29
29

UNIVERSIDAD NACIONAL AUTONOMA DE MEXICO

FACULTAD DE INGENIERIA

MONITOR GRAFICO DE RED COMO HERRAMIENTA EN LA ADMINISTRACION

T E S I S

Que para obtener el Título de:

INGENIERO EN COMPUTACION

p r e s e n t a n:

ALMA PATRICIA CRUZ MORALES
MARIA DEL CONSUELO SANCHEZ ESCOBEDO

Asesor de Tesis: Michael de Leo Gayol

México, D. F. 1997

TESIS CON
FALLA DE ORIGEN

Universidad Nacional Autónoma de México
UNAM
Biblioteca Central
Dirección General de Bibliotecas de la UNAM

UNAM – Dirección General de Bibliotecas
Tesis Digitales
Restricciones de uso

DERECHOS RESERVADOS ©
PROHIBIDA SU REPRODUCCIÓN TOTAL O PARCIAL

Todo el material contenido en esta tesis esta protegido por la Ley Federal del Derecho de Autor (LFDA) de los Estados Unidos Mexicanos (México).

El uso de imágenes, fragmentos de videos, y demás material que sea objeto de protección de los derechos de autor, será exclusivamente para fines educativos e informativos y deberá citar la fuente donde la obtuvo mencionando el autor o autores. Cualquier uso distinto como el lucro, reproducción, edición o modificación, será perseguido y sancionado por el respectivo titular de los Derechos de Autor.

ÍNDICE

Agradecemos y dedicamos el presente trabajo:

A Dios, por habernos permitido la culminación del presente, con la gente y las experiencias que nos llevan más a él.

A nuestro México, por mantener latente la esperanza de contar con instituciones como la Universidad Nacional Autónoma de México, la cual nos llena de orgullo por proporcionar diferentes perspectivas. Y particularmente a la Facultad de Ingeniería por la formación que nos ofreció.

Al Ing. Michael de Leo Gayol, por su apoyo y dirección a lo largo del desarrollo del presente trabajo y por la inconmensurable calidad humana que lo distingue.

A la Dirección General de Servicios de Cómputo Académico, por abrirnos la puerta a la aplicación y adquisición de conocimientos; a toda su gente que con su ayuda hizo posible el presente trabajo.

A la valiosa gente que conocimos a través de Bancomer, Red Uno y Centec, a las empresas mismas, por las facilidades proporcionadas y el apoyo brindado.

Dedico el presente trabajo:

A ti papá, porque me enseñaste un estilo de vivir como fórmula para alcanzar la felicidad.

A ti mamá, por ser ese ejemplo cierto, al que respeto, admiro, aprendo... ; por estar siempre conmigo.

A ustedes mis hermanos, Norma y Miguel, porque sin su presencia, cariño y apoyo mi vida no sería completa.

A mis sobrinos, por ser nueva esperanza.

A Consuelo, mi compañera de tesis y muchas cosas más.

A ustedes, Ara y Pam, porque son y serán siempre un magnífico sentimiento en mi.

Paty.

Dedico el presente trabajo:

A ustedes papá y mamá, con admiración e infinito cariño por ser mi mayor ejemplo de superación.

A mi abuelita Celina por la huella de lucha e inolvidable cariño que dejaste en mi.

A ustedes Memo y Juan, por que juntos encontramos un sentido diferente del vivir.

A ti Jaime con inmenso amor, por la felicidad de compartir cada momento de nuestra vida.

A ti Paty por que tu sencillez y gran amistad tienen un lugar muy especial en mi.

A cada uno de mis tíos por todas sus muestras de apoyo y cariño.

A mis primos y al pequeño Alan, esperándolos al final de esta meta.

A la familia Cruz Morales por su confianza y apoyo incondicional.

Consuelo.

INTRODUCCIÓN

El avance tecnológico trae consigo la transformación de muchas áreas de trabajo y el nacimiento de muchas otras, siendo la computación una de las mas sobresalientes ya que desde su inicio se ha visto en un constante y acelerado crecimiento por las ventajas que ésta ha traído a las distintas actividades humanas.

Actualmente conforme ha crecido el uso de equipo de cómputo han surgido diferentes requerimientos de hardware y software específicos para cada uno de ellos, encontrando que para el manejo de imágenes se requiere de mejores equipos que proporcionen mayor capacidad de procesamiento y despliegue; por otro lado, la manipulación de grandes volúmenes de datos que permitan a su vez ágiles consultas y operaciones, ha sido y será una de las necesidades primordiales a satisfacer.

Así como las necesidades anteriormente mencionadas, se identifica otra general para todas las áreas y mas importante aún: la necesidad de compartir información y recursos tanto de software como de hardware; se cuenta ya con redes que comunican una simple microcomputadora a redes mundiales, pasando por diversos medios de comunicación, es entonces cuando surgen nuevas preguntas a resolver: ¿Cómo saber el estado de los equipos involucrados?, ¿Cómo controlar cada uno de ellos?, ¿Cuánto se esta utilizando o subutilizando cada recurso?, etcétera. La búsqueda es ahora lograr la optimización del desempeño de cada función en cada elemento involucrado; ya que es así como el usuario final percibe los beneficios y la calidad del servicio que recibe de su equipo. Satisfacer esta necesidad del usuario final origina para el personal responsable del servicio de comunicación entre los equipos la necesidad indiscutible de lograr el punto de control de la red, que le garantice ofrecer un buen servicio.

Sin embargo, lograr este control que permita ofrecer niveles de servicio adecuados, así como la oportuna planeación y diseño del crecimiento de las redes no es cosa fácil, se requieren de diversos aspectos técnicos y administrativos a considerar para lograrlo. Así, el primer objetivo del presente trabajo es ofrecer un panorama general de los fundamentos para lograr la satisfacción de esta necesidad y como segundo objetivo proporcionar un "Monitor gráfico de red como herramienta inicial en la administración de redes", que resuelva la necesidad primaria de conocer el estado básico de los principales dispositivos de la red.

1. INTERNET

Podemos describir a la Internet como una red WAN, aunque en realidad no es una red en el sentido usual, es decir, no es propiamente un conjunto de computadoras interconectadas, sino más bien, el backbone de una red que interconecta una gran variedad de redes, que pueden ser públicas o privadas.

La Internet actual comenzó como un experimento de la Agencia de Proyectos de Investigación Avanzada (ARPA) en busca de una tecnología que inteconectara sus computadoras, lo cual hizo posible que en 1969 la red llegara a ser operacional bajo el nombre de ARPANET con nodos localizados en la UCLA, el Instituto de Investigación de Stanford (SRI), la Universidad de Santa Barbara y en la Universidad de Utah. Para 1971, la ARPANET se expandió en E.U. y en 1973 tenían conexiones en Europa.

En un principio el protocolo de comunicaciones empleado en ARPANET fue el Protocolo de Control de Red (NCP) que surge a finales de los años 70. Dicho protocolo no soportó el crecimiento de la red, razón por la cual fue sustituido por un Conjunto de Protocolos más robusto.

El conjunto de protocolos al que nos referimos es TCP/IP y fue en 1983 el año en el que el Departamento de Defensa de los E.U. ordena el empleo de TCP/IP. Más tarde en ese mismo año ARPA divide su red en 2:

- **ARPANET.** Red empleada para interconectar instituciones académicas, de investigación y desarrollo.
- **MILNET.** Red utilizada para el transporte de tráfico militar convirtiéndose en parte de la Red de Datos de la Defensa.

En 1969 AT&T desarrolló en sistema operativo UNIX y en 1976 se desarrolla UUCP (Copia de UNIX a UNIX) siendo distribuido con UNIX al año siguiente.

Para promover a los investigadores de las universidades la adopción y uso de nuevos protocolos, DARPA realizó una implementación a bajo costo. Para entonces, la mayoría de los departamentos de ciencia de las Universidades corrían una versión de sistema operativo disponible mediante la Distribución de Software Berkeley de la Universidad de California, comúnmente llamado UNIX Berkeley o UNIX BSD. Este sistema operativo llega a alcanzar el 90% de los sistemas en escuelas de informática.

El éxito de la tecnología de TCP/IP y de la Internet entre los investigadores de ciencias computacionales, condujeron a otros grupos a adoptarlo. La Fundación Nacional de Ciencia (NSF), al darse cuenta de la trascendencia que llegarían a tener las redes de comunicaciones, adoptó un papel activo en la expansión de la Internet TCP/IP para alcanzar el ámbito científico. Por lo anterior en 1985, NSF interconecta sus 6 centros nacionales de supercómputo, creando la NFSnet, la cual se conectó posteriormente a ARPANET. En 1986, NSF asigna recursos para la creación de redes regionales. Todas las redes fundamentadas en NFS usan protocolos TCP/IP y todas forman parte de Internet.

Ya en 1989 ARPANET deja de existir y se fusiona junto con la NSFnet en la actual Internet.

Actualmente la Internet se conforma de instituciones gubernamentales, educativas, de investigación, así como de corporaciones privadas de todo el mundo y actualmente se encuentra formada por alrededor de 70 países y cuenta con más de 3.5 millones de hosts en un estimado de 30,000 redes en el mundo.

Cabe mencionar que algunas de las aportaciones más significativas de TCP/IP, han sido las nuevas herramientas introducidas para organizar las capacidades de búsqueda y despliegue de información en la red. Algunas de estas herramientas son Archie, Gopher, World Wide Web (WWW) y fueron introducidas alrededor de 1991.

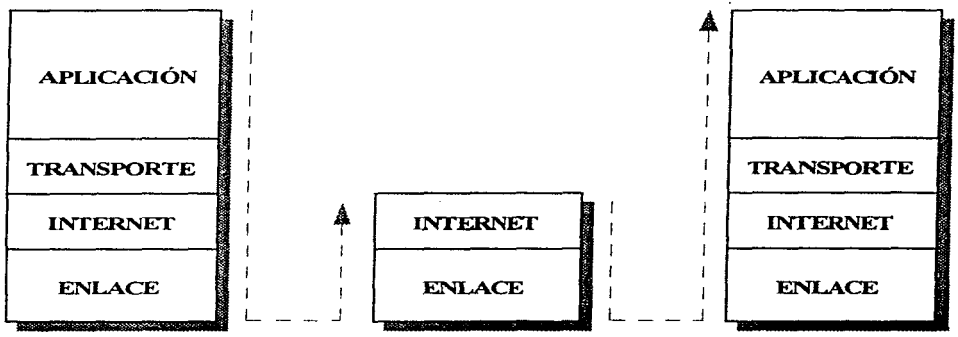
2. TCP/IP

Los Protocolos de Control de Transmisión/Protocolo Internet (TCP/IP) son un conjunto de distintos protoclos, los cuales definen algunos estándares para las diferentes tareas involucradas en la transmisión de datos. Los protocolos que lo conforman, fueron diseñados para ser independientes del hardware, sistema operativo y de la topología de la red.

En la arquitectura TCP/IP, la cual es descrita por un modelo jerárquico o modular, la implementación de cualquier aplicación, será en base al recorrido de cada uno de estos módulos, por ejemplo: para el intercambio confiable de datos se consideran procedimientos independientes. Mientras IP se encarga de unir diferentes tipos de redes en una internet, TCP proporciona una transferencia de datos confiable.

2.1 Intercambio de Información en el Modelo TCP/IP

Dado que se trata de un modelo jerárquico, la información enviada, deberá recorrer cada uno de los niveles TCP/IP. En el emisor, cada nivel tiene como función atender a la petición de servicio realizada por un protocolo de nivel superior (ULP) y así mismo proporcionar la petición necesaria al nivel inferior para el cumplimiento de la petición. En el receptor, el recorrido de los niveles es inverso, pero siempre manteniendo la correspondencia de atención y petición de servicios entre niveles adyacentes.



Intercambio de Información

2.2 El Modelo Cliente-Servidor

El modelo estándar para las aplicaciones de una red es el modelo *cliente-servidor*.

El término servidor se aplica a cualquier programa que ofrece un servicio que puede ser obtenido a través de la red. El servidor recibe una petición de servicio de algún cliente y regresa una respuesta a éste.

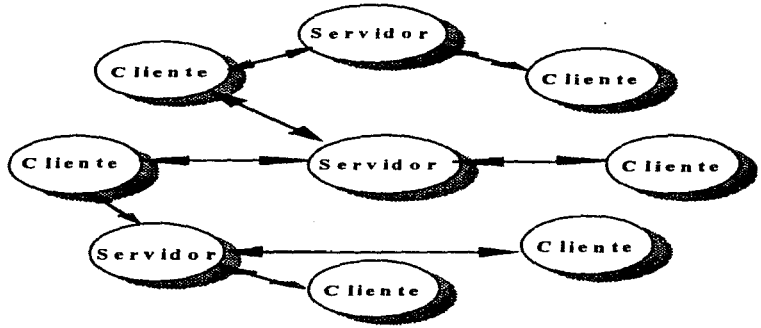
Cabe mencionar que un servidor puede ser cliente de algún otro servidor o viceversa.

Una descripción típica de la operación de este modelo es la siguiente:

1. El proceso servidor es inicializado en algún nodo. Después de este proceso, el servidor se mantiene en estado de espera para cualquier petición de servicio que le sea requerida por los clientes.

   Un servidor tiene las siguientes características:

   - Permanece activo todo el tiempo.
   - Utiliza solamente el puerto específico para realizar la acción requerida.
   - En la mayoría de los casos crea un proceso esclavo por cada petición que debe ser atendida



Modelo Cliente-Servidor

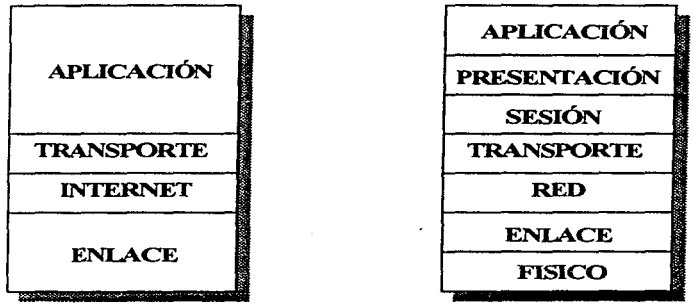
2. El cliente inicia un proceso que generalmente es inicializado por un usuario interactivo. El proceso cliente envía una petición a través de la red solicitando al servidor algún tipo de servicio.
   - Puede utilizar cualquier puerto disponible para hacer requisiciones.
   - Sólo está activo cuando hace la petición y recibe la respuesta a ésta.

Como ejemplos de servicios podemos mencionar los siguientes:

- Solicitud de hora y/o fecha.
- Impresión remota.
- Lectura o escritura de archivos.
- Solicitud del cliente para accesar al servidor.

2.3 TCP/IP contra OSI

El conjunto de protocolos TCP/IP influenció los estándares OSI, por lo que existe cierta equivalencia entre ambos modelos. Esta equivalencia se basa en las funciones asignadas a cada nivel, sin embargo, es importante mencionar que dicha equivalencia no está bien definida de un nivel a otro.

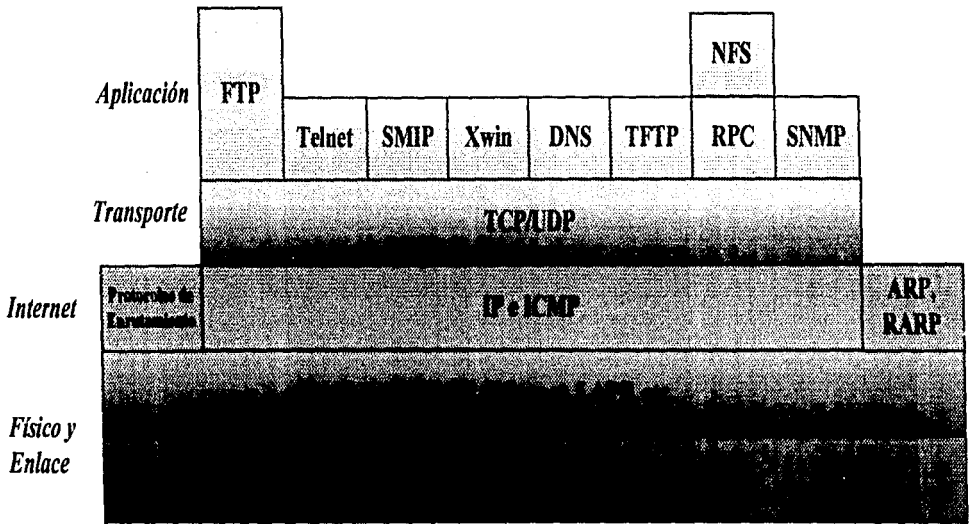


El Modelo TCP/IP contra el Modelo OSI

2.4 Cuadro Comparativo Entre TCP/IP y El Modelo OSI

Nivel TCP/IP	Nivel OSI	Características
Enlace	Físico y de enlace	Proporciona la capacidad de envío de datos dentro de los dispositivos de la red. Manejo de drivers. Control de acceso al medio. Enlaces. Señales Físicas.
Internet o IP	Red	Enrutamiento entre host. Responsable de los datos del origen al destino. No garantiza confiabilidad en el envío de datagramas.
Transporte	Transporte	Responsable de la comunicación de principio a fin. Proporciona funciones confiables de conexión y medición de tráfico Cuenta con un mecanismo que garantiza que los datos estén libres de errores.
Aplicación	Sesión Presentación Aplicación	Establece y termina la comunicación entre aplicaciones. Da un formato de transmisión que sea entendible para las aplicaciones entre sí. Proporciona interfaces para las aplicaciones finales.

TCP/IP PROTOCOLOS RELACIONADOS



3. IP

El Protocolo Internet fue diseñado para operar sin conexión, por lo que es posible que los datagramas se pierdan o lleguen fuera de secuencia entre dos estaciones finales, por lo que la responsabilidad del control de estos problemas se deja a las capas superiores.

La capa Internet tiene cuatro funciones básicas:

- Definición de la unidad básica de transmisión.
- Definición del esquema de direccionamiento.
- Fragmentación y reensamble de datos.
- Proporciona la entrega de paquetes.
- Enrutamiento.
- Especifica un conjunto de reglas para el manejo de errores.

IP trabaja con la filosofía del mejor esfuerzo, es decir, si por alguna razón no puede realizar la entrega de un datagrama, trata de avisar la razón del problema mediante el Protocolo de Control de Mensajes Internet (ICMP).

La capa Internet proporciona el servicio de fragmentación y reensamble de paquetes. La fragmentación se lleva a cabo en el enrutador, esta función es necesaria cuando la información debe viajar por redes que manejan MTUs (Unidad Máxima de Transmisión) de distintos tamaños. En caso de haberse realizado la fragmentación de paquetes, es función del host el reensamble de los mismos.

3.1 El datagrama IP

Las unidades de datos IP reciben el nombre de datagramas y se forman por un encabezado y una porción de datos. El datagrama IP se compone de los siguientes campos:

- Versión. Indica la versión de IP.
- Longitud del encabezado. Campo conocido como IHL (Internet Header Lenght),

Versión	HLEN	TOS	Longitud total	
Identificador			Banderas	Offset
TTL	Protocolo		Checksum del encabezado	
Dirección IP Origen				
Dirección IP Destino				
Opciones				Relleno
Datos.....				

Formato del Datagrama IP

- Tipo de Servicio. El campo TOS indica que protocolo particular de más alto nivel, debe manejar el datagrama actual. Mediante este campo pueden también asignarse niveles de importancia al datagrama. El campo TOS contiene 5 elementos cada uno de ocho bits.

Elementos del campo TOS

Bits	Descripción
0-3	Precedente.-Cuenta con 8 niveles de prioridad. 0 = Prioridad Normal. 7 = Prioridad Alta.
3	Retardo. 0 = Normal 1 = Bajo
4	Desempeño. 0 = Normal 1 = Alto
5	Confiabilidad. 0 = Normal 1 = Alta

Longitud Total. Indica la longitud total del paquete IP en bytes, incluyendo datos y encabezado.

Identificador. Contiene un entero que identifica al datagrama. Es un campo empleado para unir los fragmentos del datagrama.

Banderas. Especifican como puede fragmentarse el datagrama y cual será el último fragmento.

Tiempo de vida. Es un contador que indica cuando se descarta el datagrama.

Protocolo. Indica que protocolo de el siguiente nivel recibe los paquetes IP.

Encabezado del Checksum. Ayuda a asegurar la integridad del encabezado IP. Realiza un chequeo de errores en el encabezado.

Dirección fuente y destino. Indican los nodos que envían y reciben.

Opciones. Es un campo que no se usa en todos los datagramas y permite opciones tales como seguridad.

Offset del fragmento. Describe a que parte pertenece el fragmento dentro del datagrama original.

Relleno. Utilizado para dar al datagrama un alineamiento de 32 bits.

Datos. Contiene la información para el siguiente nivel.

3.2 Direccionamiento IP

Cada dispositivo dentro de una red TCP/IP, es identificado con una dirección única, las direcciones IP se forman de 32 bits, esto es, de cuatro octetos, representados por números decimales separados por puntos (Ejemplos: 132.248.161.5, 200.5.145.0). Por otra parte, las direcciones IP se componen de dos partes: un identificador de la red, y un identificador del nodo.

Dependiendo de el número de bits que se utilicen para el identificador de la red y del nodo, las redes se clasifican en 5 clases, siendo esta clasificación la siguiente:

Clase	Descripción	Rango Red	Rango nodo
A	Reservada para redes muy grandes, utiliza 8 bits para la identificación de red y 14 para los nodos.	1.0.0.0-126.0.0.0	0.0.0-255.255.255

Clase	Descripción	Rango Red	Rango nodo
B	Utilizada en redes medianas, utiliza 16 bits para red y 16 para nodos.	128.0.0.0-191.255.0.0	0.0-255.255
C	Empleada para redes pequeñas, utiliza 24 bits para red y 8 para nodos.	192.0.0.0-233.255.255.0	0-255
D	Reservada para multicast	244.0.0.0-239.255.255.255	No asignado
E	Reservada uso futuro	240.0.0.0-255.255.255.255	No asignado

Un caso especial de direccionamiento es el del empleo de subredes. Una red puede dividir su espacio de direcciones para definir múltiples redes lógicas, mediante la división de los bits reservados para la identificación del nodo. De este modo se cuenta con los bits necesarios para identificación de la subred y del nodo

Cabe mencionar también, que a la complejidad que puede presentarse por el manejo de direcciones numéricas, a estas se les asocia un nombre que puede ser recordado con mayor facilidad, dichos nombres y direcciones IP, son almacenados y mantenidos en un *"Servidor de nombres"* dentro de la red, que se rige una aplicación denominada Sistema del Dominio de Nombres (DNS).

3.2.1 Resolución de Direcciones

Para el intercambio de información entre los diferentes dispositivos dentro de una red es necesario conocer la dirección física de estos. Por lo anterior fue necesaria la implementación de algunos protocolos que permitieran realizar el mapeo entre direcciones IP y viceversa.

El protocolo conocido como Protocolo de Resolución de Direcciones (ARP) proporciona un mecanismo que permite a los dispositivos obtener la dirección física que necesitan.

El protocolo es muy simple, el transmisor envía un paquete ARP a través de la red solicitando la dirección física correspondiente a la dirección IP que necesita alcanzar; dicho paquete es enviado usando la dirección de difusión, por lo que cada dispositivo en la red recibirá la solicitud y al alcanzar la estación que reconozca esta dirección IP responderá con la dirección física correspondiente.

El proceso inverso también es necesario en estaciones que no cuentan con disco, y que por lo tanto requieren solicitar su dirección IP para ser reconocidas dentro de la red. El protocolo encargado de esto el Protocolo Inverso de Resolución de Direcciones (RARP). Este protocolo envía un paquete que contiene la dirección física de 48 bits, del dispositivo que solicita una dirección IP. Debe existir en la red, al menos un servidor RARP, pues este es el que responderá y proporcionará la dirección IP solicitada.

3.3 Enrutamiento IP

El enrutamiento, es una función muy importante de IP. En redes grandes, los enrutadores IP intercambian información que mantienen actualizadas las tablas de enrutamiento, esto puede hacerse mediante un protocolo de enrutamiento. Actualmente predomina el uso de RIP (Protocolo de Información de Enrutamiento) aun cuando OSPF (Primer Ruta Abierta más Corta) tiene mayor aceptación.

Desde la concepción de IP, se diseño el esquema de direccionamiento para mejorar la eficiencia del enrutamiento interno y externo de las redes. El modelo Internet de enrutamiento, particiona una red muy grande en muchas regiones de enrutamiento autónomas. Estas son llamadas Sistemas Autónomos. Un protocolo empleado dentro de un sistema autónomo es llamado Protocolo de Enrutamiento Interno (IGP) y un protocolo que realiza la comunicación entre diferentes sistemas autónomos es conocido como Protocolo de Enrutamiento Externo (EGP).

Un datagrama sigue una ruta constituida por una secuencia de saltos.

Una ruta fuente, que contiene una lista de saltos, puede incluirse en un encabezado IP, sin embargo el uso de rutas fuente no es la norma. Generalmente un datagrama es enrutado eligiendo su próximo salto en cada enrutador por el que pase en su trayecto. Cuando un enrutador recibe un paquete, este verifica la dirección destino e intenta asociar esta dirección con algún salto siguiente.

El enrutamiento del salto siguiente es flexible y robusto. Un cambio permanente en la topología de una red puede configurarse generalmente actualizando unos cuantos enrutadores. Estos pueden programarse para informarse entre ellos mismos de los cambios temporales y permanentes en la red y pueden conmutar el tráfico de manera dinámica por rutas alternativas cuando sea necesario, empleando algún protocolo de enrutamiento dinámico.

3.3.1 Mecanismos de Enrutamiento

IP enruta los datagramas haciendo uso de la dirección destino y el algoritmo general de enrutamiento. Dicho algoritmo emplea tres parámetros:

- Dirección IP destino.
- Máscara de red o subred.
- Tabla de enrutamiento.

La máscara de la subred permite conocer cuando dos hosts pertenecen a la misma red o subred, pudiendo así realizar la transmisión directamente.

Una tabla de enrutamiento comúnmente se crea incluyendo la dirección IP origen y una dirección IP correspondiente al salto siguiente.

El enrutamiento IP utiliza un concepto llamado Métrica, el cual se basa en la elección de diferentes parámetros para la asignación de un determinado valor a cada ruta, dicho valor se conoce como costo de la ruta. El enrutador consulta su tabla de enrutamiento, compara la dirección del encabezado del datagrama y revisa el valor del costo. Si las direcciones coinciden, selecciona la ruta del menor costo y el enrutador la utiliza para determinar el puerto de salida, en caso contrario el datagrama es descartado y se envía a la fuente un mensaje de destino inalcanzable.

IP proporciona dos opciones para el enrutamiento, el cual permite a un ULP determinar la forma en que un se enrutarán los datagramas. Dicho ULP puede pasar una lista de direcciones internet al módulo IP para conocer los nodos intermedios que serán transitados durante el enrutamiento de datagramas al destino final. Las opciones de enrutamiento que proporciona son:

- Enrutamiento Fuente Libre.- Proporciona la posibilidad a los módulos IP de dar saltos intermedios alternativos durante el transporte de los datagramas.

- Enrutamiento Fuente Estricto.- El datagrama viaja estrictamente por las redes indicadas en la lista. Si el enrutamiento estricto no puede ejecutarse, el host origen es notificado con un mensaje de error.

Un enrutador, al recibir un datagrama detecta si se trata de un PDU Internet y al ser detectado pasa a la rutina de chequeo de encabezado realizando las siguientes validaciones:

- Validación de la longitud del encabezado.
- Validación del número de versión.
- Validación de la longitud del mensaje.
- Validación del encabezado del checksum.
- Verifica que el campo de tiempo de vida sea diferente de cero.

Si estas validaciones no son exitosas, el datagrama es descartado, de lo contrario se examina la dirección destino

3.3.2 Tablas de Enrutamiento

El enrutamiento en un host o enrutador se lleva a cabo, consultando una tabla de enrutamiento. Esta tabla relaciona la dirección del destino con la dirección del enrutador usado en el siguiente salto. No existe un formato único para las tablas de enrutamiento, pero una tabla de enrutamiento típica contiene:

- Dirección de una red, subred o host destino.
- Dirección IP de un enrutador indicado como siguiente salto.
- Interface de red a ser usada.
- Máscara de la subred para esta interface.
- Distancia al destino o métrica.
- Número de segundos desde la última actualización de la ruta.

Para que las tablas de enrutamiento sean pequeñas, la mayoría o en ocasiones todas las entradas identifican redes o subredes destino. En ocasiones algunos hosts importantes son incluidos.

En un equipo UNIX una tabla de enrutamiento puede tener la siguiente forma:

Destination	Gateway	Flags	Refcnt	Use	Interface
default	132.248.204.254	UG	2	4104850	le0
148.205.2.1	132.248.204.254	UGHD	0	1789	le0
127.0.0.1	127.0.0.1	UH	3	101860	lo0
148.205.210.6	132.248.204.254	UGHD	0	1641	le0
132.248.0.0	132.248.204.1	U	36	4973308	le0
148.225.0.0	132.248.204.151	UGD	0	1437	le0
148.205.0.0	132.248.204.150	UGD	0	513	le0
200.10.143.0	132.248.204.148	UGD	0	474	le0

3.4 Relación con el modelo OSI

El protocolo del modelo OSI equivalente a IP es conocido como Protocolo sin Conexión del Nivel Red (CNLP), cuya arquitectura y funcionamiento es muy similar a IP.

Las unidades de datos CNLP se conforman al igual que en IP por un encabezado y una porción de datos, donde el encabezado contiene la misma clase de información que el encabezado IP. La estructura de las unidades de datos CNLP y algunas diferencias con IP se muestran a continuación:

CNLP (ISO 8473)	IP
El receptor conoce la cantidad total de datos esperados mediante el campo de longitud total.	El receptor conoce la longitud total hasta que llega el último fragmento.
El campo de tipo de servicio es un campo opcional dentro del campo QOS.	Requiere del campo Tipo de Servicio.
Las direcciones OSI conocidas como Puntos de Acceso al Servicio de Red (NSAP), son de longitud variable y	Las direcciones se conforman por 32 bits y existe un solo formato para

CNLP (ISO 8473)	IP
existen además diferentes formatos de direcciones OSI. No contiene un campo de protocolo ya que OSI identifica a que protocolo del siguiente nivel se enviarán los datos.	estas. Debe indicar que protocolo del siguiente nivel manejará los datos.

3.5 Ventajas de IP

El desempeño de una internet depende de los recursos disponibles en sus hosts y enrutadores y de que tan eficientemente se usan estos recursos.

Los recursos a los que nos referimos son :

- CPU.
- Buffer.
- Ancho de banda.

Comparado con OSI en el manejo de estos recursos, IP presenta las ventajas y desventajas que se muestran en la tabla siguiente:

Ventajas y Desventajas de IP

RECURSO	VENTAJAS	DESVENTAJAS
CPU	Pequeña sobrecarga en el procesamiento de datagramas. Análisis íntegro del encabezado. Agilidad en las búsquedas de siguiente salto en las tablas debido al manejo de la dirección de 32 bits. No se necesita la elaboración de software para manejar los timeouts y retransmisiones que	IP necesita enrutamiento para cada salto. El tiempo de enrutamiento depende de la sofisticación del algoritmo empleado. Requiere mayor poder de procesamiento debido a la contabilización de condiciones de producción y retardo para

RECURSO	VENTAJAS	DESVENTAJAS
	necesita un nivel de red con conexión.	el balanceo de tráfico.
Buffer	Queda disponible para uso inmediato una vez que sea transmitido el datagrama, aunque el host debe reservar una parte del buffer mientras reensambla un datagrama fragmentado	Pueden existir problemas de congestionamiento cuando un enrutador conecta redes mas rápidas, pudiéndose llenar el buffer del enrutador.
Ancho de banda	Los datagramas son transmitidos tan pronto como se encuentre disponible el ancho de banda. No existe perdida de vida a reservas de ancho de banda para trafico especifico o por espera de ACKs como en el caso de un protocolo orientado a conexión. Existen protocolos IP de enrutamiento capaces de direccionar el tráfico por múltiples rutas y pueden elegir rutas de manera dinámica de modo que pueden evitar congestionamientos.	Existe una pequeña sobrecarga debido a los mensajes de control. Bajo gran carga los datagramas comienzan a cumularse formando colas en los enrutadores. De este modo, el tiempo de envío de la fuente al destino se incrementa y algunos datagramas son descartados. Esta puede causar la retransmisión de datagramas de TCP, incrementando la carga y decrementando el desempeño efectivo.

3.6 La siguiente generación de IP

La incertidumbre sobre si IP cubrirá las demandas en el siglo XXI aumenta cada día. El esfuerzo por diseñar un protocolo IP de siguiente generación, denominado IPng (La siguiente generación de IP) para IP versión 6, también denominado IPng6.

Una de las mejoras esperadas para IPng es un espacio de direcciones expandido. Se anticipa que el espacio de direcciones actual se agotará dentro

de algunos cuantos años, si el crecimiento continua como hasta el momento. IPng manejará direcciones de 128 bits, por lo que las direcciones actuales de 32 bits podrán seguir utilizándose, así como estructuras que están por definirse. Así mismo incluirá nuevas características para hosts portátiles, asignación de rutas IP de manera dinámica, procesamiento en tiempo real y mayor seguridad.

4. ICMP

El protocolo de Control de Mensajes Internet (ICMP), es una parte integral de la capa Internet del modelo TCP/IP y por tal razón debe implementarse en cada módulo IP. ICMP tiene como función, el reportar los errores producidos durante el procesamiento de datagramas mediante la generación de mensajes de estado.

Los mensajes ICMP son enviados cuando:

- El tiempo de vida ha expirado.
- El destino o algún enrutador está congestionado.
- El destino o algún enrutador es inalcanzable.
- Se desean realizar pruebas mediante el envío de la función eco.

El propósito de los mensajes de control, es proporcionar el conocimiento de problemas en el medio ambiente de comunicación, más no hacen confiable a IP. La confiabilidad es responsabilidad de un protocolo de más alto nivel tal como TCP o algún protocolo del nivel de aplicación.

ICMP notifica al host si el destino es inalcanzable, si se ha excedido el tiempo de vida del datagrama o el encabezado IP tiene error.

Por otra parte, estos mensajes son enviados solamente en el manejo de errores en el fragmento cero.

4.1 Formato de los Mensajes ICMP

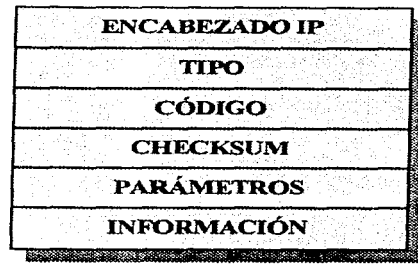
Los mensajes son enviados empleando los 8 primeros octetos del datagrama que ocasionó el error. El primer octeto de la porción de datos es un campo de tipo ICMP, el cual es empleado para indicar el uso del protocolo. Los mensajes tienen la siguiente estructura:

El campo denominado tipo, define la clase de mensaje.

El campo código, describe el tipo de error o el estado.

Checksum es un campo que sirve para calcular el complemento a uno de los dieciseis bits del mensaje ICMP. El campo de parámetros depende del

tipo de error que se envíe.



Formato de los Mensajes ICMP

4.2 Procedimiento para el Reporte de Estado y Error

Los servicios ofrecidos por ICMP para el reporte de estado y error, así como el código empleado para identificar el tipo de mensaje se muestran en la tabla siguiente:

Código	Tipo de Mensaje
0	Respuesta a la petición de eco.
3	Destino inalcanzable.
4	Disminución de flujo.
5	Redirigir.
8	Petición de eco.
11	Tiempo excedido.
12	Problema de parámetros.
13	Petición de marca de tiempo.
14	Respuesta a la petición de marca de tiempo.
15	Petición de información.

Código	Tipo de Mensaje
16	Respuesta a la petición de información.
17	Petición de máscara.
18	Respuesta a la petición de máscara.

Tiempo Excedido. Este servicio es ejecutado por un gateway cuando el tiempo de vida del datagrama IP expira y este es descartado. El servicio de tiempo excedido, es invocado también si el tiempo asignado para el reensamble de datagramas expira antes de concluir la operación.

El campo código. Dentro de la estructura de los mensajes ICMP puede tomar alguno de los siguientes valores:

0 = Tiempo de vida excedido.

1 = Tiempo de ensamble de fragmentos excedido.

Problema de Parámetros. El host o gateway destino pueden invocar este servicio si tienen problemas al procesar alguna parte del encabezado IP. Generalmente, esto ocurre cuando un campo es ilegible, impidiendo que el datagrama sea procesado, por tal razón debe ser descartado. El mensaje ICMP contiene un campo apuntador cuyo valor apunta al octeto que ocasionó el problema en el encabezado del datagrama original.

Los valores del campo código pueden ser los siguientes:

0 = Apuntador utilizado.

1 = Problema con las opciones de TOS.

Destino inalcanzable. Este servicio es usado por un gateway o un host destino. Es invocado si se encuentra algún problema para alcanzar la red especificada en la dirección IP destino. También puede usarse por un host destino si no se tiene disponible un protocolo de más alto nivel o si no se encuentra algún puerto disponible. El campo de código del encabezado ICMP se codifica de la siguiente manera:

0 = Red inalcanzable.

1 = Host inalcanzable.

2 = Protocolo no disponible.

3 = Puerto no disponible.

4 = Fragmentación necesaria.

5 = Falla en ruta fuente.

El gateway puede enviar los códigos 0, 1, 4, y 5; mientras que los códigos 2 y 3 puede enviarlos el host.

