74

UNIVERSIDAD NACIONAL AUTONOMA DE MEXICO

FACULTAD DE INGENIERIA

SISTEMA DE MONITOREO Y ADMINISTRACION DE LA RED DEL INSTITUTO DE INGENIERIA

T E S I S

PARA OBTENER EL TITULO DE:

INGENIERO EN COMPUTACION

P R E S E N T A :

CLAUDIA CECILIA RAMIREZ CASTRO

DIRECTOR: ING. MARCO AMBRIZ MAGUEY.

MEXICO, D. F.

ENERO DEL 2000.

Perseverance will prevail where all others will fail
-Anónimo-

Agradecimientos

Quisiera agradecer en primer lugar a la **Universidad Nacional Autónoma de México**, por ser la Institución que me ha forjado como profesionista y ha contribuido de manera significativa en mi desarrollo personal. Hoy por hoy para mi es un gran orgullo ser Universitaria.

Al **Instituto de Ingeniería** de la **UNAM**, por las grandes oportunidades de desarrollo profesional que me brindó.

En especial al **Ing. Marco Ambriz Maguey**, por todo su apoyo, tiempo y por su infinita paciencia.

Al **Ing. Palacios** por su apoyo e Interés.

A **Dios** por fortalecer mi espíritu.

Con todo mi cariño a mi **Mamá** por todo su amor, paciencia, esfuerzo y apoyo. Por ser un ejemplo de dedicación y entrega.

Con todo mi cariño a **Eva** por su amor y apoyo incondicional en todo momento. Por enseñarme que "todas para una y una para todas".

Con todo mi cariño para **Conchis** por todo su cariño y gran apoyo, por ser parte de mi gran familia.

A **Federico** por que la luz de tu amor siempre a guiado mi camino, por siempre creer en mi.

A **Zairis** por todo tu apoyo y por ponerle sabor a la vida.

A mis incomparables **amigos**: Gustavo, Ricardo, Fernando, Rodrigo, Lissandra, Claudia Yi, Sra. Berta Gaytan, Gabo. Gracias por vivir como si fueran suyos todos mis éxitos y mis fracasos.

A **Mariana Guati**, ya que sin tu apoyo incondicional esta tesis no hubiera podido terminarse nunca. Gracias.

A mis **compañeritos y amigos** del II, Elena, Lalo García, Guendaviani, Alejandro G., Edgar G., Adán, Gus "Clon" y todas las personas que me apoyaron siempre.

A **Miguel Ángel Bañuelos**, por ser más que un profesor un amigo.

A **Cris Casimiro** por ser un gran ejemplo.

Contenido

Capítulo 4
Alternativas y evaluación de diferentes protocolos de administración 99

Introducción

El Instituto de Ingeniería de la Universidad Nacional Autónoma de México es el centro de investigaciones en diversas áreas de la ingeniería más productivo del país. Desde su fundación, la política del Instituto ha sido realizar investigación orientada a problemas generales de la ingeniería, así como colaborar con entidades públicas y privadas para mejorar la práctica de la ingeniería en el ambiente nacional, al aplicar los resultados de las investigaciones a problemas específicos.

Las principales funciones del Instituto de Ingeniería son el desarrollo de la investigación para mejorar los conocimientos, métodos y criterios en ingeniería, contribuir a la formación de expertos en esta rama del saber, así como promover la más alta calidad en la practica profesional. En los programas de trabajo se enfatiza el interés en las necesidades de la ingeniería nacional actuales y previsibles.

Las actividades que se llevan a cabo dentro de esta Institución académica son: investigación técnica y aplicada, apoyo al desarrollo tecnológico y análisis de los requerimientos sociales a cuya solución puede aportar la ingeniería, así mismo, se proporcionan servicios de ingeniería a los diversos sectores de la sociedad con el propósito de contribuir al avance de los objetivos propios de la Universidad Nacional Autónoma de México.

La comunidad que conforma al Instituto de Ingeniería es de aproximadamente 900 personas, entre: investigadores, estudiantes de ingeniería que realizan trabajos de tesis en licenciatura, maestría y doctorado, técnicos académicos, personal secretarial y de servicio. Sus instalaciones comprenden 12 edificios.

Por el carácter y funciones de esta Institución, surge la necesidad de implantar un esquema de cómputo que satisfaga plenamente los requerimientos de su comunidad. Hoy en día se cuenta con una basta gama de herramientas que auxilian en un gran número de tareas, como son : la realización de cálculos más eficientes, el almacenamiento de grandes cantidades de información, proveer un acceso óptimo a la información mundial en las diferentes ramas de la ingeniería, comunicación electrónica, etc. las computadoras son hoy por hoy la principal herramienta de cualquier proyecto de investigación.

Durante algunos años, los equipos de cómputo en el Instituto vivieron aislados, el único medio de acceso entre una computadora y otra se realizaba utilizando algún medio magnético. Algunos usuarios afortunados tenían acceso a la red que los conectaba una computadora central de la Dirección General de Cómputo Académico. Pronto la comunidad del Instituto tuvo la necesidad de compartir información de una manera rápida y eficaz, es así como en 1988 nace la red de cómputo del Instituto de Ingeniería (REDII), siendo esta una de las principales herramientas que apoyan las actividades que en esta Institución se llevan a cabo. A la fecha REDII ha tenido un gran crecimiento, distribuyéndose en 7 edificios, contando con más de 450 computadoras entre estaciones de trabajo y computadoras personales.

La interacción de las computadoras en una red se ha vuelto cada vez más compleja, aunado a ello el número de usuarios de las mismas se incrementa día con día. REDII no es la excepción, cuenta con más de 500 usuarios, provee diversos servicios tales como: Acceso a **Internet**, comunicación electrónica, posibilidad de compartir recursos de cómputo, etc. Los usuarios de REDII esperan una excepcional eficiencia y rapidez en el acceso a los servicios y a la información que presta.

Es en este punto donde surge la necesidad de buscar alternativas que permitan mantener una alta disponibilidad en los servicios que ofrece REDII. Dicha necesidad da origen a este trabajo de tesis.

Objetivos

Considero que tomar este trabajo de tesis es una forma de aplicar la Ingeniería para la solución de un problema real de acuerdo a necesidades especificas. Como primer punto para el desarrollo de dicho trabajo, se platearan sus objetivos.

El objetivo general de este trabajo, es lograr un alto nivel de confiabilidad, disponibilidad y eficiencia en la red de computadoras con la que actualmente cuenta el Instituto de Ingeniería, por medio de la implantación de un sistema de monitoreo y administración el cual provea una fácil detección y corrección de fallas.

En una forma específica y desglosada los objetivos son los siguientes:

- Adaptar una metodología que permita realizar este trabajo de una manera estructurada.
- Realizar una investigación bibliográfica acerca de los protocolos de administración de red existentes, así como la selección del más adecuado para REDII.
- Selección de un sistema de monitoreo y administración basado en el protocolo de administración de red seleccionado.
- Implantación de los elementos seleccionados, para cubrir las necesidades de los usuarios de REDII.
- Sentar las bases para el desarrollo de proyectos en el área de monitoreo y administración de red, en el Instituto de Ingeniería.
- Aplicar los conocimientos adquiridos en la formación universitaria, en las áreas de: redes de computadoras, sistemas operativos, organización y administración de centros de cómputo, entre otras.

Metodología para la implantación de un sistema de administración y monitoreo en REDII.

Introducción

La metodología presentada a continuación fue tomada y adaptada del trabajo "Metodología de preparación, presentación y evaluación de proyectos de informática" [1]; este trabajo ha servido como referencia en distintos proyectos, siendo el más significativo el del CEPEP[2] de la Secretaría de Hacienda y Crédito Público. El objetivo de la metodología antes mencionada, es marcar una pauta que facilite la preparación, presentación, implantación y evaluación de proyectos de informática en empresas, organismos o instituciones públicas.

En este trabajo hemos adecuado dicha metodología para llevar acabo la implantación de un sistema de administración y monitoreo de la red del Instituto de Ingeniería, basándonos en los requerimientos y necesidades de la institución.

Etapas de la metodología

La metodología para la implantación de un sistema de administración y monitoreo en REDII, cuenta con varias etapas:

- Establecimiento de antecedentes.
- Diagnóstico.
- Optimización de la situación actual.
- Alternativas de diseño y proyecto.
- Selección y proyección de alternativas.
- Implantación.
- Presentación de resultados.

[1] Javier E. Mochio Secul: Trabajo de tesis de licenciatura ; Universidad de Chile; 1987

[2] Centro de Estudios de Proyección y Evaluación de proyectos

Establecimiento de antecedentes

En esta etapa, se hace un estudio de la organización y el medio ambiente donde se llevará a cabo el proyecto, es decir se pretende conocer el ambiente al cual se adaptará la solución informática, identificando sus finalidades, restricciones y estado actual. Así mismo se debe identificar la necesidad que da origen al problema.
Una clara descripción general del entorno organizacional ayudará a la justificación, Analisis y comprensión del problema en estudio y por ende, de las distintas alternativas de solución.

Diagnóstico

La información que resulte de esta etapa es clave para las etapas posteriores, por lo cual, se debe analizar la problemática actual, desarrollando de esta manera, un diagnóstico que sirva de base para diseñar soluciones de acuerdo a la naturaleza del problema. Una fuente de información importante para el diagnóstico podrían ser las opiniones de los usuarios y del equipo de administración de la red.

El diagnóstico se puede dividir en dos puntos importantes:

- *Entendimiento del sistema actual.* En este punto se definirán la problemática, deficiencias y limitaciones que se tienen en la actualidad y que se desean resolver con la solución a proponer.
- *Definición de requerimientos.* En esta sección se deben especificar los requerimientos que deberá cumplir la solución a proponer, tomando en cuenta tanto a los usuarios como al equipo de administración. Estos deberán ser descritos de manera estructurada y lo más detallado posible.

El resultado del diagnóstico debe presentar la traducción del problema en términos de requerimientos informáticos, éstos justificarán las soluciones tecnológicas que posteriormente se aplicarán.

Optimización de la situación actual

Una vez realizada la etapa de diagnóstico se debe determinar si es posible mejorar la situación actual con modificaciones mínimas en el esquema existente o con la incorporación de inversiones marginales. Toda modificación que se intente en este punto será con el mínimo de recursos (materiales, de personal o económicos).

Si las modificaciones realizadas en este punto, no resuelven la totalidad de la problemática planteada en la etapa de diagnóstico, se deberá continuar con las siguiente etapas de la metodología.

Alternativas de diseño y proyecto

Basados en los resultados del diagnóstico realizado, se deben buscar distintas alternativas de solución para la problemática encontrada. Es necesario el conocimiento teórico en el que pueden basarse las diversas alternativas existentes, es por ello recomendable que el evaluador de la mismas este empapado de conocimientos al respecto, por esta causa en el trabajo basado en esta metodología se agregan marcos teóricos de los esquemas tecnológicos que representan alternativas de solución para la problemática.

La búsqueda de estas alternativas, se pueden dejar a cargo de dos grupos:

1. Se puede otorgar la búsqueda de alternativas y diseño de las soluciones al personal propio de la empresa o perteneciente a consultores externos contratados para tal efecto.

2. Se puede efectuar un llamado a propuesta pública a las empresas proveedoras de sistemas computacionales. Esta acción se le conoce como *licitación*. El objetivo es que estas empresas realicen los diseños respectivos basados en los requerimientos presentados por la organización, y entreguen sus proposiciones al o los evaluadores del proyecto. Este tipo de solución se presenta atractiva para aquellos proyectos de gran tamaño y complejidad (cabe mencionar que la determinación de necesidades y requerimientos debe ser efectuada por el equipo evaluador del proyecto, fijando los marcos de referencia dentro de los cuales se deberán limitar los proveedores. Toda esta información más la proveniente de la etapa de diagnóstico técnico de la situación actual, permitirá elaborar la base para la licitación para los proveedores de **hardware** o **software**.)

Para la implantación del sistema de administración y monitoreo en REDII, la búsqueda de alternativas se dejo a cargo del personal propio del Instituto.

Otro aspecto que merece ser considerado en este etapa , es el carácter dinámico e interactivo que posee este tipo de estudio, es decir se pueden realizar modificaciones de los procesos durante el desarrollo del estudio. Es por esto que debe existir predisposición por parte del evaluador de la organización, o de los proveedores, a fin de efectuar cambios en el momento que éstos se hagan necesarios y en el momento oportuno, con el objeto de alcanzar la meta u objetivo de la mejor forma posible.

El resultado de la elección de las alternativas, debe ser tal que se complemente con los sistemas existentes en la organización y que se deseen mantener en operación a futuro.

Selección y proyección de alternativas.

Todas las alternativas que sean técnicamente factibles de implantar en la institución u organización, sean éstas obtenidas por estudios propios o proporcionadas por terceros, deberán ser rigurosamente evaluadas. Debiéndose tener en cuenta aspectos tales como, capacidad de operación, crecimiento, velocidad, etc. y si se aplica, hay que considerar costos, tipo de financiamiento, entre otros.

Cuando se quiera efectuar una aproximación del comportamiento futuro de cada alternativa estudiada, se puede utilizar técnicas de simulación, las que permitirán obtener una proyección del comportamiento de las alternativas bajo análisis, en el horizonte predeterminado para ésta.

Cuando se presente el conjunto de alternativas técnicamente factibles, se puede recurrir a distintas técnicas a modo de seleccionar la mejor alternativa. Un mecanismo muy utilizado y recomendable a su vez, es el de trabajar con el método de puntos, este consiste en ir ponderando los parámetros y atributos, de acuerdo a escalas de valores predeterminados. Los resultados que se entreguen permitirán seleccionar la alternativa técnica que más se aproxime a los requerimientos de la empresa.

Implantación

Después de la elección de la alternativa, que satisfaga lo mejor posible los requerimientos de la empresa o institución, se procederá a la instalación del equipo y programas seleccionados. Se realizará la configuración de cada uno de estos elementos para que en conjunto formen el esquema propuesto como solución.

En esta etapa se incluye también el mantenimiento del sistema resultante, así como las pruebas que ayuden a detectar posibles errores en su operación.

Presentación de resultados

Esta es la etapa final de la metodología, donde se expone en que medida fueron cubiertos los requerimientos y expectativas.

Para la realización de esta etapa, es necesaria la utilización de diferentes mecanismos de presentación de resultados, tales como gráficas, esquemas, diagramas de flujo, etc., que muestren de una manera simple y clara los logros alcanzados.

Organización de la tesis

Esta tesis ha sido dividida en 8 capítulos:

Capítulo 1. En este capítulo se realiza una descripción general de las características de REDII, con el fin de sentar antecedentes para la comprensión de su problemática.

Capítulo 2. El trabajo en este capítulo esta orientado al señalamiento de los problemas y carencias que REDII tiene y que pueden ser solucionados mediante la implantación de un sistema de administración y monitoreo.

Capítulo 3. Se establecen las bases teóricas para la implantación de un sistema de monitoreo en una RED.

Capítulo 4 y 5. Se exponen y evalúan diferentes alternativas para la implemetación de un sistema de administración y monitoreo. Finalmente se selecciona la más adecuada.

Capítulo 6. La implantación del programa de monitoreo es explicada a lo largo este Capítulo.

Capítulo 7. Se muestran los resultados obtenidos a lo largo de este trabajo.

Capítulo 1

La red del Instituto de Ingeniería (REDII)

1.1 Introducción

El esquema de cómputo distribuido adoptado por el Instituto de Ingeniería, basa su funcionalidad en una red de computadoras, y ésta es también tomada como una de las herramientas principales para llevar a cabo sus objetivos de trabajo e investigación.

Es fundamental conocer y analizar dicha red desde sus niveles físicos hasta sus niveles lógicos, para proponer un sistema de administración y monitoreo que facilite las tareas de administración de la red y provea un fácil detección de fallas.

En la primera parte de este capítulo, se hará una descripción de la estructura de REDII basada en los conceptos generales de las redes, posteriormente se describirán los protocolos que REDII emplea para su funcionamiento, finalmente en la tercera parte de este capítulo se presentarán los servicios que la red del Instituto de Ingeniería ofrece.

1.2 Estructura de la red del Instituto de Ingeniería

1.2.1 Conceptos generales acerca de redes.

El almacenamiento y el análisis de información ha sido uno de los grandes problemas a que se ha enfrentado el hombre desde que se inventó la escritura. No es sino hasta la segunda mitad del siglo XX que ha podido resolver, parcialmente, este problema, gracias a la invención de la computadora. Cuando el uso de la computadora se hizo popular, ésta se volvió una herramienta básica dentro de las organizaciones, entonces surgió la necesidad de compartir datos y recursos de cómputo, lo cuál llevó a diversos fabricantes y desarrolladores a idear las redes.

Como definición de una red de computadoras podemos mencionar:

Una red de computadoras es un sistema de comunicaciones de datos que provee interconexión a una variedad dada de dispositivos.

Los dispositivos que interconecta una red incluyen a cualquiera que pueda comunicarse sobre un medio de transmisión, como por ejemplo: Computadoras, terminales, periféricos, teléfonos, etc.

Las redes proporcionan las siguientes ventajas principales :

- Compartir archivos y programas
- Compartir recursos de cómputo
- Compartir bases de datos
- Creación de grupos de trabajo, estos grupos de trabajo pueden estar conformados por usuarios que no se localicen necesariamente en el mismo departamento, y puedan laborar de una manera confortable desde sus lugares de trabajo
- Intercambio de mensajes electrónicos dentro de la Institución y en el mundo entero
- Acceso a sistemas de información en todo el mundo.

Estas ventajas se traducen en que los usuarios tienen múltiples beneficios, tales como: la cantidad de información que puede almacenarse y manejarse es mayor, además de tener acceso a fuentes de información mundial; los usuarios tienen acceso a muchos más recursos de cómputo; el acceso a dichos recursos e información es más fácil y rápido; los grupos de trabajo en una empresa encuentran un medio de unificación par sus labores en una red; con la comunicación electrónica se ahorra tiempo y dinero ya que se puede contactar a cualquier persona en cualquier lugar del mundo por medio de las redes de cobertura mundial.

1.2.1.1 Tipos de redes

Una red de computadoras consta tanto de **hardware** como de **software.** El **hardware** incluye tarjetas de interfaz de red, dispositivos de interconexión, computadoras y cable, mientras que el **software** incluye sistemas operativos, protocolos de comunicación y controladores de las tarjetas interfaz de red. El sistema operativo y los protocolos que proporcionan servicios de comunicación definen el *entorno de una red.*

Clasificación de redes por cobertura geográfica.

La clasificación fundamental de una red, se basa en la cobertura geográfica que ellas alcanzan. Antes de entrar a detalle en dicha clasificación, deberemos definir un concepto importante: a dos o más dispositivos de cómputo, con trafico de información local, conectados al mismo dispositivo de interconexión de red se le conoce como *segmento de red o subred.* Este concepto nos ayudará a definir los tipos de redes clasificados mediante su cobertura geográfica:.

• ***Red de área local (LAN)***
Una **LAN** es un segmento o conjunto de segmentos de red interconectados, generalmente dentro de la misma zona, como por ejemplo un edificio, o un conjunto de edificios, un campus universitario, o una institución científica, educacional o comercial.

• ***Red de área metropolitana (MAN)***
Una red **MAN** es una red que se expande por ciudades o provincias y se interconecta mediante diversas instalaciones publicas o privadas.

• ***Red de área amplia o extendida (WAN) y redes globales***
Las **WAN** y las redes globales se extienden sobrepasando las fronteras de las ciudades, provincias o naciones. Los enlaces se realizan con instalaciones de telecomunicaciones públicas y privadas, además de enlaces por microondas y satélites.

Clasificación de redes por tipo de transmisión de paquetes

Como se vio, la clasificación anterior fue hecha en base al tamaño de las redes, sin embargo las redes también pueden clasificarse con base a la manera en que un paquete de datos en la red es transmitido y el camino por el cual dicho paquete puede alcanzar su destino.

• ***Redes de dispersión de paquetes***
Los dispositivos de cómputo (en los que se incluyen computadoras y periféricos), comparten un medio de comunicación en el cual una transmisión para un dispositivo es "escuchada" por todos los dispositivos conectados al medio. Los datos que son transmitidos son fraccionados en pedazos llamados paquetes de red, cada paquete es enviado por la red por la computadora transmisora y es recibido por todos los dispositivos conectados a la red.

•***Redes de circuito switcheado***
En este tipo de red los elementos que se ven involucrados en ella, están conectados a través de un dispositivo que permite crear un circuito virtual entre dos elementos, este circuito consiste de un camino y recursos *dedicados* para transferir datos entre ambos elementos.

1.2.1.2 Medios de Transmisión

El medio de transmisión, es el camino físico por el cual los datos viajarán del elemento transmisor hacia el elemento receptor en una red. Los medios de transmisión, se pueden clasificar como *medios guiados o medios no guiados*. En ambos casos, la transmisión de datos es en forma de ondas electromagnéticas y/u ópticas. Con el medio guiado las ondas son orientadas a través de un camino físico. Por otro lado, en la segunda clasificación no hay necesidad de guiar las ondas, ya que son transmitidas por aire o espacio.

Medio guiado.
En él se incluye el cable de metal (cobre, aluminio, etc.) y el cable de fibra óptica. El cable suele instalarse dentro de los edificios o bien por ductos subterráneos. Entre los cables de metal se incluye:

Cable coaxial: El cable coaxial consta de un núcleo de cobre sólido rodeado por un aislante que es un material dialéctico no conductor y una malla metálica externa la cual actúa como tierra atrapando señales externas y manteniéndolas fuera del núcleo, todo el conjunto está protegido por una cubierta exterior también aislante, a la que por lo general se le llama **jacket**.
Los cables coaxiales pueden ser de varios tipos y anchos. Sin embargo, su principal característica es que pueden transportar una señal eléctrica a mayor distancia entre más grueso es el conductor.
Con el cable coaxial se puede transmitir datos con una velocidad máxima de 155 Mega bits por segundo (Mbps.)[1]. Este cable, puede usarse para la transmisión de voz, vídeo y datos, su instalación es fácil, tiene una buena tolerancia a interferencias debidas a factores ambientales, y puede proporcionar distancias hasta de 600 metros sin necesidad de repetidores. En este medio las señales transmitidas son analógicas.

Cable par trenzado: El cable de par trenzado consta de conductores de núcleo de cobre rodeados por un aislante. Se trenzan dos hilos juntos para formar un par, dicho par forma un circuito por el que se pueden transmitir datos. Un cable par trenzado consta de uno o más pares trenzados rodeados por un aislante. Este tipo de cable puede clasificarse en dos: el **UTP** por las siglas en inglés **Unshielded Twisted Pair** que es el más común y el **STP** por **Shielded Twisted Pair** llamado así porque tiene un blindaje parecido al del cable coaxial. Para el cable par trenzado existen diferentes categorías:

- categoría 1: Es el cable que se utiliza comúnmente para las instalaciones telefónicas, el cual puede transmitir voz pero no datos.
- categoría 2: Es el cable par trenzado certificado para la transmisión de datos hasta 4 Mbps, este cable tiene dos pares trenzados
- categoría 3: Admite velocidades de transmisión de hasta 10 Mbps y tiene cuatro pares trenzados
- categoría 4: Esta certificado para velocidades de transmisión de hasta 16 Mbps y tiene cuatro pares trenzados
- categoría 5: Es un cable de cobre con cuatro pares de 100 ohms, que puede transmitir datos a velocidades mayores a 100 Mbps.

Este tipo de cable es fácil de instalar, puede proveer distancias de hasta 110 metros sin necesidad de repetidor en el caso de **UTP**, y de hasta 500 metros en el caso de **STP**, además cuenta con una buena tolerancia a interferencias debidas a factores ambientales.

Existen otros medios de transmisión que no están hechos de cobre , y entre ellos tenemos :

[1] Dato tomado de : The McGraw-Hill High-Speed LANs Handbook, Stephen Saunders, MacGraw-Hill, primera edición, 1996, pag. 378.

Cable Fibra óptica: Una fibra óptica es un medio pequeño (2 a 125 μm de diámetro) y flexible, capaz de conducir un rayo de luz.
Varios tipos de vidrio y plásticos pueden ser usados para fabricar un cable fibra óptica.
La fibra óptica tiene una forma cilíndrica y consiste en tres secciones concéntricas: el núcleo, el recubrimiento y la sobrecubierta. El núcleo es la sección situada más adentro y consiste de una o más fibras muy delgadas hechas de plástico o vidrio, cada una de estas fibras es rodeada de su propio recubrimiento. La capa externa que contiene a todas las fibras y sus respectivos recubrimientos, es llamada sobrecubierta.
Para la transmisión de la información en las redes, vía fibra óptica se utiliza una fibra como transmisor y otra como receptor. Es por esto que generalmente se producen en conjuntos de mínimo dos fibras por cable.
La fibra óptica puede llegar a transmitir a velocidades mayores de 700 Mbps, y transmitir una señal (sin necesidad de repetirla) hasta 2,000 metros, además de que se puede transmitir voz, datos y vídeo por el mismo canal, es inmune a las interferencias, tiene una excelente tolerancia a factores ambientales.

• ***Medio no guiado***
Representa la técnica que se utiliza para transmitir señales por aire y espacio, desde el transmisor al receptor, tales como infrarrojos, microondas y laser.
Estos medios se usan primordialmente para conectar redes que están situadas físicamente en lugares apartados uno del otro, donde es difícil implantar un medio de transmisión guiado, ya sea por la distancia, o por la dificultad que el terreno presenta, o bien si el espacio requerido para la implantación del medio guiado es espacio federal y no se tiene acceso a el.

1.2.1.3 Arquitectura de red

La arquitectura de una red está definida por su *topología física*, *método de acceso al medio*, y *protocolos de comunicación* que utiliza.

Topología física

El término topología física se refiere a la manera en la cual los nodos de la red son interconectados. Una topología física es definida por el diseño de las ligas de comunicación y los elementos de interconexión, esto determina las diferentes rutas, que tal vez los datos utilicen para la comunicación entre cualquier par de nodos.

Existen 4 topologías simples: bus, árbol, anillo y estrella:

• ***Topología de bus y de árbol.*** Una topología lineal o de bus consta de un único cable que se extiende de un dispositivo al siguiente. Los extremos del cable se terminan con una resistencia. El cable único es fácil de instalar, pero una ruptura en cualquier parte del mismo desactiva toda la red.
La topología de árbol es una generalización de la de bus. El medio de transmisión es un cable bifurcado, donde las bifurcaciones jamás se unen. Las capas del árbol empiezan en un punto conocido como la *cabecera*, uno o más cables nacen de este punto y cada uno puede tener o no bifurcaciones.

• ***Topología de estrella.*** En la topología de estrella todos los hilos parten de un solo punto, (como un servidor de archivos o un dispositivo de interconexión) hacia los dispositivos que conforman la red. La topología estrella necesita un cable a cada dispositivo, Si la comunicación en un cable se pierde, solo se desconectan las computadoras conectadas a el.

• ***Topología de anillo.*** Esta topología consiste en dispositivos de interconexión unidos por un cable en un circuito cerrado, los dispositivos deben ser capaces de recibir datos y transmitirlos tan rápidamente como estos llegan, ya que si la información transmitida no es para el dispositivo en turno, deberá ser reenviada al siguiente dispositivo del anillo. La información corre alrededor del anillo en una dirección y es fraccionada en paquetes. Cada computadora o periférico deberá estar conectado a la red mediante el dispositivo de interconexión

La figura presentada a continuación muestra los 4 tipos de topologías descritas anteriormente.

