simplex (en distintas direcciones). A su vez, cada enlace se conformaría de un canal de “ida” (para los datos) y de un canal de “retorno” (para las confirmaciones de recepción). Sin embargo, en ambos casos se desperdiciaría la capacidad del canal de retorno casi por completo.

Una mejor idea es utilizar el mismo enlace para datos en ambas direcciones. Aunque intercalar datos y tramas de control en el mismo enlace es una gran mejora respecto al uso de dos enlaces físicos separados, es posible realizar otra mejora. Cuando llega una trama de datos, en lugar de enviar de inmediato una trama de control independiente, el receptor se aguanta y espera hasta que la capa de red le pasa el siguiente paquete. La confirmación de recepción se anexa a la trama de datos de salida. En efecto, la confirmación de recepción viaja gratuitamente. La técnica de retardar temporalmente las confirmaciones de recepción salientes para que puedan viajar en la siguiente trama de datos de salida se conoce como superposición. Sin embargo, la superposición introduce una complicación inexistente en las confirmaciones de recepción independientes. ¿Cuánto tiempo debe esperar la capa de enlace de datos un paquete al cual pueda superponer la confirmación de recepción? Si la capa de enlace de datos espera más tiempo del que tarda en terminar el temporizador del emisor, se volverá a transmitir la trama y se frustrará el propósito de enviar confirmaciones de recepción. Por supuesto que la capa de enlace de datos no puede predecir el futuro, de modo que debe recurrir a algún esquema particular para el caso.

Los siguientes tres protocolos son bidireccionales y pertenecen a una clase llamada protocolos de ventana deslizante. Los tres difieren entre ellos en términos de eficiencia, complejidad y requerimientos de búfer. En ellos, al igual que en todos los protocolos de ventana deslizante, cada trama de salida contiene un número de secuencia que va desde 0 hasta algún número máximo. Por lo general este valor máximo es $2^n - 1$, por lo que el número de secuencia encaja perfectamente en un campo de n bits.

La esencia de todos los protocolos de ventana deslizante es que, en cualquier instante, el emisor mantiene un conjunto de números de secuencia que corresponde a las tramas que tiene permitido enviar. Se dice que estas tramas caen dentro de la ventana emisora. De manera similar, el receptor mantiene una ventana receptora correspondiente al conjunto de tramas que tiene permitido aceptar. La ventana del

emisor y la del receptor no necesitan tener los mismos límites inferior y superior, ni siquiera el mismo tamaño.

Los números de secuencia en la ventana del emisor representan las tramas que se han enviado, o que se pueden enviar pero aún no se ha confirmado su recepción. Cada vez que llega un paquete nuevo de la capa de red, se le asigna el siguiente número secuencial más alto y el extremo superior de la ventana avanza en uno. Cuando llega una confirmación de recepción, el extremo inferior avanza en uno. De esta manera, la ventana mantiene en forma continua una lista de tramas sin confirmación de recepción.

3.1. Ventana deslizante de un bit

Antes de tratar el caso general, se examinará un protocolo de ventana deslizante con un tamaño máximo de ventana de 1. Tal protocolo utiliza parada y espera, ya que el emisor envía una trama y espera su confirmación de recepción antes de transmitir la siguiente.

Normalmente, una de las dos capas de enlace de datos es la que comienza a transmitir la primera trama. Si ambas capas se iniciaran en forma simultánea, surgiría una situación peculiar. La máquina que arranca obtiene el primer paquete de su capa de red, construye una trama a partir de él y la envía. Al llegar esta trama, la capa de enlace de datos receptora la revisa para saber si es un duplicado. Si la trama es la esperada, se pasa a la capa de red y la ventana del receptor se recorre hacia arriba. El campo de confirmación de recepción contiene el número de la última trama recibida sin error. Si este número concuerda con el de secuencia de la trama que está tratando de enviar el emisor, éste sabe que ha terminado con la trama almacenada en el búfer y que puede obtener el siguiente paquete de su capa de red. Si el número de secuencia no concuerda, debe continuar intentando enviar la misma trama. Por cada trama que se recibe, se regresa una.

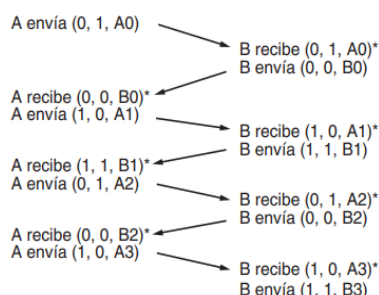


Figura 6: Ventana deslizante de 1 bit. La notación es (secuencia, confirmación, número de paquete). Un * indica el lugar en que una capa de red acepta un paquete

3.2. Ventana deslizante con retroceso N

El tiempo de viaje de ida y vuelta prolongado puede tener implicaciones importantes para la eficiencia del aprovechamiento del ancho de banda. La combinación de un tiempo de tránsito grande, un ancho de banda alto y una longitud de tramas corta es desastrosa para la eficiencia. La solución está en permitir que el emisor envíe hasta w tramas antes de bloquearse, en lugar de sólo 1. Con una selección adecuada de w , el emisor podría transmitir tramas continuamente durante un tiempo igual al tiempo de tránsito de ida y vuelta sin llenar la ventana.