Disminución de flujo. Este servicio es una forma primitiva de control de flujo y congestión en un gateway cuando el espacio en buffer de la máquina es insuficiente para ordenar los datagramas de entrada. Si el datagrama es descartado, el gateway puede mandar este mensaje al host que originó el datagrama. El host destino puede usar el servicio del mensaje de disminución de flujo, si los datagramas llegan muy rápido y no se puede atender a la petición de procesamiento. En operaciones reales este servicio actúa como una notificación de reducción en el número de datagramas que son transmitidos; por lo cual, este servicio realiza el control de flujo para la internet.

Un gateway tiene la opción de enviar el mensaje de disminución de flujo para cada datagrama que es descartado, además de recibir este mensaje se le pide al host que reduzca el tráfico de datagramas. ICMP no cuenta con un mensaje que indique el restablecimiento de la transmisión, por lo que el control de flujo es reiniciado cuando el transmisor deja de recibir mensajes de disminución de flujo. Generalmente esto quiere decir que el host probablemente pueda incrementar el tráfico gradualmente hasta que este corriendo a una velocidad de transmisión adecuada o hasta que reciba otro mensaje.

Es prudente emitir una señal de control de flujo, antes de exceder la capacidad de la máquina. Como consecuencia, el mensaje de disminución de flujo puede ser emitido cuando se está alcanzando el límite de capacidad de la máquina.

Petición de eco y respuesta al eco. Es una herramienta útil para determinar el estado de una internet. Puede ser enviado a cualquier dirección IP, tal como un gateway. El gateway debe devolver una respuesta al origen y en caso de existir algún problema no se obtiene respuesta a la petición. Un servicio que emplea el eco es el Detector de Paquetes Internet conocido como PING, el cual incluye funciones tales como: identificación de dispositivos físicos, intervalos de tiempo para alcanzar un punto determinado, detección de dispositivos habilitados, etc.

Redirigir. Es un servicio invocado por un gateway, enviando un mensaje al host fuente. Es empleado para proporcionar al host información para la administración. Este mensaje indica la existencia de una mejor ruta, lo que implica que el host envie su tráfico a otro gateway. Si los datagramas emplean la opción de enrutamiento fuente, no se enviará el mensaje de redirigir a pesar de existir una mejor ruta. El gateway puede generar alguno de los siguientes valores, mediante el campo de código.

0 = Redirigir datagramas hacia la red.

1 = Redirigir datagramas hacia el host.

2 = Redirigir datagramas para el tipo de servicio y red.

3 = Redirigir datagramas para el tipo de servicio y host.

Petición de Marca de Tiempo y Respuesta a la Petición de Marca de Tiempo. Es utilizado por hosts y gateways para determinar el retardo en el envío de tráfico a través de la red. La unidad de datos ICMP contiene tres valores para la marca de tiempo:

- Originar la Marca de Tiempo. Tiempo en el que el origen procesó el mensaje por última vez antes de enviarlo.
- Recibir Marca de Tiempo. Tiempo en el que se recibió el mensaje y se comenzó a procesar.
- Transmitir Marca de Tiempo. Es el tiempo en que el eco proceso por última vez el mensaje en el envío.
- Petición y Respuesta de Información. Este mensaje sirve para que un host conozca la red a la que pertenece.

- Petición y Respuesta de la Máscara. Empleado por un host para obtener una máscara.

4.3 Relación con el Modelo OSI

OSI envía mensajes de error en un formato muy similar al empleado por ICMP. El encabezado contiene una bandera que indica que un PDU contiene un mensaje de error. OSI necesita incluir más información que IP, ya que están involucrados los encabezados de tres niveles en vez de uno.

Las condiciones que pueden ser causa de un error dentro de OSI en un PDU son:

- Dirección destino inalcanzable
- Dirección destino desconocida.
- Problema en enrutamiento fuente.
- Tiempo de vida excedido en tránsito.
- Tiempo de vida excedido durante reensamble.
- No es posible reensamblar.
- Opción no soportada.
- Versión de protocolo no soportada.
- Opción de seguridad no soportada.
- Grabación inválida de la opción de enrutamiento.
- Error en el procedimiento del protocolo.
- Checksum incorrecto.
- PDU descartado debido a congestión.
- Error de sintaxis en el encabezado.
- Segmentación necesaria pero no permitida.
- PDU incompleto.

- Demasiados datos de usuario.

Al igual que IP, el nivel de red OSI no define acción alguna a ejecutarse en este nivel como resultado del reporte de un error. El reporte debe pasarse a una entidad apropiada de nivel superior.

5. TCP

Como se ha visto anteriormente, el protocolo IP, no fue diseñado para recuperar ciertos problemas en la red, ni para garantizar la entrega de tráfico. Por ejemplo, IP fue diseñado para descartar los datagramas que son desactualizados, cuando exceden el número de saltos permitidos.

Ciertas aplicaciones requieren asegurar que todos los datagramas lleguen a salvo a su destino. Más aún, el usuario que transmite puede requerir saber que el tráfico ha llegado a su destino.

El servicio de transporte confiable de datos que proporciona TCP/IP es definido por el Protocolo de Control de Transmisión, mejor conocido como TCP, el cual es un protocolo con conexión (UDP es un protocolo sin conexión y no proporciona esto servicios).

El trabajo de TCP puede ser muy complejo, este debe ser capaz de satisfacer un amplio rango de requerimientos de aplicaciones e igualmente importante, debe ser el acoplarse a un ambiente dinámico dentro de una internet. Debe establecer y manejar las sesiones (conexiones lógicas), esto significa que TCP debe mantener un registro de las actividades de los usuarios para soportar la transferencia de datos de los usuarios a través de la red.

5.1 Principales características de TCP

TCP es un protocolo orientado a la conexión, es decir, mantienen información del flujo de datos de cada usuario en el módulo TCP y es responsable de la transferencia de información de principio a fin en la red. El protocolo de control de transmisión proporciona los siguientes servicios:

- Administración de datos con conexión.
- Transferencia confiable de datos.
- Funciones de empuje.
- Reordenamiento.
- Control de flujo mediante ventanas deslizables.
- Multiplexaje.

- Transmisión full dúplex.
- Precedencia y seguridad
- Terminación exitoso de sesión

Puesto que TCP es responsable de la transferencia confiable de información, hace uso de mecanismos tales como reconocimientos y asignación de un número de secuencia a cada paquete transmitido. El módulo receptor emplea una rutina de checksum para verificar los datos transmitidos. Si los datos pasan dicha verificación, TCP envía un reconocimiento (ACK) positivo al módulo transmisor; en caso contrario, si existe algún error, producido durante la transmisión, el módulo receptor, descarta los datos y emplea un número de secuencia par informar al módulo transmisor sobre la existencia de algún problema.

TCP utiliza contadores para asegurar que el intervalo de tiempo no es excesivo antes de tomar medidas correctivas, las cuales pueden consistir en el envío de un ACK o la retransmisión de los datos.

Al recibirse los datos de un protocolo del nivel superior (ULP) en un ambiente orientado a flujo de caracteres. Los protocolos con esta orientación son diseñados para enviar caracteres individuales y no bloques, tramas, datagramas u otro tipo de paquete. El ULP envía un flujo, byte por byte. Cuando estos llegan al nivel de TCP, los bytes son agrupados en segmentos TCP, los cuales son posteriormente pasados a IP o a algún otro protocolo del nivel de red, para realizar la transmisión al destino. La longitud de los segmentos es determinada por TCP o bien por la persona que implementa el sistema, con lo anterior se deduce que la naturaleza de TCP permite el uso de segmentos variable , por lo que no puede ser empleado con algunas aplicaciones que manejan bloques de tamaño fijo.

Otra función de TCP, es la realización del chequeo de información duplicada, descartando la que sea redundante. La duplicación de información puede presentarse cuando el módulo receptor no maneja el envío de ACKs controlado por tiempo, además la generación de retransmisiones innecesarias se debe también al esquema inclusivo de confirmación que emplea TCP, el cual consiste en la confirmación de cada segmento mediante un ACK en el orden estricto que debe ocupar cada uno de estos.

Gracias a su capacidad de transferencia de flujo, TCP puede soportar el

concepto de función de "empuje", la cual tiene como finalidad asegurar que se realice la transmisión de todos los datos que se han pasado al nivel inferior. Para hacer esto, el ULP envía un comando "send" a TCP con un parámetro de empuje igual a uno; entonces, TCP debe convertir el tráfico de su buffer en segmentos y reenviarlo a su destino. Esta función puede ejecutarse mediante el cierre de la conexión.

Otra función de TCP consiste en el reordenamiento de segmentos en el módulo receptor, en caso de ser necesario. Este reordenamiento se realiza empleando el número de secuencia.

Por otra parte, el módulo receptor tiene la capacidad de control de flujo de los datos del transmisor, lo cual es sumamente útil evitando así exceder la capacidad del buffer y saturar la máquina receptora. Dicho control de flujo se basa en el manejo del concepto de "ventana", la cual tiene asignado un valor que indica el número de bytes a transmitir. El valor de la ventana se envía desde el receptor al transmisor, y este transmite entonces el número de bytes especificado, después de lo cual la ventana se detiene y se detiene el envío de datos.

TCP permite multiplexar sesiones de diferentes usuarios en una misma máquina a los niveles superiores, tarea que logra asignando convenciones para los puertos y sockets en los módulos IP y TCP. Cada usuario es identificado por la dirección del puerto por el que se conecta. La dirección del puerto se concatena con la dirección IP para formar un socket. Se tiene un socket para la transmisión y otro para la recepción.

Socket de Transmisión = Dirección IP fuente + Número del puerto fuente

Socket de Recepción = Dirección IP destino + Número del puerto destino

En la Internet se han publicado los números para los procesos más frecuentes de los niveles de aplicación, y existen valores posteriores a 255 para uso privado.

Ejemplos de asignación de puertos:

Puerto	Nombre	Descripción
7	ECHO	Echo

Puerto	Nombre	Descripción
21	FTP	Transferencia. de Archivos
23	TELNET	TELNET
25	SMTP	Protocolo de Transferencia simple

Debido a que los puertos pueden ser utilizados simultáneamente por más de una conexión, los usuarios pueden compartir los recursos de un puerto mediante el multiplexaje. Por otra parte, la transmisión full-dúplex entre dos entidades TCP, permite la transmisión simultánea en ambos sentidos sin tener que esperar una señal de retorno, como sucede en una comunicación semi-dúplex.

El Protocolo de Control de Transmisión tiene también la capacidad de proporcionar niveles de seguridad y precedencia (nivel de prioridad para la conexión), sin embargo, no todos los productos TCP lo implementan aún cuando el estándar lo especifica. Proporciona además la conclusión exitosa de los circuitos virtuales (conexión lógica), con lo cual se asegura que todo el tráfico sea reconocido antes de eliminar el circuito virtual.

5.2 Inicio Pasivo y Activo.

En los puertos TCP, se permiten dos formas de establecer una conexión, estas formas son:

- Inicio Pasivo. Permite al ULP decir a TCP y al sistema operativo del Host, que debe esperar peticiones de conexión del sistema remoto. Al recibirse esta petición, el sistema operativo del host asigna un número de puerto desocupado para la conexión.

- Inicio Activo. En este modo, el ULP designa específicamente otro socket a través del cual se establece la conexión. Generalmente, el inicio activo es enviado a un puerto de inicio pasivo, para establecer un circuito virtual.

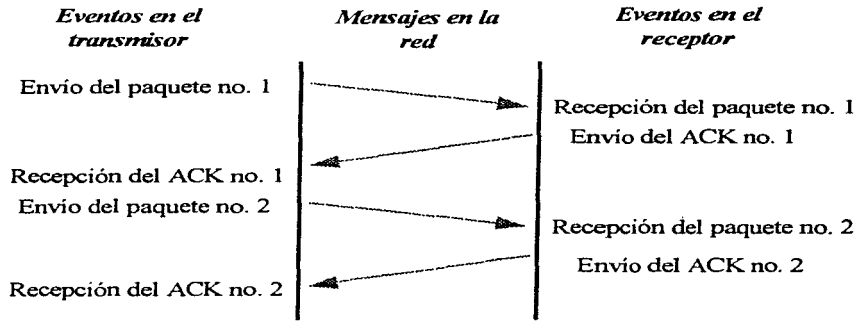
5.3 Confiabilidad

La entrega confiable de información se lleva a cabo mediante el empleo de una técnica fundamental conocida como "Reconocimiento Positivo con

Retransmisión". Esta técnica requiere que el receptor se comunique con la fuente que envía los mensajes confirmando la recepción de los datos a medida que estos son recibidos. El transmisor guarda un registro de cada paquete enviado y espera la confirmación de la recepción mediante un ACK antes de enviar el siguiente paquete. Por otra parte este transmisor inicializa un contador cuando realiza el envío, retransmitiendo el paquete si el contador expira antes de recibir el ACK.

La siguiente figura muestra como transfiere los datos el protocolo de reconocimiento más simple.

Protocolo que emplea ACK positivo con retransmisión, en la que se espera un ACK en el transmisor por cada paquete enviado.

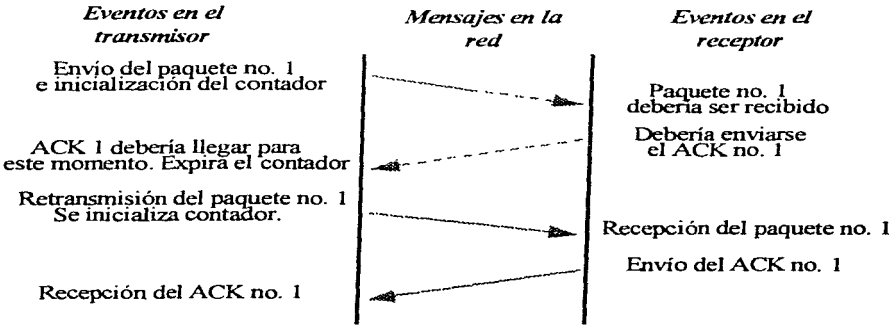


El problema final de la confiabilidad se presenta cuando existe duplicación de paquetes. Dicha duplicación puede ocurrir también cuando las redes presentan grandes retardos que pueden causar esta retransmisión prematura. Este problema debe resolverse cuidadosamente, pues puede producirse duplicación tanto de paquetes como de ACKs. Generalmente, los protocolos confiables detectan paquetes duplicados, asignando a cada uno un número de secuencia y guardando en el receptor un registro de los números de secuencia recibidos.

Para evitar confusión causada por retardo o ACKs duplicados, los protocolos con ACK positivo regresan el número de secuencia en los reconocimientos de forma que el receptor puede asociar correctamente ACKs con paquetes.

Este tipo de protocolos gasta gran cantidad del ancho de banda ya que deben esperar para enviar un nuevo paquete hasta que reciben el ACK del paquete previo.

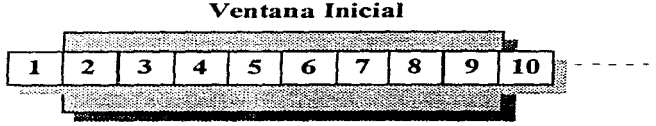
Tiempo fuera y retransmisión cuando se pierde un paquete



5.4 La Ventana Deslizable

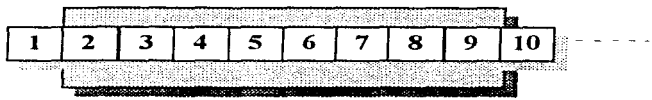
El concepto de ventana deslizable, hace que la transmisión de flujo sea más eficiente. La técnica de la ventana deslizable es una forma más compleja de ACK positivo y retransmisión. Los protocolos de ventana deslizable usan mejor el ancho de banda, ya que permiten al transmisor enviar múltiples paquetes antes de esperar un ACK. Estos protocolos ponen una ventana pequeña en la secuencia y transmiten todos los paquetes que se encuentren dentro de la ventana.

Protocolo de ventana deslizable con ocho paquetes en la ventana.



La ventana se desliza de modo que el paquete número nueve puede enviarse cuando se ha recibido el ACK correspondiente al paquete número uno.

Ventana Deslizable

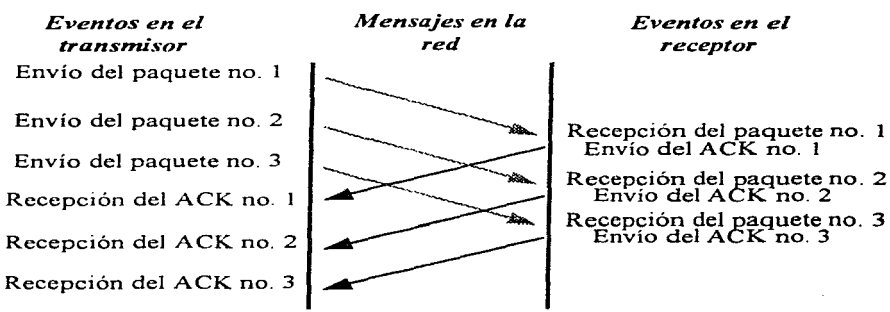


El desempeño de los protocolos de ventana deslizable depende del tamaño de la ventana y la velocidad a la que la red acepta los paquetes.

Con un tamaño de ventana igual a uno, un protocolo de ventana deslizable es igual a un protocolo de reconocimiento simple. Incrementando el tamaño de la ventana, es posible eliminar completamente el tiempo ocioso de la red.

Ya que un protocolo de ventana deslizable bien sincronizado mantiene la red saturada con paquetes, se obtiene una producción substancialmente mayor que en un protocolo de reconocimiento simple.

Envío de 3 paquetes usando un protocolo de ventana deslizable. *La clave del concepto consiste en que el transmisor puede enviar todos los paquetes en la ventana sin esperar confirmción de los paquetes recibidos en el otro extremo.*



5.4.1 Ventana de Tamaño Variable y Control de Flujo.

TCP permite que el tamaño de la ventana varíe en el tiempo. Cada ACK especifica cuantos octetos han sido recibidos, contiene un anuncio de ventana que indica cuantos octetos de datos adicionales está preparado para aceptar el receptor, es decir, puede considerarse que el anuncio de la ventana especifica el tamaño actual del buffer del receptor.

En respuesta a un aviso de incremento de ventana, el transmisor incrementa el tamaño de su ventana deslizable y procede a enviar octetos que no han sido reconocidos. En cambio para un aviso de disminución de ventana, el transmisor decrementa el tamaño de su ventana y detiene el envío de octetos fuera del límite.

La ventaja de usar una ventana de tamaño variable es que proporciona control de flujo al igual que transferencia confiable. Si el buffer del receptor no soporta más paquetes por lo que envía un pequeño aviso de ventana. En caso extremo, el receptor anuncia una ventana de tamaño cero para detener toda transmisión. Después, cuando se libera espacio en el buffer, el receptor anuncia una ventana de tamaño no nulo para disparar nuevamente el flujo de datos.

El mecanismo de control de flujo es esencial en una internet, ya que en esta se comunican dispositivos de diversas velocidades y capacidades.

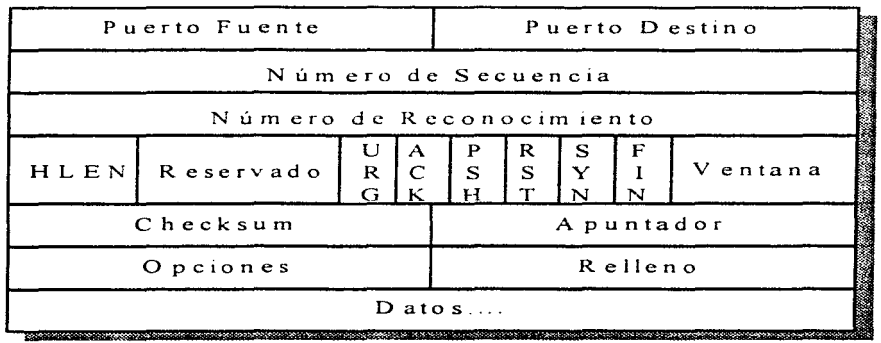
Podemos encontrar dos problemas en cuanto al control de flujo se refiere:

- Es necesario que los protocolos en la internet cuenten con un control de flujo de punto a punto (entre la fuente y el destino).
- Estos protocolos necesitan un mecanismo de control de flujo que permita a los sistemas intermedios controlar una fuente que envíe más tráfico del que puede soportar.

A la condición de sobrecarga de las máquinas intermedias se le conoce como "congestión". El mecanismo de Control Congestión, es empleado para solucionar esta condición de sobrecarga. Mediante el empleo de un esquema con ventana deslizable, TCP logra solucionar el problema de control de flujo de punto a punto; no cuenta con un mecanismo explícito para control de congestión. Sin embargo, mediante la implementación cuidadosa de un esquema de retransmisión, se puede ayudar a evitar la congestión, mientras que un esquema pobre puede agravarse.

5.5 Formato del segmento TCP

La unidad de transferencia entre dos módulos TCP es llamado segmento. Los segmentos son intercambiados para establecer la conexión, transferir datos, enviar ACKs, anunciar el tamaño de la ventana y cerrar las conexiones.



Formato del Segmento TCP

El puerto origen y el puerto destino identifican a las aplicaciones del nivel superior que están usando la conexión.

Número de secuencia. Especifica la posición de los datos en el segmento dentro de la secuencia de bytes del transmisor.

El número de ACK indica el número de secuencia del siguiente octeto de datos que se espera recibir.

HLEN es la longitud del encabezado del segmento, medida en palabras de 32 bits. Los segmentos TCP no tienen una longitud fija, ya que el campo de opciones es de longitud variable).

Las banderas contienen información sobre como interpretar el contenido de otros campos. Estas banderas son:

- URG. Empleado para indicar que el campo de apuntador urgente es significativo.
- ACK. Indica si el campo de ACK es significativo.

- PSH. Asignado en caso de solicitarse una función de empuje.
- RST. Empleado para solicitar la reinicialización de la conexión.
- SYN. Sincroniza los números de secuencia.
- FIN. Indica que el transmisor no tiene más datos que enviar.

Ventana. Su valor indica cuantos octetos puede aceptar el receptor, este valor es asignado en base al campo que contiene el número de ACK.

Checksum: como en todo PDU el checksum sirve para determinar si el segmento ha llegado libre de errores.

Apuntador de emergencia. Apunta al primer octeto de datos urgentes o de datos "fuera de banda" en el segmento.

El campo de opciones puede tomar tres valores definidos para el estándar TCP:

0 = Fin de la lista de opciones.

1 = No operación.

2 = Tamaño máximo del segmento.

El campo de relleno asegura un alineamiento de 32 bits al campo de opciones.

5.6 Datos fuera de banda

A pesar de que TCP es un protocolo con conexión, es importante que el programa en uno de los extremos de la conexión tenga la capacidad de enviar datos fuera de banda, sin que el programa en el otro extremo tenga que considerar a los segmentos que se encuentran en el flujo.

Para acomodar las señales de fuera de banda, TCP permite al transmisor el envío de datos como urgentes, significando esto, que el programa del módulo receptor debe ser notificado del envío de datos urgentes tan pronto como sea posible sin importar su posición. El protocolo especifica que cuando se encuentren datos urgentes, el módulo TCP receptor debe notificar a cualquier aspiración asociada con la conexión entrar en "modo urgente". Después de que se han consumido todos los datos urgentes, TCP indica a la aplicación que debe volver al modo de operación normal.

El mecanismo que utiliza TCP para informar sobre la existencia de datos urgentes en un segmento, es mediante la asignación de la bandera URG y el campo de apuntador urgente. Cuando esta bandera ha sido asignada, el apuntador de urgencia especifica la posición de la ventana en la que terminan los datos urgentes.

5.7 Tamaño máximo del segmento

El software de TCP, usa el campo de opciones para negociar ciertos aspectos con el otro extremo de la conexión. Una de las opciones permite especificar el Tamaño Máximo del Segmento (MSS-Maximum Segment Size) que puede recibir. Es muy importante para los dispositivos que se conectan a redes de alta velocidad escoger un tamaño máximo de segmento que llene los paquetes, o estos no harán un buen uso del ancho de banda.

A diferencia de los datagramas, los fragmentos no son mensajes independientes, todos los fragmentos deben llegar o el datagrama completo debe ser retransmitido.

En teoría el tamaño optimo de un segmento ocurre cuando los datagramas IP que llevan los segmentos son tan grandes como sea posible, sin requerir fragmentación en la ruta del origen al destino. Lo anterior es complicado por las siguientes razones:

1. TCP no cuenta con un mecanismo para dicha tarea.
2. Los gateways en una internet pueden cambiar dinámicamente, por lo que la ruta de un datagrama hacia su destino puede cambiar y por lo mismo puede presentarse algún cambio en el tamaño en que deben fragmentarse los segmentos.
3. El tamaño optimo, depende de los encabezados del protocolo del nivel inferior.

5.8 Confirmación y Retransmisión

Los ACKs hacen referencia a la posición dentro del flujo usando el número de secuencia del flujo. El receptor recolecta los octetos de los segmentos que llagan y reconstruye una copia exacta del flujo y que se está enviando.

Ya que los segmentos viajan en datagramas IP, estos pueden perderse o enviarse fuera de orden; por lo que el receptor usa el número de secuencia para

reordenar los segmentos.

Un ACK siempre especifica el número de secuencia del siguiente octeto que el receptor espera recibir.

El esquema de reconocimiento de TCP es acumulativo pues reporta la parte del flujo que se ha acumulado. Los ACKs acumulativos tienen como ventaja el que no son ambiguos y son fáciles de generar. Otra ventaja es que los ACKs perdidos no necesariamente forzan la retransmisión.

Una gran desventaja es que el transmisor no recibe información sobre las transmisiones exitosas, sino solamente sobre una sola posición en el flujo que se ha recibido

5.9 Timeouts y Retransmisiones

Cada vez que se envía un segmento, TCP inicializa un contador y se espera la confirmación de recepción mediante un ACK. Si el contador expira antes de que el segmento sea confirmado, TCP asume que el segmento se perdió o hubo corrupción en la transferencia, por lo que el paso a seguir es la retransmisión del segmento.

Puesto que TCP fue pensado como un protocolo para ser usado en una internet, un segmento que viaja entre dos puntos, debe en ocasiones pasar por redes de diferentes velocidades y múltiples gateways, por lo que es imposible determinar previamente la velocidad con la que regresará un ACK a la fuente, pues intervienen factores tales como tráfico, velocidades, retardo en los gateways, por lo que este tiempo de transmisión entre dos puntos puede variar de un instante a otro.

TCP debe por lo tanto adaptarse a estas condiciones para evitar retransmisiones innecesarias haciéndolo mediante un algoritmo de retransmisión adaptable. Esencialmente TCP, monitorea el desempeño de cada conexión y deduce valores razonables para el tiempo de vida.

Este cálculo consiste en el estimado entre el tiempo de envío el segmento y el tiempo de recepción del ACK, dicho estimado se conoce como Tiempo Muestra del Viaje Redondo o Tiempo de Viaje Redondo (Round Trip Time). TCP almacena sus tiempos estimados obteniendo con esto un promedio.

5.10 Respuesta ante la Congestión

TCP debe ser capaz de actuar ante la congestión en una internet. La congestión es una condición severa de retardo ocasionada por la sobrecarga de datagramas en uno o más puntos de conmutación. Cuando se presenta la congestión, el retardo se incrementa y las colas de datagramas en los gateways aumentan hasta que pueden ser enrutados. En el peor de los casos, el número total de datagramas que llegan a un gateway congestionado, crecen hasta que el gateway satura su capacidad y los datagramas comienzan a ser eliminados.

Debido a que los puntos terminales no conocen las causas o el lugar donde ocurre la congestión, para ellos simplemente se refleja como un incremento en el retardo Desafortunadamente, la mayoría de los protocolos de transporte emplean tiempos de vida y retransmisiones, por lo que ante el incremento en el retardo, responden con la retransmisión de datagramas, lo que lógicamente agrava el problema de congestión.

A la condición producida por el incremento del retardo ocasionado por la retransmisión, se le conoce como "Colapso por Congestión". Para evitar dicho colapso, TCP debe disminuir la velocidad de transmisión. Como se mencionó anteriormente ICMP controla esta condición con su característica de control de flujo, aun cuando los protocolos de transporte como TCP pueden ayudar a evitar la congestión, reduciendo las velocidades de transmisión automáticamente cuando se presentan retardos.

Para esto, el estándar TCP recomienda el empleo de dos técnicas relacionadas entre sí y de fácil implementación. Las técnicas a las que hacemos referencia son:

- Inicio Lento
- Decremento Multiplicativo.

TCP conoce el tamaño de la ventana en el receptor y para controlar la congestión se mantiene un segundo límite llamado "límite de congestión de la ventana" o "ventana de congestión".

En estado estático, en una conexión no congestionada, la ventana de congestión es del mismo tamaño que la ventana del receptor. El reducir la ventana de congestión, reduce el tráfico que TCP ingresa a cada conexión. Para estimar el

tamaño de la ventana de congestión, TCP asume que la mayoría de los datagramas se debe a la congestión, por lo que emplea la siguiente estrategia:

Decremento Multiplicativo. Ante la pérdida de un segmento se reduce el tamaño de la ventana de congestión a la mitad. Para los segmentos que se mantengan en la ventana permitida, se elimina el valor del contador exponencialmente (Retroceso Exponencial del Contador de Retransmisión).

Como la ventana de congestión se reduce a la mitad por cada pérdida, la ventana se decrementa exponencialmente en caso de que la pérdida continúe presentándose, esto es, si existe probabilidad de congestión, TCP reduce el volumen de tráfico y la velocidad de retransmisión exponencialmente. Si la pérdida persiste, el Protocolo de Control de Transmisión limita eventualmente la retransmisión a un solo datagrama y continua duplicando los valores de tiempo fuera antes de retransmitir. La idea es proporcionar una reducción de tráfico rápida y significativa para permitir a los gateways tener tiempo suficiente para despejar los datagramas que se encuentren en sus colas.

Para recuperarse después de una congestión, TCP emplea la técnica de "inicio lento" que permite escalar a la transmisión.

5.10.1 Recuperación mediante Inicio Lento

Cuando se inicia el envío de tráfico en una nueva conexión, o se incrementa el tráfico después de un periodo de congestión, se inicia la ventana de congestión al tamaño de un solo segmento, incrementándose en una unidad cada vez que se recibe un ACK.

El inicio lento evita recargar la internet con tráfico adicional, inmediatamente después que la congestión se ha despejado o cuando nuevas conexiones inician repentinamente.

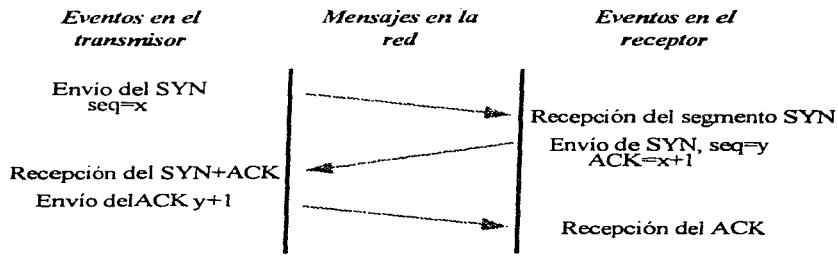
Bajo condiciones ideales, el inicio no es muy lento, TCP inicializa la ventana de congestión con un valor unitario, envía un segmento inicial y espera, Cuando se recibe el ACK, aumenta el valor de la ventana a 2, envía 2 segmentos y espera, al recibir estos dos ACKs, aumenta nuevamente el valor y envía entonces 4, etc. Con cuatro viajes redondos. TCP puede enviar 16 segmentos, con la frecuencia necesaria para alcanzar el límite de la ventana. Aún para ventanas muy grandes, solamente toma $log_2 N$ viajes redondos antes de poder enviar N segmentos.

Para evitar incrementar el tamaño de la ventana demasiado rápido y evitar ocasionar congestión adiciona, TCP agrega una restricción más. Una vez que la ventana de congestión alcanza la mitad de su tamaño original, TCP entra en una fase que evita la congestión y disminuye la proporción del incremento. Durante esta fase, se incrementa en uno la ventana de congestión, solamente si todos los segmentos en la ventana han sido confirmados.

Juntos, el incremento de inicio lento, el decremento multiplicativo, el impedimento de la congestión, la medición de la variación y la eliminación del contador exponencial, mejoran el desempeña de TCP dramáticamente sin que por ello aumento el trabajo de cálculo que debe realizar el software.

5.11 Establecimiento de una Conexión TCP

Para establecer una conexión, TCP utiliza el saludo de 3 manos. En el caso más simple, este saludo funciona de la siguiente forma:



Secuencia de los mensajes en un saludo de 3 manos
Los segmentos SYN (de sincronización) llevan la información del número de secuencia inicial.

El primer segmento del saludo es identificado porque tiene la bandera SYN asignada. El segundo mensaje tiene tanto la bandera de SYN como la de ACK asignadas, indicando que se envia el ACK del primer segmento SYN y que el saludo continua. El mensaje final es solamente un ACK y es simplemente usado para informar al destino que ambos extremos coinciden en que la conexión ha sido establecida.

El saludo de tres manos es necesario y suficiente para la sincronización correcta de los dos extremos de la conexión.

5.12 Números de secuencia iniciales

El saludo de tres manos cumple con dos funciones principales.

- Garantizar que ambos lados de la conexión, estén listos para la transferencia de datos.
- Permite a ambos lados coincidir en los números de secuencia iniciales, los cuales son enviados y confirmados durante el saludo. Cada máquina debe elegir un número de secuencia inicial aleatorio el cual empleará para identificar bytes en el flujo que se está enviando.

5.13 Terminación de una conexión TCP

Dos aplicaciones que usan TCP para comunicarse, pueden terminar la conversación exitosamente, mediante una operación CLOSE.

TCP usa internamente una variación del saludo de 3 manos para terminar una conexión. Cuando una aplicación indica a TCP que ya no tiene datos que enviar, TCP cierra la conexión en una dirección. Para cerrar la otra mitad de la conexión, el módulo TCP transmisor termina el envío de los datos restantes, espera que el receptor confirma y envía un segmento asignando la bandera FIN. El receptor confirma entonces la recepción del segmento FIN e informa a las aplicaciones de su lado que ya no existen más datos disponibles.

Una vez que la conexión se ha cerrado en una dirección, TCP termina la aceptación de datos en esa dirección. Mientras tanto, los datos pueden seguir fluyendo en la dirección opuesta, hasta que el transmisor indique el cierre. Por supuesto, los ACKs continúan fluyendo en sentido opuesto hacia el transmisor, aún después de que la conexión ha sido cerrada.

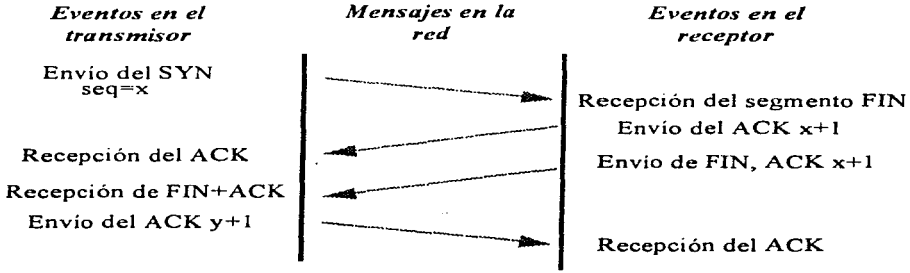
Cuando ambas direcciones han sido cerradas, el software de TCP borra el registro de la conexión.

La diferencia entre los saludos de manos para iniciar y terminar una conexión, ocurre después de que la máquina envía el segmento FIN inicial. En vez de generar inmediatamente un segundo segmento Fin, TCP envía un ACK y posteriormente informa a la aplicación sobre la petición de cierre de conexión.

El envío de una petición a la aplicación y la espera de una respuesta puede tomar mucho tiempo.

El ACK previene la retransmisión del segmento Fin inicial durante la espera. Finalmente, cuando la aplicación ordena el término de la conexión, TCP envía el segundo segmento FIN y el extremo original responde con el tercer mensaje, un ACK.

Variación del Saludo de 3 manos empleado para cerrar una conexión. El extremo que recibe el primer segmento FIN realiza su confirmción inmediata, y espera un tiempo antes de enviar el segundo segmento FIN.



5.14 Etapas de TCP

Una conexión TCP pasa por diferentes etapas. En un principio, la conexión se establece mediante el intercambio de mensajes, posteriormente los datos son transmitidos y finalmente se cierra la conexión nuevamente con el intercambio de mensajes. Cada paso en el progreso de una conexión corresponde a un estado. El software de TCP al final de cada conexión, guarda un registro del estado en que concluye la conexión.

A continuación se presentan los estados típicos de un cliente y un servidor en la progresión de una conexión:

5.14.1 Establecimiento de la conexión

Servidor:

Estado	Descripción
CLOSED	Estado ficticio, previo al inicio de la conexión.
LISTEN	El servidor espera una conexión de algún cliente.
SYN-RECEIVED	El servidor ha recibido un SYN. Envía un SYN/ACK, y permanece en espera de la confirmación de recepción
ESTABLISHED	Se ha recibido el ACK por lo que la conexión se ha establecido.

Cliente:

Estado del cliente	Descripción
CLOSED	Estado ficticio, previo al inicio de la conexión.
SYN-SENT	El cliente ha enviado un SYN al servidor.
ESTABLISHED	El cliente ha recibido un SYN/ACK del servidor y le ha regresado en respuesta un ACK. Por lo anterior la transferencia de datos procede.