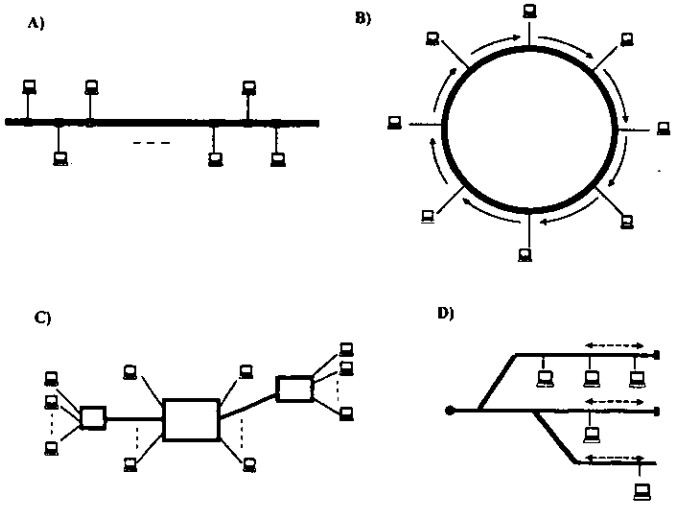


figura 1.1 Opciones de topologías de red a) bus b) anillo c) estrella d) árbol

Método de acceso al medio

Con el método de acceso al medio se describe la forma en que una computadora consigue el acceso al medio de transmisión. Cuando una computadora logra acceder al medio puede empezar la transmisión de información.

A pesar de existir diversos métodos de acceso al medio, los más conocidos son:

• *CSMA/CD* : (Acceso múltiple con detección de portadora/Detección de colisión) este método permite que cualquier computadora pueda tener acceso al medio siempre y cuando éste se encuentre desocupado, en el caso de que una computadora este transmitiendo, otra que desee acceder al medio no podrá hacerlo hasta que la primera lo desocupe. Cuando una computadora difunde una señal, todas la computadoras de la red la reciben, pero solo la computadora direccionada la atiende. Si dos computadoras envían una señal al mismo tiempo, se produce un fenómeno llamado *colisión*, posteriormente ambas se retiran del medio, esperan un período de tiempo aleatorio y vuelven a intentar la transmisión. Cabe mencionar que el rendimiento de las redes que utilizan este tipo de método de acceso se degrada al aumentar el número de colisiones y retransmisiones.

• *Token passing* : Este método de acceso, solo permite transmitir a la computadora que posee una señal testigo (**token**). Se puede pensar en un testigo como una estafeta o pase temporal para utilizar el medio. Cuando una computadora esta preparada para transmitir, debe esperar a que esté disponible la estafeta, además de esperar su turno en el arreglo de computadoras. Este método es utilizado tanto en la tecnología *token ring* (topología física en anillo) y *token bus* (topología física en bus y lógicamente en anillo).

Protocolos de comunicación

Los protocolos de comunicación son las reglas que los dispositivos conectados a una red deberán seguir para poder comunicarse entre ellos.

Los protocolos de comunicación se explicarán más detalladamente, posteriormente en este capítulo.

1.2.1.4 dispositivos de interconexión de redes

En la mayoría de los casos, una red no es una entidad aislada. Una organización (educacional, de investigación o empresa privada) puede tener más de una red local dentro de ella. Estas redes necesitan ser interconectadas para proveer a los usuarios de una red global unificada. A la vista de los usuarios finales, una red constituida por varias subredes, es considerada simplemente como una gran red, sin embargo cada red que conforma a una red mayor tiene sus propios mecanismos especiales para transmitir datos. Los responsables de interconectar redes (sin importar cual sea su topología o mecanismos de transmisión), para construir una red más grande son los *dispositivos de interconexión de red.*

*Los repetidores, puentes, enrutadores, conmutadores (**switches**) y **gateways** son las cajas negras que nos permiten utilizar diferentes topologías y protocolos dentro de un solo sistema heterogéneo..*

Los *repetidores* son los dispositivos de interconexión más sencillos, que permiten unir segmentos de red de un mismo tipo. El trabajo de estos dispositivos es retransmitir, regenerar y amplificar las señales de datos de un segmento a otro. Debemos tomar en cuenta que todos los paquetes que llegan por un segmento conectado al repetidor, pasaran al siguiente segmento conectado por este sin importar cual sea su destino, es decir los repetidores no filtran paquetes.

Los *puentes* pueden interconectar redes que cuentan con medios de transmisión, topologías y métodos de acceso, diferentes. Los puentes analizan cada paquete que llega a ellos y seleccionan a los que tienen como destino una red que este conectada al otro lado del puente, dejando pasar a ella sólo a los paquetes seleccionados, de esta manera filtran trafico en la red. Estos dispositivos tienen la capacidad de aprender las direcciones destino de los paquetes que pasan por ellos y proveen transparencia en los protocolos de alto nivel (por ejemplo **TCP/IP**), a pesar de que las redes que interconectan sean totalmente diferentes con respecto a su topología, medio físico y método de acceso al medio. Es así como en las circunstancias adecuadas, los puentes pueden usarse para interconectar redes similares como dos **Ethernet** o mezclar redes diferentes, como es una **Token ring** y una **Ethernet** (puentes traductores). Es importante mencionar que los puentes no modifican el contenido de los paquetes que pasan por ellos, ni le adicionan nada.

Los *conmutadores* ***(switches)*** permiten interconectar varias subredes, creando para cada una un circuito dedicado a un tiempo, es decir un **switch** provee una conexión dedicada para cada subred en un determinado tiempo permitiendo la unión de dicha subred con otra, logrando de esta manera que las subredes puedan integrarse con una red más grande. Al igual que los puentes, los conmutadores operan en la capa de enlace de datos (capa 2 del modelo **OSI**)

Los *enrutadores* son dispositivos más inteligentes que los puentes, ya que pueden tomar decisiones de enrutamiento que determinen la trayectoria más eficiente para la transmisión de datos entre dos redes. A los enrutadores no les interesa saber que topologías o que mecanismos de transmisión se utilizan en las redes que ellos conectan, puesto que operan en la capa 3 del modelo **OSI** (capa de red), es decir, los enrutadores no están limitados por el método de acceso al medio. Los enrutadores eligen el mejor camino para el paquete tras revisar una tabla de enrutamiento donde almacenan los posibles caminos para un paquete.

Los ***gateways*** son utilizados para interconectar redes que se construyeron totalmente en base a diferentes arquitecturas de comunicación. Es decir, el **gateway** debe traducir todos los datos que pasan entre las dos redes y de esta manera interconectarlas. Los **gateways** no proporcionan enrutamiento de paquetes entre los segmentos de red que ellos conectan, simplemente entregan los paquetes de tal forma que los equipos en los segmentos puedan leerlos.

1.2.1.5 Tecnologías

Las tecnologías de red se pueden dividir en dos tipos: tecnologías de baja velocidad, como son Ethernet, Ethernet conmutado y token ring y las tecnologías consideradas de alta velocidad, como son Fast Ethernet, FDDI y ATM.

- ***Ethernet***

Ethernet es el nombre dado a la tecnología de comunicación entre computadoras más utilizado en la actualidad, fue inventada por **XEROX** al principio de los años 70's. **Ethernet** es una tecnología de *dispersión - bus* a 10 Mbps. Se le llama de bus porque todos los dispositivos comparten un mismo medio de transmisión al cual están conectados.

Originalmente **Ethernet** utilizaba una topología de bus físico a través de cable coaxial, actualmente esta tecnología soporta diversos medios de comunicación los cuales se muestran en la siguiente tabla.

Parámetros	10Base5[13]	10Base2[5]	10BaseT[5]	10Base36[5]	10BaseF[5]
Medio de transmisión	coaxial	coaxial	par trenzado UTP	coaxial	fibra óptica
Diámetro del cable (mm)0	10	5	0.4-0.6 (26-22 AWG)	0.4-1.0	
Tasa de transmisión de datos (Mbps)	10	10	10	10	10

Tabla 1.1 Opciones en medios de comunicación para Ethernet

Esta tecnología utiliza un método de control de acceso al medio referido como CSMA/CD el cual se caracteriza por utilizar dispersión de paquetes para su funcionamiento, es decir, todos los dispositivos conectados a la red reciben todos los paquetes transmitidos.

• ***Token ring***

Token ring es una tecnología de comunicación que puede operar a 4 Mbps o 16 Mbps. Está basado en una topología de anillo físico, soportando cable par trenzado **UTP** y **STP**. El método de control de acceso al medio que utiliza es referido como **Token passing.**

• ***Ethernet conmutado (Ethernet switch)***

Ethernet conmutado, se basa en las tecnologías de conmutación para dividir una red en segmentos, lo que provee un mejor aprovechamiento del ancho de banda de 10Mbps en cada segmento. Utiliza el método de acceso al medio CSMA/CD, con topología en estrella, implantada con cable par trenzado (UTP) o fibra óptica.

• ***ATM***

Nombre que recibe de sus siglas en inglés **Asynchronous Transfer Mode** o modo de transferencia asíncrona. Es una tecnología que nos proporciona la ventaja de crecer a una red con velocidades desde los 25 Mbps hasta varios Gigabits por segundo (Gbps), permitiendo la implantación de aplicaciones que necesiten altas velocidades de transmisión, estas aplicaciones pueden manejar voz, datos y video.

ATM necesita una topología en estrella, que puede ser implementada con cable par trenzado y/o fibra óptica.

ATM permite la transmisión de datos en dos direcciones dentro del medio, esto permite la transmisión y recepción de datos simultáneamente y sin interrupciones. La implantación de una red con tecnología **ATM** es muy costosa y compleja.

• ***FDDI***

Llamada así por sus siglas en inglés **Fiber Distributed Data Interface** (**FDDI**), trabaja a 100 Mbps y requiere una topología de anillo doble. El anillo doble ofrece redundancia (tolerancia a fallas). Si se produce una falla de un enlace o se corta el cable, el anillo se reconfigura por sí solo, de modo que puede continuar la transmisión de paquetes por la red. **FDDI** utiliza un método de acceso al medio de paso de testigo llamado **Token Append** que tienen la esencia de **Token ring** pero se le adicionan mecanismos de regulación para evitar que un dispositivo mantenga el testigo durante mucho tiempo. El medio de transmisión a emplear, se puede seleccionar entre la fibra óptica y el par trenzado, cuando se selecciona éste último el anillo recibe el nombre de **CDDI (Copper Distributed Data Interface)** o **FDDI/UTP** . **FDDI** ofrece un alto nivel de interoperabilidad con tecnologías existentes que usan puentes, enrutadores y **switches**.

• ***Fast Ethernet***

Fast Ethernet o 100BASE-T **Ethernet**, es el resultado de una colaboración industrial para extender las especificaciones **Ethernet** hacia las tecnologías de alta velocidad, manteniendo el corazón de la misma y su compatibilidad con las aplicaciones existentes. **Fast Ethernet** tiene una velocidad de operación de 100 Mbps, el método de acceso al medio que utiliza es **CSMA/CD,** con una topología de estrella que puede ser implementada con fibra óptica y/o par trenzado. Una característica muy especial de esta tecnología es que puede auto-ajustar su velocidad de transmisión a 10 Mbps ó a 100 Mbps.

1.3 Descripción de la red del Instituto de Ingeniería (REDII)

Con el marco teórico anterior, reunimos los conocimientos básicos necesarios para describir a la red del Instituto de Ingeniería REDII.

1.3.1 Antecedentes

En el Instituto de Ingeniería, se cuenta con una red de computadoras, la cual brinda a dicha institución múltiples servicios. REDII, es una parte importante de la red de computadoras de la UNAM (REDUNAM).

REDII, nació en 1988, en ese entonces la red existía únicamente en un solo edificio y la topología que se empleó para implementarla fue **Token Ring**. Se contaba con pocas computadoras personales conectadas a la red con tarjetas **Token Ring** de 8 bits, y el programa que se utilizaba para la comunicación fue **Lan-Manager** de **IBM**. La comunicación hacia el exterior del instituto se realizaba vía **Modem**.

Para 1989, llegaron más computadoras al Instituto y se decide extender la red a más edificios y cambiar la topología. Para esta época REDII, se extendió a 3 edificios, y se planeó para cada edificio tener un servidor con una red local con topología de **bus**. Para la conexión entre edificios se utilizó cable coaxial y se implemento **Novell** 2.15, como servidor de archivos. La comunicación hacia el exterior del instituto seguía realizándose vía Modem.

En 1991 se decidió incorporar otro edificio a la red, siendo así 4 edificios en red. En 1992 se decide emplear cable par trenzado para la conexión en los edificios y entre ellos se utiliza cable coaxial en una topología de **bus**. Para 1993-1994 se cuenta con una red funcional en cinco edificios, basada en **TCP/IP** con servicios proveídos por los sistemas operativos **UNIX** interactuando con servidores **NOVELL**.

1.3.2 Infraestructura Actual

Actualmente, REDII es una red de área local, que abarca ocho de los diez edificios, que componen el Instituto de Ingeniería. En la siguiente figura se muestra la disposición de los edificios del Instituto dentro del campus universitario.

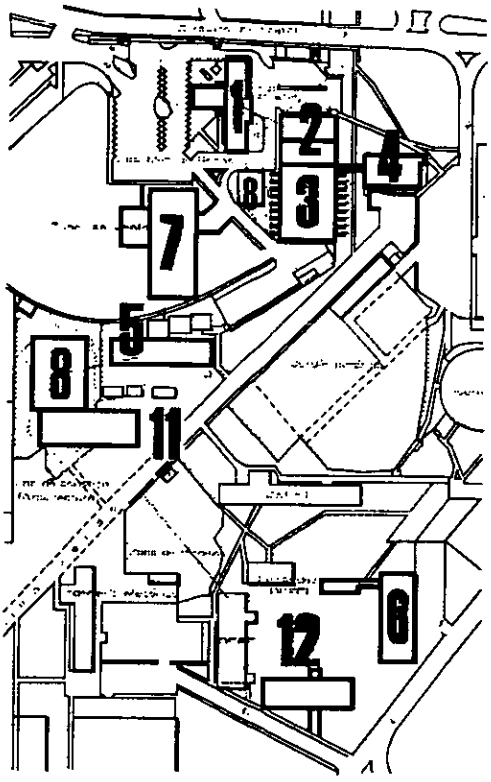


figura 1.2 Ubicación física de los edificios del Instituto de Ingeniería

Cada edificio cuenta con varios dispositivos de interconexión que permiten la unificación de REDII. La interconexión de cada edificio cuenta con peculiaridades propias, por lo que para mayor claridad se citarán los dispositivos de cada edificio en la siguiente tabla.

Edificio	Equipo
1	Concentrador SEHI-24 (Cabletron) Concentrador SEH-24 (Cabletron) Concentrador SEH-34 (Cabletron)
2	MMAC-M5FNB (Cabletron) Tarjeta de conexión a red IRBM (Cabletron)
3	Concentrador SEHI-34 (Cabletron)
4	Concentrador MMAC-M3FNB con salida IRBM (Cabletron)
5	Concentrador MMAC-M3FMB con salida IRBM (Cabletron) Concentrador HP48J2602A (HP) Concentrador SEHI-24 (Cabletron) 2 concentradores SEH-24 (Cabletron)
6	Concentrador MRXI (Cabletron)
8	Utiliza la interconexión del edificio 3
12	Concentrador MMAC-M8FNB con salida IRBM (cabletron)

Tabla 1.2 Dispositivos de interconexión por edificio.

REDII está configurada para emplear una topología física en estrella, la cual se basa en una tecnología **Ethernet** conmutado a 10 Mbps. Se emplean cable par trenzado categorías tres y cinco, además de fibra óptica como medios de transmisión, mientras que los protocolos que utiliza son : **TCP/IP** y **IPX/SPX**.

Una característica importante de la topología de REDII, es que como punto medular se utiliza un conmutador **Ethernet**, esta característica es la que hace que la tecnología en la que se basa esta red se designe como **Ethernet** conmutado.

Aunque la topología de REDII es básicamente en estrella se tiene un enlace **FDDI** desde el conmutador **Ethernet** hacia el concentrador MMAC del edificio 12, además de un enlace entre los servidores principales a través de **CDDI**, esto con el fin de tener un ancho de banda de 100 Mbps en la conexión hacia éstos.

REDII cuenta con más de 400 puntos de red, ocupados por computadoras personales, más de 60 estaciones de trabajo con sistema operativo **UNIX**, un servidor **NOVELL** y con cinco **Windows NT**.

Por último, cabe mencionar que REDII, se enlaza a REDUNAM, a través de un enlace de fibra óptica a un enrutador Cisco, de esta manera nuestra comunidad puede utilizar los recursos de cómputo de toda la UNAM, y tener acceso a **Internet**. A continuación se muestra en la siguiente figura un esquema actual de REDII.

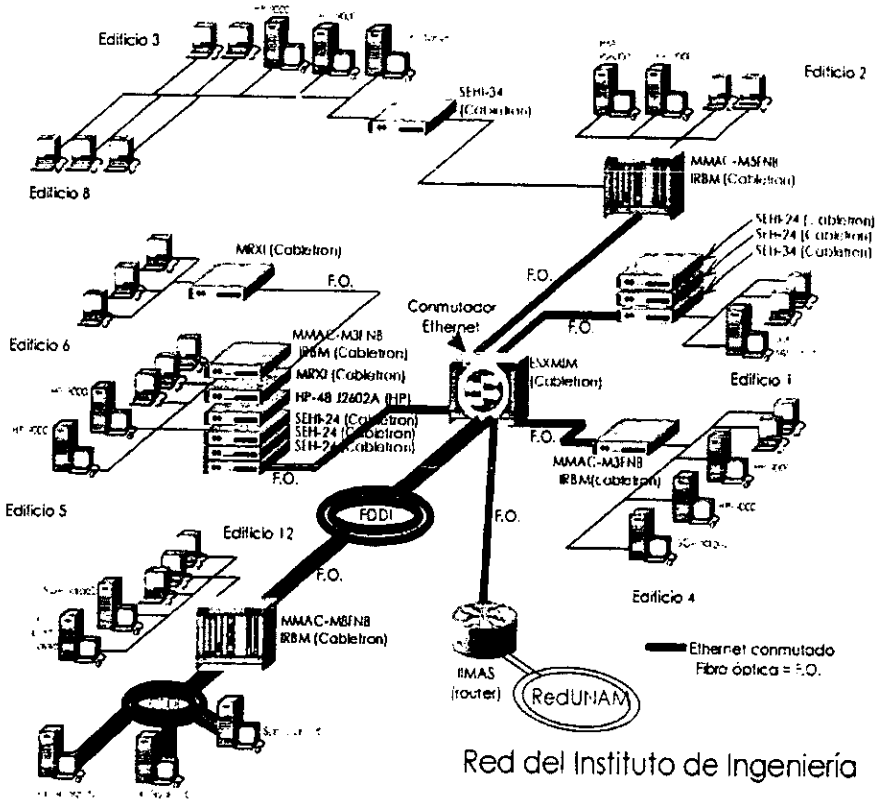


figura 1.3 La red del Instituto de Ingeniería REDII

1.4 Protocolos usados por REDII

1.4.1 Introducción

Los ***protocolos*** permiten que los sistemas de procesamiento en una red puedan hablar unos con otros. Técnicamente son reglas de comunicación, que coordinan el intercambio de datos entre sistemas de procesamiento, haciendo este intercambio eficiente. Es imposible que un solo protocolo realice todas las tareas de comunicación, es por eso que se tiene un esquema de niveles o capas de protocolos con diferentes funciones, juntas estas capas proveen uno o varios servicios para el usuario.
La conjunción de estos niveles de protocolo es conocida como ***modelo de capas de protocolos***, cada industria que desarrolla tecnologías de computo, tiene su propio modelo de capas, sin embargo existe un modelo en el cual muchos desarrolladores basan sus implementaciones, o bien sirve como una referencia para analizar y comparar modelos de capas de protocolos. Este modelo es conocido como **Open Systems Interconnection (OSI)**, que fue desarrollado por **International Organization of Standars (ISO)** en 1978, este modelo cuenta con 7 capas:

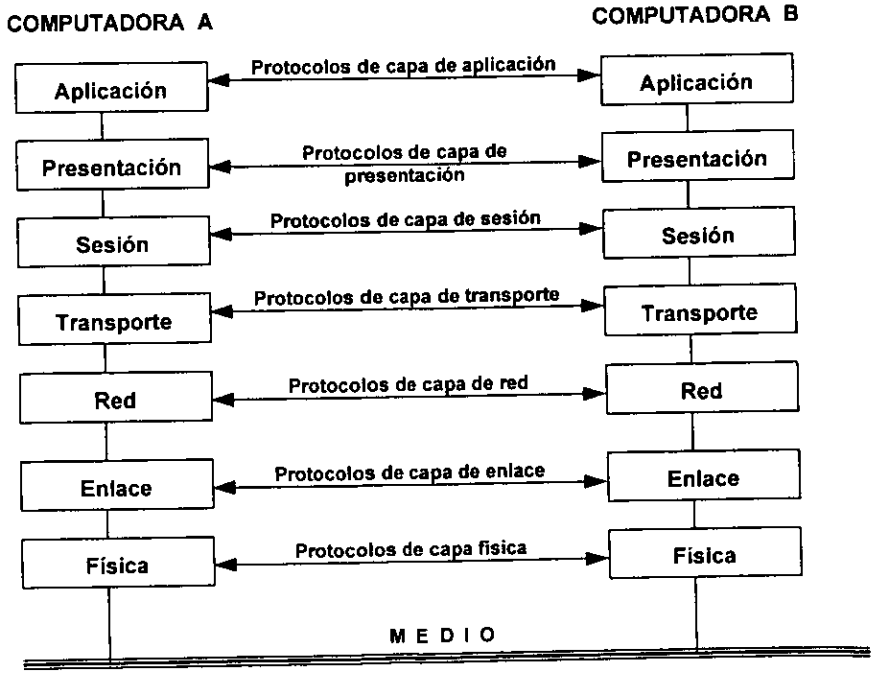


Figura 1.4 Modelo de capas de protocolo OSI

Capa física:
Esta capa se encarga de controlar el intercambio de bits sobre un canal de comunicación o medio de transmisión, esta información puede ser información del usuario, e información referida al control de datos sobre la red como el tipo de transmisión, tipo de codificación, conexión, etc.

Capa de enlace:
La tarea de esta capa es asegurarse que la transmisión de las unidades de información (paquetes de red), y la conexión entre nodos a través del medio de transmisión se realice. Es en esta capa donde se empiezan a manejar los primeros y mas elementales mensajes de error.

Capa de red:
En esta capa se establecen caminos virtuales punto a punto entre dispositivos dentro de la red. Es en esta capa donde se realiza la tarea llamada enrutamiento, la cual consiste en establecer un camino lógico para que los paquetes lleguen a su destino, también es donde se regula el flujo de datos. En la capa de red se hace posible la interconexión entre dos redes, estableciendo un mecanismo de direccionamiento.

Capa de transporte:
Esta capa es responsable de la entrega de los paquetes de información que las aplicaciones generan, esta capa debe asegurase de que los datos no estén corruptos, para ello construye un mecanismo de verificación de datos y manejo de errores para informar de ellos o tratar de recuperar la información perdida o dañada.

La capa de transporte es la parte media del modelo **OSI**, las tres capas que siguen son usualmente implementadas para el **software** de red dentro de la computadora.

Capa de sesión:
La capa de sesión, permite realizar una conexión lógica entre aplicaciones, estableciendo así una sesión entre ambas. Como ejemplos de sesión podemos mencionar a un usuario comunicándose a una máquina con un proceso de **login** o bien comunicándose a una máquina para una transferencia de archivos. Cuando se lleva acabo una sesión, se debe controlar varias situaciones, este control se realiza precisamente en esta capa. Dentro de estas tareas de control se encuentran:
Control de dialogo: Una sesión en general, permite que la información viaje hacia ambos sentidos en una comunicación **(full duplex)**, algunas aplicaciones requieren que la información solo viaje en un solo sentido en una comunicación **(half duplex)**, la capa de sesión provee las dos maneras de comunicación, a la elección de alguna de ellas se le llama control de diálogo.
Manejo de actividad: La capa de sesión permite llevar el manejo de la actividad dentro de una sesión, con el fin de prevenir la pérdida total de la información si al momento de que esta sea manipulada en la sesión se provoque una falla importante en la red.

Capa de presentación:
La capa de presentación define el formato en el que los datos van a ser representados. Los sistemas de cómputo algunas veces usan diferentes métodos de codificar textos, números, etc., en esta capa, se determina el tipo de codificación de datos para que estos puedan ser intercambiados en diferentes sistemas.

Capa de aplicación:
Es la capa final de la arquitectura **OSI,** y es en dónde pueden ejecutarse las aplicaciones finales, tales como transferencia de archivos, acceso a bases de datos, acceso a sistemas de archivos remotos, sistemas de monitoreo, **software** para manejo de dispositivos etc.

En adición a las tareas de los protocolos de direccionar a los puntos finales, el control de flujo de datos y todas las tareas mencionadas anteriormente, otra de las tareas importantes de los protocolos es proveer servicios de transmisión de datos, con estos servicios nos referimos a la detección de errores, la eliminación de los mismos, y el redireccionamiento cuando una dirección falla.

En este capitulo realizaremos el análisis de los protocolos usados por **REDII** basados en las características del modelo de capas de protocolo **OSI.**

1.4.2.1 TCP/IP

Introducción

Al principio de los sesenta, varias universidades y centros de investigación de los Estados Unidos, externaron la necesidad común de tener una red de computadoras con el fin de aprovechar los recursos que tenían dentro de esta área.
Entonces la agencia de investigación de proyectos avanzados de la defensa de los Estados Unidos **(ARPA),** crea **ARPANET**, una red conectada por nodos llamados Procesadores de mensajes **Internet** o **MIPs**. Al principio la red contaba con pocos **MIPs**, entre los cuales se encontraban la Universidad de California los Ángeles, Santa Barbara, el Instituto de Investigaciones de Stanford y la Universidad de Utah. Esta red fue construida para estudiar técnicas de comunicación de datos que fueran robustas y confiables.
El éxito de **ARPANET** fue tal, que muchas de las organizaciones que estaban conectadas a ella empezaron a utilizarlas, cada vez más.
En 1975 **ARPANET** pasó de ser una red experimental a ser una red operacional, y la responsabilidad de administrarla cayó sobre la Agencia de Comunicaciones de la Defensa Norteamericana, **(DCA-Defence Comunications Agency).** La tecnología desarrollada hasta entonces por **DARPA** incluyó un conjunto de protocolos llamados **TCP/IP.** Este conjunto de protocolos fue adoptado como estándar militar para 1983, y aquí empieza el verdadero auge de **TCP/IP.** Para 1983 **ARPANET**, es dividida en dos redes: **MILNET,** la parte de la red que se refiere a la defensa y una nueva **ARPANET,** más pequeña . Es entonces cuando aparece el termino **INTERNET**, que fue utilizado para referirse a ambas redes. En 1990 **ARPANET** desaparece formalmente, y la red que resulta es la red **INTERNET.**

Para entender, con mayor facilidad cómo funciona **TCP/IP**, debemos hablar primero de la arquitectura de la red **INTERNET.**
La red **Internet,** está formada por millones de computadoras, pertenecientes a miles de subredes distribuidas por todo el mundo.
Esta red nos proporciona una interconexión universal, no dependiente de la tecnología de las subredes ni del **hardware** de las computadoras que la conforman. Esto es posible gracias a la construcción de un sistema de red unificado, cooperativo, y con una interconexión de redes que soporte un servicio de comunicación universal.
Cada elemento que constituye esta red, es identificado por un nombre único, llamado también dirección. El direccionamiento de estos nombres únicos, es un método global. Las direcciones están formadas por 32 bits y se les conoce con el nombre: dirección **Internet** o dirección **IP.** Esta dirección contiene cuatro campos de 8 bits cada uno, cada campo es separado del otro por un punto. Una dirección **IP,** es dividida en dos partes: una parte de dirección de red o dominio y una parte de **host** o **local.**
El siguiente es el formato general de una dirección **IP:**

campo1.campo2.campo3.campo4

Estas direcciones se encuentran divididas en diferentes clases que dependen del espacio de direcciones que define la parte de red. Existen tres clases de direcciones:

clase A, clase B, clase C

Direcciones clase A: Para estas direcciones el primer campo representa la parte de dirección de red o dominio, y los tres últimos campos son la parte local, se pueden conectar hasta 2 a la 24 dispositivos.
Direcciones clase B: Los dos primeros campos representan la parte de la dirección de red y los últimos dos representan la parte local de la dirección. Estas redes pueden contener hasta 2 a la 16 dispositivos.
Direcciones clase C: Los primeros tres campos de la dirección representan la parte de la dirección red, y el restante representa la parte local, esta red permite que existan hasta 254 dispositivos, conectados a ella.