La necesidad de una ventana grande en el lado emisor se presenta cuando el producto del ancho de banda por el retardo del viaje de ida y vuelta es grande. Si el ancho de banda es alto, incluso para un retardo moderado, el emisor agotará su ventana rápidamente a menos que tenga una ventana grande. Si el retardo es grande, el emisor agotará su ventana incluso con un ancho de banda moderado. El producto de estos dos factores indica básicamente cuál es la capacidad del canal, y el emisor necesita la capacidad de llenarlo sin detenerse para poder funcionar con una eficiencia máxima. Esta técnica se conoce como canalización.

Hay dos métodos básicos para manejar los errores durante la canalización. Una de las opciones, llamada retroceso N, es que el receptor simplemente descarte todas las tramas subsecuentes, sin enviar confirmaciones de recepción para las tramas descartadas. En otras palabras, la capa de enlace de datos se niega a aceptar cualquier trama excepto la siguiente que debe entregar a la capa de red. Si la ventana del emisor se llena antes de que expire el temporizador, el canal comenzará a vaciarse. En algún momento, el emisor terminará de esperar y retransmitirá en orden todas las tramas cuya recepción aún no se haya confirmado, comenzando por la trama dañada o perdida.

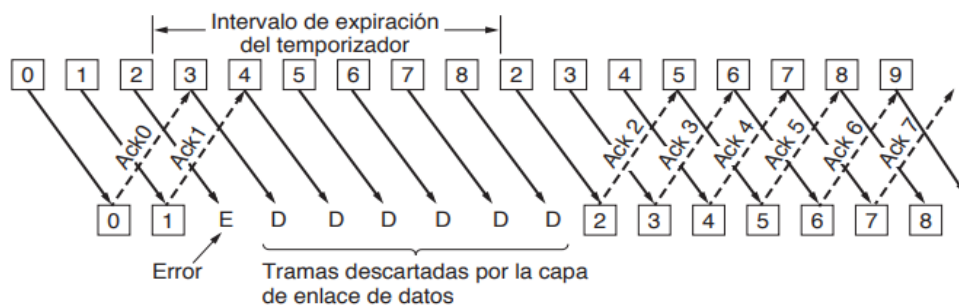


Figura 7: Ventana deslizante con retroceso N

En la anterior figura se puede ver el retroceso N para el caso en que la ventana del receptor es grande. Las tramas 0 y 1 se reciben y confirman de manera correcta. Sin embargo, la trama 2 se daña o pierde. El emisor, sin saber sobre este problema, con-

método puede requerir grandes cantidades de memoria en la capa de enlace de datos.

A menudo, la repetición selectiva se combina con el hecho de que el receptor envíe una confirmación de recepción negativa NAK al detectar un error. Las NAK estimulan la retransmisión antes de que el temporizador correspondiente expire y, por lo tanto, mejoran el rendimiento.

Una vez más, la cátedra propone al alumno, tratar de implementar estos protocolos en lenguaje C o Python.

4. Anexo: Introducción a Packet Tracer

Cisco Packet Tracer es un software propiedad de Cisco System Inc., diseñado para la simulación de redes basadas en los equipos de la compañía. Junto con los materiales didácticos diseñados con tal fin, es la principal herramienta de trabajo para pruebas y simulación de prácticas en los cursos de formación de Cisco System. Para obtener el software deberán crearse una cuenta en Cisco e inscribirse al curso de Introducción. Luego de esto, la plataforma les permitirá descargarlo gratuitamente. Si bien podrán trabajar con la versión que deseen, desde la cátedra proponemos la variante 7.2.2 y al mismo tiempo es la que utilizaremos a continuación para explicar las bases.