5.14.2 Cierre de la conexión

Extremo que cierra la conexión:

Estado	Descripción
FIN-WAIT-1	El extremo que cierra la conexión espera a que el otro extremo envíe un FIN. En este estado todavía se reciben datos del otro extremo.
FIN-WAIT-2	El extremo que cierra la conexión ha recibido un ACK del otro extremo diferente a un FIN. Espera un FIN, aceptando mientras tanto los datos que lleguen.
CLOSING	Ha llegado un FIN/ACK. El extremo que cierra envía un ACK. Este estado puede alcanzarse ya sea por

5.14.1 Establecimiento de la conexión

Servidor:

Estado	Descripción
CLOSED	Estado ficticio, previo al inicio de la conexión.
LISTEN	El servidor espera una conexión de algún cliente.
SYN-RECEIVED	El servidor ha recibido un SYN. Envía un SYN/ACK, y permanece en espera de la confirmación de recepción
ESTABLISHED	Se ha recibido el ACK por lo que la conexión se ha establecido.

Cliente:

Estado del cliente	Descripción
CLOSED	Estado ficticio, previo al inicio de la conexión.
SYN-SENT	El cliente ha enviado un SYN al servidor.
ESTABLISHED	El cliente ha recibido un SYN/ACK del servidor y le ha regresado en respuesta un ACK. Por lo anterior la transferencia de datos procede.

5.14.2 Cierre de la conexión

Extremo que cierra la conexión:

Estado	Descripción
FIN-WAIT-1	El extremo que cierra la conexión espera a que el otro extremo envíe un FIN. En este estado todavía se reciben datos del otro extremo.
FIN-WAIT-2	El extremo que cierra la conexión ha recibido un ACK del otro extremo diferente a un FIN. Espera un FIN, aceptando mientras tanto los datos que lleguen.
CLOSING	Ha llegado un FIN/ACK. El extremo que cierra envía un ACK. Este estado puede alcanzarse ya sea por

Estado	Descripción
	un estado FIN-WAIT-1 o bien por un estado FIN-WAIT-2.
TIMED WAIT	La conexión se mantiene en el limbo el tiempo necesario para que todos los mensajes de conexión qucce pudieran existir todavía en la red hayan sido concluidos. Si se recibe cualquier mensaje, TCP sabe que pertenecen a una conexión ya cerrada, por lo que los descartará. El periodo de tiempo fuera es dos veces el estimado del tiempo de vida máximo de un segmento.
ESTABLISHED	Se ha recibido el ACK por lo que la conexión se ha establecido.

Estados del otro extremo para el cierre de la conexión

Estado	Descripción
CLOSE WAIT	Se ha recibido un ACK. La aplicación puede enviar más datos opcionalmente. TCP envía un ACK y espera a que la aplicación termine la conexión.
LAST ACK	La aplicación ha terminado la conexión, y TCP ha enviado un FIN. Simplemente espera el ACK par la confirmación de la terminación. Este estado persiste durante un periodo de tiempo fuera.
CLOSE	Se borra toda la información sobre la conexión.

5.15 Funciones de TCP

Finalmente podemos resumir las funciones que desempeña este protocolo, siendo estas las que se listan a continuación:

- Asociación de puertos con conexiones.
- Establecimiento de conexiones mediante un saludo de tres manos.

- Segmentación de datos para la transmisión.
- Numeración de datos.
- Reconocimiento positivo con retransmisión.
- Manejo de duplicación de segmentos.
- Cálculo de checksums.
- Regulación del flujo de datos con ventanas se transmisión y recepción.
- Termino de conexiones en orden de ocurrencia.
- Cancelación de conexiones.
- Interacción con los niveles de aplicación.
- Empuje de datos.
- Señalización de datos urgentes.
- Chequeo y reporte de errores.
- Ajuste ante la congestión de la red.

5.16 Protocolo del Datagrama de Usuario (UDP)

UDP es un protocolo mucho mas simple que TCP, y es útil para situaciones en las que los poderosos mecanismos de confiabilidad de TCP no son necesarios. La sobrecarga generada por la recepción y envío de todos los mensajes requeridos para establecer y terminar una sesión se evita enviando simplemente una petición y una respuesta, por lo que UDP es un constructor de bloques perfecto para la realización de funciones de monitoreo, depuración, administración y funciones de evaluación.

UDP es un protocolo sin conexión, por lo que no proporciona confiabilidad, control de flujo y medidas de recuperación de errores. Sirve principalmente como un multiplexor/demultiplexor para la recepción y envío de trafico IP; sirve simplemente como una interface de aplicación para IP.

UDP hace uso del concepto de puertos para dirigir los datagramas a las aplicaciones apropiadas del nivel superior.

5.16.1 Mecanismos de UDP

UDP tiene asignado un identificador de protocolo único. Su identificador es el numero 17, el cual es asignado al campo protocolo del datagrama. IP para indicar la salida de mensajes UDP. Los mensajes de IP con valor de diecisiete, en el campo de protocolo son enviados a UDP. UDP forma un mensaje simplemente agregando un encabezado a los datos de la aplicación.

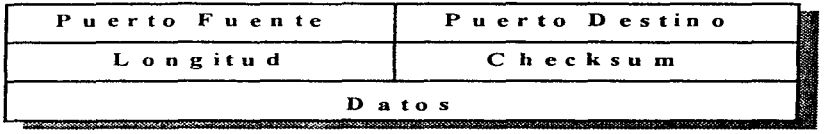
5.16.2 Formato de los mensajes UDP

Puerto Fuente. Identifica el puerto del proceso de aplicación que transmite.

Puerto Destino. Identifica el proceso receptor en la maquina destino.

Longitud. Este campo indica la longitud del datagrama de usuario incluyendo el encabezado y los datos.

Checksum. Para validación del contenido del datagrama



Formato del mensaje UDP

5.16.3 Desbordamientos de UDP

Cuando una aplicación adquiere un puerto UDP, se reserva algún espacio en el buffer para mantener una cola de los datagramas de usuario que llegan a ese puerto. Generalmente un servidor basado en UDP no tiene forma de predecir o controlar cuantos datagramas le serian enviados en cualquier momento.

Si el servidor es bombardeado con mas datagramas de los que puede manejar, el desbordamiento es simplemente descartado. El hecho de que esto suceda, se manifestará en los reportes de las estadísticas de la red con un encabezado tal como "Desbordamientos del socket UDP".

6. ADMINISTRACIÓN DE RED

Dentro del desarrollo de las comunicaciones en sistemas se presentan dos tendencias principales en el desarrollo del software para la infraestructura, los estándares definidos por OSI (Open System Interconection) muestran la interconexión en sistemas abiertos, que hasta ahora a tenido un desarrollo lento, pero seguro en su colocación dentro del mercado; y las definiciones de TCP/IP hasta ahora utilizada en la mayoría de redes. A la instalación de una red sigue de manera obligatoria la ardua tarea de administrarla. De ésto depende la buena calidad de desempeño con que se trabaje en la red, por tanto, conocer las funciones y necesidades a cubrir por un administrador de red, es más que importante, obligatorio.

6.1 Consideraciones en la Administración de Red

Dentro de los puntos críticos que debemos cubrir al tener la responsabilidad del buen funcionamiento de una red se encuentran:

Control estratégico de gastos: el costo en equipo de red y cómputo debe ser comparado contra los beneficios que en servicio éste ofrezca, de tal modo que en un esquema costo-beneficio la inversión resulte equitativa.

Balance de necesidades: se deben asignar y controlar recursos de acuerdo a las diversas necesidades de los múltiples tipos de usuarios.

Reducción de tiempos muertos: se busca el aprovechamiento al cien por ciento de recursos sin descuidar su disponibilidad.

Ya que el objetivo primordial de llevar a cabo la administración de red es lograr el funcionamiento óptimo de ésta, se debe considerar lo que el usuario final y el mismo administrador esperan como beneficios intrínsecos a las funciones de administración; por tanto a continuación se mencionan los puntos mas significativos:

- Un servicio eficiente para el usuario final (comunicación constante y buen tiempo de respuesta)
- Capacidad de solución a problemas físicos.
- Capacidad de solución a problemas lógicos.
- Detección de problemas de manera rápida (en la mayoría de los casos se considera al usuario final la alarma potencial para saber de anomalías),

en este aspecto la inteligencia artificial tiene un gran campo de desarrollo.

- Capacidad de detección y diagnóstico del fallas en aplicaciones, red y software.
- Posibilidad de análisis del desarrollo de la red en tiempo real.
- Almacenamiento automático de la información necesaria para la realización de un estudio estadístico.
- Notificación de fallas de manera no redundante, es decir, evitar el aviso de fallas derivadas del problema principal.
- Contar con el conjunto de variables que permitan la administración, análisis y planeación de la red.
- Contar con una definición específica de cada variable a controlar o monitorear.
- Respuesta continua y rápida a cualquier cambio que se presente en aplicaciones miembros, dispositivos, servicios o tarifas y no perder la comunicación mientras se da esta actualización de información.
- Posibilidad de expansión dinámica y reconfiguración.
- Existencia de una sola fuente de administración de la red para evitar divergencia en el direccionamiento.
- Contar con alto nivel de seguridad de aplicaciones, sistemas, transmisión y equipo.
- Simplificar la contabilidad de la información para lograr el ágil control de costos.
- Integración con arquitecturas múltiples de red, sistemas de procesamiento y elementos de red.
- Contar con un centro de administración que permita la administración remota en redes de área local.
- Reportes integrados de flujo de información.

- Reducción de tiempos muertos.
- Disminución del personal dedicado a atender fallas en la administración de la red.

Para cubrir todos los estos aspecto es necesario identificar los siguientes factores en la Administración de una red:

- Manual de procedimientos para la administración de red, incluyendo el uso de las herramientas con que se disponga para dicho fin.
- Instrumentos tanto de hardware como de software que contemplen bases de datos que permitan una predicción del futuro.
- Recursos humanos que den soporte a las funciones de administración de red.

6.2 Áreas Funcionales de Administración.

Siguiendo el modelo OSI para la administración de redes se muestran a continuación las áreas funcionales que permiten cubrir las responsabilidades antes mencionadas.

6.2.1 Administración de Fallas.

Actividades para un mantenimiento dinámico del servicio de red. Acciones rápidas que mediante el reconocimiento de problemas y degradación del desempeño, actúan en caso necesario, incluyendo: diagnóstico, reparación, prueba y respaldo, es decir, facilidad para determinar con exactitud

- La falla.
- El aislamiento, para lograr en los demás dispositivos la operación continua.
- La reconfiguración y modificación de la red con el menor impacto posible.
- La corrección de irregularidades o reemplazo de componentes, disminuyendo el tiempo empleado en ésto.

Es importante considerar la redundancia de componentes, que permitan ofrecer un mejor servicio.

6.2.2 Administración de la Contabilidad.

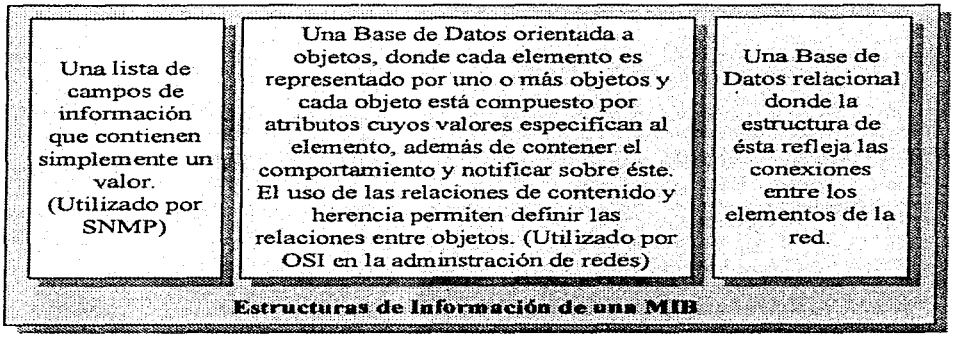
Capacidad de habilitar el uso de objetos o recursos y al mismo tiempo valuar el uso de los mismos, evitando el abuso de privilegios de acceso a la red de usuarios o bien el uso ineficiente de los mismos. Incluye recolección, interpretación, procesamiento y reporte del costo por recurso utilizado.

6.2.3 Administración de Configuración

Donde se tiene control físico, lógico, electrónico, mantenimiento y distribución de software. Debe existir una revisión continua de la información y servicios existentes en la red, con el propósito de asistir y proveer de éstos para lograr la interconexión de servicios. La revisión debe estar concentrada en la inicialización, mantenimiento y baja de los componentes individuales y subsistemas lógicos; realizando una configuración total durante la instalación de recursos de cómputo y comunicación.

La administración de configuración incluye:

Definición de la información de configuración, describiendo la naturaleza y el estado de los recursos en red (tanto físicos: enrutadores, puentes, módem, etc. así como lógicos: contadores, circuitos virtuales, etc.), así como sus especificaciones y atributos (nombre, dirección, características de operación, versión, etc.). Esta información puede estructurase:



Definir y modificar los valores de los atributos, considerando siempre el nivel de seguridad permitido en cada dispositivo, modificaciones en la base de datos, actualizaciones en las tareas a seguir ante determinadas situaciones y la limitante de que no todos los atributos (o dispositivos)

pueden ser modificados de manera remota.

Definir y modificar las interacciones tales como: topología, jerarquía, conexiones físicas y lógicas o administración del dominio.

Inicializar y terminar las operaciones de red, incluyendo mecanismos que habiliten a los usuarios a definir y redefinir los dispositivos en la red o sus atributos, realizando una revisión de las definiciones que se hagan. Antes de finalizar cualquier procedimiento se debe considerar la presentación de la información para estudios estadísticos.

Distribución de software

Examinar relaciones y conexiones

Reporte de estado de configuración

6.2.4 Administración del Rendimiento

La facilidad de evaluar y controlar los objetos y su efectividad en actividades de comunicación podemos dividirla en dos categorías: monitoreo y control de la red. Por lo que se debe identificar:

- La capacidad de utilización
- El nivel de tráfico
- Niveles inaceptables de rendimiento
- Tiempos de respuesta.
- Verificación de un servicio constante.
- Identificar cuellos de botella.
- Estabilización y reporte de rutas de decisión en la construcción y planeación de las mismas.
- Construcción y mantenimiento del desempeño de la base de datos y actualización de procedimientos para control operacional.

6.2.5 Administración de la Seguridad

Utilizada sobre todo en esquemas OSI de administración de red, soportada por llaves encriptadas (password). Se deben considerar un nivel de seguridad tanto

en los equipos de cómputo como en la red que permita un acceso controlado y restringido a los sistemas de las personas que lo requieran y especificando las tareas que cada una puede llevar acabo sin provocar con esto limitantes que entorpezcan el acceso al mismo. Como funciones específicas tenemos:

Mantenimiento de la información de seguridad, autenticidad de la información de permisos de acceso, operación de parámetros de servicio y mecanismos de seguridad, eventos login, reporte de violaciones de seguridad

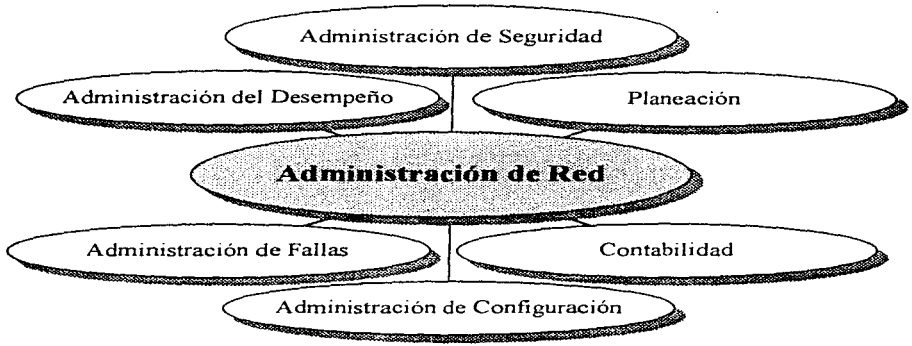
Control de acceso de recursos, envuelve la verificación y autorización de servicios de acuerdo a permisos: códigos de seguridad, fuentes de enrutamiento, alarma para diferentes niveles, tablas de contabilidad.

Procesos de control de encriptación, verificando que los procesos de desencriptación sean ejecutados por los usuarios autorizados.

6.2.6 Administración de la Planeación

Proceso para determinar la red óptima, basada en la información del desempeño de la red, flujo de tráfico, utilización de los recurso, requerimientos de red estimación del crecimiento de presentes y futuras aplicaciones, modelado incluso del tamaño de interfaces en los dispositivos. La gente debe

Áreas Funcionales de Administración de Red

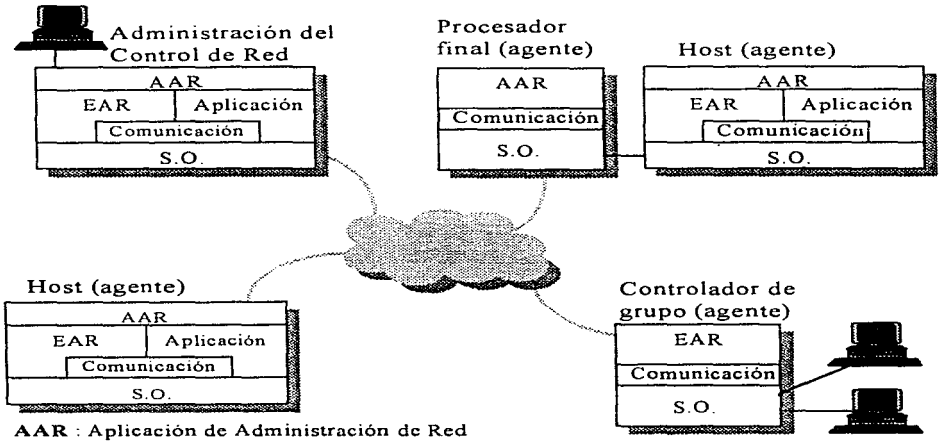


tener conocimientos técnicos y entender cargas de trabajo, además de proyectar cargas de trabajo y determinar puntos críticos.

6.3 Sistemas de Administración de Red.

Dada la extensa gama de funciones a cubrir por el administrador, se hace necesaria la utilización de herramientas físicas y lógicas, que faciliten el monitoreo y control de todos y cada uno de los dispositivos que conforman el ambiente de red, en la figura contigua, se muestra un esquema de la arquitectura típica de configuración para administración de red. En donde:

Cada Entidad de la Aplicación de Red es responsable de colectar estadísticas en comunicaciones y actividades relacionadas con la operación, almacenar estadísticas locales, responder a comandos desde el centro de control de red, incluyendo: transmisión de estadísticas recolectadas al centro de control, cambio de parámetros, proveer de información de estado, generar tráfico artificial para pruebas. Estas funciones también las deberá llevar a cabo cada Aplicación de Administración de Red considerando su arquitectura (sistema operativo y sistema de comunicaciones) para mantener una alta disponibilidad. Repartiendo sus funciones en centros de recolección estadística (con una arquitectura sencilla) y un centro para el control de la información.



AAR : Aplicación de Administración de Red
EAR : Entidad de la Aplicación de Admin. de Red
S.O. : Sistema Operativo

6.4 Arquitectura del software de Administración de Red

Actualmente la administración de red sigue una arquitectura dividida en tres niveles:

1. ***Software de presentación para el usuario***. la llave para un efectivo sistema de administración es la definición de una interfaz única para el usuario, de tal modo que sea similar para cualquier nodo de cualquier proveedor, permitiendo así la administración de una configuración heterogénea con mínimo entrenamiento. Dado que el volumen de información que se maneja puede ser muy elevado, así como las funciones a realizar, se debe contar con herramientas de presentación que permitan organizar, sumar y sobre todo simplificar esta información y funciones.

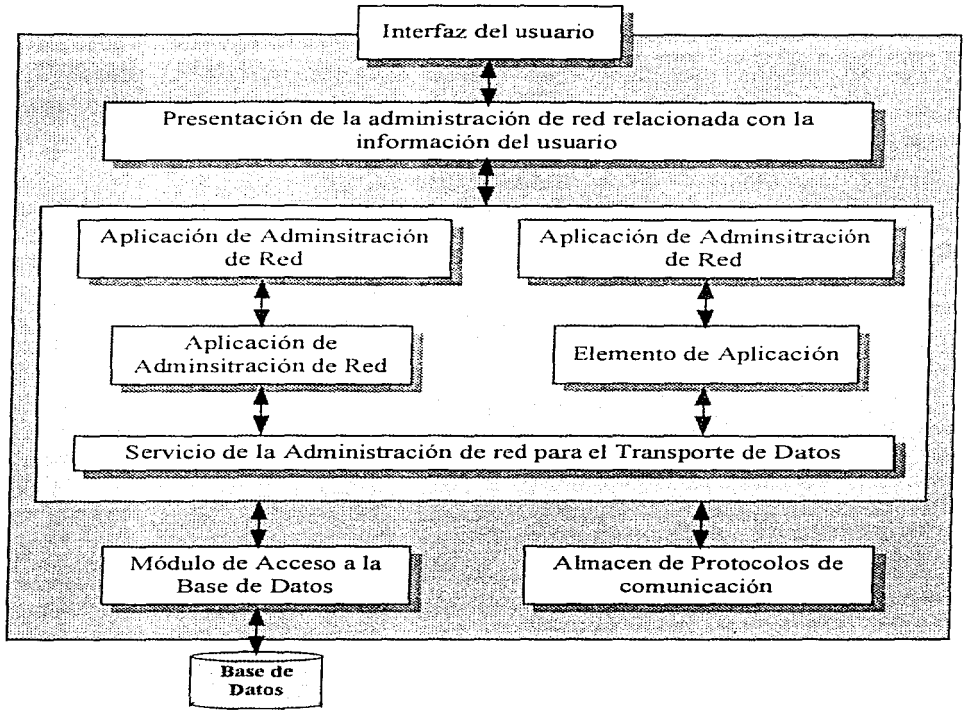
2. ***Software administrador de red***, provee especificaciones en la administración de red. Puede ser muy simple (SNMP, por ejemplo) o muy complejo (Sistemas de Administración de OSI). La figura siguiente ilustra la manera en que el software administrador de red está dividido en tres niveles:

   - Conjunto de Aplicaciones de Administración de Red, que proveen el servicio de interés para el usuario. Por ejemplo: Administración de fallas, contabilidad, configuración, desempeño y seguridad.

   - Elementos de la Aplicación, es decir módulos mas primitivos y de propósito general de funciones de administración de red. Permitiendo así su fácil reutilización de software.

   - Servicio de Administración de Red de Transporte de Datos, consiste de un protocolo usado para intercambiar información de administración del centro de control a sus agentes y viceversa. Generalmente provee de muchas funciones primitivas, tales como la obtención de información, definición de parámetros y generación de notificaciones.

3. ***Software de comunicación y soporte de la base de datos***, este software necesita accesar un MIB local, el cual es un agente que contiene la información, que refleja la configuración y el manejo del nodo, así como los parámetros necesarios para el control de su operación. La comunicación con otros nodos (agentes y manejadores) es soportada por un conjunto de protocolos (de OSI o de TCP/IP).



6.5 Características de las Aplicaciones para la Administración de Red.

- Separación de administración física (notificación de fallas en circuitos, líneas, multiplexores o cualquier otro dispositivo que se tenga de principio a fin de la comunicación) y lógica (vigilar sesiones y visibilidad de flujo de tráfico).

- Separación de arquitecturas de red, cada uno ofrece una solución particular para la administración de red no intercambiable (se utilizan

gateways para la comunicación entre diferentes arquitecturas pero éstos no relacionan la administración).

Los usuarios son vistos potencialmente como alarmas para conocer de los problemas que se presentan en la red.

El centro de administración debe observar:

Medios de transmisión, procesos de principio a fin, multiplexiones estáticas, sistemas de vídeo y satelitales, conexiones análogas, líneas privadas, conexiones digitales, etc., que causa conflicto y contradicción para nombrar cada estructura.

- Reportes innecesarios en papel, una tercera parte de los sistemas de administración de red ofrecen una generación de reportes gráficos que bien podrían cambiarse por la vista opcional de determinados aspectos de la red.

- En lo que se refiere a los recursos humanos, existe un alto índice de cambio de personal en la administración de una red, además

- Surge la necesidad de incrementar el número de personas de soporte.

6.5.1 Funciones del personal de soporte en la Administración de Red

Estructura de organización.

Control Central de la Red: Cubre fallas y desempeño de la red.

- Escritorio de ayuda, la gente debe contar con dos grandes cualidades, ambas igualmente importantes, destreza en la solución de problemas y en el trato interpersonal, ya que este puede ser el único contacto del soporte técnico con el usuario final.

- Soporte técnico, gente con conocimientos específicos del sistema y sus componentes se requieren gentes especialistas y analista, los primeros responsables del sistema, los segundos darán soluciones en base a un análisis preciso, así el diseño e implementación de procedimientos automatizados para operar la red, modelado y planeación serán tareas intrínseca de ésta área.

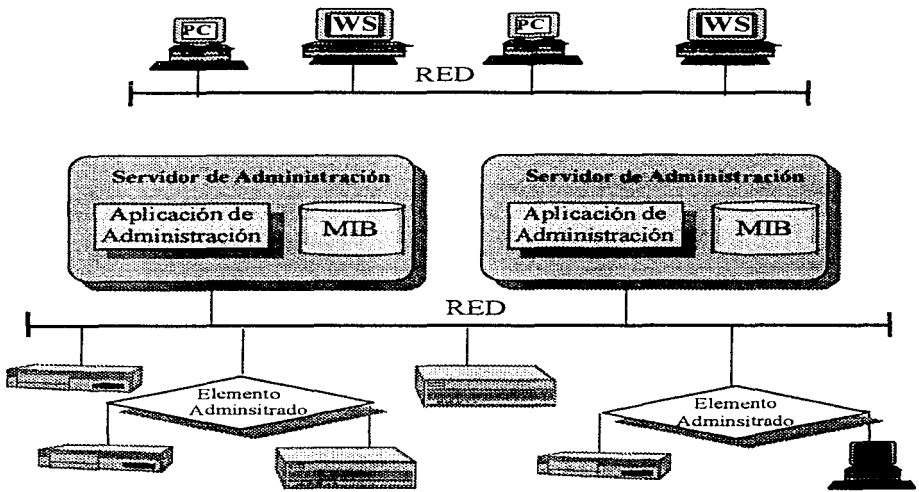
Además se debe contar con personal disponible a dar asistencia a usuarios remotos, esta gente debe cubrir todas las actividades antes mencionadas.

Cabe aclarar que dentro de los aspectos que provocan un incremento de costos directo en el crecimiento de una red, el que se refiere a los recursos humanos es el que incrementa de manera exponencial mientras que el equipo tiende a disminuir con respecto a las facilidades que ofrece.

6.6 Administración Distribuida.

El modelo distribuido de datos y la proliferación de PCs poderosas y estaciones de trabajo a bajo costo, así como LANs departamentales que traen consigo la necesidad de un control local que permita la optimización de aplicaciones distribuidas, da lugar a la arquitectura de un modelo distribuido de administración de red.

En el esquema se observan estaciones de trabajo localizadas de manera distribuida en el esquema global, esta estrategia da como consecuencia niveles de administración de red departamental que cubran requerimientos de red, del sistema y aplicaciones de usuarios locales. Para lograr esta arquitectura jerárquica se deben tomar en cuenta los siguientes elementos:



- Estaciones de administración distribuida con acceso limitado para el monitoreo y control.

- Una estación de trabajo central con respaldo de los derechos de acceso y disponibilidad para administrar todos los recursos de la red. Dependiendo de los niveles de acceso una estación de trabajo cliente puede accesar uno o mas servidores de administración siendo éstos el soporte de las definiciones de la aplicación administradora.

Como beneficios de un esquema distribuido de administración de red se tiene:

- Disminución del tráfico por administrador de red, ya que es limitado al ambiente local.
- Gran posibilidad de reducción de costos para equipo de monitoreo por el escalamiento de alcance y de la MIB.
- Los servidores administradores contienen además información de aplicaciones y clientes de protocolo similar que permita compartir esta información entre ellos.
- Para alcanzar los recursos de administración de red los servidores deben contar con agentes de software compatible o proxies

6.7 Proxies

El uso de proxies se hace necesario cuando la configuración con que se cuenta en algún dispositivo no soporta los nuevos estándares de administración, un proxy es una implementación en el servidor administrador que obtiene información de los nodos que no soportan al agente administrador pero pueden comunicarse con el agente proxy. El agente proxy traduce los requerimientos de la Aplicación Administradora de Red al dispositivo y viceversa.

6.8 Tendencias

Es importante considerar por otro lado el desarrollo de TCP/IP (Transmision Control Protocol / Internet Protocol) que actualmente domina el mercado. De manera semejante se presentan dos principales desarrollos para el software de administración de red: OSI-System Management, para sistemas OSI; y SNMP (Simple Network Management Protocol) para sistemas TCP/IP. De ambos dependen su difusión del crecimiento de los sistemas en sus respectivas plataformas.

7. MONITOREO

El objetivo principal del monitoreo es permitir al administrador de la red, conocer el estado actual de esta, logrando localizar cuellos de botella y otros problemas, así como realizar planes futuros en base a la observación obtenida del monitoreo.

La actividad de monitoreo dentro de una red debe considerar 3 aspectos importantes que incluyen el funcionamiento de la red, el estado de la red y la utilización de ésta.

7.1 Recursos de Monitoreo

Un sistema de monitoreo puede ser por hardware o por software. El monitoreo por hardware está basado en el hecho de que la mayoría de las características del funcionamiento de los sistemas computacionales pueden medirse mediante la detección de señales de transición de voltaje en la circuitería del hardware, mediante sensores que tienen la capacidad de medir las características físicas de algún elemento de la red. Las señales capturadas, son combinadas lógicamente utilizando algún tipo de contabilidad, almacenamiento y técnicas de mapeo. Los resultados, son almacenados secuencialmente o incluidos en una base de datos con el fin de describir el funcionamiento de los dispositivos observados.

Otro tipo de monitores pueden desarrollarse por software, los cuales, tienen el objetivo de analizar y evaluar el funcionamiento de los elementos de la red. Este tipo de monitor a diferencia del de hardware, no pretenden ser usados continuamente, para evitar saturación innecesaria. Muchas veces un monitor de software es capaz de interpretar los resultados obtenidos e indicar las causas de los problemas que se presentan con mayor frecuencia. Las mediciones pueden darse a conocer para el soporte de una administración operacional, o pueden generarse reportes para el soporte de un análisis del funcionamiento y la planeación.

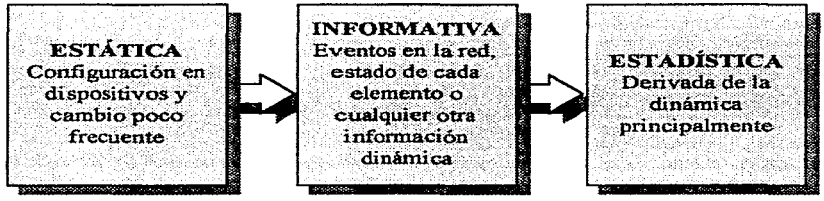
Lo más conveniente es la combinación de hardware y software, obteniendo así monitores más poderosos. Para reducir la carga de la interfaz de software, los datos pueden enviarse a una bitácora o base de información, los cuales pueden ser analizados por otro conjunto de programas independientes.

7.2 Diseño y Arquitectura de Monitores para Red.

La primera consideración para llevar a cabo el diseño de un sistema monitor de red es la definición de las tareas de administración de red que desean cubrirse, tomando en cuenta tres áreas de diseño:

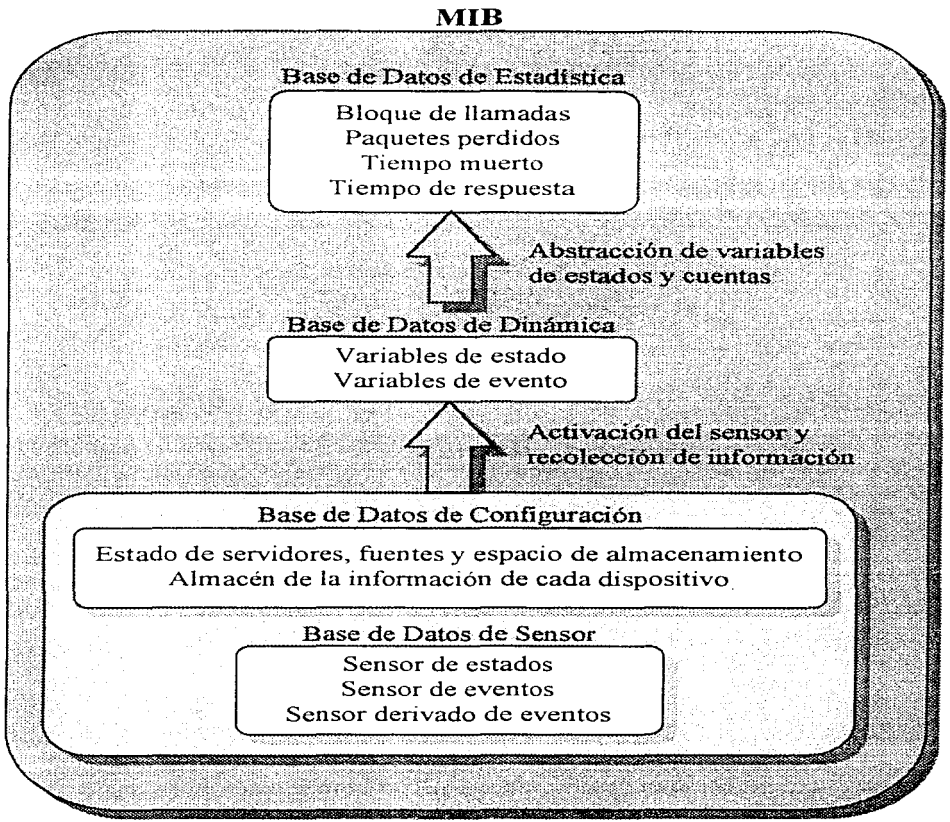
1. Acceso al monitoreo de información, es decir, como llegará la información desde un dispositivo al administrador.
2. Diseño del mecanismo de monitoreo, seleccionando la mejor manera de obtener información de los dispositivos.
3. Aplicación de la información monitoreada, identificando las áreas de aplicación que utilizarán esta información.

Por tanto la información que se defina para su monitoreo debe ser perfectamente identificada, pudiendo clasificarla en:



Siguiendo el esquema anterior se obtiene el modelo para una Base de Información Administrativa (MIB), en un esquema de monitoreo en tiempo real (propuesto por Mazumdar y Lazar) obteniendo el mostrado en la figura siguiente.

El propósito de contar con un monitor de red es el de medir y analizar periódicamente los parámetros más importantes. Además, cada dispositivo dependiendo de su tipo tiene diferentes características y posibles variables a monitorear.


MIB
Base de Datos de Estadística
Bloque de llamadas
Paquetes perdidos
Tiempo muerto
Tiempo de respuesta
Abstracción de variables de estados y cuentas
Base de Datos de Dinámica
Variables de estado
Variables de evento
Activación del sensor y recolección de información
Base de Datos de Configuración
Estado de servidores, fuentes y espacio de almacenamiento
Almacén de la información de cada dispositivo
Base de Datos de Sensor
Sensor de estados
Sensor de eventos
Sensor derivado de eventos

7.2.1 Aspectos a considerar en el diseño:

1. Módulo de Captura de Información. Incluyen datos generales del dispositivo, tales como dirección, memoria, sistema operativo, CPU, etc.

2. Flexibilidad en la selección de la ubicación de la interfaz digital para la obtención de información. Los módulos de obtención de información, pueden ligarse en cualquier parte de la interfaz digital. Dependiendo de la topología de la red, puede ser necesario el contar con información remota.

3. Facilidad de Control Central. El monitor del CPU de la red controla, la operación de todo el sistema, incluyendo la obtención múltiple de datos de los dispositivos involucrados.

4. Manejo de una Base de Datos. Para definir los umbrales óptimos y rediseñar la red, debe contarse con datos históricos. El contenido de la base de datos sobre el funcionamiento de la red, puede incluir medidas orientadas al servicio, (tales como disponibilidad, tiempo de respuesta, etc.) y eficiencia (relacionada con medidas tales como utilización de recursos, costos de los recursos, etc.).

5. Independencia del sistema. Los monitores deben ser independientes del hardware y sistema operativo de cada dispositivo. Para interpretar el tráfico, sin embargo, los protocolos deben ser conocidos por la unidad de obtención de datos.

6. Administración sobre el mantenimiento de la red. Para soportar actividades de mantenimiento de pequeña escala, deben de recolectarse y procesarse los datos de manera regular para mantener la información de configuración actualizada.

7. Indicadores del Funcionamiento. Dependiendo del producto bajo consideración, se pueden emplear diferentes indicadores. En promedio, un programa de medición debe incluir al menos uno de los siguientes indicadores: disponibilidad del enlace, tiempo de respuesta, retraso de la red, retraso del nodo, integridad y confiabilidad, parámetros de eficiencia, utilización del enlace y del software.

8. Dispositivos de alerta que generalmente incluyen una gran cantidad de alarmas empleadas para comunicar a los administradores el estado actual de la red.

9. Se deben generar reportes a diferentes niveles, esto es, mediante la recopilación de datos históricos de manera jerárquica que pueden incluir tiempos de respuesta por hora del día, para el día anterior, mes o año, registro de tráfico, registro de ocurrencia de errores, estado actual (tiempo de respuesta, tiempo de operación, etc.)

10. Analizar el funcionamiento actual de la red y las futuras tendencias, pudiendo de alguna forma evitar problemas potenciales en cuanto a la configuración de la red y obteniendo información necesaria para una futura reconfiguración.

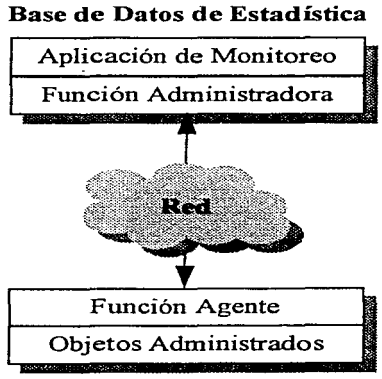
11. Dar aviso en el momento en que se presente alguna falla.