Ejemplo:

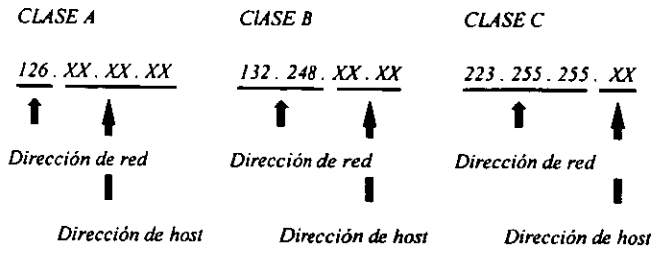


Figura 1.5 Clases de direcciones

Todas las subredes que constituyen a **Internet**, están interconectadas por una computadora, encargada de pasar los paquetes de información de una subred a otra. Esta computadora recibe el nombre de ***gateway***, estas computadoras *proveen todas las interconexiones entre redes físicas*.
El concepto de **gateway** en **TCP/IP** no debe ser confundido con el concepto de **gateway** que se definió en el capítulo anterior, el **gateway TCP/IP** es análogo al enrutador definido también en dicho capítulo. El paso de los paquetes de información que realiza el **gateway**, se le llama *encaminamiento o ruteo*.

Al incrementarse el número de máquinas en **Internet**, se hace mas compleja la entrega de paquetes entre las máquinas. Es por esto que los **gateways** enrutan paquetes, basándose en la localización de la red, no en la localización de la computadora destino. Evitando así la necesidad de que los **gateways**, sean máquinas con una gran capacidad de almacenamiento tanto en disco como en memoria, y si no realizara este tipo de encaminamiento, los **gateways** tendrían la necesidad de almacenar información referente a todas las máquinas existentes en **Internet.**

Conceptualmente **Internet** provee tres grupos de servicios, mostrados en el siguiente esquema:

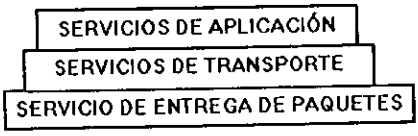


Figura 1.6 Grupos de servicio de Internet

Los grupos se han mostrado de esta manera, tratando de establecer la relación de dependencia entre ellos. Así es que en el nivel más bajo encontramos, al servicio de entrega de paquetes, que crea una plataforma en donde se basan los subsecuentes grupos. En el siguiente nivel encontramos al servicio de transporte, que crea una plataforma para que el ultimo nivel, que es el nivel donde corren las aplicaciones, pueda existir.
El **software** que hace posible la comunicación y el manejo de la red **Internet**, fue diseñado alrededor de estos tres grupos de servicios, y es por esto que la arquitectura de **Internet** es robusta y adaptable.
Para cada uno de estos grupos de servicios está asociado un protocolo. El modelo de capas de protocolo **TCP/IP** comparado con el modelo de protocolo **OSI** es el siguiente:

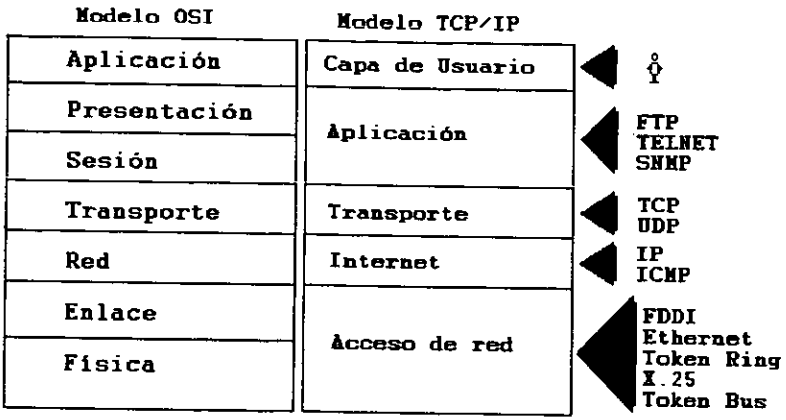


Figura 1.7 Modelo de capas TCP/IP comparado con el modelo de capas OSI

El modelo de capas de protocolo **TCP/IP**, cuenta con 5 capas, la capa de usuario y de aplicación conforman el *nivel de aplicación* para este modelo, ambas capas ofrecen una plataforma uniforme para que las aplicaciones puedan ejecutarse. Quien provee la interconexión en la red para los programas de aplicación, es el llamado *nivel de red,* conformado por las capas de transporte, **Internet,** y acceso de red, este nivel provee un mecanismo que entrega paquetes provenientes de la máquina fuente hacia la máquina. A continuación empezaremos el estudio de estos protocolos con el **Protocolo Internet o IP.**

1.4.2.1 El protocolo Internet IP.

El protocolo **Internet (IP)** es una parte medular en la arquitectura **TCP/IP**. El protocolo que define la entrega de paquetes en la arquitectura **Internet** es **IP.**
Cada uno de los paquetes es tratado de manera independiente, el protocolo hace el mejor esfuerzo para que los paquetes sean entregados, no descarta paquetes caprichosamente, esto ocurre cuando las fuentes o los destinos de los paquetes se encuentran deshabilitados, o han ocurrido fallas en la red.
Además este protocolo se encarga de la fragmentación de paquetes. La fragmentación se lleva a cabo cuando el paquete es muy grande y no puede navegar completo por la red, el protocolo se encarga también de re-ensamblarlo.
IP define la unidad básica de transferencia de datos y el formato exacto de todos ellos, además incluye un conjunto de reglas que especifican qué paquetes deben ser procesados y qué errores deben ser manejados. Traza caminos para enviar los paquetes a su destino.

El datagrama Internet

La unidad básica de transferencia para este protocolo es llamada ***Datagrama Internet*** o ***Datagrama IP***. El datagrama esta dividido en el área de encabezado y el área de datos. El encabezado del datagrama contiene la dirección fuente y la dirección destino, estas direcciones son de tipo **Internet.**
La siguiente es la forma general de un datagrama:

Figura 1.8 Forma general de un datagrama

Debido a que los datagramas son transferidos de una máquina a otra, es necesario transportarlos en ***frames físicos o paquetes de red***, que no son más que una analogía de los datagramas pero a nivel físico, los datagramas son manejados por **software** y los **frames** son reconocidos por **hardware**. Un paquete esta dividido igualmente en una área de encabezado y una área de datos, en el encabezado lleva igualmente la dirección fuente y la dirección destino, pero físicas. El datagrama al ser transportado forma parte del área de datos del paquete. A este proceso se le conoce como ***Encapsulamiento de datagrama.***

Existe un tamaño máximo del datagrama que debe ser contenido en un paquete, a este número se le conoce como ***MTU (maximum transfer unit).*** Para cada arquitectura de red existe un **MTU** diferente. Por ejemplo en las redes **Ethernet,** el límite es de 1500 octetos por paquete, y en las redes **proNET-10**, el límite es de 2000 octetos por paquete.
Debido a que en la red **Internet** no importa la arquitectura de las subredes que la conforman, el paquete deberá viajar por varias arquitecturas antes de llegar a su destino. Recordemos que para cada arquitectura se define un **MTU**, entonces el tamaño del paquete deberá ser modificado para que pueda viajar libremente por todas las arquitecturas. Si el paquete contiene un datagrama muy largo dentro de sí, el datagrama será "partido" en pequeñas piezas llamadas *fragmentos*, los cuales contendrán un **MTU** pequeño y de esta manera podrán viajar a través de la red sin ningún problema, y después *reensamblarse* en un datagrama completo. A este proceso se le conoce como ***fragmentación***, y es realizado por **IP**. En **Internet** si un datagrama se fragmenta, los fragmentos viajan a través de la red hasta llegar a su destino, y es entonces cuando el reensamble se lleva acabo; si alguno de los fragmentos se pierde el protocolo no podrá reestructurar el datagrama original. **Internet** no limita los datagramas a un tamaño especifico, pero sin embargo sugiere que todos los componentes de las subredes que conforman a **Internet (hosts y gateways),** estén preparados para manejar datagramas con un tamaño mayor a 576 octetos, sin que sean fragmentados.
El esquema real del datagrama es más complejo que el anteriormente mencionado. A continuación se presenta las divisiones de un datagrama:

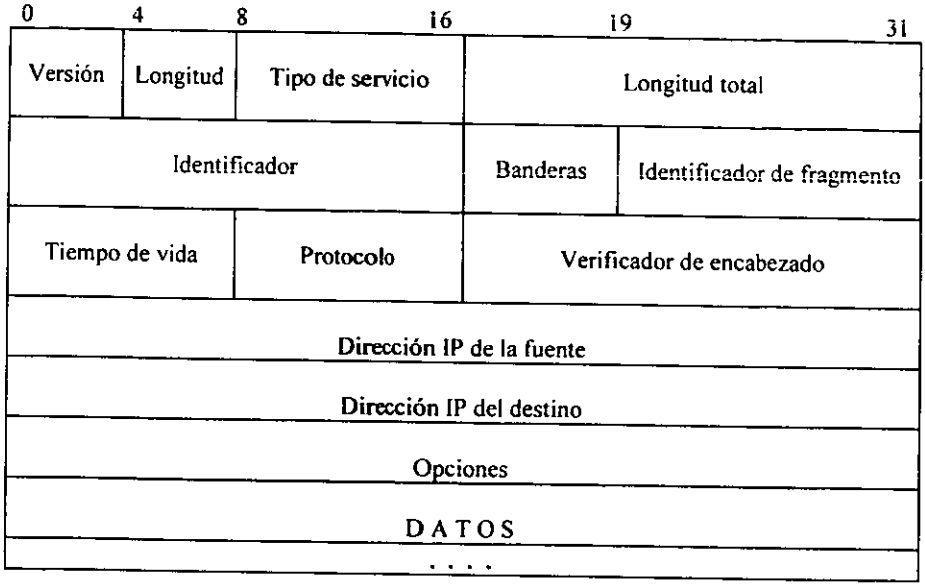


Figura 1.9 Esquema formal de un datagrama

- **Versión:** especifica la versión de IP que se está usando, checa que la máquina que envía el datagrama, la que lo recibe y el **gateway**, estén de acuerdo en el formato de éste.
- **Longitud:** Especifica el largo del encabezado del protocolo IP en palabras de 32-bits.
- **Tipo de servicio:** La especificación del tipo de servicio, actúa como una sugerencia al algoritmo de encaminamiento, ayudando a éste a escoger entre varios caminos que llevan al destino del datagrama. Tomando en consideración las tecnologías disponibles en estos caminos.
- **Longitud total:** Este campo muestra el tamaño del datagrama dividido en octetos, incluyendo el encabezado del mismo.

En caso de fragmentación, **IP** utiliza los siguientes tres campos dentro del datagrama, para tener el control sobre dicha fragmentación:

- **Identificador:** Es un identificador único para cada datagrama enviado, si la fragmentación se lleva a cabo, cada fragmento es reconocido de un mismo datagrama porque éste campo debe ser el mismo para todos los fragmentos.
- **Banderas:** Este campo contiene el estado del datagrama relacionado con la fragmentación. Esto es si el datagrama a sido fragmentado o no.

- **Identificador de fragmento:** Identificador de fragmento de un datagrama. Esto es, cuando un datagrama es fragmentado, cada uno de estos fragmentos recibe un número que lo identifica, para después poder re-ensamblar el datagrama.
- **Tiempo de vida:** Este campo define el tiempo que el paquete puede esperar en la red antes de ser descartado.
- **Protocolo:** Contiene el identificador del protocolo de transporte que se está usando.
- **Verificador de encabezado:** asegura la integridad del encabezado del datagrama.
- **Direcciones IP fuente y destino:** Direcciones de tipo **Internet** del transmisor y receptor del datagrama.
- **Opciones:** Este campo contiene las opciones que permiten un manejo más específico de los datagramas, por ejemplo se le puede indicar un camino definido para llegar al receptor a través de la red.
- **Datos:** Es el área donde se almacena la información que será enviada .

Ruteo de datagramas IP

Es importante conocer el algoritmo de ruteo que utilizan los **gateways**, para la entrega de paquetes a su destino final, ya que de esta manera conoceremos el aspecto operacional del protocolo **IP.**
El termino ***ruteo***, se refiere al proceso de selección de un camino sobre el cual serán enviados los paquetes al receptor.
El termino ***enrutador***, se refiere a cualquier máquina encargada de realizar esta selección.
El algoritmo de ruteo de **Internet**, selecciona el mejor camino a seguir basado en la ruta más corta.
hay dos tipos de ruteo: el ***ruteo directo*** y el ***ruteo indirecto***.

Ruteo directo: Este tipo de ruteo se realiza cuando hay una transmisión de datagramas **IP**, entre dos máquinas en una red física simple, es decir en esta red no se ven involucrados **gateways;** el transmisor encapsula el datagrama en un paquete o **frame** y envía este directamente a su destino o receptor. Todas las máquinas contenidas en una misma red física tienen el mismo identificador de red; así es que para que un paquete sea entregado, el transmisor extrae el identificador de red, de la dirección destino y la compara con el identificador de red de su propia dirección, si éstos son iguales quiere decir que las dos máquinas, (la receptora como la transmisora) están en la misma red, de esta manera se efectúa un ruteo directo de datagramas. Es posible que un **gateway** se vea involucrado en el ruteo directo, si la máquina transmisora tiene preestablecido a dicho **gateway** como su salida a otra red.

Ruteo indirecto: El ruteo indirecto de paquetes, es el que involucra más de un **gateway** para que los paquetes puedan llegar a su destino. Los **gateways** en **Internet,** forman una estructura cooperativa. Los datagramas pasan de **gateway** a **gateway** hasta encontrar uno que permita al datagrama ser entregado a su destino. Cuando un paquete llega a un **gateway** es analizado para determinar su dirección **IP**, entonces el **gateway** busca en su almacenamiento de direcciones cual es el camino viable para dicho paquete y lo envía al siguiente **gateway**. Cuando un datagrama pasa a través de uno o varios **gateways** y éstos

buscan el mejor camino para que dichos paquetes lleguen a su destino, se está efectuando un ruteo indirecto de datagramas.

Para realizar el ruteo de datagramas, el protocolo **IP** utiliza un algoritmo de decisión (***algoritmo de ruteo***), el cual se basa en la manipulación de tablas que contienen direcciones **Internet** de subredes y de **gateways**, estas tablas son llamadas ***tablas de ruteo.*** Todas la máquinas, tienen una tabla de ruteo, ésta almacena información referente a las posibles direcciones destino; la tabla no contiene el total de las rutas existentes en **Internet**, ya que se necesitaría una gran capacidad de almacenamiento en cada máquina. La información que contienen las tablas se reduce a las redes o **gateways** cercanos a dicha máquina.

El tamaño de la tabla de ruteo, es pequeño debido a que el algoritmo de ruteo basa sus decisiones en la localización de la RED destino No en la localización de la MÁQUINA destino.

Para ilustrar lo anterior, se presenta el siguiente diagrama:

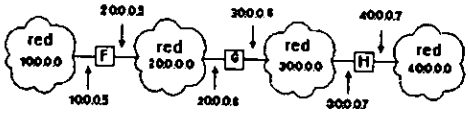


Tabla de ruteo del gateway G

Máquinas contenidas en la red	Ruteo por está dirección
20.0.0.0	ruteo DIRECTO
30.0.0.0	ruteo DIRECTO
10.0.0.0	gateway F
40.0.0.0	gateway H

Figura 1.10 Ejemplo de red Internet, con 4 redes y 3 gateways, junto con la tabla de ruteo del gateway G.

En el caso de que el algoritmo no logre encontrar una ruta en la tabla para la dirección destino del datagrama, el algoritmo envía el datagrama por una ruta predeterminada (**default router**).

Aunque hemos mencionado, que la entrega de paquetes se basa en la búsqueda de direcciones de la red destino, no por la dirección de la máquina destino; el protocolo **IP** permite definir rutas específicas para algunas máquinas, estas rutas son conocidas como *rutas estáticas.*

El algoritmo de ruteo de datagramas **IP**, funciona de la siguiente manera:

Extrae la dirección destino **(ID)** del datagrama.
Extrae de **ID** la dirección identificadora de red o dirección de red **(IN)**

Si **IN** aparece en la tabla de ruteo como una red directa:
Envía el datagrama a su destino sobre la red.
(encapsula el datagrama y envía el paquete)

En caso contrario
Si **ID** aparece como una máquina con ruta específica:
rutea el datagrama, por el camino que especifica la tabla de ruteo, encapsula el datagrama y envía el paquete.

En caso contrario
Si **IN** aparece en la tabla:
rutea el datagrama por donde lo especifica la tabla, encapsula el paquete y lo envía.

En caso contrario
Si **IN** no aparece el la tabla Y existe una ruta predeterminada:
rutea el datagrama por el camino predefinido, encapsula el paquete y lo envía.

En caso contrario
ERROR DE RUTEO.

1.4.2.2 Mensajes de control y error (ICMP)

En el sistema descrito anteriormente cada **gateway** y máquina operan de manera autónoma, cada cual haciendo sus propias funciones. Este es un sistema virtual y todo en él funciona sin mayor problema. Pero en realidad, influyen muchos factores para poder llevar a cabo una entrega de paquetes exitosa.

IP falla en la entrega de paquetes cuando la máquina destino está desconectada temporal o permanentemente de la red, cuando el tiempo de vida del datagrama expira, cuando el **gateway** está congestionado y no puede procesar más paquetes o cuando está congestión de paquetes afecta a la máquina destino.

Para permitir que las máquinas en **Internet** reporten errores o provean de información acerca de circunstancias inesperadas, **IP** contiene un mecanismo conocido como ***Internet Control Message Protocol (ICMP),*** o Protocolo de Control de Mensajes **Internet. ICMP** es una parte fundamental de IP y está incluido en todas la implementaciones del mismo.

*ICMP provee comunicación entre el protocolo **IP** contenido en una máquina y el protocolo **IP** contenido en otra, alertando e informando mutuamente de posibles errores y mensajes de control.*

Cuando se genera un problema en la entrega de un paquete, **ICMP** genera un mensaje, que viaja a través de **Internet** en la parte de datos del datagrama. Los datagramas que contienen mensajes **ICMP** , son ruteados exactamente igual que los datagramas que contienen información, pero los mensajes de error pueden también ser perdidos o descartados , se hace una excepción si un datagrama conteniendo un mensaje de **ICMP** causa un error, la excepción establece que todos los mensajes **ICMP** no son generados por errores que resultan de datagramas conteniendo mensajes **ICMP**. Los mensajes **ICMP** , son encapsulados y enviados usando **IP**.
Cada mensaje **ICMP** tiene su propio formato, pero todos los mensajes empiezan con tres campos:

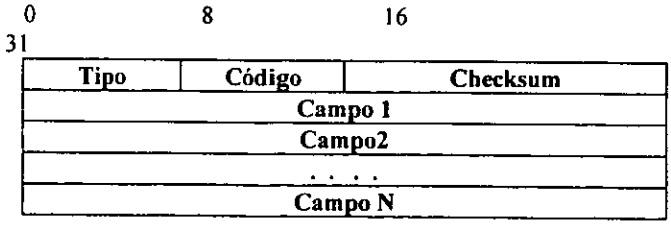


Figura 1.11 Formato general de mensajes ICMP

- **Tipo:** Define el significado del mensaje, y el formato del resto del paquete.
- **Código:** Proporciona información complementaria, acerca del tipo del mensaje.
- **Checksum:** Al igual que en el formato del datagrama **IP**, el campo **checksum**, asegura la integridad del encabezado del mensaje.

Los tipos de mensaje que puede generar **ICMP,** son:

IDENTIFICADOR DE CAMPO	TIPO DE MENSAJE ICMP
0	**Echo reply** , repetición de respuesta
3	**Destination unreachable,** destino inaccesible
4	**Source quench,** Fuente congestionada
5	**Redirect,** Redirección (cambio de ruta)
8	**Echo request,** repetición de petición
11	**Time exceeded for a datagram,** Tiempo de vida terminado para un datagrama
12	**Parameter problem on a datagram,** Problema de parámetro dentro de un datagrama.
13	**Timestamp request,** petición de tiempo,para sincronía
14	**Timestamp reply,** respuesta para la petición de tiempo
15	**Information request,** Petición de información
16	**Information reply,** Respuesta a la petición de Información
17	**Address mask request,** Petición de mascara
18	**Address mask reply,** Respuesta de mascara

Tabla 1.3 Tipos de mensaje ICMP

Pruebas de conectividad y estado de la máquina destino

Una máquina o un **gateway**, envían un mensaje **ICMP** , para cerciorase que la máquina destino se encuentre "viva" y la conectividad con ella sea óptima, este mensaje es del tipo 8 **(echo request).** Cualquier máquina que recibe un mensaje de este tipo, debe de formular una contestación y enviarla a la máquina o **gateway**, que envió el mensaje. El mensaje respuesta a un mensaje **echo request,** es del tipo 0 **(echo reply).**
Éstos mensajes tienen los tres campos que anteriormente mencionamos. El campo TIPO, especifica de que mensaje se trata, y su valor puede ser de petición (8) o de respuesta (0).El

campo CÓDIGO, tendrá por valor cero (0). Además de estos campos todos los mensajes de este tipo incluyen:
Los campos IDENTIFICADOR y NUMERO DE SECUENCIA; son usados por la máquina destino, para poder enviar la respuesta a la máquina fuente.
El campo de DATOS OPCIONALES, contiene los datos que la máquina fuente, envía a la máquina destino, a su vez la máquina destino envía los mismos datos sin modificación a la máquina fuente, cuando hace la contestación.
Este mecanismo de petición/respuesta, es una de las herramientas más usadas dentro de **Internet**.

Reportes de destinos inaccesibles

Cuando un **gateway** no puede entregar un datagrama **IP**, éste envía un mensaje de "destino inaccesible" a la máquina fuente. Los destinos pueden ser inaccesibles, porque el **hardware**, este temporalmente fuera de servicio (máquinas o **gateways**), porque la dirección destino no exista, o bien porque el **gateway** no tenga una ruta para la máquina destino, este ultimo caso se da en muy raras ocasiones.
Este mensaje incluye el encabezado y un pedazo del datagrama que no pudo ser entregado, con el fin de identificar totalmente a éste.
El campo CÓDIGO, en este formato puede contener un número entero, el cual puede aportar más información acerca del problema por el cual el datagrama no pudo ser entregado. Estos número pueden ser:

VALOR DE CÓDIGO	SIGNIFICADO
0	Red inaccesible
1	Máquina inaccesible
2	Protocolo inaccesible
3	Puerto inaccesible
4	Fragmentación necesaria y DF definido
5	Falla en ruta fuente

Tabla 1.4 Códigos de destinos inaccesibles

Además de las causas anteriormente descritas, la entrega de datagramas puede no llevarse a cabo cuando una fragmentación es necesaria y el datagrama tiene definido el bit DF (**don't fragment** o no fragmentación); de esta manera no le es permitida a **IP** la fragmentación, imposibilitando la entrega del datagrama. Otra causa, se da cuando hay una ruta predefinida para la máquina fuente y esta ruta no está correcta.

Control del flujo del datagrama

Cuando los datagramas llegan demasiado rápido a una máquina o a un **gateway**, éstos puede ser descartados. La máquina que descarta a los datagramas envía un mensaje **ICMP**; este mensaje es del tipo 4 **source quench.** Regularmente la máquina envía un mensaje por cada datagrama que es descartado.
El campo CÓDIGO debe tener valor cero. En este mensaje también se incluye el encabezado **Internet**, y un pedazo del datagrama que no pudo ser entregado.

Peticiones de cambio de ruta desde gateways

Usualmente los **gateways,** conocen las mejores rutas a cualquier máquina que esté cerca de ellos; cuando un **gateway,** detecta que alguna máquina esta usando una ruta "no-optima", éste envía un mensaje **ICMP** de redirección (5 **redirect**), este mensaje viene acompañado del datagrama original que envío la máquina por la ruta no-optima. Con el mensaje enviado, el **gateway** no soluciona el problema de la mala utilización de las rutas por la máquina.

*Los **gateways**, sólo envían mensajes de redirección a máquinas en redes simples, y jamás envían estos mensajes a otros **gateways**.*

El campo DIRECCIÓN GATEWAY, contiene la dirección **Internet** del **gateway**, que la máquina fuente utilizaba para buscar el destino del datagrama.
En este mensaje también se incluye el encabezado **Internet**, y un pedazo del datagrama que no pudo ser entregado.
El campo CÓDIGO, en este formato puede contener un número entero, el cual puede aportar más información a la máquina destino de cómo interpretar el mensaje. Los valores pueden ser los siguientes:

VALOR DE CÓDIGO	SIGNIFICADO
0	Redirección de datagrama por la red
1	Redirección de datagrama por la máquina
2	Redirección de datagrama por el tipo de servicio y por la red
3	Redirección de datagrama por el tipo de servicio y por la máquina.

Tabla 1.5 Códigos de redirección

Tiempos excedidos para datagramas

Cada datagrama **IP**, contiene un contador de tiempo-de-vida del mismo, generalmente llamado **hop count**, o contador de salto. Para prever que algunos datagramas circulen por siempre en la red **Internet** cuando un datagrama pasa por un **gateway** éste decrementa el contador de salto del datagrama.
Cuando un **gateway**, encuentra un contador de salto en cero, el datagrama es descartado, y se envía un mensaje **ICMP** de tiempo excedido (**11 time exeeded**).

En el caso de ocurrir una fragmentación del datagrama, la máquina que recibe éste, inicia un contador al arribo del primer fragmento del datagrama, si el contador expira antes de la llegada de todos los fragmentos, la máquina también envía un mensaje de error tiempo excedido.

El valor del campo CÓDIGO en este formato, puede tener los siguientes valores:

VALOR DE CÓDIGO	SIGNIFICADO
0	Tiempo de vida excedido para el datagrama
1	Tiempo de reensamblaje de fragmentos excedido

Tabla 1.6 Tiempos excedidos

En este mensaje también se incluye el encabezado **Internet**, y un pedazo del datagrama que no pudo ser entregado.

Errores en el contenido del datagrama.

Cuando un **gateway** o máquina encuentra problemas con el encabezado de una datagrama, éstos envían un mensaje de error **ICMP** de tipo 12 (**Parameter Problem on a Datagram**, o problema de parámetro dentro de un datagrama), a la máquina que envía el datagrama.
En el formato de estos mensajes, se incluye además el campo POINTER, que contiene el octeto de datagrama que está causando problemas.

Sincronización de relojes entre máquinas

Una máquina que desee enviar un paquete de información, manda un mensaje de tipo **timestamp request** o petición de tiempo para sincronia, a la máquina receptora de dicho paquete, para que la máquina receptora a su vez le responda mandando un mensaje de tipo **timestamp replay** o respuesta a la petición de tiempo, donde le envía el tiempo corriente de ésta máquina.

