39
2ej.

UNIVERSIDAD NACIONAL AUTONOMA DE MEXICO

FACULTAD DE INGENIERIA

SISTEMA DE ADMINISTRACION PARA REDES TCP/IP UTILIZANDO SNMP

T E S I S

Que para obtener el titulo de

INGENIERO MECANICO ELECTRICO
(ELECTRICO - ELECTRONICO)

p r e s e n t a

FLAVIO CIENFUEGOS VALENCIA

Director de Tesis: Ing. Ricardo Martinezgarza Fernández

México, D. F. **Marzo 1997**

A mi Papá y Mamá,

Fidel Cienfuegos A. y María Guadalupe Valencia F.

por brindarnos la oportunidad de prepararnos académicamente, por darnos libertad de decisión, y por ese gran cariño y apoyo incondicionales que nos dan.

A mi hermano y hermanas,

Lizzette, Velvet, Berenice, Lupita y Fidel

por mostrarme su apoyo y respeto en todas las acciones y decisiones que he tomado.

A quienes han sido mis maestros,

tanto en mi vida académica como en lo personal.

A quienes me han ofrecido su amistad y cariño,

a prueba de fuego y en todo momento, ya que gracias a ello he aprendido a no olvidar los valores que heredé de mi familia y he adquirido nuevas perspectivas.

A la Universidad Pública y Gratuita,

Universidad Nacional Autónoma de México

por ofrecer un espacio de educación integral a los estudiantes de cualquier estrato social, además de darnos la oportunidad de estar en contacto con las diversas manifestaciones del pensamiento humano.

A todos ellos dedico este pequeño esfuerzo en mi vida.

Flavio Cienfuegos Valencia

Quisiera primeramente agradecer a Dios por permitirme alcanzar una de las metas mas importantes en mi vida. Y dedicar este trabajo de Tesis a mi Mamá Linda y mi Abuelita Jovita que estarían orgullosas de mi.

Quisiera Agradecer a mi Papá José por el apoyo brindado y en forma especial a mi abuelo Pedrito por darme todo su cariño, apoyo y agradecer todos sus consejos.

A todos mis hermanos, Guadalupe, José Luis, Valentín, Felipe, Diana y Eduardo, por toda su ayuda, apoyo y comprensión que me brindan.

A mis tíos, Emma, Irene, Beto, y Estela por su apoyo y consejos, que me ayudaron a superar las dificultades para seguir adelante.

A todas las personas que me brindaron su amistad y apoyo incondicional para poder realizar este trabajo y en especial a Irma Plata Colín por estar a mi lado en todo momento.

A la Universidad Nacional Autónoma de México, la Facultad de Ingeniería y en especial a todos mis Maestros con todo mi agradecimiento y respeto, por darme la oportunidad de realizar mis estudios profesionales y poder disfrutar este momento.

Alonso Alcaraz Contreras

Agradecemos profundamente :

Al Ingeniero Ricardo MartínezGarza Fernández por su apreciable dirección, asesoría y por darnos la oportunidad de desarrollarnos profesionalmente en el área de Telecomunicaciones.

Al Ingeniero José Luis Legorreta García por todas las facilidades y apoyo que nos brindó para realizar este proyecto.

Al los compañeros del Centro de Operación de RedUNAM por su apoyo en la realización de las pruebas para nuestro sistema.

A todo el Personal de la DTD por darnos su amistad y apoyo en todo momento.

Alonso Alcaraz Contreras

Flavio Cienfuegos Valencia

CONTENIDO Página

Página

Página

Página

INTRODUCCION

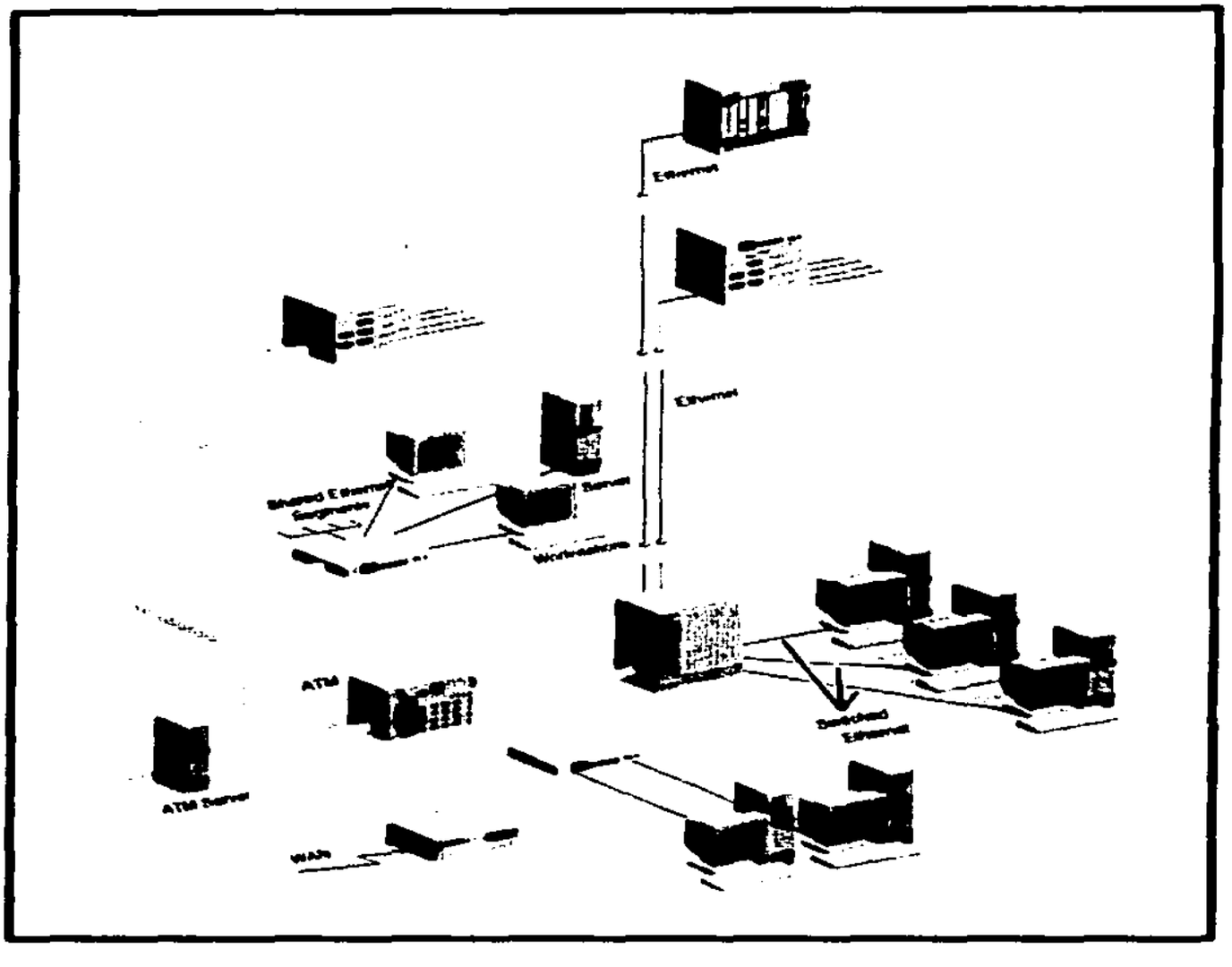
La adopción de las computadoras personales, estaciones de trabajo, y servidores de información, revolucionó por completo la estructura de las redes de comunicaciones de datos. Donde alguna vez hubo terminales *tontas* y minicomputadoras o *hosts* inteligentes, ahora existen grupos de sistemas inteligentes que se transmiten datos entre sí (p.ej. el esquema Cliente/Servidor o el cómputo distribuido). El mercado de las comunicaciones respondió con una diversidad de nuevos dispositivos - puentes locales y remotos, ruteadores multiprotocolo, concentradores distribuidos y concentradores conmutados. Los requerimientos de mayor ancho de banda en la interconexión de redes de área local, trajo como consecuencia la implantación de equipo de alto rendimiento, tales como las unidades DSU/CSU (*Data Service Unit/Channel Service Unit*, respectivamente) o las interfaces *Frame-Relay*. La figura de la siguiente página muestra la estructura de una red moderna.

Los usuarios empezaron a adquirir sistemas y equipos con diferentes proveedores. Cuando los clientes le demandaron a sus proveedores que les suministraran los medios para configurar, monitorear, y probar el equipo de red, cada uno de estos produjo un producto *propietario* que sólo era útil para la administración del equipo desarrollado por ellos. Entonces, cada vez que un nuevo equipo era introducido al ambiente, se tenía que desarrollar una herramienta para su control y esto provocó una nube de productos en los Centros de Operación de las Redes. Cada herramienta medía y contabilizaba de acuerdo a diferentes reglas.

Dos razones principales son las que hacen necesaria la existencia de un *Sistema de Administración* de redes de comunicaciones de datos. Una, es la necesidad de mantener las redes trabajando en forma adecuada, es decir, con gran confiabilidad y disponibilidad, y la otra de obtener información de tráfico y utilización con el fin de planear, diseñar, e implantar el crecimiento de las redes.

En la actualidad, existe un protocolo ampliamente implantado para la administración de redes : SNMP (*Simple Network Management Protocol*). Se afirma en el mercado que este protocolo ha demostrado su factibilidad para administrar dispositivos que van desde repetidores de señal hasta supercomputadoras. SNMP es uno de los estándares internacionales de

administración de red que existen en la actualidad, y fué desarrollado por parte de la Comunidad Internet.



ESTRUCTURA DE UNA RED DE COMUNICACIONES DE DATOS

Debido a la filosofía en la que se fundamenta SNMP, es posible implantarlo en una amplia gama de plataformas o ambientes de programación, así como también es factible desarrollar el Sistema de Administración de una manera modular y extensible, es decir, podemos estructurarlo en varias etapas

de desarrollo, de tal manera que en cada una de éstas se vayan agregando nuevas características y funciones. Es así como, después de un análisis de las posibilidades del sistema, se eligió llevarlo a cabo en cuatro etapas de implantación.

En la primera etapa, se pretende establecer los fundamentos de operación del protocolo SNMP, además de realizar un programa básico de aplicación con el cual se lleve a cabo la lectura de variables de operación de algunos nodos administrados.

En la segunda etapa, se describirán y analizarán las características de la versión 2 y más reciente de SNMP, además de implantar RMON (*Remote Monitoring*), el estándar para monitoreo y control de LAN (*Local Area Network*) remotas. Muy ligado a esto se encuentra la puesta en operación de MIBs (*Management Information Base*) desarrolladas por fabricantes de equipo para interconexión de redes.

Siguiendo con el proceso, llegamos a la tercera etapa, en la cual se añadirá al sistema de administración, la función de monitoreo "Fuera de Banda" (*Out of Band*). Además, se analizarán las posibilidades de la realización de un *software* de agente sobre plataforma DOS.

Como parte final del proyecto global, se ha pensado en aprovechar las nuevas versiones, y por lo tanto las funciones mejoradas, de las especificaciones de las APIs (*Application Programming Interfaces*) utilizadas, las cuales son conocidas como WINSOCK y WinSNMP.

El contenido de este trabajo, conforma solamente la primera etapa de implantación ya que su objetivo principal es determinar si SNMP es un protocolo útil y práctico para llevar a cabo la administración de redes de datos. Es decir, se tratará de demostrar si es factible la implantación de SNMP como parte central de un Sistema de Administración y también si puede llevarse a cabo con la relativa sencillez que afirman quienes lo promueven. Un objetivo complementario es la descripción de la arquitectura y operación de SNMP.

CAPITULO

1

ANTECEDENTES

1.1 Introducción.

El tema central de este proyecto de tesis está inmerso en el ámbito de la Tecnología de Administración de Redes, es por esto que es necesario describir algunos aspectos de esta tecnología que ha cobrado gran importancia en años recientes.

Existen diversos puntos de vista sobre la definición de Administración de Redes (*Network Management*). Aplicando la definición de administración de negocios en el área de las redes de comunicaciones de datos, podemos ver que la administración de redes involucra lo siguiente : Planeación, Organización, Monitoreo, Contabilidad, y el Control de actividad y recursos. Sin embargo, las estructuras de administración de redes de OSI (*Open System Interconnection*) e Internet, se enfocan primordialmente en el monitoreo, contabilidad, y el control de actividad y recursos. Los otros dos aspectos, planeación y organización, no están contemplados en los esquemas citados.

La planeación y la organización en las redes de comunicaciones de datos son los puntos medulares en la administración de redes, ya que consumen la mayoría de los recursos humanos y económicos de las empresas. Es por esto que si las redes no cumplen con una planeación y una organización, no serviría de nada la información obtenida con el monitoreo, la contabilidad y el control de actividad y recursos.

Existen dos esquemas principales para la administración de redes, el definido dentro del modelo OSI y el creado por la Comunidad Internet. La administración de redes según OSI es más robusta que el esquema planteado por Internet, ya que subdivide el sistema en cinco áreas funcionales de administración : Configuración, Contabilidad, Seguridad, Rendimiento y Manejo de Fallas. La desventaja de este último esquema es que su implantación es más compleja y laboriosa.

Este proyecto utilizará utilizar el esquema de administración de redes creado y utilizado por la Comunidad Internet.

1.2 Modelo y conceptos de administración de red.

Un sistema de administración de red contiene tres componentes:

a) varios *nodos administrados*, cada uno de los cuales contiene un *agente*;
b) una *estación de administración de red* (*Network Management Station*);
c) y, un *protocolo* de administración de red, el cual es utilizado por la estación y los agentes para intercambiar información de administración.

1.2.1 Nodos Administrados.

Un nodo administrado se refiere a cualquier dispositivo activo de una red. El común denominador entre estos dispositivos es que todos tienen alguna forma de conexión a red. Algunos implantan el grupo de protocolos de Internet, mientras que la función principal de otros es dependiente del medio de transmisión. Como podemos ver, la diversidad potencial de los nodos administrados puede ser muy alta, cubriendo el espectro desde *mainframes* hasta *modems*.

El axioma fundamental.

Un sistema de administración de red que pretenda ser exitoso tiene que tomar en cuenta las diferencias tecnológicas y funcionales de la gran diversidad de dispositivos que existen en una red, y de esta manera proporcionar un marco de referencia apropiado. El éxito de IP (Protocolo de Internet) es grande debido a los mínimos requerimientos que impone bajo la capa de interfaz de red. Una filosofía similar fué adoptada en el ***Marco de Referencia estándar de Administración de Red de Internet***:

El impacto de agregar la capacidad de administración a los nodos de la red debe ser mínimo, con el fin de formar un común denomindor entre los nodos administrados.

El axioma fundamental es obligado por las amplias diferencias que pueden existir entre los nodos que se administran en una red.

Características comunes de los nodos administrados.

Básicamente, cualquier nodo puede ser conceptualizado como si estuviera formado por tres componentes:

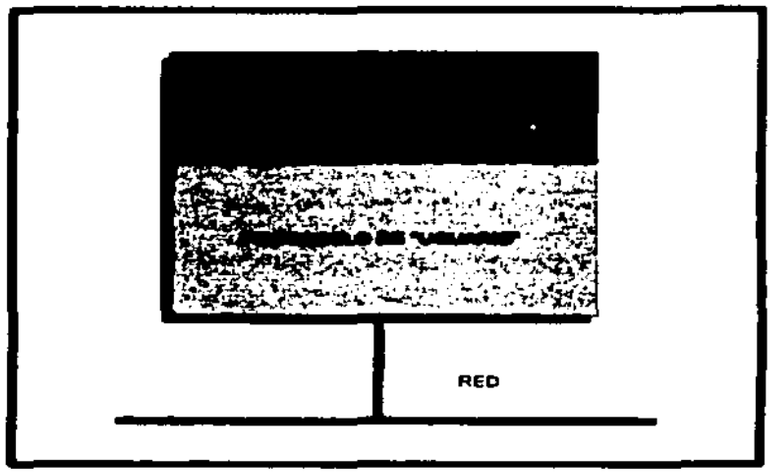


Figura 1.1 Componentes de un nodo administrado

Estos son:

a) *protocolos de usuario*, los cuales llevan a cabo las funciones deseadas por el usuario;
b) un *protocolo de administración*, el cual permite el monitoreo y control del nodo administrado;
c) y la *instrumentación*, la cual interactúa con la implantación del nodo administrado para poder lograr el monitoreo y el control.

La interacción entre estos componentes es directa: la instrumentación actúa como un enlace entre los protocolos de usuario y el protocolo de administración. Usualmente, esto se logra mediante un mecanismo de comunicaciones interno en el cual las estructuras de datos para los protocolos de usuario pueden ser accesadas y manipuladas en el momento de una *consulta* del protocolo de administración.

En la actualidad, este punto de vista es un poco simplista: los intercambios de la información de administración, *per se*, son insuficientes para lograr funcionalidad en la administración del nodo. El protocolo tiene que proporcionar un *marco de referencia administrativo*, el cual implante políticas de autentificación y autorización.

1.2.2 Estaciones de Administración de Red (NMS).

Una Estación de Administración de Red se refiere al sistema *host* que esta ejecutando:

- El protocolo de administración
- La aplicación de administración de redes.

Si tomamos en cuenta que el protocolo de administración es el que proporciona el *mecanismo de administración*, entonces son las aplicaciones las que determinan la políticas usadas.

Anteriormente notamos que el Axioma Fundamental indicaba que la "adición de administración a la red" debe tener un mínimo impacto en los nodos administrados. Como consecuencia, la carga del sistema es desplazada a la estación central de administración. Es posible visualizar que existen muchos más nodos que estaciones de administración en una interred, y es por esto que la escalabilidad favorece este enfoque: es mejor requerir una funcionalidad significativa de un pequeño porcentaje de dispositivos, que requerirlo de la gran mayoría.

1.2.3 Protocolo de Administración de Red.

Dependiendo del modelo usado para la administración de red, un protocolo de administración puede tomar varias formas. Por ejemplo, en un modelo de ejecución remota, el protocolo es usado para intercambiar "fragmentos de programa" que son ejecutados en el nodo.

En el Marco de referencia estándar de Administración de Red de Internet, se utiliza un modelo de "depuración remota". Cada nodo es visto como si tuviera varias *variables*. El nodo es monitoreado y controlado, por medio de la lectura y el cambio de estas variables, respectivamente. La ventaja del uso de este planteamiento es que es relativamente sencillo construir un protocolo para lograr estos objetivos.

Además de la lectura y escritura de variables, existen otras dos operaciones que son requeridas:

- una *operación de visualización*, la cual permite que una estación de administración determine que variables soporta un nodo, además de poder manipular tanto las que son escalares (p.ej. colisiones) como las no-escalares (p.ej. tablas de ruteo); y

- una *operación trap*, la cual le permite a un nodo administrado reportar un evento extraordinario a la estación central.

1.2.4 Administración *Proxy*.

Hasta este momento sólo se ha considerado que los nodos administrados tienen la capacidad de implantar parte o todo el grupo de protocolos TCP/IP. Pero, cuando existen dispositivos que no tienen la capacidad para hacer esto, como pueden ser los repetidores y puentes, surge la necesidad de utilizar un procedimiento que se conoce como *agente proxy*. Los dispositivos antes mencionados se conocen como *extranjeros*.

Para administrar los dispositivos extranjeros es necesario que la estación de administración se comunique con ellos a través del agente *proxy*. Este agente traduce las consultas que recibe de la estación central en instrucciones apropiadas para el nodo administrado.

1.3 Representación de los datos.

La implantación de cada uno de los protocolos de Internet, de las capas más bajas (interfaz, IP, y TCP- *Transmission Control Protocol*), tiene una *representación interna*, la cual denota cada una de las estructuras utilizadas por estos. Estas estructuras de datos dependen del lenguaje de programación, del compilador de éste, y la arquitectura de máquina de cada plataforma. Debido a que los paquetes de información intercambiados a estos niveles son relativamente simples, es posible utilizar un esquema de ordenamiento por bytes. En la capa de aplicación del modelo, las estructuras de datos intercambiadas son potencialmente más complejas. Por consiguiente, es necesario introducir un nuevo formalismo que describa estas estructuras.

Este nuevo formalismo es conocido como *sintaxis abstracta*, la cual se usa para definir los datos sin tomar en cuenta las restricciones y estructuras orientadas a máquinas. En el Marco de Referencia de Internet, es utilizado un lenguaje OSI para llevar a cabo este propósito, y se conoce como Notación de Sintaxis Abstracta Uno (ASN.1).

Esto es, la sintaxis abstracta es usada para describir las estructuras de datos intercambiadas en el nivel del protocolo, y para representar la información de administración que es transportada por medio de dichas estructuras de datos.

Una vez que las estructuras pueden ser descritas en una forma tal que sean independientes de las máquinas, tiene que existir una manera en la cual

transmitir estas estructuras a través de las redes. Esta tarea se realiza mediante la utilización de una *Sintaxis de Transferencia*. Esta se implementa utilizando lo que se conoce como Reglas de Codificación Básicas (BERs).

CAPITULO

2

SMI
ESTRUCTURA DE LA INFORMACION DE ADMINISTRACION

2.1 Introducción.

La estrategia elegida por la IAB (*Internet Activities Board*), a finales de los ochentas, estuvo fundamentada en la implantación de los protocolos que desarrollarían dos grupos de trabajo. El primero, trataría un proyecto a corto plazo, en el cual el protocolo SGMP (*Simple Gateway Monitoring Protocol*) sería modificado de tal manera que se aprovechara la experiencia obtenida de su aplicación en redes en operación; esto daría como resultado la creación de un nuevo protocolo: SNMP. El segundo grupo investigaría, a largo plazo, la funcionalidad de un protocolo reciente, CMIP (*Common Management Information Protocol*) del grupo de protocolos OSI.

Este doble enfoque trae consigo la desventaja de que es necesaria una etapa de transición del corto al largo plazo. Para lograr una migración ordenada entre las tecnologías, fué obligatorio definir un marco de referencia al cual se adaptaran ambos protocolos. En éste se tendrían que establecer una serie de reglas que definieran los objetos que serían administrados y éstas fueron construidas de tal forma que fueran totalmente independientes de los protocolos de administración utilizados para transportar la información.

Los diseñadores de SNMP necesitaron encontrar una forma en la cual organizar:

- **Una Estructura Administrativa.** La única forma de resolver cómo un componente de la red será administrado es delegar el trabajo dentro de cada especialidad en particular a expertos en el campo.

- **Una Estructura de Información.** Hoy en día, para administrar las redes multi-proveedor es necearia una gran cantidad de información. Es por esto, que se requiere de una estructura para la información de administración de las redes que podamos extender conforme se descubran nuevos requerimientos.

- **Una Estructura de Nombres.** Existirán miles de variables que se definirán para la administración de redes. Es por esto que es conveniente utilizar un método consistente de definición, descripción, y asignación de nombres de variables.

2.2 Objetos administrados.

En la terminología de la administración de redes se habla de *objetos*, más que de variables. Una *instancia* es un elemento particular de un objeto. Un objeto consta de lo siguiente:

- Un nombre, el cual se conoce como *Identificador de Objeto* (OID).
- Atributos:
  - Un tipo de datos
  - Una descripción detallada del objeto
  - Información de status: si es definición actual u obsoleta
- Operaciones válidas sobre él: Lectura y/o Escritura

Ejemplos de objetos administrados son la descripción de un sistema, tiempo de puesta en operación, la dirección IP interfaces de un ruteador, etc. Por ejemplo, un objeto podría ser el status de las interfaces de un ruteador, y una de sus instancias sería el status de la tercer interfaz.

En este trabajo usaremos indistintamente los términos instancia y variable de un nodo administrado.

