Anales PANEL'81/12 JAIIO
Sociedad Argentina de informática
e Investigación Operativa. Buenos Aires, 1981

DISEÑO DE PROTOCOLOS PARA LA RED LOCAL UNIANDES

Rodrigo Paredes
Laureano A. Gaitán
Roberto Pardo

Departamento de Ingeniería de Sistemas y Computación
Universidad de Los Andes
Apartado Aéreo 4976, Bogotá - Colombia.

O B J E T I V O

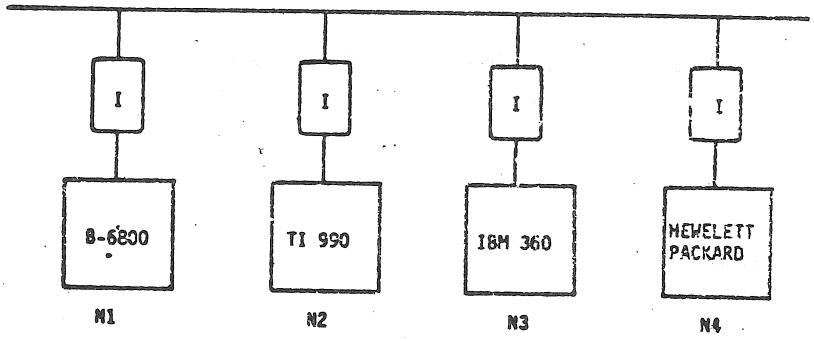
Este artículo describe las principales decisiones de diseño realizadas en el desarrollo de protocolos para la red local UNIANDES. Específicamente se concentra en dos tipos de protocolos : el de comunicación entre procesos o de transporte (PT) y el de transferencia de archivos (PTA).

La descripción de cada protocolo comprende : a) las funciones que debe realizar, b) las suposiciones, c) una especificación, d) la interfase que define, y e) los módulos necesarios para su implantación. Varios ejemplos y sugerencias sobre implantación complementan estas descripciones.

TERMINOS CLAVES : Redes locales, Protocolos de Alto Nivel, Implantación de Protocolos.

1. INTRODUCCION:

La "RED UNIANDES" es una red local experimental - actualmente en fase de diseño - nació como un esfuerzo conjunto entre los Departamentos de Ingeniería de Sistemas y Computación e Ingeniería Eléctrica de la Universidad de los Andes. El esquema general de la red es el de un conjunto de nodos hetereogéneos (computadores grandes, medianos y pequeños existentes en la Universidad) que a través de interfases se interconectan al compartir un medio de transmisión común (topología tipo bus) (Ver figura 1).



I: INTERFASE (M 6800)
N1: NODO I
FIGURA 1 - Topología Red UNIANDES.

Durante el desarrollo del proyecto se han identificado varios problemas para atacar: el medio de transmisión, el hardware de la interfase, el software de comunicación residente en las interfases y en los nodos, software distribuído, etc. Los componentes más importantes del software de comunicación son los protocolos, o sea las reglas y convenciones necesarias para intercambiar información entre entidades (interfases, procesos). El objetivo de este artículo es describir las decisiones de diseño de dos tipos de protocolos de la red y que residen en los nodos: el de transporte (PT) y el de transferencia de archivos (PTA).

Tipicamente los protocolos de una red forman una jerarquía bien definida de servicios, donde los servicios de un nivel se 'aumentan' mediante un protocolo del nivel siguiente hacia arriba. En la sección 2 se describe la arquitectura de protocolos de la red.

Para describir el diseño de cada protocolo, secciones 3 y 4, se utiliza la siguiente metodología: primero se describen las funciones que deben hacer y los supuestos en cada caso. A continuación se hace una especificación, es decir se describe sin ambiguedades lo que el protocolo debe hacer. Tam

bién en esta parte se definen las primitivas (interfase) externas que 've' el nivel de más arriba para cada protocolo. Finalmente se procede a describir detalladamente los módulos necesarios para implantar cada tipo de protocolo. Varias sugerencias sobre detalles de implantación se ilustran en el artículo.

2. ARQUITECTURA RED UNIANDES.

En la figura 2 se muestra los principales niveles conceptuales que constituyen una arquitectura. Los niveles son importantes puesto que además de aislar y modularizar funciones específicas, sierven de soporte para el desarrollo de niveles superiores.

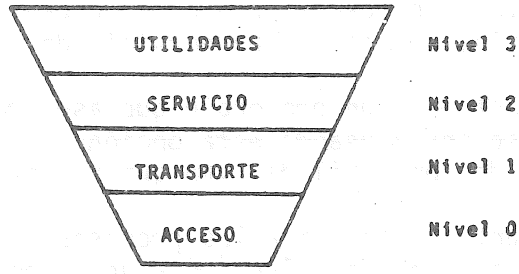


FIGURA 2 - Niveles conceptuales de una Arquitectura

El nivel más bajo (nivel 0) corresponde a un protocolo de acceso al medio de transmisión compartido y su función es la de transmitir y recibir paquetes a/de la línea. El siguiente nivel (nivel 1) corresponde a los protocolos de transporte los cuales permiten la comunicación entre procesos. En el nivel intermedio (nivel 2) están los protocolos que implementan servicios (v, g., transferencia de archivos, interacción con terminales, etc). El nivel superior (nivel 3) corresponde a los protocolos de más alto nivel tales como: sistemas de archivos distribuídos, base de datos distribuídas.

Los protocolos definidos inicialmente para la red se ilustran en la figura 3.