Este proceso sirve para que la máquina transmisora pueda realizar los cálculos necesarios con los que estimará el tiempo que puede tardarse en viajar un paquete desde ella hasta la

máquina receptora, así como también sincronizar los relojes de ambas. Éstos cálculos pueden ser llevados a cabo ya que cuando la máquina receptora envía el mensaje de respuesta, en él manda el tiempo en el cual fue enviado a ella el mensaje **timestamp request,** además del tiempo en que recibió dicho mensaje, y el tiempo en que ella realizo la transmisión de la respuesta.

Obtención de la dirección de red.

Las máquinas utilizan el mensaje **ICMP information request** o petición de información, para obtener la dirección **Internet** de la red con la que desean tener enlace. Los **Gateways** responden a dicha petición enviando un mensaje **ICMP information reply** o respuesta a la petición de información, en donde envían la dirección, que la máquina transmisora del mensaje pidió.

Obtención de la máscara de subred.

Además de las direcciones **IP** para cada máquina perteneciente a **Internet,** existe otro tipo de dirección llamada ***Máscara de subred,*** la cual permite que varias redes físicas compartan la misma dirección de red.
Cuando una máquina desea enviar un paquete a otra máquina que está en una red física diferente a la que ella pertenece, la máquina transmisora envía un mensaje **ICMP address mask request** o petición de máscara, en donde pide al **gateway** que hace posible la conexión con la red de la máquina receptora, que le envíe dicha dirección. El **gateway** por su parte envía un mensaje de tipo **address mask reply** o respuesta a la petición de máscara, en donde envía la dirección pedida.

1.4.2.3 Protocolos de capa de transporte, TCP y UDP

Protocolo de control de transmisión TCP

TCP es un protocolo de capa de transporte, por lo tanto ***TCP es un protocolo de transmisión*** que sienta la base para que el sistema de entrega de paquetes (**IP**) pueda hacer su trabajo, contrario a lo que parezca **TCP** no es una parte integral de **IP**, TCP es un protocolo independiente de propósito general que puede ser usado con otros sistemas de entrega de paquetes; debido su versatilidad , es considerado como el más popular de los protocolos de transporte y es implementado desde redes muy sencillas como las de área local o en redes muy grandes y complejas como lo es **Internet**.

Los protocolos de entrega de paquetes que pertenecen a las capas bajas de los modelos de protocolo, como IP por ejemplo, se encargan de establecer rutas a seguir para la entrega de la información, también de fragmentar los paquetes para que puedan viajar por diferentes redes, pero esta información puede perderse o ser destruida cuando un error de transmisión ocurre. El trabajo para el protocolo de RED, en este caso IP, se puede hacer bastante grande aunando a lo que ya hace, el asegurar que los paquetes sean entregados en su destino

teniendo en cuenta todas las tareas que deberá realizar para alcanzar este propósito, como son verificar que los paquetes no estén corruptos o duplicados, si hay una pérdida de información en el paquete o bień si el paquete se pierde re-enviarlo, además de manejar los mensajes de control para las capas de protocolo superiores e inferiores. Debemos considerar también que las capas de aplicación necesitan que la información que generan o que necesitan sea transmitida al programa correcto en la máquina destino, y en esta máquina no hay un solo programa requiriendo o enviando información, así pues fue necesario desarrollar un protocolo que se hiciera cargo de todas estas tareas de una manera eficiente y proporcionar una entrega de paquetes confiable a nivel de transmisión. Es importante hacer notar que no todos los modelos de protocolo usan un protocolo de transporte.

Conexiones y puertos

Decimos que **TCP** provee una interface entre los programas de aplicación cuando transfieren datos, ya que **TCP** se encarga de crear un circuito virtual de conexión entre ambas aplicaciones, dicho circuito es parecido a una llamada telefónica. Antes que la transferencia empiece, se debe crear este circuito, la aplicación debe interactuar con su respectivo sistema operativo informándole de su necesidad de transmisión de datos, la máquina transmisora deberá hacer una llamada a la máquinas destino la cual deberá ser aceptada, entonces los protocolos en ambos sistemas operativos se comunican por medio de mensajes a través de la red, verificando que la conexión está autorizada, y ambas partes están listas, después de dichas verificaciones el protocolo anuncia a las aplicaciones que el circuito virtual de conexión está hecho y que la transmisión de datos puede empezar.

Además **TCP** provee una conexión llamada ***full duplex***, lo que significa que en el circuito virtual que genera los datos pueden viajar en ambas direcciones, la ventaja de una conexión *full duplex* es que el protocolo puede enviar información de control de flujo en una dirección, y al mismo tiempo enviar los paquetes conteniendo los datos en la dirección opuesta.

TCP permite que múltiples programas de aplicación en una máquina puedan comunicarse concurrentemente con otros programas de aplicación en otras máquinas, demultiplexando los datos que llegan a la máquina a sus respectivas aplicaciones. **TCP** incorpora objetos abstractos llamados *puertos* para identificar el destino final (las aplicaciones) de un paquete de datos en una máquina. Cada puerto es identificado por un número entero, este numero es asignado localmente y es único en cada máquina. Por lo tanto el destino de todos los paquetes de datos que forman parte del trafico TCP, es especificado por la dirección IP de la máquina y por el número de puerto asociado a una aplicación especifica dentro de la máquina.

El segmento TCP

Los datos son enviados de una aplicación a otra en conjuntos de octetos de 8 bits o bytes llamados ***streams***, como el tamaño de los **streams** no esta determinado, **TCP** es libre para dividir a un **stream** en sus propios paquetes de transmisión TCP. Este paquete **TCP** es

conocido como ***segmento***, los segmentos son utilizados por **TCP** para establecer conexiones, para transferir mensajes de reconocimiento ACK, para cerrar conexiones etc., no solo para enviar **streams**. Usualmente cada segmento viaja a través de **Internet** en un solo paquete IP.
Los segmentos tienen su propio formato como los paquetes **IP**. Entre los campos que contiene un segmento son los siguientes:

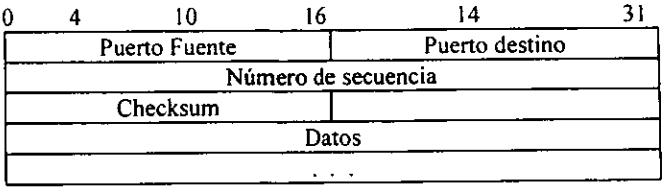


Figura 1.12 Segmento TCP

Los campos ***puerto destino*** y ***puerto fuente*** contienen los puertos de la aplicación que transmite los datos o las peticiones y de la aplicación que deberá recibirlos.
Como los **streams** son divididos para poder ser almacenados en los segmentos, en el campo ***numero de secuencia*** se identifica la posición de los datos contenidos en el segmento dentro de el **stream** original. Los segmentos, como todos los paquetes que viajan a través de una red, contienen un campo para verificar la integridad del segmento, este campo recibe el nombre de campo ***checksum*** ; los campos anteriores son los mas importantes en un segmento, aparte de estos se encuentra el área de datos, en donde viajan los mismos.

Entrega de datos garantizada

En este punto debemos mencionar que la entrega de **streams** hecha por **TCP**, esta garantizada: *Todo **stream** enviado debe llegar a la aplicación destino sin duplicación, y sin pérdida de datos*. Esta entrega garantizada **TCP**, la logra con una técnica de envío de mensajes de reconocimiento y retransmisión de paquetes llamada ***behind sliding windows*** que a continuación explicaremos.
La técnica **behind sliding windows** está basada a su vez en otra mas sencilla llamada *reconocimiento positivo simple*, esta técnica consiste en que la máquina que es transmisora enviará un paquete a la máquina receptora y esperará a que ésta lo reciba y envíe un mensaje de reconocimiento **(ACK)** a la máquina transmisora el cual le indicará que el paquete llegó si problemas y que es tiempo de enviar el siguiente paquete y así sucesivamente hasta terminar con todos los paquetes. La máquina transmisora inicia un contador cuando envía el paquete, si el contador expira antes de que el mensaje de reconocimiento **(ACK)** de la máquina receptora llegue, entonces la máquina transmisora reenvía el paquete. A continuación se muestra gráficamente la técnica de reconocimiento positivo simple.

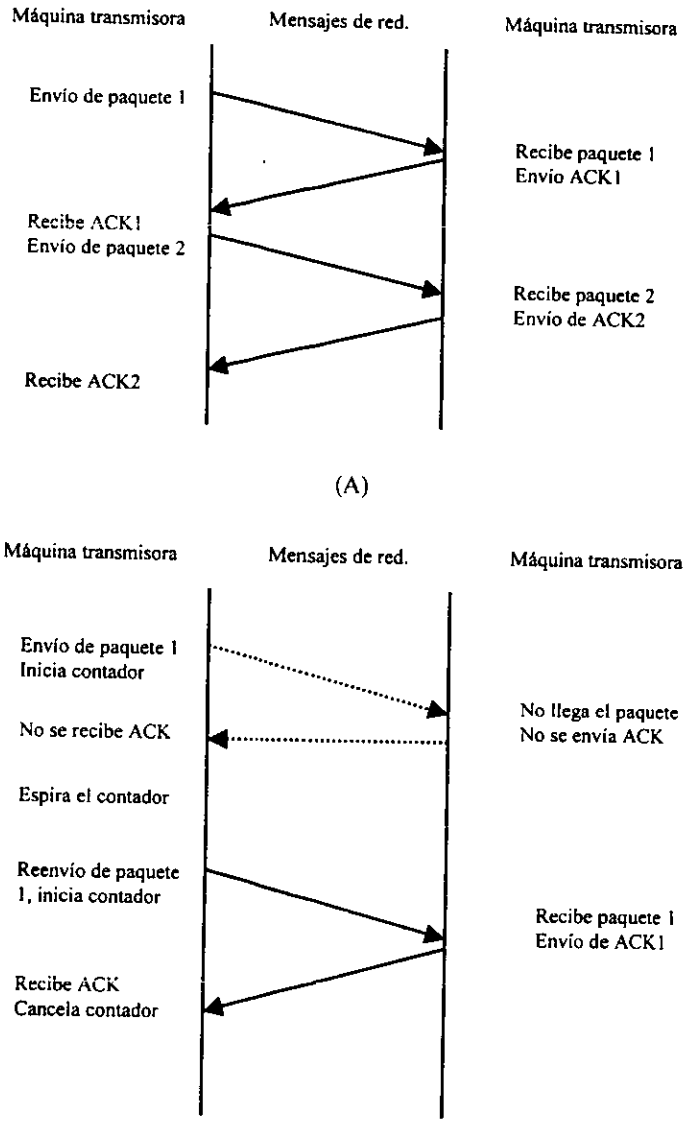


Figura 1.13 (A) Protocolo con reconocimiento positivo en el que se espera el reconocimiento para cada paquete enviado. (B) Retransmisión al expirar el contador que ocasiona que se considere el paquete como perdido

Como se ve, al enviar a los paquetes uno por uno y esperar el mensaje de reconocimiento para cada uno, la red no se utiliza a su máximo, por el contrario se desperdician recursos. La técnica ***sliding window*** sigue el mismo patrón de envío y espera de mensajes de reconocimiento para los paquetes, pero en vez de enviar un solo paquete se envían varios paquetes antes de esperar un mensaje de reconocimiento; el número de paquetes a enviar es conocido como *tamaño de ventana*. En la figura siguiente el tamaño de ventana es tres:

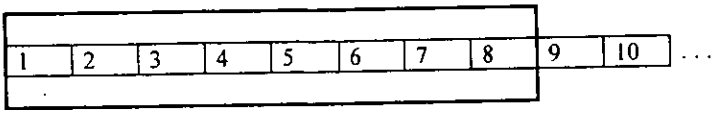


Figura 1.14 (A) Sliding window con 8 paquetes en la ventana (Posición inicial)

Para este caso, se envían los tres primeros paquetes, el cuarto paquete podrá ser enviado cuando el mensaje de reconocimiento del primer paquete sea recibido. Se le llama técnica de reconocimiento de ventana, ya que virtualmente hay una ventana que encierra a varios paquetes, que son los que serán transmitidos al principio, después la ventana se irá recorriendo para encerrar al siguiente paquete y transmitirlo cuando llegue un mensaje de reconocimiento .

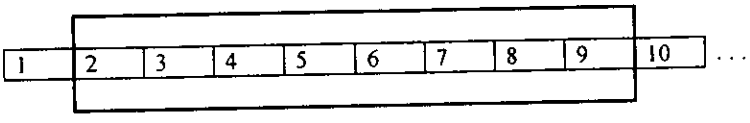


Figura 1.14 (B) La ventana se desliza al recibir el reconocimiento del paquete 1, de esta forma el paquete 9 puede ser enviado.

TCP en la máquina transmisora activará y almacenará un contador para cada paquete transmitido, si un mensaje **ACK** asociado a un paquete no llega y el contador expira, entonces **TCP** retransmite el mensaje. A continuación se muestra de manera gráfica la técnica de **sliding window.**

La idea de la retransmisión de paquetes, basada en un sistema de reconocimiento, es de las más importantes y hacen a **TCP** un protocolo confiable y popular.

UDP

En el esquema **TCP/IP** existe un protocolo de transporte que también es muy conocido y utilizado, se trata de **UDP (User Datagram Protocol)**. **UDP** es también un protocolo de capa de transporte y es utilizado por las aplicaciones para transferir datos de una máquina a

otra y encontrar la aplicación a quien van dirigidos los datos, se basa en **IP** para hacer dicha entrega. **UDP** no usa mensajes de reconocimiento para asegurarse que los paquetes de datos enviados fueron recibidos, tampoco los ordena, ni sigue el flujo de paquetes entre la máquina receptora y la máquina transmisora, así es que los paquetes **UDP** pueden ser perdidos, duplicados o pueden arribar fuera de orden. Sin embargo de esta manera los paquetes llegan más rápido. Entonces podemos decir que: ***UDP provee una entrega de datos no garantizada.***

Formato de paquetes UDP

Cada paquete **UDP** es llamado *datagrama* y consiste en dos partes: El encabezado y el área de datos.

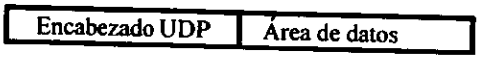


figura 1.15 Datagrama UDP

El encabezado es dividido en 4 campos de 16 bits como se muestra en la figura, el puerto por el cual el mensaje fue enviado, el puerto al cual el mensaje es destinado, la longitud del mismo y el **Cheksum.**

figura 1.16 Encabezado del datagram Udp

El campo *puerto fuente* y *puerto destino* contiene números de puerto que **UDP** utiliza para demultiplexar datagramas a procesos que están esperando recibirlos, así se crea una transferencia bidireccional entre el puerto fuente y el puerto destino. El campo *longitud* contiene el tamaño de todo el datagrama en octetos. El campo **checksum** contiene un número con el que **UDP** verifica la integridad de la información que viaja en el datagrama.

Entrega de datos UDP

Hemos visto que existen varias capas de protocolo en el esquema **TCP/IP**, estas capas están ordenas jerárquicamente así es que el software deberá enviar sus datos a través de ellas para que finalmente sea recibido por el software que espera esta información en la máquina

destino. **UDP** provee otra manera de entrega de datos para la capa de transporte. A esta entrega de datagramas se le llama *demultiplexión*. **UDP** recibe el datagrama de la capa de red (**IP**) y la demultiplexa es decir la envía al software destino basado en el número de puerto.

En apariencia cada capa de protocolo es independiente, pero en la practica cada una está fuertemente ligada a la otra e interactuan entre si.

1.4.3 Netware

Dentro de REDII, contamos también con el servicio de una red **Novell Netware** . La estructura de una red **Netware** consta de varias computadoras llamadas ***estaciones de trabajo*** que hacen peticiones de servicios a otra computadora llamada ***servidor*** que almacena información y presta dichos servicios a las demás. En general llamaremos *nodo Netware* a cualquier computadora que conforma una red **Novell Netware** .

Los protocolos nativos de **Netware,** son tres: **IPX, SPX y NCP** . A continuación se presenta la comparación entre el modelo de capas **Netware** y el modelo **OSI.**

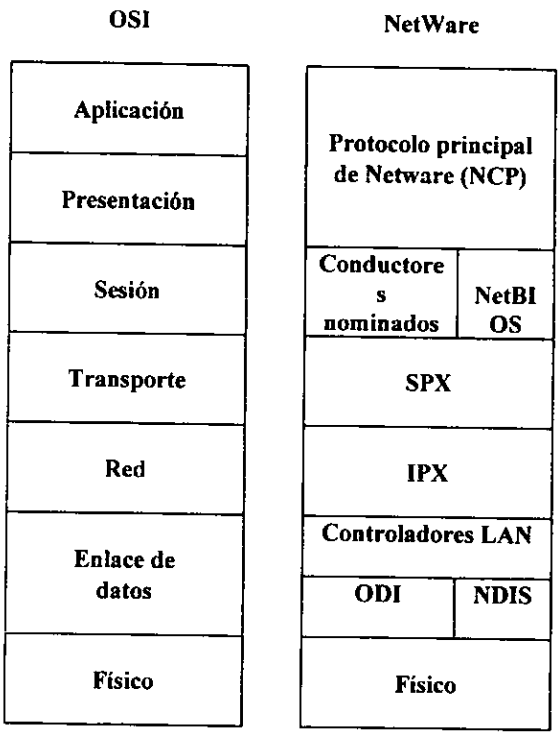


Figura 1.17 Comparación modelo OSI contra Netware

Internet Packet Exchange, IPX

El protocolo **IPX**, es un protocolo de capa de red, por lo tanto su función es la entrega de datos contenidos en paquetes con una estructura particular llamados ***datagramas*** o ***paquetes IPX***, cuando dichos paquetes no pueden viajar completos por la red, deben ser fragmentados, tarea conocida como ***fragmentación*** que también corresponde a este protocolo.
Para que **IPX** pueda entregar los datagramas a su destino, **Netware** tiene un sistema de direccionamiento constituido por direcciones para cada nodo de la red. La dirección para un nodo **Netware** consiste en dos partes: la parte de dirección de red con 32 bits y la parte de dirección de nodo con 48 bits haciendo un total de 10 bytes por dirección. Usando estas direcciones es como los datagramas se enrutan y llegan a su destino.

A continuación se presenta el esquema de un datagrama **IPX.**

2 bytes	2 bytes	1 byte	1 byte	4 bytes	6 bytes	2 bytes	4 bytes	6 bytes	2 bytes	
Checksum	**Longitud**	**Control de transporte (heps)**	**Tipo de paquete**	**Red destino**	**Nodo destino**	**Socket destino**	**Red fuente**	**Nodo fuente**	**Socket fuente**	**Datos**

Figura 1.18 Datagrama IPX

Datagrama IPX

El primer campo del datagrama **(Checksum),** es utilizado por **IPX** para constatar la integridad de la información que viaja en el mismo; el segundo campo (longitud) se refiere a longitud del datagrama **IPX** en bytes, el cual debe ser como límite de 576 bytes; el siguiente campo (control de transporte) guarda el número de rutas por donde viajó el paquete **IPX**, si este número llega a 16 entonces el paquete es desechado; el cuarto campo (Tipo de paquete) identifica cual de las capas superiores en el modelo recibirán la información que viaja en la porción de datos del paquete, por ejemplo para hacer referencia a la capa en donde reside **SPX** , este campo tendrá un valor de código 5, para **NCP** el código será el 17, el código 0 es reservado para destinos desconocidos.
En un nodo **Netware** puede haber varios procesos corriendo, para enviar los datos transportador por el paquete **IPX** a los procesos , se crea un número único que identifica a cada uno de estos, dicho número es conocido como ***número de socket*** que tiene por longitud 16-bits y es asignado a cada proceso que esta en espera o ejecutándose y para comunicarse usa los servicios de **IPX**. Los campos Red destino, nodo destino y socket destino, están formados por una dirección de nodo **Netware** junto con un número de socket que identifican a que máquina y a qué proceso va la información contenida en el paquete,

de la misma manera los campos Red fuente, nodo fuente y socket fuente identifican de que máquina y de qué proceso viene el paquete.
Además de ésto la dirección de red destino juega un papel muy importante ya que es usada para decidir si el paquete **IPX** deberá ser enviado localmente o bien se enviará a un ruteador.
IPX al entregar los paquetes no siempre se encuentra que la dirección destino está en una misma red, de ser así **IPX** se encargará de ***rutear*** estos paquetes a su destino a través del camino más conveniente para ello. La manera como realiza esta tarea **IPX**, es enviando un paquete de prueba de ruta, si un ruteador **IPX** toma este mensaje entonces regresa una respuesta a la máquina originadora del mensaje, dicha respuesta contiene la dirección del ruteador **IPX**, el paquete **IPX** es entonces enviado por esta ruta. Los ruteadores **IPX** guardan tablas de ruteo conteniendo la información de ruteo de todas las redes a su alcance, estas tablas son refrescadas cada 60 segundos.

Sequenced Packet Exchange (SPX)

El protocolo **SPX** es un protocolo de capa de transporte y es el responsable de establecer un circuito virtual de conexión entre dos máquinas cuando es necesario. Se dice que se establece dicho circuito cuando para una transmisión de datos, **SPX** envía paquetes de control para establecer una conexión, entonces un número identificador (ID) es asignado a este circuito virtual. Cuando la transmisión termina es enviado un paquete de control anunciando el final y la conexión es terminada.
Cuando una conexión es establecida, **SPX** toma el control de flujo de paquetes y les proporciona una secuencia para asegurar que los mismos arribaran a su destino con un orden; además asegura que en la máquina destino el **buffer**[2] no esté lleno con datos que estén llegando demasiado rápido.
SPX utiliza un esquema de reconocimiento para asegurarse de que los paquetes llegan a su destino, este esquema consiste en enviar una paquete de reconocimiento hacia la maquina destino, la que a su vez deberá de enviar un paquete de respuesta que indica a **SPX** iniciar la transmisión, dentro de este esquema **SPX** utiliza un algoritmo de tiempo fuera[3] para decidir cuando un paquete necesita retransmitirse, dicho algoritmo es ajustado dinámicamente basándose en los retardos anteriores experimentados en esa específica transmisión de paquetes.
SPX utiliza también una estructura especial en los paquetes que él transmite, la siguiente figura nos muestra esta estructura.

[2] Memoria intermedia. Porción reservada de la memoria que se utiliza para almacenar datos mientras son procesados.
[3] Tiempo fuera, se le llama cuando un paquete a llegado al límite establecido de espera para o en su transmisión

1 byte	1 byte	2 bytes	2 bytes	2 bytes	2 bytes	2 bytes	
Control de conección	Tipo de flujo de datos	Conección ID fuente	Conección ID destino	Número de secuencia	Número de reconocimiento	Número de localidad	Datos

Figura 1.19 Paquete SPX

El primer campo (control de conexión) sirve para la regulación del flujo de datos, en este campo se envían códigos de bits que indican una tarea especifica, por ejemplo se envían códigos de final de conexión o códigos que indican que una petición de reconocimiento fue hecha. El siguiente campo (tipo de flujo de datos) indica la naturaleza de los datos contenidos en el campo de datos del paquete **SPX**, y es usado para identificar la capa superior en la cual **SPX** deberá entregar la información que porta el paquete, este servicio es similar a el campo "tipo de paquete" en el paquete **IPX**. Los campos conexión ID de la fuente y conexión ID del destino, son los números que identifican el circuito virtual usados para identificar una transmisión. El quinto campo (número de secuencia), **SPX** numera todos los paquetes enviados en una transmisión dándoles un numero secuencial, y es almacenado en este campo, **SPX** usa esta numeración para detectar paquetes perdidos o bien fuera de secuencia. El siguiente campo (número de reconocimiento) es usado para indicar cual es el siguiente paquete que la máquina destino deberá recibir. El séptimo campo (numero de localidad) indica cuantos **buffers** libres hay en la máquina destino en la conexión, **SPX** utiliza este número para ayudar a no saturar a los **buffers** con pautas que no tengan cabida en ellos y estén en espera de que sean desocupados. Finalmente tenemos el campo de datos en el cual viajan los datos que no son de control, es decir es la información como tal que deberá ser entregada a la máquina destino en una transmisión.
En general los nodos **Netware** no usan **SPX**, estos generalmente usan **IPX** directamente. Sin embargo **SPX** es usado para establecer conexiones remotas entre un servidor de impresión y las impresoras, y en algunos casos de manejo de bases de datos por ejemplo con **SQL Netware**, además de conexiones remotas con el servidor de archivos a través de un comando especial llamado **RCONSOLE** .

Netware Core Protocol, NCP

El protocolo **NCP** es muy importante en el esquema **Netware** ya que es el encargado de implementar los servicios de archivos, impresión, seguridad en archivos, entre otros.

NCP es implementado tanto en una estación de trabajo como en un servidor. En una estación de trabajo **NCP** es limitado a crear peticiones de servicios para un servidor **NCP** . Por otra parte en un servidor está la implementación completa de **NCP** que puede ejecutar o

procesar peticiones. **NCP** provee servicios totalmente transparentes para las estaciones de trabajo.

NCP usa directamente a **IPX** para la entrega de paquetes, haciendo a un lado a **SPX** en protocolo de transporte. **NCP** tiene su propio mecanismo de control de sesiones, detección de errores, y retransmisión de paquetes por lo tanto también maneja su propio formato de paquete **NCP** que se muestra a continuación.

2 bytes	2 bytes	2 bytes	2 bytes	2 bytes	
Tipo de petición	Número de secuencia	Número de prueba	Reservado	Código de servicio	Datos

Figura 1.20 Paquete NCP

El campo tipo de petición indica el tipo de petición **NCP** hecha por una estación de trabajo, ejemplos de peticiones **NCP** son: Petición de conexión, petición de espacio en un buffer, salida de sesión, petición de fecha y hora, final de trabajo. El número de secuencia es usado como un identificador (ID) de esa especial conexión para contestar una petición, e identifica una petición **NCP** y su correspondiente respuesta. El código de servicio identifica la petición de servicio hecha por la estación de trabajo. Finalmente el campo datos almacena la información que será transferida.

1.4.4 NFS

NFS llamado así por sus siglas en inglés (**Network File System**), fue desarrollado por la empresa **SUN Mycrosystems Inc.** en 1983 y fue diseñado para proveer una conectividad entre computadoras con diferentes manufacturas y trabajando bajo diferentes sistemas operativos, con la finalidad de compartir recursos entre ellas.

Usualmente **NFS** es proveído como una extensión del sistema operativo y debe ser adquirido por separado y adicionarlo al sistema en uso. Una excepción a esta regla son las estaciones de trabajo trabajando con sistema operativo **UNIX**, ya que incluye a **NFS** como parte de su configuración básica. Otros sistemas operativos que pueden manejar **NFS** son MS-DOS, VMS, MVS y Windows 95.

NFS sigue el modelo cliente-servidor[4] en sus implementaciones. El servidor **NFS** pone a disposición "pedazos" de disco llamados **File Systems** o Sistemas de Archivo (sistemas de archivos exportados), que los clientes **NFS** pueden acceder e incorporarlos a sus máquinas como si fueran discos locales propios (sistemas de archivos montados). Los usuarios de las máquinas con dichos clientes, pueden acceder a los discos virtuales , de una manera totalmente transparente, esta transparencia se lleva a cabo exitosamente gracias a la velocidad con que **NFS** realiza el acceso de datos sobre la red, el usuario usualmente no encuentra diferencia entre el acceso a disco virtual y el acceso a disco local.

El producto **NFS** consiste en una gama de protocolos que hacen posible su funcionamiento. Estos protocolos tienen diversas tareas asociadas a ellos, existe un protocolo especial llamado *protocolo NFS,* que hace realidad el acceso a los archivos remotos.