12. Incluir información sobre la topología de la red, caracterización de los nodos y el tráfico en la red.

13. La expansión es un punto importante a considerar, tomando en cuenta que la conectividad en las redes se incrementa constantemente.

7.3 Esquema de Monitoreo

Un esquema de configuración general para un monitor de red es:



Esta configuración requiere que el administrador y sistemas agentes compartan el mismo protocolo de administración y la misma sintaxis así como la semántica de la MIB.

Las técnicas para habilitar el administrador son: poleo y reporte de eventos. El reporte de eventos, es básicamente la interacción del administrador y el agente donde el administrador solicita información y el agente responde a estas solicitudes. El poleo a diferencia del anterior, manejará información periódica y predefinida, de tal forma que el administrador sólo estará pendiente para la recepción de ésta.

Para decidir que técnica debe utilizarse, deben considerarse los siguientes factores:

- Tráfico que genera cada requisición (poleo o reporte de eventos).
- Tiempo de retraso para notificar al administrador de red.
- Tiempo de procesamiento en dispositivos de administración
- Aplicaciones que en el monitor de red están siendo soportadas.

7.4 Clasificación de Monitoreo.

El monitoreo de red se puede enfocarse a dos tendencias anteriormente mencionadas:

- Orientado a servicio
- Orientado a eficiencia

7.4.1 Orientados a Servicios

Donde según el modelo de Terplan (1992) la prioridad la tienen los identificadores orientados a servicios, estos identificadores son:

Disponibilidad: Porcentaje de tiempo que un dispositivo o programa de red está disponible para el usuario. Éste debe medirse en relación con la aplicación para definir cifras de aceptación. La disponibilidad está en función de la confiabilidad de los componentes en la red. La confiabilidad, es la probabilidad de que un componente funcione adecuadamente bajo condiciones específicas. De tal forma que la disponibilidad puede expresarse como:

$$A = \frac{PTEF}{PTEF + PTR}$$

Donde:

PTEF: Promedio de Tiempo Ente Fallas.

PTR: Promedio de Tiempo de Reparación.

Tiempo de Respuesta: Tiempo que debe esperar el usuario para que la respuesta aparezca en la terminal y después de su requerimiento, por supuesto el tiempo de respuesta se antoja lo mas corto posible. Sin embargo debe considerarse lo anterior en relación al costo que lograr ésto significa, tanto de la computadora que procesa como de la competencia de recursos.

Para identificar estos rangos de respuesta se mencionan a continuación los definidos según un estudio realizado en 1988.

- Más de 15 segundos: son permitidos sólo para determinadas aplicaciones, el sistema debe ser diseñado para que el usuario pueda cambiar a otras actividades y verificar la respuesta solicitada mas adelante.
- Más de 4 segundos: rango alto para requerimientos de conversación, los cuales requieren que el operador retenga la información en memoria.
- De 4 a 2 segundos: pueden inhibir las operaciones en la terminal hasta recibir la respuesta.
- 2 segundos: en terminales este límite de respuesta es importante.
- 1 segundo: necesario en aplicaciones gráficas.
- 0.1 segundo: utilizado para selección con ratón o teclado.

Exactitud: Porcentaje de veces que no ocurren errores en la transmisión y entrega de información

7.4.2 Orientado a Eficiencia

Cantidad de información que transita (tráfico) Razón en la cual ocurren los eventos (transacciones, mensajes, transferencia de archivos).

Utilización Porcentaje de capacidad teórica de un recurso que se está usando (multiplexor, línea de transmisión, conmutación). El uso principal de medir la utilización, es encontrar los potenciales cuellos de botella y áreas de congestión, ésto es importante porque el tiempo de respuesta se incrementa exponencialmente con el incremento de utilización de recursos.

7.4.3 Monitoreo del Desempeño

El monitoreo del desempeño se compone de:

- Medición del desempeño
- Análisis del desempeño

Consta de módulos agentes dentro de los dispositivos en la red (hosts, enrutadores, puentes, etc.), para observar el tráfico de entrada y salida de cada nodo; el número de conexiones y el tráfico por conexión, permiten la realización de reportes que faciliten el análisis tales como:

Reportes de Apoyo para el Análisis en el Monitoreo de Desempeño

Nombre	Variables	Descripción
Matriz de Comunicación a Host	Fuente X Destino	Número o porcentaje de paquetes, información, octetos, etc.
Matriz de Comunicación a Grupo	Fuente X Destino	Número o porcentaje de paquetes, información, octetos, etc.
Histograma de tipo de paquetes	Tipo de paquetes	Número o porcentaje de paquetes por tipo.
Histograma del tamaño de paquetes	Tipo de paquetes por tamaño	Número o porcentaje de paquetes por longitud en bytes.

Nombre	Variables	Descripción
Distribución de tráfico - utilización	Fuente	Total de octetos, octetos de datos transmitidos.
Histograma de paquetes - tiempo de llegada	Tiempo de llegada	Señales de tiempo de acarreo consecutivo (red ocupada)
Histograma del canal de adquisición - retardo.	NIU (unidad de interfaz de red) Retardo de adquisición	Número o porcentaje de paquetes retrasados por NIU por un canal.
Histograma comunicación - retardo	Paquetes retardados	Tiempo que tarda el paquete de origen a destino.
Histograma cuenta - colisión	Número de colisiones	Número de paquetes por número de colisión.
Histograma cuenta - transmisión	Número de transmisiones	Número de paquetes por transmisión por el número de intentos de transmisión.

Lo anterior nos permite conocer cual es la situación actual de la red, pudiendo determinar las causas posibles de algún problema y las medidas correctivas o preventivas pertinentes.

7.4.4 Monitoreo de Fallas

Los principales problemas que podemos resolver con el monitoreo de fallas son: la falta de observación de problemas típicos, tales como los cuellos de botella entre los dispositivos, o bien la no observación de las fallas de manera más específica, tanto en un equipo determinado que permita identificar el nivel de falla: aplicación, transmisión, etc. En un ambiente completo y heterogéneo de tecnologías la detección del equipo específico de falla es elemental, para su corrección y visión de las áreas que en consecuencia presentarán problemas. Por lo que un monitor de fallas debe apoyar al aislamiento de fallas incluyendo pruebas de: conectividad, integridad de la información, integridad del

protocolo, saturación de la información, saturación de conexiones, tiempo de respuesta, históricos, funciones y diagnóstico.

7.4.5 Monitores de Contabilidad

El objetivo principal es cuantificar los recursos utilizados por los usuarios, ya sea de manera especifica, cuentas, grupos, proyectos, etc. o simplemente por departamentos.

Dentro de los servicios que pueden ser contabilizados están:

- Facilidades de comunicación: LANs, WANs, líneas, sistemas PBX, etc.
- Hardware: estación de trabajo, servidores, etc.
- Software y sistemas: aplicaciones y utilerías de software en servidores, centro de información, etc.
- Servicios: todas las comunicaciones comerciales y servicios de información disponibles para usuarios de la red.

8. SNMP

8.1 Antecedentes

Para lograr las funciones de administración de una red TCP/IP se utiliza en muchos casos ICMP como protocolo de apoyo (el programa mas utilizado es PING), sin embargo, conforme el número de nodos se incrementa en la red surge la necesidad del desarrollo de un protocolo para la administración de red que permita cubrir todos los requerimientos, es así como surgen los siguientes:

HEMS Sistema de administración de entidades, es la generalización de los primeros protocolos usados en la Internet, partiendo de HMP (Protocolo de Monitoreo de Host).

SNMP Protocolo simple de administración de red, es una versión aumentada del SGMP (Protocolo Simple de Monitoreo de Gateway).

CMIP Protocolo Común de Administración de la Información, incorpora servicio y estructura de base de datos que comienza a estandarizar OSI para la administración de red.

Donde cada uno se encuentra obligado a definir su respectiva MIB, como intento de estandarización entre SNMP y CMIP se creó CMOT, pero finalmente resulto inoperable. En el presente capítulo describiremos el protocolo SNMP.

8.2 Arquitectura del administrador de red en TCP/IP

La arquitectura de administración definida en redes basadas en TCP/IP se compone de:

- Estación de administración, que se compone de:
  - Conjunto de aplicaciones para el análisis y recuperación de fallas.
  - Interfaz con la cual el administrador de red puede monitorear y controlar la red.
  - Capacidad de traducir los requerimientos de administración de red dentro del monitor en uso y control remoto de los elementos en la red.

- Base de datos de la información de acuerdo a la MIB de los elementos de la red.

- Agente de administración, el cual responde a los requerimientos de información o acción de la estación de trabajo.
- MIB
- Protocolo de administración de red.

De manera general la administración en TCP/IP se realiza mediante los recursos de la red que son representados como objetos, donde cada objeto es esencialmente una variable que representa un aspecto del agente administrador, la colección de objetos es referida como MIB.

Una estación de administración que desempeña la función de monitoreo puede cargar un valor del objeto MIB, o puede causar una acción para tomar el lugar en un agente, o puede cambiar la configuración de un agente definiendo la modificación de un valor de variables específicas.

La estación de administración y los agentes son ligados por el protocolo de administración de red, el protocolo que emplea TCP/IP es SNMP, el cual cuenta con las funciones de:

Get: habilitar la estación de administración para conseguir el valor de los objetos en el agente.

Set: habilita la estación de administración para definir el valor de los objetos en el agente.

Trap: habilitar un agente para notificar a la estación de administración de eventos significantes sin que ésta lo solicite.

8.2.1 MIB SNMP

La MIB definida por TCP/IP es una colección de objetos, donde cada nodo del sistema mantendrá una MIB, que reflejará el estado de los recursos administrados. Teniendo como objetivo que los objetos utilizados para representar un recurso en particular sea el mismo en cada uno de los nodos; además de contar con un esquema de representación para lograr la interoperatividad.

8.2.2 SMI SNMP

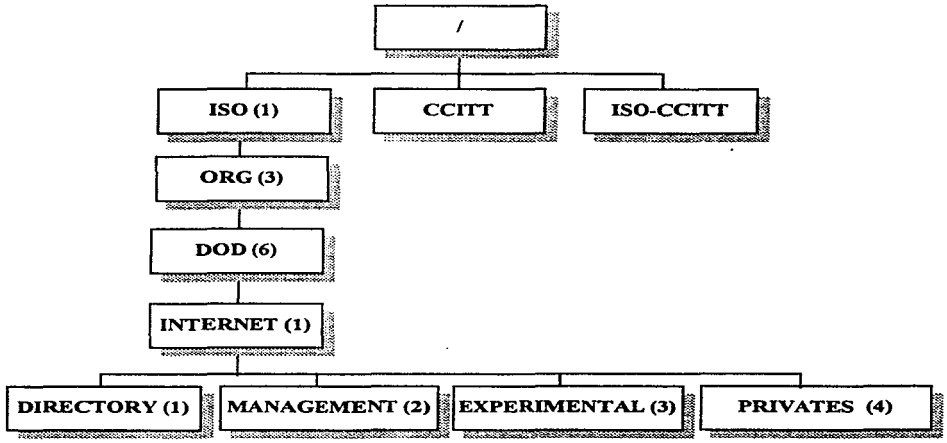
La estructura de la información de administración (SMI) identifica el tipo de datos que pueden utilizarse en la MIB y como los recursos dentro de ésta son representados y nombrados.

SMI define sólo tipos de datos escalares y arreglos bidimensionales de escalares, con el fin de conservar una estructura simple y estándar como lo es ASN.1, descrita en el Anexo A del presente trabajo.

8.2.3 Técnicas de Estandarización

Retomando las definiciones de SMI para proveer de una MIB que logre la administración de diversos equipos y tecnologías, se establecen las siguientes técnicas de estandarización:

- Definición de la estructura particular de una MIB. Definiendo para cada objeto un identificador (OBJECT IDENTIFIER) de tipo jerárquico como se observa en la figura siguiente:



Donde:

El primer nivel será la raíz (/) que identifica el uso del estándar ASN.1

En el segundo nivel organizaciones de definición de estándares internacionales, (ISO, CCITT).

Los niveles siguientes de nuestro interés son los que llegan a definir la administración del protocolo Internet, por lo que observamos que bajo ISO está el nodo correspondiente a organizaciones (ORG) y una de estas es el Departamento de Defensa de los Estados Unidos de América (DOD), llegando así al nodo de Internet, que establece que cualquier subárbol de éste será administrado por quien IAB (Internet Activities Board) designe.

SMI define bajo Internet cuatro nodos:

Directory, reservado para uso futuro con el directorio OSI.

Management, utilizado para definir los objetos identificadores en documentos aprobados por la IAB, MIB-I y MIB-II, ésta última extensión de la primera, que no pueden ser utilizadas de manera simultánea. Los objetos pueden ser definidos por la MIB de una de las siguientes formas:

- La MIB-II puede ser sustituida por completo por una nueva versión (MIB-III).
- La construcción de una MIB experimental puede ser construida para una aplicación en particular (token ring, FDDI, etc.).
- Extensiones privadas pueden ser agregadas desde el nodo private.

Experimental, usado para experimentos en la Internet.

Private, para la identificación de objetos definidos unilateralmente.

La definición de objetos de manera individual, incluyendo sintaxis y valor de cada objeto. Dentro de la clase UNIVERSAL se permite el uso de los siguientes tipos de datos:

- INTEGER (UNIVERSAL 2)
- OCTECT STRING (UNIVERSAL 4)

- NULL (UNIVERSAL 5)
- OBJECT IDENTIFIER (UNIVERSAL 6)
- SEQUENCE, SEQUENCE OF (UNIVERSAL 16)

Un objeto identificador es único para cada objeto y consiste de una secuencia de enteros conocida como subidentificadores, por ejemplo:

iso	*org*	*dod*	*internet*	*mgmt*	*MIB-II*	*tcp*	*tcpConnTable*
1	*3*	*6*	*1*	*2*	*1*	*6*	*13*

El identificador de la variable es: *1.3.6.1.2.1.6.13*

Los tipos de datos de aplicación son:

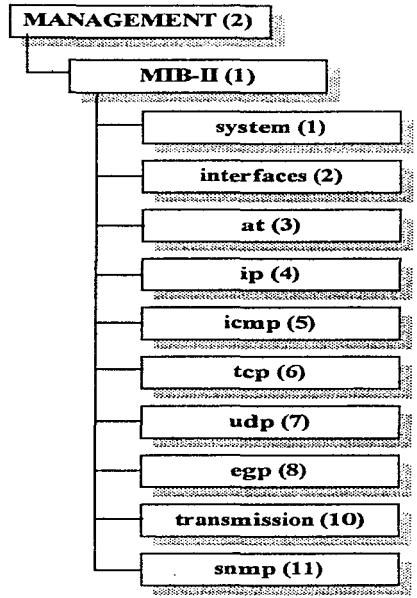
- NetworkAddress, utiliza CHOICE para la selección del formato de la dirección para cada protocolo de la familia (sólo se ha utilizado la dirección IP).
- IpAddress, dirección de 32 bits usando las especificaciones IP.
- Counter, entero no negativo, siempre en incremento con valor máximo de 2^{32} -1 (si llega a este valor se inicializa en cero).
- Gauge, entero no negativo con valor máximo de 2^{32} -1, si se alcanza el valor máximo lo mantiene hasta ser restaurado.
- Time Ticks, entero no negativo que cuenta el tiempo en centésimas de segundo.
- Opaque, soporta la capacidad para pasar información de manera arbitraria, la información es codificada usando el tipo OCTECT STRING, para la transmisión los datos pueden estar en cualquier formato definido por ASN.1 o cualquier otra sintaxis.

8.3 MIB-II

Entendiendo que la administración en SNMP se fundamenta en la estructura de la Base de Datos de objetos administrados, se debe considerar el contar con un conocimiento general de la clasificación que guardan estos objetos.

8.3.1 Grupos de Objetos MIB

Los objetos definidos para la administración de redes en MIB-II se clasifican en:



Donde cada grupo de la MIB-II se compondrá de varios objetos identificadores. Con el fin de tener una idea general de los valores que cada uno de éstos guarda, se muestra a continuación una breve explicación de cada grupo y a manera de ejemplo algunas de las variables que le pertenecen.

8.3.1.1 Grupo System(1)

Reside información general acerca del sistema administrado. Ejemplos de variables del grupo son:

OBJETO	ACCESO	DESCRIPCIÓN
sysDescr	RO	Datos de la entidad: Sistema Operativo, versión, hardware, etc.
sysUpTime	RO	Última vez en que se reinicializó el sistema.
sysLocation	RW	Localización física del nodo.

8.3.1.2 Grupo Interface(2)

Guarda información general acerca de las interfaces físicas de la entidad incluyendo información estadística y de configuración de los eventos ocurridos en cada interfaz. La MIB puede contener información adicional específica por el tipo de interfaz, en general los objetos de este grupo pueden ser utilizados para el monitoreo del desempeño y control de fallas. Ejemplos de variables del grupo son:

OBJETO	ACCESO	DESCRIPCIÓN
ifTable	NA	Lista de interfaces en la red
ifType	RO	Tipo de interfaz según su protocolo físico y/o de enlace, por ejemplo: (15)FDDI, (9)Token Ring, (6)Token Bus, etc.
ifSpeed	RO	Estimado de la capacidad de la interfaz para la transmisión de datos en bits por segundo.
ifPhysAddress	RO	Dirección física de la interfaz.
ifOutOctects	RO	Número total de octetos transmitidos en la interfaz, incluyendo caracteres de encapsulamiento.
ifOutDiscards	RO	Número de paquetes descartados sin error.

OBJETO	ACCESO	DESCRIPCIÓN
ifOutErrors	RO	Número de paquetes que podrían no ser transmitidos por error.

8.3.1.3 Grupo Address(3)

Este grupo consiste en una simple tabla donde a cada renglón corresponde una interfaz física del sistema, obteniendo así un mapeo de direcciones de red con sus correspondientes direcciones físicas. Este grupo carece de importancia en MIB-II y es incluido solamente para mantener la compatibilidad con nodos MIB-I, ya que en MIB-II la información de traducción de direcciones es provista por cada uno de los distintos protocolos de red; los motivos de dicho cambio son: a) la necesidad de soportar nodos multiprotocolo y b) la necesidad de mapear dos caminos. Ejemplos de variables del grupo son:

OBJETO	ACCESO	DESCRIPCIÓN
atTable	NA	Contiene la dirección de red y su correspondiente dirección física.
atPhysAddress	RW	Dirección física dependiente del medio.

8.3.1.4 Grupo IP(4)

Información de la implementación y operación de IP a un nodo, es implementado para sistemas finales (host) y sistemas intermedios (enrutadores). Contiene información utilizada para el monitoreo de desempeño y de fallas en la red. En este grupo existen tablas que contienen:

- Información de la dirección IP asignada a cada entidad con referencia a su dirección física, esta tabla no puede ser modificada, por lo que no puede utilizarse para cambiar direcciones IP.
- Información utilizada para el enrutamiento en internet y para el monitoreo de la configuración, además del control de enrutamiento.

Tablas de traducción de direcciones físicas a direcciones IP similar a las del grupo address pero agregando el tipo de mapeo utilizado. Ejemplos de variables del grupo son:

OBJETO	ACCESO	DESCRIPCIÓN
ipInHdrErrors	RO	Número de datagramas de entrada descartados por que la dirección IP destino no era válida para la entidad.
ipInUnknowProtos	RO	Número de datagramas direccionados localmente, recibidos de manera exitosa pero descartados porque la entidad no soporta el protocolo o es desconocido para ella.
ipInDelivers	RO	Número de datagramas de entrada exitosamente recibidos de protocolos IP.
ipReasmTimeout	RO	Número máximo de segundos que un fragmento que recibe espera para ser reensamblado.
ipReasmOKs	RO	Número de datagramas IP exitosamente reensamblados en la entidad.
ipAdEntNetMask	RO	Máscara de la subred asociado con la dirección IP en esta entidad.
ipNetToMediaTable	NA	Tabla de traducción de direcciones IP a mapeo de direcciones físicas.

8.3.1.5 Grupo ICMP(5)

Contiene información para la implementación y operación de ICMP, mediante el uso de contadores de mensajes que manda y recibe ICMP. Las variables de este grupo son:

OBJETO	ACCESO	DESCRIPCIÓN
icmpInMsgs	RO	Número total de mensajes ICMP que la

OBJETO	ACCESO	DESCRIPCIÓN
		entidad recibe.
icmpInErrors	RO	Número total de mensajes ICMP que la entidad recibe, pero son descartados por tener errores específicos de ICMP.
icmpInDestUnreachs	RO	Número total de mensajes ICMP que la entidad recibe, con destino inalcanzable.
icmpTimeExcds	RO	Número total de mensajes ICMP que la entidad recibe, con tiempo excedido.
icmpInParmprobs	RO	Número total de mensajes ICMP que la entidad recibe, con problema de parámetros.
icmpInSrcQuenchs	RO	Número total de mensajes ICMP que la entidad recibe, con señal de disminución de flujo.
icmpInRedirects	RO	Número total de mensajes ICMP recibidos redireccionados.
icmpInEcho	RO	Número total de mensajes ICMP recibidos con petición de eco.
icmpInEchoReps	RO	Número total de mensajes ICMP recibidos como respuesta a petición de eco.
icmpInTimestamps	RO	Número total de mensajes ICMP recibidos con petición de marca de tiempo.

OBJETO	ACCESO	DESCRIPCIÓN
icmpInTimestampsReps	RO	Número total de mensajes ICMP recibidos con respuesta a la petición de marca de tiempo.
icmpInAddrMask	RO	Número total de mensajes ICMP de entrada con petición de máscara de dirección
icmpInAddrMaskReps	RO	Número total de mensajes ICMP de entrada con respuesta a la petición de máscara de dirección.
icmpOutDestUnreachs	RO	Número total de mensajes ICMP enviados con destino inalcanzable.
icmpOutTimeExcds	RO	Número total de mensajes ICMP enviados con tiempo excedido.
icmpOutParmProbs	RO	Número total de mensajes ICMP enviados con problema de parámetros.
icmpOutSrcQuenchs	RO	Número total de mensajes ICMP enviados con señal de disminución de flujo.
icmpOutRedirects	RO	Número total de mensajes ICMP enviados con redireccionamiento.
icmpOutEchos	RO	Número total de mensajes ICMP enviados con petición de eco
icmpOutEchoReps	RO	Número total de mensajes ICMP enviados en respuesta a petición de eco.
icmpOutTimestamps	RO	Número total de mensajes ICMP enviados con marca de tiempo.
icmpOutTimestampReps	RO	Número total de mensajes ICMP enviados como respuesta a petición de

OBJETO	ACCESO	DESCRIPCIÓN
		marca de tiempo.
icmpOutAddrMask	RO	Número total de mensajes ICMP enviados con petición de máscara de dirección.
icmpOutAddrMaskReps	RO	Número total de mensajes ICMP de salida con respuesta a la petición de máscara de dirección.

8.3.1.6 Grupo TCP(6)

Información referente a TCP. Algunas de las variables de este grupo son:

OBJETO	ACCESO	DESCRIPCIÓN
tcpRtoMin	RO	Valor mínimo de tiempo para la retransmisión.
tcpRtoMax	RO	Valor máximo de tiempo para la retransmisión.
tcpMaxConn	RO	Límite del total de conexiones TCP que puede soportar la entidad.
tcpInSeg	RO	Número total de segmentos recibidos incluyendo los que contengan error.
tcpRetransSegs	RO	Número total de segmentos retransmitidos.
tcpConnLocalPort	RO	Número de puertos locales para esta conexión
tcpInErrs	RO	Número de segmentos recibidos con error.

8.3.1.7 Grupo UDP(7)

Información referente a la implementación y operación de UDP, conteo de mensajes enviados de este protocolo y puntos en los que una aplicación recibe

estos datagramas para cada usuario UDP, la tabla contiene la dirección IP y el puerto UDP. Ejemplo de las variables del grupo son:

OBJETO	ACCESO	DESCRIPCIÓN
udpInDatagrams	RO	Número de datagramas UDP enviados por usuarios UDP.
udpNoPorts	RO	Número total de datagramas UDP recibidos para los cuales no había una aplicación con el puerto destino.
udpOutDatagrams	RO	Número de datagramas UDP enviados por la entidad.

8.3.1.8 Grupo EGP(8)

Además de la información propia del protocolo contiene información de los gateways vecinos conocidos como entidad. Ejemplo de variables EGP son:

OBJETO	ACCESO	DESCRIPCIÓN
egpInMsg	RO	Número de mensajes EGP recibidos sin error.
egpOutMsgs	RO	Número total de mensajes EGP generados localmente.
egpNeighTable	NA	Tabla de vecinos EGP
egpNeighMode	RO	Modo de poleo utilizado para la entidad: (1)activo, (2)pasivo.

8.3.1.9 Grupo Transmision(10)

Pretende contener objetos que provean detalles del medio de transmisión de cada interfaz del sistema, actualmente no se cuenta con una definición específica para cada tipo de interfaz, sin embargo existen varios objetos definidos mediante RFCs como parte de la MIB experimental, tal es el caso de Token Ring (RFC 1239), Bus (RFC 1239) y FDDI (RFC 1512).

8.3.1.10 Grupo SNMP(11)

Este grupo contiene información relevante sobre la implementación y operación de SNMP, protocolo que en el siguiente capitulo explicaremos a detalle. Ejemplo de variables SNMP son:

OBJETO	ACCESO	DESCRIPCIÓN
snmpInPkts	RO	Número de mensajes SNMP que la entidad a recibido.
snmpOutpkts	RO	Número de paquetes SNMP que la entidad ha enviado.
snmpInBadCommunityUses	RO	Número de mensajes SNMP que recibe la entidad con el nombre de la comunidad errónea.
snmpInTraps	RO	Número de mensajes SNMP trap (mensajes de alerta) aceptados y procesados por la entidad.

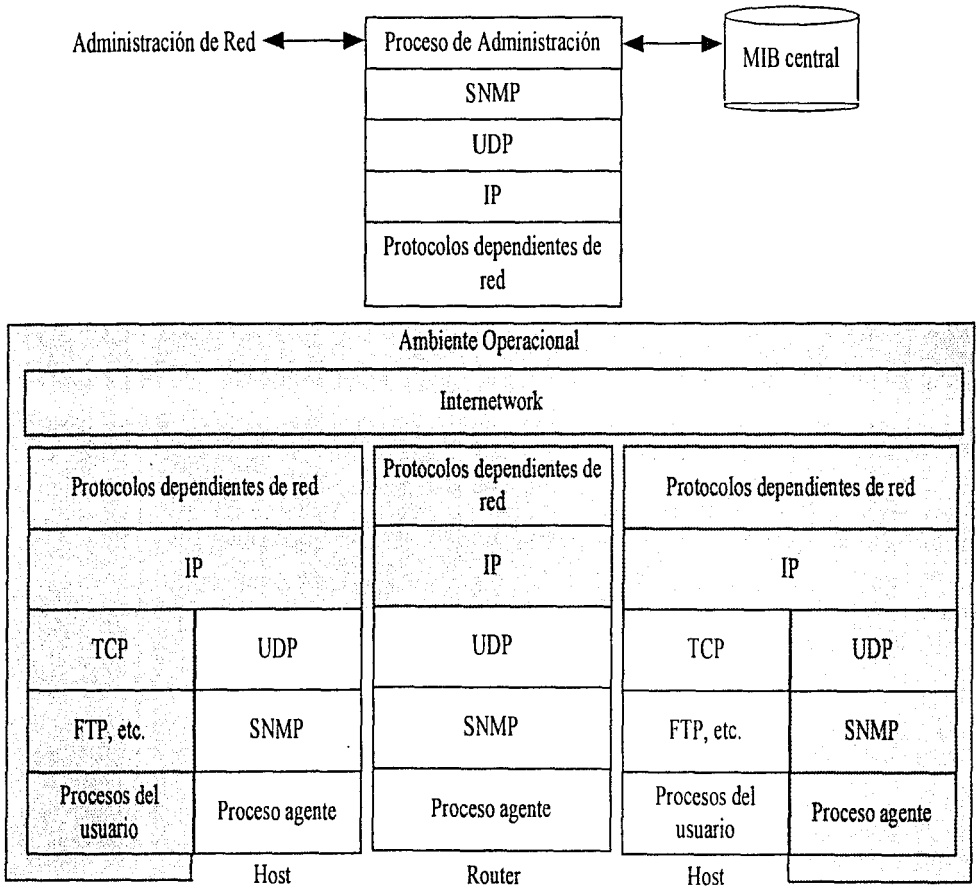
8.3.2 Consideraciones

Es importante aclarar que SNMP está habilitado para la realización del monitoreo de los objetos definidos en su MIB, soportándolos desde el nivel de transporte, conociendo así la manera en como establecen las conexiones y conteo de paquetes, pero no puede administrar recursos propios del sistema, por ejemplo: servidores de correo e impresión.

Además comparado con el esquema de OSI para la administración, sacrifica en gran medida funcionalidad por simplicidad, ya que existe la necesidad de verificar concordancia entre el objeto definido en la MIB y lo que el equipo utilizado refleja, para evitar posibles errores. Sin embargo por su diseño cubre la necesidad de crecer conforme a las necesidades de nuevos objetos.

8.4 Arquitectura del Protocolo de Administración

SNMP fue desarrollado en el nivel de aplicación del modelo TCP/IP, para operar sobre datagramas UDP, una configuración típica de protocolos para SNMP es:


Administración de Red
Proceso de Administración
SNMP
UDP
IP
Protocolos dependientes de red
MIB central
Ambiente Operacional
Internetwork
Protocolos dependientes de red
IP
TCP
UDP
FTP, etc.
SNMP
Procesos del usuario
Proceso agente
Host
Protocolos dependientes de red
IP
UDP
SNMP
Proceso agente
Router
Protocolos dependientes de red
IP
TCP
UDP
FTP, etc.
SNMP
Procesos del usuario
Proceso agente
Host

Donde:

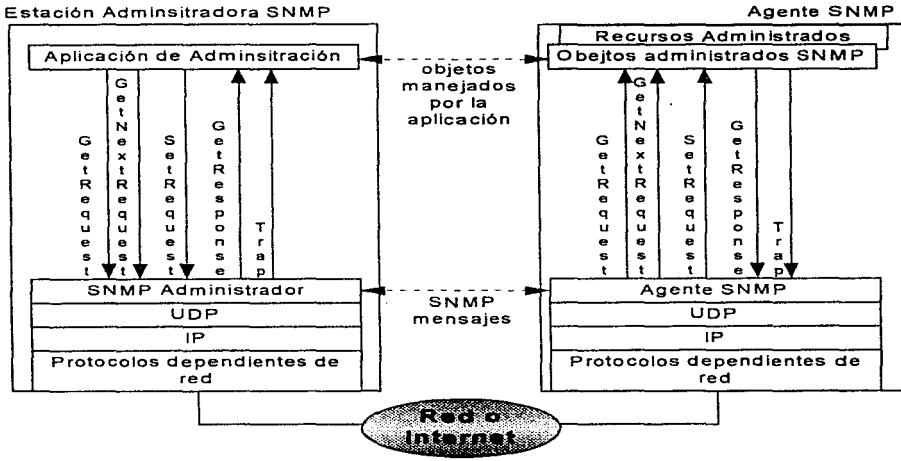
La estación de administración, necesita de un administrador de procesos para el control de acceso a la MIB y provee de una interfaz para el administrador de red.

El proceso administrador activa al administrador de red para usar SNMP, implementado como aplicación que emplea UDP, IP y el protocolo relevante dependiendo del tipo de red (Ethernet, FDDI, X.25).

Cada agente pudiera sólo implementar SNMP, UDP e IP.

Los tipos de mensaje SNMP son:

- GetRequest Función get.
- GetNextRequest Función get.
- SetRequest Función set.
- GetRespons Función get.
- Trap Función trap.



SNMP tiene varias características no típicas de las aplicaciones distribuidas, basado en un esquema de relación n a n entre la estación administradora y sus respectivos agentes ya que una entidad administradora recibe información de las diferentes entidades donde radican sus agentes y dado que pueden existir varias estaciones administradoras un mismo agente puede ser manipulado por dos o mas estaciones administradoras. En el agente se define cuales serán estas estaciones administradoras existiendo para cada una un nombre identificador conocido con el nombre de comunidad.

Los siguientes tres aspectos son definidos considerando que cada agente controla la MIB en su propia entidad.

- Servicio de verificación: acceso restringido para determinar las estaciones que podrán administrarla. Todos los mensajes (get request) enviados desde la estación administradora incluyen un nombre de comunidad el cual servirá como password y el mensaje es tomado como válido si coincide con el definido en el agente. No incluye encriptación y desencriptación.

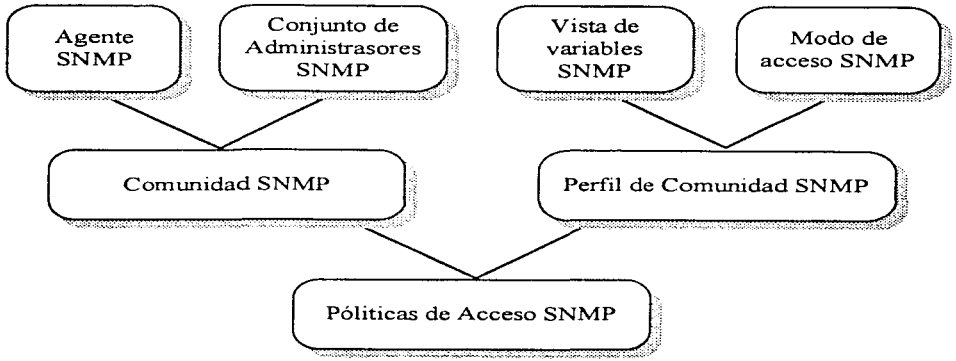
- Políticas de acceso: el agente puede dar diferentes privilegios de acceso a cada administrador. Este acceso puede controlar dos aspectos:

  - Vistas de la MIB SNMP: subconjunto de objetos dentro de una MIB, definiendo diferentes vistas para diferentes comunidades. Los objetos en la vista no deben pertenecer a un subárbol simple de la MIB.

  - Modo de acceso a SNMP: cada comunidad tiene definido su modo de acceso RO (Read Only) o RW (Read and Write). Al conjunto de vista y modo de acceso de la comunidad se le conoce como perfil de la comunidad.

- Servicio Proxie: el agente realiza funciones de proxie para otra estación administrada.

La combinación de una comunidad SNMP y un perfil de comunidad SNMP es referida como una política de acceso SNMP.

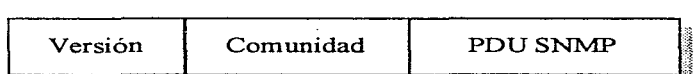


8.5 Identificación de Instancias

La manera en como SNMP referenciará a cada uno de los objetos será mediante el identificador que SMI define para cada objeto de la MIB, sin embargo para los objetos que hacen referencia a entidades particulares de la red no establece la manera de referenciarlos (ya que la manera de archivar sus valores pertenece a cada mecanismo del protocolo particular utilizado). SNMP define dos técnicas para estos casos: acceso serial y acceso aleatorio. En donde ambas involucran el uso de objetos indexados; por otra parte para referenciar a los objetos que no pertenecen a tablas agrega al final un cero para identificar el tipo de entidad.

8.6 Formatos SNMP

Cada mensaje utilizado para el intercambio de información está compuesto de:



Para Get Request PDU, GetNextRequest PDU y SetRequest PDU.

Tipo de PDU	Identificador de requerimiento	0	0	Variable ligada

Para GetResponse PDU

Tipo de PDU	Identificador de requerimiento	0	0	Variable ligada

Para Trap PDU

Tipo de PDU	Objeto generador	Dirección agente	Trap Genérico	Marca de tiempo	Variable ligada

Donde la variable ligada es:

Nombre 1	Valor 1	Nombre 2	Valor 2	...	Nombre n	Valor n

Y los campos significan:

Versión: versión de SNMP

Comunidad: nombre de la comunidad utilizado para la verificación

Identificador de requerimiento: utilizado para identificar cada requerimiento con un único ID.

Estado de error: Indica alguna irregularidad ha ocurrido en el procedimiento de una petición.

0- No error

1- Too Big

2- NoSuchName

3- Bad Value

4- Read Only

5- GenErr

Índice del error: cuando el estado de error es 5, este campo agrega información sobre la variable que causo la alteración.

Variable ligada: una lista de nombre de variables y sus correspondientes valores, en algunos casos es nulo (GetRequest PDU).

Objeto generador: Tipo de objeto que generó el trap (sysObjecyID).

Dirección agente: dirección del agente que generó el trap.

Trap General: Su valor puede ser:

0- ColdStart - La entidad SNMP es reinicializada ella misma inesperadamente y por lo tanto la configuración del protocolo puede ser alterada.

1- WarmStart - Reinicialización del protocolo sin alterarlo.

2- LinkDown - Señala una falla en alguna de las ligas de comunicación, indicando en el mensaje el número de interfaz de falla.

3- LinkUp - Señala que uno de los agentes de comunicación levanta sus servicios.

4- Authentication Failuire - enviado por la entidad cuando ha recibido un mensaje con falla de verificación.

5- EgpNeighborLoss - Indica la perdida de conexiones punto a punto con un vecino EGP.

6- EnterpriseSpecific - Especifica que se trata de un trap específico de la entidad, el cual será explicado en el campo de trap específico.

Trap Específico: es propio de la entidad, su valor es codificado.

Marca de tiempo: lapso de tiempo entre la última reinicialización de la entidad en red y la generación del trap, contiene el valor de sysUpTime.