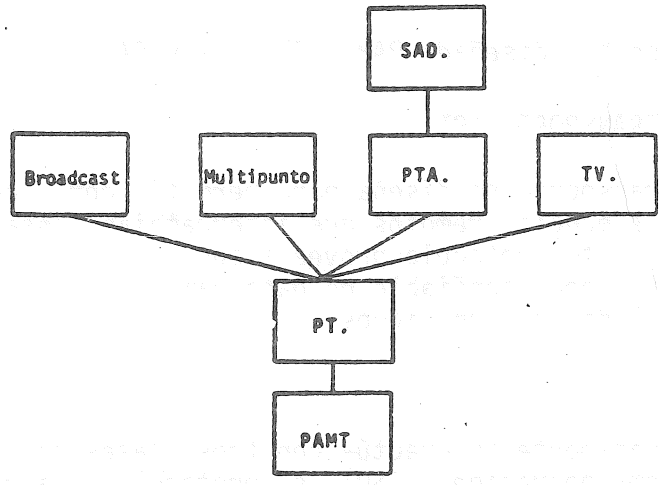


FIGURA 3 - Arquitectura Red UNIANDES

Sus funciones brevemente son las siguientes:

Protocolo de Acceso al Medio de Transmisión compartido (PAMT): Este protocolo solo entiende paquetes y corresponde a un protocolo de contención el cuaL se encarga de transmitir y retransmitir paquetes a la línea en el caso que exista alguna colisión. Este protocolo se implementa en la interfase.

Protocolo de Transporte (PT): Este protocolo solo "entiende" cartas o mensajes y dentro de él se definen dos niveles de confiabilidad: uno bastante confiable, para intercambiar cartas de una manera segura entre procesos, es decir detectar duplicados, preservar orden, controlar flujo, etc. y otro menos confiable, que ofrece un servicio de datagrama.

Protocolo de Broadcast: Es un protocolo que asegura a los procesos usuarios que un mensaje que se envíe usando esta opción, llegará a todos los nodos de la red con el nivel de confiabilidad con que se envió.

Protocolo de Multipunto: Asegura a los procesos usuarios que un mensaje que se envíe usando esta opción, llegará a un grupo de usuarios previamente definido.

Protocolo de Transferencia de Archivos (PTA): Es usado para transferir archivos entre los nodos de la red. Este protocolo resuelve incompatibilidades en la representación de los archivos, permite hacer puntos de chequeo, etc.

Protocolo de Terminal Virtual (TV): Este protocolo permite interactuar con terminales localizadas en los diferentes nodos. Resuelve incompatibilidades en cuanto a la representación de las terminales.

Sistema de Archivos Distribuído (SAD): Es un sistema diseñado para dar soporte a aplicaciones de propósito general, dando primitivas bien definidas para el diseño de aplicaciones.

En la actualidad se han diseñado PAMT, PT, PTA y SAD.

3. PROTOCOLO DE TRANSPORTE (PT).-

El protocolo de transporte se diseña para permitir comunicación entre procesos, programas en ejecución remotos que intercambian mensajes o cartas a través de la red. Este protocolo provee a sus usuarios con dos niveles de servicios: un nivel poco confiable de datagrama y un nivel superconfiable que ofrece la facilidad de conexiones.

3.1 SUPOSICIONES.

El protocolo de transporte interactúa con tres clases de procesos dentro de un nodo: el sistema operacional (SO), el protocolo de acceso al medio de transmisión (PAMT) y los procesos usuarios (PU).

Los procesos usuarios son los que demandan servicios del protocolo por medio de un conjunto de primitivas, cuya función es indicar al protocolo sobre los datos que se desean transmitir en una conexión. Estos datos son fragmentados en cartas (si se necesita) que son las que entiende el protocolo a este nivel; posteriormente las cartas son partidas en paquetes para que puedan ser manejadas por el protocolo de acceso al medio de transmisión.

A partir de este momento nos referimos al nivel superconfiable como el protocolo de transporte puesto que este nivel es más interesante dado que permite asociar a cada usuario un canal lógico seguro, por medio del cual se pueden transmitir mensajes confiablemente, aunque en realidad solo se está compartiendo un canal físico inseguro.

3.2 FUNCIONES.

El PT debe proveer métodos que aseguren un manejo confiable y eficiente de los mensajes o cartas intercambiadas entre procesos usuarios, por esto se hace necesario desarrollar mecanismos que implanten las siguientes funciones:

a. Direccionamiento : A nivel de la red existe un conjunto de direcciones lógicas que son conocidas con el nombre de puertos y el PT debe ser capaz de asociar a cada dirección lógica (PTO) la dirección física dentro de la máquina.

b. Manejo de conexiones: Incluye la forma en que se deben abrir y cerrar conexiones, de manera tal que no se permitan errores como aceptar paquetes de una conexión previamente cerrada.

c. Manejo de paquetes: Incluye el poder detectar y manejar situaciones en las que los paquetes se han dañado, perdido, duplicado o transpuesto en relación a la forma en que fueron enviados.

d. Fragmentación y ensamblaje : Dentro de una red se debe definir el tamaño máximo de los paquetes que viajan por las líneas; por lo tanto, si un usuario desea transmitir una carta que tenga un tamaño mayor al definido para un paquete, esta debe ser fragmentada en paquetes en el nodo emisor, y estos a su vez deben ser ensamblados por el nodo receptor.

e. Control de Flujo : Es el mecanismo usado por el PT para controlar que no se envíen más paquetes de los que se pueden recibir sobre cada conexión.

El Protocolo de transporte también debe poder formatear e interpretar la información de control que viaja con los paquetes, controlar los eventos concurrentes que provienen de los usuarios y además multiplexar, es decir que una sola copia del protocolo atienda a varios usuarios simultaneamente.

3.3 ESPECIFICACION.

La especificación que se encuentra en este numeral se refiere a los mecanismos de timers (FLET78). Por medio de la especificación se muestran los diferentes estados por los que puede pasar el protocolo al enviar o recibir un paquete sobre una conexión, lo cual ayuda a entender la secuencia de pasos que se deben seguir al implementar las reglas de timers en los módulos.

PARTE DEL EMISOR.

Las transiciones aquí mostradas dependen de eventos internos generados por el protocolo y eventos externos o de los usuarios.

Las transiciones son definidas en ambos casos, (emisor, receptor) por las reglas de timers.