2.2.1 Ejemplo de definición de un objeto.

La definición formal del objeto *sysDescr*, como se especifica en el RFC (*Request For Comment*) 1213, es la siguiente:

```
sysDescr OBJECT-TYPE
            SYNTAX DisplayString (SIZE (0..255))
            ACCESS read-only
```

STATUS mandatory
DESCRIPTION
"Una descripción textual de la entidad. Este valor debe incluir el nombre completo y la versión del tipo de hardware del sistema, sistema operativo de software, y de software de red. Es obligatorio que sólo contenga caracteres ASCII imprimibles."
::= { system 1 }

2.3 Estructura de información ISO-CCITT.

La ISO (*International Standarization Organization*) y el CCITT (*Comité Consultivo Internacional de Telegrafía y Telefonía*) promovieron la idea de estructurar la información en un árbol global de nombres y asignar un identificador a cualquier objeto que necesitara un nombre. Este árbol se muestra en la figura 2.1.

El subárbol Internet bajo el nodo *dod* (*Department of Defense*, de los EUA) es propiedad de la IAB y es administrado por la IANA (*Internet Assigned Numbers Association*). Las estructuras administrativa, de información y de nombres están integradas en este esquema global.

2.4 Identificadores de Objetos e Instancias.

Los objetos que requerimos administrar son sub-nodos del árbol ISO-CCITT. A cada nodo se le asigna una etiqueta que consiste de un entero y una breve descripción textual. El Identificador de Objeto es una serie de enteros que marcan la trayectoria que va desde la raíz del árbol hacia el objeto. Por ejemplo, el identificador del objeto *sysDescr* (descripción del sistema) es : 1.3.6.1.2.1.1.1, o en forma textual iso_org_dod_internet_mgmt_mib-2_system_sysDescr.

Un OID indica la clase de objeto que queremos conocer. Un *identificador de instancia o nombre de variable* se utiliza para obtener el valor preciso que necesitamos. Por ejemplo, en el caso del objeto *sysDescr*, debido a que no tiene más que sólo un elemento (la descripción del sistema), el identificador de instancia es el identificador del objeto con un número "0" agregado al final de la serie de enteros : 1.3.6.1.2.1.1.1.0. En el caso de que el objeto tuviera varios elementos, por ejemplo el status de las interfaces de un ruteador, los valores de las variables de cada interfaz serían recuperados agregando números consecutivos a cada interfaz.

2.5 Base de datos de la Información de Administración (MIB).

Es conveniente conceptualizar toda la información de configuración, status y estadísticas de cierto dispositivo, como una base de datos. Esta, puede ser implementada en arreglos de *switches*, variables y tablas en-memoria, o archivos. En los estándares SNMP esta base lógica de datos es conocida como MIB (*Management Information Base*). La MIB estándar de Internet (la primera se denominó MIB-I), como está definida en el RFC 1157, describe aquellos objetos que se espera sean implantados por los nodos que estén ejecutando el grupo de protocolos TCP/IP; esto es con el fin de llevar a cabo la administración de una interred que opere con dichos protocolos.

En realidad, sólo existe una MIB y ésta enmarca todos las subsecuentes *estructuras* que se vayan creando.

Una de las grandes ventajas de SNMP es que no está limitado a manejar un cierto número de objetos, ya que debido a la estructura de desarrollo de la MIB, es posible que se agreguen objetos de administración para un equipo o sistema de red específico. Es así como, entidades de investigación y empresas privadas pueden agregar objetos de administración para sus productos pero dentro del marco establecido por la MIB original.

2.5.1 Grupos de la MIB-II.

Todo nodo administrado debe implantar la MIB-II. Sin embargo, no es necesario que un nodo use objetos que no le competen, por ejemplo un ruteador no necesita objetos de un correo electrónico. Con el fin de dar orden a la base de datos, se crearon 8 grupos de objetos, en los cuales se manejan datos que van de acuerdo al título con el que fué creado cada grupo. Entonces, cada nodo administrado puede sólo utilizar los que vayan de acuerdo a su función. En la tabla 2.1 se muestran los grupos de la MIB-II.

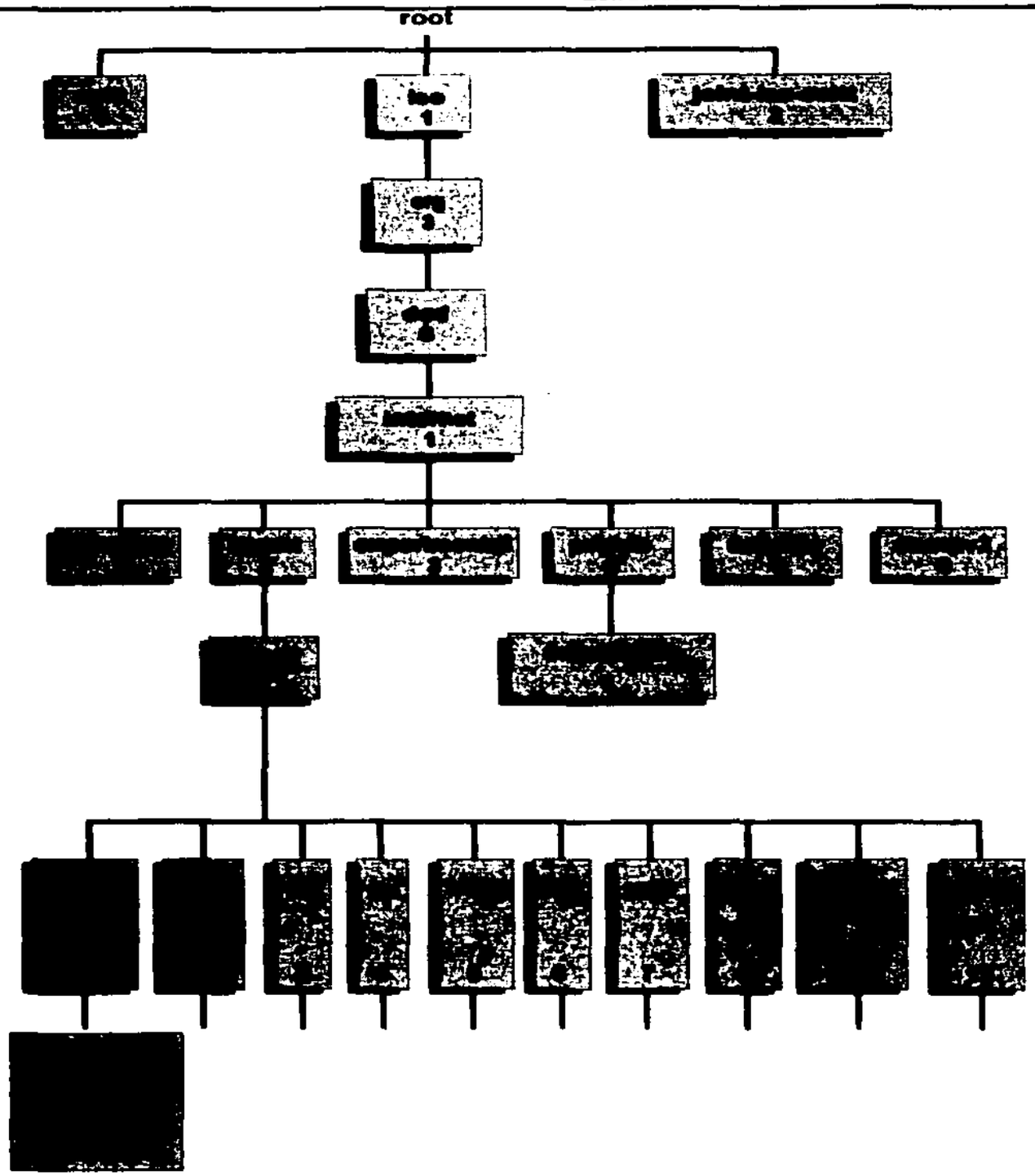


Figura 2.1 Arbol de Identificadores de Objetos.

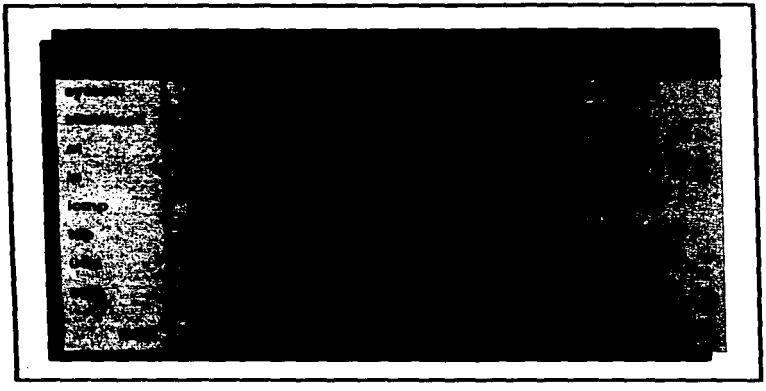


Tabla 2.1 Grupo de Variables de la MIB-II

2.5.2 MIBs propietarias.

El subárbol *private/enterprises* da la posibilidad a los proveedores de *hardware y software*, así como a universidades que aprovechen las facilidades de SNMP. A cada uno de estos, se le asigna un sub-nodo bajo el antes mencionado y de esta manera pueden crear sus propios objetos de administración, pero siempre manteniendo una estructura de desarrollo.

CAPITULO

3

SNMP
ARQUITECTURA Y OPERACIÓN

3.1 Introducción.

El Protocolo Sencillo de Administración de Redes, SNMP, basa su filosofía de operación en la facilidad de su operación, es decir, en la definición precisa de sus funciones de consulta y cambio de variables. Además, ha cobrado gran auge debido a que es relacionado estrechamente con TCP/IP (y con el concepto de los sistemas abiertos), y se considera un protocolo de la capa de aplicación del modelo TCP/IP.

3.2 Objetivos de SNMP.

La arquitectura de SNMP está organizada alrededor de los siguientes conceptos y objetivos:

- Mantener el *software* en el agente tan reducido como sea posible.
- Soportar funciones de administración remota para explotar al máximo los recursos de una interred.
- Desarrollar la arquitectura en forma modular, para tener la facilidad de agregar funciones en el futuro.
- Hacer que SNMP sea independiente de los diversos tipos de *hosts* y *gateways* que existen.

3.3 Relaciones administrativas de SNMP Versión 1.

Un marco de referencia administrativo es el que determina las políticas de *Autentificación y Autorización* utilizadas entre entidades de aplicación de SNMP. Se define como una *comunidad* a la relación que existe entre un agente

y uno o más administradores. Una comunidad es una cadena de octetos no legible, por ejemplo una serie de caracteres ASCII no comunes.

Cuando se intercambian mensajes SNMP, estos constan de dos partes:

1. *un nombre de comunidad*, además de la información que valide que la entidad transmisora es miembro de esa comunidad.

2. *datos*, los cuales contienen una operación SNMP y los operandos asociados.

AUTENTIFICACION. Si el nombre de la comunidad que va insertada en un mensaje, corresponde a una comunidad conocida para la entidad receptora, entonces se considera que el transmisor es autentificado como un miembro de esa comunidad.

AUTORIZACION. Una vez que la entidad transmisora es autentificada, el nodo administrado tiene que determinar que *nivel de acceso* es permitido. Un subconjunto arbitrario de los objetos visibles para una comunidad en particular se conoce como *muestra*. Para cada objeto en la muestra existe un modo de acceso. Haciendo una intersección entre los modos de acceso y la muestra, definidos por la comunidad, tenemos lo que se conoce como perfil de la comunidad para cada objeto en la muestra. Estas dos políticas las vemos ejemplificadas en la figura 3.1

Donde :

Figura 3.1 Políticas de Autentificación y Autorización

3.4 Arquitectura TCP/IP y SNMP.

SNMP fué diseñado por la IETF (*Internet Engineering Task Force*) para su uso en interredes. En la actualidad, está implantado para que se ejecute sobre el Protocolo UDP (*User Datagram Protocol*), como lo muestra la figura 3.2, aunque no existen razones técnicas para que no pueda operar sobre otro protocolo.

SNMP es un protocolo *sin-conexiones*, debido a que utiliza UDP. Como con todos los protocolos de este tipo, la cantidad de información de control agregada (*overhead*) es poca, además de que se obtiene simplicidad. En la figura 3.2 se esquematiza el grupo de protocolos de Internet y la ubicación que tiene SNMP dentro de este modelo. Como podemos ver, SNMP se encuentra dentro de la que se conoce como capa de aplicación.

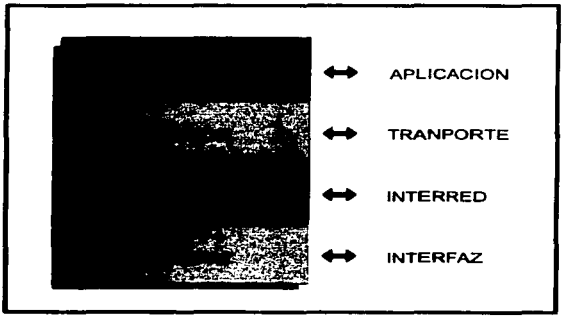


Figura 3.2 Ubicación de SNMP en el Modelo TCP/IP

3.5 Operación de SNMP.

SNMP es un protocolo que utiliza el esquema *Cliente/Servidor* considerando a la Estación de Administración de Red como el Cliente y al Agente ubicado en el nodo administrado como el Servidor. Además, SNMP es un protocolo asíncrono, lo cual significa que no necesita de esperar por una

respuesta después de enviar un mensaje en particular; SNMP consta básicamente de tres componentes : Un administrador, un agente en el nodo administrado, y la propia red de transporte. La figura 3.3 muestra lo que sería el concepto formal de una administración de red utilizando SNMP.

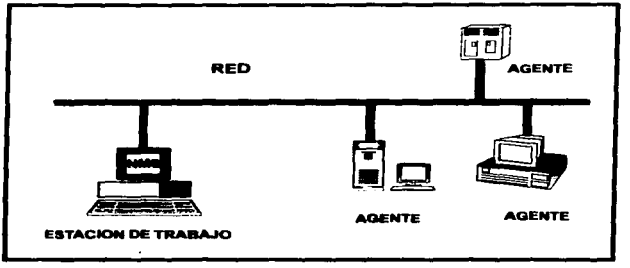


Figura 3.3 Esquema básico de Administración de Red

3.5.1 Unidades de Datos del Protocolo SNMP (PDUs).

Las operaciones en las que se basa el protocolo SNMP están limitadas y básicamente pueden subdividirse en tres áreas. Una, es la de consulta/respuesta de variables en el nodo (GET REQUEST, GET-NEXT REQUEST, GET RESPONSE), otra es la de cambio de valores en las variables (SET), y la última es la de eventos extraordinarios ocurridos en el agente (TRAPs).

- **Get Request**: Este PDU (*Protocol Data Unit*) se utiliza para accesar al agente y obtener valores de una lista. Contiene identificadores para distinguirlo de múltiples consultas, así como valores para proporcionar el status del nodo de red.
- **Get-Next Request**: Es similiar al anterior, con la diferencia que no permite la recuperación del siguiente identificador lógico en una MIB, es decir, hace un *barrido* o *visualización* de ésta.
- **Set Request**: Este PDU es utilizado para escribir una acción que se llevará a cabo en un elemento, además de cambiar valores en una lista de variables.

- **Get Response**: Este PDU responde a los anteriores. Contiene un identificador que lo asocia con el PDU previo. Además proporciona información del status de la respuesta (códigos de error, status de error, y una lista de información adicional).
- **Trap**: Este PDU permite reportar un evento extraordinario a la NMS, proveniente de un agente pudiendo ser por cambios en la configuración o falla del mismo.
-

En la figura 3.4, se muestran los PDUs de SNMP y sus direcciones de operación.

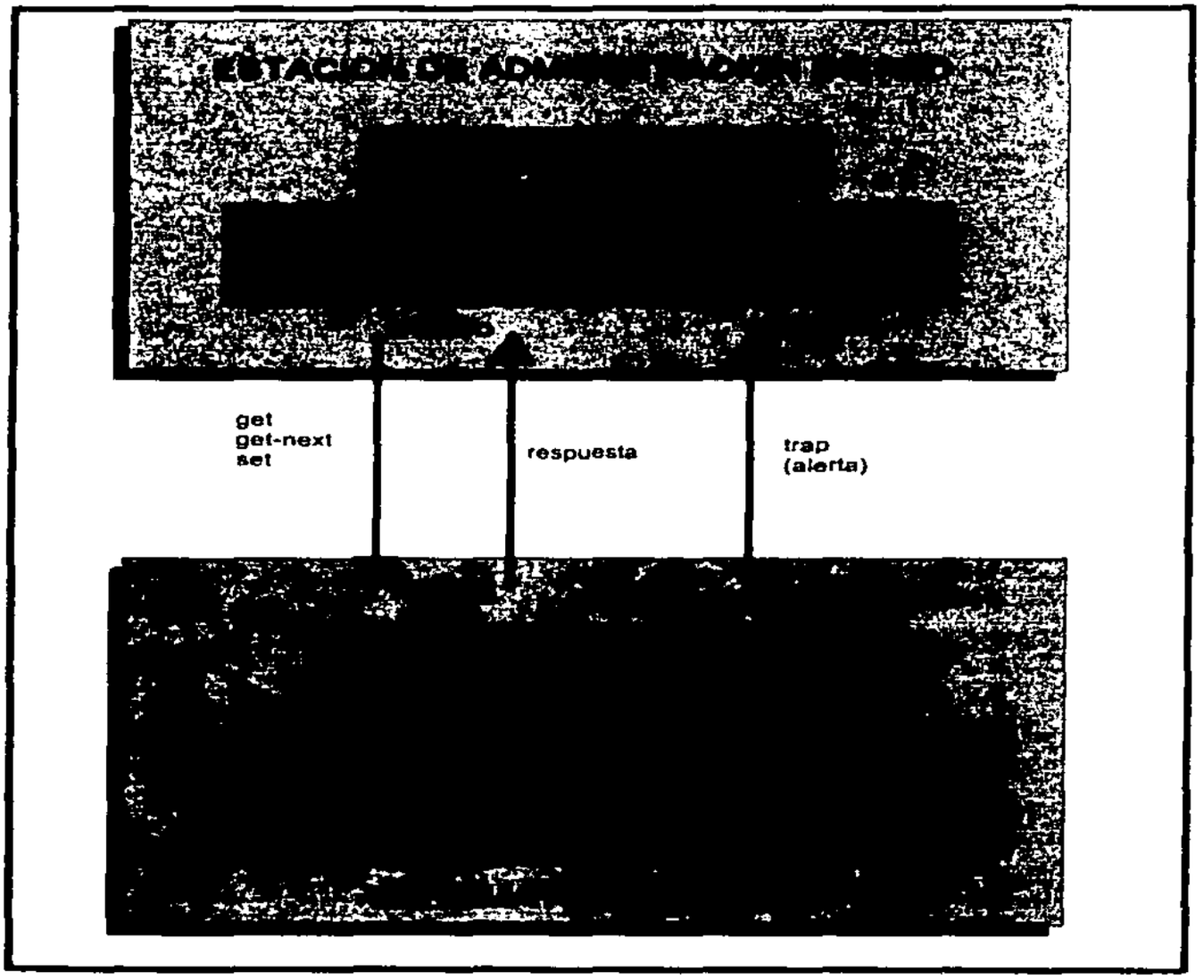


Figura 3.4 PDUs de SNMP

3.6 Mapeos de transporte.

Los *mapeos de transporte* se refieren a las posibles opciones que existen para encapsular los mensajes SNMP en los diversos protocolos de transporte. Al referirnos a protocolos de transporte no sólo nos referimos a los protocolos de las capas que específicamente se conocen como *Capas de Transporte* de algunos modelos, sino a todos los protocolos que intervengan directamente en la transmisión de los datos, como pueden ser : UDP, TCP, Ethernet, Token-Ring, CLNP (*Connection-Less Network Protocol*) de OSI, etc.

SNMP fué concebido con la idea de ser independiente del protocolo de transporte. Todos los *mapeos* tienen un detalle en común. Los mensajes SNMP son enviados a través de la red por medio de un proceso que se llama *serialización*. Esto nos permite que cualquier estructura de datos sea codificada como una secuencia de *bytes* para su envío. Cuando estos se reciben, deben ser re-construidos a la estructura de datos original. Es de entender que existe un mapeo uno-a-uno entre estructuras ASN.1 y una cadena de *bytes*.

3.6.1 Mapeo sobre UDP.

Este es el mapeo comúnmente utilizado, y precisamente es el que se especifica en el RFC 1213. Los mensajes SNMP son serializados y se encapsulan en el paquete UDP, dentro del campo de datos de éste. El formato del paquete UDP se muestra en la figura 3.5:

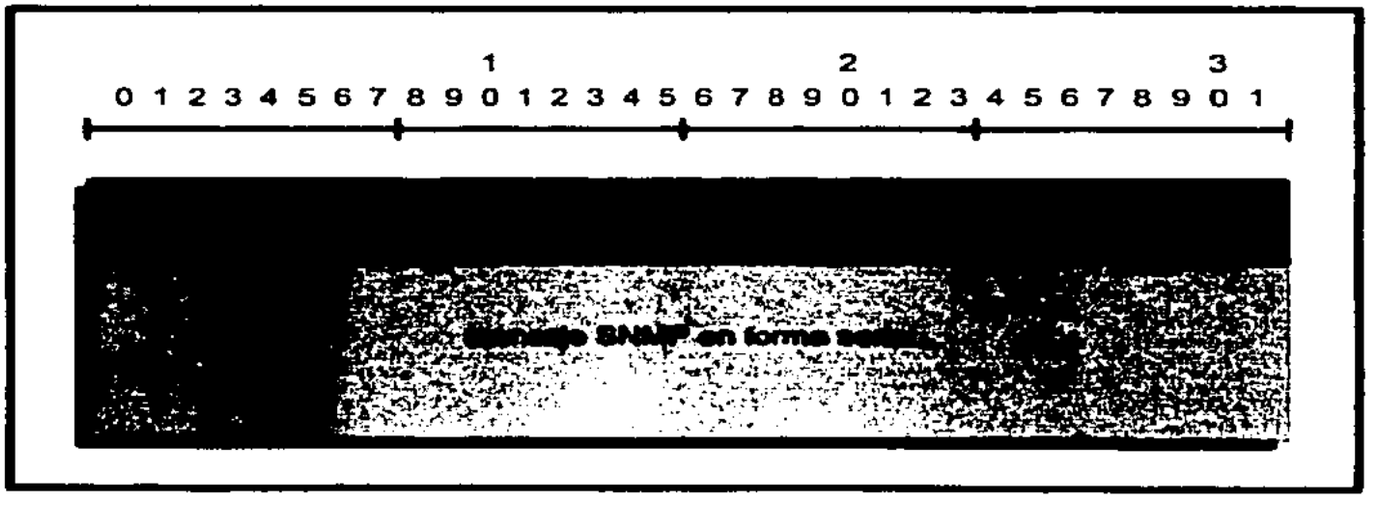


Figura 3.5 Formato del paquete UDP

La dirección de transporte consiste de una dirección IP y un puerto UDP. Todos los agentes SNMP reciben en el puerto 161. Si el mensaje contiene un *trap*, el proceso en el administrador recibe en el puerto 162. Por convención, todas las respuestas son enviadas con los puertos fuente/destino de la consulta, intercambiados.

UDP es el protocolo que se escogió originalmente para transportar mensajes SNMP, ya que es un protocolo que introduce muy poca carga en el medio de transmisión debido a que es un protocolo sin-conexiones, es decir, no establece un circuito virtual de comunicación entre el origen y el destino. Además, al utilizar este UDP estamos asegurando que obtendremos interconectividad entre redes locales ya que opera sobre el protocolo de interred o IP.

3.6.2 Mapeo sobre Ethernet.

Este mapeo es muy poco usado porque no nos brinda los beneficios de la interconectividad, es decir, sólo podemos llevar a cabo la administración dentro de una sola red de área local. Los mensajes SNMP son encapsulados en la trama Ethernet.

Además, con la implantación de mapeos sobre protocolos de red y transporte, es posible realizar el monitoreo remoto de LANs.