8.6.1 Transmisión de un mensaje SNMP

Para la transmisión de un mensaje SNMP de una entidad a otra:

- Se construye un PDU usando la estructura definida ASN.1
- El PDU pasa al proceso de verificación junto con la dirección de la fuente, el destino y el nombre de la comunidad, este proceso incluye encriptación de la información y agrega un código de verificación, regresando el resultado.
- El protocolo construye un mensaje que consiste del campo de versión, comunidad y resultado del paso anterior.
- Este objeto es codificado y pasado al servicio de transporte.

8.6.2 Recepción de un mensaje SNMP

La entidad que recibe el mensaje

- Realiza un chequeo básico de sintaxis del mensaje descarta aquellos contengan error.
- Verifica el número de versión y descarta los que no concuerden
- Revisa el nombre del usuario, las direcciones fuente y destino para dar servicio de verificación si ésta falla, envía un mensaje trap a quien lo envió.
- Si la verificación es exitosa,, utiliza el nombre de la comunidad para revisar que las políticas de acceso se cumplan.

8.6.3 PDU GetRequest

Enviado desde la entidad administradora solicitando del agente el o los valores de variables específicas, ya sea con permiso RO o RW.

Al ser recibido por un agente lo responde con un PDU GetResponse mediante la siguiente lógica:

SI *el objeto (variable ligada) no concuerda con alguno de los disponibles para la obtención de su valor.*

ENTONCES *envía el GetResponse con el estado de error (noSuchName) y si respectivo índice.*

SI NO, SI *el valor del objeto es muy grande*

ENTONCES *GetResponse = TooBig*

SI NO, SI *El valor del objeto es inaccesible por alguna razón*

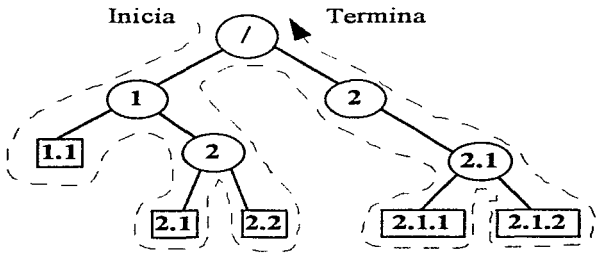
ENTONCES *GetResponse = error genérico y su respectivo índice.*

SI NO *GetResponse = valor de los objetos solicitados*

8.6.4 PDU GetNextRequest

Funciona de manera similar a la del GetRequest, sólo que el valor que regresa es el consecutivo (según el orden lexicográfico de la MIB) al especificado en la requisición, esto permite a la estación administradora descubrir la estructura de una MIB en algún dispositivo.

La manera en que se recorrerá el árbol en la MIB se ilustra en el siguiente esquema:



8.6.5 PDU SetRequest

Solamente se puede ser utilizado desde una estación administradora que define valores de la MIB, en algún agente, el cual enviará un GetResponse con el nuevo valor asignado; además de actualizaciones a las variables de la MIB puede definir su borrado físico o indicando valores inválidos (ejemplo: ipRouteDest). Este comando esta sujeto a los permisos propios de cada variable en la MIB y los respectivos permisos que el administrador tenga de estos en el agente, de tal forma que si una variable no se encuentra dentro del dominio del administrador o la variable es definida como RO la respuesta será un mensaje de error (noSuchName).

8.6.6 PDU Trap

Este tipo de mensaje es enviado desde cualquier agente hacia la estación administradora, utilizada para proveer a ésta de una notificación asíncrona sobre algún evento significativo.

8.7 Soporte en el Nivel de Transporte

Dado que SNMP es un protocolo de aplicación el servicio de transporte será tomado de su plataforma principal, TCP/IP. SNMP utiliza UDPs para el envío de sus mensajes, el cual, es un protocolo sin conexión y mediante arquitectura OSI con CLTS (servicio de transporte sin conexión).

8.8 Estación Administradora

Para la elección de la estación administradora se debe considerar (según estudio realizado por Wikinson y Capen en 1992) que cumplá con:

- Soporte de una MIB extendida, que permita el soporte de las distintas MIBs definidas en cada uno de sus agentes.
- Ser una interfaz amigable, proporcionando al administrador facilidad y potencia para conocer el estado de sus agentes, dentro de un ambiente de ventanas y gráfico.
- Descubrimiento automático de agentes existentes en la comunidad o dominio.
- Programación de eventos que permitan informar al administrador de manera llamativa cualquier alteración que se presente en los agentes.

- Control avanzado de la red, permitiendo al administrador optimizar el funcionamiento de la red mediante la correcta configuración de los dispositivos.
- Administración controlada por objetos, ya que éste puede ser fácilmente actualizado para correr en múltiples protocolos de administración simultáneamente.
- Representación gráfica de la topología de red.

8.9 Frecuencia de Poleo

Para conocer el estado real de cada uno de los dispositivos en la red, es necesario monitorear el valor de cada variable MIB de interés con determinada frecuencia, principalmente si se desean realizar estudios estadísticos. Aunque el tiempo en el que se puede llevar a cabo depende en gran medida de la capacidad del equipo administrador y el tráfico en la red; es importante considerar que un solo agente será atendido a la vez. Por lo tanto la siguiente ecuación resulta de sumo interés.

$$N \leq \frac{T}{\Delta}$$

Donde:

T = Número de agentes

Δ = Promedio de tiempo que tarda el proceso en un poleo simple, el cual depende de:

- Tiempo para generar una respuesta en la estación administradora.
- Retardo de la red desde el administrador al agente.
- Tiempo de proceso en el agente para interpretar el mensaje.
- Tiempo de proceso en el agente para generar la respuesta.
- Retardo de la red del agente al administrador
- Tiempo de proceso del administrador para recibir e interpretar el mensaje.

- Número de requerimientos y/o respuestas intercambiados para obtener todos los requerimientos de información dese el agente.

Además de los factores mencionados en la fórmula anterior debe considerarse fuertemente el tráfico en la red que genere el mismo poleo, ya que este no debe disminuir el buen desempeño de ésta.

8.10 Limitaciones de SNMP

Por las características propias de SNMP se listan a continuación las siguientes limitantes según BenArtzi, Chanday Warrier (1990).

- Limita considerablemente el número de dispositivos a administrar por la disminución del desempeño con el poleo a cada agente.
- Los trap no requieren de la entidad administradora una respuesta de recepción del mensaje, por lo tanto no se garantiza el aviso de algún mensaje de alerta del agente.
- Dado el bajo nivel de verificación es mas utilizado para monitorear que para controlar.
- SNMP no soporta implementación de comandos (llamadas a programas con parámetros, condiciones y estado) siendo de manera indirecta la definición del valor de algún objeto en el agente.
- La MIB de SNMP no soporta queries sofisticados (basados en valores de objetos o tipos).

SNMP no soporta comunicaciones de administrador a administrador, por ejemplo no cuenta con mecanismos que permitan a un sistema administrador conocer de otros dispositivos administrados por otro sistema administrador.

9. SNMPv2

El origen de SNMP versión 2 (SNMPv2) se encuentra en los documentos que definen a S SNMP (Seguridad de SNMP) y SMP que fueron publicados como estándares para la administración de redes a mediados de 1992, mencionando y cubriendo deficiencias de SNMP. El objetivo del nuevo protocolo era ser soportado por TCP/IP y también ser llevado a OSI. Para octubre de 1992 se establece así oficialmente el inicio del trabajo sobre el desarrollo de SNMPv2, basado en SMP, para aspectos de protocolo y administración de información y S SNMP, para aspectos de seguridad.

9.1 Seguridad SNMP (S SNMP)

La seguridad de SNMP se basa principalmente en el nombre de la comunidad que se incluye en cada encabezado de sus propios mensajes, por tanto resulta un protocolo sumamente inseguro, ya que una vez enviado el mensaje puede copiarse este nombre para realizar cualquier modificación. Con el objetivo de cubrir los requerimientos de seguridad y como medida correctiva surgen una serie de RFCs (1351, 1352, 1353 y 1321) en 1992 conocidos en su conjunto como Seguridad SNMP (S SNMP), los cuales especifican un encabezado del mensaje diferente.

9.1.1 Requerimientos de Seguridad

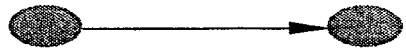
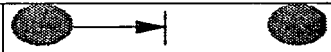
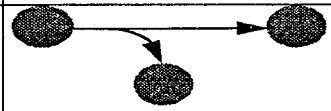
- Confidencialidad de la información, es decir la información en un sistema computacional y de transmisión de información (impresión, despliegue, etc.) debe ser sólo accesible para el personal autorizado.
- Autenticidad: el origen del mensaje debe ser correctamente identificado, con garantía de que la identidad no es falsa.
- Integridad del sistema de cómputo y de la transmisión de información, ya que ésta debe ser modificada únicamente por la parte autorizada para ello.
- Disponibilidad en el momento que se requiera.

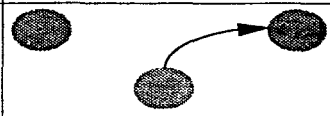
9.1.2 Conceptos básicos de seguridad

Amenaza de seguridad: cualquier acción que comprometa la seguridad del dueño de la información por una organización.

Servicio de seguridad: servicio de comunicación que apoya los sistemas de proceso de información y transferencia de las organizaciones.

Mecanismos de seguridad, diseñados para detectar, prevenir o recobrar por una amenaza de seguridad.

Amenazas de la Información		
Flujo normal de la información 		
Interrupción		Alguna parte del sistema (de cómputo o comunicaciones) es destruido o comienza a ser inaccesible o inutilizable. Amenaza la disponibilidad. Clasificada como amenaza activa.
Intercepción		Una entidad no autorizada accesa la información o recursos del sistema; publicación del contenido de los mensajes y el análisis del tráfico, esta amenaza es muy difícil de detectar. Amenaza la confidencialidad de la información. Clasificada como amenaza pasiva.
Modificación		Una entidad no autorizada cambia valores de la información del sistema o de los mensajes enviados. Amenaza la integridad de la información. Clasificada como

Amenazas de la Información		
		amenaza activa.
Mascarada		Una entidad no autorizada agrega objetos al sistema o información. Amenaza de autenticidad de la información. Clasificada como amenaza activa.

9.1.3 Mecanismos de Seguridad

Llamaremos mecanismos de seguridad de S SNMP a todas las definiciones especificadas para lograr la seguridad en un esquema de administración SNMP, éstos son:

- Integridad de la información: Utilización de un algoritmo de mensaje-compendio que carga sobre los mensajes SNMP un compendio de información que garantiza que la información no ha sido modificada. El mensaje solamente incluye una marca de tiempo basada en los relojes de sincronización del administrador y el agente. El receptor del mensaje usa esta marca de tiempo para verificar que el mensaje es reciente y determinar la secuencia apropiada de mensajes múltiples. S SNMP utiliza el algoritmo MD5.

- Verificación del origen de la información, el mensaje incluye un prefijo con un valor *a priori* dependiente del emisor y el receptor, el valor no se incluye en el mensaje.

- Confidencialidad de la información, una porción del mensaje SNMP es encriptada mediante un algoritmo de encriptación simétrica, DES (encriptación estándar de la información).

9.1.4 Modelo Administrativo

Es importante recordar que S SNMP no esta sujeto a SNMP, sino que SNMP debe considerar las especificaciones de S SNMP para cubrir las deficiencias en cuanto a seguridad se refiere; por tal motivo las siguientes diferencias en el modelo administrativo son de gran importancia.

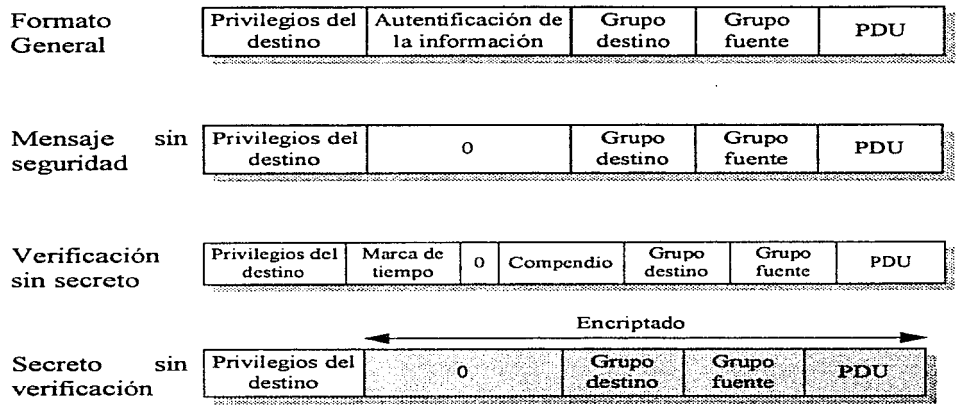
La verificación y la integridad del mensaje dependen de la fuente, esta es responsable de incluir información en cualquier mensaje que asegure que el origen del mensaje es auténtico, además de desarrollar las funciones necesarias para conservar la integridad del mensaje.

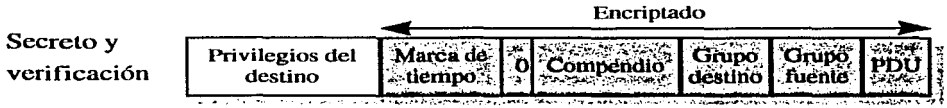
Por otro lado el destinatario se encargará de la encriptación necesaria para la verificación y confidencialidad del mensaje; en el momento de encriptar la información se debe considerar el destino como único elemento capaz de desencriptarla.

El modelo administrativo dicta que todas las entidades SNMP mantienen una base de datos local que representa todos los grupos SNMP, controlada mediante grupos (entidades con funciones similares) y vistas de la MIB (subconjuntos de objetos en la MIB con permisos similares) .

9.1.5 Mensajes S SNMP

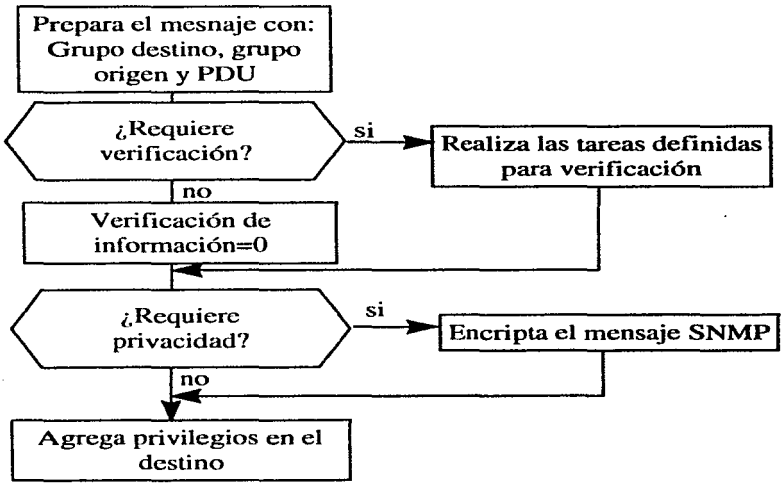
Al igual que SNMP el envío de información se realiza mediante la utilización de mensajes, cada uno de estos consta de un encabezado que varia según los requerimientos a cubrir y uno de los cinco PDUs definidos en SNMP, de tal forma que de manera esquemática se observa:



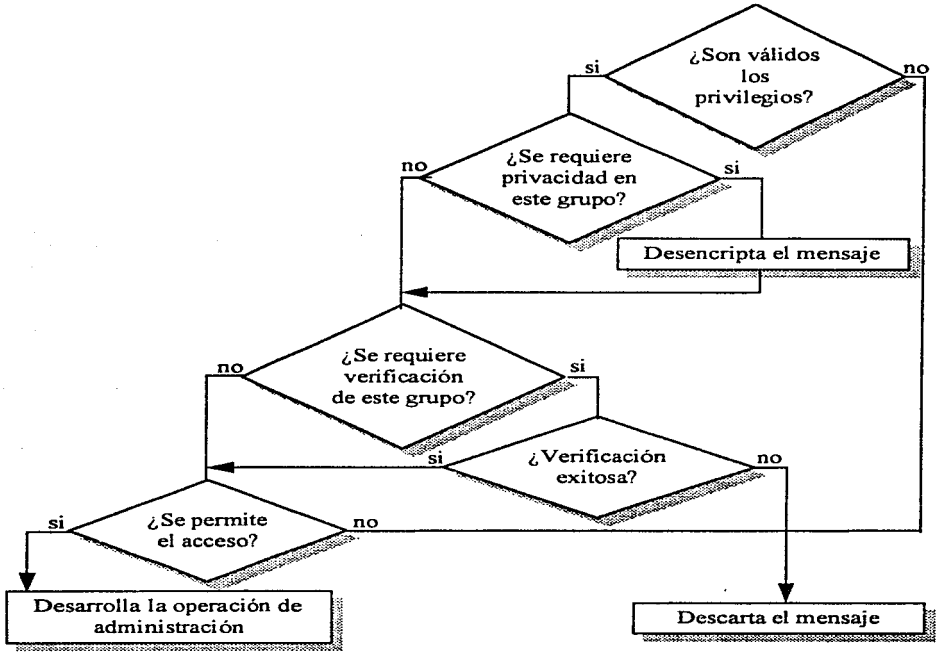


Donde: Grupo destino y fuente son las entidades con funciones similares de destino y fuente, el campo que se indica con un cero, solamente es utilizado cuando se requiere la verificación del mensaje, se incluye entonces el campo de marca de tiempo y el compendio de verificación; privilegios del destino representa el objeto específico del destino.

9.1.5.1 Transmisión de Información



9.1.5.2 Recepción de mensajes



En general los mensajes se transmiten y se reciben de manera similar que en SNMP pero verifican las funciones implementadas, ya sea de verificación o encriptación, como lo muestran los esquemas anteriores.

9.2 Alcances SMP

Define las siguientes categorías de alcance:

Visión.- Diseñado para administrar los recursos de la red de manera arbitraria, por lo que puede ser utilizado para la administración de aplicaciones, sistemas de administración y comunicación de administrador a administrador.

Tamaño, velocidad y eficiencia.- Con el objetivo principal de conservar simplicidad. Pretende el desarrollo de implementaciones chicas y rápidas. Desarrolla la capacidad de transferencia en bloque para mejorar el intercambio de cantidades grandes de información.

Seguridad y privacidad:- Incorpora las funciones dadas por S SNMP.

Desarrollo y compatibilidad.- diseñada para correr sobre aplicaciones TCP/IP, OSI y otras arquitecturas de comunicaciones.

Además de interoperatividad con plataformas SNMP usando un subconjunto de las capacidades de SMP.

SNMPv2 provee un sistema de administración bajo un esquema centralizado o distribuido, en este último una entidad puede jugar el papel de administrador y agente, quedando disponible la información almacenada (de la entidad misma o un agente de ésta) para un administrador de mayor jerarquía.

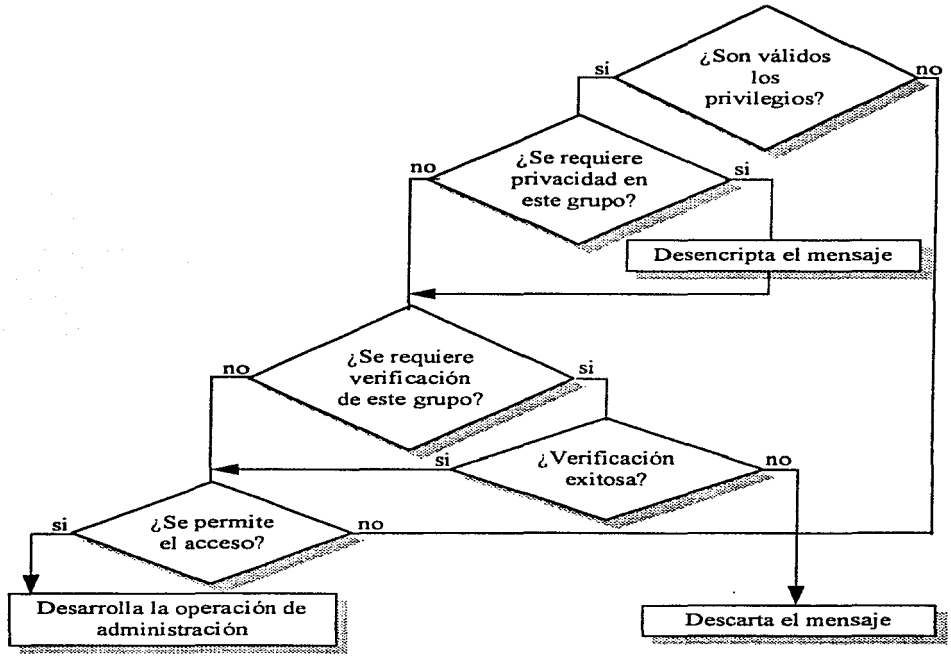
9.2.1 SMI

La definición de objetos se basa en la notación ASN.1 y con definición de tipos similares a los de SMI SNMP, sin embargo entre las definiciones encontramos que define los enteros y contadores a 32 y 64 bits respectivamente, sin que estos últimos tengan un valor inicial; agrega además una cláusula opcional de unidad, para referenciar a la unidad utilizada por ese objeto (ej. segundos, milisegundos, etc.). Establece una nueva categoría de permisos de lectura-creación (RC) para agregar tantos objetos a la MIB como se requiera en los grupos que así lo permitan.

Al igual que SNMP define objetos como simples escalares e información compleja en tablas, definiendo dos categorías:

- Las que prohiben la creación y borrado de entradas por el administrador, controladas por el agente con permisos máximos de lectura-escritura RW.
- Las que permiten la creación y borrado de entradas por el administrador, estas tablas pueden inicializarse sin datos.

9.1.5.2 *Recepción de mensajes*



En general los mensajes se transmiten y se reciben de manera similar que en SNMP pero verifican las funciones implementadas, ya sea de verificación o encriptación, como lo muestran los esquemas anteriores.

9.2 Alcances SMP

Define las siguientes categorías de alcance:

Visión.- Diseñado para administrar los recursos de la red de manera arbitraria, por lo que puede ser utilizado para la administración de aplicaciones, sistemas de administración y comunicación de administrador a administrador.

Tamaño, velocidad y eficiencia.- Con el objetivo principal de conservar simplicidad. Pretende el desarrollo de implementaciones chicas y rápidas. Desarrolla la capacidad de transferencia en bloque para mejorar el intercambio de cantidades grandes de información.

Seguridad y privacidad:- Incorpora las funciones dadas por S SNMP.

Desarrollo y compatibilidad.- diseñada para correr sobre aplicaciones TCP/IP, OSI y otras arquitecturas de comunicaciones.

Además de interoperatividad con plataformas SNMP usando un subconjunto de las capacidades de SMP.

SNMPv2 provee un sistema de administración bajo un esquema centralizado o distribuido, en este último una entidad puede jugar el papel de administrador y agente, quedando disponible la información almacenada (de la entidad misma o un agente de ésta) para un administrador de mayor jerarquía.

9.2.1 SMI

La definición de objetos se basa en la notación ASN.1 y con definición de tipos similares a los de SMI SNMP, sin embargo entre las definiciones encontramos que define los enteros y contadores a 32 y 64 bits respectivamente, sin que estos últimos tengan un valor inicial; agrega además una cláusula opcional de unidad, para referenciar a la unidad utilizada por ese objeto (ej. segundos, milisegundos, etc.). Establece una nueva categoría de permisos de lectura-creación (RC) para agregar tantos objetos a la MIB como se requiera en los grupos que así lo permitan.

Al igual que SNMP define objetos como simples escalares e información compleja en tablas, definiendo dos categorías:

- Las que prohiben la creación y borrado de entradas por el administrador, controladas por el agente con permisos máximos de lectura-escritura RW.
- Las que permiten la creación y borrado de entradas por el administrador, estas tablas pueden inicializarse sin datos.

9.2.2 Operaciones del protocolo

9.2.2.1 Tipos de acceso

Existen tres tipos de acceso para la administración de la información, utilizados para recibir o modificar información asociada con el dispositivo administrado, éstos son:

- Requerimiento de respuesta del administrador al agente.
- Requerimiento de respuesta del administrador a otro administrador.
- Sin confirmación del agente al administrador.

9.2.2.2 Formatos

SNMPv2 utiliza los formatos de mensajes definidos por S SNMP y los PDUs GetRequest, GetNextRequest y SetRequest de PDUs de SNMP, el Trap es similar al de SNMP pero utiliza un formato parecido al de GetBulkRequest; este último y el PDU InformRequest son agregados por SNMPv2.

- GetBulkRequest PDU

  Tiene como propósito principal reducir el envío de paquetes para requisiciones largas. Su formato es el siguiente:

Tipo de PDU	Identificador del requerimiento	n (núm. de variables sin repetición)	m (núm. máximo de repeticiones)	Variables ligadas

  Los campos n y m son utilizados para la implementación del algoritmo que permitirá conocer el valor de las variables ligadas y el número de sucesores a cada una de estas variables.

- InformRequest PDU

  Utilizado para intercambiar información entre administradores con un formato similar al utilizado por GetRequest, como variables ligadas al PDU incluye siempre la variable que indica el tiempo transcurrido desde

la última inicialización de la entidad (sysUpTime), SNMPv2EventID, el cual es un objeto definido en la MIB administradora que identifica el tipo de evento y las variables que se le especifiquen.

9.3 MIB

La MIB de SNMPv2 consiste de cinco grupos principales:

- Grupo Estadístico SNMPv2

  Información básica relativa al tráfico para la operación de SNMPv2.

- Grupo Estadístico SNMP

  Información básica relativa al tráfico para la operación de SNMP.

- Grupo de Objetos de Recursos

  Actúa en el agente para describir recursos de éste, que son de configuración dinámica manejada por un administrador.

- Grupo Trap

  Este grupo guarda información de los traps enviados (tiempos y tipos) al administrador.

- Grupo Set

  Sirve para realizar operaciones múltiples en algún objeto de la MIB, desde el administrador.

9.3.1 MIB de Administrador a Administrador

En esta parte de la MIB, los objetos describen el control de una entidad SNMPv2 que actúa como administrador; consta de dos grupos:

- Grupo de Alarma

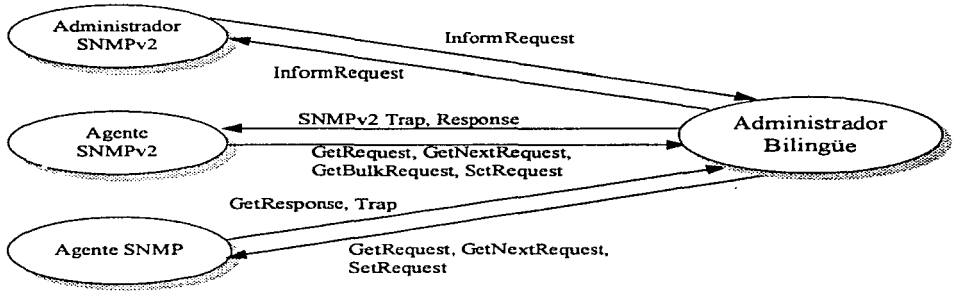
  Este grupo es muy similar al grupo de alarma de RMON, es una colección de objetos que permite la descripción y configuración de tareas desde una entidad SNMPv2 con doble papel (administrador y agente).

- Grupo de Evento

Es también similar a su correspondiente en RMON, soporta definiciones de eventos en la configuración y descripción del disparo de eventos bajo ciertas condiciones.

9.4 SNMP-SNMPv2

A pesar del esfuerzo de lograr la integración transparente de sistemas SNMP con SNMPv2, se hicieron necesarias algunas modificaciones que permitieran lograr la coexistencia de administradores y agentes SNMPv2 y agentes SNMP. SNMPv2 cuenta con una guía técnica para poder llevar a cabo la integración, donde se especifica que la información a administrar y las operaciones del protocolo posibles, estableciendo el menor número de cambios en la MIB SNMP. La utilización de un agente proxy que comunique al administrador SNMPv2 con agentes SNMP y cambios en la seguridad SNMP para que soporte las funciones de SNMPv2. El esquema de funciones mediante el envío de PDUs se muestra en la figura siguiente:



9.4.1 Diferencias SNMP- SNMPv2

Como principales diferencias entre la versión previa a SNMPv2 se detectan:

La ampliación considerable de los tipos de información y el permiso para asociar documentación con el objeto. Se permite además, la creación y borrado conceptual de renglones en la tabla de MIB. Distinguiendo dos partes dentro de ésta: la información relativa al tráfico de operación por el protocolo mismo y la información relacionada a la configuración de SNMPv2. Por otro lado, su MIB

apoya una arquitectura de administración distribuida, muy similar a las definidas por la RMON.

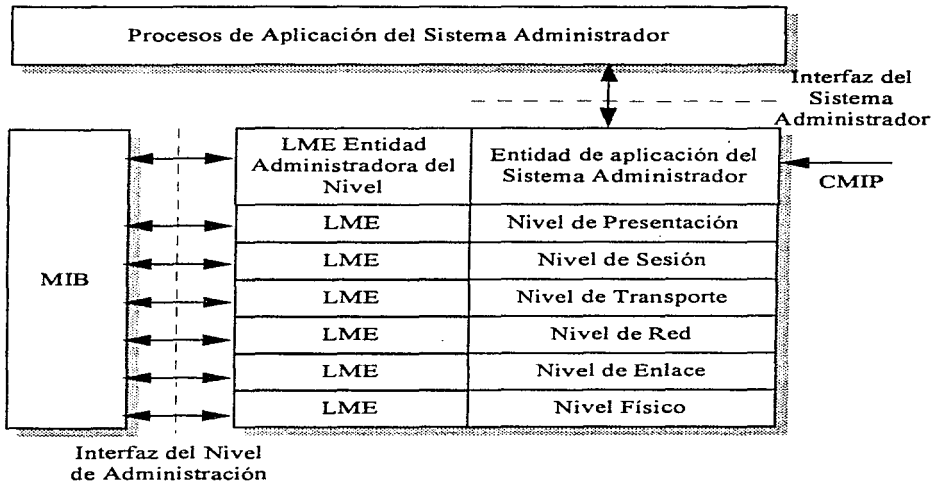
Además SNMPv2 incluye dos nuevos PDUs GetBulkRequest e InformRequest para el envío de información a grandes volúmenes y entre administradores, respectivamente. Dejando el esquema de seguridad muy similar a lo establecido por S SNMP.

10. SISTEMA DE ADMINISTRACIÓN OSI

Para el desarrollo de estándares internacionales en la administración de red ISO (Organización Internacional para estandarización) y CCITT (Comite Consultivo en Telegrafía y Telefonía) han trabajado conjuntamente y logrado así la definición del Sistema de Administración OSI, un conjunto de documentos que contemplan:

- Conceptos para en el Sistema de Administración OSI.
- SMI
- CMIS
- CMIP

10.1 Arquitectura del Modelo de Administración OSI.



Donde:

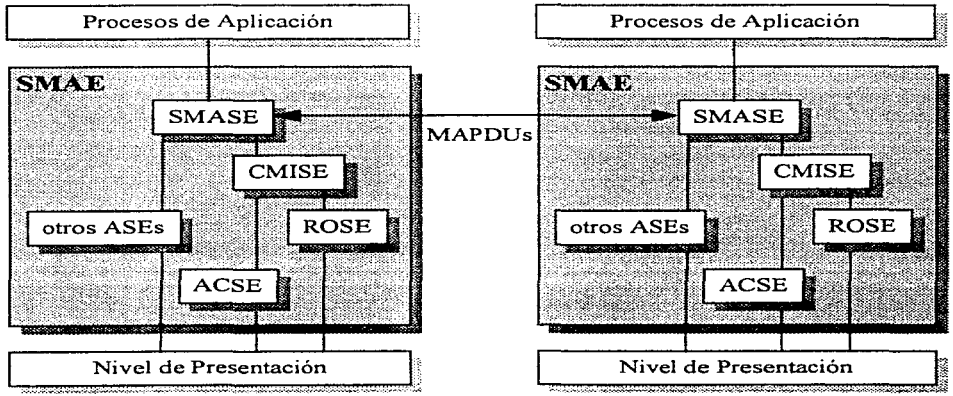
- Procesos de Aplicación del Sistema Administrador, es el software local con un sistema que es responsable de la ejecución de funciones del sistema administrador, con un sistema simple (host, procesador final, enrutador, etc.) éste tiene acceso a los parámetros del sistema y capacidades, manejando todos los aspectos del sistema y coordinarse con SMAPs (procesos de la aplicación del sistema administrador) en otro sistema.

- Entidad de Aplicación del Sistema Administrador; es responsable del intercambio de información de administración con pares de SMAEs (Elementos del Servicio de Aplicación del Sistema Administrador), en otros nodos, especialmente con el sistema que ejecuta funciones centrales para el control de red.

- Entidad del Nivel de Administración; lógicamente es relacionado con cada nivel de la arquitectura OSI para proveer funciones de administración en cada nivel de manera específica.

- MIB; colección de información de cada nodo perteneciente a la administración de red.

Ahora debemos especificar la estructura en el nivel de aplicación.



Donde:

- SMAE: Entidad de Aplicación del Sistema Administrador.
- SMASE: Elemento de Servicio en la Aplicación del Sistema Administrador.
- CMISE: Elemento de Servicio Común de Administración de Información.
- ASE: Elementos de Servicios de la Aplicación.
- ROSE: Elementos de Servicios Remotos de Operación.
- ACSE: Elementos Asociados con el Servicio de Control.
- SMAP: Proceso de Aplicación del Sistema Administrador.

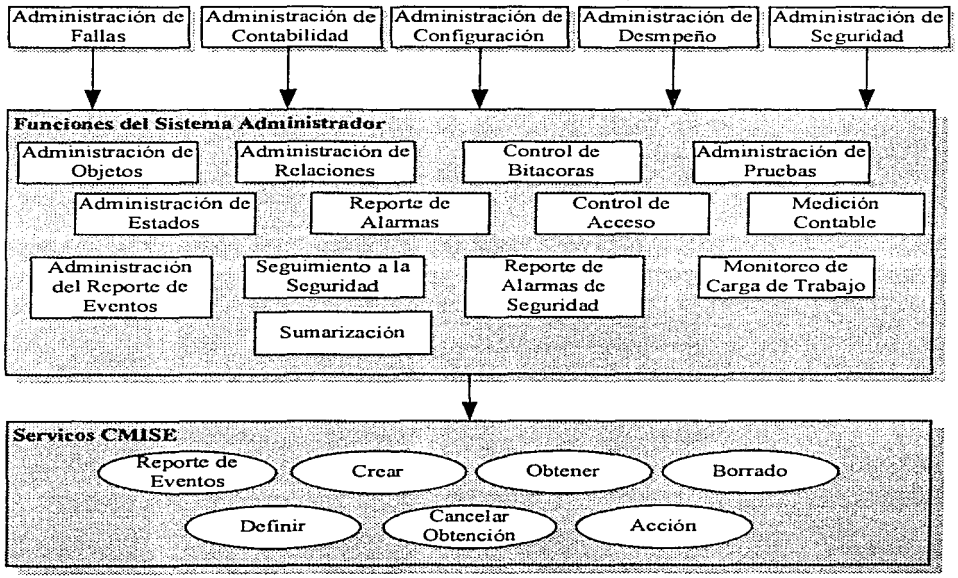
El SMAE puede ser definido como una interrelación de ASEs. Dos ASEs específicos en la administración de red son el CMISE.

El CMASE provee de varios servicios disponibles para el administrador de red y aplicaciones con los que se implementan funciones de administración de red.

Dado que se trata de un sistema de administración distribuida todos los elementos de las dos figuras anteriores deben ser distribuidas en todos los sistema de la red sujetos a administración; las actividades de administración son efectuadas a través de la manipulación de objetos, cada sistema contiene un número de objetos, cada uno de estos es una estructura de datos que corresponden a una entidad actual para ser administrada.

El SMAP en un sistema está habilitado para desempeñar cualquiera de los dos papeles, como agente o como administrador.

El desempeño de la administración de red OSI establece la relación mostrada en la figura siguiente:



10.2 SMI OSI

La estructura de información para la administración define:

- Modelo de información de objetos administrados y sus atributos.
- Los principios para nombrar los objetos y atributos para que puedan ser identificados y manejados por el protocolo de administración.
- La estructura lógica
- Descripción del concepto de clases de objetos manejadores, su relación con lo que ellos pueden llevar a cabo, incluyendo herencia, especialización, alomorfismos y contenido.

10.2.1 Definición de Términos de SMI OSI.

Término SMI OSI	Término común Orientado a Objetos	Descripción
Objeto Administrado	Objeto	La administración OSI ve al recurso dentro de un ambiente OSI que puede ser administrado a través del uso de protocolos de administración OSI, es decir, la administración de los recursos identificándolos según su nivel en el modelo OSI.
Clase de objetos administrados	Clase de objetos	Un conjunto de objetos administrados que comparten el mismo nombre, definición de atributos, notificaciones y operaciones de administración que comparten las mismas condiciones para la presencia de sus paquetes.
Atributo	Variable	Una propiedad del objeto administrado y tiene un valor definido.
Operación	Mensaje	Una operación en un objeto administrado para el sistema administrador.
Comportamiento	Método	Descripción del modo por el cual los objetos administrados, nombre relacionado, atributos notificaciones y acciones interactúan con los recursos actuales entre ellos
Notificación	Mensaje	Información emitida por un objeto administrado, referente a

Término SMI OSI	Término común Orientado a Objetos	Descripción
		un evento que ha ocurrido con éste.
Plantilla	---	Formato estándar para la documentación del nombre relaciones de nombre y definición de objetos administrados y sus componentes: paquetes, parámetros, atributos, grupos de atributos, definición de comportamiento, acciones o notificaciones.
Encapsulación	Encapsulación	Encapsula entre el objeto administrado y sus componentes para garantizar su integridad, donde cada tipo de recurso para ser administrado en el sistema es representado por una clase de objeto administrado. Una instancia específica que el recurso es representado por una instancia del objeto administrado.
Herencia	Herencia	El mecanismo conceptual por el cual atributos, notificaciones, operaciones y comportamiento son adquiridos por una subclase desde una superclase.
Especialización	Inherencia	La técnica de derivar nuevas clase de objetos administrados desde una clase existente por la adición de nuevas capacidades

Término SMI OSI	Término común Orientado a Objetos	Descripción
		(nuevos atributos o notificaciones.
Contenido	Contenido	Una estructura relacional para instancias de objetos manejados en la cual la existencia de ésta es dependiente en la existencia del contenido de la instancia del objeto administrado.
Alomorfismo	Polimorfismo	Habilitación de un objeto administrado de una clase dada para reensamblar uno o mas clases de objetos a otra.
Paquete	---	Colección de atributos opcionales, notificaciones, operaciones y comportamiento que están todos presentes o todos ausentes en un objeto administrado. La presencia o ausencia de un paquete es condicional a la capacidad del recurso en cuestión.