Para ver la especificación global y total del protocolo y ademas los cambios de estado producidos por otros eventos ver (PAGA80)

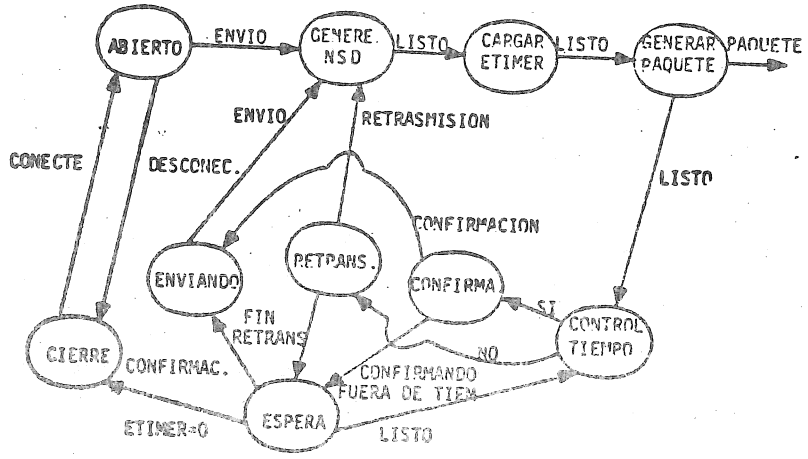


FIGURA 4a. Especificación del PT: Parte Emisor

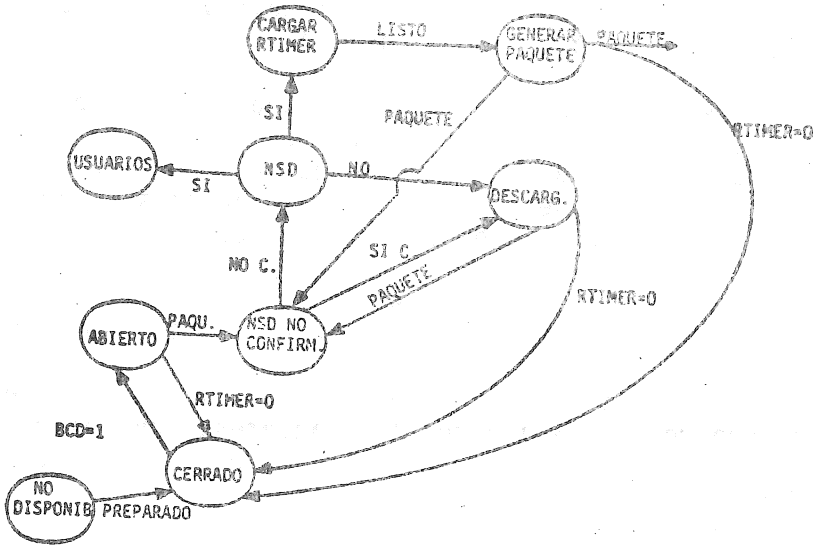


FIGURA 4b. ESPECIFICACION DEL PT: Parte Receptor

3.4 INTERFASE.-

Las interacciones que realiza el PT con sus "vecinos" (SO, PU, PAMT) son conocidas como interfases, y se hacen por medio de un conjunto de primitivas las cuales definen los eventos que se deben ejecutar.

INTERFASE USUARIO - PT:

En esta interfase se definen las primitivas por medio de las cuales se comunican el PT y el proceso usuario. Dichas primitivas son un conjunto de eventos de llamada/retorno que no se encuentran en una relación determinada.

PRIMITIVAS DE INICIACION/TERMINACION DE CONEXIONES:

- Conecte (pide una conexión entre dos puertos)
- Preparado (indica al PT de la disponibilidad de recibir mensajes de un puerto remoto).
- Desconecte (finaliza una conexión en forma diferida).
- Destruye (termina una conexión en forma inmediata).

PRIMITIVAS PARA LA TRANSFERENCIA DE DATOS:

- Envíe (indica al PT que se desea transmitir una carta sobre una conexión).
- Reciba (indica al PT que se dispone de un buffer para recibir una carta sobre una conexión).
- Cancele-envío (indica al PT que un pedido de envío ha sido cancelado).
- Cancele-reciba (indica al PT que un buffer previamente ofrecido para recibir una carta ha sido retirado).

PRIMITIVAS DE SINCRONIZACION:

- Interrumpa (utilizada para mandar mensajes prioritarios sobre una conexión).
- Reinicie (indica el reinicio normal del flujo de paquetes sobre una conexión).

En los ejemplos se dan algunas secuencias que pueden ocurrir en un normal uso del PT y además cómo es el flujo de estas a través de los módulos que implantan el protocolo.

3.5 MODULOS DEL PT.-

Los módulos funcionales del protocolo (ver figura 5) estan compuestos de un conjunto de procesos y rutinas que en unión con las estructuras de datos (tablas, colas, etc) realizan sus funciones asignadas.

A continuación se describe de una manera general cada módulo y sus funciones principales. (En los ejemplos, sección 3.6, se ilustra la descripción más detallada de un módulo.)