CAPITULO

4

WinSNMP
INTERFAZ PARA LA PROGRAMACION DE APLICACIONES

4.1 Introducción.

Esta especificación define una interfaz para aplicaciones de administración de redes que se ejecutan dentro del ambiente Windows en computadora personal, habilitándolas para que hagan uso de una *máquina* SNMP lógicamente externa. Este documento ha sido creado por empresas, investigadores, y desarrolladores de sistemas con la finalidad de incrementar el desarrollo de aplicaciones basadas en SNMP. En este documento se marca la diferencia entre las versiones 1 y 2 de SNMP, pero se define una estructura tal, que exista compatibilidad entre ambas, es decir, cuando se requiera migrar de la primera versión a la más reciente, no será necesario re-estructurar la interfaz. Además, se apega a los documentos oficiales de los protocolos de Internet y en particular a los de SNMP: los *Request For Comment* (RFC).

4.2 Especificación WinSNMP.

Esta especificación define las llamadas de procedimientos, tipos de datos, estructuras de datos, y la semántica asociada con la cual un desarrollador de software puede implantar aplicaciones SNMP.

En la figura 4.2, se muestra la ubicación de WinSNMP en un escenario de conectividad SNMP, en el cual se observan los roles de un administrador (extremo izquierdo) y un agente (extremo derecho). Este diagrama presenta una concepción de alto-nivel del modelo que se enmarca en la versión actual de WinSNMP. Es posible soportar otros modelos por la especificación, conforme se cumpla la característica de independencia del protocolo de transporte.

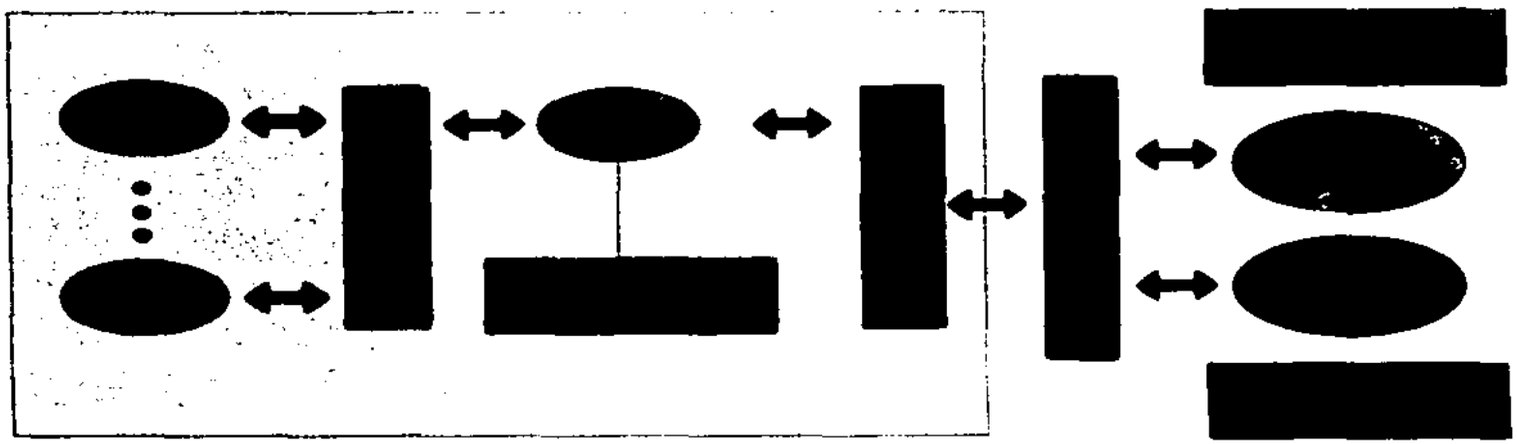


Figura 4.2 Escenario de WinSNMP en un esquema SNMP

En la figura 4.3 se muestra un posible esquema de la posición relativa de la ubicación de WinSNMP y WINSOCK (*Windows Sockets*, nos proporciona una interfaz para el manejo del software TCP/IP) dentro del modelo TCP/IP. La palabra "posible" trata de remarcar que WINSOCK y WinSNMP son interfaces o APIs (*Application Programming Interfaces*), más no son parte del grupo de protocolos de Internet.

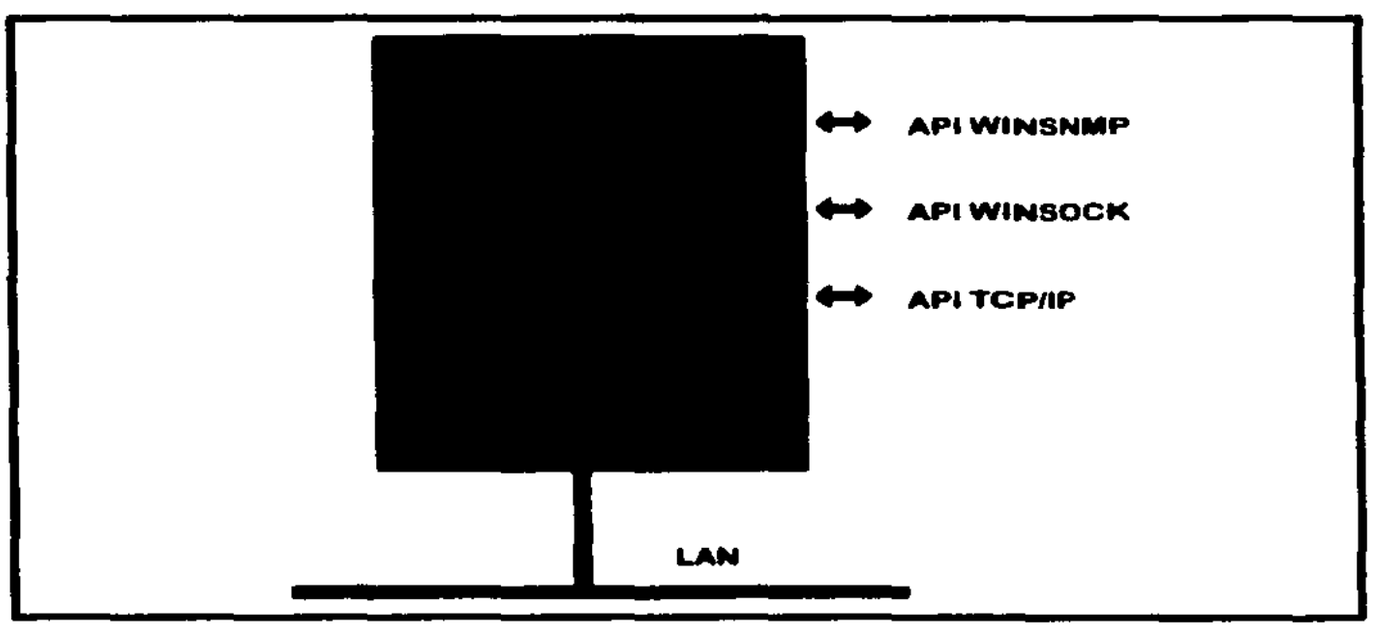


Figura 4.3 Ubicación relativa de WINSOCK y WinSNMP en TCP/IP

WinSNMP ofrece los siguientes beneficios con el fin de acelerar el desarrollo de aplicaciones de administración y monitoreo de redes:

- Tecnología que permite usar SNMP para implantar aplicaciones funcionales de administración de red. WinSNMP nos proporciona la facilidad de poder implantar una aplicación SNMP y sus operaciones derivadas (p.ej. la codificación en lenguaje ASN.1 de las estructuras de los objetos administrados, de las estructuras de los mensajes SNMP, etc., además de facilitar la compilación de dichas estructuras).

- Independencia del proveedor del servicio de SNMP. Una aplicación desarrollada con SNMP, interactúa sin problemas con cualquier otra aplicación compatible con esta especificación.

- Soporte uniforme de SNMPv1 y SNMPv2. Una aplicación WinSNMP no tiene que conocer precisamente la versión de SNMP que soporta el agente. La implantación WinSNMP llevará a cabo todos los mapeos necesarios entre SNMPv1 y SNMPv2 de acuerdo a los RFCs correspondientes.

Todos los productos de *software* que soporten esta especificación se consideran que son "Compatibles con WinSNMP". Existen cuatro niveles de soporte de SNMP en esta especificación, y estos son:

- Nivel 0 = Solo codificación/decodificación de mensajes
- Nivel 1 = Nivel 0 + Interacción con agentes SNMPv1
- Nivel 2 = Nivel 1 + Interacción con agentes SNMPv2
- Nivel 3 = Nivel 2 + Interacción con otros administradores SNMPv2

El Nivel que soporta el sistema que desarrollaremos es el Nivel 1, es decir, podremos interactuar con agentes SNMPv1. Por tal razón, se dice que el sistema tiene una "Interfaz WinSNMP" y el sistema es una "Aplicación WinSNMP".

4.3 Programación con WinSNMP*

El modelo de programación define por omisión que la especificación WinSNMP/Manager API será implantada como un archivo DLL (Biblioteca de Enlace Dinámico, WINSNMP.DLL). Este DLL puede llevar a cabo las funciones SNMP en forma local o a través de un DLL auxiliar que realice consultas SNMP sobre una plataforma remota y mande las respuestas a la aplicación que se ejecuta en la computadora MS-WINDOWS local.

* En la referencia bibliográfica 4 se detallan a profundidad los aspectos relacionados con la programación con WinSNMP.

En cualquier caso, los aspectos principales de la implantación WinSNMP que afectan el desarrollo de la aplicación son los siguientes:

- Niveles de soporte SNMP
- Soporte de la interfaz de transporte
- Modos de traducción Entidad/Contexto
- Información de la Base de Datos Local
- Características de las sesiones
- Manejo de memoria
- Modelo Asíncrono
- Poleo y Retransmisión
- Manejo de Errores
- Tipos de Datos

4.4 Interfaces de WinSNMP.

Las diversas funciones que son definidas en esta especificación pueden agruparse en las siguientes seis categorías (se les conoce como interfaces):

- Funciones de Base de Datos local
- Funciones de comunicaciones
- Funciones de Entidad/Contexto
- Funciones de PDU
- Funciones de variable-valor
- Funciones de utilerías

4.5 Ejemplo de una Sesión WinSNMP.

```
// Se inicializa la estructura WinSnmp
s := SnmpStartup(@se.nMajorVersion, @se.nMinorVersion, @se.nLevel, @se.nTranslateMode,
            @se.nRetransmitMode);
 if s = SNMPAPI_FAILURE then begin
  s := SnmpGetLastError(se.hSnmpSession);
  MessageDlg('Error SnmpStartup : ' + IntToStr(s), mtInformation, [mbOk], 0);
  Result := False;
  Exit;
 end;

// Define el modo de traducción
 if (se.nTranslateMode <> SNMPAPI_UNTRANSLATED_V1) then begin
  s := SnmpSetTranslateMode(SNMPAPI_UNTRANSLATED_V1);
  if s = SNMPAPI_FAILURE then begin
   s := SnmpGetLastError(se.hSnmpSession);
   MessageDlg('Error SnmpSetTranslateMode : ' + IntToStr(s), mtInformation, [mbOk], 0);
   Result := False;
```

```
      Exit;
    end;
   end;
   // Se activa el modo de Retransmisión
    s := SnmpSetRetransmitMode(SNMPAPI_ON);
   se.bWinSnmpStarted := True;
 end;

 // Abre una sesión Snmp
  sed.snmpHandle := Handle;
  sed.snmpMessage := WM_SNMP_MSG;
  if (sed.hSnmpSession = 0) then begin
   sed.hSnmpSession := SnmpOpen(sed.snmpHandle, sed.snmpMessage);
  end;
  if (sed.hSnmpSession = SNMPAPI_FAILURE) then
   Exit;

// Se crea una VBL (Lista de Variable-Valores)
se.hVbl := SnmpCreateVbl(se.hSnmpSession, @se.oid, nil);
 // define la VBL del arreglo TVbls
 for i := 0 to MAX_CONSULTAS_VBLS do begin
  val.syntax := SNMP_SYNTAX_NULL;
  val.empty := 0;
  s := SnmpStrToOid(PChar(TVbls[i]), @oid);
  s := SnmpSetVb(se.hVbl, 0, @oid, @val);
  s := SnmpFreeDescriptor(SNMP_SYNTAX_OID, @oid);
 end;
 se.requestId := requestId;

 // Crea PDU
 se.hPdu := SnmpCreatePdu(se.hSnmpSession, SNMP_PDU_GETNEXT, se.requestId, 0, 0,
se.hVbl);

 // Envía consulta a la red
 s := SnmpSendMsg(se.hSnmpSession, se.hManagerEntity, hAgentEntity, se.hViewContext,
se.hPdu);
 // Libera información interna WinSNMP
 s := SnmpFreeVbl(se.hVbl);
 s := SnmpFreePdu(se.hPdu);
end;
```

CAPITULO

5

SISTEMA DE ADMINISTRACION PARA REDES IP

5.1 Introducción.

El Sistema de Administración para Redes IP está constituido por un programa de aplicación y algunos elementos de *hardware* (tarjetas de red, cableados, PC, etc.). Nuestro sistema se desarrollará con el fin de evaluar si SNMP es un protocolo que nos proporciona elementos útiles para llevar a cabo la Administración de una Red IP/Ethernet, además de determinar si la implantación de SNMP en una cierta plataforma es factible, y su utilización no presenta problemas de rendimiento en la red.

5.2 Análisis del Sistema.

5.2.1 Definición del problema.

El objetivo principal de este proyecto es el desarrollo de una aplicación que lleve a cabo la función de una Estación de Administración de Red (*NMS-Network Management Station*), tal como se especifica en el Modelo de Administración de Redes desarrollado por el área de ingeniería de Internet. Este *software* será el componente central del Sistema de Administración para Redes IP (redes que utilizan TCP/IP), que además de esta aplicación, está compuesto por agentes de administración, por la misma red de comunicaciones, y por un protocolo de administración.

El mecanismo principal que se utilizará para administrar redes IP será SNMP, con el fin de determinar si :

- SNMP es un protocolo que proporciona información real de la actividad y recursos de las redes IP.

- El costo-beneficio resultante de la utilización de SNMP, justifica su implantación.

- La utilización de SNMP en una red, degrada o no el rendimiento de la misma.

5.2.2 Propuesta de solución.

La solución que proponemos para resolver el problema ya definido es una aplicación que denominaremos Sistema de Monitoreo y Diagnóstico (SMD-97). Este sistema define varias características, en las que se tomarán en cuenta los siguientes aspectos :

- Funcionalidad
- Costo
- Rendimiento
- Simplicidad tecnológica
- Ciclo de desarrollo

El SMD-97 se ejecutará en la estación de administración y presentará las siguientes características :

PLATAFORMA.

La plataforma es la base de *hardware* que utilizará la estación de administración, y se propone la siguiente :

Computadora personal con microprocesador Intel 486 o Pentium con 8 Mbytes de memoria RAM como mínimo, 80 Mbytes en disco duro y monitor VGA a color.

SISTEMA OPERATIVO Y AMBIENTE.

Se utilizará el ambiente operativo Windows 95 (sistema a 32 bits)

HERRAMIENTA Y LENGUAJE DE DESARROLLO.

El lenguaje de programación será Object Pascal y el ambiente de desarrollo será Delphi 2.0.

TECNOLOGIA DE RED LOCAL.

La tecnología de red de área local que se utilizará será IEEE 802.3 (Ethernet estandarizado por el IEEE). Esta tecnología será implantada en una tarjeta conocida como NIC (*Network Interface Card*) y un programa manejador de la tarjeta (*device driver*), quienes en conjunto formarán la interfaz entre la aplicación y el medio de transmisión. La NIC se instalará en una ranura de expansión de la computadora personal.

SOFTWARE PARA TRANSPORTE DE RED.

El protocolo de transporte de red será TCP/IP. Como primera opción, se utilizará el módulo TCP/IP que incluye Windows 95. También es posible utilizar otro producto que ofrezca este servicio para Windows 95, siempre y cuando estemos seguros que se implante el estándar para programación en red conocido como Windows Sockets 1.1 o superior. Este *software* será implantado como una librería de enlace dinámico (p.ej. winsock.dll).

SOFTWARE PARA ADMINISTRACION DE RED.

El Sistema de Monitoreo y Diagnóstico será una aplicación Windows que se desarrollará utilizando una interfaz para la programación de aplicaciones (API) conocida como WinSNMP/Manager 1.1a. Esta API nos proporciona un marco de desarrollo para crear aplicaciones de administración de redes IP. Esta interfaz será implantada como una librería de enlace dinámico (winsnmp.dll), y nos proporciona funciones específicas de administración de red.

Los esquemas de desarrollo e implantación de la solución propuesta se muestran en la figura 5.1.

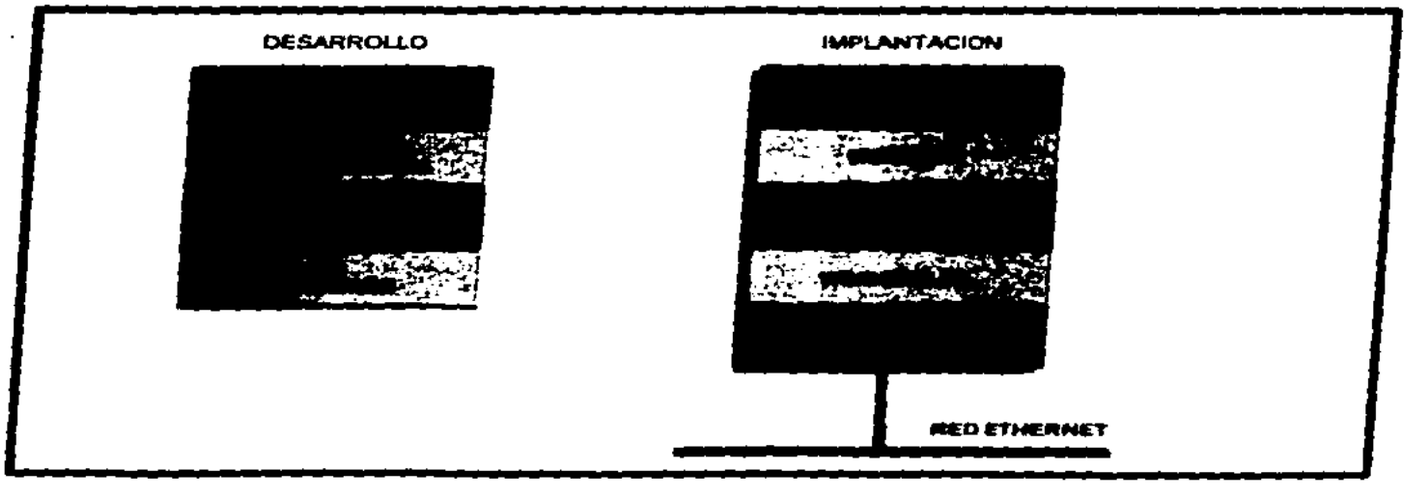


Figura 5.1 Esquemas de desarrollo e implantación.

5.2.3 Justificación de la solución.

Las características definidas para el SMD-97 impactarán en el diseño e implantación de la solución, por tal razón es necesario justificarlas basados en los aspectos que se enunciaron anteriormente.

Primero, para realizar la justificación es conveniente establecer prioridades en los aspectos citados, tal como lo muestra la siguiente figura.

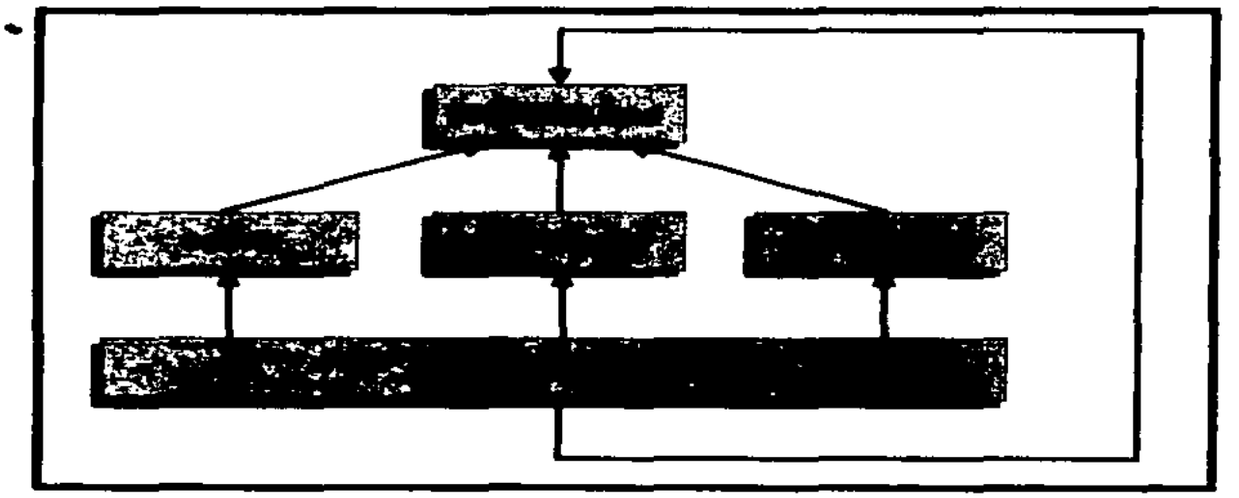


Figura 5.2 Prioridades en los aspectos de evaluación.

En torno a la funcionalidad se encuentran los demás aspectos organizados en dos niveles jerárquicos. En el primer nivel están el costo, la simplicidad y el rendimiento del sistema, y en un segundo estrato podemos ver el ciclo de desarrollo del sistema.

La funcionalidad define explícitamente si la solución cumple con los requerimientos. El costo, nos determina cuánto capital implica el desarrollo del sistema. El rendimiento nos dice si la solución degrada el desempeño del equipo de cómputo y red. El aspecto de la simplicidad tecnológica nos muestra la relativa sencillez o complejidad que presenta la solución tecnológicamente hablando, es decir, si implica el desarrollo de algunos algoritmos, estructuras de datos, etc. que puedan hacer más sencillo o complejo el sistema.

En el segundo estrato, podemos observar un concepto un tanto abstracto que es el ciclo de desarrollo del sistema. Lo que se trata de definir aquí es que el conjunto de las características antes citadas se basa e influye directamente en el ciclo de desarrollo de un sistema.

En resumen, cada una de las características de la propuesta de solución será evaluada y justificada basándonos en la jerarquía establecida, es decir, primero veremos si se cumple con la funcionalidad y enseguida se definirá si el costo, simplicidad y rendimiento las justifican. Sin embargo, es posible que en algunas de las características sólo se tomen en cuenta sólo algunos de los aspectos.

PLATAFORMA.

En la actualidad, podemos encontrar computadoras personales muy poderosas que ponen a disposición del usuario una gran capacidad de procesamiento. En estos equipos se pueden ejecutar paquetes de software tan complejos y prácticos, como pueden ser los usados para diseño arquitectónico, diseño gráfico, análisis y simulación de dispositivos electrónicos, administración de sistemas y redes, y los paquetes comunes que se usan en una oficina.

Los detalles principales que presentan las computadoras personales actuales son el uso de una forma mejorada de la tecnología CISC (*Complex Instruction Set Computer*), direccionamiento de 32 bits y altas velocidades de *bus* interno.

Comparativamente hablando, la tecnología RISC (*Reduced Instruction Set Computer*) es más rápida que la CISC ya que utiliza sólo instrucciones sencillas de procesador, es decir, que no necesitan varios ciclos de reloj para ejecutarse. RISC es muy utilizada en estaciones de trabajo de uso especializado.

La justificación para usar una computadora personal con microprocesador Intel como plataforma es guiada básicamente por el costo y la simplicidad, ya que funcionalmente una computadora personal y una estación de trabajo cumplen con lo requerido. El costo de una computadora personal es de diez a quince veces menor que el de una estación de trabajo, además de que la primera la podemos adquirir en un tiempo muy corto. Los procesadores Intel son los más usados y por lo tanto los más probados en el mercado. Los parámetros de memoria, disco duro y video definidos, son los mínimos necesarios que se requieren.