10.2.2 MIB OSI

La MIB del Sistema de Administración OSI esta diseñada en conceptos orientados a objetos, donde cada recurso es monitoreado y controlado por el sistema, cada uno de éstos es representado por un objeto administrado. La MIB, es una colección estructurada de objetos, un objeto administrado puede ser definido por cualquier recurso que una organización desee monitorear y/o controlar, por ejemplo: las estaciones de trabajo, PBX, LANs, multiplexores, programas, algoritmos de enrutamiento rutinas de administración de archivos, son recursos a representar por objetos diferentes. A los objetos que refieren el

recurso a un nivel específico del modelo OSI es llamado objeto administrado del nivel N (donde N es el nivel, según el modelo OSI, del al que pertenece el recurso); por otro lado si pertenece a más de un nivel es llamado objeto administrador del sistema.

Debemos considerar que para la MIB OSI

- Un objeto administrado es una abstracción que esta habilitado para las funciones del sistema administrador.
- Un objeto puede representar un solo recurso o varios elementos en la red.
- No todos los recursos necesitan ser representados por algún objeto administrado (casos en los cuales el recurso existe pero no está habilitado para sistemas de Administración OSI)
- Los objetos que ayudan a soportar las funciones de administración y no representan ningún recurso.

Las especificaciones de la MIB son especificadas en su SMI que establece:

10.3 Áreas de Administración OSI

Las funciones de administración implementadas por SMASE cubren las siguientes áreas.

Administración de:	Procedimientos que OSI provee:
Fallas	• Detecta y reporta la ocurrencia de fallas utilizando un protocolo de reporte de eventos. • Bitácora de recepción de reportes de eventos, para ser examinados y procesados. • Planeación y ejecución de pruebas de diagnóstico e inicio de corrección de fallas.

Administración de:	**Procedimientos que OSI provee:**
Contabilidad	• Informe del costo incurrido por usuario, utilizando el reporte de eventos y software de manipulación de información. • Habilita límites de contabilidad para ser definidos por el uso de recursos administrados. • Habilita costos para ser combinados donde múltiples recursos son usados para ejecutar comunicaciones necesarias.
Configuración	• Colección y diseminación de información concerniente al estado corriente de los dispositivos mediante protocolos estándar. • Define y modifica parámetros relacionados a los componentes de red y software del nivel OSI. • Cambios de configuración. • Definición de objetos y sus nombres asociados.
Desempeño	• Colección y diseminación de información concerniente al nivel corriente de desempeño de recursos. • Mantenimiento y examinación de bitácoras de desempeño para propósitos de planeación y análisis.
Seguridad	• Facilidades de autorización • Control de acceso • Encriptación y llave de administración • Verificación • Bitácora de Seguridad.

Las funciones en el modelo de administración OSI se describe en base a sus responsabilidades, cada área funcional involucra el uso de funciones específicas, éstas son referidas como SMFs (Funciones del Sistema Administrador) que pueden cubrir las funciones de una o varias áreas funcionales (el reporte de eventos es un ejemplo de ésto).

10.4 Áreas Funcionales OSI

10.4.1 Administración de Configuración

Definición de atributos y recursos; especifica el rango y tipo de valores para el cual el atributo puede ser definido.

A. Definición y modificación de valores de atributos; carga los atributos predefinidos como default y define relojes.

B. Define y modifica relaciones entre los recursos de la red. Agrega, borra y modifica relaciones en línea.

C. Examina el valor de atributos y relaciones de manera local o remota y conserva sus cambios históricamente.

D. Distribución de software en la red, provee mecanismos para examinar, actualizar y manejar diferentes versiones de software e información de enrutamiento.

E. Inicialización y terminación de operación de operaciones en la red.

F. Verificación de autorización de usuarios, capacidad de especificar jerarquías y autorización de funciones de configuración ; provee de métodos para asignar validación de varios niveles de autorización.

G. Reporte del estado de configuración; el sistema debe estar habilitado para informar al sistema agente bajo que condiciones y donde cambio la configuración. El usuario debe estar habilitado para las etapas de configuración.

10.4.2 Administración de Fallas

Detección de reporte de fallas

H. Provee de mecanismos que permitan a los usuarios formar la bitácora de eventos y errores, incluyendo especificaciones de definición de

filtros de la bitácora, incluyendo su inicialización y parado, además de especificar la información a ser almacenada. Define también niveles para establecer cuando debe ser dada una notificación.

I. Corrección de fallas; cambia el valor de los atributos de recursos que se encuentren con falla.

10.4.3 Administración de Seguridad

Control de

J. Control de acceso a recursos , permitiendo o negando acceso a partes específicas de la red.

K. Archivando y recuperando información de seguridad, lo que permite almacenar la información apropiada.

L. Manejando y controlando el proceso de encripatación.

10.4.4 Administración de Desempeño

Monitoreo del desempeño:

M. En relación a eventos, recursos y medidas. Especificando tiempo de inicio y fin del monitoreo. Además especifica medidas para ser generadas.

N. Ajuste y control de desempeño; provee de mecanismos para ejecutar pruebas predefinidas de desempeño y recolecta el resultado para diagnosticar problemas de desempeño y determinar estrategias de rastreo.

O. Evaluación del ajuste de desempeño; conserva los puntos de ajuste en términos de criterios específicos por el usuario.

P. Reportes del monitoreo de desempeño, ajuste y rastreo. Generando notificaciones de cambios en el desempeño anormales.

Q. Pruebas de capacidad y condiciones específicas corriendo pruebas para determinar los efectos potenciales de cargas adicionales a la red.

10.4.5 Administración de Contabilidad

R. Grabando y generando información de contabilidad. Especificando tiempos para la recolección de información de contabilidad generando también mensajes de este tipo.

S. Especificación de información de contabilidad a ser recolectada.

T. Control de almacenamiento de fin de acceso a información de contabilidad.

U. Reporte de información de contabilidad; reportando el grado de utilización del recurso y sus respectivas cargas según el nivel especificado.

V. Define y modifica los límites de contabilidad, cambia las prioridades asignadas a usuarios para accesar recursos de red.

W. Define métricas de contabilidad.

10.5 SMF Funciones del Sistema Administrador

Las Funciones del Sistema Administrador es un conjunto de estándares que han sido implementados para definir la funcionalidad específica en las cinco áreas funcionales para los sistemas de administración (SMFAs) con la intención de evitar duplicidad de funciones, así con una SMF puede ser implementada una o mas SMFA. Las SMF son:

10.5.1 Función de Administración de Objetos

Es la tarea fundamental de todas las SMF. para la creación, borrado y cambio de los valores, para ello existen tres caminos:

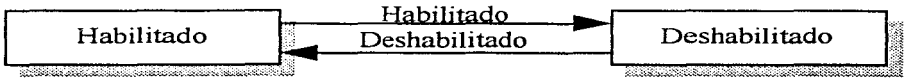
- A través de un proceso de configuración en el ambiente del sistema local que está fuera del panorama OSI.
- A través del nivel de operación N del nivel N de administración de un sistema abierto.
- A través de la función de administradora de objetos como parte de los servicios del sistema administrador.

10.5.2 Función de Administración de Estados

Especifica un modelo de como el estado administrador de un objeto es representado; el modelo permite al usuario de administración OSI conocer el estado pasado de objetos administrados y recibir noticias de cambios de estados. Los servicios son definidos para monitorear operatividad y uso de los recursos del sistema y para disposición de restricciones administrativas.

Diferentes bases de objetos pueden tener diferentes atributos que sean relevantes para el monitoreo y operación de un recurso asociado, sin embargo ña administración de estados estandariza para los diferentes recursos, definiendo tres diagramas de estados correspondientes a los tres factores primarios que efectúan las administración del estado de un objeto.

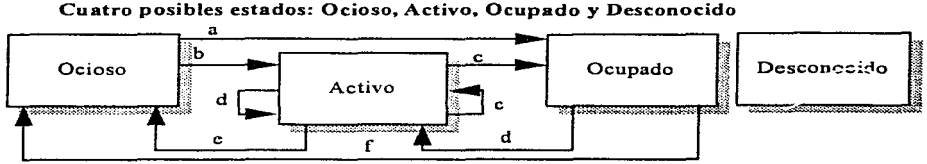
Dos posibles estados: Habilitado y Deshabilitado



Operatividad: si el recurso esta instalado y opera, o no.

Utilización: si esta el recurso activamente en uso un instante específico, o no y si éste cuenta con capacidad para usuarios adicionales en determinado instante, o no.

Cuatro posibles estados: Ocioso, Activo, Ocupado y Desconocido



Donde:
a Nuevo usuario (objeto no existente)
bNuevo usuario
c Nuevo usuario o decrementa la capacidad
d El usuario se sale y/o incementa la capacidad
e Se sale el último usuario
f Salida del usuario (objeto no existente)

Administración: si un objeto puede o no ser usado.

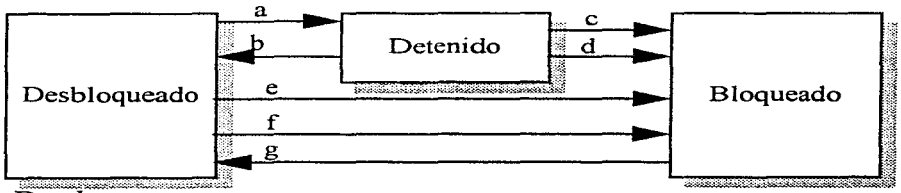
Las posibles combinaciones de los tres modelos anteriores tienen las siguientes interpretaciones:

Deshabilitado, ocioso y bloqueado: el recurso es inoperable y no provee servicio al usuario, es también bloqueado para su administración se requieren acciones correctivas.

Habilitado, ocioso y bloqueado: el recurso es operable pero no provee servicio a los usuarios, está administrablemente bloqueado, será necesario desbloquearlo para que reinicie sus servicios.

Habilitado, activo y detenido: el recurso es operable, pero su uso es restringido por el usuario corriente. El recurso puede hacer habilitaciones para usuarios adicionales sólo si

Tres posibles estados: permite al administrador bloquear o desbloquear el acceso a un recurso, incluyendo el de dar de baja el sistema



Donde:
a Detener
b Desbloquear
c Salida del último usuario
d Bloquear
e Detener si el usuario no existe
f Bloqueado
g Desbloqueado

administrativamente se cambia a desbloqueado. De otra manera cuando todos los usuarios corrientes terminen, el recurso pasará al estado habilitado, ocioso y bloqueado.

Habilitado, ocupado y detenido: El recurso es operable pero su uso es restringido por los usuarios actuales y el sistema no tiene capacidad para mas usuarios. El recurso estará disponible para mas usuarios, sólo si es cambiado administrativamente a desbloqueado y si uno o mas usuarios terminan. Si no se cambia administrativamente cuando todos los usuarios salgan el sistema cambiará a habilitado, ocioso y bloqueado.

Deshabilitado, ocioso y desbloqueado: el recuso es inoperable y no provee servicios a los usuarios, se requieren acciones correctivas.

Habilitado, ocioso y desbloqueado: el recurso está disponible pero no provee servicio a ningún usuario.

Habilitado, activo y desbloqueado: el recurso esta disponible y ya provee de servicio a algún usuario.

Habilitado, ocupado y desbloqueado: el recurso es operado y provee servicio a usuarios pero no tiene capacidad para mas usuarios.

10.5.2.1 Atributos de Estado

I. Estado de alarma:

- A. Bajo recuperación
- B. Crítica
- C. Mayor
- D. Menor
- E. Detección

II. Estado de procesamiento:

- A. Requiere inicialización
- B. No inicializado
- C. Inicializado
- D. Reportando el resultado

E. Terminando

III. Estado de Disponibilidad

A. En prueba

B. Falla

C. Potencia baja

D. Fuera de línea

E. Fuera de control

IV. Estado de Control

A. Objeto en prueba

B. Parte de servicio bloqueado

C. Reservado para prueba

D. Suspendido

V. Estado de espera:

A. Espera sincronizada a otro recurso

B. Espera no sincronizada a otro recurso

C. Provee servicio y espera a otro recurso

VI. Estado desconocido:

A. Verdadero

B. Falso

10.5.3 Función de Administración de Relaciones

Esta función especifica un modelo para representar y manejar relaciones entre objetos administrados y provee servicios para soportar el módulo. En general una relación es un conjunto de reglas que describen como la operación de un objeto administrado afecta la operación de otro objeto administrado.

10.5.3.1 Modelo de Relación

Las relaciones son definidas en base a sus atributos, a cada tipo de relación se le conoce con el nombre del rol; pueden ser 1 a n, n a 1, 1 a 1, n a n, encontrando tres categorías en el modelo de administración OSI:

Relaciones Contenidas: en esta existe un objeto administrado que depende de la existencia de un objeto administrado contenido.

Relación Recíproca: existe una relación entre dos objetos para representar la relación, cada objeto incluye un atributo cuyo valor es el nombre del otro objeto.

Relación un camino: esta es una relación asimétrica entre dos objetos en los cuales la relación es expresada en el valor del atributo de la relación se encuentra en uno de los dos objetos.

10.5.4 Función de Reporte de Alarmas

Soporta la definición de alarmas de falla y uso de notificaciones para reportes, especificando los eventos de notificaciones de alarmas, parámetros y semántica, provee tipos de errores para causas probables:

Se definen cinco categorías de alarmas: error de comunicación, error en la calidad del servicio de un objeto administrado o servicio, error en el proceso de un objeto administrado, error en el ambiente referido a un ambiente específico.

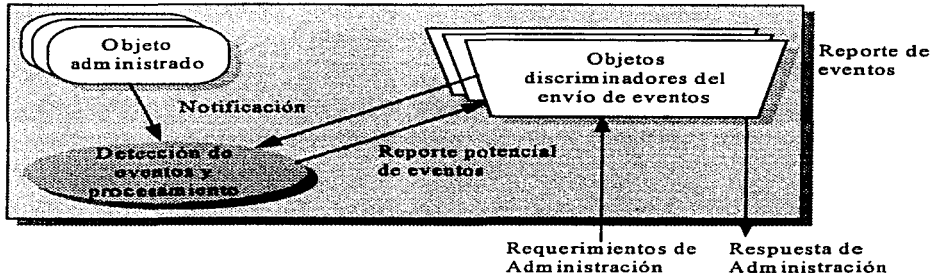
Donde toda notificación incluye: causa del problema, problema específico, percepción de la severidad de la falla (en base a prioridades: crítica, mayor, menor, advertencia, indeterminada, limpia de alarmas anteriores), estado de respaldo, objeto respaldado, umbral extendido que hizo activar la alarma, notificación correlacionada, definición de cambio de estado, atributos monitoreados, acciones propuestas para reparar la falla, texto adicional para describir la falla e información adicional.

10.5.5 Función de Administración de Reporte de Eventos

La administración del control de transmisión de reportes de eventos incluye especificaciones de los receptores del reporte y especificaciones de criterios para generar y distribuir reportes. Habilita al usuario a crear un evento discriminador en base a un criterio para que la notificación de los eventos reportados se encuentren solo en los rangos que el evento discriminado lo permita. Esta función requiere:

- La definición flexible de un servicio de control de reporte de eventos que permita a los sistemas la selección de un reporte de eventos específico a un sistema administrativo
- Especificación de los destinos para el cual el evento es reportado.
- Especificación del mecanismo para control del envío del reporte de eventos.
- Disponibilidad para un sistema de administración externo para modificar las condiciones usadas en el reporte de eventos.
- Disponibilidad para diseñar un respaldo para el reporte de eventos si la primera localización no está habilitada.

Modelo de Administración del Reporte de Eventos



10.5.6 Función de Control de Bitácoras

Creación y almacenamiento de registros mediante criterios de especificación, este modelo es muy similar al de la función anterior, ya que cuenta también con un módulo donde se define el criterio de almacenamiento. Cubriendo principalmente:

- Una definición flexible del servicio de control de almacén que permita la selección de registros a ser almacenados.
- Capacidad para un sistema externo para modificar el criterio usado en los registros almacenados.

- Capacidad para un sistema externo para determinar donde las características de almacenamiento fueron modificadas o donde los almacenes se perdieron.
- Capacidad de un sistema externo para traer, borrar y crear registros de almacenamiento.

En general los registros en un bitácora son almacenados en el orden en que se reciben y su identificador es un número secuencial.

10.5.7 Función del Reporte de Alarmas de Seguridad

Se definen aquí parámetros y semántica para implementar servicios y mecanismos referentes a la seguridad, siendo su objetivo cubrir la necesidad de los usuarios administradores de identificar un ataque o un potencial ataque al sistema de seguridad.

Soporta cinco tipos de alarmas:

- Violación de la integridad: la información ha sido ilegalmente modificada, insertada o borrada.
- Violación de la operación: un servicio a sido inhabilitado se le ha involucrado de manera errónea.
- Violación física: un dispositivo ha sido alterado.
- Violación del mecanismo o servicio de seguridad:
- Violación del dominio del tiempo, es decir un evento ocurre fuera del periodo de tiempo definido (password expirado, horas fuera de actividad, retraso de la información, etc.).

10.5.8 Función de Seguimiento a la Seguridad

Especifica las clases de eventos que deben estar contenidas en una bitácora usada para la evaluación de la seguridad, así como el desempeño de mecanismos de seguridad, puede ser utilizada para buscar ataques de seguridad que no se detectaron cuando ocurrieron, es en si una extensión de la función de control de bitácoras. Incluye conexiones, utilización de mecanismos de seguridad, administración de operaciones, contabilidad de utilización, etc. El servicio puede indicar uno de los siguientes valores:

Reporte de servicio: un indicador de un evento perteneciente a la prevención, rechazo o recuperación de un servicio.

Reporte de utilización: un indicador de un registro que contiene información utilizada para estadísticos.

10.5.9 Función del Control de Acceso

Define un modelo para el control de información y operaciones, especificando objetos administrados y atributos para ser utilizados para dar o quitar acceso según las políticas de acceso representadas por información de la administración de control de acceso.

10.5.10 Función de Medición Contable

Especifica el modelo para la contabilidad del uso de los recursos del sistema y mecanismos para limitar su ejecución, el estándar define parámetros de contabilidad y bitácoras específicas para cobrar utilización, obtención de reportes y registro de utilización de registros. Cubriendo así dos aspectos importantes: el control de reportes de información asociada con el uso de los recursos y la especificación del almacenamiento de la información.

10.5.11 Función de Monitoreo de Carga de Trabajo

Especifica el modelo para monitorear los atributos de los objetos administrados, éste define objetos que pueden reportar eventos basados en valores de contadores y medidas que reflejan el desempeño del sistema. Utilizada sobre todo para detectar situaciones de sobrecarga reales o posibles para la consideración de planeación de dispositivos en la red. El sistema administrador monitorea el nivel de demanda en la capacidad de varios recursos. El servicio se caracteriza por:

- Un modelo de utilización de recursos, tasa de rechazos de los recursos y tasa de requerimientos del recurso.
- Atributo de objetos administrados: controles definidos y no definidos por el administrador y rechazo.
- Medición de objetos: monitoreo de rechazo y medida del monitoreo del propósito del objeto.

El proceso básico de monitoreo de cargas de trabajo pueden dividirse en:

Captura de información: la información es extraída desde los objetos observados, es decir, la observación de los valores de un atributo cada determinado intervalo.

Conversión de la información: Si hay algún interés de observar el rango de variación de un contador entonces se realiza la resta entre el valor anterior del encontrado y el nuevo sobre el periodo de tiempo en que se tomaron ambas muestras.

Mejoramiento de la información: para encontrar tendencias con la información un algoritmo de mejoramiento de información puede ser utilizado para interpretar de mejor forma la información observada.

Análisis de la información: el valor de la medición calculada puede ser comparada con umbrales para el disparo de alarmas.

10.5.12 Función de Administración de Pruebas

Soportando procedimientos para diagnóstico de pruebas y confiabilidad, por lo que define clases de objetos administrados que son utilizados para el control de pruebas interactivas o asíncronas, con resultados para ser reportados mas adelante. Tipos de pruebas:

- Prueba síncrona contra asíncrona.
- Reportes solicitados contra no solicitados.
- Terminación explícita contra terminación implícita.

El modelo de pruebas cambia si se trata de pruebas síncronas o asíncronas; en pruebas síncronas el resultado final de la prueba es enviado con la inicialización de la prueba, por el contrario en pruebas asíncronas el resultado se envía por petición del administrador o como evento de notificación.

10.5.13 Función de Sumarización

Definición de medidas estadísticas para ser aplicadas a atributos y reporte de información sumarizada, definiendo modelo y clases de objetos utilizados para la sumarización y aplicación de análisis estadístico para la administración de la información. La sumarización de valores de atributos incluyen objetos específicos de tiempo. La función de sumarización cubre también la extracción de la información desde el objeto administrado y su paso a la sumarización del

objeto. La información puede ser obtenida desde los atributos de objetos representados como atributos métricos de objetos y registros almacenados.

Toda función de sumarización esta basada en el concepto de un recorrido, éste recorrido es un proceso simple de calores de atributos observados en puntos y tiempos específicos.

10.6 CMIS y CMIP

La función básica de comunicación entre el administrador y el agente en el protocolo de administración es definido por OSI como el elemento de envío común de información de administración (CMISE), el cual consta de:

CMIS La interfaz con el usuario, especificando el servicio que provee: servicio de información de administración común.

CMIP El protocolo que especifica el formato de PDU a utilizar y sus procedimientos, protocolo de información común para la administración.

10.6.1 CMIS

Provee de siete servicios para el desempeño de operaciones de administración en forma de primitivas de servicio. Este servicio es invocado por el proceso administrador para lograr la comunicación remota.

Los servicios CMIS son especificados en términos de primitivas que pueden ser vistos como comandos o llamadas a procedimientos con parámetros, ofreciendo dos tipos de servicios: requerimientos de confirmación de servicios que un proceso de administración remoto envía para indicar recepción exitosa o con falla de la operación y servicios son confirmación que no provee respuestas.

Identificándose tres categorías principales:

Servicios de asociación entre aplicaciones para lograr la comunicación.

Servicio de notificación de administración. Este servicio es usado para asignar información a la notificación la definición de notificación y su consecuente manejo de comunicación entre entidades; depende de las especificaciones del objeto administrado que genera la notificación. Definidas mediante las primitivas M-EVENT-REPORT donde cada notificación es acompañada por un identificador único para proveer la

inicialización del usuario la entrega, el tipo de evento, tiempo de su generación y reporte de errores entre otros.

Servicios de operación de administración, al igual que el anterior es dependiente de las especificaciones del objeto administrado.

Todos los servicios anteriores cuentan con dos estructuras posibles de comunicación:

- Respuesta múltiple para una confirmación de operación, puede ser ligada la operación por el uso de parámetros de identificación de enlace; mediante las primitivas de servicio M-GET, M-SET, M-ACTION, M-DELETE (objetos administrados).

M-GET: permite obtener información de la MIB donde un proceso administrativo envía una solución de respuesta a un proceso de administración con rol de agente, sobre un objeto administrado o un conjunto de estos.

M-SET: es el servicio que permite la modificación de la información de administración para cambiar valores de uno o mas atributos en uno o mas objetos, pudiendo ser con servicio de confirmación o no.

M-ACTION: permite la invocación de una acción predefinida como procedimiento como parte del objeto administrado. el requerimiento especifica el tipo de acción y los parámetros de entrada, también puede ser cuenta con servicio de confirmación o no.

M-CREATED: usado para crear una nueva instancia de una clase de objetos de paquetes condicionales y valores de atributos que el objeto administrado tendrán, deben ser especificados como parte de la petición o una instancia existente puede ser usada como modelo. Siempre mediante el servicio de confirmación.

M-DELETE: servicio para borrar uno o mas objetos administrados, siempre es un servicio de conformación.

M-CANCEL-GET: usado para detener una operación larga, solamente para la operación de get, protegiendo así la integridad de la MIB, por lo que siempre es un servicio confirmado.

- Operaciones que pueden ser desarrolladas en múltiples objetos administrados, seleccionados para satisfacer algún criterio a una condición de sincronización.

10.6.2 CMIP

Es empleado como el protocolo de intercambio de información basado en los servicios del elemento de servicio de operaciones remotas. Define procedimientos para la transmisión de información y define la sintaxis para los servicios de administración (CMIS), definiendo once PDUs con tres tipos de información cada uno.

Para los servicios de operación de administración el CMISE emplea un CMIP para intercambiar PDUs, CMIP se apoya en los servicios de los elementos de servicio de operación remota (ROSE); para lo que respeta las siguientes reglas:

- El iniciador de la asociación y el que responde puede invocar operaciones.
- Para operaciones CMIS confirmadas usa el método de reporte, regresando resultado para éxito o fracaso y el modo de operación puede ser síncrono o asíncrono.
- Para operaciones no confirmadas, no hay respuesta, su operación es asíncrona.

10.6.2.1 Características de CMIP

Se pueden establecer tres requerimientos involucrados con el sistema administrador de una red.

1. No deberán degradar seriamente la operación de la red que intenta mantener el ancho de banda (según Aronoff 1989, usando no mas del 5%).
2. Sus decisiones deben realizarse y acciones de control deben tomarse rápidamente antes que las condiciones de control cambien de manera significativa, de otro modo la inestabilidad puede aumentar en el ciclo de alimentación de acciones de control y medición.
3. Debería proporcionar un conjunto amplio de servicios para manejar un amplio rango de funciones de administración de red, así como para proporcionar un monitoreo y control detallados.

10.6.3 Ventajas y Desventajas de la Administración OSI

- Amplio rango de funciones por nodo administrado y administrador.
- La MIB OSI es muy compleja pero esta basada en objetos, atributos y recursos, lo que la hace particular a cada nodo según las necesidades de administración.
- Los PDUs tienden a ser muy grandes, pero independientes de la máquina; por lo que lo hace un protocolo ampliamente compatible.
- Eficientiza el acceso a los objetos, identificándolos previamente a la clase a que se refiere.
- Es un protocolo orientado a un esquema de administración global que distribuye sus funciones a un conjunto de administración local.

11. XWINDOW

Desarrollado en el Instituto de Tecnología de Massachusetts como un sistema de manejo de ventanas en red para computadoras con pantalla de mapa de bits, de tal forma que el usuario final pueda tener a la vista simultáneamente el despliegue de varios programas sin necesidad de varios monitores.

Xwindow puede correr en diferentes tipos de redes. El manejo de ventanas puede hacerse de manera local y remota mediante el uso de sockets Berkeley

UNIX y el protocolo TCP/IP o DECnet, que son los protocolos de red de bajo nivel más utilizados para soportar servidores X.

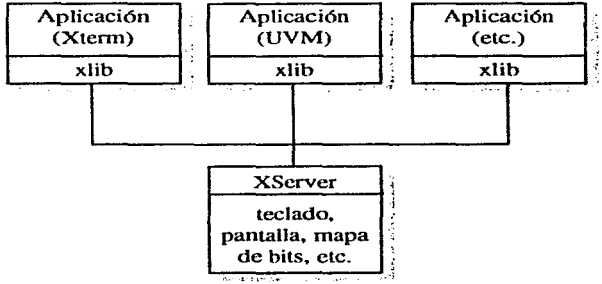
Xwindow consta de dos módulos principales:

- Xlib
- X-server

Xlib reside en la aplicación cliente, es un módulo en lenguaje C que acepta las entradas del usuario, el módulo es responsable de enviar la información en ambas direcciones para la terminal del usuario, este módulo es llamado también X-client.

X-server reside en el host, es el software de despliegue el cual recibe y envía información para la cliente.

Ejemplos de dispositivos que X maneja como servidor son: teclado, mouse, pantalla, etc.



11.1 Reglas del sistema:

- Un X-server debe ser instalado en el host que servirá a varios X-client identificando la interfaz apropiada.
- El X-server no mantiene las ventanas, ya que esto es responsabilidad del cliente quien recibe la llamada de transacción por el X-server cuando algo ha cambiado en pantalla.
- XWindow es manejado a través del concepto pilas, y ventanas jerárquicas de padres e hijos.
- Xlib cuenta con un número de herramientas que permiten trabajar en un nivel más alto a Xlib para el desarrollo de aplicaciones.

11.2 Protocolo del Sistema XWindow

Este protocolo es utilizado para lograr la comunicación entre X-server y X-client y consta de cuatro tipos de mensajes:

Requerimiento: una instrucción para el servidor de la estación de trabajo para ejecutar una acción (por ejemplo: dibujar una línea).

Respuesta: manda desde el servidor la contestación a un requerimiento.

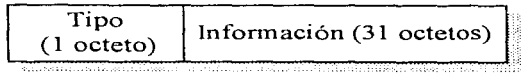
Evento: usado por el servidor para informar a la aplicación de cambios que la afectan (por ejemplo: un click con el ratón, requerimiento del cursos, etc.).

Error: enviado a la aplicación cliente por el servidor si algo esta mal (por ejemplo: el usuario selecciona un proceso que requiere más memoria de la disponible).

11.2.1 Formato de Requerimientos

Código principal (1 octeto)	Longitud (8 otetos)	Código (1 octeto)	Información (variable)

11.2.2 Formato Respuesta, Error, Evento.



Muchos mensajes de X no garantizan una respuesta. Por ejemplo un mensaje del movimiento del ratón es una respuesta de acción predefinida, en cuanto a la atención que da el servidor a esta solicitud, este tipo de mensajes son frecuentemente almacenados y enviados de manera "oculta" lo que permite al usuario utilizar un requerimiento de xlib y realizar otra operación. Por otra parte existen transacciones de viaje redondo, donde se regresa una respuesta específica, la cual es esperada por la aplicación.

Un mensaje de evento es enviado solamente si una aplicación ha solicitado el tipo de evento que esta siendo enviado. Esta aproximación es muy importante ya que permite a la aplicación recibir sólo eventos relevantes.

11.3 Conexión con la Pantalla de Despliegue

Una aplicación debe entablar una conexión con la pantalla antes de que estos puedan comunicarse; utilizando para esto X Open Display de las bibliotecas de xlib, que establece la comunicación con la estación de trabajo. Esta y xlib intercambian información entre si, durante la estabilización de la conexión; una vez lograda xlib crea una estructura de despliegue que contiene la información necesaria de configuración para lograr la comunicación adecuadamente entre la aplicación y la estación de trabajo. (La respuesta a X Open Display es un apuntador a la estructura de desplegado).

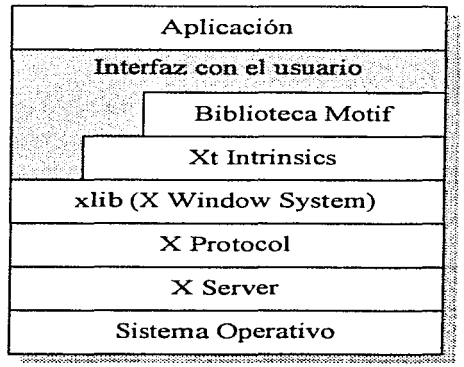
Dentro de las operaciones de X las aplicaciones usan requerimientos estándar para operar en los objetos o recursos. Algunos de estos recursos son: ventanas, contextos gráficos, color, fuente de letras, mapa de pixeles, cursor, etc..

11.3.1 Ambientes de Desarrollo

Existen muchas formas de programar aplicaciones en X, ya que este sistema no es restringido a un solo lenguaje o sistema operativo, el único requerimiento de una aplicación X es que genere y reciba mensajes para el protocolo X, acorde a las especificaciones del Consorcio de Protocolos X; sin embargo el camino más comúnmente utilizado es Xt Intrinsics y Xm Motif porque:

- Son herramientas poderosas que facilitan la programación.
- Dan como resultado operaciones que operan fácilmente con otras.
- Soportan diferentes tipos de interfaces de usuario.
- El lenguaje C se encuentran disponible comúnmente.

La figura muestra los niveles de software en una aplicación que utiliza Xt Intrinsics y Motif.



Xlib provee de un acceso completo a las capacidades del Protocolo X pero hace más fácil su programación, ya que maneja la interfaz entre la aplicación y la red.

Xt está construido sobre xlib, su propósito es proveer de un nivel orientado a objetos que soporte una interfaz abstracta para el usuario, llamada widget. Un widget es una pieza de código configurable y reutilizable que opera independientemente de la aplicación a excepción de una previa definición de interacción.

Xlib y Xt están disponibles como software público, no así los widgets de Motif y Open Look pero son distribuidos por un costo mínimo por lo proveedores mismos: OSF, AT&T y Sun.

11.4 Motif

Motif es una herramienta para el desarrollo de una Interfaz Gráfica que nos permite especificar la forma en que debe presentarse al usuario una aplicación en pantalla y las forma en que interactúa con el usuario. A este tipo de interfaces se les denomina GUI (Interfaz Gráfica con el Usuario).

Motif, fue creado por OSF (Open Software Foundation), un consorcio de compañías tales como Hewlett-Packard, Digital, IBM, y otras corporaciones. Se decidió que la flexibilidad de X para el manejo de cualquier clase de interfaz gráfica, hacía conveniente el basar las herramientas de OSF/Motif en el Sistema de X Windows, dejando como Interfaz para Programación de la Aplicación (API) a Xt.

La mayoría de las aplicaciones Motif cuentan con botones para iniciar las acciones de una aplicación. Motif hace uso de sombreado, tercera dimensión y otros recursos para representar las acciones tomadas. Cuando se presiona un botón, la aplicación puede tomar una acción inmediata o desplegar una ventana adicional conocida como "caja de diálogo", que pide al usuario más información o presenta más opciones.

Las especificaciones en Motif contemplan dos aspectos básicos:

1. El modelo de salida, que se refiere principalmente a la apariencia del objeto en pantalla. Dicho modelo debe incluir la forma de los botones, efectos tridimensionales, el uso de cursores y bitmaps y el posicionamiento de ventanas y subventanas.

2. El modelo de entrada que especifica la forma de interacción del usuario con los elementos en pantalla.

En Motif se tienen dos opciones para el desarrollo de aplicaciones: crear la aplicación completa o emplear las herramientas existentes para la creación de ésta.

11.4.1 Diseño de la Interfaz con el Usuario

El diseño de una interfaz apropiada no es sencillo, puesto que se esperan respuestas concretas, que el sistema debe ser capaz de interpretar.

Es conveniente observar las siguientes reglas para en el diseño de una interfaz:

- Mantener la simplicidad en la interfaz.
- Hacer conexiones directas con objetos y conceptos reales.
- Improvisar en caso de no contar con herramientas.
- No sacrificar funcionalidad por simplicidad.

11.4.2 El Modelo de Programación Motif

A pesar de que Motif es una implementación independiente, OSF eligió tomar como plataforma las herramientas de X (Xt toolkit intrinsics), al igual que su sistema de ventanas para la Interfaz para Programación de la Aplicación (API).

El marco de trabajo que proporciona Xt es un ambiente orientado a objetos, que permite la creación de componentes reutilizables y configurables llamados "widgets" que pueden ser botones, cajas de diálogo, menús, etiquetas, barras deslizables, y áreas de despliegue e introducción de texto. Existen widgets administradores, que se encargan de controlar la disposición de otros widgets, de manera que la aplicación no interviene en la ubicación del widget, cuando ésta es modificada ya sea en tamaño o localización. Un widget opera de manera independiente a la aplicación a excepción de cuando se presentan interacciones preasignadas.

Agrupados en clases, conforman las bibliotecas de Motif (Xm) donde se define el comportamiento general de un widget. Existen además subclases de widgets definidas por Xt cuyo comportamiento puede ser heredado o modificado por otra clase de widget, formando así un esquema jerárquico de los distintos tipos de widgets.

Xt cuenta además con objetos de menor jerarquía, llamados "gadgets", que aparecen como widgets cuyo comportamiento está dado por un widget administrador. Tanto widgets como gadgets en su mayoría, heredan sus características de los objetos que tienen mayor jerarquía que ellos. Que Xt esté orientado a objetos, significa que la aplicación es completamente independiente del código propio del widget. A lo que se tiene acceso es a la

creación, manipulación y destrucción de widgets, así como a los recursos de estos. Los valores por definición para los recursos configurables, se especifican generalmente en el archivo "app-defaults", que por lo general tiene el mismo nombre que la aplicación.

Es por lo anterior, que para la creación de una interfaz con Motif es necesario el empleo tanto de la librería de Motif como de la de Xt.

Como se ha mencionado, una aplicación puede hacer llamadas al nivel de Xlib para poder hacer uso de gráficas o eventos del sistema de ventanas, además puede llegar a requerirse el hacer llamadas de más bajo nivel, esto es, al sistema operativo, sistema de archivos o manejadores, por lo que la aplicación se encontrara interactuando con todos los niveles.

11.4.3 Inicialización del marco de trabajo

Por lo mencionado anteriormente, antes de comenzar a trabajar con los widgets de motif, debe establecerse un marco de trabajo, y ésto lo haremos mediante el empleo de la función XtVaAppInitialize(), con la que se genera un widget de clase shell, el cual será la liga entre el administrador de ventanas y los widgets que de aquí en adelante se generen. Esta función incluirá la dirección del contexto de la aplicación; y definirá un nombre a ésta, que servirá para relacionarla a sus widgets. Es importante establecer aquí la lista de recursos que utilizará éste widget.