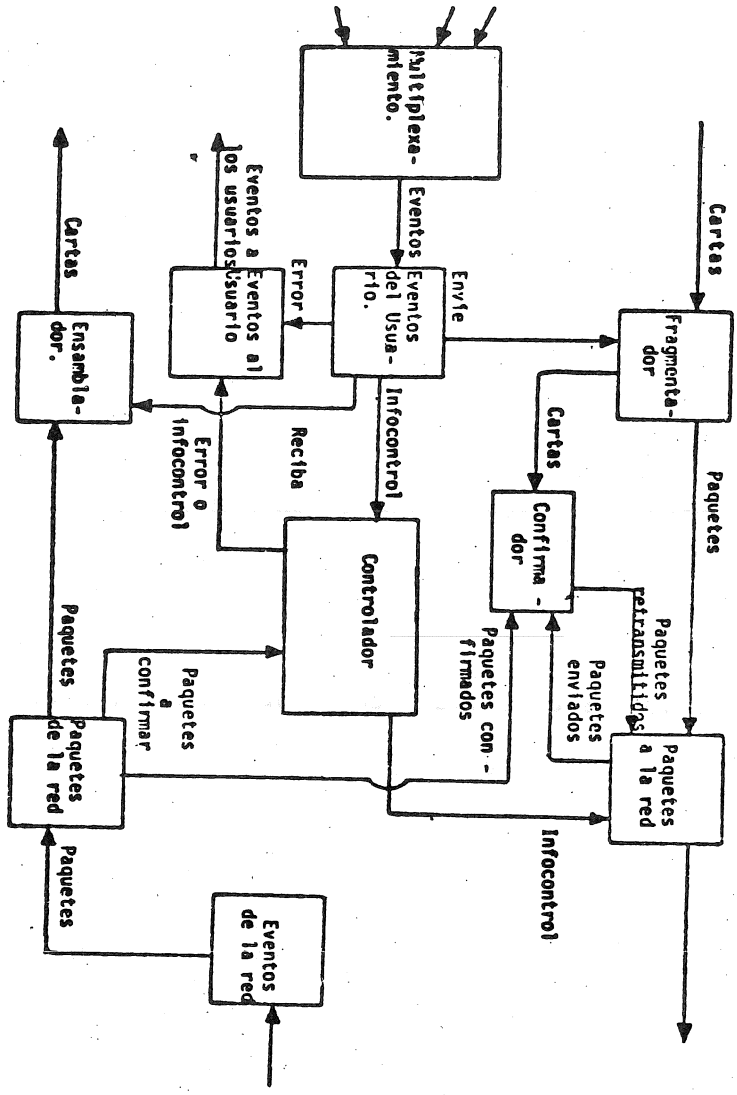


FIGURA 5. Modulos del PT

MODULO DE MULTIPLEXAMIENTO : La función de este módulo es la de controlar los eventos concurrentes producidos por los procesos usuarios del PT; esto se puede hacer por medio de alguno de los métodos de comunicación entre procesos. El módulo debe ir recolectando los eventos y colocándolos en una cola de eventos la cual será procesada por otro módulo.

MODULO DE EVENTOS DEL USUARIO: La función que realiza este módulo es la de interpretar y analizar los eventos provenientes de los usuarios (v.g: ver si los parámetros están correctos). Además se debe tomar alguna acción según el evento, por ejemplo : si es "Envíe" se debe avisar al módulo fragmentador que hay una nueva carta para enviar; esto se hace colocando la dirección y el tamaño de la carta en la cola de cartas del usuario la cual será procesada por el módulo fragmentador. Si es un "reciba" se le avisa al módulo ensamblador. Si son eventos de control (v.g. conecte, preparado,...) se comunicará al módulo controlador para que este los procese..

MODULO FRAGMENTADOR : Este módulo lo que hace es partir las cartas cuya

longitud exede a la permitida por un paquete (datos) de una manera lógica.) Crea una cola de paquetes, una por cada conexión para que el módulo que envía paquetes a la red se encargue de colocarle la información de control necesaria para ser entendido por el PT remoto.

MODULO ENSAMBLADOR: La función de este módulo es la de ensamblar cartas, para un proceso usuario, basado en los paquetes que llegan sobre esta conexión; para esto tiene una cola de paquetes recibidos de la red, la cual tiene como información la dirección de donde está el paquete y su longitud.

MODULO QUE ENVIA PAQUETES A LA RED: Este módulo es el encargado de colocar los Números de Secuencia de Datos a los paquetes y de confirmar al otro PT los datos que han arrivado, de manejar las reglas de timers del emisor cada vez que un paquete es formateado y enviado a la interfase coloca la información sobre cada uno de estos paquetes en una cola para que el módulo confirmador cada vez que reciba información sobre la entrega de cada paquete en el PT remoto lo libere.

MODULO CONFIRMADOR: Es el encargado de controlar que todos los paquetes enviados sean confirmados y cuando completan una carta, libera el buffer donde estaba la carta para que pueda ser utilizado. También tiene la función de indicar al módulo que envía paquetes a la red de una retransmisión después de pasado un intérvalo de tiempo en que un paquete que no ha sido confirmado.

MODULO DE EVENTOS DE LA RED: Tiene la función de estar pendiente de la llegada de nuevos paquetes provenientes de la interfase. Coloca la dirección donde quedó el paquete en una cola de paquetes de la red la cual será procesada por el módulo de recepción de paquetes de la red.

MODULO RECEPTOR DE PAQUETES DE LA RED: Este módulo es el encargado de procesar los paquetes provenientes de la red. Controla que los paquetes que han arrivado lleguen en el orden esperado y dentro de la ventana preestablecida. Avisa al módulo confirmador de paquetes confirmados sobre cada conexión y al módulo controlador para que confirme los elementos arrivados. También coloca la información de cada paquete aceptado en la cola de paquetes recibidos para el módulo ensamblador.

MODULO CONTROLADOR: Es el encargado de procesar los eventos de control que se ejecutan sobre cada conexión. Tiene asociada una cola de eventos de control donde va información acerca de paquetes confirmados, etc. Debe controlar situaciones como una interrupción del usuario, etc.

MODULO DE EVENTOS AL USUARIO : La función de este módulo es la de avisar a los procesos usuarios de los diferentes problemas y situaciones anormales o pedidos hechos por el usuario.

3.6 EJEMPLO;

Supongamos la siguiente conversación en dos nodos de la red:

NODO 1

Usuario X PT
Conecte X, Y
Evento 1
Envíe C1
Evento 2
Envíe C2
Evento 3
Desconecte X,Y
Evento 4

NODO 2

PT Usuario Y
Preparado X
Evento 1
Reciba
Evento 2
Reciba
Evento 3

En el nodo 1, inicialmente el módulo de multiplexamiento monitorea los eventos del usuario X y los pasa a una tabla de eventos, la cual es analizada por el módulo de eventos del usuario.