Es conveniente hacer notar que la plataforma utilizada será preponderante en la elección de las demás características del sistema.

SISTEMA OPERATIVO Y AMBIENTE.

Para este sistema se eligió Windows 95 como ambiente operativo ya que presenta básicamente tres grandes ventajas sobre sus versiones

anteriores (Windows 3.1 y 3.11) : independencia del sistema operativo MS-DOS ; es una ambiente de 32 bits lo cual permite optimizar el uso del sistema principalmente de la memoria RAM y por lo tanto agiliza y da velocidad a la ejecución de las aplicaciones ; además, es un sistema multitarea, queriendo decir con esto que podemos ejecutar varias aplicaciones simultáneamente.

Se considera que Windows 95 es un sistema operativo real, ya que no necesita de MS-DOS como base para instalarlo. Sin embargo, Windows 95 no es un sistema independiente del procesador, ni proporciona facilidades de multiprocesamiento simétrico como lo hace Windows NT. De hecho, podemos decir que Windows 95 es una etapa intermedia entre la migración del ambiente original MS-DOS/Win 3.1 al novedoso y más potente NT.

En la actualidad, Windows 95 está siendo utilizado en un gran número de computadoras ya que además de presentar grandes avances en la interfaz del usuario, está siendo distribuido en los equipos que venden los proveedores.

Aunque la solución se haya diseñado para Windows 95, esto no es una limitante para que no pueda ser ejecutado en Windows 3.1 y 3.11, sólo que sería necesario implantar una de las siguientes opciones:

- Agregar un programa que maneje funciones de 32 bits, tal como Win32s, o
- Recompilar el sistema con una versión a 16 bits de la herramienta de desarrollo.

LENGUAJE Y HERRAMIENTA DE DESARROLLO.

Existen diversas razones por las que se ha elegido utilizar Delphi 2.0 como herramienta para desarrollar el SMD-97. En este caso, los aspectos que ayudaron a tomar tal decisión fueron la funcionalidad, simplicidad y rendimiento.

Realmente, cualquier herramienta de programación moderna pude llegar a ser adecuada dependiendo del tipo de aplicación y requerimientos que se soliciten, además de considerar las preferencias de cada programador.

La elección de un equipo, sistema operativo, o lenguaje de programación, siempre se fundamenta en la comparación que se hace con entidades similares a cada una de éstas. Por esta razón, es necesario comparar Delphi 2.0 con herramientas de desarrollo similares tomando en cuenta los tres aspectos antes mencionados. Algunas herramientas existentes en el mercado son Visual Basic, Visual C y PowerBuilder.

Delphi 2.0 es una herramienta orientada a objetos que utiliza *Object Pascal* como lenguaje de programación. Object Pascal proporciona la simplicidad de uso de un lenguaje de cuarta generación (4GL) y el rendimiento y flexibilidad de uno de tercera generación (C,Pascal, Fortran, etc.), además de presentar todas las características de un lenguaje orientado a objetos real.

En Delphi 2.0, ya no es necesario programar paso a paso las aplicaciones Windows como se hacía con Trubo Pascal o lenguaje C, ya que brinda la facilidad de crear y agregar el código que utilizarán las interfaces al usuario conforme éstas se diseñan. Delphi almacena la información del diseño de la interfaz al usuario en *formas* y todo el código correspondiente en *unidades*. En realidad Delphi oculta el modelo de manejo de mensajes de Windows, para facilitar la programación en este ambiente.

La gran ventaja que presenta Delphi 2.0 sobre Visual C, es que su compilador es más rápido, y Object Pascal es mucho más sencillo de aprender y entender que el lenguaje C. Ciertamente C es más flexible y ofrece independencia de la plataforma, pero estas características no son limitantes en nuestro desarrollo. Aunque si en un futuro se decidiera migrar a C, existen herramientas que nos traducirían el código de Object Pascal a C sin tener que rehacer totalmente el programa.

Delphi proporciona varias ventajas sobre Visual Basic y PowerBuilder, sin embargo las siguientes son las principales :

- Delphi tiene un compilador propio de 32 bits con el cual se generan aplicaciones más pequeñas y más rápidas, que aprovechan las características específicas de 32 bits de Windows 95.
- Las otras herramientas generan aplicaciones en código-p lo que hace necesario el uso de un intérprete. El programa ejecutable (*.EXE) obtenido contiene el código-p y su intérprete, lo cual implica un mayor tamaño del archivo y menor velocidad de ejecución.
- Object Pascal es un lenguaje de programación orientado a objetos real.
- Delphi 2.0 ofrece la capacidad de poder crear librerías de enlace dinámico propias (*.DLL) para una aplicación y las otras herramientas no.
- Delphi 2.0 presenta la facilidad de poder crear componentes personalizados muy rápidamente. Se cuenta con una gran biblioteca de componentes virtuales (VCL).

TECNOLOGIA DE RED LOCAL.

Aunque el SMD-97, al igual que TCP/IP, no impone reestricciones en la capa de enlace de datos, es decir, no utiliza una tecnología específica, se eligió utilizar Ethernet debido a la sencillez de su implantación y operación, además de que su costo es reducido con respecto a tecnologías como Token-Ring, FDDI y ATM.

SOFTWARE PARA TRANSPORTE DE RED.

Debido a la decisión de utilizar el ambiente operativo Windows 95, se reestringen las opciones del software de transporte de red, ya que necesariamente se deberá implantar utilizando una DLL porque los TSR/DOS no son cien por ciento funcionales en este ambiente, ya que se trabaja en un modo protegido.

Realmente, la elección es utilizar el estándar para el desarrollo de aplicaciones para red, conocido como Windows Sockets 1.1 (a finales de 1997 se libera la versión 2.0). Existen diversos fabricantes que han desarrollado productos TCP/IP usando Windows Sockets, pero se eligió utilizar el módulo que proporciona Windows 95 porque es una interfaz a 32 bits funcional, además de que no implica mayores costos.

SOFTWARE PARA ADMINISTRACION DE RED.

La interfaz *WinSNMP/Manager* es muy valiosa ya que simplifica en gran medida el desarrollo de aplicaciones de administración de red para Windows. Si no existiera esta API, sería muy complicado y laborioso la programación de tareas de adminstración de redes. Es conveniente mencionar que la interfaz que se usará es de dominio público, por lo tanto se ahorra tiempo y esfuerzo en el desarrollo de una propia.

5.3 Diseño del Sistema.

En el desarrollo del Sistema de Monitoreo y Diagnóstico se utilizará la estrategia de diseño que utiliza Delphi 2.0. A continuación se muestran las etapas que se incluyen en esta estrategia :

- Especificación de requerimientos del sistema.
- Algoritmo del sistema.
- Diagrama de flujo de operaciones y organización en módulos funcionales.

5.3.1 Especificación de requerimientos.

a) Interfaz del usuario.

Se requiere desarrollar una aplicación para el ambiente Windows con todos los atributos y funcionalidades que presentan este tipo de aplicaciones, es decir, con una interfaz del usuario que tenga las siguientes características:

- Menú descendente (*Pop-up*)
- Inclusión de elementos gráficos como imágenes, iconos, etc.
- Minimización y maximización de ventanas.
- Movimiento de las ventanas a través de la pantalla.
- Posibilidad de manejar ventanas descendientes (*child windows*).
- Ayuda en linea.

b) Pantallas del sistema.

Mínimo, se requiere la creación de las siguientes pantallas para el Sistema de Monitoreo y Diagnóstico :

- Pantalla de inicio del sistema. Esta presentará el menú principal, rango de direcciones IP que se consultarán, fecha y hora del sistema. Esta pantalla será la referencia para las demás, de hecho, algunas utilizarán la misma área de cliente para presentar información.
- Pantalla de configuración del sistema. En esta pantalla se presentarán e introducirán los siguientes datos : dirección IP de la computadora donde se ejecuta el SMD-97, número de retransmisiones, tiempo de duración de la consulta, comunidad y tiempo entre consultas.
- Pantalla de ejecución y resultados del sistema. En esta se presentarán los resultados obtenidos de la ejecución del sistema.
- Pantalla de ayuda en linea del sistema. En esta pantalla se podrá visualizar la información de ayuda del sistema.

c) Información resultante del sistema.

Determinar cuántos agentes SNMP existen en una red IP clase C, y presentar en forma tabular y en tiempo real, la siguiente información de cada uno de los agentes :

- Nombre de cada nodo que contiene agente SNMP.
- Dirección IP del nodo.
- Tiempo de activación (tiempo transcurrido desde la última vez que se reinicializó el nodo).
- Responsable de administrar el nodo.
- Descripción del nodo.

Además, se deberá implantar un mecanismo mediante el cual se realicen las consultas SNMP de manera automática y periódica. Los datos anteriores son objetos que forman parte del grupo *system* de la MIB-II.

5.3.2 Algoritmo general del Sistema.

El algoritmo del Sistema de Monitoreo y Diagnóstico se muestra a continuación :

1. Inicialización. En esta etapa se definen e inicializan variables, constantes, procedimientos, etc.
2. Configuración. Aquí se definen y capturan los diversos parámetros de configuración.
3. Ejecución. Esta parte ejecuta la consulta de nodos y de manera indirecta maneja el *loop* de mensajes propio de Windows.
4. Presentación de resultados. Lleva a cabo la presentación de los resultados requeridos.
5. Terminación. Libera los recursos utilizados por la aplicación y termina la aplicación.

Cabe hacer notar que en todo momento se tendrá la disponibilidad de obtener ayuda en linea.

5.3.3 Diagrama de flujo de procedimientos del Sistema.

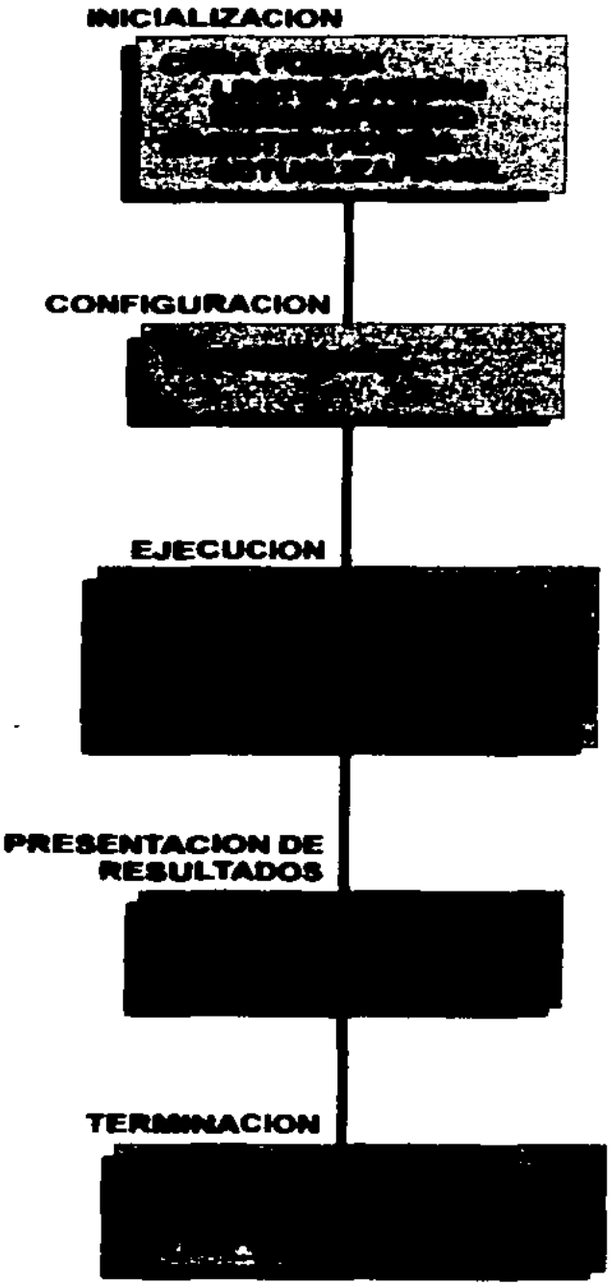


Figura 5.3 Diagrama de Flujo del Sistema.

DESCRIPCION DE PROCEDIMIENTOS.

INICIALIZACION

CREAFORMA
Se realiza la construcción de la forma de Consulta en donde se realizan las siguientes funciones:

LECTURAREGINI
En este procedimiento se hace la lectura de los valores iniciales desde el archivo de registro, aquí se manejan valores como el tamaño de la ventana, las direcciones IP de inicio y final para la realización de la búsqueda de Agentes Snmp así como también los valores para el tamaño de las columnas de la ventana donde se despliegan los datos.

WINSNMPINICIO
Aquí se inicializa la interfaz WinSnmp y se define información global, se hace referencia a la función SnmpStartup la cual notifica que se habilitó el servicio para realizar cualquier otra llamada de funciones de Winsnmp.

MUESTRAFORMA
En este procedimiento se realiza la activación de la forma de Consulta del programa. Se realiza la inicialización de variables globales.

ACTUALIZA PANEL
Se realiza la actualización del Panel de Estado para verificar el estado del programa.

CONFIGURACION

PARAMETROS
En este procedimiento se definen los parámetros para la realización de las consultas, se definen la Dirección IP local del Administrador, el número de retransmisiones, el tiempo de consulta por cada retransmisión, la comunidad de las MIB y el tiempo entre consultas para monitorear los agentes SNMP en intervalos de tiempo definidos.

EJECUCION

INICIACONSULTA
Este procedimiento realiza la consulta de todos los datos de trabajo y envía las consultas a la red.

SNMPINICIOCONEXION
Se abre una nueva sesión e inicializa Entidades, Contexto, Tiempo límite de consulta y el número de retransmisiones.

SNMPCONSULTAINICIAL
En este procedimiento se crean y se envían consultas de estado, se define la transmisión del PDU de la entidad destino. Cuando la consulta es recibida WinSnmp determina la versión de SNMP.

ACTUALIZAPANEL
Aquí se actualiza el panel de estado, se realiza el cambio del panel en modo gráfico y en modo texto el cual me indica el estado del la consulta.

CANCELARCONSULTA
En este procedimiento se cancela la operación de consulta y se actualiza el panel de estado.

PRESENTACION DE RESULTADOS

SNMPEVENTO
En este procedimiento se manejan los mensajes de WinSnmp, se verifica si existen mensajes en espera para poder procesarlos, posteriormente se obtienen los datos del PDU y una vez realizada la consulta se inicializa el registro de resultados, se incrementa el contador de agentes descubiertos y se actualiza el panel de estado.

VALORESSALIDA
Presenta los datos de salida SNMP OID-VALOR, es importante en este procedimiento liberar los descriptores de OID y VALOR después de terminar el procedimiento ya que si no se realiza provoca problemas de pérdida y bloqueo de memoria.

TERMINACION

SNMPCERRARCONEXION
Este procedimiento cierra la sesión SNMP liberando el contexto de la entidad SNMP.

ESCRITURAREGINI
En este procedimiento se hace la escritura de los últimos valores utilizados al archivo de registro, aquí se manejan valores como el tamaño de la ventana, las direcciones IP de inicio y final para la realización de la búsqueda de Agentes Snmp así como también los valores para el tamaño de las columnas de la ventana donde se despliegan los datos.

5.4 Implantación del Sistema.

5.4.1 Codificación.

Una vez creados los módulos funcionales del sistema y definido las variables de entrada y salida de cada uno de estos en la fase de diseño, procederemos a codificarlos en Object Pascal, que es el lenguaje de programación que utiliza Delphi.

Debido a lo extenso del código fuente generado (aproximadamente 1450 lineas y 24 páginas), sólo presentaremos algunos módulos del sistema. El programa se llama *Consulta.pas* y se mostrarán los módulos *IniciaConsulta*, *SnmpConsultaInicial* y *WinSnmpInicio*.

```
{
 SpeedButtonIniciaConsultaClick - Inicia la consulta

 Aquí es donde la consulta inicializa todos los datos de trabajo y envía las
 consultas a la red.
}
procedure TFormaConsulta.IniciaConsulta(Sender: TObject);
 var
   tempo : integer;
   var1  : string;
   linea_horafecha : TListItem;
   i, j : Integer;
   s : SNMPAPI_STATUS;
   ts : array[0..40] of Char;
   DirIParranque : LongInt;
   DirIPparo : LongInt;
   DirIPactual : LongInt;
   IParranque : array[0..3] of Byte;
   IPparo  : array[0..3] of Byte;
   IPactual : array[0..3] of Byte;
begin
 // Abre una sesión Snmp
 sed.snmpHandle := Handle;
 sed.snmpMessage := WM_SNMP_MSG;
 if (sed.hSnmpSesion = 0) then begin
   sed.hSnmpSesion := SnmpOpen(sed.snmpHandle, sed.snmpMessage);
 end;
 if (sed.hSnmpSesion = SNMPAPI_FAILURE) then
   Exit;
 // Habilita/deshabilita botones aceleradores y actualiza la forma
 SpeedButtonInicioConsulta.Enabled := False;
 SpeedButtonCancelarConsulta.Enabled := True;
{  ListaConsulta.Items.Clear;  }
 Update;
 // Coloca la barra de progreso en 0 y actuliza el panel de estado
 // La actualización del panel de estdo conmutará el panel 1 de modo texto
 // a gráfico (ownerdraw)
```

```
i := 0;
BarraEstadoConsulta.Position := 0;
BarraEstadoConsulta.Step := 10;
ConsultaPendiente := True;
ActualizaPanel;
// Obtiene las dir. IP de arranque y paro de las comboboxes
DirIParranque := inet_addr(PChar(ValorInicialConsulta.Text));
DirIPparo := inet_addr(PChar(ValorFinalConsulta.Text));
// Copia las direcciones para su uso posterior
CopyMemory(@IParranque[0], @DirIParranque, 4);
CopyMemory(@IPparo[0], @DirIPparo, 4);
{
  Define la IP actual como dirección de arranque
  Consulta está diseñado para manejar direcciones IP que pertenezcan al rango
  de las redes clase C. Si se teclean rangos de redes clases A o B en las
  StartAddressComboBoxes sólo se enviarán consultas (queries) para las prime-
  ras 255 direcciones en el rango.

  Si en en futuro, se piensa que la consulta se extienda a redes clase A y B
  no se deben de enviar más de 255 consultas en un instante (sin embargo,
  esto es posible). La mejor manera de llevar a cabo esto es enviar las 255
  consultas, después esperar un momento y en periodos breves de espera enviar
  las 255 nuevas consultas...
}
// Define los descriptores de la consulta
CopyMemory(@IPactual[0], @DirIParranque, 4);
j := 0;
for i := IParranque[3] to IPparo[3] do begin
  IPactual[3] := i;
  CopyMemory(@DirIPactual, @IPactual[0], 4);
  rseda[j].ocupado := True;
  rseda[j].DirIP := inet_ntoa(TInAddr(DirIPactual));
  Inc(j);
  if j >= MAX_ENTIDADES then
    Break;
end;
// Salir si no se ha definido ningún descriptor para la consulta
if j = 0 then
  Exit;
// Inicializa conexión
SnmpInicioConexion(sed);
// Inicializa información de la consulta
ts := '1.3.6.1.2.1.1.3';    { se define el OID para sysUpTime }
if sed.oid.ptr <> nil then
  SnmpFreeDescriptor(SNMP_SYNTAX_OID, @sed.oid);
SnmpStrToOid(ts, @sed.oid);
// no existe condición de paro de OID
sed.OidParo.len := 0;
sed.OidParo.ptr := @sed.ValorOidParo[0];
// Se crea una consulta y se envía mensaje
// Envía todas las consultas en un loop
BarraEstado1.Panels.Items[2].Text := 'Consultando ...';
for i := 0 to j - 1 do begin
  // se brinca si el descriptor no está ocupado
```

```
    if rseda[i].ocupado = False then
      Continue;
    // se define el ID de la consulta (request ID)
    rseda[i].IDConsulta := i + 100;
    // se define la entidad del agente SNMP remoto
    rseda[i].hEntidadAgente := SnmpStrToEntity(sed.hSnmpSesion, PChar(rseda[i].DirIP));
    // se definen las retransmisiones y timeout para la entidad
    s := SnmpSetRetry(rseda[i].hEntidadAgente, StrToInt(FormaParametros.Retransmision.text));
    s := SnmpSetTimeout(rseda[i].hEntidadAgente, StrToInt(FormaParametros.TiempoConsulta.text));
    // Se crea un PDU
    SnmpConsultaInicial(sed, rseda[i].hEntidadAgente, rseda[i].IDConsulta);
    // coloca la posición de la barra de progreso, texto de la barra de estado y
    // las actualiza
    BarraEstadoConsulta.Position := i * 80 div j;  //MAX_ENTIDADES;
    BarraEstado1.Panels.Items[1].Text := 'Consulta ' + IntToStr(i) + ' ...';
    ActualizaPanel;
  end;

  {
   Reinicializa el progreso
   Cuando se envían las consultas la barra de progreso se coloca para mostrar
   la cantidad de consultas enviadas a la red.

   En la recepción de consultas, se observa el progreso con el fin de mostrar
   el tiempo aproximado que falta para que termine la operación. Este progreso
   se modifica de acuerdo al timer.
  }
  BarraEstadoConsulta.Position := 0;
  BarraEstado1.Panels.Items[2].Text := 'Esperando Respuestas ...';
  BarraEstado1.Panels.Items[1].Text := IntToStr(0) +  ' Agentes SNMP Descubiertos.';
  ContadorNodosEncontrados := 0;
  TiempoEsperaRespuesta:=StrToInt(FormaParametros.Retransmision.text)*
  StrToInt(FormaParametros.TiempoConsulta.text) div 10;

  tiempoRespConteoDesc := TiempoEsperaRespuesta;
  if TiempoEsperaRespuesta > 0 then
    Timer1.Enabled := True;
  //**********************************
   linea_horafecha := Listaconsulta.Items.Add;
   linea_horafecha.Caption := datetostr(date) +  '' + timetostr(time);
   var1:='';
   tempo:=0;
   tempo:=strtoint(formaparametros.tiempomonitor.text);
   tempo:=tempo*60;
   var1:=inttostr(tempo)+'000';
   timer2.interval:=strtoint(var1);
   timer2.enabled:=true;
   Application.Restore;
 end;

procedure TFormaConsulta.SnmpConsultaInicial(var se: SnmpEntidad;
                                             hEntidadAgente : HSNMP_ENTITY;
                                             IDConsulta : smiUINT32);
```

```
 var
   s: SNMPAPI_STATUS;
   i : Integer;
   oid : smiOID;
   val : smiVALUE;
begin
 // Crea la VBL
 se.hVbl := SnmpCreateVbl(se.hSnmpSesion, @se.oid, nil);
 // define la VBL del arreglo TVbls
 for i := 0 to MAX_CONSULTAS_VBLS do begin
   val.syntax := SNMP_SYNTAX_NULL;
   val.empty := 0;
   s := SnmpStrToOid(PChar(TVbls[i]), @oid);
   s := SnmpSetVb(se.hVbl, 0, @oid, @val);
   s := SnmpFreeDescriptor(SNMP_SYNTAX_OID, @oid);
 end;
 se.IDConsulta := IDConsulta;
 // Crea PDU
 se.hPdu := SnmpCreatePdu(se.hSnmpSesion, SNMP_PDU_GETNEXT, se.IDConsulta, 0, 0, se.hVbl);
 // Envía consulta a la red
 s := SnmpSendMsg(se.hSnmpSesion, se.hEntidadAdmin, hEntidadAgente, se.hContextoVisual, se.hPdu);
 // Libera información interna WinSNMP
 s := SnmpFreeVbl(se.hVbl);
 s := SnmpFreePdu(se.hPdu);
end;
{
 WinSnmpInicio
 Inicializa WinSNMP y se definen información global
}
function TFormaConsulta.WinSnmpInicio(se : SnmpEntidad) : Boolean;
 label
   terminar;
 var
   error: Bool;
   s: SNMPAPI_STATUS;
   ts : String;
begin
 Result := True;
 // Inicia WinSnmp.dll
 s := SnmpStartup(@se.nMajorVersion, @se.nMinorVersion, @se.nLevel, @se.nTranslateMode,
             @se.nRetransmitMode);
 if s = SNMPAPI_FAILURE then begin
   s := SnmpGetLastError(se.hSnmpSesion);
   MessageDlg('Error SnmpStartup : ' + IntToStr(s), mtInformation, [mbOk], 0);
   Result := False;
   Exit;
 end;
 // Define el modo de traducción
 if (se.nTranslateMode <> SNMPAPI_UNTRANSLATED_V1) then begin
   s := SnmpSetTranslateMode(SNMPAPI_UNTRANSLATED_V1);
   if s = SNMPAPI_FAILURE then begin
     s := SnmpGetLastError(se.hSnmpSesion);
     MessageDlg('Error SnmpSetTranslateMode : ' + IntToStr(s), mtInformation, [mbOk], 0);
     Result := False;
```

```
    Exit;
   end;
  end;
  s := SnmpSetRetransmitMode(SNMPAPI_ON);
  se.bWinSnmpIniciado := True;
end;
```

5.4.2 Documentación y ayuda en linea.

Código fuente.