11.4.4 Creación de Widgets

La creación de widgets y gadgets es muy similar, y la función con que sean creados los va a definir en su correspondiente clase. Las funciones con que se crean los widgets contendrán como parámetros sus propios recursos, el nombre de su widget administrador y la clase predefinida con que se identifica. Ejemplos de las funciones con que se crean son: XmCreateBotton, XmCreateScrolledList.

11.4.5 Definición de Recursos.

Los recursos pueden definirse u obtenerse de acuerdo a las necesidades de la aplicación y cada clase definirá sus específicos. Los recursos pueden definirse en el momento de la creación del widget, con argumentos identificados por el prefijo XmN que tienen una prioridad menor que los especificados desde el archivo fuente o desde la línea de comando (opción -xrm).

Cada recurso contará con determinado tipo de información a asignar, la cual será definida por su clase o superclase.

Además se cuentan con la función XtVaSetValues que permite la definición o modificación de los recursos después de su creación, Es importante aclarar que el uso de ambos métodos en una aplicación es totalmente permitido.

11.4.6 Obtención de los recursos definidos.

Si se desean modificar los recursos ya definidos en los widgets se deberá hacer uso de la función XtVaGetValues, su uso será similar a la función XtVaSetValues con la diferencia de que se le asignarán variables donde se almacenarán los valores obtenidos de cada recurso.

11.4.7 Argumentos de los Widgets.

En la creación de los widgets en motif el primer argumento que se recibe, es el nombre de su padre, el segundo es el nombre que identificará al widget y el tercero y cuarto son las especificaciones de los recursos.

11.4.8 Manejo de Eventos.

Los eventos son aquellas acciones que se presentan por cualquier dispositivo de entrada para modificar los recursos o estados de los widgets. Se presentan de manera asíncrona, ya que pueden ocurrir en cualquier orden y ventana. El modo de controlarlos es mediante métodos definidos por motif, mediante tablas de traslación.

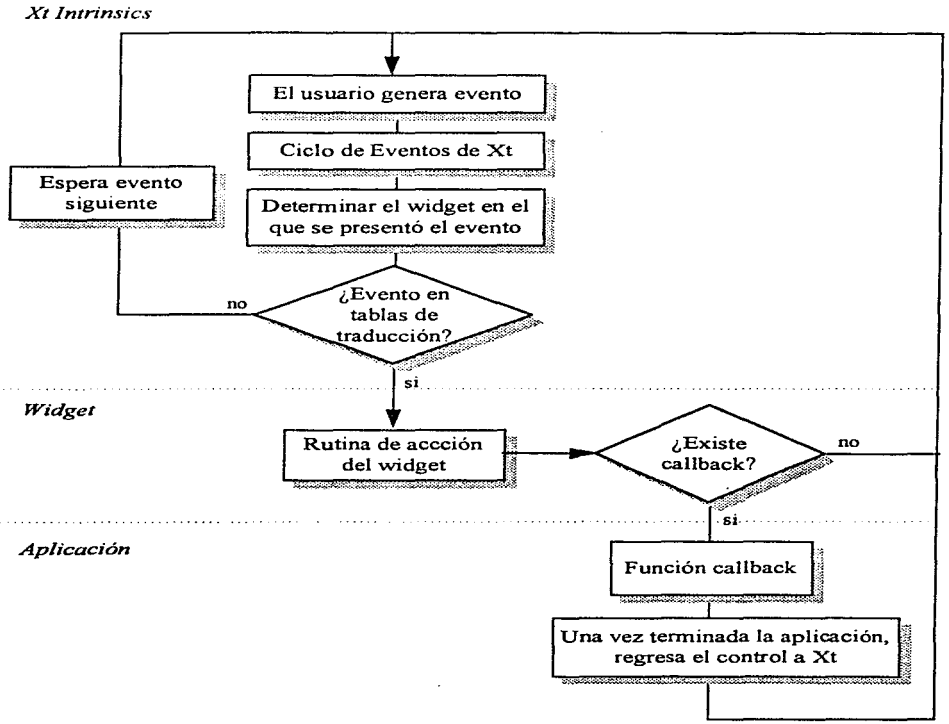
11.4.9 Recursos de Callback.

Además del control sobre las acciones de cada widget, se debe tener control sobre las acciones que se ejecutarán en la aplicación cuando se presente algún evento definido en las tablas de traslado. Las funciones específicas de la aplicación son invocadas mediante la función XtAddCallback(). Motif seguirá así el flujo que se observa en el diagrama de la página siguiente.

Donde los argumentos que recibirá son: el nombre del widget, el nombre del recurso de callback, la función de la aplicación a ejecutar y pueden además indicarse argumentos para la función.

Una vez definidos widgets y funciones se deben finalizar las especificaciones de motif cerrando el ciclo de programación con las funciones XtRealizeWidget(padre) y XtAppMainLoop(app). La primera tendrá como

parámetro al padre de todos los widgets, indicando que se desplieguen todos sus hijos según la definición previa; la segunda mantendrá un ciclo constante para el control de la realización de los eventos interactuando con la aplicación sólo en caso necesario y su argumento será el apuntador al contexto de la aplicación empleado en la inicialización del padre.

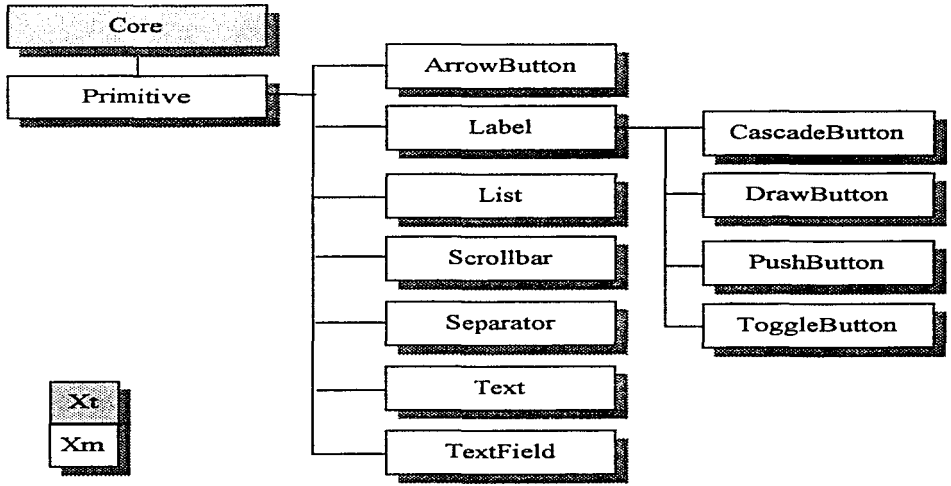


11.4.10 Clasificación de Objetos.

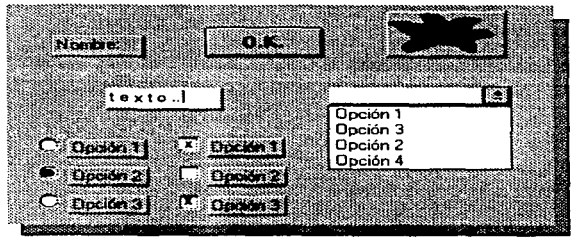
El tipo de widget con que se trabaje definirá las funciones disponibles para el manejo del mismo o de otros que dependan de él y dispondrá de los métodos que su superclase le defina, por lo que a continuación se presentan los widgets de Motif, según sus clases y superclases.

11.4.10.1 Widgets Primitivos.

Estos widgets crearán su propia ventana al ser creados, por lo que basta con definir cualquiera de éstos para tener las propiedades de cambio de tamaño, cambio de posición, etc.. El esquema de clase al que pertenecen son:



Se muestra a continuación algunos de los widgets antes mencionados más utilizados:

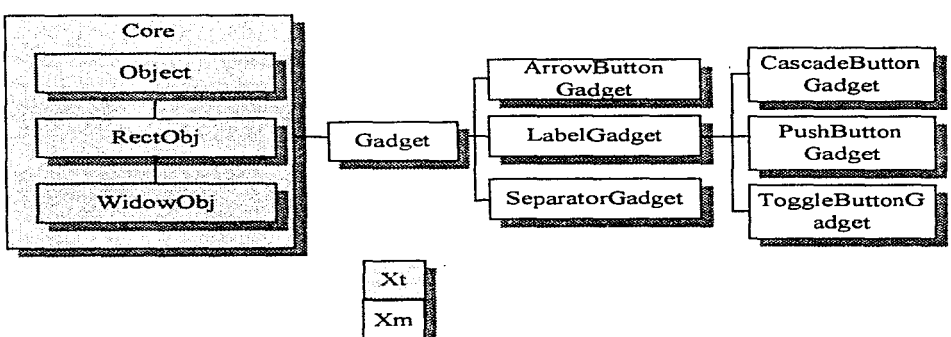


11.4.10.2 Gadgets

Los elementos de esta clase no poseen ventana X propia, sus propiedades las heredan del widget que las contiene (color, bordes, etc.), es decir, son totalmente dependientes de ellos.

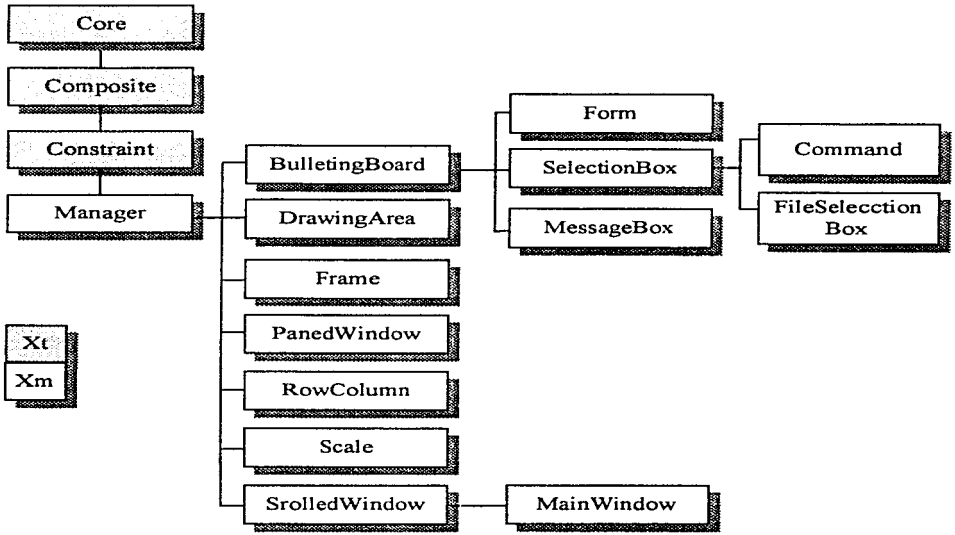
Es importante mencionar que el uso de éstos disminuyen el buen desempeño de la aplicación, incrementando además, el tráfico en la red.

Su diagrama de clases es el siguiente:



11.4.10.3 Clase Manejadora.

Su propósito principal es la definición de tamaño y posición de los widgets que contiene, además de ser una plataforma para gadgets, pero contiene además una gran cantidad de llamadas a funciones y la capacidad de recibir gran número de eventos. Su diagrama jerárquico de clases es:



El pertenecer a la clase Constraint le da las propiedades básicas de un widget capaz de manejar widgets hijos. Además la clase Constraint agrega la capacidad de restringir la posición y tamaño de sus hijos.

Daremos a continuación una breve explicación de cada uno de los elementos mencionados.

BulletinBoard: Define las características de alineación de sus hijos y las áreas donde podrán ubicarse, pero al cambiar su tamaño no cambiará el de sus hijos.

DrawingArea: Área de despliegue de gráficos.

Frame: Provee de bordes a widgets que generalmente carezcan de ellos, por lo que sólo puede tener un hijo.

PanedWindow: Organiza a sus hijos en formato vertical y lo forza al tamaño definido por éste, sin importar la visibilidad

RowColumn: El más utilizado para acomodar widgets en renglones o columnas, sin tener un manejo de las funciones de cada uno de sus hijos.

Scale: Permite el cambio a escala de sus hijos.

ScrollWindow: Ventana con barras de desplazamiento utilizadas en el despliegue de textos o gráficos cuando estos son más grandes que la ventana.

Form: Subclase de BulletingBoard define posición y tamaño de widgets hijos, permitiendo ligarlos a otros.

SelectionBox: Subclase de BulletingBoard que muestra una lista desplegable con posibles opciones para utilizar.

MessageBox: También perteneciente a la clase de BulletingBoard que permite tan solo desplegar mensajes comúnmente acompañados de tres botones (ok, cancel, help).

MainWindow: Ventana principal de la aplicación. Es una subclase de ScrollWindow.

Command: Permite la ejecución de comandos.

FileSelectionBox: Utilizada para mostrar directorios y seleccionar archivos.

11.4.10.4 La Ventana Principal.

Esta clase de widget funciona como un manejador; no crea ninguno de los widgets que maneja. Sin embargo, proporciona facilidad en el control del tamaño y posición de otros widgets, siendo la interfaz principal con el usuario.

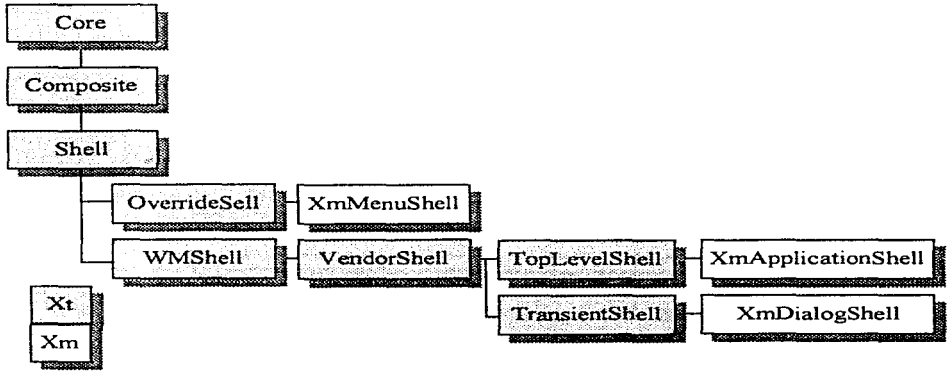
La ventana principal, puede crearse empleando la función: XtVaCreate ManagedWidget(), incluyendo en ésta el tipo xmMainWindowWidgetClass, así como los recursos que se desea asignar a ella. La clase MainWindow es una subclase de las ventanas con barra de desplazamiento por lo que posee todas las características de éstas y cuenta además con la posibilidad de incluir barras de menú, áreas de comandos y mensajes.

11.4.10.5 Menús

Dentro de los elementos más importantes y básicos que contiene la ventana principal, se encuentran tres diferentes tipos de menús, barra de menús, menús que descuelgan opciones y menús a la derecha de la opción seleccionada. En si, las opciones del menú son un widget renglón-columna, que maneja un arreglo de botones (el comportamiento de un botón en cascada es diferente en cuanto a que este último es desplegado en cualquier parte de la aplicación). Pero hay que considerar que Motif provee de menús que son desplegados en cualquier posición de la aplicación, por medio de combinación de teclas, selección mediante ratón, etc.

11.4.10.6 Shells.

En el estricto manejo jerárquico de objetos en Xt, se cuenta con uno que permitirá "montar" sobre él cualquier objeto que se dibuje (incluyendo la ventana principal), éste es el widget shell; cuyo diagrama jerárquico de clases se muestra a continuación:



Los shell wigets deben comunicarse con la ventana manejadora para negociar el estado real de la pantalla (tamaños y posiciones de las ventanas) y otras propiedades. La información que es intercambiada es definida por el Consorcio X de Convenciones para Comunicaciones Intercliente (ICCC), con el objeto de que diferentes distribuidores de sistemas basados en Xt y sistemas de ventanas sean soportados por diversas plataformas.

11.4.10.7 Diálogos

Existen ventanas transitorias llamadas cajas de diálogo que permiten el intercambio de información entre el usuario y la aplicación. Dependiendo de sus funciones se dividen en:

Diálogos de Mensaje: que simplemente proporcionan algún tipo de mensaje al usuario y generalmente incluyen botones que permiten responder al mensaje de manera simple (si, no, cancelar), según el mensaje que despliega se clasifican en mensajes de: error, información, pregunta, advertencia de error y aviso de tiempo de proceso.

Diálogos de Selección: son utilizados cuando el usuario necesita proporcionar una respuesta mas amplia ya sea seleccionando un valor dentro de una lista de opciones o introduciendo el texto directamente.

Diálogos personalizados: son aquellos que involucran diversos elementos no contemplados en los diálogos definidos por motif, pero se apegan a las definiciones de este.

Todos los diálogos consideran dos componentes principales:

- El área de control o área de trabajo, que despliega la pregunta a responder por el usuario, y
- El área de acción, que despliega las posibles opciones de respuesta.

Cada región es conceptual y puede ser representada por múltiples widgets.

12. MONITOR GRÁFICO DE RED

12.1 Objetivo

Proporcionar un "Monitor gráfico de red como herramienta inicial en la administración de redes", que resuelva la necesidad primaria de conocer el estado básico de los principales dispositivos de la red.

12.2 Alcance

Desarrollo de una herramienta que permita conocer el estado básico en tiempo real de los principales dispositivos de la red de manera clara y sencilla.

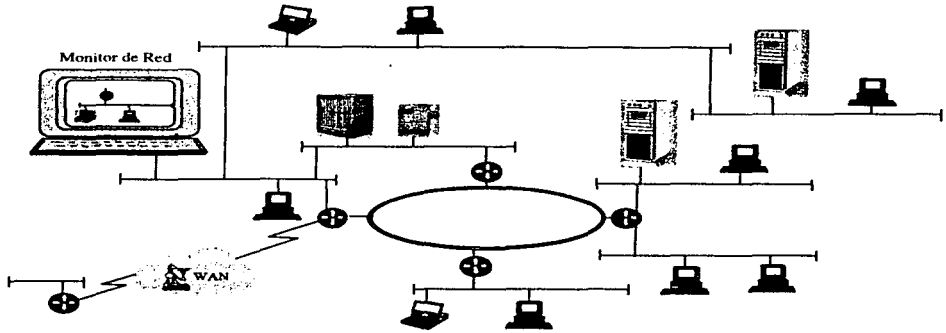
12.3 Plataforma

Siendo TCP/IP el conjunto de protocolos de mayor uso en la actualidad para la implantación de inter-redes, se toman como requerimientos a satisfacer el hardware y software que lo soporten en el mayor número de equipos.

Como parte de los criterios a considerar en la implantación de un monitor de red, se encuentra la selección de hardware, de los protocolos de comunicación y topología entre los diferentes equipos, de modo tal, que para la selección del equipo utilizado para el monitoreo se deben considerar las siguientes características:

- El protocolo de transporte deberá soportar al protocolo de administración seleccionado.
- El uso de una presentación amigable, es necesario para proporcionar facilidad en la administración, por lo tanto el equipo debe contar con capacidad de despligue gráfico.
- Capacidad de procesamiento y almacenamiento para obtener una alarma lo mas pronto posible en caso de presentarse alguna falla en cualquier equipo.
- El nodo deberá estar ubicado de manera estrategica, según la(s) topología(s) existente(s), lo cual dependerá de su fácil acceso a cada nodo de la red contando con el mayor número de alternativas de rutas.

Así, podemos ejemplificar un esquema típico de red donde se indica el mejor punto de ubicación para el monitor de red, el esquema siguiente es un ejemplo de ello.



El presente desarrollo se realiza en lenguaje Motif, C y shell de UNIX,. Por lo cual considera que la estación de trabajo que se utilice como estación monitora, debe contar con:

- Sistema operativo UNIX.
- Interfaz gráfica XWindow.
- Soporte del conjunto de protocolos TCP/IP.

12.4 Consideraciones

Con el fin de hacer del monitor de red una herramienta que cubra la necesidad básica de conocer el estado de los dispositivos en tiempo real, se hacen las siguientes consideraciones para el sistema monitor de red:

1. Contar con capacidad de descubrimiento de dispositivos activos en la red.
2. Que sea de fácil manejo para el administrador.
3. Obtención de la información básica de cada dispositivo incluido en el monitor.
4. Almacenamiento de información histórica.
5. Ayuda en línea.

12.5 Desarrollo

12.5.1 Diagrama Estructural

El sistema consta con las sigueintes funciones principales:

Definición de dispositivo: consiste en la captura de los datos generales que identifican a cada uno de los dispositivos a monitorear.

Definición de rango de direcciones IP: consiste en la captura de direcciones inicio y fin utilizado en el descubrimiento automático.

Definición de representación gráfica del dispositivo: aquí se realizará el dibujo (o bien optar por seleccionar el predefinido) con el que se quiera representar cada uno de los elementos que se monitorearán.

Descubriemiento de dispositivos en la red: dado que en muchas ocasiones no se conoce con certeza todos y cada uno de los dispositivos que se encuentran activos en la red, el sistema cuenta con esta función que permite al usuario la definición de un rango de direcciones IP para que el sistema evalúe si los dispositivos se encuentran activos o no.

Despliegue gráfico: este proceso realizará el despliegue de todos los dispositivos definidos para el monitoreo, según la definición del gráfico que se haya seleccionado como la representación de cada uno, actualizando el despliegue adecuado según el resultado del monitoreo.

Monitoreo: es la función principal del sistema, ya que ésta es la que de manera permanente verifica el estado de los componentes de la red, definidos en el sistema de monitoreo para su actualización en el despliegue.

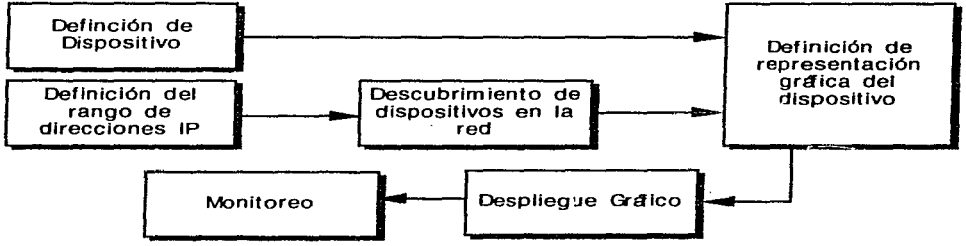
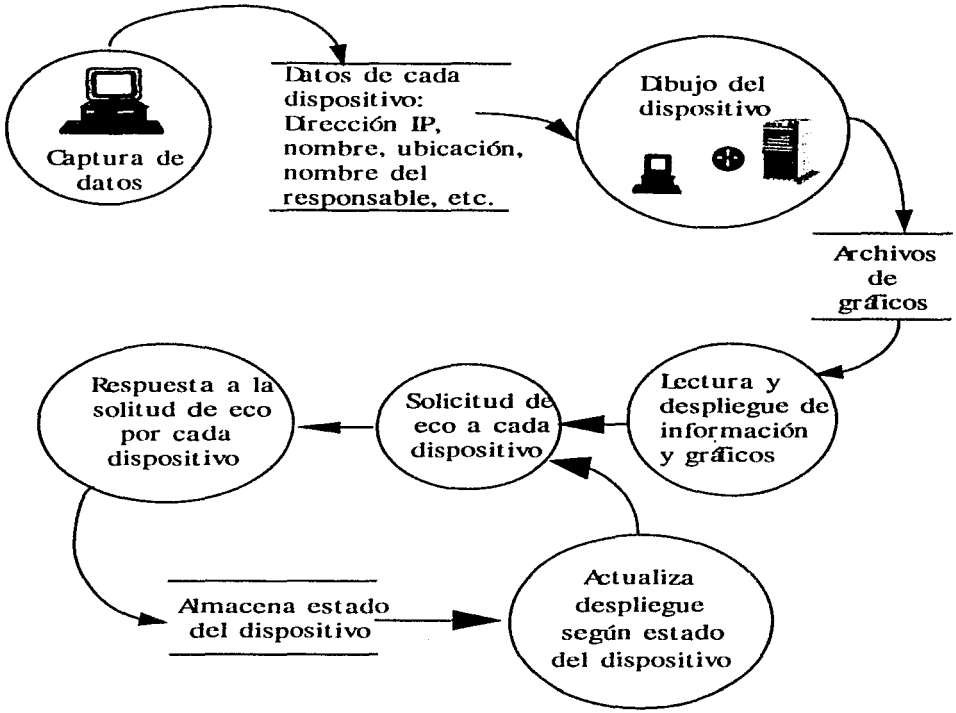


Diagrama Estructural de Funciones

12.5.2 Diagrama de Flujo de Datos



12.5.3 Esquema de Pantallas

12.5.3.1 Pantalla Principal

Archivo	Herramientas	Ayuda
Nuevo	Descubrimiento automático	
Abrir	Ver bitácora de estados	
Borrar	Trazar ruta	
Salir		

Esta pantalla deberá estar siempre abierta mientras se ejecute la aplicación, ya que contiene el widget padre de todos los objetos desplegados por la aplicación, además contendrá el código de colores de despliegue para identificar fácilmente el estado de los dispositivo, el cual es como se muestra a continuación:

La función de Salir terminará la aplicación completamente, por lo que al aplicarla cerrará todas las "pantallas de mapa" que se encuentren abiertas.

Las funciones del menú de Archivo realizarán las acciones indicadas por su nombre, en relación a los archivos de almacenamiento de direcciones de dispositivos, que llamaremos "archivos de mapas" *, con el fin de identificar fácilmente la información a que se hace referencia.

La opción de Descubrimiento automático del menú de Herramientas realizará la petición de respuesta de eco para todos y cada uno de los dispositivos que se encuentren dentro de un rango de direcciones IP especificado, y posteriormente realizará el despliegue gráfico de aquellos dispositivos que contestaron dicha petición.

Las opciones Ver bitácora de estados y Trazar ruta servirán como un pequeño apoyo de monitoreo para conocer: 1) los diferentes estados que ha presentado un dispositivo desde el inicio del proceso de monitoreo hasta el momento y 2) conocer la ruta por la que pasan los datagramas, respectivamente.

La opción de ayuda en este menú y en general en toda la aplicación desplegarán un texto de ayuda para poder llevar a cabo la realización del monitoreo.

---

* Ver detalle de "archivo de mapa" en la sección 12.5.5

12.5.3.2 Pantalla de Mapa

Archivo	Dispositivo	Ayuda
Guardar como...	Nuevo	
Cerrar	Modificar	
	Eliminar	

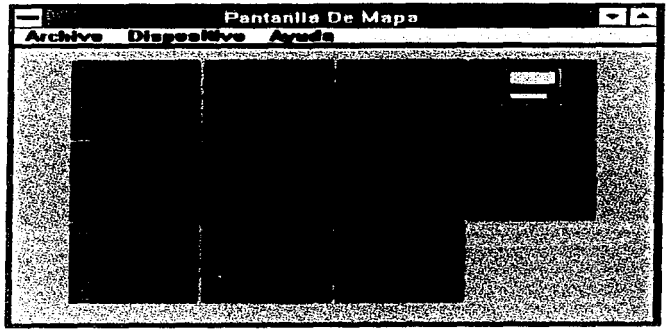
Esta pantalla realizará el despliegue gráfico de cada uno de los dispositivos definidos para monitorear su estado, dado que cada pantalla se encuentra asociada a un "archivo de mapa" * se podrán tener tantas pantallas de mapa como los recursos de la estación monitora lo permitan.

La opción de Guardar como del menú de archivo permite realizar la copia del archivo de mapa por si se desea modificar sin perder el detalle anterior; es importante considerar que se omite la opción grabar debido a que el archivo se encuentra en constante utilización y simepre está siendo accesado y actualizado por el proceso mismo.

En el menú de Dispositivo se contemplan las funciones básicas para actualizar el archivo de mapa, en cuanto a altas, bajas y cambios de dispositivos dentro de éste.

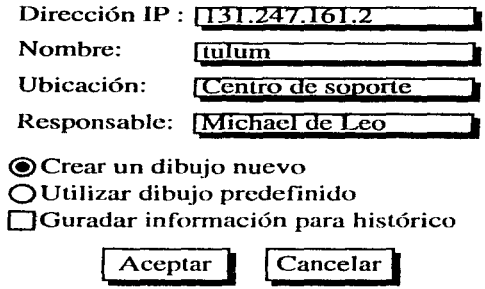
La opción de Cerrar sólo cerrará la ventana que contiene a la pantalla de mapa.

Un ejemplo del despliegue de esta ventana en operación sería:



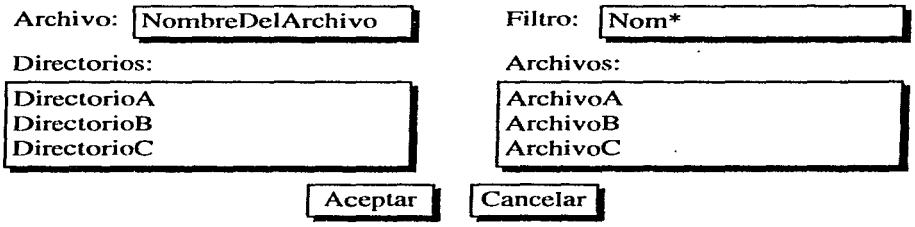
* Ver detalle de "archivo de mapa" en la sección 12.5.5

12.5.3.3 Pantalla de definición de dispositivo



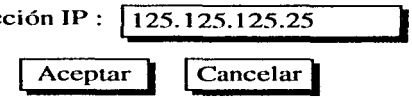
Esta pantalla se utilizará para la definición de los datos básicos del dispositivo, donde el único dato requerido para monitorear el dispositivo es la dirección IP.

12.5.3.4 Pantalla de Archivos



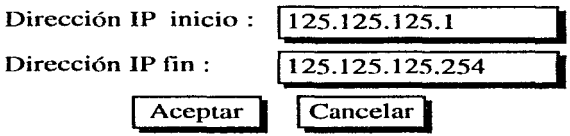
Esta pantalla será utilizada de manera general en la aplicación siempre que se necesite especificar un nombre de archivo para la realización de las acciones.

12.5.3.5 Pantalla de Especificación de Dispositivo



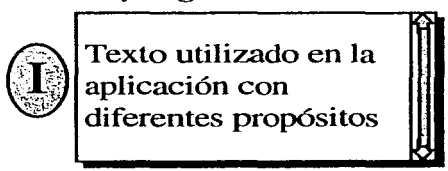
Pantalla de uso general utilizada para que el usuario especifique la dirección IP, de un dispositivo para la realización de una acción determinada.

12.5.3.6 Pantalla de Rango de Direcciones



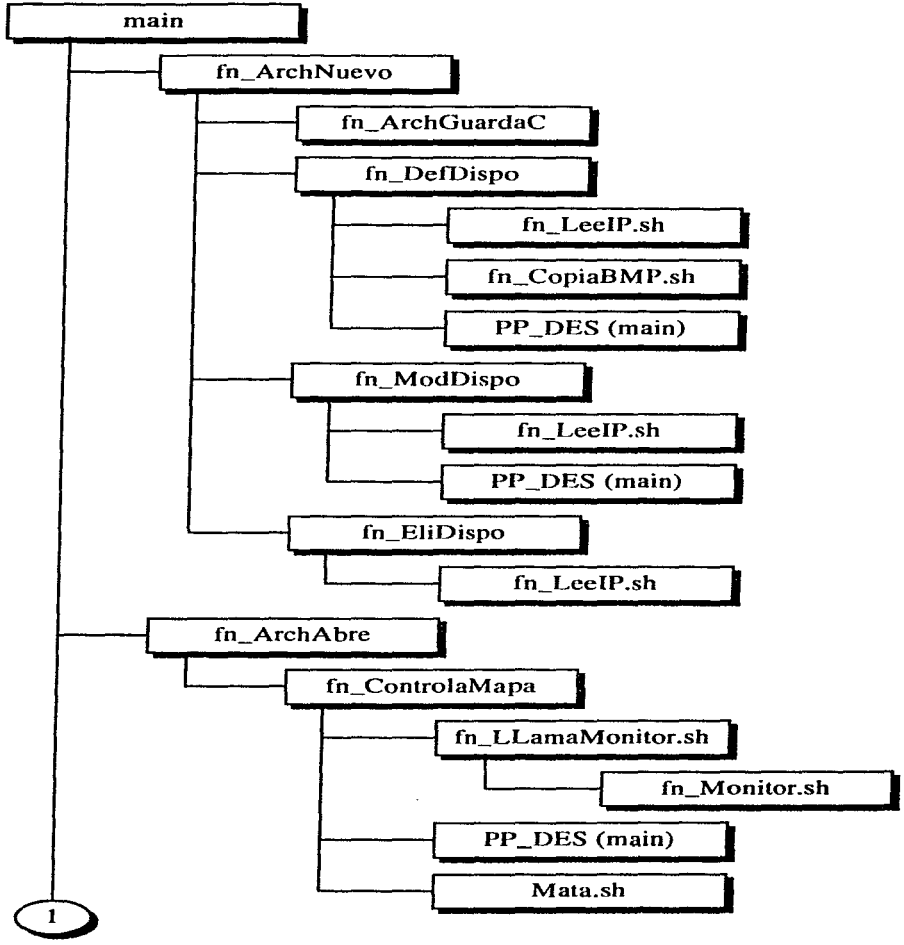
Esta pantalla solamente es utilizada en la función de Descubrimiento automático en donde es necesario que el usuario especifique el rango de búsqueda.

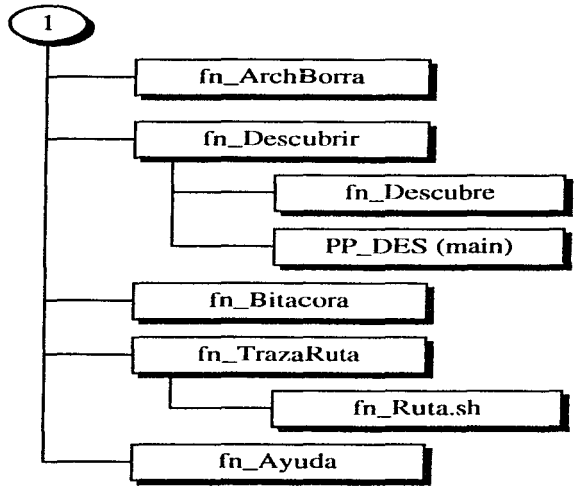
12.5.3.7 Pantalla de Despliegue de Textos



Pantalla general utilizada en toda la aplicación para el despliegue de texto ya sea de información almacenada en archivos que no cambian, como lo son los textos de ayuda; así como de información resultado de una operación, como: la bitácora de estados y la ruta que siguen los datagramas.

12.5.4 Funciones principales del Sistema





Función	**Descripción**	**Nombre del archivo**
main	Crea la ventana principal del sistema donde se define de manera global el contexto gráfico, widget padre de todos los objetos utilizados y el archivo de trabajo.	MtfVentana.c
fn_ArchNuevo	Genera la(s) ventana(s) hija(s) donde se desplegarán los mapas de dispositivos para su monitoreo.	Mtf_fn_ArchNuevo
fn_ArchGuardaC	Genera una copia del archivo que se está monitoreando con un nombre específico dado por el usuario.	Mtf_fn_ArchGuardaC

Función	Descripción	Nombre del archivo
fn_ DefDispo	Despliega la ventana para registrar un nuevo dispositivo a ser monitoreado en el mapa.	Mtf_fn_ DefDispo
fn_LeeIP.sh	Toma como parámetro el texto de una ventana para interpretar éste como una dirección IP.	LeeIP.sh
fn_CopiaBMP.sh	Copia un archivo BMP predefinido para ser asignado como la representación gráfica de un nuevo dispositivo.	CopiaBMP.sh
PP_DES	Programa utilizado en la actualización gráfica.	PP_DES
fn_ModDispo	Despliega la ventana con los datos del dispositivo especificado para ser modificados.	Mtf_fn_ModDispo
fn_EliDispo	Elimina de la pantalla de monitoreo, del archivo de direcciones del mapa y del archivo de bitácora el dispositivo especificado.	Mtf_fn_EliDispo
fn_ArchAbre	Despliega una nueva ventana de mapa con los dispositivos que se hallan especificado dentro de éste.	Mtf_fn_ArchAbre
fn_ControlaMapa	Programa que abre el mapa de botones y permanece en ciclo infinito de monitoreo, actualizando el color de cada gráfico en el mapa según el estado de éste.	Mtf_fn_ControlaMapa
fn_LLamaMonitor.sh	Realiza la llamada a la función de monitoreo.	LLamaMonitor.sh

Función	Descripción	Nombre del archivo
fn_Monitor.sh	Esta función utiliza una implementación de ICMP, haciendo una solicitud de eco a cada uno de los dispositivos en el mapa.	Monitor.sh
fn_Mata.sh	Obtiene el número del proceso que se encontraba en ciclo infinito de monitoreo para concluirlo.	Mata.sh
fn_ArchBorra	Elimina el archivo que se encuentra en despliegue y cierra la ventana de mapa.	Mtf_fn_ArchBorra
fn_Descubrir	Despliega una pantalla de captura de dos direcciones IP.	Mtf_fn_Descubrir
fn_Descubre	En base a dos direcciones capturadas, el sistema envía una solicitud de respuesta de eco a cada uno de los dispositivos que se encuentran dentro del rango especificado; el sistema mostrará entonces una nueva ventana de mapa que despliegue los dispositivos que hallan tenido una respuesta exitosa.	Descubre
fn_Bitacora	Despliega los diferentes estados del dispositivo especificado por el usuario durante el periodo de monitoreo.	Mtf_fn_Bitacora
fn_TrazaRuta	Despliega la ruta que siguió la solicitud de eco para llegar al destino especificado por el usuario.	Mtf_fn_TrazaRuta

Función	Descripción	Nombre del archivo
fn_Ruta.sh	Realiza la acción de llamada a una implementación de ICMP, para solicitar respuesta de cada uno de los dispositivos por los que pasa el datagrama antes de llegar al destino final.	Ruta.sh
fn_Ayuda	Despliega una ventana de texto de apoyo para la operación.	Mtf_fn_Ayuda

12.5.5 Archivos de Almacenamiento

Nombre del Archivo	Contenido	Descripción
ARCHIVO DE MAPA Nombre:{dirección IP}.DAT	Dirección IP Nombre del dispositivo Ubicación Responsable Estado del dispositivo Indicador del guardado de bitácora	Se generarán tantos archivos de este tipo como el usuario desee, su función básica es la de agrupar un conjunto de dispositivos a ser monitoreados en una sola ventana.
ARCHIVO DE BITACORA Nombre:{dirección IP}.BIT	Fecha Hora Respuesta a la petición de eco del dispositivo.	Se generará un archivo de este tipo por cada dispositivo seleccionado con la opción de guardar bitácora.
ARCHIVO DE BITMAP Nombre:{dirección IP}.BMP	Bitmap	Se generará un archivo de este tipo por cada dispositivo monitoreado.