Protocolos que usa NFS

NFS está basado en un modelo de capas de protocolo correspondiente al modelo **OSI.**
En la siguiente tabla se muestra la relación entre los protocolos que **NFS** utiliza y las capas de **OSI**

[4] El término *servidor* es aplicado a un programa en una máquina determinada, que ofrezca un servicio el cual puede ser requerido sobre una red, mientras que un cliente es un programa que efectúa peticiones a un servidor y espera una respuesta, el *modelo cliente-servidor* sigue un formato de petición respuesta que involucra a uno o más *clientes* y a un *servidor.*

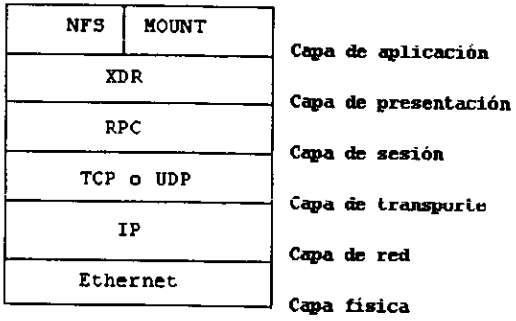


Figura 1.21 Protocolos usados por NFS

NFS, emplea todos los protocolos que aparecen en la figura para funcionar, cada uno de ellos tiene tareas especificas.
El transporte y entrega de paquetes se realiza por medio de **TCP o UDP e IP,** en las capas de transporte y de red.

Llamadas a procedimientos remotos, RPC

Para la capa de sesión se utilizan las llamadas a procedimiento remotos o **Remote Procedure Call (RPC). RPC** es un modelo que forma la base para el intercambio de mensajes en todas las aplicaciones **NFS** y es también la base del diseño de aplicaciones distribuidas.
RPC puede ser utilizado en el diseño desarrollo de servicios de red, los cuales como su nombre lo indica, son usados de una manera similar a las llamadas a subrutinas o procedimientos que se usan en los lenguajes de programación de alto nivel, este modo de trabajo facilita las tareas del programador en el diseño e implementación de programas distribuidos.

El modo de operación de un **RPC** consiste en los siguientes pasos:

1. Activación por el programa cliente. Los parámetros de petición son empaquetados y enviados en paquete de datos a través de la red hacia el servidor.
2. Los parámetros son desempaquetados en el servidor.
3. Ejecución de la petición (el procedimiento) en el servidor.
4. El resultado es empaquetado y enviado por la red al cliente.
5. Desempaquetado de los resultados por el cliente y continúa con la ejecución normal del programa.

El protocolo **RPC** es el chasis del transporte de peticiones al servidor, en donde estas peticiones desembocan en procedimientos. En una máquina puede haber varios pedazos de programa o procedimientos, un grupo coherente y funcional de procedimientos es llamado ***servicio***. Cada servicio es asignado a un ***número de programa*** .

Usualmente solo un servicio es proveído por un programa servidor

Acabamos de mencionar que el protocolo **RPC** es el chasis del transporte de peticiones al servidor, cada paquete de peticiones debe tener definidos tres apartados importantes:

- número de programa
- número de procedimiento
- número de versión

El número de programa indica a que servicio va dirigida la petición.
Los procedimientos individuales de un programa son usualmente numerados secuencialmente, el número de procedimiento hace referencia a esta numeración y ayuda a establecer a qué nivel de servicio van las peticiones.
El número de versión valida que la petición hecha por el cliente sea soportada por el servidor.

Los mensajes **RPC** muchas veces contienen también información que le permite al servidor validar la petición del cliente, esto es si el cliente envía una petición al servidor y esta petición es valida, el servidor debe checar que el cliente que hace dicha petición esté autorizado para hacerla.

Es importante mencionar la dependencia de **RPC** con los *protocolos de transporte* que usa para lograr sus fines. Un **RPC** puede ser ejecutado sobre **TCP** o **UDP** . Algunos servidores ofrecen sus servicios sobre ambos protocolos y permiten que la elección del protocolo la haga el cliente. Como un estándar **RPC** utiliza a **UDP**.
Con **UDP** los paquetes tienen un máximo de 8 kbytes, cuando se usa **TCP** los mensajes son empaquetados en bloques de 4 bytes, en teoría varios millones de bytes pueden ser movidos en un **RPC**. Además cuando se usa **UDP**, la verificación de la llegada del paquete al servidor no la hace **UDP**, si no que debe ser implementada por **RPC**, habilitando un contador en el cliente, si el contador expira entonces se retransmite el paquete. Cuando se usa **TCP** la implementación anteriormente dicha no es necesaria ya que **TCP** tiene su propio sistema de rectificación de la llegada de paquetes, tanto al servidor como al cliente.

Representación de datos externa, XDR

NFS utiliza **XDR**, en la capa de presentación, en seguida mencionaremos cual es la utilidad de este protocolo.

Cada arquitectura de computadora tiene su propia definición para la representación de datos, esta representación puede ser determinada por el tipo de hardware y software que utiliza la computadora. Algunas computadoras almacenan el byte menos significativo de un entero en la más baja localidad de memoria, otras almacenan el byte más significativo en la localidad de memoria antes mencionada.
Cuando se utiliza una sola computadora para ejecutar diversos programas en ella, no hay problema ya que la representación de los datos es la misma, para cualquier programa. El problema empieza cuando tenemos dos o más computadoras con diferente arquitectura comunicándose a través de clientes y servidores, ya que al tener diferentes arquitectura tienen también diferentes maneras de representar datos. **TCP/IP** resolvió el problema creando una función que permitiera pasar la representación de datos de la computadora que desea transmitir los datos a una representación estándar de red y viceversa, esta representación de datos estándar es independiente de la arquitectura que se estén usando.
Antes de enviar datos a través de la red, la representación de datos de la computadora transmisora es convertida a la representación de datos estándar de red, la máquina receptora convierte la representación estándar de red a su propia representación de datos después de recibir el paquete.
La representación estándar de red usada para que los datos viajen a través de la misma es conocida como ***Representación de datos externa*** o **eXternal Data Representation (XDR).**

El Protocolo NFS

Anteriormente mencionamos que un grupo coherente y funcional de **RPCs** puede conformar un servicio; de la misma manera un grupo de **RPC** puede representar un protocolo.

El protocolo NFS es en si mismo un grupo coherente y funcional de RPCs .

Este protocolo define a un servidor **NFS** como un simple programa remoto, esto es un programa que espera peticiones de un cliente y las responde.
En cualquier protocolo definido por llamadas a procedimientos remotos, el cliente debe de empezar todas las operaciones, mientras que el servidor solo deberá responder peticiones individuales de los clientes.

El servidor **NFS** a través de varios procedimientos, permite a un cliente *crear, borrar, leer, escribir* o *buscar* archivos y directorios , así como obtener información tal como permisos, tamaño, fecha de creación de los mismos, en un sistema de archivos local, el cual es puesto a la disposición de los clientes. El protocolo **NFS** esta constituido por un programa remoto que implementa a **18** procedimientos:

NFSPROC_NULL (Procedimiento 0)
Este procedimiento no esta asociado con ninguna acción. Si un cliente o aplicación hace una llamada a este, usualmente es para checar que el servidor este activo y respondiendo a las peticiones.
NFSPROC_GETATTR (Procedimiento 1)
Un cliente hace un llamado al procedimiento 1 para obtener los atributos de un archivo. Estos atributos pueden ser permisos, propietario, tamaño y tiempo de ultimo acceso.
NFSPROC_SETATTR (Procedimiento 2)
Permite a un cliente dar atributos a un archivo.
NFSPROC_ROOT (Procedimiento 3)
Este procedimiento fue desarrollado para las versiones anteriores de **NFS**, pero ahora es obsoleto y fue reemplazado por el protocolo mount.
NFSPROC_LOOKUP (Procedimiento 4)
Los clientes llaman al procedimiento 4 para hacer una búsqueda de algún archivo en un directorio.
NFSPROC_READLINK (Procedimiento 5)
Permite que el cliente pueda leer el valor de una liga simbólica.
NFSPROC_READ (Procedimiento 6)
Este procedimiento provee uno de las más importantes funciones, ya que este permite que un cliente pueda leer los datos de un archivo.
NFSPROC_WRITECACHE (Procedimiento 7)
Este procedimiento no es usado en la versión corriente del protocolo; se esta desarrollado para la siguiente versión.
NFSPROC_WRITE (Procedimiento 8)
Este procedimiento provee otra de las funciones básicas, permite que un cliente pueda escribir datos en un archivo remoto.
NFSPROC_CREATE (Procedimiento 9)
Un cliente llama a este procedimiento para crear un archivo en un directorio.
NFSPROC_REMOVE (Procedimiento 10)
Un cliente invoca al procedimiento 10, para borrar un archivo existente.
NFSPROC_RENAME (Procedimiento 11)
Este procedimiento permite a un cliente renombrar un archivo. La operación de renombramiento corresponde al comando *mv* de **UNIX.**
NFSPROC_LINK (Procedimiento 12)
Los clientes pueden hacer una llamada al procedimiento 12 para hacer una liga dura a un archivo existente.
NFSPROC_SYMLINK (Procedimiento 13)
Este procedimiento crea una liga simbólica a un archivo.
NFSPROC_MKDIR (Procedimiento 14)
Un cliente llama al procedimiento 14 para crear un directorio. Si el procedimiento se lleva a cabo se manda al cliente el nombre del nuevo directorio y la lista de los atributos asociados a el.
NFSPROC_RMDIR (Procedimiento 15)
Este procedimiento borra un directorio remoto. Al igual que en **UNIX** el directorio debe de estar vacío para que pueda ser borrado.
NFSPROC_READIR (Procedimiento 16)
Este procedimiento permite leer los atributos de un directorio.
NFSPROC_STATFS (Procedimiento 17)
El procedimiento 17 permite al cliente obtener información acerca de el **Fyle System** en el cual un archivo determinado reside, esta información puede ser: los permisos de acceso, el tamaño de bloque[5], el número de bloques en el dispositivo, el número de bloques que actualmente están usados, y el número de bloque disponibles.

[5] Un bloque es un conjunto de bytes, que se utiliza como unidad mínima de almacenamiento direccionable en los sistemas UNIX, generalmente un bloque consiste de 512 bytes.

El protocolo mount

Mount, es de vital importancia en el servicio ofrecido por **NFS**, sin embargo el protocolo **Mount** no es parte del programa **NFS**.

El protocolo **Mount** también está constituido por un grupo de **RPCs** y a su vez define a un programa remoto llamado servidor **Mount**., el cual provee cuatro servicios básicos que los clientes necesitan antes de poder acceder al servidor **NFS**. El primer servicio permite a los clientes obtener una lista de los sistemas de archivos que estén disponibles en un servidor; el segundo permite que el cliente, pueda reconocer todos los posibles caminos a directorios o archivos dentro de un sistema de archivos exportado; el tercero checa la validez de cada petición que se realiza por un cliente, además de validar los permisos que tiene el cliente para poder acceder a un sistema de archivos. El servidor asigna una marca a cada directorio o archivo en un sistema de archivos exportado, dicha marca sirve para que el cliente pueda identificar la posición del archivo o directorio en el mismo, el directorio raíz es decir donde empieza el árbol de dicho sistema, también tiene una marca que lo diferencia de los demás, esta marca deberá ser conocida por el cliente ya esa será la principal referencia para el acceso de todos loa archivos o directorios que se deriven del directorio raíz, el cuarto servicio del protocolo **Mount** proporciona al cliente la marca del directorio raíz.

El protocolo **Mount**, está constituido por 6 procedimientos:

MNTPROC_NULL (Procedimiento 0)
Este procedimiento no esta relacionado con ninguna acción. Los clientes que hacen acceso a este procedimiento, solo comprueban que el servidor este recibiendo adecuadamente las peticiones.
MNTPROC_MNT (Procedimiento 1)
El cliente llama al procedimiento 1 para obtener la marca identificadora del directorio raíz, de un sistema de archivos especifico.
MNTPROC_DUMP (Procedimiento 2)
Permite a un cliente obtener un lista de los sistemas de archivos exportados que otro cliente esté utilizando.
MNTPROC_UMNT(Procedimiento 3)
Este procedimiento permite informar al cliente que el servidor o alguno de sus sistemas de archivos exportados saldrá de servicio.
MNTPROC_UMNTALL(Procedure 4)
Este procedimiento permite que el servidor pida al cliente que desmonte todos los sistemas de archivos que tiene de este servidor, ya que saldrá de servicio.
MNTPROC_EXPORT(Procedimiento 5)
Permite al cliente obtener los nombres de todos los sistemas de archivos accesibles en el servidor.

1.4.5 Slip/PPP

Existen varios tipos de conexión a **Internet**, la conexión directa es cuando se tiene una computadora con una interfaz de red que este conectada a un medio de transmisión con acceso directo a **Internet** y este acceso sea posible por cualquier dispositivo de interconexión. La conexión directa representa el ultimo modo de acceso a **Internet** para un usuario particular ya que es muy costoso y solo empresas o instituciones educativas pueden tenerlo. La alternativa es la conexión indirecta por vía telefónica haciendo uso de los protocolos **SLIP** y **PPP** .

SLIP y **PPP** tienen la capacidad de transportar trafico de paquetes **TCP/IP** sobre líneas seriales tal es el caso de las líneas telefónicas, esto permite a los usuarios particulares obtener acceso a **Internet** desde su propia PC en casa u oficina, con solo usar un **modem**[6] y una línea telefónica.

SLIP

El protocolo **SLIP (Serial Line IP)** es un protocolo muy simple que define una secuencia de caracteres que enmarcan paquetes IP para ser transmitidos por una línea serial .

SLIP define dos caracteres especiales para enmarcar a los paquetes IP : **END** y **ESC** . Todos los paquetes **SLIP** empiezan con la secuencia de caracteres que representa a **ESC** , seguido por los datos, cuando el ultimo byte de datos es transmitido, entonces la secuencia de caracteres **END** es enviada marcando el fin del paquete **SLIP**. Los paquetes **SLIP** no contienen ningún tipo de dirección u otro campo.

La única función de **SLIP** es solamente contener a los paquetes **IP** en un formato especifico para que puedan ser transmitidos en una línea serial, no tiene funciones de direccionamiento o de fragmentación de paquetes u ordenamiento de paquetes.

PPP

El **PPP** (Protocolo Punto a Punto) fue diseñado para proveer una liga simple la cual transporta paquetes procedentes de diferentes protocolos entre dos puntos. Esta liga proporciona operación bidireccional y entrega de paquetes en orden.

Paquete PPP

El paquete **PPP** esta diseñado para que el protocolo punto a punto pueda multiplexar paquetes de diferentes protocolos (**IP, TCP, UDP** etc.) simultáneamente sobre la misma liga.

---

[6] *Modulador-demodulador* . Dispositivo que convierte señales digitales a una forma adecuada para transmisión sobre medios de comunicación analógicos y viceversa.

El paquete **PPP** requiere ser enmarcado para indicar el inicio y el fin del paquete. Una gráfica de un paquete **PPP** se muestra a continuación, los campos son transmitidos de izquierda a derecha.

figura 1.22 Paquete PPP

El campo *protocolo* contiene un identificador que permite a **PPP** determinar el protocolo que envío los paquetes. El campo *información* contiene el paquete del protocolo especificado en el campo anterior, la máxima longitud de este campo es 1500 bytes.

Control de conexión

Para ser lo suficientemente versátil y portable en una amplia variedad de ambientes, **PPP** provee un control de conexión, que se encarga de manejar errores que se presenten durante ella, además de controlar el fin de conexión. Además determina cuando una de las máquinas en la conexión no esta trabajando apropiadamente.

Control sobre protocolos

Ya que **PPP** esta habilitado para manejar diferentes esquemas o familias de protocolos, se enfrenta a diversos problemas como ejemplo el direccionamiento **IP** , estos problemas son manejados por una familia de protocolos de control de red **(NCPs Network Control Protocols)** los cuales se encargan de las necesidades especificas requeridas por cada protocolo.

1.5 Servicios que ofrece REDII

En el Instituto de Ingeniería se realizan un gran número de investigaciones. El desarrollo de este trabajo requiere herramientas de vanguardia, y una de estas herramientas es el modelo de cómputo en el cual investigadores y demás usuarios se apoyan para realizar sus labores.
El modelo de cómputo en el Instituto de Ingeniería, debe satisfacer diversas necesidades, tales como: la posibilidad de usar recursos de cómputo disponibles y de fácil acceso, por ejemplo, impresoras, unidades de respaldo y almacenamiento, periféricos etc., tener acceso a fuentes de información mundial actualizada, mecanismos que permitan tener comunicación de una manera sencilla con otros investigadores en todo el mundo, etc.
Debido a las necesidades de cómputo del Instituto y a que se cuenta con recursos en cómputo diversos, se eligió hace algún tiempo, un modelo de cómputo distribuido para cubrir las necesidades de la institución. Así es que para entender de una mejor manera los servicios que REDII ofrece a sus usuarios, describiremos lo que es un modelo de cómputo distribuido ya que en este se basan la mayoría de dichos servicios.

1.5.1 Cómputo distribuido

Un sistema o modelo de cómputo distribuido puede definirse como un colección de computadoras y periféricos conectados por algún sistema de comunicación e integradas lógicamente por un sistema operativo para ambientes distribuidos. Dichas computadoras pueden ser desde microcomputadoras hasta supercomputadoras.

La filosofía de un sistema de este tipo, es distribuir procesamiento, información y recursos donde sean necesarios, permitiendo a personas y procesos tener acceso a ellos. Esta distribución deberá ser totalmente transparente, lo que lo convierte en un sistema sumamente adaptable a los recursos existentes en una empresa, centro de investigación o educación, además de contar con la facilidad de aumentar recursos en la medida que sea necesario.

Otra de las facilidades importantes que nos da un sistema de cómputo distribuido es que diferentes sistemas (entiéndase por sistemas: sistemas de **hardware** o plataformas de computadoras como estaciones de trabajo, sistemas de **software** como programas de aplicación y protocolos) puedan interactuar para proveer a los usuarios mayores beneficios. Lo que permite esta interacción de los sistemas que participan en un sistema de cómputo distribuido es que éstos son *sistemas abiertos*. Un sistema abierto provee portabilidad de **software** a través de plataformas estándar, interoperabilidad entre aplicaciones, conectividad entre sistemas y flexibilidad en el manejo de información. Los sistemas abiertos son desarrollados por la unión de varios fabricantes, que buscan que su producto llegue a ser un estándar.

El papel de una red de computadoras en un sistema de cómputo distribuido es vital ya que es el sistema de comunicación del mismo, su infraestructura. Un sistema distribuido utiliza todos los recursos de una red para alcanzar sus objetivos. Para que la interacción de procesos, procesadores y recursos se lleve a cabo de una manera satisfactoria en una red, es necesario estandarizar las formas de comunicación existentes en ella, y es por ello que se crean los protocolos. **REDII** utiliza varios protocolos de comunicación para poder llevar acabo sus servicios, en el capitulo anterior hablamos profundamente de los mismos, en este capitulo solo los relacionaremos con los servicios que se prestan en nuestra red.

Sistemas operativos para modelos de cómputo distribuido

La piedra angular de un modelo o sistema de cómputo distribuido son los sistemas operativos que utiliza, ya que gracias a ellos es posible distribuir procesos, tareas de administración y recursos como son periféricos, medios de almacenamiento de datos etc., además que sientan la base para poner en funcionamiento todos los servicios.
Los sistemas operativos para modelos de cómputo distribuido se pueden agrupar en :

• *Sistemas operativos de red:* Un sistema operativo de red tendrá que realizar tareas de administración de memoria, periféricos, dispositivos de entrada/salida, archivos, etc. Deberá contar con un control de entradas al sistema por los usuarios y ejercer tareas de seguridad en cada máquina.
Estos sistemas operativos necesitan incorporar módulos que provean comunicación en cada computadora con el resto de los dispositivos en la red.
Cada computadora, tiene de manera independiente instalado el sistema operativo, es así como los usuarios trabajando en este sistema, necesitan conocer la localización de los recursos remotos para poder utilizarlos, además de conocer la localización exacta de donde están almacenados sus archivos, en otras palabras el usuario esta enterado que esta trabajando en múltiples e independientes computadoras, compartiendo recursos de una manera transparente.

• *Sistemas operativos distribuidos:* Un sistema operativo distribuido realiza las tareas antes mencionadas por el sistema operativo de red, pero sin la necesidad de adicionar los módulos de comunicación, ya que estos son parte integral del sistema operativo.
Un sistema operativo distribuido es compartido por todas la computadoras conectadas a la red, convirtiéndolo así en un sistema operativo común. La primera tarea de un sistema operativo de este tipo es controlar la localización de los recursos (computadoras, periféricos, dispositivos de almacenamiento, procesadores etc.) y los servicios de red, y entonces integrarlos en un solo sistema. El acceso para los usuarios es totalmente transparente, ya que ellos no necesitan conocer donde están localizados los recursos para poder accederlos.

Ya sea sistemas operativos distribuidos o de red, los dos proporcionan grandes facilidades para la implantación de sistemas de cómputo distribuidos. En general un sistema operativo para ambientes distribuidos lleva a cabo las siguientes tareas:

- Capacidad para la administración de recursos
- Control de acceso de usuarios y dispositivos primarios de seguridad
- Control de procesos y asignación de recursos para ejecutarlos
- Comunicación entre procesos
- Llevar un registro de los recursos utilizados por procesos
- Integración y utilización de la red
- Proporcionar transparencia en los servicios y en la ejecución de los procesos
- Amplia variedad en los servicios que pueden ser implementados sobre el.

Modelo cliente-servidor

El término *servidor* es aplicado a un programa en una máquina determinada, que ofrezca un servicio el cual puede ser requerido sobre una red, mientras que un *cliente* es un programa que efectúa peticiones a un servidor y espera una respuesta, el modelo cliente-servidor sigue un formato de petición respuesta que involucra a uno o mas clientes y a un servidor.

En una relación cliente-servidor el procesamiento se divide en las dos partes, el cliente ejecuta una aplicación que muestra una interfaz al usuario, da formato a las peticiones al servidor, y muestra la información o los mensajes enviados por el servidor. El servidor realiza el procesamiento posterior de la información enviada por el cliente. Los servidores son a menudo potentes sistemas, capaces de gestionar adecuadamente las múltiples y simultáneas peticiones que reciben de los clientes.

1.5.2 Servicios

Los servicios de red son un conjunto de aplicaciones que generalmente trabajan bajo un modelo cliente-servidor, las cuales proporcionan un gran número de beneficios y facilidades para los usuarios.
En este trabajo, los servicios son catalogados en: servicios básicos, lo cuales vienen comúnmente integrados como parte del sistema operativo y sirven como base para servicios más complejos; por otro lado tenemos los servicios de información y recursos compartidos los cuales son aplicaciones más elaboradas y que deben ser integradas a los modelos de cómputo ya existentes.

1.5.2.1 Servicios básicos

Sesiones remotas

Las computadoras en una red proveen una característica muy especial, que permite el acceso a una máquina desde otra que este o no en su misma red, es decir si un usuario trabaja en una máquina y tiene la necesidad de procesar o consultar información en otra máquina cualquiera, puede tener acceso a ella mediante un procedimiento que permite una conexión con dicha máquina llamada también *máquina remota*. Esta conexión es posible gracias a una *emulación* de terminal y proporciona una ambiente de operación casi idéntico de la máquina remota como si se estuviera en el mismo lugar físico de esta máquina.

Desde hace varios años, varios fabricantes de computadoras han inventado facilidades que permiten a sus usuarios hacer sesiones remotas, pero con **Internet** la herramienta de sesiones remotas más comúnmente usada es **telnet** , su nombre viene del protocolo basado en **TCP/IP** para soportar sesiones remotas en **Internet**, **telnet** asume que **TCP/IP** está encargándose de las actividades en las capas inferiores del modelo de protocolo y solo trabaja a nivel de protocolo de aplicación.
Ya que **TCP/IP** es asociado fuertemente con los sistemas Unix, **telnet** usualmente es incluido como un comando de Unix, sin embargo **telnet** es un comando de **TCP/IP**. Un gran ventaja es que no es necesario tener una computadora con Unix para soportar **telnet**, gracias a que **TCP/IP** es aplicable a una amplia gama de sistemas computacionales desde supercomputadoras y **mainframes** hasta computadoras personales corriendo desde VAX/VMS hasta DOS, Microsoft Windows, MacOs y otros.

Telnet como muchas otras facilidades de **Internet** es un servicio cliente-servidor. **Telnet** es una gran herramienta que combinada con la conectividad proveída por **Internet**, da a los usuarios de una red un universo muy amplio de información, por ejemplo:

- Sesiones remotas vía **Internet** para establecer conexión a las máquinas dentro de nuestra propia red desde cualquier lugar del mundo.
- Sistemas de bibliotecas en línea, sus catálogos y bases de datos
- Acceso a supercomputadoras (asumiendo que se tiene acceso permitido a ellas)
- Información geográfica, sismológica, bases de datos en línea, o bien captura en tiempo real desde instrumentos que recopilan dicha información.

Comunicación electrónica

Otro servicio básico que ofrece la red, es la comunicación en línea, o bien a través del envió de mensajes. Con este servicio el usuario puede entablar en forma confiable y eficiente comunicación con otros usuarios dentro del Instituto o bien con personal en otras instituciones nacionales o extranjeras, que obviamente estén integradas a una red mundial con los mismos servicios, tal es el caso de **Internet** .

La herramienta de comunicación electrónica mas utilizada a nivel mundial, es el *correo electrónico*. El correo electrónico simplemente, es un camino para que usuarios de una red puedan intercambiar mensajes a nivel mundial. Los mensajes que se envían por el correo electrónico generalmente son solo texto, pero también pueden ser enviados archivos conteniendo imágenes, como fotografías y gráficas, o bien archivos en procesadores de texto que manejen formatos específicos, también programas ejecutables y una gran variedad de datos.

El correo electrónico permite el envío de un mensaje de un usuario a varios usuarios, lo que ha dado pie a la creación de varios servicios basados en el correo electrónico, como ejemplo podemos citar las listas de interés, que son grupos de personas con un interés

común y que envían sus comentarios o preguntas a una sola dirección, y llegan automáticamente a todos los miembros de la lista.

El correo electrónico es proporcionado por dos tipos de programas: El programa que permite que los usuarios puedan crear, leer y manejar sus mensajes y el programa que actúa como una "oficina de correo" y se encarga de entregar a su destino los mensajes que el usuario genera y de recibir los mensajes que llegan al usuario y hacerlos accesible a el.
Los programas más populares utilizados para manipular correo electrónico son: **mail, sendmail, pine, elm** para sistemas Unix, y **Eudora, Netscape,** para PC.

En el Instituto de Ingeniería el servicio de correo electrónico que provee **REDII** es una parte muy importante en el desarrollo de las actividades de investigadores y en general de todo el personal académico, ya que provee un camino de comunicación rápido y confiable, además de ser una fuente de información actualizada mediante los grupos de interés. Actualmente estos grupos de interés han ido en aumento en las áreas de desarrollo de la Institución.

Otra herramienta muy importante de la comunicación electrónica es la *comunicación en línea* también conocida como **talk** . Este servicio funciona igual que una llamada telefónica, estableciendo una conversación en tiempo real, entre dos personas ya sea localmente o a nivel mundial. Dentro del campo de la comunicación en línea, esta también el **Chat,** servicio que permite la comunicación en línea entre dos o más personas.