El primer evento es "conecte X,Y" entonces el módulo de evento mira si la conexión ya existe. Si existe, la rechaza, si no le avisa al módulo de control para que cree el registro de conexión. En este momento no se produce ningún intercambio de mensajes, sino se espera por el primer evento de envíe. Una vez el módulo de multiplexamiento haya pasado el segundo evento, (envíe C1) a la tabla de eventos el módulo de eventos del usuario se encarga de colocar la dirección y el tamaño de la carta a ser enviada (C1) en una cola de cartas del usuario.

Por medio de la cola de cartas del usuario el módulo fragmentador las convierte en paquetes, lo cual se hace de manera "lógica" puesto que en realidad solo se crea otra cola, 'de paquetes fragmentados' en la que se tiene la dirección, el tamaño de los paquetes, los números de secuencia, etc. Cada vez que una carta es totalmente fragmentada es colocada en una cola de cartas fragmentadas, para que cuando se confirme su entrega en el nodo receptor se pueda liberar el buffer donde estaba la carta.

El módulo de paquetes a la red es el encargado de procesar los paquetes fragmentados no para que estos sean enviados a la interfase una vez sean formateados, es decir después de que se coloque toda la información de control para que sea interpretada por el nodo receptor. Todo paquete que es mandado a la red es colocado en la cola de paquetes enviados no para que cuando sean confirmados por el PT remoto sean liberados por parte del módulo confirmador, el cual recibe información de control sobre paquetes que han arrivado correctamente el nodo remoto por parte del módulo de paquetes de la red.

El módulo confirmador también es el encargado de indicar al módulo de paquetes a la red cuando retransmitir un paquete que ha sido enviado y no confirmado. Por esta sucesión de pasos pasarían los eventos 2 y 3 del ejemplo.

Cuando ocurre el evento 4 "desconecte", el módulo de eventos del usuario informa al módulo de control sobre este evento y el módulo de control deja transmitir los paquetes que faltan por enviar y luego deja caer el timer del emisor a cero para asegurarse de que no quede vivo ningún paquete de esta

conexión en la red. Si mientras está en período de desconexión ocurre algún evento por parte del usuario sobre la conexión es rechazado, esto es lo que ocurre en el caso del nodo 1.

En el nodo 2 ocurriría lo siguiente: el evento 1 "preparado X" es pasado por el módulo de eventos del usuario al controlador, el cual crea un registro de conexión que se completará cuando llegue el primer paquete de uno de los usuarios especificados en la primitiva. El evento 2 "reciba" indica un buffer en el cual se colocarán los paquetes provenientes de un usuario remoto. El tamaño de los buffer disponibles determinará el control de flujo sobre la conexión. Una vez todos los paquetes han sido recibidos el controlador dejará caer el timer del receptor a cero y destruirá el registro de conexión. Los paquetes que arrivan son ensamblados en cartas por parte del módulo ensamblador.

Más detalles sobre algoritmos se encuentran en (PAGA80).

4. PROTOCOLO DE TRANSFERENCIA DE ARCHIVOS (PTA).-

Los archivos en una máquina pueden ser vistos como una colección de datos representados de acuerdo a las convenciones locales del sistema operacional (v.g; alfabeto, convención de fin archivo, etc.) Para realizar operaciones sobre estos archivos cada máquina tiene un sistema de manejo de archivos local, el cual provee funciones tales como: copiar archivos, borrar archivos, crear archivos, etc.

El problema al transferir o manipular archivos en una red, es el resolver los conflictos de las representaciones contextuales y el de extender las facilidades disponibles localmente. Para esto, es necesario diseñar un sistema (PTA) que permita estandarizar las representaciones de los archivos y proveer métodos para hacer operaciones sobre archivos a nivel de la red.

4.1 SUPOSICIONES.

El PTA interactúa dentro de un nodo con tres tipos de procesos: Proceso usuarios (PU); Sistema de manejo de archivos local (SAL); Protocolo de transporte (PT). El protocolo interactúa con los procesos usuarios por medio de un lenguaje interactivo, con el sistema de manejo de archivos locales para ejecutar las operaciones que él ofrece (copiar, borrar, etc) y con el protocolo de transporte para utilizar los servicios que él presta (v.g; establecer conexiones, recuperación y detección de pérdida y duplicado de paquetes, etc.) (Figura 6).

4.2 FUNCIONES.

COmo se dijo anteriormente el problema al hacer operaciones en los archivos de una red, es el resolver los conflictos de las representaciones contextuales y el extender las facilidades disponibles localmente. Siempre que se manipule un archivo dos partes van a tomar lugar en la operación: el usuario y el servidor. El usuario inicia las acciones y el servidor responde, es decir existe una relación de maestro/esclavo entre el usuario y el servi

dor (figura 6). Para simplificar la situación de las representaciones contextuales de un archivo, cada vez que se opere un archivo, los comandos entre el usuario y el servidor viajarán con una información de control describiendo parcial o totalmente el archivo; esta información de control puede ser generada por cualquier proceso que decida transmitir un archivo y similarmente, cualquier proceso que lo reciba puede usar esta información para completar la operación que sobre él se realice.

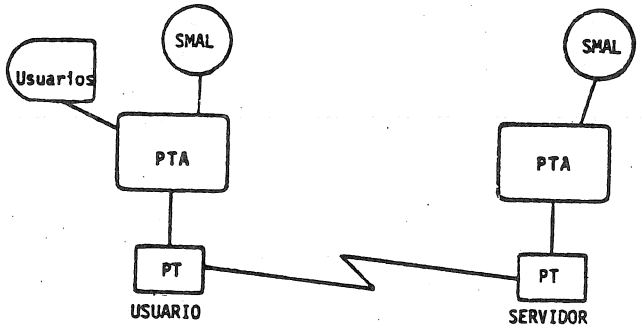


FIGURA 6 - Operación del PTA

Representación Estandar de los Archivos.