La documentación del código fuente de un sistema empieza por la definición de los nombres de las variables y etiquetas, para seguir después con los procedimientos y funciones. Como se puede observar, en los módulos codificados que se presentaron anteriormente existen enunciados que delimitados por los caracteres especiales "//" y "{}" proporcionan información de cada procedimiento, indicando su objetivo principal, y los datos de entrada/salida.

Instalación.

Un aspecto muy importante de la documentación, es el procedimiento de instalación del *software*. A continuación se muestra una secuencia de pasos a seguir para instalar el SMD-97 por primera vez en una computadora personal (PC):

- Asegurarse de tener instalado Windows95 en la PC.
- Habilitar el servicio de TCP/IP en la PC :
  1. Configurar TCP/IP (asignar dirección IP local, del gateway, etc.)
  2. Realizar pruebas de conectividad en la red IP (p.ej. telnet "nodo-remoto")
- Instalar el SMD-97 :
  3. Colocar el diskette de instalación denominado SMD97v1.0 en un drive de 31/2 y teclear "a :\install". A continuación el programa de instalación le hará algunos requerimientos básicos (automáticamente, el SMD97v1.0 se instala en el subdirectorio C :\smd97 y se genera un grupo de programas donde está incluida la aplicación).
  4. Opcionalmente, crear acceso-directo para el SMD-97.

Ayuda en linea.

En la actualidad, no se concibe que existan aplicaciones que no proporcionen *Ayuda en Linea*, es decir, que presenten información del sistema (instalación, utilización, configuración, etc.) en cualquier momento durante la ejecución del mismo. El Sistema de Monitoreo y Diagnóstico-97 ofrece la facilidad de la ayuda en linea, permitiéndolo al usuario contar con un manual de operación en cualquier instante.

La ayuda que presenta el SMD-97 es un archivo denominado SMD97.hlp el cual fué generado utilizando una herramienta conocida como *HelpWriter* para Delphi.

5.5 Prueba y Verificación del Sistema.

En esta etapa del desarrollo del Sistema de Monitoreo y Diagnóstico-97, se llevan a cabo tareas que nos ayudarán a probar y verificar la operación y resultados del sistema. Se define un escenario de prueba con el fin de mostrar lo siguiente :

- Operación del SMD-97
- Rendimiento de la red utilizando SMD-97
- Comparación de la funcionalidad del SMD-97 con CiscoWorks

5.5.1 Escenario de la prueba.

En la figura 5.4 se muestra un esquema donde se interconectan los elementos necesarios para realizar las pruebas del sistema .

Figura 5.4 Escenario de la prueba.

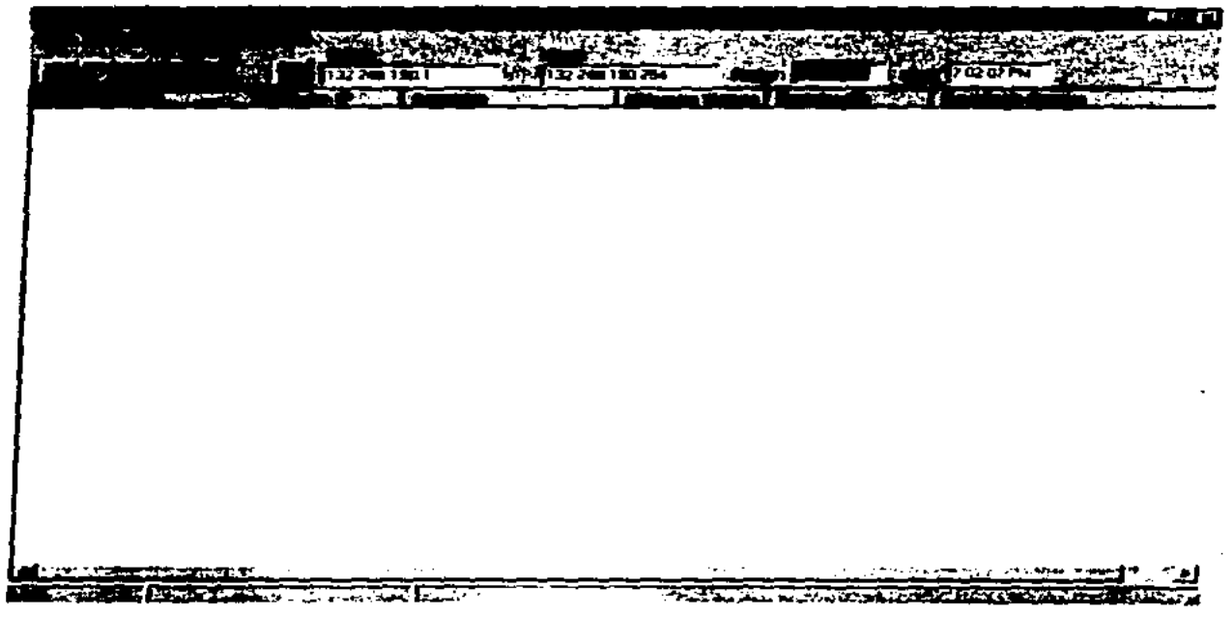
En la computadora personal denominada SMD-97 se encuentra instalado el Sistema de Monitoreo y Diagnóstico-97, en la computadora etiquetada *SNM* se ejecutará la herramienta *SunNetManager* de SUN™ con el fin de obtener información de monitoreo para después compararla con la del SMD-97. El equipo etiquetado *sniffer* será un analizador de protocolos con el cual se obtendrá información de la utilización del canal Ethernet.

Se realizará el monitoreo para las siguientes redes IP ubicadas en la U.N.A.M. : 132.248.10.x y 132.248.190.x.

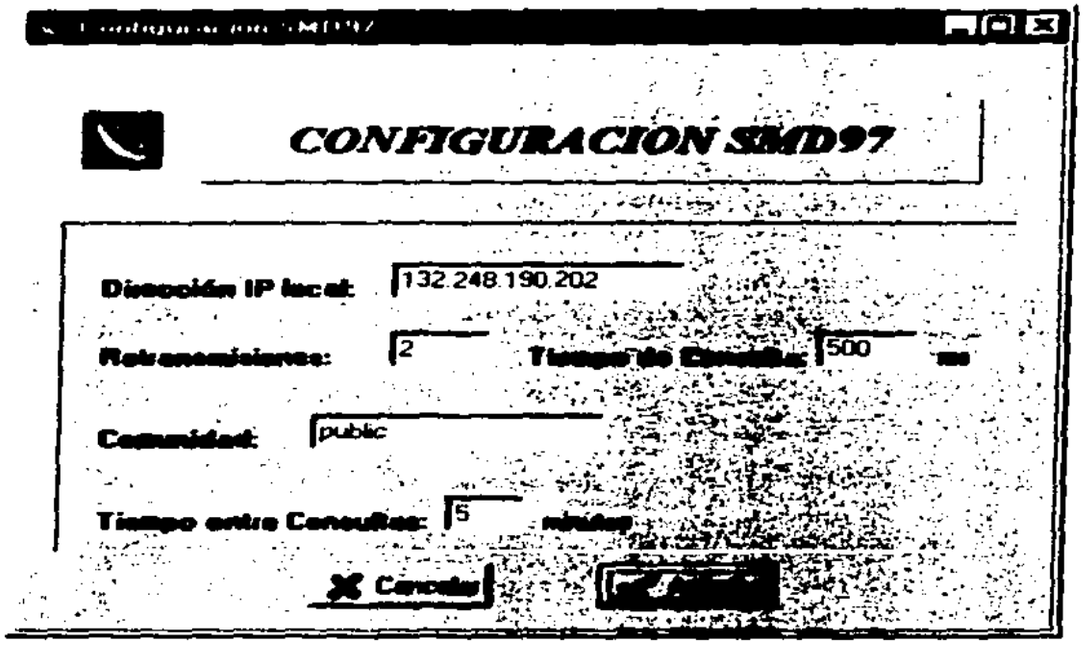
Como podemos notar, las direcciones IP anteriores corresponden a redes clase B, pero como se lleva a cabo un esquema de subrredes (utilizando máscara 255.255.255.0), se generan redes clase C que se identifican por el tercer *byte*.

5.5.2 Operación del SMD-97.

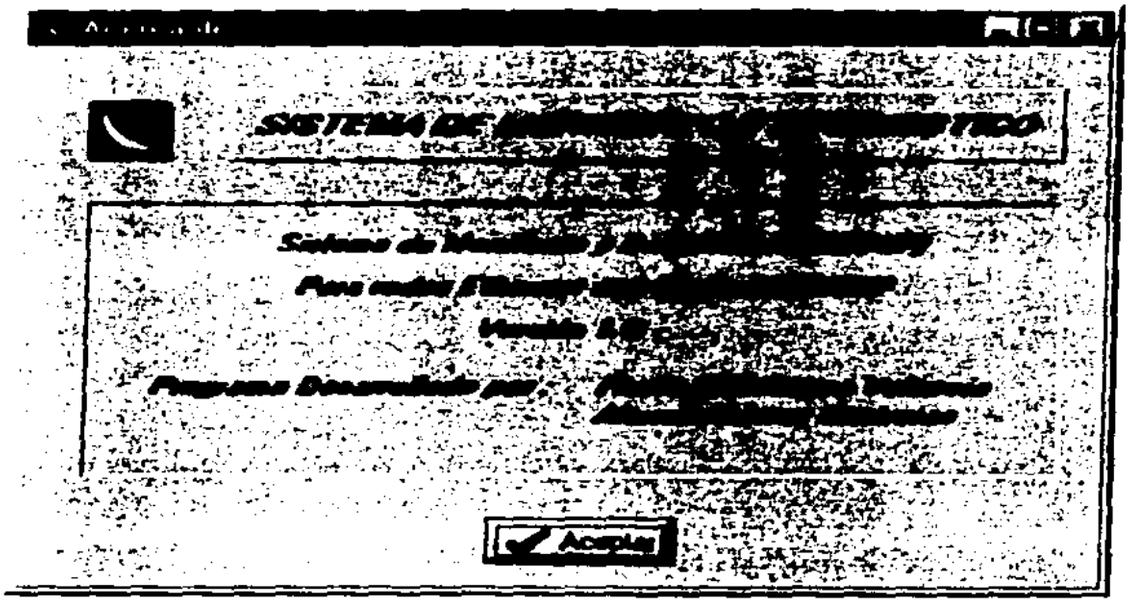
Utilizando el escenario definido, se presentarán las ventanas de operación del sistema, incluyendo : ventana inicial o principal, ventana de configuración, ventana de presentación de resultados (una para cada una de las redes monitoreadas) y la ventana de ayuda en linea.



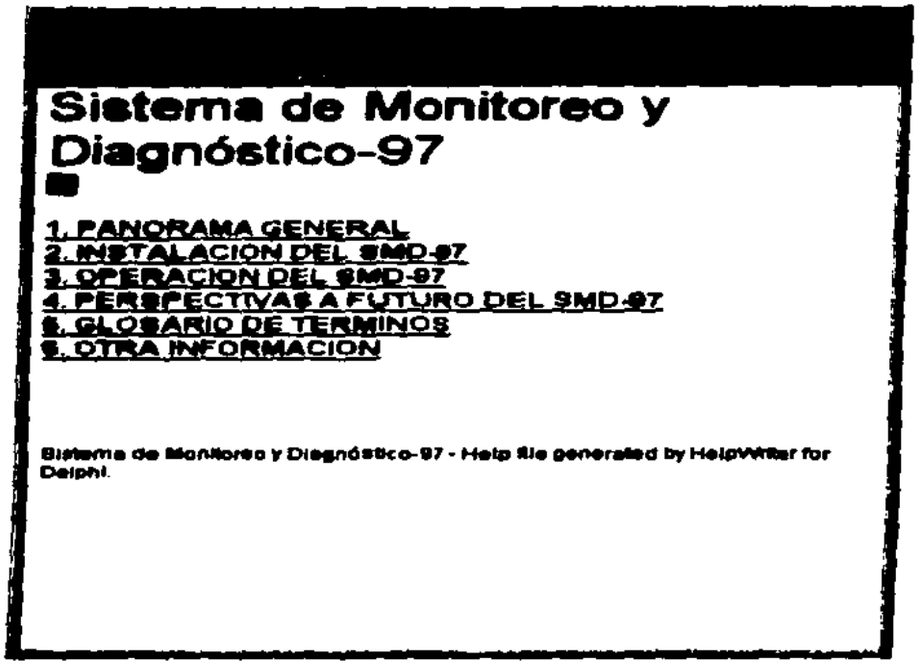
1. Ventana *Inicial*.



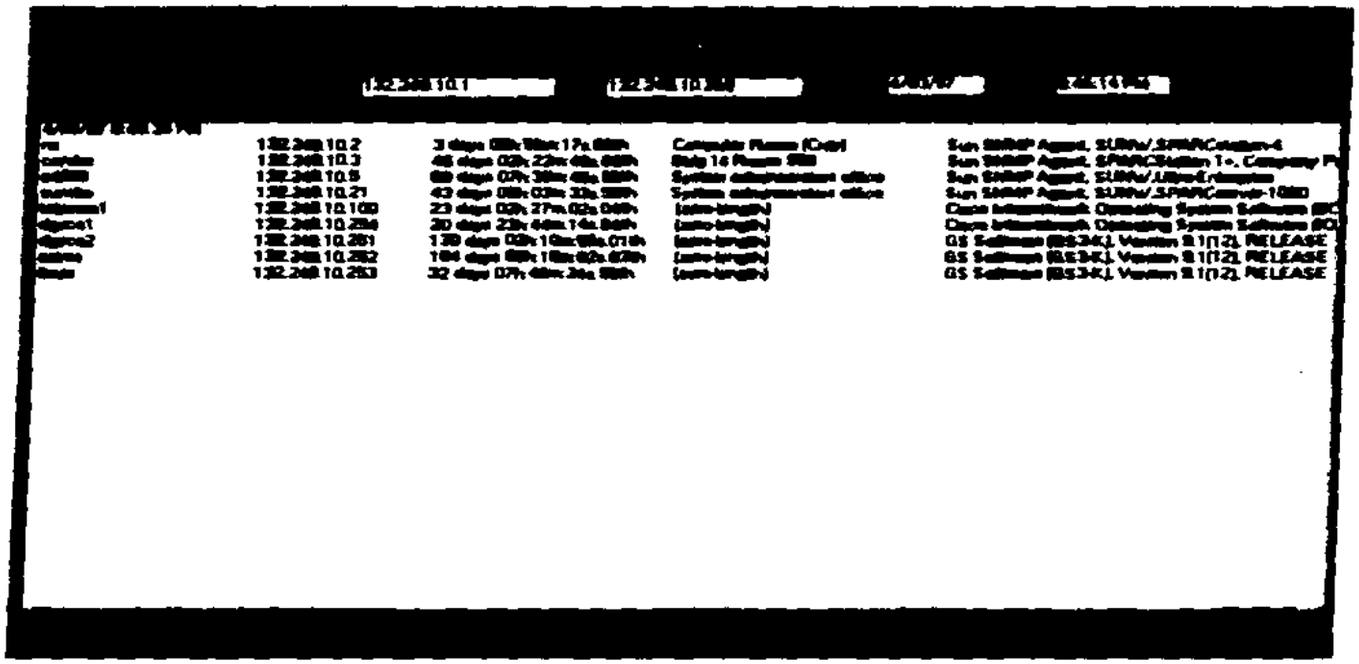
2. Ventana de *Configuración*.



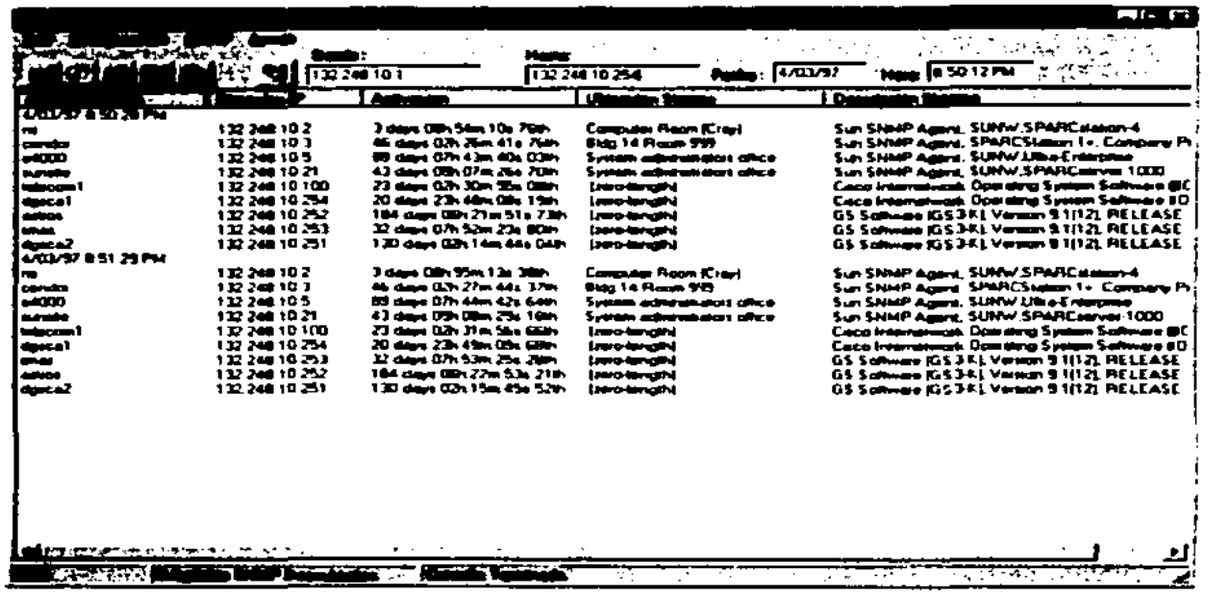
3. Ventana *Acerca de...*



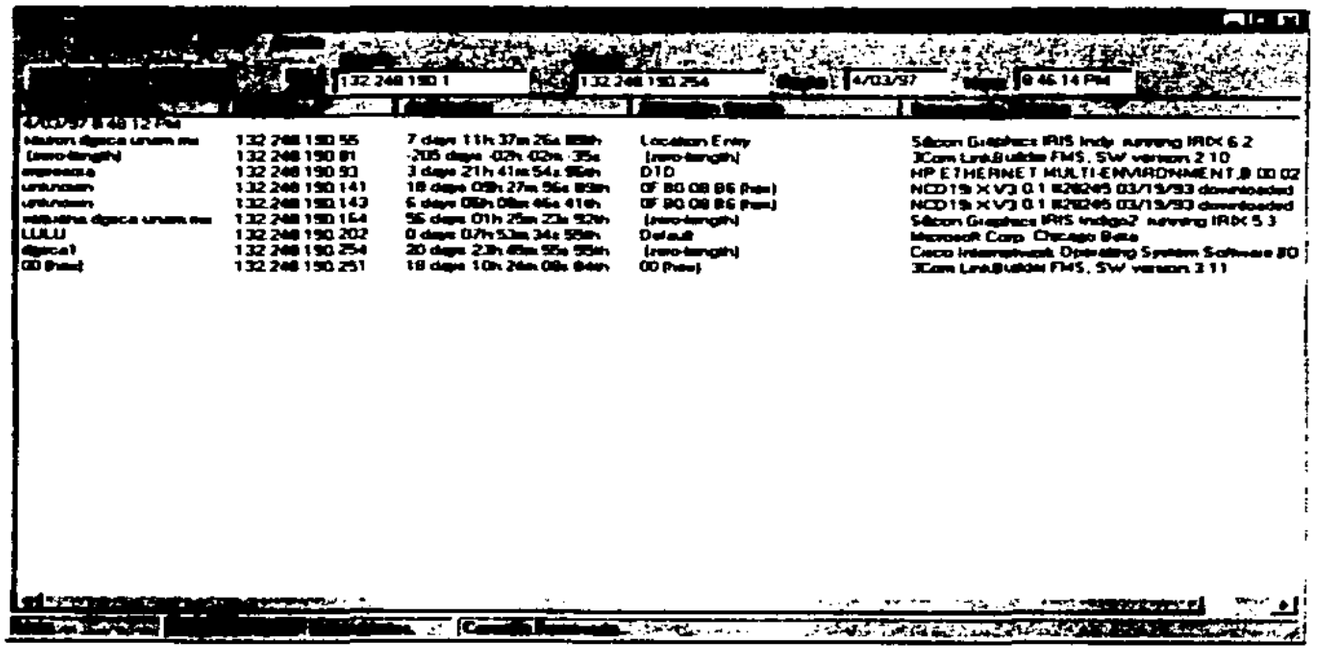
4. Ventana *Temas de Ayuda*.



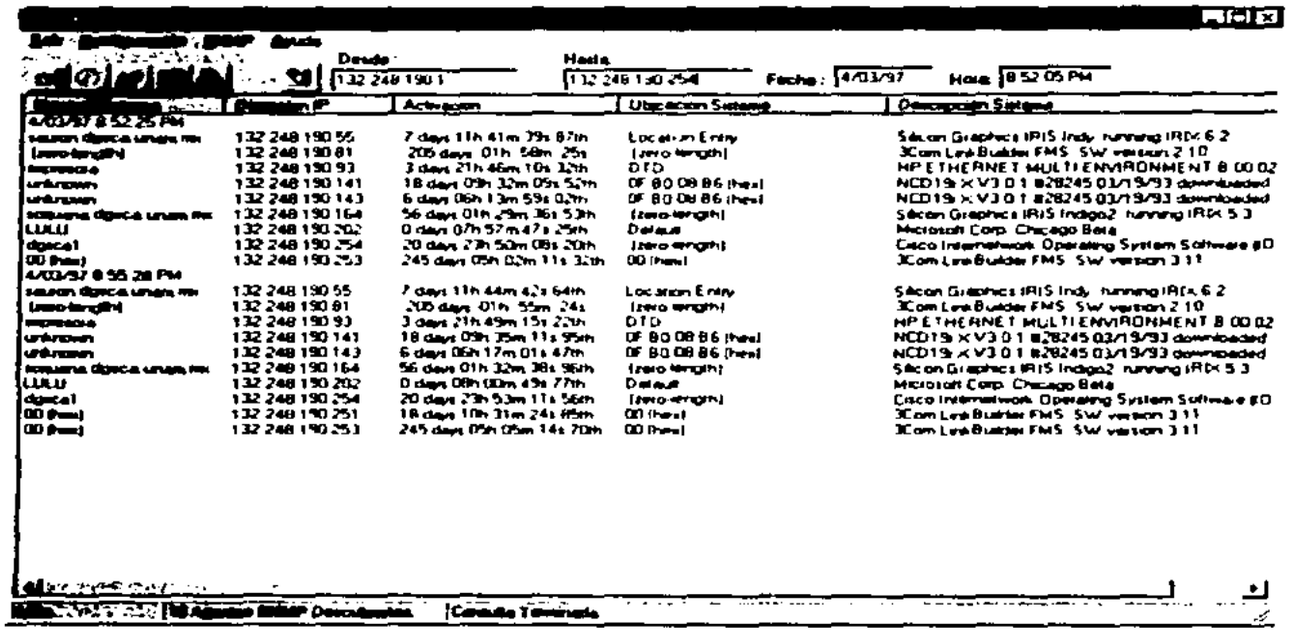
5. Ventana de *Resultados*. Red 132.248.10.x. Monitoreo inicial.