Transferencia de archivos

La transferencia de archivos, es el movimiento de archivos desde una computadora a otra sin necesidad de usar unidades de almacenamiento secundarios tales como los disquetes, fue una de las razones originales para la creación de las redes.
Para muchas organizaciones, la facilidad de transferir archivos se ha convertido en una necesidad, el Instituto de Ingeniería no es la excepción, la herramienta con la que se cuenta en **REDII** para dar este servicio es **ftp.**
Ftp (File transfer protocol) es el nombre de la herramienta y el nombre del protocolo de aplicación que juntos realizan la transferencia de archivos sobre **TCP/IP** . Al igual que **telnet** , usualmente **ftp** es incluido como un comando de Unix, pero en realidad es una aplicación **TCP/IP,** por lo mismo no solo es utilizado por sistemas Unix, también existen versiones para DOS, Microsoft Windows, MacOs, etc.

Los archivos que pueden ser transferidos son de cualquier tipo: binario, texto, imágenes etc. y se puede tener acceso a ellos desde cualquier sistema operativo. Debemos mencionar que para que esta versatilidad tome efecto, **ftp** cuenta con modos especiales de transferencia los cuales nos permitirá enviar un archivo sin que sufra la menor modificación, además de tomar otras medidas como es la codificación de los archivos a transferir.

1.5.2.2 Servicios de Información.

En **Internet** actualmente hay una gran variedad de servicios que ponen a disposición de los usuarios una gran variedad de información. Dichos servicios están basados en el esquema cliente servidor.
Si se desea solamente consultar los servicios de información ya existentes en **Internet** se debe configurar los programas clientes para poder realizar dicho acceso, por otro lado si se desea implementar un servicio de información propio, el servidor tendrá que ser configurado en la propia red. En **REDII** existen varios servicios de información que han sido implementados tanto en cliente como en servidor, estos servicios se han convertido en herramientas indispensables para la transferencia, búsqueda y divulgación de la información que se genera y se necesita en el Instituto.

FTP anónimo

La transferencia de archivos es uno de los servicios más usados dentro de nuestra red y en general en **Internet**, por ello se ha creado un tipo de servicio de información a usuarios en general dentro de esta red llamado *servidor de FTP anónimo*. Un **FTP** anónimo es una máquina (servidor) que pone a disposición de los usuarios de **Internet** un árbol de archivos que pueden contener información de diferentes tipos (archivos texto con información de cualquier tópico, programas ejecutables, imágenes, gráficas, etc.), toda esta información puede ser accedida por cualquier usuario en cualquier parte del mundo con un simple **FTP** a una cuenta publica en el servidor. Los servidores de **FTP** anónimo son los servidores de información más comunes, toda máquina dentro de **Internet** puede ser cliente o servidor de **FTP** o ambos.

World Wide Web

El **World Wide Web** también conocido como **WWW** es el más nuevo y poderoso servicio de información en **Internet** . **WWW** es un navegador de **Internet,** es gráfico y utiliza lenguaje *hypertexto,*, dicho lenguaje permite crear una página de texto muy amigable, la cual contiene ligas virtuales que pueden conducir al usuario a otras partes en el mismo texto y más aun puede ligarlo directamente a otros servidores, sin que el usuario se entere de estos cambios. **WWW** también tiene la capacidad de integrar en su ambiente a la mayoría de servicios de información y servicios básicos, como es el caso de todos los servicios mencionados anteriormente, así es que **WWW** provee un camino simple para usar y entender **Internet** .

Los documentos **WWW** son escritos en un lenguaje llamado **HTML (Hypertext Markup Leanguaje),** cuando un cliente de **WWW** realiza una petición para leer un documento a un servidor de **WWW** , este debe transferir el documento al cliente y el cliente debe desplegarlo en un formato legible y estructurado al usuario, el servidor de **Web** transfiere dichos documentos a través de un protocolo llamado **HTTP** ***HyperText Transfer Protocol.***
Como mencionábamos anteriormente los documentos de **Web** contienen ligas virtuales, las cuales podemos diferenciar del texto común, por ser palabras o frases subrayadas y en

diferente color. Cuando un usuario sigue una de estas ligas (comúnmente haciendo un click sobre la frase), el cliente de **Web** realizará un salto a otra parte del mismo documento o bien a otro documento que puede estar o no en el mismo servidor, además se puede ligar de la misma manera un servidor de **ftp, gopher, archie, telnet, e-mail** etc.

Servicio de resolución de nombres

Como hemos visto en capítulos anteriores el método para encontrar nodos dentro de **Internet** se basa en un sistema de direccionamiento que involucra a las direcciones **IP**, por facilidad para los usuarios a estos nodos se les da un nombre único asociado a la dirección **IP**, el cual no es un número sino un conjunto de palabras comunes que identifican a este nodo como único en toda **Internet** .
Como ejemplo, tenemos que para acceder al servidor principal de la red del Instituto de Ingeniería tenemos dos caminos, la dirección **IP** y el nombre del servidor :

Dirección **IP** del servidor principal del Instituto de Ingeniería	Nombre del servidor principal del Instituto de Ingeniería
132.248.53.245	pumas.iingen.unam.mx

En realidad la búsqueda del nodo se realiza por dirección **IP** no por el nombre de nodo, así es que se tiene la necesidad de contar con un sistema que sea capaz de buscar la dirección **IP** asociada a un nombre de nodo. El sistema que realiza la tarea antes mencionada es el **DNS (Domain Name System).**
El **DNS** tiene dos aspectos conceptualmente independientes. El primero es abstracto, especifica la sintaxis de los nombres asociados a las direcciones **IP**, y establece una jeraquia entre dichos nombres. El segundo es concreto, implanta un sistema cliente-servidor, que transforma eficazmente los nombres en direcciones **IP**.

1.5.2.3 Recursos compartidos

En un esquema de cómputo distribuido, es necesario implementar herramientas que nos permitan compartir recursos, como dispositivos de almacenamiento, impresoras, procesadores, etc. Al implementar estas herramientas en REDII se tienen grandes beneficios, ya que se optimizan los recursos de cómputo existentes, además de reducir el gasto en equipo de cómputo, software y dispositivos periféricos.
A continuación se mencionaran las herramientas que se han implementado en REDII para poder compartir recursos.

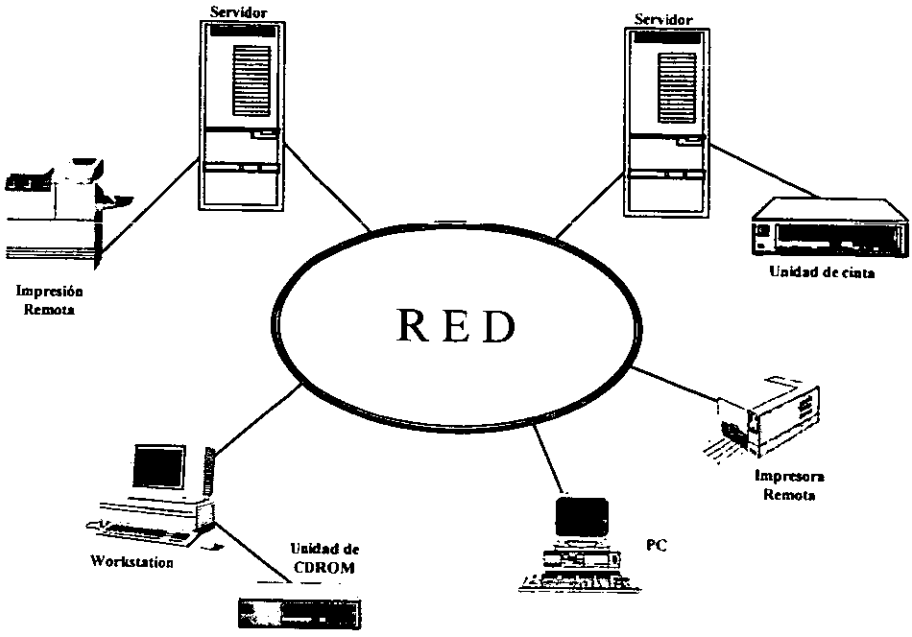


Figura 1.22 Esquema de recursos compartidos en una red

Sistemas de archivos en red y recursos compartidos

En el Instituto de Ingeniería se cuenta con Windows NT para proveer este servicio. **Windows NT** básicamente realiza el manejo y la administración de un sistemas de archivos ofreciendo el ambiente operativo Windows de Microsoft.

La herramienta de mayor uso entre estaciones de trabajo **UNIX** para compartir dispositivos de almacenamiento en el Instituto, es **NFS. NFS (Network File System)** es un programa que provee transparencia en el acceso a unidades de almacenamiento remotas. **NFS** permite compartir información y proveer espacio de almacenamiento para todos los usuarios de la red ; de esta manera los usuarios no necesitan hacer un acceso remoto vía telnet o rlogin hacia los sistemas que almacenan información o tienen espacio disponible.
NFS es un programa que se basa en el esquema cliente-servidor. Un servidor **NFS** es el que contiene espacio en disco disponible y permite compartirlo con todos los miembros de la red, mientras que un cliente **NFS** toman el espacio que el servidor ofrece y lo integra a su sistema otorgando a los usuarios un acceso transparente a dichos recursos.

Originalmente **NFS** fue desarrollado para sistemas **UNIX**, de hecho todos las computadoras que trabajan con este sistema operativo incluyen esta herramienta y es posible compartir disco desde cualquier plataforma.

Impresión Remota

Uno de los recursos compartidos más solicitados por los usuarios son las impresoras. La impresión remota en el Instituto de Ingeniería se realiza para las estaciones de trabajo **UNIX**, mediante servidores de impresión manejados por un solo servidor, mientras que para las computadoras personales los servidores de impresión son manejados por **Windows NT**.

Acceso vía modem

Investigadores en este Instituto requieren cada día más los servicios de **REDII**, ya que algunos de ellos tienen la necesidad de trabajar en el interior de la república o desde sus casas y necesitan acceso a todos los servicios de información de la red. Este servicio se proporciona con los protocolos **SLIP** y **PPP**.

Comunicación interactiva

La comunicación interactiva nos proporciona un camino de comunicación nuevo y poderoso para nuestros usuarios, un ejemplo claro de este tipo de comunicación son las vídeo conferencias.

Las vídeo conferencias son transmisiones de imagen, voz y datos por un medio, este medio puede ser líneas telefónicas, o bien una red como **Internet.**
Para llevar acabo una vídeo conferencia se deben controlar y manejar diferentes circunstancias tales como :

- Audio/Vídeo ; captura, compresión y despliegue
- Comunicaciones de red ; Protocolos
- Hardware de captura de imágenes y sonidos ; como cámaras, micrófonos, etc.

En el Instituto de Ingeniería se esta investigando sobre los avances tecnológicos en esta área, para poder desarrollar un buen servicio de vídeo conferencias sobre nuestra red. El software que en este momento se utiliza para la realización de este servicio es **CuSeeMe** y **Net Meeting** para **Microsoft Windows**.

Capítulo 2

Problemática de REDII y estado actual del monitoreo y administración de la misma.

2.1 Introducción

Las redes de computadoras están siendo aceptadas y requeridas enormemente en la vida diaria de Instituciones publicas, privadas, educacionales, de investigación científica y tecnológica, comerciales etc. Estas redes crecen y junto con ellas los servicios que ellas proveen y se hacen más complejos, además que incrementan sus usuarios en grandes cantidades. Los usuarios por su parte esperan una excepcional eficiencia y rapidez en el acceso a los servicios y a la información que presta la red.

Día con día los usuarios de las redes se hacen más dependientes a ellas y los usuarios del Instituto de Ingeniería no son la excepción. REDII ha crecido en los últimos años, se han incrementado los servicios que ella provee y el número de usuarios que hacen acceso a la misma. Con este incremento de REDII en todos los aspectos, también se han incrementado los problemas que la red presenta y que se deben tener presentes para que REDII provea un servicio de alta disponibilidad y confiabilidad a todo el personal que es usuario de la misma y que la ve como una herramienta vital en el desarrollo de proyectos de importancia para este Instituto.

2.2 Análisis de los problemas, carencias y fallas de REDII en cuanto al monitoreo y administración de la misma.

2.2.1 Problemas y carencias en REDII

• *Falta de un mecanismo de alerta en caso de que una falla ocurra en el equipo de REDII.*
Cuando se presenta una falla en REDII, muchas veces el personal de soporte para la misma es el último en enterarse de dicha falla. Los usuarios desesperados llaman a la oficina de soporte para reportar la falla y es entonces cuando el personal de soporte va a tratar de resolverla.
El equipo de computo y de red siempre esta expuesto a fallas de muchos tipos, y que el personal de soporte no sea el primero en enterarse de cualquier falla que ocurra, incrementa el tiempo total en que la falla es solucionada, esto repercute directamente en la disponibilidad de la red.

• *Falta de un mecanismo de alerta de posibles sucesos que puedan convertirse en fallas en REDII*

Existen sucesos que pueden convertirse en graves fallas en la red si no son detectados oportunamente, tal es el caso de los servidores principales, si alguno de sus dispositivos de almacenamiento se llena este suceso puede repercutir en la caída del sistema. O bien si alguna de las fuentes de alimentación a los dispositivos de interconexión de la red falla y las unidades de reserva de poder están con niveles de carga bajos y apunto de apagarse, esto puede ser una falla de gran importancia ya que repercutiría en la disponibilidad de una o más subredes que conforman a la red. Estos sucesos podrían prevenirse o resolverse a tiempo, evitando así fallas importantes en la red.

• *Falta de un mecanismo que nos permita realizar tareas de administración en forma controlada.*

Las tareas de administración de la red se realizan actualmente en REDII de una manera disgregada lo que resta confiabilidad y disponibilidad a la red. Es decir para realizar una tarea de administración como levantar o dar de baja un dispositivo de la red, es necesario ir hasta donde esta dicho dispositivo y realizar las acciones pertinentes, estas acciones podrían llevarse acabo desde la misma oficina del personal de soporte, de esta manera se ahorra tiempo en los procesos de administración y soporte de la red.

• *Falta de información estadística.* No existen estadísticas de uso de la red, para planear crecimientos futuros.

Teniendo en cuenta lo anterior podemos decir que es necesaria la implementación de un sistema que nos permita cubrir las siguientes necesidades para la red del Instituto de Ingeniería :

• *Control de la expansión de la red.*
El continuo crecimiento del número de componentes de la red, usuarios, servicios etc. provoca que el personal de soporte de la red, pierda el control sobre que esta conectado a la red, que recursos y servicios son usados. Es por ello necesario tener un sistema que nos permita tener el control sobre todos los dispositivos que conforman nuestra red, además que sea posible anexar cualquier elemento nuevo a nuestro sistema.

• *Reducción de tiempo en las caídas de la red.*
Las redes para las instituciones u organizaciones son hoy día la herramienta principal de sus actividades, es por ello muy importante que una red tenga un porcentaje de disponibilidad muy alto. Las fallas que ocurran en la red y que provoquen una caída de la misma, deben ser localizadas y aisladas oportunamente, es por ello la necesidad de un sistema que haga posible esta tarea.

• *Control de costos.*
La utilización de los recursos debe ser monitoreada y controlada con el fin de satisfacer las necesidades de los usuarios con los recursos que ya se tienen, es decir optimizar los recursos existentes en la red y también optimizar con ello los recursos monetarios para la compra de nuevo equipo.

2.2.2 fallas

En el punto anterior se analizaron los problemas y necesidades de REDII, ahora nos toca analizar un punto muy importante : *Las fallas* .
Una *falla* dentro de REDII puede definirse como : *una deficiencia o interrupción en el servicio que ofrece la red.*

Dentro de REDII se han clasificado las fallas en dos clases:

• *Fallas debidas a la red principal y a otras redes:*

Físicas (menos comunes)
Lógicas (más comunes)

• *Fallas en la propia REDII:*

Físicas
Lógicas

Es necesario hacer mención de que las fallas debidas a la red principal (REDUNAM), se encuentran fuera del control del personal de soporte de REDII, estas fallas repercuten directamente en la funcionalidad de nuestra red, ya que REDUNAM nos provee el enlace con el resto del mundo.

Entre las *fallas fisicas* dentro de REDII encontramos :

- Problemas con el cableado
- Fallas debidas al medio ambiente
  - Rayos
  - Lluvia
  - Interferencia electromagnética
- Daños en los dispositivos de interconexión
  - Bridges
  - Concentradores
  - switches
  - Tarjetas de conexión a red de Pc´s y estaciones de trabajo
- Fallas en la conexión con REDUNAM
- Mal uso del equipo de cómputo por los usuarios
- Fallas por interrupciones abruptas en la corriente eléctrica

Entre las *fallas lógicas* dentro de REDII encontramos :

- Daños en los servidores de REDII y sus dispositivos de almacenamiento y periféricos
- Incorrecto manejo de los equipos de cómputo (Pc´s y estaciones de trabajo), por los usuarios
- Incorrecta configuración de los equipos de interconexión de red
- Saturación de paquetes de información que viaja por la red.
- Saturación en los medios de almacenamiento

Entre las *fallas fisicas* en REDUNAM encontramos :

- Problemas con el cableado
- Fallas debidas al medio ambiente
  - Rayos
  - Lluvia
  - Interferencia electromagnética
- Daños en los dispositivos de interconexión de red
  - ruteadores
  - **gateways** (principalmente en el **gateway** de el IIMAS)
- Problemas con el enlace a los Estados Unidos

Entre las *fallas lógicas* en REDUNAM encontramos :

- Problemas con los servidores de REDUNAM que proveen servicios de datos y comunicaciones, tal es el caso del servicio de resolución de nombres **DNS**[1]
- Incorrecta configuración de los dispositivos de interconexión de red
- Saturación de paquetes de información en la red
- Problemas de direccionamiento de red

2.3 Estado actual del monitoreo y administración de REDII

El estado actual del monitoreo y la administración de la red del Instituto es deficiente y esta enfocado principalmente a la corrección de fallas.

Existen pasos a seguir en la detección y corrección de una falla en REDII, estos pasos a seguir pueden ser catalogados como un *proceso de detección y corrección de fallas* y son englobados en el siguiente diagrama de flujo:

[1] El servicio de resolución de nombres, consiste en traducir el nombre de una máquina a su respectiva dirección **IP**, este servicio es necesario, ya que el acceso se realiza a través de direcciones **IP** no por nombre de la máquina

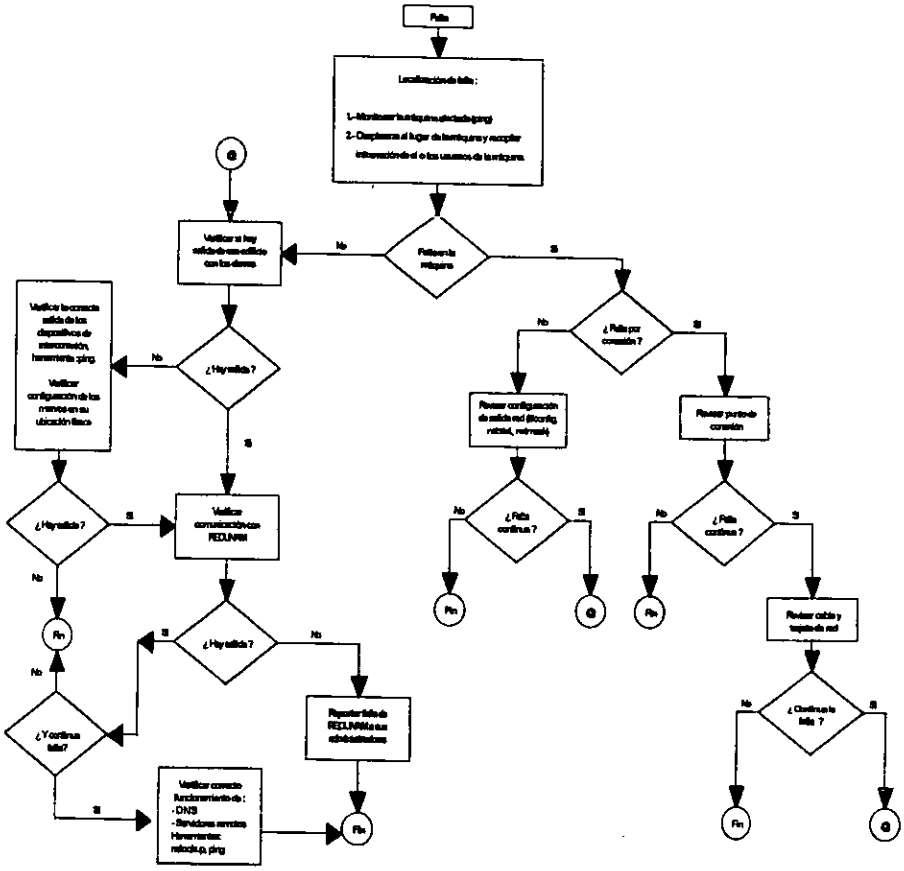


Figura 2.1 Proceso de detección y corrección de fallas en REDII

Para la detección y corrección de fallas se cuenta con *herramientas de monitoreo y administración* para realizar dicha labor, dichas herramientas trabajan sobre los distintos protocolos con los que cuenta REDII (**TCP/IP e IPX/Netware**), todas ellas se encuentran integradas perfectamente a los distintos sistemas operativos (**UNIX** y **Netware**), y paquetes (**PcNfs, Pctcp/ip** y **Novell**) que las proveen, asimismo no necesitan una implementación mayor para su configuración y uso, además de aparecer como simples comandos en los sistemas y paquetes.

Para los protocolos **TCP/IP :**

arp. El programa **arp** es un traductor de direcciones **IP** a direcciones **MAC**[2] Esta herramienta nos proporciona ayuda para localizar una dirección **IP** asociada a dos o más direcciones MAC, si esta doble asignación ocurre puede convertirse en una falla muy grave para la red.

ifconfig. Permite configurar la tarjeta que proporciona la conexión a red en una computadora. Esta herramienta ayuda a encontrar parámetros erróneos en la configuración de tarjetas, estos parámetros pueden causar conflictos y no permitir el acceso en forma correcta a la red.

netstat. Despliega información estadística de cada tarjeta de conexión a red de la máquina, tablas de ruteo y **sockets**[3]. Esta información nos permite tener un panorama amplio de lo que ocurre a la salida o entrada de información en nuestra computadora, además de poder realizar un análisis del trafico en la red, o bien analizar la actividad en niveles de protocolo, como por ejemplo a nivel **IP** o **TCP**.

iostat. Reporta actividad de entrada/salida de discos, terminales y utilización de CPU. Esta herramienta nos permite visualizar el estado de dispositivos importantes como son los discos y la utilización de CPU, para prevenir o detectar posibles fallas.

nslookup. Es un programa interactivo que permite contactar servidores **DNS** para realizar peticiones de resolución de nombres acerca de **hosts** o dominios específicos. Con esta herramienta es posible diagnosticar errores en la conexión debido a nombre de máquinas que estén equivocados o bien si algún servidor **DNS** esta fuera de servicio.

ping. Es un programa que usando mensajes **ICMP** puede detectar si una maquina se encuentra disponible o no dentro de la red. **ping** puede también enviar pequeños paquetes de información hacia una maquina determinada, con el fin de mostrarnos si los paquetes llegaron y en cuanto tiempo lo hicieron, con estas funciones **ping** es una herramienta de detección de errores muy utilizada.

---

[2]Una dirección MAC es la dirección asociada a cada tarjeta de red o bien a un dispositivo de interconexión. Esta dirección es manejada en niveles más bajos que IP y TCP.

[3] Socket : Función BSD de UNIX que permite a una aplicación acceder a un protocolo de comunicación

lpc. Controla la operación de la o las impresoras en un sistema. Este comando puede ser usado para monitorear el estado general de la impresora, inicializar o deshabilitar una impresora, revisar el estado de las colas de impresión, borrar colas, reacomodar trabajos de impresión.

Para los protocolos **Netware** :

consule. Este comando permite establecer un modo interactivo entre el usuario y el servidor de archivos, donde se permiten comandos de administración y monitoreo.

monitor. El comando **monitor** es usado para desplegar las actividades en todos los nodos de la red que estén realizando sesiones con el servidor de archivos.

chkvol. Es usado para desplegar información acerca de un volumen. Esta información incluye el nombre de el servidor de archivos donde esta localizado el volumen, el nombre del volumen, la capacidad total de almacenamiento del volumen, el numero de bytes usados, el numero de archivos en existencia, el numero de bytes libres, y el numero de directorios.

disk. Este comando permite monitorear el estatus de los discos de red. Este puede ser usado para ver cuales discos están funcionando normalmente y cuales no.

printer. Este comando es utilizado para controlar y monitorear trabajos de impresión en un servidor de archivos.

nprinter. Es usado para enviar, controlar y monitorear archivos a una impresora en red .

slist. Proporciona una lista de los servidores de archivos conectados a la red.

Como se ve no hay un sistema de monitoreo y administración en REDII, las anteriores son solo herramientas que nos ayudan a detectar fallas y a corregirlas. Además con las carencias y necesidades primeramente mencionadas que tiene la red de este instituto es necesaria la búsqueda e implementación de un *sistema de monitoreo y administración* eficaz que nos permita solucionar de una manera optima su problemática.

Capítulo 3

Principios teóricos para la implantación de un sistema de administración y monitoreo de red.

3.1 Introducción

Debido al crecimiento de las redes y a la importancia que ellas han ido adquiriendo, cada vez son más las personas que trabajan con ambientes distribuidos que involucran el uso de una red. Las caídas de dichas redes y sistemas afectan cada día más a las empresas y organizaciones.
Si se administra de una manera eficiente a las redes (esto incluye todos los elementos que la conforman: equipo de interconexión, servidores, **software**, etc.), se puede reducir grandemente la falta de disponibilidad del servicio y garantizar que los recursos de cómputo se estén utilizando adecuadamente.

3.2 Conceptos

3.2.1 Definición

Un sistema de administración de red es un conjunto de herramientas (**software** y **hardware**) que proporcionan monitoreo y control sobre una red. Su función principal es auxiliar al administrador de red a resolver cualquier tipo de problema que se le presente, para dar una solución rápida de una manera ordenada y secuencial.
Un administrador de red puede hacer manualmente las mismas tareas que un sistema de administración realiza, pero es preferible que el **software** de administración realice dichas tareas de una manera automatizada, de esta manera el trabajo del administrador es más eficiente y obtiene libertad y tiempo para dedicarse a tareas que signifiquen beneficios para la red.

Concretamente podemos definir a un sistema de administración de red de la siguiente manera:

*Integración de herramientas de **software** y **hardware** que controlan y monitorean redes de datos, para maximizar su eficiencia y productividad.*

Ya que un sistema de administración de red puede realizar muchas tareas a la vez, este requiere de suficiente poder de computo. La plataforma más utilizada para implementar un sistema de administración de red, son las estaciones de trabajo corriendo **UNIX** como sistema operativo con una interfaz gráfica de ventanas como el **X11**.

3.2.2 Características

Un sistema de administración de red completo debe tener las siguientes características:

- El sistema deberá proveer una interfaz gráfica que pueda producir una estructura jerárquica de la red y permitir conexiones lógicas entre los diferentes niveles de la jerarquía, esto puede lograrse con un mapa que plasme la topología actual de la red.
- La interfaz deberá contar con un conjunto de comandos amigables, pero poderosos para realizar las tareas de administración de red.
- El sistema deberá proveer una base de datos confiable que pueda almacenar y poner a disposición cualquier información requerida.
- El sistema deberá ser fácil de construir y expandir. Es decir el sistema deberá ser adaptable a cualquier tipo de red que se tenga, e igualmente si la red tiene o no la misma plataforma operativa, además de facilitar la adición de aplicaciones y desarrollos requeridos por el administrador.
- Un mínimo de equipo adicional. Esto es, mucho del **hardware** y **software** requerido por el sistema de administración deberá ser encontrado en el equipo existente.
- El sistema deberá ser capaz de manejar protocolos de administración actuales.
- El sistema deberá contar con herramientas de análisis de datos y graficación de los mismos.