Para poder estandarizar las representaciones de un archivo en la red, se ha definido un archivo "virtual" el cual es entendido en todos los nodos de tal manera que cuando se transmita un archivo, éste debe ser convertido ("mapeado") por el PTA a la representación virtual y una vez arrive al PTA remoto será convertido ("mapeado") a las convenciones locales del nodo receptor (Figura 7).

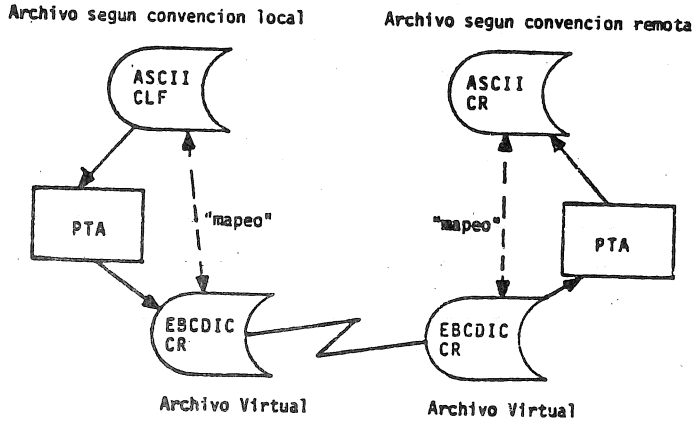


FIGURA 7 - Estandarización de los Archivos

Puntos de chequeo.

Otra de las funciones del PTA cuando está transfiriendo un archivo es el proveer una manera para realizar puntos de chequeo en la transmisión, es decir el colocar marcas de control sobre los datos del archivo enviado para que cada vez que estos arriven al nodo receptor, el PTA de este nodo confirme el recibimiento satisfactorio de ellos.

Esta función es importante puesto que si llega a ocurrir una falla en la transmisión de un archivo no es necesario retransmitirlo todo nuevamente, sino a partir del último punto de chequeo confirmado sobre los datos del archivo.

4.3 ESPECIFICACION.

La especificación descrita en esta sección muestra los estados globales del protocolo de transferencia de archivos y su interacción con el SAL y el PT.

Cada estado del protocolo es cambiado por un conjunto de eventos internos y externos, que definen los pasos a seguir por él. La especificación detallada de los mecanismos utilizados por el PTA pueden ser vistos en (PAGA80).

A continuación se describe la parte del protocolo cuando está trabajando en modo usuario y luego en modo servidor.

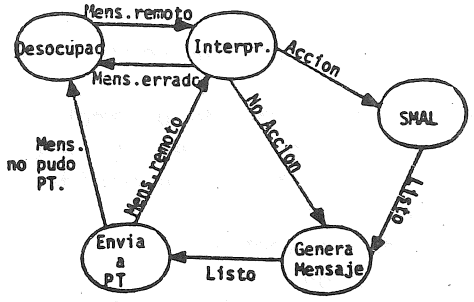


FIGURA 8a- Especificación del PTA: Modo Servidor

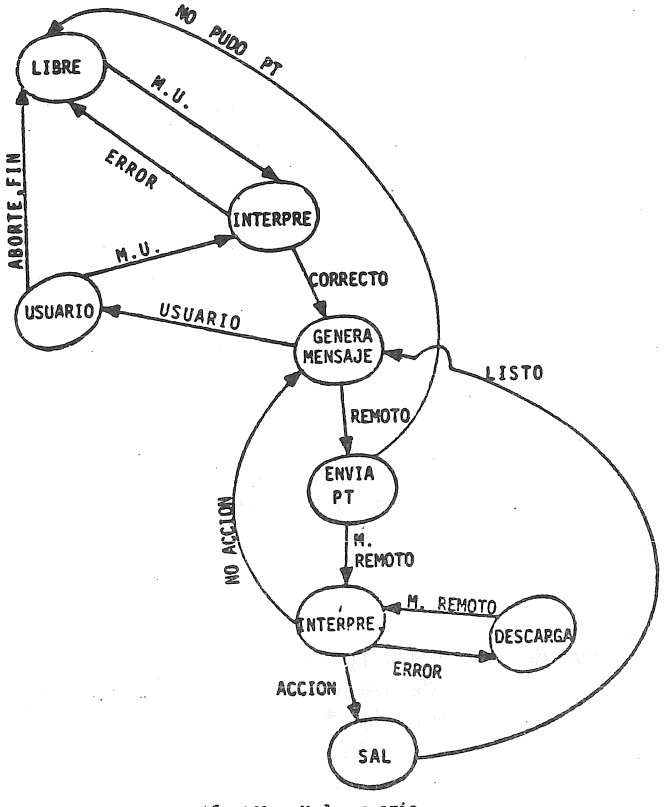


Figura 8b. Especificación PTA : Modo usuario

4.4 INTERFASE.

Las interacciones que realiza el PTA con sus vecinos (proceso usuarios, sistema de manejo de archivos local, protocolo de transporte) son conocidas como interfases y se hace por medio de un conjunto de primitivas las cuales definen los eventos que se deben ejecutar.

PRIMITIVAS DEL USUARIO AL PTA:

Son los comandos que permiten el acceso al protocolo y son utilizados por el usuario por medio de un lenguaje interactivo.

Comandos:

- Comienzo: Indica la iniciación de una serie de pedidos del usuario al protocolo.
- Tráigame: pide al servidor en envío de un archivo.
- Transfiera: Pide al servidor que almacene un archivo.
- Aborte: Anula el servicio previamente requerido.
- Cree: Demanda la creación de un archivo.
- Borre: Pide que sea borrado un archivo.
- Renombre: Cambia el nombre a un archivo.
- Copie: Hace la copia/concatenación de archivos.
- Fin: Indica el fin de los pedidos del usuario al protocolo.
- Ctransf: Implica el principio de la fase de transmisión de datos.

PRIMITIVAS DEL PTA AL USUARIO:

Estos comandos tienen por objeto informar al usuario si el servicio requerido se ejecutará o no y la culminación de él en caso de su realización.