6. Ventana de *Resultados*. Red 132.248.10.x. Monitoreo Automático cada minuto.



7. Ventana de *Resultados*. Red 132.248.190.x. Monitoreo inicial.

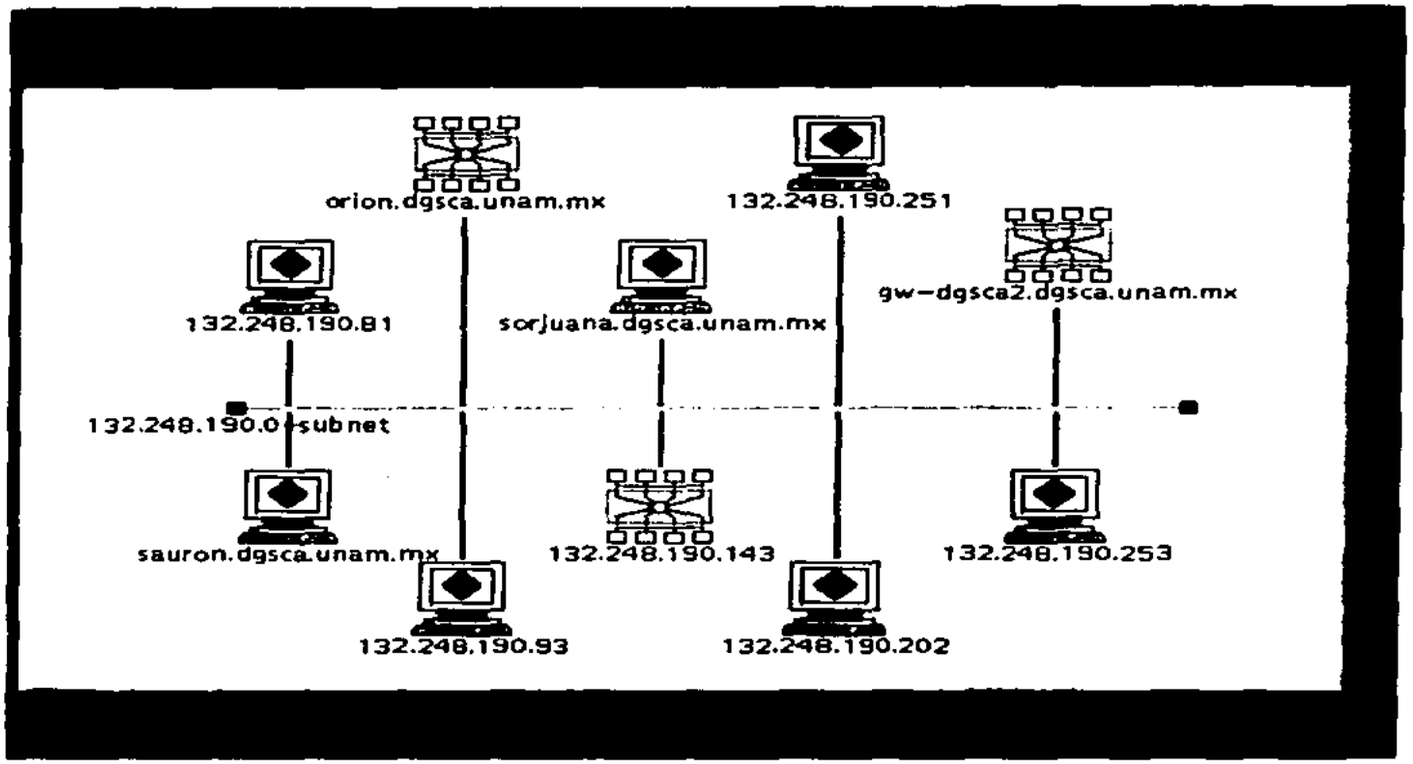


8. Ventana de *Resultados*. Red 132.248.190.x. Monitoreo automético cada tres minutos.

5.5.3 Rendimiento de red al utilizar SMD-97.

Por medio del uso de un analizador de protocolos (*sniffer*), que entre muchas otras cosas, nos proporciona información sobre la utilización del canal Ethernet, se determinará si la ejecución del SMD-97 degrada o no el rendimiento de la red local donde está instalado el SMD-97.

5.5.4 Resultados de monitoreo con *SunNetManager*.




```
Tue Mar  4 20:37:35 1997 [ 132.248.190.202 ] : Quick Dump: snmp-mib11.system

sysDescr=Microsoft Corp. Chicago Beta.
sysObjectID=Microsoft.1.1.3.2
sysUpTime=07:44:46.43
sysContact=Contact: Default
sysName=LULU
sysLocation=Default
sysServices=76
```

```
Tue Mar  4 20:38:24 1997 [ sauron.dgsca.unam.mx ] : Quick Dump: snmp-mib11.syste

sysDescr=Silicon Graphics IRIS Indy  running IRIX 6.2
sysObjectID=Silicon Graphics, Inc..1.130
sysUpTime=179:27:31.70
sysContact=Contact Entry
sysName=sauron.dgsca.unam.mx
sysLocation=Location Entry
sysServices=72
```

```
Starting Discover ...

        Maximum Hops 0
        Adding Link Information
        Find SNMP Objects

Reading Routing Table ...

Adding Networks to Database
Creating view.network 132.248.0.0 in view Home
Creating view.subnet 132.248.190.0 in view 132.248.0.0
Adding Networks Done.
Adding Connections to Database
Adding Connections Done.
Probing Subnets for hosts
Pinging subnet 132.248.190.0
Creating Ethernet 132.248.190.0-subnet to view 132.248.190.0
Adding Host sauron.dgsca.unam.mx to view 132.248.190.0
Adding Link sauron.dgsca.unam.mx-132.248.190.0-subnet-link
Adding Host 132.248.190.81 to view 132.248.190.0
Adding Link 132.248.190.81-132.248.190.0-subnet-link
Adding Host 132.248.190.93 to view 132.248.190.0
Adding Link 132.248.190.93-132.248.190.0-subnet-link
Adding Host orion.dgsca.unam.mx to view 132.248.190.0
Adding Link orion.dgsca.unam.mx-132.248.190.0-subnet-link
Adding Host 132.248.190.143 to view 132.248.190.0
Adding Link 132.248.190.143-132.248.190.0-subnet-link
Adding Host sorjuana.dgsca.unam.mx to view 132.248.190.0
Adding Link sorjuana.dgsca.unam.mx-132.248.190.0-subnet-link
Adding Host 132.248.190.202 to view 132.248.190.0
Adding Link 132.248.190.202-132.248.190.0-subnet-link
Adding Host 132.248.190.251 to view 132.248.190.0
Adding Link 132.248.190.251-132.248.190.0-subnet-link
Adding Host 132.248.190.253 to view 132.248.190.0
Adding Link 132.248.190.253-132.248.190.0-subnet-link
Adding Host gw-dgsca2.dgsca.unam.mx to view 132.248.190.0
Adding Link gw-dgsca2.dgsca.unam.mx-132.248.190.0-subnet-link
Adding Ethernet 132.248.190.0-subnet to view 132.248.190.0
Probing Subnets Done.
Discovery Done.
Content-Type: TEXT/PLAIN; charset=US-ASCII; name="10.txt"Content-ID:
<Pine.GSO.3.95.970305102936.10071D@apollo.noc.unam.mx>
Content-Description:
```

5.5.5 Análisis de resultados.

La realización de las pruebas de operación anteriores nos presentan varios resultados que es conveniente analizar.

Primero, la operación del SMD-97 es muy sencilla ya que basta con configurar cuatro parámetros principalmente (Retransmisiones, TiempoEsperaConsulta, Comunidad y Tiempo entre consultas) y después hacer un click en el botón correspondiente para obtener los resultados de la consulta.

Segundo, el uso de un *sniffer* para medir el desempeño de la red antes y después de la ejecución del SMD-97 nos ayudó a determinar que SNMP es un protocolo que genera muy poca carga en la red y por lo tanto no degrada el rendimiento de las redes.

Y por último, los resultados que obtuvimos al utilizar *SunNetManager* ™ nos permite afirmar que el SMD-97 es una aplicación con la que podemos obtener información útil y real de los agentes SNMP encontrados.

5.6 Perspectivas de Desarrollo futuro del Sistema.

Las perspectivas de desarrollo del sistema de administración para redes IP se basan principalmente en las etapas de implantación del mismo. Algunos de estas futuras implantaciones son las siguientes: Monitoreo Remoto (*RMON, Remote Monitoring*) y fuera de banda (*Out-of-Band*) de redes locales, utilización de MIBs propietarias y muy ligado a estos está el desarrollo de un *software* de Agente para MS-DOS; además, la descripción de las nuevas versiones, tanto de SNMP (Versión 2), como de las APIs utilizadas para el desarrollo y puesta en operación del sistema (WINSOCK 2.0 y WinSNMP 2.0).

5.6.1 Monitoreo remoto (RMON).

Un monitor de red es un dispositivo inteligente el cual tiene conexión directa a la red de área local que se desea analizar o monitorear. Este equipo tiene la ventaja de tener la capacidad de *observar fielmente* el tráfico de información en la red, sin la necesidad de realizar *polling*. Un monitor puede detectar problemas que, utilizando otro método, sería muy difícil de observar, como por ejemplo puede ser la detección de direcciones IP duplicadas.

Los monitores llevan a cabo una función muy importante para las redes que son administradas en forma centralizada, ya que si en un momento dado el enlace que une a la red remota está inactivo, es posible analizar la información que el monitor ha registrado mediante lo que se conoce como monitoreo *fuera de banda*, es decir, levantando un enlace hacia la red central, a través de linea

telefónica conmutada. De esta manera, es posible resolver el problema que provocó la caída del enlace principal entre las dos redes.

En 1991, en el RFC 1271 se definió el estándar de la MIB para RMON. Esta MIB contiene las herramientas que necesita una estación de administración de red para configurar y controlar un monitor, leer datos y reportes, y recibir alarmas. El administrador puede obtener información útil que va desde estadísticas globales hasta capturas detalladas de paquetes.

El estándar proporciona un marco de referencia general para la recolección de estadísticas, alarmas, y captura de datos. Algunas variables son específicas de la tecnología. Por ejemplo, el número de colisiones en un segmento Ethernet, o parámetros que afecten una red Token-Ring. El RFC mencionado define un gran número de variables, en conjunto con las variables específicas para monitorear una red Ethernet.

La MIB RMON está organizada en grupos y cada uno de estos contiene variables que son afines dentro de un grupo dado. Los grupos definidos son los siguientes: *statistics, history, host, hostTopN, matrix, filter, packet, capture,* y *event*.

La manera de implantar un monitor de red que se guíe por estándares varía según el fabricante de equipo y la visión que tenga éste de la tecnología de las redes de datos. Realmente, un monitor puede ser implantado de diversas maneras, pero siempre tendrá características comunes que deberán ser cumplidas. Dentro de estas características podemos encontrar las siguientes:

- Tarjeta de interfaz de red
- Implantar el estándar RMON: completo o por módulos de variables
- Puerto auxiliar para conexión a línea conmutada
- Fuente de alimentación propia

Dentro de los aspectos que pueden variar podemos decir que es la manera de implantar la captura y recolección de datos, es decir, es posible desarrollarlas en un programa en memoria EPROM o un software de aplicación basado en una plataforma específica.

En este proyecto se eligió la implantación de un *Agente SNMP* que maneje RMON para el sistema operativo *MS-DOS*, y se instale en un CPU con microprocesador Intel 386 y el cual contenga un puerto serial RS232C para el monitoreo fuera de banda. El agente se realizará utilizando el estándar para programación con *sockets* en MS-DOS. Un esquema aproximado de la implantación de el monitoreo remoto y fuera de banda, además de la utilización de CPUs con Agentes MS-DOS, se muestra en la figura 5.5. La elección de esta opción se basó en la gran funcionalidad que se puede obtener y el costo económico de la plataforma.

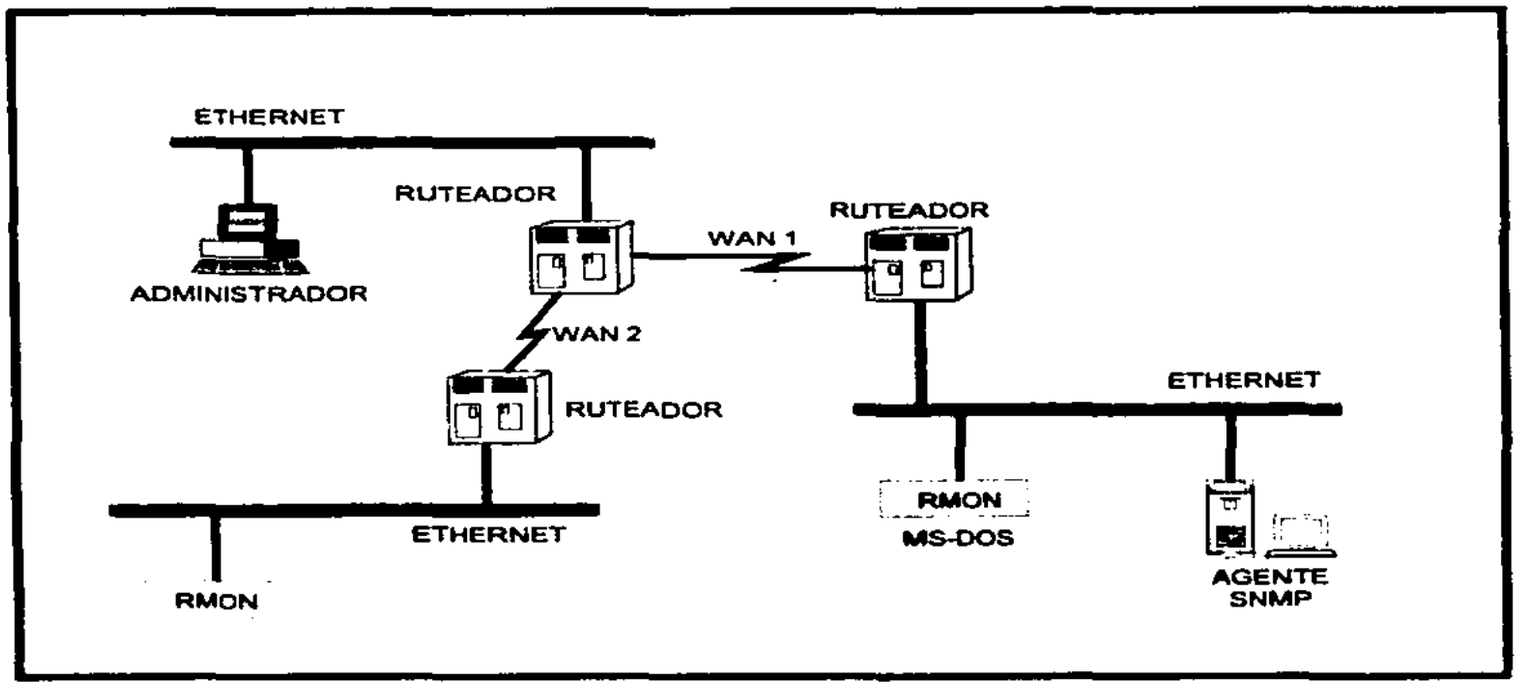


Figura 5.5 Monitoreo Remoto de LANs usando RMON

5.6.2 SNMP Versión 2.0

5.6.2.1 Introducción.

La primera versión de SNMP abrió el camino para un suceso muy importante en el área de la administración de redes. Tanto, que se ha llegado a convertir en un estándar *de facto*. Pero, como todo en esta área, y en muchas otras más, cada creación es susceptible de mejorar y actualizar conforme transcurren el tiempo y los adelantos tecnológicos. SNMP no es la excepción.

En el año de 1993, se publicaron once RFCs que están relacionados con la nueva versión de SNMP. En el RFC 1441 se proporciona una introducción a la versión 2 de SNMP. La figura 5.6 muestra un panorama general de las mejoras hechas en la nueva versión.

Las principales diferencias entre SNMPv1 y SNMPv2 son las siguientes:

- Se define un nuevo PDU para la recolección de grandes bloques de datos en forma eficiente, éste es el *get-bulk-request*

- Se define un nuevo PDU para la intercomunicación entre estaciones de administración de red, éste se conoce como *inform-request*
- Se definen dos nuevas MIBs: la MIB de SNMPv2 y la MIB M2M (*Manager-to-Manager*)
- Se especifican implantaciones de SNMP sobre otros protocolos de transporte
- Se adicionan nuevos elementos de seguridad en los mensajes con respecto a la primera versión.

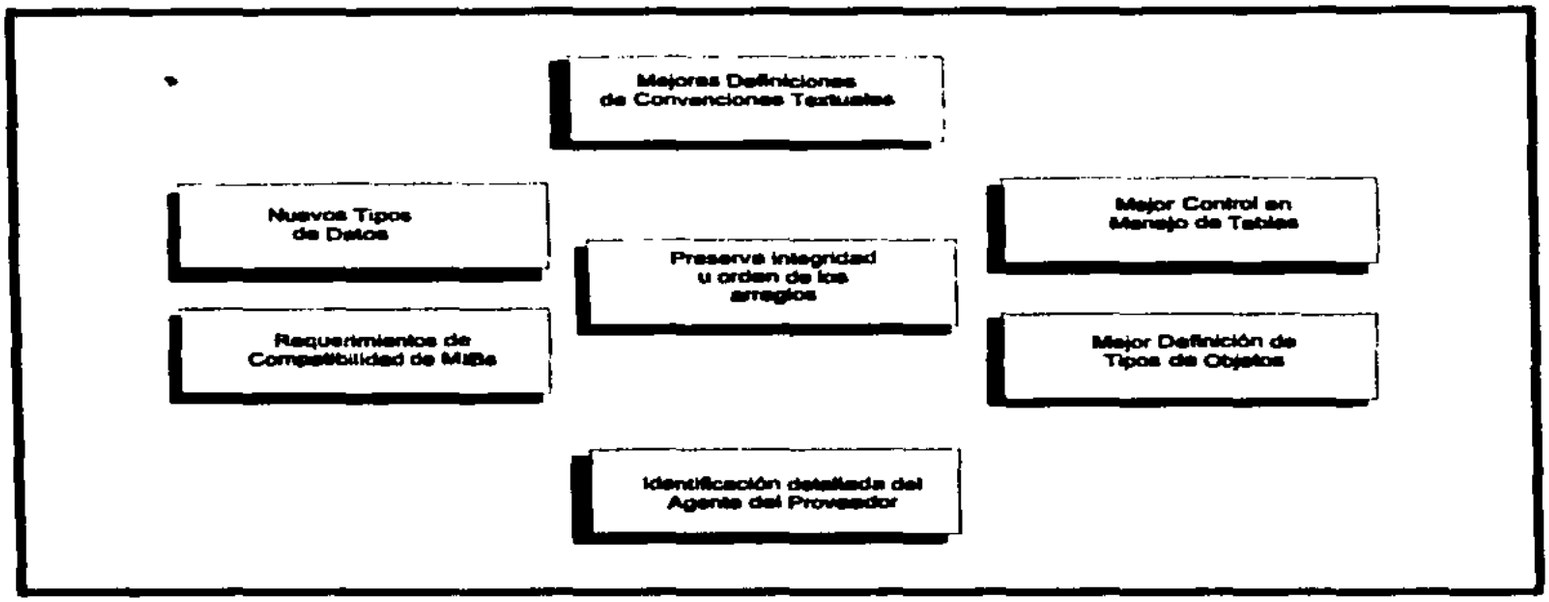


Figura 5.6 Panorama General de SNMPv2 y sus cambios

Los problemas más importantes que resuelve SNMPv2, están relacionados con la *seguridad de la información*, y estos son: la autentificación de los originadores de los mensajes, la protección contra violación de los mensajes, y los controles de acceso a las variables de las bases de datos MIB. Lo anterior implica el cambio en el formato de los mensajes SNMP. En la figura 5.7 se muestran los formatos utilizados en ambas versiones.

Al igual que en SNMPv1, los mensajes SNMPv2 contienen dos partes. La segunda parte del mensaje SNMPv2 es virtualmente idéntica a la del mensaje SNMPv1.

La primer parte del mensaje, también conocida como *wrapper*, contiene la mayoría de las diferencias que existen entre ambos formatos. Este campo incluye toda la información de autentificación y privacidad, además de un *contexto*. Un contexto especifica los objetos visibles para una operación dada.

Las especificaciones de SNMPv2 definen dos protocolos de seguridad: *Protocolo de Autentificación Clasificada* y *el Protocolo de Privacidad Simétrica*.

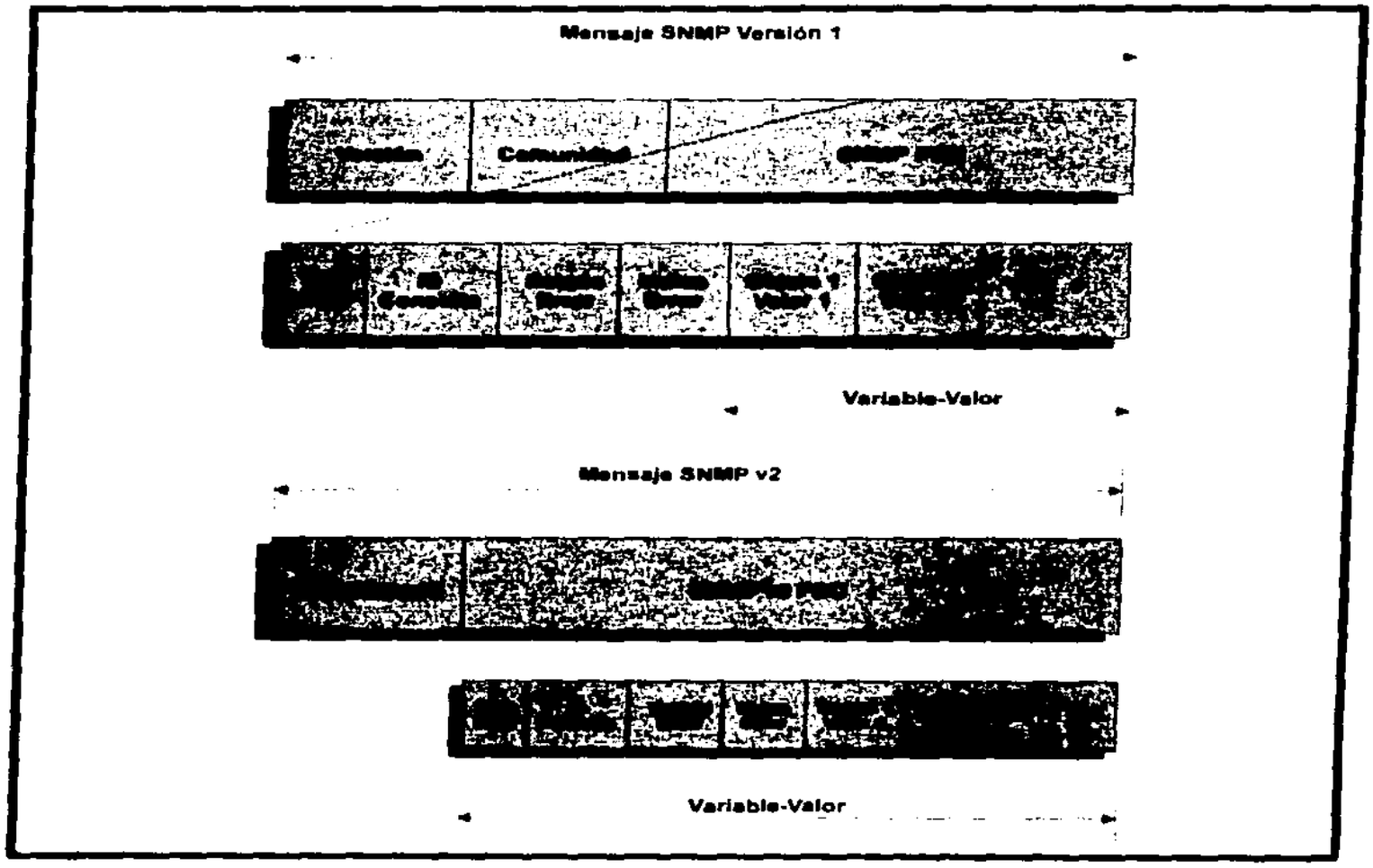


Figura 5.7 Formatos de Mensajes SNMPv1 y SNMPv2.