3.2.3 Áreas que un sistema de administración debe cubrir

La Organización Internacional de Estandarizaciones (**OSI**) ha decretado las áreas que un sistema de administración debe cubrir como las siguientes:

3.2.3.1 Administración de fallas

El objetivo es determinar lo más rápido posible el punto de la red donde se presenta una falla para que ésta se corrija lo antes posible. El proceso de localización de un problema o falla en la red, envuelve los siguientes pasos:

1. Determinar exactamente donde esta la falla.
2. Separar el sitio donde este ocurriendo la falla del resto de la red, para que pueda seguir funcionando sin interferencia.
3. Determinar la causa o causas de la falla.
4. Reparar el problema (si es posible).

También se logra detectar problemas que puedan degenerar posteriormente en una falla. Los usuarios esperan una solución rápida y real de cualquier problema que se presente en la red e interfiera en su trabajo diario. Si una falla ocurre, los usuarios desean ser notificados y que la falla sea corregida inmediatamente. Para tener este nivel de respuesta a fallas en la red, se requiere una muy rápida y confiable detección de fallas provista por el sistema de administración.

3.2.3.2 Administración del desempeño

Es el proceso de medición del desempeño de los dispositivos que conforman a una red, **hardware**, **software**. Los factores que reflejan el desempeño pueden ser: porcentaje de utilización, capacidad disponible, tiempo de respuesta, trafico, cantidad de errores, etc. Los administradores emplean esta información para planear crecimientos de la red, cambios de la misma o bien para mantener o incrementar su funcionalidad.

3.2.3.3 Administración de Acceso

Es el seguimiento de la utilización de los recursos de red por los usuarios. Este tipo de administración es totalmente necesaria ya que con ello el administrador puede:

- Darse cuenta que un usuario o un grupo de usuarios pueden estar abusando de sus privilegios de acceso.
- Darse cuenta que los usuarios pueden estar realizando un uso ineficiente de la red.
- Asegurase que sus usuarios estén obteniendo exactamente los recursos que ellos necesiten.
- Planear el crecimiento de la red en base a un conocimiento preciso de la actividad de los usuarios (procesamiento, utilización de periféricos, utilización de espacio de almacenamiento, etc.)

3.2.3.4 Administración de configuración

Es el proceso de configurar a los elementos de la red, desde una terminal remota. La administración de configuración involucra los procesos de inicialización, mantenimiento, y paro de los elementos individuales de la red, ya sean recursos físicos (por ejemplo: servidores, dispositivos de interconexión, etc.) o lógicos (por ejemplo: Los contadores de retransmisión del protocolo de transporte) y los estados de los mismos. Además de configurar sus parámetros.

3.2.3.5 Administración de la seguridad

Es el proceso de monitoreo y control sobre los accesos a la información dentro de la red. Alguna información almacenada en ciertas computadoras no puede ser vista por todos los usuarios, esta información es llamada información confidencial o crucial. Se debe ser cuidadoso con ella y mantener un constante control sobre los accesos a la misma y determinar cuales fueron validos. Los archivos *log* son una herramienta muy útil para realizar esta tarea.

3.2.4 Arquitectura de un sistema de administración de red

Existen tres posibles arquitecturas para construir un sistema de administración de red:

La arquitectura centralizada . Significa tener una sola maquina en toda la red que se encargue de correr las aplicaciones del sistema y allí se almacenen también los datos que requieran estas aplicaciones.

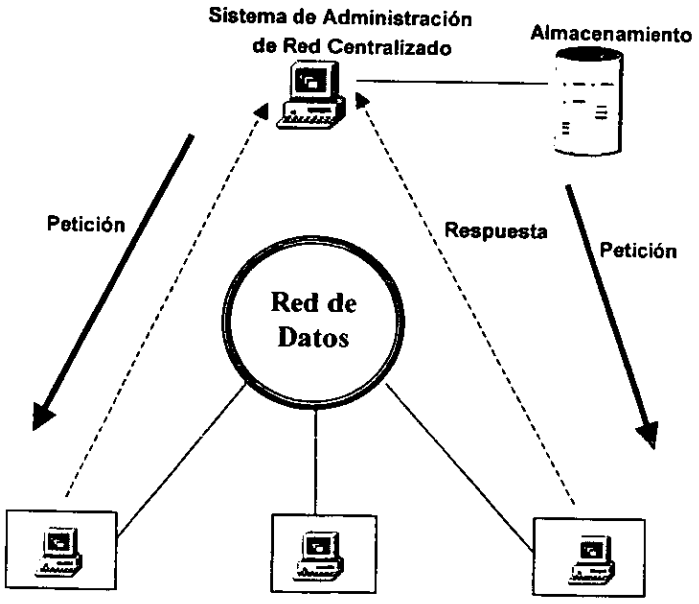


figura 3.1 arquitectura centralizada para un sistema de administración de red.

La arquitectura distribuida. llamada así ya que se tienen varios sistemas de administración corriendo simultáneamente en varias maquinas distribuidas a través de la red. Bajo esta arquitectura cada sistema puede administrar por ejemplo una parte especifica de la red llamada región, tal es el caso de una subred o de un edificio o un conjunto de ellos.

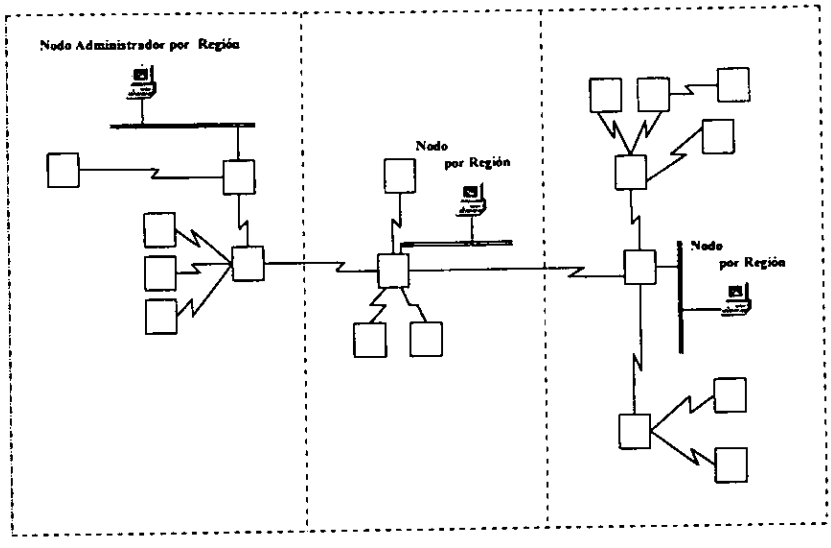


figura 3.2 Arquitectura distribuida para un sistema de administración de red.

La arquitectura mixta. Esta tercera arquitectura combina la arquitectura centralizada con la distribuida en una arquitectura jerárquica. En este tipo de arquitectura existirá un sistema principal, que almacenara toda la información esencial de la red y a su vez se delegarán tareas de administración especificas a otros sistemas en la red. Esta combinación de arquitecturas es muy poderosa.

3.2.5 Elementos de un sistema de administración de red

Llamaremos a cada elemento (servidores, dispositivos de interconexión, etc.) de una red administrada *nodo*. Habrá nodos que estén capacitados para administrar, es decir, ejecutar tareas que lleven a detectar fallas, medir desempeño, observar el acceso a recursos, etc., además de almacenar datos que estas tareas requieran o generen. Estos nodos son llamados *nodos administradores.*

También existirán nodos que estén capacitados para ser administrados, estos reciben el nombre de *nodos administrados.*

Los nodos administradores y administrados son los elementos primarios de un sistema de administración. Dentro de cada uno de estos nodos se encuentran otros elementos del sistema que permiten que la interacción entre los elementos primarios se lleve a cabo.

En el siguiente esquema se muestra a un nodo administrador y a un nodo administrado, dentro de ambos se pueden apreciar los elementos citados en el párrafo anterior.

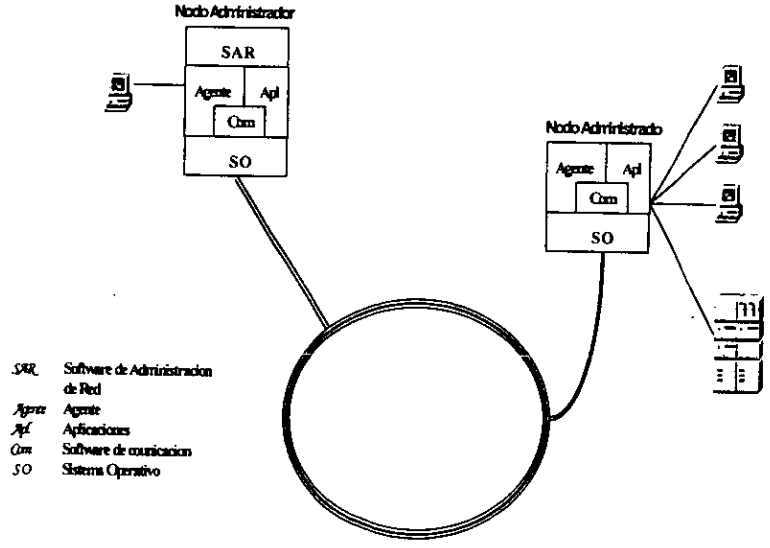


Figura 3.3 elementos de un sistema de administración de red.

Como podemos apreciar en la figura 3.3, ambos nodos contienen un elemento llamado *agente*.

*Un agente es una colección de **software** dedicado a las tareas de administración*

Cada agente realiza las siguientes tareas:

- Obtiene información sobre el estado y actividades del nodo en que reside.
- Almacena localmente los datos recopilados.
- Responde a las peticiones que un nodo administrador le envía como :
    - Cambiar un parámetro en el nodo.
    - Transferir datos del estado y de las actividades del nodo.

Los agentes se encuentran presentes en diversos nodos de la red como son: PC´s, Estaciones de trabajo, concentradores, puentes, **switches**, enrutadores, etc.

Como también vemos, en la figura 3.3, los nodos administradores requieren de un elemento llamado: **Software** de administración de red (*SAR*).

El software de administración de red, permite realizar las tareas de administración sobre los nodos administrados y los nodos administradores, incluyendo el nodo en el que reside.

Ademas de los agentes y el SAR, todas las nodos y dispositivos de interconexión pueden contar con: Sistema operativo (SO), software de aplicación (Apl) y software de comunicación (Com.).

3.2.5.1 Arquitectura del software de administración de red (SAR) .

El **software** de un sistema de administración de red, generalmente se divide en tres categorías:

- *Software de presentación* . La interacción entre un usuario y el **software** de administración de red es provista por el **software** de presentación que es una interface gráfica. Dicha interface es necesaria en cualquier sistema de administración con el fin de permitir al usuario administrar y controlar la red, por una vía sencilla que le provea de comandos para ejecutar acciones de administración y analizar los datos que estas acciones generen. La llave para lograr un efectivo sistema de administración de red es una interface unificada, es decir, la interface deberá ser capaz de concentrar a todos los dispositivos de la red en un mapa o esquema que represente fidedignamente su topología, esto permitirá al usuario tener una administración heterogénea de su red.

- *Software que realiza las tareas de administración.* En general el **software** que realiza las tareas de administración de red esta organizado en tres capas. En la capa más alta encontramos un colección de aplicaciones que se encargan de las tareas de administración sobre las diferentes áreas (fallas, acceso, configuración, desempeño y seguridad). En el nivel medio encontramos módulos que implementan funciones primitivas y de propósito general, como son: la generación de alarmas o resúmenes de datos, estas funciones son utilizadas por las tres capas. El nivel más bajo es donde se encuentra el servicio de transporte de datos, este servicio lo constituye un protocolo de administración de red, usado para intercambiar información de administración entre nodos administradores y agentes.

- *Software de soporte de administración de red.* Para realizar sus funciones, el **software** que realiza las tareas de administración de red necesita tener acceso a una base de información de administración local conocida como **MIB (Management Information Base)**. El **MIB** local de cada agente contiene información que refleja la configuración, el comportamiento del nodo en que reside y parámetros que pueden ser usados para controlar la operación del mismo. El manejo de la **MIB** es ejercido por cada agente, el cual puede extraer información de la **MIB** y ponerla a disposición. Toda información contenida en una **MIB** es conocida como información administrativa.

3.2.5.2 Proxies (delegados)

Hasta ahora hemos supuesto que cada elemento dentro de la red tiene un agente y todos se comunican con el mismo protocolo de administración. Esto no siempre es posible en la realidad ya que en una red que va a ser administrada puede contar entre sus elementos equipos muy viejos que no soportan los estándares de administración que se desean usar, o bien componentes como **modems** y multiplexores que no soportan software adicional.
Para manejar estos casos es común tener una agente sirviendo como ***Proxy o delegado.*** Cuando un agente desempeña un rol de delegado, este actúa como mediador entre un nodo y el nodo administrador. Si se cuenta en la red con elementos que no sean capaces de "hablar" con el mismo estándar de administración es necesario utilizar un delegado. El nodo administrador enviara su petición (de información o de control) a un delegado para que este la transfiera en una forma apropiada al elemento deseado. Cuando la petición es resuelta el elemento deberá transmitir su respuesta en el mismo camino en que la petición llego.

3.3 Tareas primordiales en las que se basa un sistema de administración de red.

Para llegar a sus objetivos, un sistema de administración de red se basa en dos acciones : El monitoreo de red y el control de red. El monitoreo de red es una acción de *lectura*, cuya función es obtener información acerca del estado y comportamiento de los elementos de la red. El control de red es una acción de *escritura*, cuya función es alterar los parámetros de los componentes de la red para realizar funciones especificas.

Las áreas funcionales que la administración de red debe cubrir (fallas, desempeño, acceso, configuración y seguridad) involucran al monitoreo y control. Sin embargo el monitoreo hace énfasis en las tres primeras áreas, las ultimas dos están más relacionadas con el control.

3.3.1 Monitoreo de red

El monitoreo de red consiste en observar y analizar el estado y comportamiento de cada elemento de la red administrada

El aspecto mas importante dentro de un sistema automatizado de administración de red, es el monitoreo. Tal es su importancia que muchos de estos sistemas consisten solo de funciones de monitoreo, no incluyendo las funciones de control.

3.3.1.1 Clasificación de la información obtenida por el monitoreo

El propósito del Monitoreo es obtener información. Esta puede ser dividida en :

- *Información estática*: Información referida a la configuración de cada elemento en la red. Por ejemplo el número de identificación de los puertos en un concentrador, los nombres de los archivos de dispositivo asociados a una partición de disco, etc.

- *Información dinámica*: Información relacionada con eventos en la red, como un cambio en el estado de un protocolo, o la transmisión de un paquete en la red.

- *Información estadística*: Información que es derivada de la información dinámica, como el promedio del número de paquetes transmitidos por unidad de tiempo por computadora.

La información estática es usualmente generada por el elemento involucrado, por ejemplo un concentrador contiene su propia información de configuración. Esta información puede estar disponible al sistema monitoreo directamente desde el elemento si este tiene un agente apropiado.

La información dinámica es recolectada y guardada por el nodo de red que la genera, sin embargo, mucha de esta actividad puede ser observada y almacenada por otro nodo

conectado a la misma red. El termino *monitor de red* o *nodo monitor* es usado para referirse a un dispositivo en la red que observa las actividades de los demás nodos conectados, así como el tráfico de paquetes depositados en la red.

La información estadística puede ser obtenida por cualquier sistema que tenga acceso a una fuente de información dinámica. La información estadística es usualmente generada por un monitor de red. Para ello será necesaria la transmisión de la información dinámica hacia el monitor, donde este la procesará y la analizará. Si el monitor no tiene acceso a la información dinámica de un nodo, entonces el nodo mismo tendrá que procesar y analizar su información para posteriormente enviar el resultado de los cálculos al monitor.

3.3.1.2 Elementos y configuración de un sistema de Monitoreo

Los elementos en términos funcionales que componen a un sistema de monitoreo de red son:

Objetos administrados: Son objetos que representa recursos (**hardware** y **software**) en la red.

Agente: Es un conjunto de **software** dedicado a tareas de administración, el cual manipula la información administrativa contenida en una **MIB** y comunica esta información a un nodo monitor. La inclusión de un agente en un nodo, lo hace monitoreable.

Modulo administrador: Este modulo se encuentra en el nodo monitor y realiza la función básica de recuperar información de los agentes en la red.

Aplicaciones de monitoreo: Son los programas de aplicación que realizan tareas sobre los datos recaudados y que presentan sus resultados a los usuarios.

También en un sistema de monitoreo existen dos elementos primarios: *Los nodos monitores* (definidos en el punto anterior), y los *nodos monitoreados*.
El nodo monitor es por si mismo un elemento de la red y por lo tanto un objeto administrado monitoreable, dicho nodo generalmente incluye un agente. Es de vital importancia monitorear el estado y comportamiento del nodo monitor para garantizar que este siga realizando de una manera eficiente sus funciones.

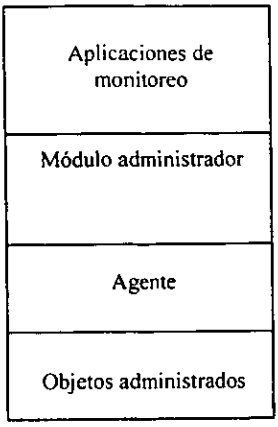


Figura 3.4 Elementos de monitoreo en un nodo monitor.

En un nodo monitoreado comúnmente solo incluye dos elementos de monitoreo: El agente y los objetos administrados.

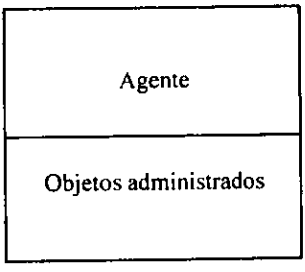


Figura 3.5 Elementos de monitoreo en un nodo monitoreado

Existen varias configuraciones que un sistema de monitoreo puede adoptar. La configuración más simple es la más común, dicha configuración requiere que el nodo monitor y el nodo monitoreado compartan el mismo protocolo de administración y la misma sintaxis y semántica de MIB.

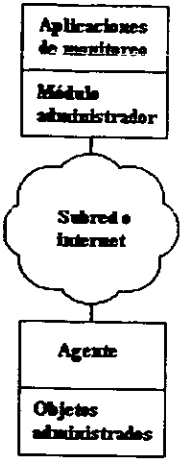


Figura 3.6 Configuración básica de un sistema de monitoreo

Un sistema de monitoreo puede también incluir uno o más agentes que monitorean las actividades en la red desde fuera del o los nodos monitores, estos agentes son conocidos como *monitores externos* o *monitores remotos*, esta configuración se ejemplifica en la siguiente figura.

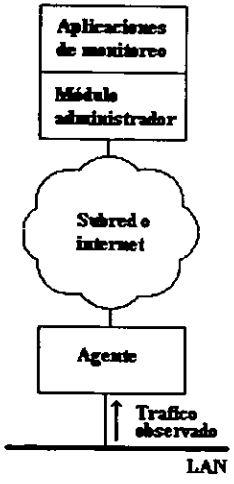


Figura 3.7 Configuración con un monitor externo

Como hemos mencionado anteriormente no siempre podemos contar con que todos los elementos en nuestra red son capaces de adaptarse a el mismo estándar de administración, entonces tendremos que emplear un agente **proxy** para poder tener comunicación con dichos elementos, en la gráfica siguiente se muestra la configuración mencionada.

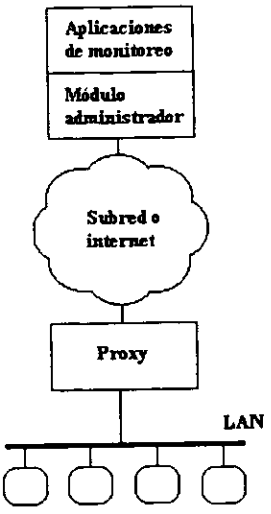


Figura 3.8 Configuración con agente proxy

3.3.1.3 Polling y reporte de eventos

Como mencionamos anteriormente la información que es útil para propósitos de monitoreo es recolectada y guardada por los agentes y transferida al módulo administrador en un nodo monitor.

Dos técnicas son usadas para poner disponible la información del agente al módulo administrador: El ***polling*** *(Encuesta)* y el *reporte de eventos.*

El **polling** es una interacción petición/respuesta entre el módulo administrador y el agente. El módulo administrador puede realizar una petición de información a cualquier agente (siempre y cuando este autorizado), y el agente responderá con la información de su **MIB**.

Un sistema de monitoreo puede utilizar el **polling** entre sus elementos para diversos fines como son: obtener condiciones que periódicamente se actualizan, o para investigar una área en detalle después de que ha sido descubierto un problema en ella.

En el reporte de eventos, la iniciativa de transferencia es del agente y el módulo administrador esta actuando como receptor esperando por información.
Un agente puede generar un reporte periódicamente para dar al sistema de administración el estado actual de cualquier elemento, también puede generar un reporte cuando un evento significativo (por ejemplo un cambio de estado) o inusual (por ejemplo una falla) ocurre.

El período de reporte (el tiempo que debe esperar el agente para enviar la información al módulo administrador) es definido por el agente, pero puede ser modificado por el nodo monitor.

El reporte de eventos es muy útil para detectar problemas tan rápidamente como estos ocurren. Es más eficiente que el **polling** para monitorerar objetos cuyos estados o valores cambien con poca frecuencia.

Las dos técnicas son muy utilizadas, un sistema de monitoreo regularmente emplea ambos métodos.

3.3.1.4 Monitoreo de fallas

El objetivo de este monitoreo, es identificar fallas lo más rápido que sea posible, después de que estas ocurran, además de ayudar a determinar sus causas así como la acción correctiva que puede ser tomada.

Funciones del monitoreo de fallas

El monitoreo de fallas, básicamente deberá detectar y reportar fallas. Como mínimo, el agente deberá mantener *un archivo log o archivo resumen* que contenga los eventos significativos y errores que se generen, la información en estos archivos deberá estar disponible para los nodos monitores autorizados. Si el monitoreo de fallas puede implementar un método **polling** o un método de reporte de eventos, estos archivos **log** serán mucho más confiables.
El agente envuelto en este tipo de monitoreo deberá tener la capacidad de reportar fallas a uno o más sistemas de administración.

Un buen monitoreo de fallas, debe anticiparse a las mismas, esto puede lograrse definiendo *valores de umbral o frontera*. Estos valores son limites preestablecidos que están cerca de ser una falla, cuando los resultados de un evento rebasan los valores de umbral definidos, se genera una alarma, de esta manera se detectan situaciones que puedan degenerar en una falla para dispositivos de interconexión de red, servidores, etc. Por ejemplo, en un servidor **UNIX**, un sistema de archivos al 100% puede generar varios problemas tanto administrativos como de funcionalidad del mismo servidor, más aún cuando se trata del sistema de archivos / . Para este caso se pueden definir valores frontera de 90% , un sistema de archivos a esta capacidad no representa graves problemas, si se rebasa este valor el administrador del servidor puede ser avisado para que ponga atención en el asunto y este no llegue a convertirse en una falla.

El monitoreo de fallas deberá también asistir en el aislamiento y diagnostico de la falla. En una situación compleja, las fallas serán diagnosticadas, aisladas, y finalmente corregidas por el esfuerzo mutuo entre el administrador humano y los programas de monitoreo.

En la detección de fallas, es necesario que el sistema de monitoreo cuente con una eficiente interfaz, que permita interactuar de manera sencilla y rápida al administrador humano con los programas de monitoreo. Algunas de las pruebas que una interfaz debe tener disponible en forma de comandos, son las siguientes:

- Prueba de conectividad
- Prueba de integridad de datos
- Prueba de integridad de protocolos
- Prueba de saturación de datos
- Prueba de saturación de conexión
- Prueba de tiempo de respuesta
- Prueba de funcionalidad del protocolo de monitoreo

3.3.1.5 Monitoreo de desempeño

Un requisito absoluto para la administración de una red es la posibilidad de medir el desempeño de la misma. Nosotros no podemos esperar administrar y controlar una red sin antes monitorear el desempeño.

Indicadores de desempeño

Una de las dificultades a las que se enfrenta un administrador de red es la selección y el uso de los indicadores apropiados para medir el desempeño de su red. Hay un gran número de indicadores, pero en este punto hablaremos solo de los indicadores más útiles.

Existen dos categorías de indicadores: *orientados a servicios* y *orientados a eficiencia.*
Los indicadores orientados a servicios son los que se relacionan con la satisfacción de las necesidades de los usuarios por medio de los servicios que proporciona la red (almacenamiento de datos, correo electrónico, impresión, etc.). Por otro lado los indicadores orientados a eficiencia se relacionan con la funcionalidad de los elementos de la red.

Orientados a servicios

Disponibilidad. Es el porcentaje de tiempo que una red, un elemento, o una aplicación esta disponible para un usuario. Dependiendo de la circunstancia de una red, una alta disponibilidad puede ser muy importante, por ejemplo si en una oficina de reservaciones de boletos de avión su red esta fuera por 10 minutos, puede causar $10,000 en perdidas, en un banco si la red esta fuera por una hora, puede causar millones de dólares en perdidas.

La disponibilidad esta basada en la confiabilidad de los componentes individuales de una red. La confiabilidad es la probabilidad de que un componente desempeñe una función especifica por un tiempo especifico, bajo condiciones especificas.

Tiempo de respuesta. Es el tiempo en que una respuesta aparece en la terminal de un usuario después de que este realizo una petición. Es decir es el tiempo entre la ultima tecla oprimida por el usuario y el principio del despliegue del resultado en el monitor.
Un rápido tiempo de respuesta es la llave de la productividad cuando se trabaja con aplicaciones de computo. Cuando un usuario y una computadora interactuan de modo que ninguno tenga que esperar al otro, la productividad se incrementa notablemente y la calidad se mejora. Con el tiempo de respuesta podemos identificar cuellos de botella o lugares donde sea posible que se formen.

Precisión El porcentaje de tiempo en el cual no ocurren errores en la transmisión ni en la entrega de información. Sobre la precisión generalmente el administrador no tiene control, ya que depende directamente de las capas de protocolo inferiores. Sin embargo este indicador puede ser muy útil al advertir posibles fallas, si la precisión es muy baja. Este indicador puede advertirnos fallas como: posible falla en la línea de transmisión, o posibles interferencias o ruido que deben ser corregidos.

Orientados a eficiencia

Rendimiento. Es la velocidad en la cual un evento de una aplicación (Transferencia de archivos, mensajes, etc.) ocurre. El rendimiento puede verse afectado por un mal funcionamiento de los elementos de la red a los cuales la aplicación este asociada, por ejemplo si una transferencia de archivos es lenta, probablemente la velocidad del canal de comunicación ya no sea la adecuada, o bien puede haber un cuello de botella en algún lugar.

Utilización. Es el porcentaje de tiempo que un recurso esta en uso sobre un período de tiempo dado. El mas importante uso de este identificador es la búsqueda de cuellos de botella potenciales y áreas de congestión , además de que usualmente el tiempo de respuesta se incrementa exponencialmente a razón de el incremento en la utilización de un recurso.

Funciones del monitoreo de desempeño.

El monitoreo del desempeño tiene dos funciones básicas : medición del desempeño, el cual obtiene estadísticas acerca de los elementos en la red; análisis del desempeño, como su nombre lo indica es el análisis de los datos estadísticos.

La medición del desempeño es casi siempre llevada a cabo por los agentes dentro de los dispositivos en la red (computadoras, enrutadores, puentes, etc.) . Estos agentes están en posición de observar la cantidad de tráfico de paquetes dentro y fuera de un elemento, el número de conexiones en los capas de protocolo (red, transporte, y aplicación), el tráfico

por conexión, y otras mediciones que proveen un detallado esquema del comportamiento de un elemento de la red.