Comandos:

- Respuesta: Informa al usuario si su pedido se puede realizar (si) si no es posible (no)
- Terminación: Informa al usuario la culminación normal o no de su pedido.

PRIMITIVAS ENTRE PTA:

Son los comandos de control de la transferencia de datos. Durante la transferencia los datos pasan desde el transmisor al receptor hasta un comando de final de datos (FD).

Comandos:

- "Archivo": Comando de principios de datos.
- "FD" : Comando de fin de datos.
- "Espere" : Ejecuta un pedido de espera (Hold (3)).
- "RR" : Pedido de restauración.
- "CPOS" : Confirmación positiva/negativa de algún evento.

En el ejemplo se dan algunas secuencias que pueden ocurrir en un normal uso de PTA y el flujo de estas a través de los módulos que implementan el protocolo.

INFORMACION DE CONTROL QUE VIAJA CON LAS PRIMITIVAS.

La información de control está compuesta por parámetros los cuales especifican un atributo y su valor. Dentro de los parámetros hay dos clases principales:

- Parámetros que sirven para identificación de un archivo (v.g.; unidad DØ1, nombre = datos, versión = Ø1, etc.)
- Parámetros que permiten negociar las características del archivo virtual (v,g; alfabeto= ascii, conversión-fín-de-línea=cr, etc).

Como la información de control es generada por cualquier proceso (usuario) que desee transferir o manipular un archivo, el PTA debe estar en capacidad de responder a todos los atributos permitidos y sus valores.

ATRIBUTOS DE LA INFORMACION DE CONTROL:

- "Unidad"
- "Nombre-Archivo"
- "Versión"
- "Nombre-Directorio"
- "Tipo" - Especifica el tipo de archivo: texto o datos.
  Datos bytes (3) de longitud 1 a n
  Texto: 8 bits (3) ASCII (3)
- "Alfabeto"
- "Convención-de-fín de línea"
- "Tamaño-bytes"
- "Marca-de-control" Especifica el número de cartas (ventana) después de las cuales se debe confirmar el recibimiento de los datos.

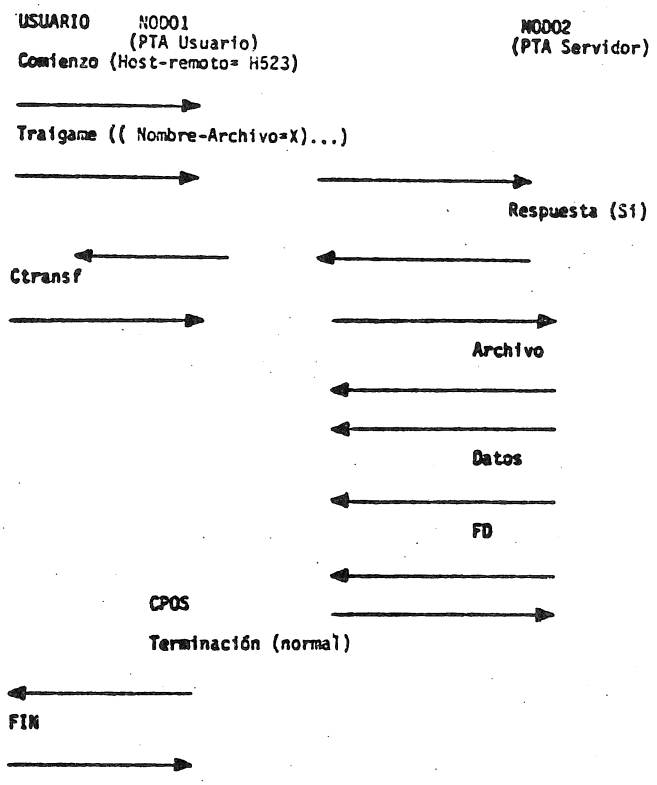
4.5 MODULOS DEL PTA.

Los módulos funcionales del protocolo (ver figura 9) están compuestos por un conjunto de procesos y rutinas que en unión con las estructuras de datos (tablas, colas, etc.) ejecutan las funciones asignadas.

La forma general de cada módulo es muy parecida a la de los módulos del PT; una descripción detallada de ellos se encuentra en (PAGA80)

4.6 EJEMPLO :

Supongamos la siguiente secuencia de eventos para la transmisión de un archivo :



El monitor de eventos del usuario se encarga de pasar los comandos del usuario al interpretador de comandos; en el ejemplo el primer comando es "comienzo", este evento luego de ser analizado sintácticamente por el interpretador es pasado al controlador, el cual crea un registro para el usuario y avisa al módulo que interactua con el PT para pedirle una conexión a un host remoto.

El segundo comando "Tráigame" pide que se envíe un archivo del nodo receptor al nodo emisor, para esto se avisa al módulo ensamblador del recurSO en el va a colocar el archivo. El controlador genera un mensaje de control para el PTA remoto indicando ese pedido; el mensaje es pasado al módulo de cartas al PT y también se crea un envíe para el PT sobre esta carta.

Después de esto se espera un mensaje de respuesta del PTA remoto el cual será pasado por el módulo de cartas del PT al controlador el que lo interpretará y por medio del módulo de comandos al usuario le avisa a este si se va a realizar su pedido.

El tercer evento del usuario "Ctransf" indica que se ejecute la acción, este evento como el anterior es pasado al controlador el cual genera un mensaje al PTA remoto. Una vez realizado esto se inicia la transferencia de datos por parte del PTA servidor, estos llegan en forma de cartas las que son chequeadas por el módulo de cartas del PT. Cada carta de las que conforman un archivo son pasadas al módulo ensamblador para que este cree el archivo en

el área de almacenamiento asignada para guardarlo. Una vez se ha terminado el proceso el controlador genera un mensaje de confirmación sobre la recepción del archivo y avisa al usuario del fin de la ejecución de su pedido, por medio del módulo de comandos al usuario.