4.1. Entorno de trabajo

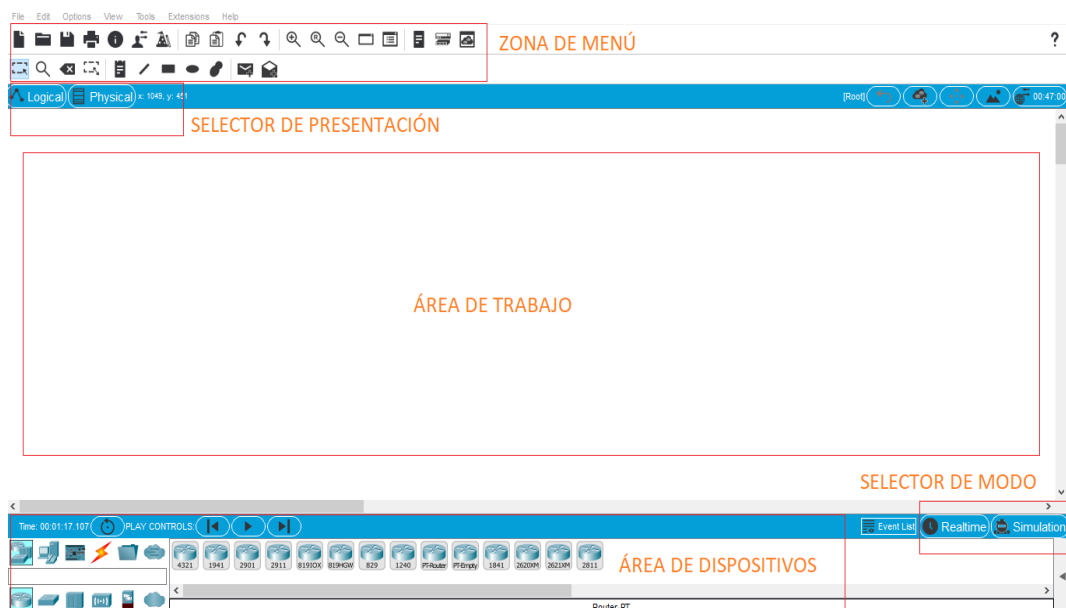


Figura 9: Ventana deslizable con repetición selectiva

En el espacio de trabajo de Packet Tracer se encuentran diferentes zonas:

- Zona de menú. Es el área donde se encuentran las opciones típicas de todos los programas para la gestión y la configuración del software.
- Selector de presentación. Permite cambiar entre esquema lógico y esquema físico a la hora de presentar los dispositivos. Lo habitual es trabajar con el esquema lógico.
- Espacio de trabajo. Es la zona donde se situarán los dispositivos que conforman la red.

- Selector de modo. Para cambiar entre el modo de Tiempo Real o el modo Simulación, el cual permite un análisis más detallado de todos los paquetes de los diferentes protocolos que intervienen en una comunicación en la red.
- Área de dispositivos. Es la zona que permite seleccionar los dispositivos que van a ser incluidos en el espacio de trabajo, así como la conexión entre estos. La zona izquierda recoge los dispositivos por grupos y la zona derecha del área ofrece los dispositivos incluidos, de acuerdo con la numeración utilizada por Cisco System.

4.2. Creación de topología de red

El modo de operación con Packet Tracer es muy sencillo ya que se trata de un programa muy intuitivo. La primera operación consistirá en seleccionar los dispositivos que forman la red, para ello se seleccionará el grupo correspondiente a la izquierda del área de dispositivos. Cada uno de los dispositivos seleccionable se corresponde con un dispositivo fabricado por Cisco System. La selección de los dispositivos puede hacerse uno a uno (señalándolo en el grupo y haciendo clic en el escenario para colocarlo) o si se trata de varios dispositivos similares, señalándolo en el grupo a la vez que se pulsa la tecla Ctrl.

El conexionado de los distintos equipos se puede realizar eligiendo personalmente el tipo de conexión o mediante la herramienta de conexionado automático. En cualquier caso, hay que señalar sobre los dispositivos a conectar y, si el caso lo requiere, se nos ofrecerá la posibilidad de elegir el tipo de interfaz.

Cuando los dispositivos se encuentran sobre el escenario, al situar el cursor sobre ellos aparecerá un recuadro con la información acerca de su configuración a nivel de red. En cada una de las conexiones aparecerá un indicador de conectividad a nivel físico que podrá estar rojo (no hay conectividad), naranja (la interface está en proceso de inicio) o verde (la interfaz está operativa). La configuración de los parámetros de red será un proceso que deberán realizar ustedes.

Al marcar un dispositivo se abrirá la ventana del dispositivo en la que aparecen tres pestañas seleccionables:

1. Físico. Muestra una representación del equipo físico y los módulos de ampliación y/o configuración disponible para el citado equipo (según referencia de Cisco System), de manera que es posible quitar o poner módulos a voluntad para que el equipo disponga de las interfaces o módulos previstos en el diseño. Para hacer esta operación será necesario primero apagar el dispositivo, ya que, por defecto, todos los dispositivos se encienden cuando son colocados en el escenario.

2. Config. Ofrece las opciones de configuración del dispositivo a nivel general (Global), de enrutamiento en el caso de routers y de las interfaces instaladas de manera individual (Interfaz).
3. CLI. Sólo disponible en routers y switches. Sirve para programar el dispositivo en modo comandos tal como se haría a través de la consola en un dispositivo real. Este es el modo que recomendamos usar para afianzar las directivas que luego se utilizarán (similares) en Kathará.

La comprobación de la correcta configuración de los dispositivos, una vez que todos los indicadores de conexión física están en color verde, se puede realizar de forma rápida situando el cursor en cada uno de los dispositivos y analizando el resumen de la configuración que se muestra en una ventana emergente.

Como actividad, se propone al alumno hacer una conexión básica entre dos dispositivos cualesquiera, arrastrandolos del área de dispositivos hacia el área de trabajo.

Con esta información, ya es posible iniciar el trabajo práctico de Capa de Enlace de Datos disponible en el campus. El mismo debe estar correctamente justificado con los contenidos desarrollados en este informe y en el posterior, que se entregará en breve sobre la Subcapa de Acceso al Medio.