CONCLUSIONES

El desarrollo del presente trabajo nos ha permitido:

- Observar a la función de administración de red, como parte de la eficiencia misma del servicio que se ofrece, es decir, el personal encargado de dar servicio y soporte a las redes de computadoras deberá ser responsable de considerar los diferentes tipos de usuarios, los objetivos y prioridades de cada uno de ellos y con ésto delimitar el alcance y la manera de que esto se lleve a cabo.
- Identificar como una necesidad primordial para lograr el control y mejoramiento de servicios de red, que las personas involucradas en el servicio, conozcan el entorno en que se trabaja:
    * Los recursos de apoyo para monitoreo y/o administración de los equipos que conforman la red, propios del ambiente y los específicos con que se cuente.
    * Las necesidades de los usuarios finales.
    * El esquema de comunicación global y particular.
    * Necesidades propias de los administradores de la red.
    * Niveles de seguridad.
    * Horarios críticos de servicio.
    * Mecanismos y/o equipo de contingencias.

- Ejemplificar con el desarrollo de una herramienta básica de monitoreo, que es posible la implantación de mecanismos de control que permitan el mejoramiento de servicios de soporte, sin contar con herramientas sofisticadas y/o costosas.

Habiendo cumplido con los objetivos planteados al inicio del desarrollo del presente trabajo, mostramos a continuación los puntos más importantes que seguirían al desarrollo del presente:

- El desarrollo de un sistema administrador de red, que contemple funciones tales como:

* De control de los diferentes aspectos de administración de redes mediante el uso de protocolos específicos para esta función y presentación de información de manera estadística y específica.
* Mecanismos de alarma y notificación a administradores de manera personalizada (correo, mensajes de radio, etc.)
* Mecanismos emergentes de corrección automática de fallas.
* Control jerárquico de mapas.
* Almacenamiento de la información mediante el uso de un DBM.

Así, observamos con gran satisfacción que la formación recibida en la Facultad de Ingeniería nos ofrece las bases necesarias para profundizar en el desarrollo de una área inherente a la carrera de Ingeniería en Computación.

ANEXO A.

SOCKETS

La interface con el usuario

Los protocolos de la internet no especifican un estándar para la interface entre aplicaciones y el software, y los detalles específicos varían de un sistema a otro. Cabe mencionar que en muchos sistemas los protocolos se implementan en el kernel del sistema operativo, haciendo la interface del usuario especifica al sistema que empleen.

La mayoría de las implementaciones TCP/IP ofrecen una interface de programación que sigue un sólo modelo; la interface de programación de sockets. Esta interface fue introducida en un principio en 1982 con una versión BSD de UNIX.

La interface de programación de sockets fue diseñada para emplearse con diversos protocolos de comunicación, no exclusivamente para TCP/IP, sin embargo al terminarse la especificación de transporte para OSI se encontró que dicha interface no satisfacía los requerimientos generales de OSI.

En 1986, AT&T introdujo la interface del nivel de transporte (TLI) para el sistema V de UNIX; interface que puede emplearse con el nivel de transporte OSI, TCP y otros protocolos.

Sin embargo los sockets son la API más empleada, considerada como estándar por su disponibilidad.

Orientación en UNIX

La interface de sockets original fue escrita para el sistema operativo UNIX. La arquitectura de dicho sistema proporciona un marco de trabajo en el que los estándares para los archivos, terminales y comunicación de I/O, operan de forma similar. Por ejemplo, cuando un programa abre un archivo, al programa le es asignado un entero, el cual es denominado *descriptor*, y es usado por el programa para identificar al archivo en operaciones subsecuentes. Las operaciones son ejecutadas en el archivo por medio de llamadas tales como:

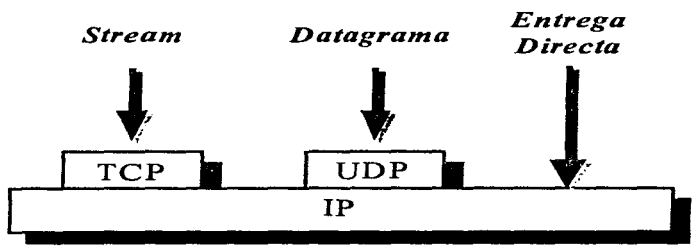
- read(descriptor, buffer, longitud)
- write(descriptor, buffer, longitud)

- close(descriptor)

Para la comunicación con sockets en TCP/IP se emplea un marco de trabajo idéntico. La diferencia principal entre la interface de programación de socketes y una interface de I/O para archivos en UNIX, es que se necesita cierto número de llamadas preliminares para ensamblar toda la información necesaria para la comunicación.

Servicios de los Sockets.

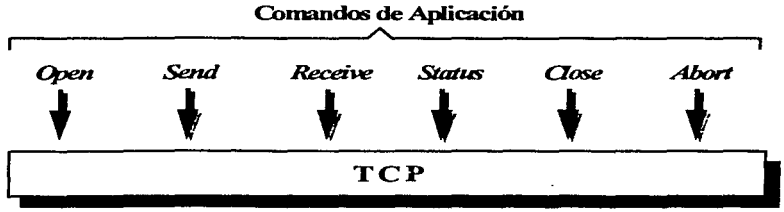
Los sockets pueden usarse para la comunicación de sistemas que realiza TCP, para comunicación de datagramas en UDP y para la simple entrega de datagramas al nivel IP. En la figura, se muestran estos servicios:



Interfaces de Programación de Sockets

La interface del Servicio TCP

Recuérdese que la definición de los servicios de las interfaces no indican exactamente como deben implementarse las funciones, sino proporcionar las guías para los comandos y parámetros que necesitan pasarse através de la interface entre 2 niveles. En este caso, como se muestra en la figura, la interface reside entre una aplicación y un proveedor de servicio TCP. Los comandos de la interface son:



Interfce del Servicio TCP

Apertura de la conexión

Un comando OPEN es usado para preparar la comunicación. El servidor ejecuta una apertura pasiva que inicializa la comunicación. La creación de una estructura de datos que mantendrá información de comunicación es un elemento muy importante en la implementación de una "apertura".

Bloque de Control de Transmisión (TCB)

La estructura de datos en la que TCP guarda toda la información relacionada con una conexión es llamada un bloque de control de transmisión (TCB). Generalmente, existen más de 50 parámetros en un TCP. A continuación se listan algunos para dar idea del tipo de información que se incluye:

- La dirección IP local.
- El protocolo (TCP, UDP).
- Puerto remoto.
- Estado actual de TCP.
- Tiempo estimado de viaje redondo.
- Valor de tiempo fuera actual para la retransmisión.
- Tamaño actual de la ventana de envío.
- Número de secuencia del último byte que fue confirmado.
- Número de secuencia del siguiente byte a enviarse.
- El puerto local.
- Dirección IP remota.
- Tamaño del buffer del transmisor.

- Tamaño del buffer del receptor.
- Desviación estimada en el viaje redondo.
- Número de retransmisiones que su han enviado.
- Tamaño máximo del segmento máximo de recepción.

El comando "OPEN"

Existen 3 tipos de comando "open". Los dos primeros habilitan un servidor para que esté al tanto de las peticiones que lleguen, mientras que la tercera es usada por un cliente para iniciar una conexión de manera activa. Una operación open causa la creación de una estructura TCB.

Un comando open regresa el nombre de una conexión local. Una aplicación usa este nombre en todas las referencias futuras de la conexión.

En la interface de programación de sockets, este nombre es un entero.

- Inicio Pasivo no Especificado (Unspecified Passive Open). Permite a un servidor escuchar la petición de conexión de algún cliente.
  - Input. Puerto local, [timeout], [precedencia], [parámetros de seguridad], [tamaño máx. del segmento]
  - Output. Nombre de la conexión local.
- Inicio Pasivo Especificado Totalmente (Fully Specified Passive Open). Permite al servidor escuchar la petición de conexión de un cliente específico.
  - Input. Puerto local, dir. IP remota, puerto destino, [timeout], [precedencia], [parámetros de seguridad], [tamaño máx. del segmento]
  - Output. Nombre de la conexión local.
- Inicio Activo. Permite a un cliente iniciar activamente la conexión con un servidor.
  - Input. Puerto local, dir. IP destino, puerto destino, [timeout], [precedencia], [parámetros de seguridad], [tamaño máx. del segmento]
  - Output. Nombre de la conexión local.

El comando OPEN y la Interface de Sockets

El comando "open" corresponde a una secuencia de funciones de la interface socket para inicializar un TCB y establecer los valores de un número de variables. El parámetro opcional de timeout en las llamadas "open" intenta establecer un límite para la entrega exitosa y datos cualesquiera cedidos a TCP. Si el tiempo fuera expira, entonces TCP debe abortar la operación.

Cabe hacer notar que la mayoría de las implementaciones codifican un valor default para el timeout, de modo que un programador que usa la interface socket puede no percatarse de este parámetro.

Envío y Recepción de Datos

Los comandos "send" y "receive" mueven los datos entre una aplicación y TCP.

- Send. Permite a un cliente o servidor pasar a TCP un buffer de datos para la transmisión.
  - Input. Nombre de la conexión local, dir. del buffer, cuenta de bytes, bandera PUSH, bandera URGENT, [tiempo fuera].
- Receive. Identifica un buffer receptor para los datos que llegan.
  - Input. Nombre de la conexión local, dir. del buffer, tamaño del buffer en bytes.
  - Output. Núm. de bytes ubicados realmente en el buffer, bandera PUSH, bandera URGENT.

Primitivas *Send/Receive* y la Interface Socket

Para la programación de un socket pueden emplearse las llamadas send() o write() que son muy similares a las funciones empleadas en la escritura de un archivo.

Existen también 2 llamadas que concatenan los datos que son almacenados en una secuencia de buffers y luego los envían, estas llamadas son sendv() y writev(). Cada una de las llamadas anteriores ocasionan automáticamente que los datos sean forzados a ser enviados.

Por otra parte la función "receive" es implementada usando las llamadas recv() y read() que haciendo la analogía con la lectura de un archivo resultan muy similares. Las llamadas que acumulan una secuencia de buffers con los datos que llenan son *recvmsg()* y *readv()*.

Otros Comandos

Los comandos restantes y que a continuación se presentan son utilizados tanto para obtener el estado de una conexión como para terminarla:

- Status. Obtiene información sobre una conexión.
  - Input. Nombre de la conexión local.
  - Output. Depende de la implementación, pero generalmente incluye la dirección de los sockets local y remoto, nombre de la conexión local, ventana receptora, ventana transmisora, estado de la ventana, número de buffers que esperan confirmación, número de bufares pendientes de recepción, estado urgente, precedencia, información de seguridad, timeout de transmisión.
- Close. Solicita el cierre de la conexión.
  - Input. Nombre de la conexión local.
- Abort. Solicita a TCP que descarte los datos que actualmente se encuentran en los buffers de envío y recepción y aborta la conexión.
  - Input. Nombre de la conexión local.

Relación con la Interface Socket.

El estado del socket se obtiene mediante la función "*getsockopt()*", así como por otras funciones del sistema local. Tanto *close* como *abort* son implementadas mediante una llamada "close()".

Llamadas de Bloqueo y no Bloqueo.

El estado típico de un servidor TCP es en el que un proceso maestro se mantiene la mayor parte del tiempo esperando a que los clientes demanden sus servicios.

Cuando un cliente se conecta, que es comúnmente el caso en el que el servidor crea un proceso, que es el que realiza el trabajo para el cliente, pasa el cliente al nuevo proceso y regresa al modo de espera.

Algunas veces, los clientes llegan más rápido que lo que el proceso maestro puede obtenerlos. El mecanismo estándar para su manejo es que cuando el maestro inicie se le indique a TCP que cree una cola que pueda mantener un número específico de peticiones. Los clientes que no pueden ser servidos inmediatamente son puestos en la cola y servidos en turno.

Apertura Pasiva del Servidor TCP

Un comando de apertura pasiva en un servidor es implementada haciendo uso de una serie de llamadas:

socket() El servidor identifica el tipo de comunicación (TCP en este caso). El sistema local crea una estructura de datos TCB apropiada para la comunicación y regresa un nombre de conexión local. El nombre de la conexión es un entero pequeño llamado descriptor del socket.

bind() El servidor establece la dirección IP local y el puerto que quiere usar. Recuérdese que un servidor puede tener direcciones IP múltiples. El servidor puede indicar una dirección IP, o aún indicar que se encuentra disponible para aceptar las conexiones que llegan de cualquier dirección local. El servidor puede solicitar un puerto local específico, o dejar a la llamada "bind" la obtención de un puerto libre que pueda usarse.

listen() El servidor asigna la longitud de la cola del cliente.

accept() El cliente está listo para aceptar conexiones del cliente. Si la cola no está vacía, se acepta la primera petición de conexión del cliente. La llamada "accept()" crea un nuevo descriptor del socket que será usado para esta conexión del cliente y regresa este nuevo descriptor al servidor. Usualmente se usa una forma asíncrona de accept, de modo que si la cola se encuentra vacía, accept() esperará petición del siguiente cliente antes de regresar.

Apertura Activa del Cliente TCP

socket(). El cliente indica el tipo de comunicación (TCP en este caso). El sistema local crea una estructura de datos TCB apropiada para la comunicación y regresa un descriptor del socket local.

connect() El cliente identifica la dirección IP de un servidor y el puerto. TCP intentará establecer una conexión con el servidor.

Si el cliente desea especificar exactamente el puerto local que quiere usar, el cliente debe hacer una llamada bind() antes de la llamada connect(). Si el puerto está disponible, bind() lo asigna al cliente.

Si el cliente no llama a bind() para solicitar un puerto, entonces la llamada connect() asigna al cliente un puerto no utilizado. El número de puerto se incluye en el TCB.

Otras Llamadas

Las llamadas restantes son usados de igual forma tanto para el cliente como para el servidor.

Los parámetros de entrada, en las llamadas send() y recv() son específicas a los sockets y soportan el envío y recepción de datos urgentes, así como de datos ordinarios. Las llamadas read() y write() son diseñadas para aparecer como archivos de I/O ordinarios, y no pueden reconocer datos urgentes.

send() Escribe un buffer de datos para el socket. Puede usarse la llamada write() alternativamente.

sendv() Pasa al socket una secuencia de buffers. Alternativamente puede usarse la función writev().

recv() Recibe del socket un buffer de datos. Otra alternativa para esta llamada es la función readv().

recvmsg() Recibe del socket una secuencia de buffers. Otra alternativa es readv ( ).

close () Termina una conexión y cierra el socket.

getsockopt() Lee la información del TCB. Opcionalmente, un sistema puede proporcionar llamadas adicionales al sistema de I/O que pueden usarse para leer varias partes del TCB.

A continuación se indican algunas funciones que pueden sustituir a algunos defaults:

getsockopt() Asigna un número de parámetros TCB tales como tamaños de buffer de entrada y salida, uso de bitácora, posibilidad de que los datos urgentes debieran ser recibidos en el orden de la secuencia normal y si un cierre debe bloquearse hasta que todos los datos de salida han sido enviados de manera segura.

iocntl() o fcntl() Asigna el I/O del socket como de bloqueo o no bloqueo.

Interface de Programación del Socket UDP

La interface de programación para TCP es más complicada que la interface para UDP. En UDP las llamadas socket() y bind() se completan rápidamente y tienen un retorno inmediato.

ANEXO B.

ASN.1

Notación de Sintaxis Abstracta. Uno o bien ASN.1, fue desarrollada por CCITT e ISO como lenguaje formal, para:

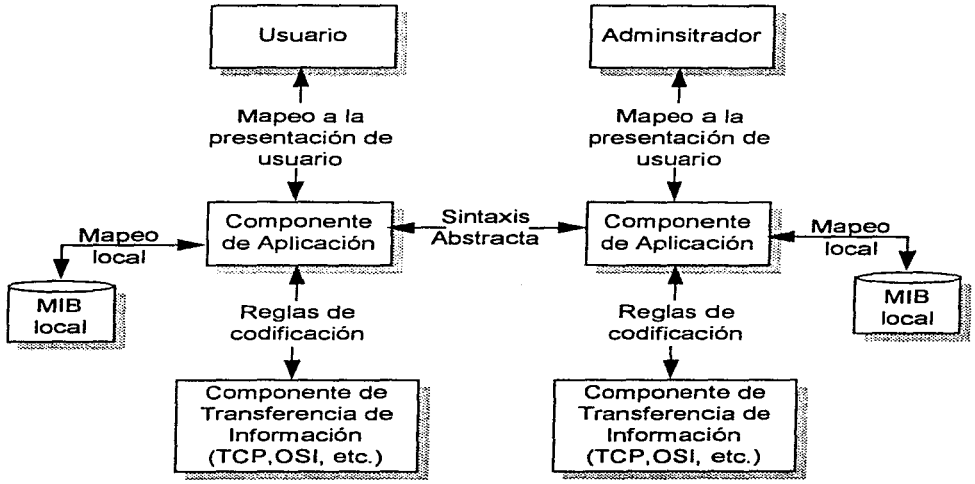
- La definición de sintaxis abstractas de información de aplicación.
- Definición de estructuras de aplicación y presentación de unidades de datos de diferentes protocolos.
- Utilizada en la definición de la MIB para SNMP y para la Administración en OSI.

La sintaxis abstracta es utilizada por el componente de transferencia de información, la información recibida desde una aplicación es especificada en valores binarios que pueden ser directamente ensamblado en SDUs (service data unit) para el paso dentro niveles y dentro de PDUs, para pasar entre entidades del protocolo dentro del mismo nivel. El componente del nivel de aplicación se concentra así en la vista final que tendrá el usuario de la información ya sea en base de datos, documentos, tipo texto o imágenes.

El componente de aplicación, presenta información (en formato binario) representada en una sintaxis abstracta que acepta tipos de datos y valores de datos. Esta sintaxis abstracta es utilizada para el intercambio de información entre componentes de aplicación de diferentes sistemas mediante PDUs. Así un protocolo de aplicación describe sus PDUs en términos de una sintaxis abstracta, utilizada para el intercambio de información. La sintaxis utilizada además de ser mapeada para ser empleada por el usuario final, debe serlo para un formato de almacenamiento. El componente debe traducir entre la sintaxis abstracta de la aplicación y la sintaxis de transferencia (que describe la información en modo binario) utilizada por la interacción del componente de transferencia de información (Ejemplo: sintaxis abstracta char, sintaxis de transferencia ASCII, EBCDIC).

Entonces la sintaxis de transferencia define la representación de la información.

Para lograr la traducción entre las dos sintaxis se especifican reglas de codificación y representación de cada valor así como cada tipo de dato. De este modo se obteniene así representación similar para el intercambio de información en ambientes distribuidos y heterogéneos, entre diferentes aplicaciones en un solo sistema. El esquema general de intercambio de información es:


Usuario
Adminsitrador
Mapeo a la presentación de usuario
Mapeo a la presentación de usuario
Mapeo local
Componente de Aplicación
Sintaxis Abstracta
Componente de Aplicación
Mapeo local
MIB local
MIB local
Reglas de codificación
Reglas de codificación
Componente de Transferencia de Información (TCP,OSI, etc.)
Componente de Transferencia de Información (TCP,OSI, etc.)

ANEXO C

MANUAL DE USUARIO

Instalación

Para realizar la instalación del sistema monitor deberá descomprimir el archivo monitor.tar.Z con el siguiente comando:

uncompress monitor.tar.Z

Este comando dejará el archivo en formato tar, por lo que deberá ejecutarse el comando:

tar xvf monitor.tar

Lo anterior extraerá una serie de archivos entre los cuales se encuentra el Makefile requerido para la creación de los programas ejecutables, la acción necesaria para crear estos ejecutables es:

make

Para ejecutar el programa basta con teclear la palabra:

monitor

Operación

Consideración: el presente manual asume que el usuario conoce la forma de trabajar en ambiente Xwindow.

Las opciones de Aceptar y Cancelar corresponderán en todos los casos a la realizar la operación seleccionada (indicada en el título de la ventana) o a abortar la operación, respectivamente.

Inicio

Una vez inicada la aplicación la primera pantalla desplegada es la siguiente:

Esta pantalla controlará la activación o término de todas las otras pantallas con que se trabaje en esta aplicación, por lo que es importante considerar que esta pantalla no deberá cerrarse a menos de que se desee terminar con todos los procesos de monitoreo activos.

Esta pantalla contiene de manera fija el código de colores para la identificación del estado de cada dispositivo monitoreado en donde:

Falla (rojo): significa que no es posible el intercambio de información con ese dispositivo.

Inestable (amarillo): indica que existe perdida de datos en el envío o recepción.

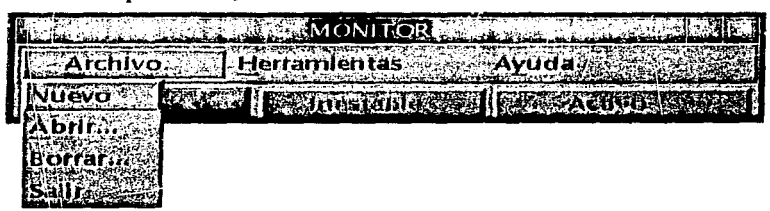
Activo (verde): señal de que existe comunicación con este dispositivo.

Crear, Abrir o Borrar un mapa de monitoreo

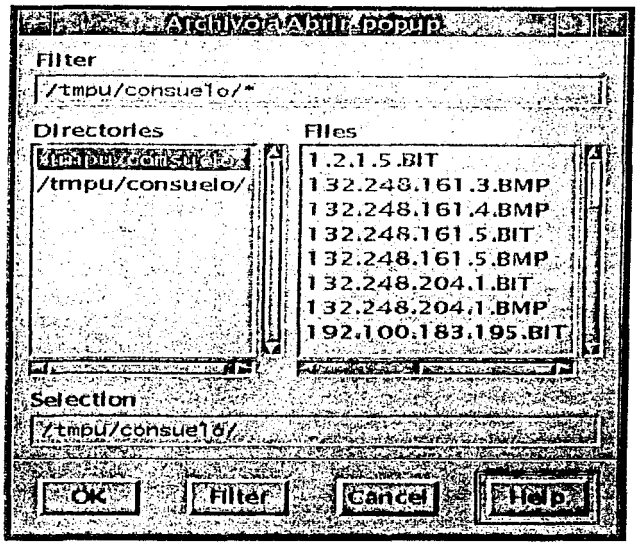
El monitoreo de dispositivos se llevará a cabo en base a la definición de "mapas", esto es: se definirá un archivo en el cual se especificarán cada uno de los dispositivos a controlar.

Es importante mencionar que existe la facilidad de generar tantos archivos como se desee para ordenar el despliegue de dispositivos monitoreados según los requerimientos de operación u organización con que se cuente, pudiendo tener varias pantallas de mapa monitoreando a la vez.

Para la realización de un nuevo mapa de monitoreo deberá seleccionarse la opción Nuevo del menú Archivo, lo cual desplegará la pantalla de Mapa (ver Definición de dispositivos).



Si el archivo de especificación de dispositivos ya existe y desea activarse, entonces deberá seleccionar la opción Abrir del menú Archivo, lo cual desplegará la ventana que se muestra a continuación, para la selección del archivo deseado.

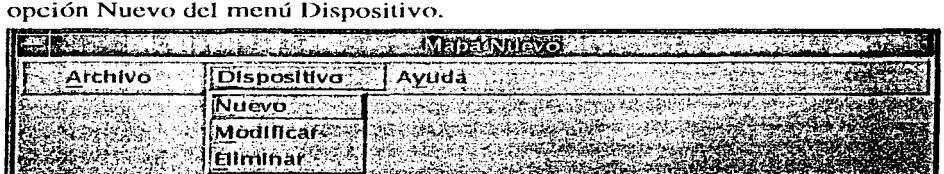


El menú de Archivo cuenta además con la opción de Borrar, la cual permite la eliminación de un archivo existente.

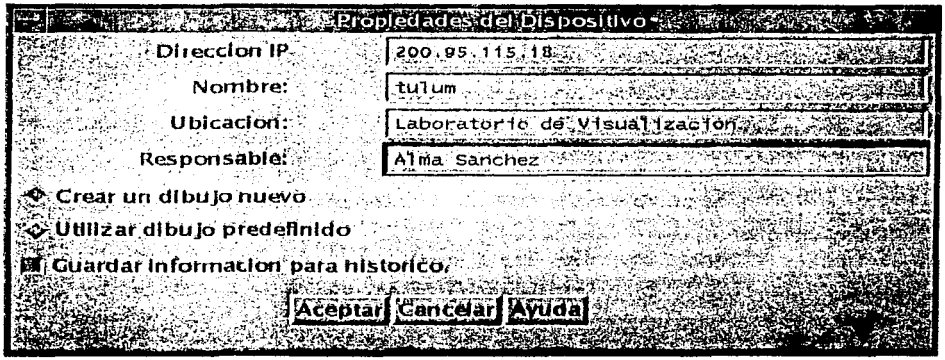
Definir, Modificar o Borrar un dispositivo

Una vez definido el archivo de mapa de trabajo, se podrán definir los dispositivos que conformarán a éste.

Para la definición de un nuevo dispositivo en el mapa se deberá seleccionar la opción Nuevo del menú Dispositivo.



Para especificar los datos de cada dispositivo aparecerá la pantalla siguente:



Donde se deberán teclear cada uno de los datos requeridos:

Dirección IP: La captura de este campo deberá realizarse de manera obligatoria y deberá seguir el siguiente formato: n.n.n.n (0<n<255), siempre es requerido.

Nombre: Es el alias del dispositivo relacionado en la tabla de host, es un dato opcional.

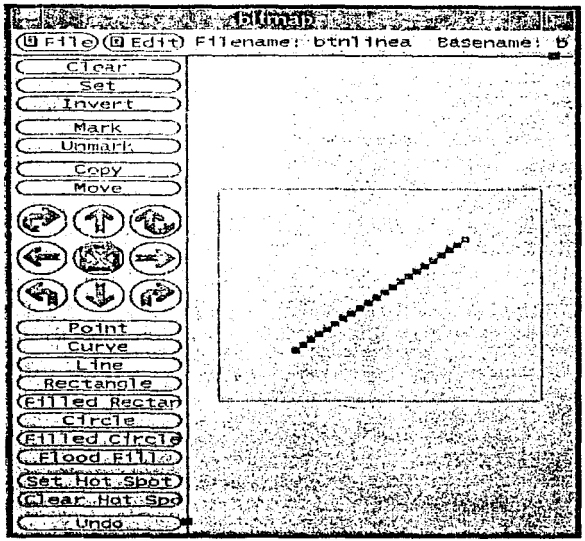
Ubicación: Ubicación geográfica del dispositivo, no es obligatorio.

Responsable: Nombre de la persona responsable de los servicios de red del dispositivo, es opcional.

Crear un dibujo nuevo o Utilizar dibujo predefinido: Deberá seleccionarse una y sólo una de éstas opciones, dado que es la representación gráfica que tendrá el dispositivo dentro del mapa.

Si la opción seleccionada es la de utilizar el dibujo predefinido la representación del dispositivo en el mapa será:

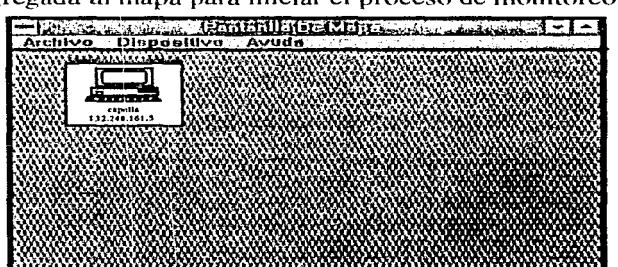
Si en cambio la opción seleccionada fue Crear un dibujo nuevo, entonces se abrirá la aplicación de creación de un bitmap con la cual se podrá dibujar la representación deseada.



ESTA TESIS NO DEBE
SALIR DE LA BIBLIOTECA

Guardar información para histórico: si esta opción es seleccionada se generará un archivo de bitácora para el almacenamiento de los diferentes estados presentados durante el monitoreo.

Una vez aceptada la información anterior, la representación gráfica del dispositivo es agregada al mapa para iniciar el proceso de monitoreo.



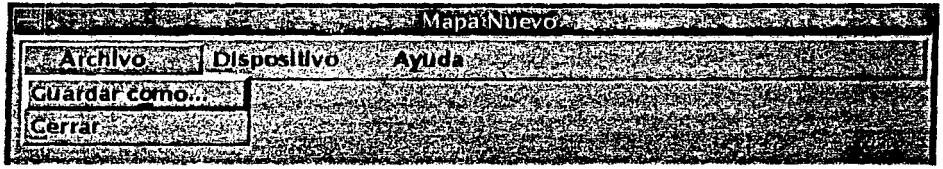
Para llevar a cabo la modificación de datos de un dispositivo se deberá especificar la opción Modificar del menú Dispositivo, que mostrará una ventana de especificación de la dirección IP del dispositivo a modificar.

Una vez aceptada e indicada la dirección IP del dispositivo se abrirá la ventana de propiedades con la información almacenada, en donde podrá modificarse cualquiera de los datos ahí definidos.

Así mismo para borrar un dispositivo deberá seleccionarse la opción Eliminar del menú Dispositivo, acción que desplegará una ventana solicitando la dirección IP del dispositivo a ser borrado.

Un ejemplo de ventana de especificación de dirección IP de dispositivo es:

Además en esta ventana cuenta con la facilidad de cambiar el nombre del archivo de mapa utilizando la opción Guardar como del menú Archivo.

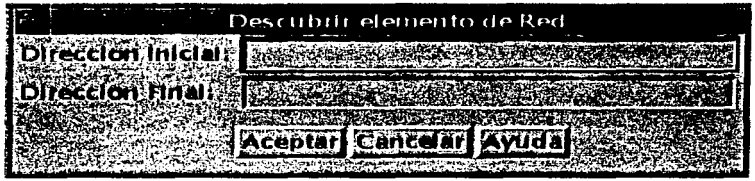


Para cerrar la ventana de mapa deberá accesar la función Cerrar del menú Archivo.

Herramientas de Monitoreo

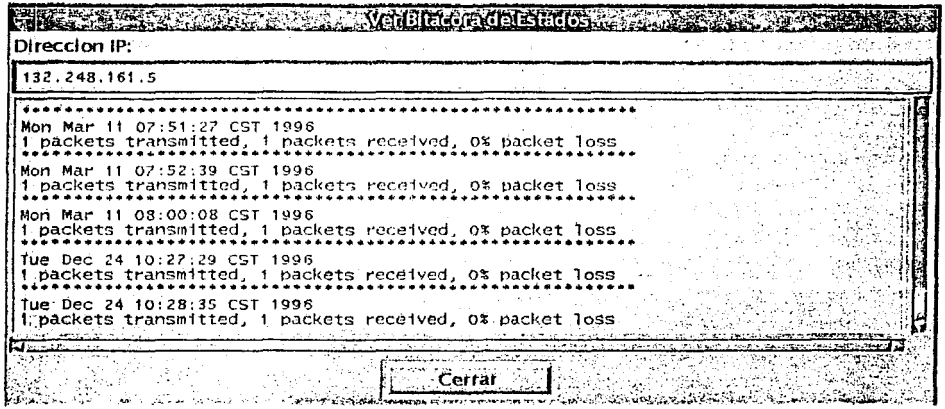
El sistema cuenta con tres herramientas que apoyan las funciones básicas de monitoreo. Estas funciones están disponibles en la pantalla principal del sistema, en el menú de Herramientas y son:

Descubrimiento Automático: Realiza el descubrimiento de los dispositivos activos dentro de un rango, al seleccionar esta opción se abrirá la siguiente ventana:

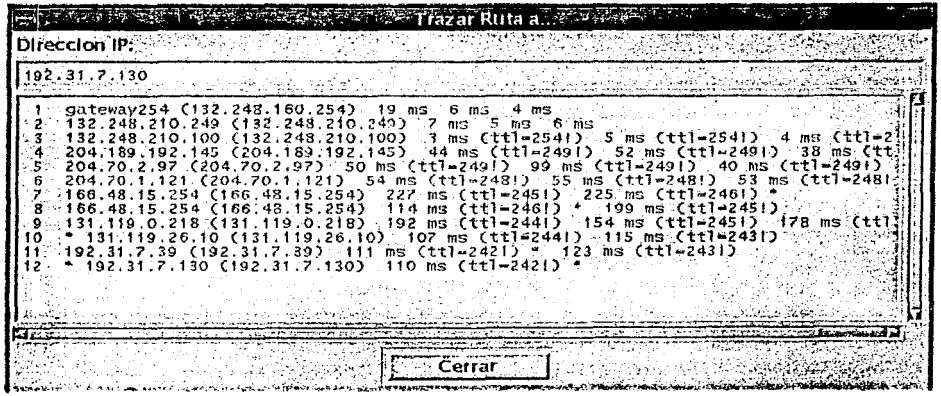


en donde se deberá especificar la dirección inicio y fin del rango de búsqueda. Una vez aceptada esta opción, se generará de manera automática un archivo de mapa conteniendo únicamente la dirección IP de los dispositivos y cuya representación gráfica será el dibujo predeterminado, mostrando en la pantalla de mapa todos los dispositivos activos.

Ver Bitácora de Estados: Esta herramienta despliega el archivo histórico de los diferentes estados que ha presentado el dispositivo desde que se inicio el procesos de monitoreo, (solamente para aquellos dispositivos a los que se haya especificado la opción de Guardar información para histórico, en el momento de la definición). Un ejemplo de despliegue de esta pantalla es:



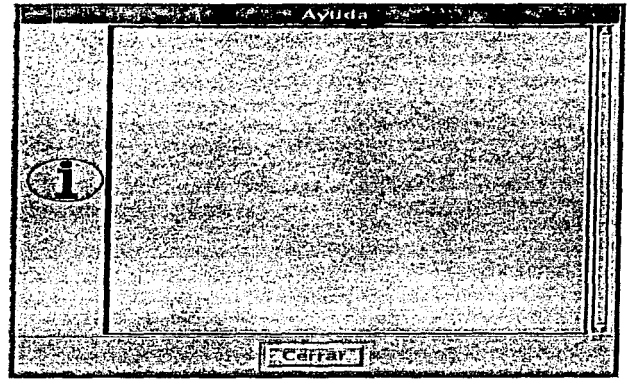
Trazar Ruta: Con esta herramienta es posible conocer cada uno de los saltos que realiza la trama de datos para llegar a un dispositivo específico. Un ejemplo de despliegue de esta pantalla es:



Ayuda en el Sistema

El sistema cuenta en varias pantallas con la opción de Ayuda, por ejemplo en la pantalla principal se localiza directamente en el menú de Ayuda.

Al seleccionar esta función se desplegará una ventana de despliegue de texto de ayuda como la siguiente:



BIBLIOGRAFÍA

Administración de UNIX
Manual de capacitación DGSCA
1991

C Manual de Referencia
Hebert Schildt
Osborne/McGraw-Hill
1990

SNMP,SNMPv2 and CMIP
The Practical Guide to Network Management Standards
William Stallings
Addison
Junio 1993

RFC 11552 substituye RFC 1065
Structure and Identification of Management Information for TCP/IP Based Internet
M. Rose & McCloghrie
Performance Systems International
K.McCloghrie
Hughes LAN Systems
May 1990

The Definitive Guides to the X Window System Vol. 6
Motif Programming Manual
OSF/Motif version 1.1.
Dan Heller
O'Reilly & Associates, Inc.
Julio 1992

The Definitive Guides to the X Window System Vol. 4
X Tool kit Intrinsic Programming Manual
OSF/Motif Edition for X11, Release 4
Adrian Nye and Tim O'Reilly
O'Reilly & Associates, Inc.
1990

RFC 1470 FYI on a Network Management tool catalog: tools for monitoring and debugging TCP/IP Internet and Interconnected Devices

RFC 1213 Management Information Base for Network Management TCP/IP Based Internet: MIB II

RFC 1212 Concise MIB Definition

RFC 1180 A TCP/IP Tutorial

RFC 1155 Structure and Identification of Management Information for TCP/IP Based Internet

RFC 793 Transmission Control Protocol

RFC 791 Internet Protocol

RFC 792 Internet Control Message Protocol

RFC 1098 Simple Network Management Protocol

RFC 1442 Structure of Management Information for version 2 of the Simple Network Management Protocol SNMPv2

RFC 1214 OSI Internet Management: Management Information Base

RFC 1189 The Common Management Information Service and Protocols for the Internet

RFC 1095 Common Management Information Service and Protocol over TCP/IP CMOT

UNIX Review
The Magazine for systems and solutions developers
Interface Design
September 1995 vol 13 no.10