5.6.2.2 WINSOCK 2.0 y WinSNMP 2.0.

En la actualidad, todas las aplicaciones desarrolladas para administrar redes IP en el ambiente MS-Windows, tienen en común lo siguiente:

- Los protocolos de transporte son UDP/IP
- La API utilizada como interfase entre la aplicación y la implantación de TCP/IP (desarrrollada por diversos proveedores) es el estándar WINSOCK 1.1
- La API utilizada para desarrollar la aplicación de administración es WinSNMP 1.1

como los bloques de construcción básicos para la comunicación entre aplicaciones. (Básicamente, un socket es un punto terminal establecido por una aplicación de tal manera que puede *conectarse* a otro en la misma o en otra específica).

WinSNMP 2.0 contiene estas funciones que implantan los sockets, pero además agrega otras funcionalidades. Primero, la independencia del protocolo de transporte, nos permite utilizar esta interfase con otros protocolos como pueden ser SPX/IPX, Appletalk, DecNet, etc. Segundo, soporta múltiples mecanismos de resolución de servicios y nombres. Tercero, mejora el rendimiento de las aplicaciones, y soporta una gran variedad de servicios adicionales (comunicaciones multipunto y *multicast*, y mejoras en la seguridad).

Al igual que con WINSOCK, WinSNMP 2.0 ofrece nuevas funcionalidades y mejoras al desarrollador de aplicaciones de administración de redes. Se puede decir que en al versión 2 de WinSNMP, se tiene la posibilidad de administrar no sólo redes IP sino otras que se basen en otros protocolos. Además, ofrece mejoras en los aspectos relacionados con la programación y manejo de memoria de las aplicaciones. Básicamente, en esta nueva versión se implantan las nuevas características definidas en los RFCs relativos a SNMPv2.

Estas novedades realmente son muy importantes para el futuro desarrollo de la tecnología de la administración de redes, puesto que, en la actualidad no se conciben los sistemas aislados, es decir, tencnologías propietarias aparte de los sistemas abiertos y viceversa, realmente estamos inmersos en la era de la integración tanto de servicios como de plataformas de implantación de sistemas.

CONCLUSIONES

Una vez terminado este proyecto, hemos llegado a diversas conclusiones sobre cada uno de los aspectos que componen o enmarcan el sistema desarrollado.

Administración de Redes.

En los últimos años ha cobrado gran auge la Administración de la Tecnología de la Información (*IT Management*), la cual se define como la aplicación de diversas metodologías y tecnologías con el fin de asegurar el uso y control adecuados de la información. La Administración de Redes es parte medular en ente campo.

Este proyecto de tesis está inmerso en este tópico, ya que su objetivo principal es desarrollar un Sistema de Administración para Redes IP.

Es conveniente recalcar que la Administración de Redes se compone de varias áreas : Planeación, Organización, Monitoreo y Control de actividades y recursos. El trabajo que se ha realizado se ha enfocado al Monitoreo y Control de actividad y de recursos, pero no serviría de nada para el administrador de redes toda la información obtenida con esta herramienta, si no se llevan a cabo una planeación y organización programada de las redes de datos.

SNMP.

En la literatura de Administración de Redes y en los grupos de discusión de Internet, existen diversos puntos de vista sobre las posibles ventajas y desventajas que presenta el protocolo SNMP. Con el estudio y utilización que hemos hecho de SNMP podemos, bajo nuestra perspectiva, emitir varios comentarios al respecto.

La mayor ventaja de SNMP sobre otras alternativas (por ejemplo *CMIP, Common Information Monitoring Protocol*) es que ha llegado a ser un estándar *de facto* en la industria. Existen agentes SNMP disponibles para equipos de red que van desde computadoras, modems, hasta impresoras. El hecho de que exista tal soporte para SNMP, le da razón de existir. SNMP ha llegado a ser interoperable, a través de diversos dispositivos y varios fabricantes.

Además, SNMP es un protocolo de administración flexible y extendible. Debido que se puede hacer que los agentes SNMP manejen datos específicos de un equipo, y porque existe un mecanismo definido para lograr que los programas administradores interactúen con las capacidades especiales de los agentes (utilizando archivos ASN.1), SNMP puede realizar actividades específicas en impresoras, ruteadores, puentes, etc., proporcionando así un mecanismo estándar de monitoreo y control para las redes.

SNMP, presenta algunas debilidades. A pesar de su nombre (Protocolo *Sencillo* de Administración de Redes), SNMP no es tan sencillo de implantar como lo afirman quienes lo desarrollaron, y de hecho ya ha sido propuesto otro nombre para éste : MNMP o Protocolo *Moderado* de Administración de Redes. También, no es un protocolo particularmente muy eficiente, ya que se pierde *ancho de banda* con la transmisión de información que no es muy necesaria, como por ejemplo la versión SNMP. La manera en que se identifican las variables SNMP (como cadenas de *bytes*, donde cada uno de estos corresponde a cada nodo en la MIB) genera manejadores de datos innecesarios que consumen partes importantes del mensaje SNMP. Otra desventaja, aunque transparente para el usuario de los sistemas administradores, es la complejidad que tienen para implantarse las reglas de codificación o BERs.

En nuestra opinión, SNMP no debe ser visto sólo como una alternativa para las herramientas tradicionales de recolección de información de administración (tales como *ping, rsh, netstat*, etc.), sino que debemos tratarlo como una herramienta para analizar y administrar redes de datos de manera automática, de tal modo que podamos obtener datos de operación, configurar dispositivos y ejecutar procedimientos correctivos.

Realmente, el argumento más convincente que podemos dar con respecto a la utilización de SNMP es que en la actualidad no existen alternativas con las ventajas y facilidad que ofrece SNMP. Mientras que el protocolo por sí mismo puede ser menos que perfecto, SNMP proporciona la única forma para administrar redes de gran escala eficazmente (por ejemplo, Internet, o redes TCP/IP regionales).

Sistema de Monitoreo y Diagnóstico-97.

Con respecto al Sistema de Monitoreo y Diagnóstico desarrollado, podemos decir que se vieron cumplidos los objetivos planteados desde un principio, ya que se ha demostrado lo siguiente :

- SNMP es un protocolo realmente útil y funcional para administrar redes de datos, ya que mediante la aplicación básica desarrollada pudimos presentar información de administración de los agentes SNMP encontrados en las redes. Además, es posible extender la aplicación

mediante la expansión de la base de datos MIB y la adición de una cantidad de código no muy grande, para poder obtener datos de otros grupos de la MIB y de equipos específicos.

- Aunque SNMP por sí mismo no es óptimo, en lo que se refiere a su composición y estructura (trama del mensaje SNMP), pudimos constatar mediante la utilización de un *sniffer*, que SNMP no degrada en gran medida el rendimiento del canal *Ethernet*. Obviamente, se debe hacer un compromiso entre las políticas de poleo y retransmisión y los requerimientos que se definan para la obtención de los datos (por ejemplo, la periodicidad en la ejecución del monitoreo).
- La relación beneficio/costo obtenida es considerable ya que hemos conseguido las funcionalidades que requeríamos a un costo relativamente bajo. El costo es bajo, principalmente por las plataformas de desarrollo e implantación elegidas. Además, cabe aclarar que si no hubieran existido ya interfaces desarrollas, tales como *WinSnmp.dll y Winsock.dll*, este proyecto se hubiera demorada un tiempo mayor debido a que tendríamos que haber desarrollado una interfaz WinSnmp nosotros mismos, lo cual implica demasiado trabajo de programación.

APENDICE

A

REDES ETHERNET

La tecnología de redes de área local Ethernet, originalmente fué creada por Xerox, pero se convirtió en un estándar en 1980, cuando las compañías DEC, Intel y Xerox, unieron esfuerzos para formar lo que hoy conocemos como el estándar DIX Ethernet. El estándar 802.3 del IEEE define una red similar pero ligeramente diferente en el formato de la trama. Este último, ha sido adoptado por la ISO. En la figura A.1 se muestran ambas estructuras de trama.

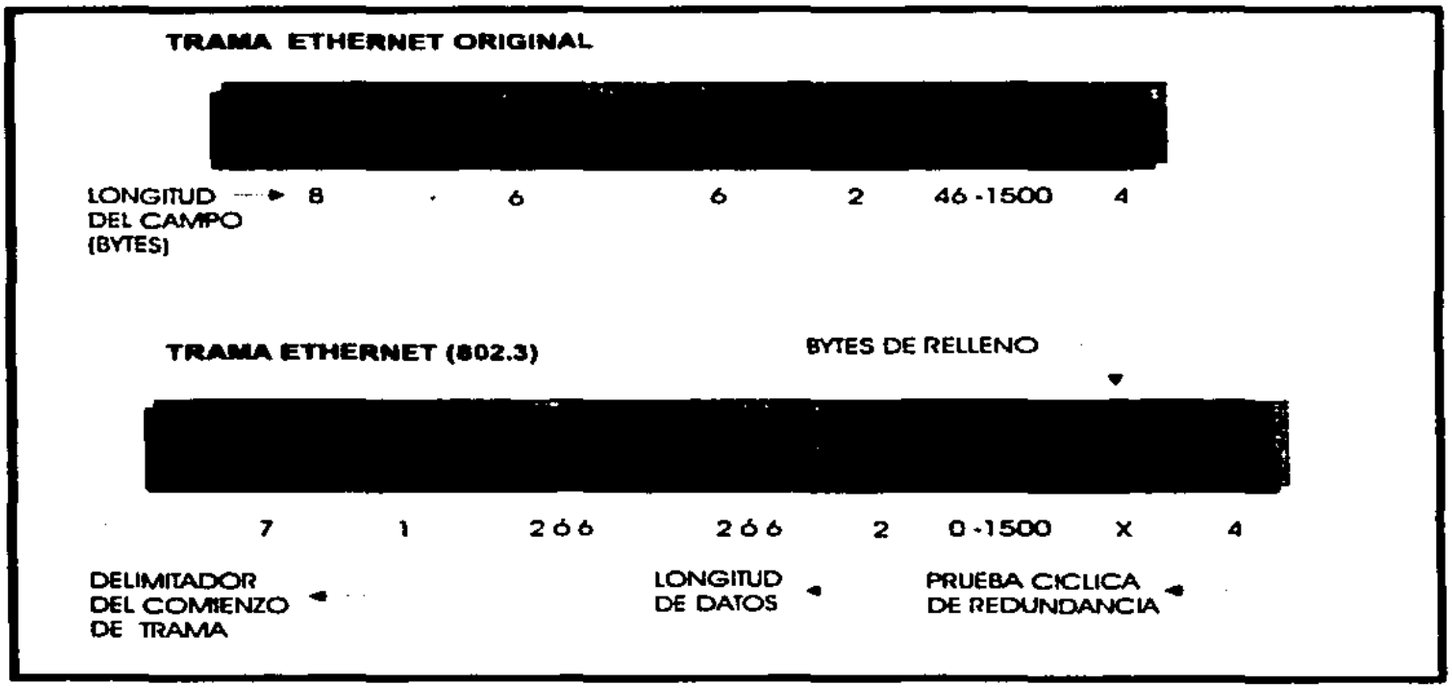


Figura A.1 Estructuras de tramas Ethernet

En Ethernet se maneja una velocidad de transmisión de 10 Mbps y se utiliza un método de acceso en el cual las estaciones de trabajo comparten un cable, pero sólo una de éstas puede usarlo en un instante dado. El método de

Múltiple Acceso con Sensado de Portadora y Detección de Colisiones (CSMA/CD) es utilizado para arbitrar el acceso al medio.

Los adaptadores Ethernet transmiten series de bits únicamente cuando tienen acceso exclusivo al medio. La detección de colisiones se refiere al método usado para resolver los accesos simultáneos al cable. Una colisión ocurre cuando dos estaciones tratan de transmitir datos en un mismo instante y esto da como resultado que la información se altere. El protocolo CSMA/CD utiliza un mecanismo que sensa la presencia de colisiones, y es en este momento cuando ambas estaciones dejan de transmitir por un periodo de tiempo aleatorio y reanudan su actividad cuando éste concluye.

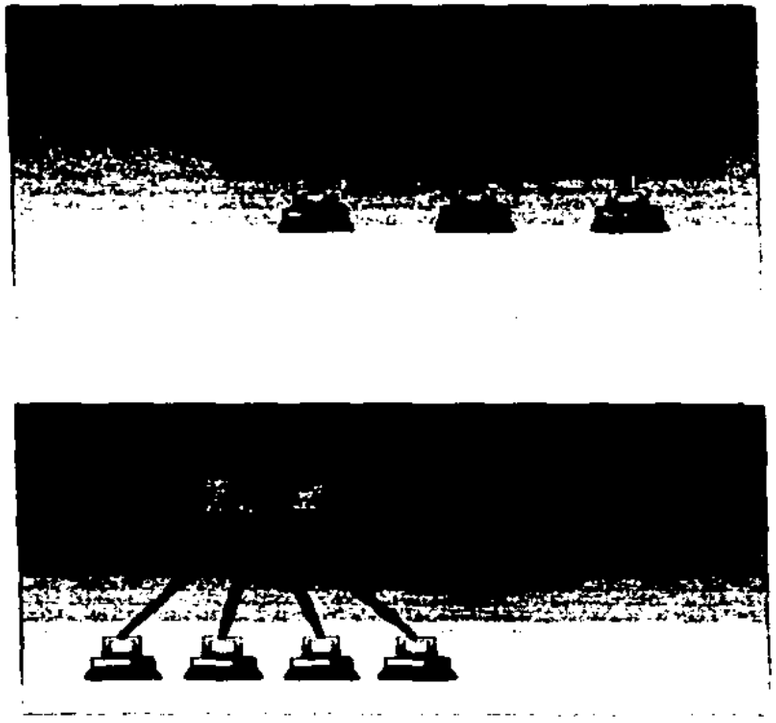


Figura A.2 Estándares Ethernet 10BaseT y 10Base2

Este método es eficiente únicamente cuando el tráfico de la red es ligero. Conforme el flujo de datos se incrementa, la ocurrencia de colisiones aumenta, provocando una degradación en el rendimiento de la red. El problema de las colisiones es un factor que impone ciertos límites en la capacidad de la

red. Estas limitantes son específicas para cada adaptación del estándar IEEE 802.3.

Los estándares más populares son 10Base2 y 10BaseT. La topología de la mayoría de las redes Ethernet es un segmento lineal con CSMA/CD como método de acceso. En implementaciones con cable coaxial delgado (10Base2) las estaciones de trabajo son conectadas en *cadena* (daisy-chan). Los segmentos de cable forman un gran sistema de cableado el cual se conoce como *troncal*. La versión de par-trenzado de Ethernet (10BaseT) es configurada como una topología en estrella en la cual el cable que va hacia cada estación se extiende desde un concentrador central (Hub). En la figura A.2 se muestran ambos estándares.

En la actualidad se han venido implantado variantes del estándar Ethernet original, con el fin de incrementar el rendimiento de las redes. Estos incluyen otros tipos de acceso al medio, mejor calidad en el cableado, etc. Los principales hasta el momento son Fast Ethernet, que originalmente fué creado por un grupo de compañías, para después convertirse en un estándar del IEEE que administra el comité 802.3 (se le conoce como 100BaseX). Y el estándar 100VG-AnyLAN el cual maneja el comité 802.12 del organismo mencionado.

APENDICE

B

TCP/IP

El grupo de protocolos TCP/IP permite que un gran número de computadoras de diversas capacidades, diferentes fabricantes, y aún con diversos sistemas operativos, puedan intercomunicarse de una manera ordenada y confiable. Su uso ha sobrepasado el estimado por sus creadores. Lo que inició en los finales de los años sesentas como un proyecto de investigación financiado por el gobierno de los Estados Unidos, se ha convertido en la tecnología de intercomunicación de redes más ampliamente usada en los años noventas. Realmente, es un *sistema abierto* ya que no está enfocado a una plataforma en particular, debido a que la documentación e implantaciones del mismo están disponibles al público en general a bajo o ningún costo.

TCP/IP forma la base de la Internet, la cual es una red de área amplia (WAN) de más de cinco millones de computadoras que literalmente cubren el mundo entero. Los protocolos de Internet, tal como se le conoce a este grupo, es la mejor opción que existe para interconectar la gran diversidad de tecnologías de redes LAN y WAN que existen.

Los protocolos de red normalmente se desarrollan en *capas*, asignando a cada una de éstas una responsabilidad en la intercomunicación. Un *grupo de protocolos* tal como TCP/IP, es la combinación de diferentes protocolos en varias capas. TCP/IP se estructura en cuatro capas, como lo muestra la figura B.1

Cada una de las cuatro capas tiene una función específica:

- La capa de *enlace*, también conocida como *interfaz de red*, normalmente contiene el controlador del dispositivo en el sistema operativo y la tarjeta de red en la computadora. En conjunto, ellos manejan todo lo relacionado con los detalles de hardware que se necesita, con el fin de implantar la interfaz con el medio que se esté utilizando.

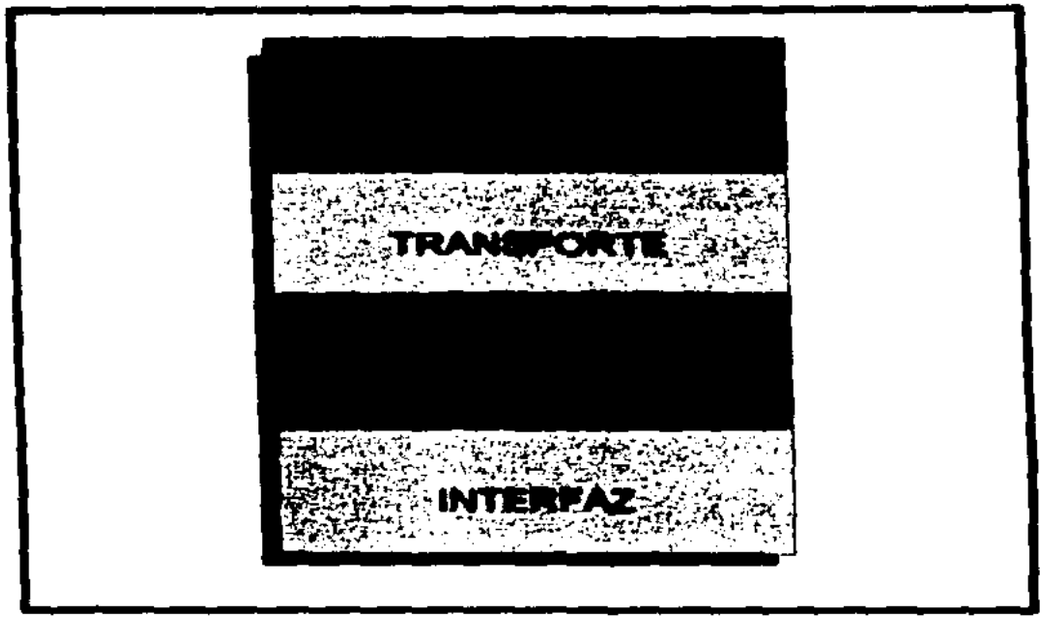


Figura B.1 Modelo TCP/IP.

- La capa de *red* (algunas veces llamada *interred*) maneja el movimiento de paquetes de datos a lo largo de la red. El enrutamiento de paquetes se lleva a cabo aquí. Otros protocolos ubicados en este estrato son ICMP e IGMP.

  El protocolo de interred (IP), además de encargarse del enrutamiento en las redes, proporciona reportes de errores, fragmentación y reensamble de las unidades de información (datagramas) para su transmisión sobre las redes que manejan distintos tamaños de MTU. IP representa el núcleo del grupo de protocolos de Internet, y es por esto que es común denominar a las redes que utilizan TCP/IP, como *redes IP*.

  Las direcciones IP son números de 32-bits globalmente únicos, los cuales son asignados por el Centro de Información de Internet (InterNIC). Estas direcciones asignadas en forma única en el mundo, permiten que las redes IP en cualquier parte se intercomuniquen sin problemas. Es común representar estas direcciones con cuatro números decimales separados por un punto (A.B.C.D).

  Una dirección IP se divide en tres partes. La primer parte es la dirección de la red, la segunda denota la dirección de subred, y la tercera nos da la dirección de un host en particular.

Este direccionamiento soporta cinco clases de redes, siendo las primeras tres las que se utilizan en gran medida, dejando las otras dos para uso reservado. En la figura B.2 se esquematizan las diferentes clases de redes IP.

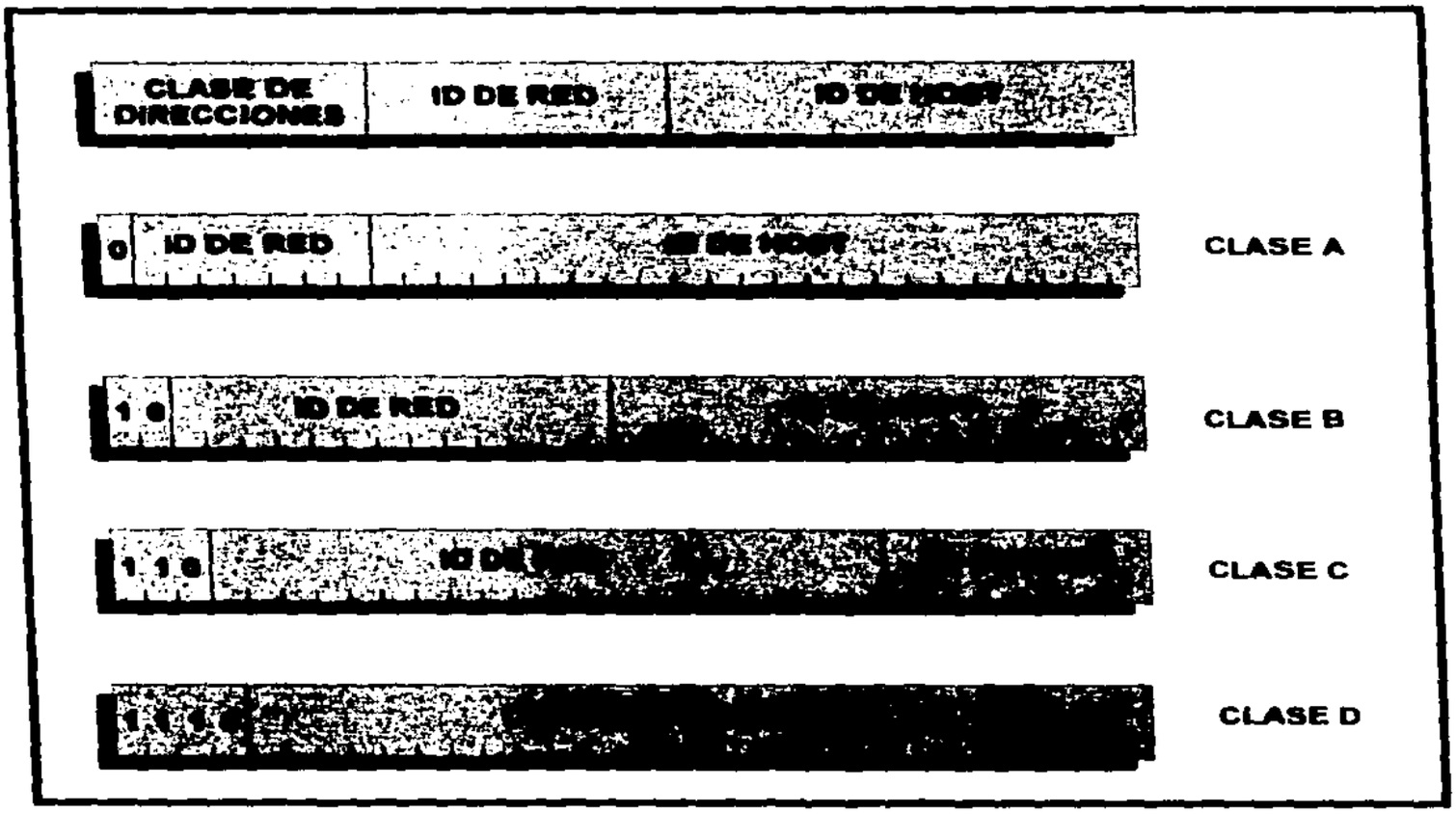


Figura B.2 Clases de Direcciones IP.