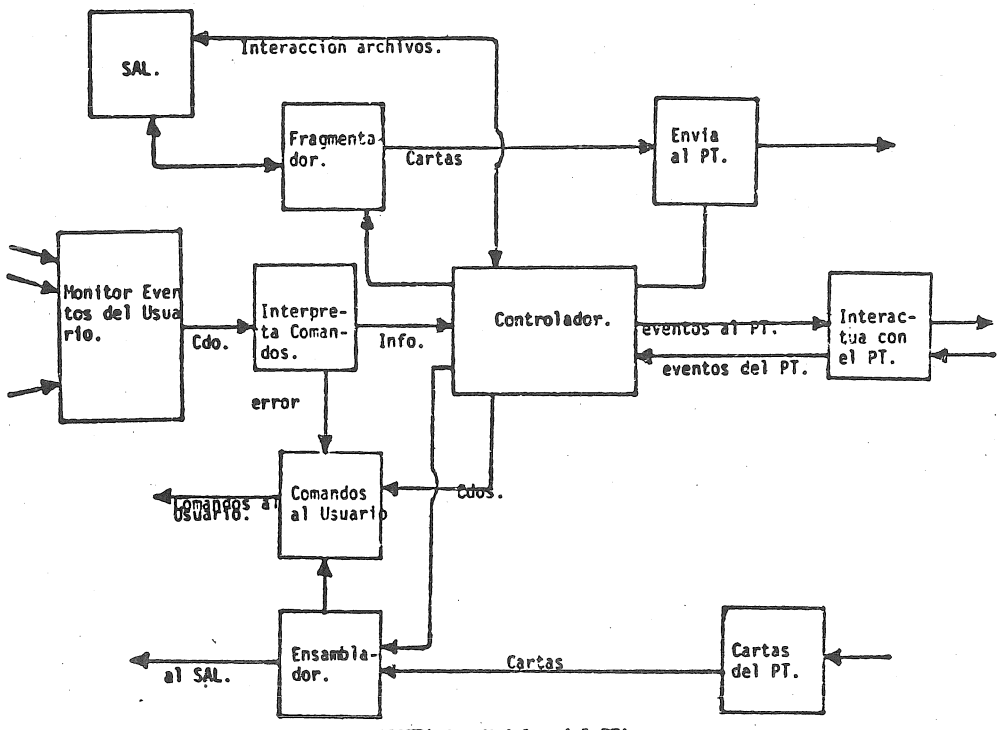


FIGURA 9 . Modulos del PTA.

5. CONCLUSIONES.

En este artículo se han descrito algunos de los factores más importantes que fueron tenidos en cuenta en el diseño de los protocolos de alto nivel (PT y PTA) para la red Uniandes.

En el desarrollo de las secciones se analizaron los aspectos más sobresalientes y su importancia dentro del diseño del protocolo. No obstante, la complejidad de dichos temas implica que sean estudiados con mayor profundidad.

El artículo no pretendía definir en su totalidad el diseño de un protocolo sino mostrar los tópicos que se deben tener en cuenta para dicho diseño.

De la arquitectura definida para red Uniandes solo se describieron el PTA y el PT porque son los protocolos básicos sobre los cuales se fundamentan los servicios y utilidades prestadas por una red de computadores. De el diseño, a la implantación de estos protocolos hay un alto grado de trabajo, que implica conocer el sistema operacional, el leguaje y el hardware de cada máqui

na.

No se debe pasar por alto decir que estos protocolos fueron diseñados para ser implantados en máquinas que permitan multiprogramación y comunicación entre procesos. Para equipos más pequeños las funciones y los mecanismos son más simples y menos complejos, por ejemplo en máquinas con monoprogramación no hya multiplexamiento, además las tablas y colas y demás estructuras de datos solo son necesarios para un solo proceso usuario.

R E F E R E N C I A S

(BOCH78) Bochmann G., Finite State Description of Communication Protocols., Danthine; Computer Network Protocols., Universite de Liege., 1978, pp F3-1-F3-10.

(BOCH80) Bochmann G., A General Transition Model for Protocols and Communication Services., IEEE: Transactions on Communications., Vol. COM 28, No. 4, Abril 1980, pp 643-650.

(BOCH80) Bochmann G., y Sunsshine C., Formal Methods in Communication Protocol Desing., IEEE: Transactions on Communications., Vol COM-28, No. 4, Abril 1980, pp 624-631.

(BOGG80) Boggs D., Shoch J. y Taft E., Pup: An Internetwork Architecture, IEEE: Transactions on Communication, Volumen COM-28, No. 4, Abril 1980, pp 612-613.

(CERF78) Cerf V., Mckenzie A., Scantlebury R., y Zimmermann H., Proposal for an Internetwoerk End-to-End transport Protocol., Computer network protocols., febrero 1978, pp H-5-H-25

(DANT80) Danthine A., Protocol Representation With Finite-State Models., IEE: Transactions on Communications., Vol. COM-28, No. 4, Abril 1980, pp 632-643.

(DOST76) Dostel J., Transmission control Protocol Specification., U.S. Departament of Commerce., Julio 1976.

(FLET78) Fletcher J., y Watson R., Mechanisms for a Reliable Timer-Based Protocol., Computer Network Protocol., febrero 1978., pp C5-1-C5-17.

(GIEN78) Gien M., A File Transfer Protocol., Danthine: Computer Network Protocol., Universite de Liege., 1978., pp D5-1-D5-7.

(HLPG77) High Level Protocol Group., A Network Independent File Transfer Protocol., Diciembre 1977.

(PAGA80) Paredes R., Gaitán L., Diseño de la arquitectura y los Protocolos de transporte y transferencia de archivos para la red local Uniandes., Universidad de los Andes., Bogotá., Diciembre 1980.

(WATS80) Watson R., Comparison Between TCP and Timer Based Protocol., ACM: Computer Protocol., 1980.

(WOOD79) Wood D., Holmgren S. y Skelton A., A cable-Bus Protocol Architecture, ACM: Sixth Data Communications Symposium, Noviembre 1979, pp 137-146.