Las clase A están enfocadas a pocas redes que contienen muchos nodos, debido a que solo tienen 8 bits en el campo de identificación de red.

Las clase B, tienen 16 bits en este campo y las clase C, cuentan con 24 bits. En los tres casos el o los bits que están más a la izquierda nos indican el tipo de red que estamos utilizando. En la figura B.3 se muestra una tabla con los rangos de direcciones para cada una de las clases de red.

- La capa de *transporte* proporciona un flujo de datos entre dos hosts o equipos terminales, ofreciendo así, soporte a la capa de aplicación. En el

grupo TCP/IP existen dos protocolos de transporte: TCP (Protocolo de Control de la Transmisión) y UDP (Protocolo de Datagramas del Usuario).

TCP proporciona un flujo de datos confiable entre hosts. En este se tratan aspectos tales como el correcto dimensionamiento de los paquetes de datos para transportarlos a la capa de red, el establecimiento de *timeouts* para llevar un control de los paquetes enviados y de los recibidos,etc.

UDP, por otra parte, ofrece un servicio mucho más simple a la capa de aplicación. Solamente envía paquetes de datos llamados *datagramas* desde un host a otro pero sin la garantía de que estos llegarán a su destino. En este caso, la capa de aplicación es la encargada de implantar la confiabilidad en la transmisión de la información.

Existen diversos usos para cada tipo de protocolo de transporte. Por ejemplo, la transferencia de archivos requiere de TCP, mientras que, a primera instancia, el protocolo SNMP utiliza UDP.

- La capa de *aplicación* maneja los detalles de una aplicación en particular. Existen muchas aplicaciones de TCP/IP que la mayoría de las implantaciones ofrecen:

    1. Telnet, para acceso remoto a hosts.
    2. FTP, para transferencia de archivos.
    3. SMTP, para correo electrónico.
    4. SNMP, para administración de redes.

APENDICE

C

ASN.1

La idea de proporcionar una notación para definir estructuras de datos, de establecer reglas de codificación para tales estructuras que fueran independientes de la arquitectura de las computadoras (y de la representación física de los datos), y herramientas que llevaran a cabo tales codificaciones, es atribuida a la Especificación Courier Xerox, que forma parte del grupo de protocolos XNS.

A principios de los años ochentas un comité del CCITT se basó en la especificación antes citada, y creó la recomendación X.409 que formaba parte de la serie X.400 (correo electrónico). Posteriormente ISO adoptó esta notación como la base para definir y escribir los protocolos de la capa de aplicación del modelo OSI, y se crearon dos documentos que se derivaban del original X.409 del CCITT. El primero, ISO 8824 se conoce como Notación de Sintaxis Abstracta número UNO (ASN.1), y el segundo, ISO 8825 se llama Reglas de Codificación Básicas (BER). En las recomendaciones de 1988, el CCITT consideró la importancia que estaba tomando esta notación y asignó dos nuevas recomendaciones X.208 (ASN.1) y X.209 (BER).

Usando ASN.1, un desarrollador de estándares puede definir tipos de datos *primarios* simples, tipos de datos *compuestos*, y formatos de mensajes completos. Las definiciones escritas en lenguaje ASN.1 son muy fáciles de leer (las definiciones de las MIB se escriben en este lenguaje).

TIPOS DE DATOS PRIMARIOS ASN.1

Los tipos de datos primarios de ASN.1 son los siguientes:

- INTEGER. Un número entero.
- ENUMERATED. Especifica un grupo limitado de enteros, y cada uno tiene uasignado un significado, tal como *rojo(1), blanco(2),* y *azul(3).*

- OCTET STRING. Representa una cadena de octetos. Cuando un octeto se representa en forma decimal, se encuentra en el rango 0-255.
- OBJECT IDENTIFIER. Es una cadena de números enteros que se obtiene árbol jerárquico de nombres, y es usado para identificar un objeto.
- NULL. Un valor nulo.
- BOOLEAN. Toma el valor cierto o falso.
- BIT STRING. Utilizado para el manejo de *banderas*, en las cuales el valor de un bit es significativo.
- REAL. Es utilizado para expresar un número real (mantisa, base y exponente).

TIPOS DE DATOS COMPUESTOS ASN.1

Las estructuras compuestas pueden ser definidas combinando los tipos de datos primarios, utilizando los *constructores.*

- SEQUENCE. Es una lista ordenada de tipos de datos distintos.
- SEQUENCE OF. Es una lista ordenada pero cada objeto es del mismo tipo.
- SET. Es una lista desordenada de tipos de datos.
- SET OF. Es una lista desordenada, del mismo tipo de datos.
- CHOICE. Es una elección de una selección de tipo de datos.

En la versión 1 de SNMP sólo se utiliza un subconjunto de los tipos de datos vistos anteriormente, ya que se consideró que entre menos existieran, los mensajes serían más simples y esto implica desarrollar menos *software.*

TIPOS DE DATOS EN SNMP v1

- INTEGER. Representa medidas numéricas, tales como el número de interfaces en un sistema. Además se usa para enumeraciones como *up(1), down(2), testing(3).*

- OCTET STRING. Se usa para representar datos en hexadecimal, tales como, la dirección física de una interface. También para cadenas de texto.

- OBJECT IDENTIFIER. Es utilizado para nombrar un objeto administrado por medio de una cadena de números.

- NULL. Es un valor nulo. Por ejemplo, una operación *get-request* contiene una lista de OBJECT IDENTIFIERS de variables, cada uno ligado con un NULL en el campo del valor.

Para SNMP, se agregaron otros tipos de datos primarios como son: *Counter, TimeTicks, Gage, IpAddress, NetworkAddress, y Opaque*.

Los únicos tipos de constructores permitidos, en SNMP v1, son SEQUENCE y SEQUENCE OF. Por ejemplo, una tabla es una SEQUENCE OF SEQUENCES.

ELABORACION DE MACROS

Una plantilla *macro* puede ser diseñada de tal manera que represente cualquier objeto que se necesite definir. Por ejemplo, es necesario agrupar cierta información en la definición de un objeto MIB, tal como:

1. El OBJECT IDENTIFIER que se usará para identificar el objeto MIB.
2. El tipo de dato del objeto.
3. Si es necesario definir un *rango* de valores para el objeto.
4. Si se reestringirán las *operaciones* que pueden llevarse a cabo en el objeto.
5. Una *descripción textual* que se pueda necesitar para implantar este objeto.

Por ejemplo,

```
ipAdEntReasmMaxSize OBJECT-TYPE
    SYNTAX INTEGER (0..65535)
    ACCESS read-only
    STATUS mandatory
    DESCRIPTION
            "El tamaño del datagrama IP más grande que esta entidad
             puede re-ensamblar dados los datagramas IP fragmentados
             que se recibieron en esta interface."
    ::= { ipAddrEntry5}
```

Esta definición denota lo siguiente:

6. El **OID** para *ipAdEntReasmMaxSize* es { *ipAddrEntry* 5}, el cual puede obtenerse siguiendo la trayectoria respectiva en el árbol y es 1.3.6.1.2.1.4.20.1.5.
7. La **SYNTAX**, o tipo de dato, para el valor de la variable es **INTEGER**.
8. El *integer* está limitado al rango de 0-65535.
9. El **ACCESS** nos dice que la estación de administración sólo puede leer este dato, pero no actualizarlo.
10. El **STATUS** nos dice que esta variable debe incluirse siempre en cualquier implantación.
11. La **DESCRIPTION** nos define que el valor de esta variable es el tamaño del datagrama más grande que puede ser re-ensamblado de los fragmentos en la interface.

SINTAXIS DE TRANSFERENCIA

Los lenguajes de programación de alto nivel deben de ser traducidos (compilados) en un formato de máquina antes de que el programa pueda ser ejecutado. En forma similar, las instrucciones ASN.1 tienen que ser traducidas en un flujo serial de bytes antes de que puedan ser transmitidas en la red.

Estas reglas de compilación son conocidas como SINTAXIS de TRANSFERENCIA. Los creadores de ASN.1 especificaron una sintaxis que se conoce como *Basic Encoding Rules* o BER. Existe la posibilidad de desarrollar otro tipo de sintaxis de transferencia, como por ejemplo, una en la que se incluya encriptamiento de la información. Otras reglas de codificación desarrolladas en la actualidad son PER, LWER,etc.

Las BER se basan en un esquema en el cual cada *campo* de un PDU es *autodefinible*. Un campo tiene un *presentador* que nos dice qué es y qué tan largo es éste. El patrón básico utilizado para codificar un valor es:

[identificador] [longitud (del contenido)] * [contenido]

El identificador declara el tipo de datos del contenido. El contenido puede ser uno de los tipos primarios de datos (p.ej. un *integer*) o uno de los compuestos (p.ej. una *SEQUENCE OF* valores). Los mensajes SNMP se construyen con este último tipo de datos.

En las BER, los tipos de datos se dividen en cuatro clases:

- *Universal*. Se usa en cualquier protocolo. Los tipos primarios y los compuestos son universales.

- *Application*. Es para una aplicación específica. Por ejemplo, *IpAddress* es específico para TCP/IP.

- *Context-specific*. Este es contenido en un tipo de datos más grande.

- *Private*. Se usa para definir tipos de datos privados.

El identificador es un byte, donde los dos primeros bits, de izquierda a derecha, representan una de las cuatro clases anteriores, es decir,

Universal	00
Application	01
Context-specific	10
Private	11

El tercer bit nos indica lo siguiente,

Primario	0
Compuesto	1

Los restantes cinco bits, nos representan un *tag* que se asocia con un tipo de datos dado.

Por ejemplo, para codificar el tipo de datos *IpAddress*, hacemos lo siguiente:
La definición en la MIB es,

```
IpAddress::=[APPLICATION 0]
        IMPLICIT OCTET STRING (SIZE(4))
```

Este es un tipo de datos simple, por lo tanto es *primario*. [APPLICATION 0] nos dice que su clase es *APPLICATION* y su *tag* es 0. Entonces, el identificador es 0100 0000 o en hexadecimal, '40'H. De acuerdo a la MIB, la longitud del contenido *tiene que* ser 4. El contenido consiste de los cuatro bytes de la dirección IP. La codificación hexadecimal de la dirección IP 128.1.1.1 es:

Identificador	Longitud	Contenido
40	04	80 01 01 01

GLOSARIO

AGENTE. Generalmente, software que procesa consultas y regresa respuestas a una aplicación. En sistemas de administración de redes, los agentes residen en todos los dispositivos administrados y reportan los valores de las variables específicas a las estaciones de administración.

AGENTE PROXY. Es una software de agente que media entre una estación de administración de red y un equipo de red que no soporta SNMP, es decir, el proxy traduce las consultas del administrador a instrucciones adecuadas que el equipo en cuestión puede entender.

API. Es un grupo definido de llamadas a procedimientos, tipos de datos, estructuras de datos, y la semántica asociada para incorporar un servicio lógicamente externo en una aplicación.

ASN.1. NOTACION DE SINTAXIS ABSTRACTA 1, es un lenguaje utilizado para definir tipos de datos. Es utilizado en los estándares OSI y en las especificaciones de administración de red de TCP/IP.

ATM. MODO DE TRANSFERENCIA ASINCRONA, es una tecnología de empaquetamiento y conmutación de información que maneja celdas de longitud fija de 53 bytes. Será utilizada en redes locales y redes de área amplia.

BER. REGLAS DE CODIFICACION BASICAS, es un conjunto de reglas utilizadas para traducir instrucciones ASN.1 en un serie de bytes para transmitirlos a través de la red.

BRIDGE. PUENTE, es un dispositivo que conecta y pasa tramas entre dos redes físicamente separadas. Operan en el nivel 2 del modelo OSI (enlace de datos).

CLIENTE/SERVIDOR. Término utilizado para describir los sistemas en los cuales las responsabilidades de las transacciones se dividen en dos partes: cliente y servidor. Ambos términos pueden ser aplicados tanto a programas de software o a dispositivos físicos.

CMIP. Protocolo OSI de administración de redes creado para administrar redes heterogeneas.

CMOT. CMIP sobre TCP. Es el uso de CMIP sobre el grupo de protocolos TCP/IP.

COMUNIDAD. En SNMP, es un grupo lógico de dispositivos administrados y NMSs dentro de un mismo dominio administrativo.

CSMA/CD. Es un mecanismo de acceso al medio donde los dispositivos que desean transmitir primero verifican si ya está transmitiendo alguien. Si no existe señal portadora por algún periodo de tiempo, los dispositivos pueden transmitir. Si dos dispositivos transmiten al mismo tiempo ocurre una *colisión* y es detectada por todos los dispositivos involucrados y estos esperan un tiempo aleatorio para retransmitir. Esta técnica es utilizada por Ethernet y IEEE 802.3.

CSU. Unidad de Servicio del Canal. Es un dispositivo de interfase digital utilizado para conectar el equipo del usuario final con el equipo del prestador del servicio digital.

DATAGRAMA. Es un agrupamiento lógico de información que se envía como una unidad de la capa de red sobre un medio de transmisión sin el previo establecimiento de un circuito virtual. Los dategramas IP son las unidades de información primaria en la Internet.

DSU. Unidad de Servicio de Datos. Es un dispositivo usado en la transmisión digital para conectar una CSU a un DTE.

DTE. Equipo Terminal de Datos.Equipo por lo general ubicad en la infraestructura del usuario final.

ENTIDAD. Es un proceso SNMP, operando ya sea en el rol de agente o administrador, o ambos, el cual lleva a cabo operaciones de administración de red por medio de la generación o respuesta de mensajes SNMP.

ETHERNET. Es un estándar de LAN en banda base inventado por XEROX y desarrollado en conjunto por Xerox, Intel, y DEC. Estas redes operan a una velocidad de 10 Mbps usando CSMA/CD. Es similar a lo serie de estándares producidos por el IEEE, y se conocen como IEEE 802.3.

FRAME. TRAMA, Es un agrupamiento lógico de información que se envía como una unidad de la capa de enlace de datos sobre un medio de transmisión.

FRAME-RELAY. RELE de TRAMAS, es un protocolo utilizado a través de la interface entre dispositivos de usuario (por ejemplo, hosts y ruteadores) y equipo de red (por ejemplo, nodos de conmutación). Es mas eficiente que X.25, y es considerado como su reemplazo.

GATEWAY. En la comunidad IP, es un término que se usaba para describir a un equipo de ruteo. En la actualidad, el término ruteador es usado para describir nodos que realizan esta función, y un gateway se refiere a un dispositivo de propósito especial que realiza la conversión de protocolos de nivel 7 del modelo OSI.

HOST. Es un sistema de cómputo en una red. Es similar a los términos *dispositivo* o *nodo* excepto que *host* usualmente implica un sistema de cómputo, mientras que los otros se refieren a equipo de redes.

IAB. Internet Activities Board. Es un grupo de investigadores en el área de interconexión de redes que se reúne periódicamente para analizar cuestiones relacionadas con la Internet.

IANA. Internet Assigned Numbers Authority (IANA), es la autoridad responsable de controlar la asignación de una variedad de parámetros, tales como, los puertos TCP "bien-conocidos", direcciones multicast, identificadores de sistemas, etc.

IEEE-802.3. Es un protocolo IEEE de LAN que especifica una implantación de la capa física y la subcapa de Control de Acceso al Medio de la capa de enlace de datos. IEEE 802.3 usa CSMA/CD a una variedad de velocidades sobre varios tipos de medios físicos.

IETF. Internet Engineering Task Force, es un grupo que dirige la IAB que se encarga de resolver problemas a corto plazo de la Internet.

internet. interred, es la interconexión de redes de comunicaciones de datos.

Internet. Es el término utilizado para referirse a la interred más grande del mundo, y la cual conecta miles de redes en todo el mundo y tiene la "cultura" que se basa en la simplicidad, investigación, y la estandarización basada en su uso en la "vida real".

IP. Protocolo de Interred, es un protocolo de nivel 3 que contiene información de direccionamiento y alguna información de control que permite que sean ruteados paquetes de información.. Está documentado en el RFC 791.

ISO. International Organization for Standerdization, es un organismo internacional que es responsable de definir una gran variedad de estándares, incluyendo los relacionados con la interconexión de redes. Este organismo desarrolló el modelo de referencia OSI.

IT-MANAGEMENT. Es la unión de Metodologías y Tecnologías que se aplica a los procesos de manejo de la información, con el fin de manterlos en un nivel de calidad y control adecuados.

LAN. Red de Area Local, es una red que cubre un área geográfica relativamente pequeña (usualmente no más grande que un pequeño campus de edificios).

LanSWITCH. Es un dispositivo de red de área local utilizado principalmente para microsegmentar redes y de esta manera asignar, por ejemplo en un red Ethernet,un enlace "completo" de 10 Mbps a un Servidor muy concurrido. En la mayoría de las veces trabaja a nivel 2 del modelo OSI.

MENSAJE. Es un agrupamiento lógico de información en el nivel de aplicación.

MIB. Management Information Base, es una base de datos de información de objetos administrados que puede ser accesada por protocolos como SNMP y CMIP.

MIB-VIEW. Es una muestra o subconjunto de todas las instancias de todos los objetos MIB.

MODEM. MODulador-DEModulador, es un dispositivo que convierte señales digitales en un formato adecuado para ser transmitidas sobre un medio de transmisión analógico, y viceversa.

MULTIPLEXOR. Es un dispositivo que sirve para manejar o colocar múltiples señales en un sólo canal de transmisión.

NIC. Network Interfase Card, es una tarjeta que contiene los dispositivos y circuitos necesarios para conectar un equipo de cómputo o comunicaciones a una LAN. Generalmente, esta tarjeta implanta los niveles 1 y 2.

NMS. Network Management Station, es la estación de administración de red y es responsable del manejo de por lo menos una parte de toda la red. Generalmente, es un equipo de cómputo poderoso y bien equipado, es decir, con gran cantidad de memoria, espacio en disco, y monitor de gran resolución.

NODO. Término genérico utilizado para referirse a una entidad que puede tener acceso a una red. Es usado en forma intercambiable con *dispositivo*.

OID. Object Identifier, identificador de objeto de una MIB. Es una cadena de enteros separados por puntos que nos identifica a un objeto nombrado por una autoridad administrativa.

OSI. Open Systems Interconnection, es el modelo de referencia de comunicaciones de datos definido por la ISO y el CCITT para desarrollar estándares para la interconexión de redes, y de esta manera facilitar la interoperabilidad entre equipos de múltiples proveedores.

OUT-OF-BAND. Es un término utilizado muy frecuentemente para denotar la transmisión de datos en otro canal que no sea el principal del tráfico de la información. Muy usado en el área de monitoreo y administración de redes.

PAQUETE. Es un agrupamiento lógico de la información que incluye un encabezado y datos propios del usuario. Actualmente, es una unidad de datos en cualquier capa de procesamiento.

PDU. Protocol Data Unit, es otro término para la unidad de datos (encabezado y datos del usuario) definido por OSI. Un PDU es intercambiado entre dispositivos dentro de un nivel específico del modelo OSI.

PROTOCOLO. Es una descripción formal de un conjunto de reglas y convenciones que gobiernan el intercambio de información entre dispositivos en una red.

RED. Es un grupo de dispositivos y computadoras que se crea para intercambiar información y recursos.

RFC. Request For Comment, son documentos usados como el medio principal para transmitir información relativa a la Internet. La mayoría de los RFCs documentan protocolos de comunicaciones tales como IP, FTP, TELNET, SNMP, etc.

RUTEADOR. Es un dispositivo de nivel 3 de OSI, que tiene la capacidad de decidir cual de varias trayectorias existentes para llegar a un destino determindo es la óptima, basado en ciertos parámetros predefinidos.

SGMP. Simple Gateway Monitoring Protocol, Es el protocolo antecesor de SNMP en lo que se refiere a administración de dispositivos, pero sólo contemplaba el manejo de gateways en la terminología de Internet.

SMI. Estructura de la Información de Administración, es un documento (RFC 1155) que especifica las reglas usadas para definir objetos administrados en la MIB.

SNMP. Protocolo Sencillo de Administración de Redes, es un protocolo asíncrono de consulta/respuesta que sirve para intercambiar información de administración a través de una red.

SOCKET. Es un paradigma de comunicaciones creado por la Universidad de Berkeley, el cual define puntos terminales de comunicaciones entre entidades de red. Un socket está formado por la unión de una direccion IP y un número de puerto TCP o UDP.

TCP. Protocolo de Control de la Transmisión, que se ubica en la capa de Transporte de TCP/IP y el cual proporciona una manera confiable de transmisión de datos. Es un protocolo orientado a la conexión.

TRAP. Es un mensaje "no solicitado" enviado por un agente SNMP a una estación de administración de red y el cual indica la ocurrencia de un evento determinado.

UDP. Protocolo de Datagramas del Usuario, que se ubica en la capa de Transporte de TCP/IP y el cual
proporciona una forma no-confiable de transmisión de datos. Es un protocolo sin conexiones.

WAN. Red de Area Amplia, es una red de comunicaciones que cubre un área geográfica relativamente extensa.

BIBLIOGRAFIA

1. BLACK, UYLESS D. TCP/IP AND RELATED PROTOCOLS. McGRAW-HILL 1992.

2. COMER, DOUGLAS E., AND STEVENS, DAVID L. INTERNETWORKING WITH TCP/IP-VOLUME 11: DESIGN, IMPLEMENTATION, AND INTERNALS. PRENTICE-HALL, 1991.

3. FEIT, SIDNIE. SNMP: A GUIDE TO NETWORK MANAGEMENT. McGRAW-HILL, 1994.

4. NATALE, BOB. WinSNMP/MANAGER API v1.1, 1994.

5. PERKINS, DAVID T. "UNDERSTANDING SNMP MIBS". Rev. 1.1.6, Sept., 1993.

6. RFCs. SNMPv1: 1089, 1155, 1156, 1157, 1213, 1215, 1284, 1303, 1351, 1352.
   SNMPv2 : 1441, 1442, 1443, 1444, 1445, 1446, 1447, 1448, 1449, 1450, 1451, 1452.

7. ROSE, MARSHALL T. AND McCLOGHRIE, KEITH Z. HOW TO MANAGE YOUR NETWORK USIN SNMP: THE NETWORK MANAGEMENT PRACTICUM. PRENTICE-HALL , 1990.

8. ROSE, MARSHALL T. THE SIMPLE BOOK: AN INTRODUCTION TO MANAGEMENT OF TCP/IP-BASED NETWORKS. PRENTICE-HALL, 1990.

9. STALLINGS, WILLIAM. SNMP, SNMPv2, Y CMIP: THE PRACTICAL GUIDE TO NETWORK MANAGEMENT STANDARDS. ADDISON-WESLEY, 1993.

10. STEVENS, W. RICHARD. TCP/IP ILUSTRATED-VOLUME 1: THE PROTOCOLS. ADDISON-WESLEY, 1994.