3.3.1.6 Monitoreo de acceso

Es el seguimiento del uso de los recursos de la red por los usuarios. Es necesario conocer con precisión toda la información relacionada con los accesos a los recursos de la red, para así poder tener un control más efectivo sobre los mismos, y poder planear el crecimiento de la red con datos reales.

Algunos de los accesos a recursos que pueden ser observados son:

- Accesos a equipo de computo: servidores, computadoras personales, impresoras, y en general cualquier periférico (discos duros, unidades de cinta, cdrom, etc.)
- Accesos a sistemas y programas: aplicaciones y programas dentro de los servidores, bases de datos, etc.
- Accesos a servicios: Incluye todos los servicios de comunicación y servicios de información disponibles para los usuarios de la red (correo electrónico, WWW, gopher, etc.)
- Accesos a líneas de comunicación: Entrada a la red vía líneas telefónicas, etc.

La información que deberá obtenerse de cada usuario estará basada en los requerimientos del administrador y el tipo de recurso que sea monitoreado. Los siguientes son ejemplos de algunos datos que pueden ser recopilados por cada usuario:

- Identificación de usuario. Es el nombre con el que el o los sistemas conocen al usuario
- Máquina desde donde se realiza la conexión
- Número de paquetes por acceso
- tiempo de inicio y fin del acceso
- Recursos usados.

3.3.2 Control de Red

El control de red es la acción de modificar parámetros establecidos en los elementos de la red (servidores, dispositivos de interconexión de red, etc.) con el fin de realizar tareas de administración.

Existen dos áreas funcionales que la administración de red debe cubrir, en las cuales el control se desarrolla de una manera amplia, y son : configuración y seguridad.

3.3.2.1 Control de configuración.

La administración de configuración se fundamenta en el siguiente principio básico: el nodo administrador tiene acceso a los elementos de la red (ya sea físicos o lógicos) para poder ejercer sobre ellos, paros, incializaciones, configuración de sus parámetros y estados. Estas acciones no son más que acciones de control también referidas como *control de configuración.*

Mientras la red esta en operación, la administración de configuración es responsable de realizar cambios en las configuraciones de los elementos de la red, en respuesta a otras funciones de administración de red o bien por petición del administrador. Por ejemplo si a través del monitoreo de fallas se detecta y aísla una falla, por medio del control de configuración se puede alterar la configuración de uno o varios elementos para eludir la falla mientras esta se repara.

Funciones del control de configuración

Las funciones de control de configuración son las siguientes:

Definición de la información de configuración. La información de configuración describe la naturaleza y el estado de los recursos que son de interés para ser administrados. Es decir la información de configuración incluye una especificación de los recursos bajo administración y los atributos de los mismos. Los recursos administrados pueden ser físicos (computadoras, dispositivos de interconexión, etc.) y/o lógicos (temporizadores, contadores, circuitos virtuales, etc.). Los atributos incluyen por ejemplo, nombre del elemento, dirección, características de operación, versiones de **software** , etc.

La información de configuración puede ser estructurada de las siguientes formas:

- Una simple lista de campos, en cada campo se encuentra un valor simple.
- Una base de datos orientada a objetos. Cada elemento de interés es representado por uno o más objetos. Cada objeto contiene atributos cuyos valores reflejan las características de cada elemento representado.
- Una base de datos relacional. Cada campo dentro de la base contiene valores que reflejan características de los elementos administrados. Además de que la estructura de la base de datos refleja vínculos entre los elementos de la red.

Esta información debe ser accesible al nodo administrador, generalmente, la información es almacenada cerca del recurso administrado en cuestión, en donde se encuentre el agente, si dicho agente se encuentra dentro del elemento administrado entonces allí también se almacenará la información.

Configuración y modificación de atributos. El control de configuración deberá habilitar al nodo administrador para que remotamente pueda configurar y modificar los atributos.

Existe una limitante a esta capacidad: Algunos de los atributos reflejan la realidad en un recurso y no pueden, por su naturaleza, ser modificados remotamente. Por ejemplo, un atributo puede ser el número de puertos en un concentrador, el número de puertos puede ser únicamente cambiado por una acción física en el concentrador, no por una acción remota; no obstante se puede remotamente habilitar o deshabilitar puertos en cualquier tiempo.

Una modificación a un atributo será una modificación a la información de configuración en la base del agente.
En general las modificaciones pueden ser clasificadas como:

- Actualización de la base únicamente. Cuando un administrador ejecuta un comando dentro del sistema de administración hacia un nodo administrado y este comando repercute en un cambio de valor en los atributos de la base, pero no cambia atributos de configuración ni de operación del nodo. Por ejemplo cuando es cambiado el nombre y la dirección del administrador de la red (nombre y dirección de la persona responsable de la red).
- Actualización de la base + modificación del recurso. En adición a la actualización de la base de datos en el agente, un comando puede tener un efecto en los atributos de configuración y operación del recurso administrado. Por ejemplo, si el estado del atributo de un puerto físico en un concentrador es configurado como "deshabilitado" entonces el agente no solo actualiza el atributo en la base de datos, si no también deshabilita el puerto.
- Actualización de la base + acción. En algunos sistemas de administración no existen comandos de acción directa disponibles para los administradores, sin embargo existen parámetros en la base de datos que cuando son configurados, producen una cierta acción. Por ejemplo, un puente puede mantener definido un parámetro de reinizialización en su base de datos, si este parámetro es configurado como "verdadero" por un administrador autorizado, el puente realizará el proceso de reinicialización, cuando este proceso este terminado el parámetro regresará a su estado original "falso".

El control de configuración deberá permitir la configuración y modificación de los atributos de los recursos administrados sin que toda la red o parte de ella sean dados de baja.

Definición y modificación de vínculos. Los vínculos describen una asociación o conexión entre elementos de la red. Ejemplos de vínculos son los siguientes: una topología, estructura jerárquica de los componente, conexión física o lógica.
El administrador deberá tener el control sobre estos vínculos en el sistema de administración y en cualquier momento podrá adicionar, borrar, y modificar los mismos.

Inicialización y terminación de la operación de la red. El administrador de red podrá a través de su sistema de administración, terminar o iniciar la operación de toda la red o bien de alguna subred. El proceso de inicialización incluye la verificación de que todos los recursos se hayan "levantado " correctamente, así como los vínculos, y la notificación a los usuarios y al mismo administrador. Para el proceso de terminación deberá incluir la capacidad de notificar a los usuarios antes de que este proceso sea terminado.

Reporte de los atributos en la configuración y exploración de vínculos. Un administrador puede requerir información acerca de los atributos existentes en la base de datos de cualquier agente, además de explorar las relaciones entre los objetos administrados, esta no es una acción de control si no de monitoreo, pero el control de configuración deberá proveer esta facilidad al administrador basado en las funciones de monitoreo anteriormente definidas.

3.3.2.2 Control de seguridad.

El recurso más valioso dentro de una compañía o institución es la información. Las computadoras, se han convertido en una fuente de almacenamiento y proceso de dicha información, por lo que se ha generado la necesidad de salvaguardar a los sistemas de cómputo de posibles ataques. Al aparecer los esquemas de cómputo distribuido y la utilización de redes para la comunicación de datos han producido un gran cambio en la concepción de la seguridad de computo, por lo que se ha pensado en la seguridad de los datos que viajaban por la red y se delinearon acciones para proteger dichos datos. El nombre genérico de la colección de herramientas y estrategias diseñadas para proteger información dentro de un sistema de cómputo este o no en red, se le conoce como *seguridad en computo.*

La administración de seguridad o control de seguridad, tendrá que ver con todos los aspectos de la seguridad en computo, y deberá ejercerse sobre los recursos que son administrados, incluyendo por supuesto al propio sistema de administración.

El control de seguridad en cómputo define dos requerimientos básicos:

- *información confidencial.* La información y los parámetros de configuración en todo los elementos de la red debe estar disponible (escritura, lectura, o ejecución) solo para usuarios autorizados.
- *Integridad.* La información y los parámetros de configuración en todos los elementos de la red, pueden ser modificados únicamente por usuarios autorizados.

Ataques a la seguridad

Los ataques a los sistemas de computo son muy variados, y para tomar medidas sobre ellos, es necesario catalogarlos. A continuación definiremos diferentes tipos de ataques.

Conceptualmente existe un solo flujo de la información, que va de la fuente de información al destino de la información. Los tipos de ataques tienen que ver con este flujo.

- *Interrupción.* Un valor en la red es destruido, no utilizable o no disponible. Como ejemplo podemos citar: La destrucción de una pieza de **hardware** , como un disco duro, el corte de una línea de comunicación, el apagado abrupto de un servidor etc.; o bien la deshabilitación de un elemento lógico como un sistema de archivos.
- *Intercepción.* Un individuo no autorizado obtiene acceso a un valor dentro de la red. El individuo puede ser una persona, un programa o una computadora. Por ejemplo la intervención de una transmisión de archivos para realizar una copia ilícita de los mismos.
- *Modificación.* Un individuo que no solo obtiene acceso a un valor de la red si no también modifica la información. Por ejemplo el cambio de datos en un archivo, la modificación de un programa para que se comporte de una manera diferente, o la modificación de un mensaje enviado por la red.

*Ataques al **hardware.*** Son los ataques ejercidos directamente al **hardware** para causar un daño físico. Los ataques incluyen daños accidentales e intencionales.

*Ataques al **software*** Los ataques al **software** incluyen ataques al sistema operativo, utilerias, y programas de aplicación. El **software**, en especial los programas de aplicación, son fáciles de borrar y este es un ataque muy grave, además existe la posibilidad de alterar o dañar al **software** dejándolo no utilizable o en el peor de los casos sigue funcionando pero realiza procesos o tareas que no están autorizadas, pudiendo dañar los datos con los que este trabaja, los virus entran dentro de este tipo de ataques.

Ataques a los datos. Los datos son una parte muy vulnerable dentro de un sistema de computo. Los ataques a los datos los podemos dividir en dos: accesos no autorizados (lectura de los datos) y modificación de los mismos.

Ataques a las líneas de comunicación. Este tipo de ataques tiene que ver con la manipulación de los datos que viajan por la red. Por ejemplo el agresor puede intervenir los paquetes que son transmitidos para modificar las datos en ellos. O bien modificar el flujo de paquetes incrementando estos para causar un daño en la comunicación de la red. También el agresor puede tomar una dirección asignada de la red y utilizarla para sus propios fines. En general este tipo de ataques son muy difíciles de detectar debido a su naturaleza.

Ataques al Sistema de administración de red. Como un sistema de administración involucra un conjunto de programas de aplicación, elementos de **hardware** y bases de datos, los ataques citados anteriormente pueden ser considerados como ataques al sistema de administración de red, además de dichos ataques podemos definir ataques específicos a un sistema de administración:

1. Usuario enmascarado. Es un usuario no autorizado, que trata de realizar tareas de administración de red. Puede tener acceso a las aplicaciones de administración y a la información de los objetos administrados.
2. Sistema enmascarado. Es una computadora que trata de obtener los derechos del nodo administrador sobre los elementos administrados.
3. Interferencia con el intercambio de información e instrucciones entre el nodo administrador y los agentes. Un ataque grave es la observación del trafico del protocolo de administración por un extraño para extraer información. Mas dañina es la modificación de este tráfico para romper la operación del agente con los recursos o el nodo administrador.

Funciones del control sobre la seguridad.

La seguridad en computo y en una red consiste en un conjunto de herramientas, servicios y mecanismos, los cuáles sobrepasan el objetivo de esta tesis, así es que en este punto se trataran únicamente los temas de seguridad concernientes al control sobre la seguridad en un sistema de administración de red.
Hay tres funciones básicas de seguridad que un sistema de administración debe cumplir:

1. *Mantenimiento de la seguridad de la información.* Un sistema de administración de red deberá proveer medidas para limitar y validar el acceso a la información que éste maneja. Ejemplos de estas medidas son: derechos de accesos a la información, contraseñas, validación de la información, etc.

2. *Seguimiento de actividades.* Se debe dar seguimiento a toda actividad realizada, a través del registro de eventos, monitoreo del uso de recursos, reporte de violaciones a la seguridad del sistema.

3. *Control de acceso a recursos.* Un importante servicio que un sistema de administración deberá proveer es el control de los accesos a los recursos, este proceso involucra la validación y autenticación de cualquier, usuario, máquina o programa antes de permitir dicho acceso. Ya que se haya realizado la validación, un usuario podrá crear o borrar objetos administrados, tener acceso a las fuentes de información, cambiar los atributos, etc.

Capítulo 4

Alternativas y evaluación de diferentes protocolos de administración

4.1 introducción

Los administradores de red necesitan un método consistente para obtener información de todos los componentes de su red. Muchas veces los administradores se basan en las herramientas genéricas de sus sistemas (**ping, arp, ifconfig, disk, printer**, etc.) para realizar tareas de monitoreo y en base en la información obtenida realizar acciones de control sobre la red. Estas herramientas son generalmente fáciles de usar y no necesitan implementaciones anexas para su funcionamiento, generalmente están integradas a los sistemas operativos. Sin embargo estas herramientas no fueron diseñadas propiamente para la administración de una red, generalmente basan sus funciones en un intercambio de paquetes a nivel capa de red y no pueden manejar la cantidad adecuada de información para proporcionar una información administrativa confiable. Para conseguir la información que necesitan de los elementos de la red, utilizan la técnica de **polling** y no son capaces de generar reportes de eventos por si mismos, además de no proveer abundante información acerca de los sucesos que están ocurriendo en la red, esta información en muchos casos no es suficiente para tomar las decisiones más acertadas para la administración de la red. Por esta razón surgió la necesidad de desarrollar tecnologías especificas para la administración de red, y es así como se originan los protocolos de administración de red.

Los protocolos de administración de red son una colección de especificaciones de comunicación capaces de manipular información administrativa a través de los elementos de una arquitectura de administración de red.
Los sistemas de administración de red, ejercen sus funciones y tareas basados en la manipulación y comunicación de información que proveen dichos protocolos.

Existen dos grandes familias de protocolos de administración: ***SNMP*** (por sus siglas en inglés ***Simple Network Management Protocol***) y ***CMIS/CMIP*** (de sus siglas en inglés ***Common Management Information Services/Common Management Information Protocol***). Ambos protocolos proveen un camino uniforme para acceder a cada elemento de una red con el objeto de obtener información administrativa y proporcionar control. Estos protocolos tienen también la capacidad de acoplarse al modelo **OSI**, de hecho **CMIS/CMIP** fue desarrollado totalmente bajo estos estándares.

4.2 SNMP

El origen de **SNMP** (Protocolo simple de administración de red) fue provocado por el desarrollo descomunal de **Arpanet** hoy **Internet**. El creciente número de subredes que se unían a **Arpanet** hizo imposible que solo unos cuantos expertos en la red pudiesen resolver diversos problemas presentados en las diferentes subredes existentes, es así como los desarrolladores de **TCP/IP** pensaron en la creación de un protocolo estándar que proporcionara un camino para el monitoreo y control de toda la red, además de ser fácil de aprender para otras personas que tuvieran bajo su responsabilidad una subred. **SNMP** fue diseñado como respuesta a los problemas anteriormente mencionados.

SNMP consiste en una colección de especificaciones de comunicación de red, las cuales cubren todas las funciones básicas de la administración de red en un método que no somete a un gran esfuerzo a la red administrada.

La evolución de **SNMP** ha sido un espejo de la evolución de **TCP/IP.** Con el tiempo se desarrollo una nueva versión de **SNMP: SNMPv2**, la cual incorpora muchos de los desarrollos de el primer **SNMP** y adiciona algunas mejoras.

4.2.1 La arquitectura SNMP

Implícito en la arquitectura **SNMP** esta una colección de nodos administradores y elementos administrados. Como mencionamos en el capitulo anterior la función de los nodos administradores es ejecutar aplicaciones de administración las cuales monitorean y controlan elementos de la red, esta función de administración se aplica de igual manera a los nodos administradores en una arquitectura **SNMP** . Los elementos de la red pueden ser dispositivos como: dispositivos de interconexión de red, computadoras personales, servidores, etc., estos elementos cuentan con agentes responsables de realizar las peticiones que los nodos administradores les envían. El protocolo simple de administración de red **SNMP**, es el encargado de comunicar información administrativa entre los nodos administradores y los agentes en los nodos administrados.
La disposición de estos elementos puede adoptar cualquiera de las arquitecturas fundamentales de un sistema de administración (arquitectura centralizada, descentralizada y mixta).

Elementos de la arquitectura SNMP

Sabemos que implícitos en la arquitectura **SNMP** existen nodos administradores, elementos o nodos administrados, además de agentes, el protocolo de comunicación y una base de datos donde almacenar la información de administración.

El *nodo administrador* deberá tener como mínimo los siguientes elementos :

- Una interfaz por la cual el administrador de red pueda monitorear y controlar la red
- Un conjunto de aplicaciones para análisis de datos
- La capacidad de llevar los requerimientos de monitoreo y control a los elementos de la red
- Un base de datos que almacene la información extraída de los elementos administrados.

SNMP esta a cargo solamente de los dos últimos elementos.

Por otra parte los *agentes* deben estar en los nodos administrados y en el nodo administrador si se desea que este sea **autocontrolable** y **automonitoreable**.
A través del agente, el nodo administrador puede obtener información y ejercer acciones de control sobre los elementos de la red. **SNMP** deberá ser integrado en cada elemento administrado en forma de agente, con el cual el nodo administrador podrá establecer un vinculo de control y monitoreo con cada uno de estos elementos.

Toda la información de los elementos administrados se almacena en una base de datos llamada ***MIB***, **SNMP** proporciona una sintaxis especifica para almacenar la información dentro del **MIB**, de esta manera un nodo administrador puede entender la información proveída por otros agentes **SNMP**.

El nodo administrador y los nodos administrados son comunicados a través de un *protocolo de administración de red.* El protocolo de administración de red usado para redes basadas en **TCP/IP** es **SNMP** (simple network management protocol), el cual tiene las siguientes habilidades:

- Habilita a el o los nodos administradores para recaudar información de un objeto administrado.
- Habilita a el o los nodos administradores para establecer o modificar valores o parámetros en los agentes de los nodos administrados.
- Habilita a uno o mas agentes para notificar eventos significativos a el nodo administrador.

4.2.2 SNMP dentro del modelo de capas de protocolo TCP/IP

SNMP es parte del grupo de protocolos **TCP/IP** y fué diseñado para ser un protocolo de capa de aplicación.
El protocolo de capa de transporte en el que esta basado **SNMP** es **UDP**. Para el enrutamiento de paquetes usa a **IP** , y para las capas inferiores puede basarse en una amplia variedad de protocolos (CSMA/CD, X.25, etc.). En la siguiente figura se muestra el modelo de protocolo **TCP/IP** incluyendo a **SNMP**.

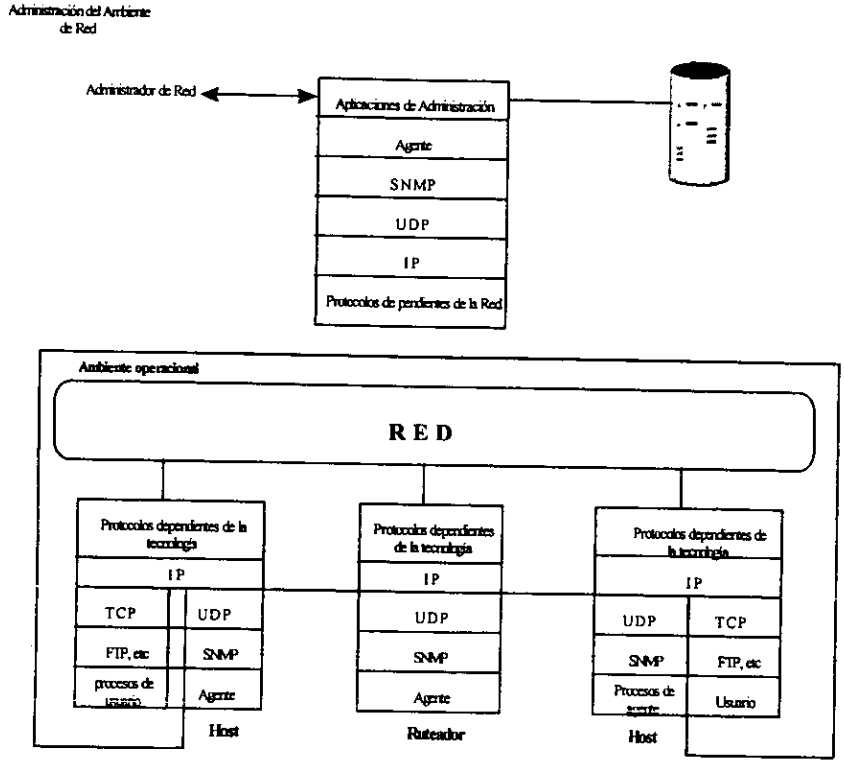


Figura 4.1 SNMP dentro del modelo de capas TCP/IP

SNMP requiere el uso de un servicio de transporte para la entrega de sus mensajes, **SNMP** no sabe si el protocolo de la capa de transporte es o no orientado a conexión. En el caso de **TCP/IP** como mencionamos en el párrafo anterior, **UDP** provee el servicio de entrega de paquetes para **SNMP**, este protocolo de transporte no es orientado a la conexión. Los puertos **UDP** que han sido asignados para la transferencia de paquetes **SNMP** son dos: Los agentes escuchan peticiones, por el puerto 161, los nodos administradores escuchan cualquier entrega o petición de información por el puerto 162.

Como **UDP** es un protocolo sin reconocimiento de paquetes, es posible que un mensaje **SNMP** pueda perderse. **SNMP** fue desarrollado para ser usado sobre un protocolo de transporte no orientado a conexión, la razón para ello es que un protocolo con estas características no incrementa la carga de paquetes sobre la red, aminorando de esta manera la eficiencia con la que los paquetes **SNMP** son recibidos por cualquier elemento dentro de la configuración administrada. **SNMP** por si mismo no provee reconocimiento de paquetes.

4.2.3 Base de datos de información administrativa (MIB), y estructura de la información administrativa (SMI)

El principio por el cual los recursos en una red pueden ser administrados es debido a que estos pueden ser representados como objetos. La colección de dichos objetos es conocida como: *Base de datos de información administrativa* ***MIB*** (***Management Information Base)***. Cada objeto es esencialmente, una variable de datos.

Cada nodo en la red deberá mantener una **MIB** que refleje el estado de los recursos administrados en ese nodo. Un nodo administrador puede monitorear y controlar los recursos en un nodo administrado a través de la lectura y modificación de los valores de los objetos en la **MIB.**

Para que la **MIB** pueda ser útil y proporcionar la información requerida por el nodo administrador, deben cumplirse dos objetivos:

- El objeto u objetos usados para representar un recurso particular deben ser los mismos en cada nodo.
- Se debe usar un esquema común para la representación de la información.

La estructura de la información administrativa ***SMI (Structure of Management Information)*** define el esquema general dentro del cual una **MIB** puede ser definida y construida. La **SMI** identifica los tipos de datos que pueden ser usados en el **MIB** y como los recursos dentro de la misma son representados y llamados.

La filosofía de la **SMI** es fomentar la simplicidad y vastedad dentro de la **MIB** , así es que por definición de **SMI** la **MIB** puede solamente representar tipos simples de datos: escalares y arreglos bidimensionales de escalares. **SNMP** puede recaudar solo escalares.

Para proveer un camino estándar de representación de la información administrativa, **SMI** debe proveer técnicas estándar para :

- Definir la estructura de un **MIB** particular
- Definir objetos individuales, incluyendo la sintaxis y el valor de cada objeto
- Codificación de valores de objetos.

Los objetos (variables de datos) dentro de la **MIB** son definidos usando la colección de reglas **ANS.1 (Abstract Syntax Notation One)**. Las variables pueden ser divididas en dos clases: *variables simples y tablas.* Las variables simples incluyen tipos como enteros con signo o sin signo, cadenas de caracteres, y conjuntos de datos que corresponden a las estructuras de el lenguaje C. Las tablas corresponden a arreglos bidimensionales, una sola tabla puede contener muchas variables simples y también tablas. Mientras el tamaño de las variables simples esta definido, el tamaño de las tablas puede cambiar en función de los objetos que esta representando.

Cada objeto tiene un nombre, una sintaxis y una codificación.

Nombre de variables y tablas. Cada objeto es identificado en la **MIB** por un nombre de variable. Estos nombres están definidos por **ANS.1** que establece una estructura jerárquica para cada variable, así es que el nombre de cada variable refleja su posición dentro de la jerarquía. Dicha estructura consiste de una raíz conectando a numerosos nodos, cada nodo a su vez puede tener nodos hijos. Esta definición jerárquica de nombres garantiza que si bien muchas organizaciones introducen nuevos nombres de variables, cada nombre será único y absoluto.

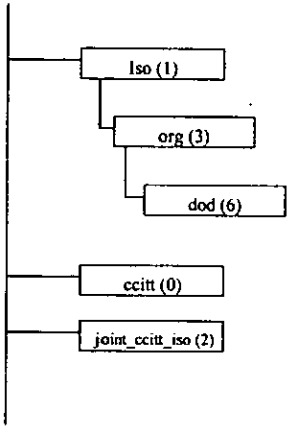


figura 4.2 Estructura jerárquica de nombres de la MIB

En el esquema se puede observar que la raíz no tiene un nombre asignado, ésta a su vez tiene tres nombres hijos: un nodo administrado por la Organización Internacional de Estandarizaciones **OSI** el cual recibe el nombre de iso(1), otro es administrado por el Comité Internacional de Telegrafía y Telefonía con nombre ccitt(0), y el tercero es administrado conjuntamente por las dos anteriores organizaciones con nombre joint-iso-ccitt(2).

Bajo el nombre iso(1), **OSI** ha destinado un nombre para el uso de otras organizaciones internacionales, org(6) en el cual un lugar fue asignado para el departamento de defensa de los Estados Unidos, dod(6), este nodo esta dispuesto a recibir a la comunidad **Internet** bajo el nombre internet(1) . El siguiente esquema muestra la estructura del nodo iso(1).

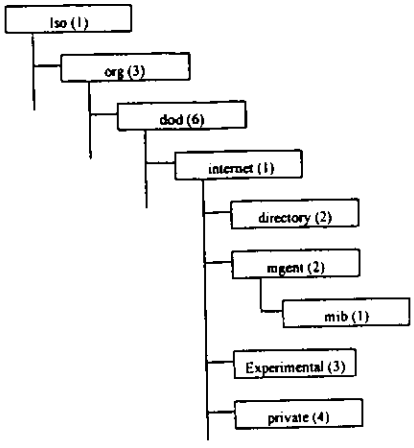


figura 4.3 El nodo OSI en la estructura jerárquica del MIB.

En el esquema podemos observar que el nodo internet(1) tiene a su vez varios nodos hijos: el nodo directorio, directory(1) esta reservado para el uso del estándar X.500 de **OSI**, el nodo experimental, experimental(3) es usado para definir objetos en experimentos de **Internet** , el nodo privado, private(4) es usado para definir objetos por la empresas privadas que hacen desarrollos, finalmente el nodo administración mgmt(2) contiene las definiciones de las bases de información de administración que han sido aprobadas, dichas definiciones están dentro del nodo **MIB**, mib(1). Hasta el momento dos versiones de **MIB** se han desarrollado: **MIB-I** y **MIB-II**, el segundo es una extensión del primero, en cualquier configuración solo una definición de **MIB** deberá estar presente.

Dentro del nodo **MIB** están definidos grupos de objetos en los que deberán estar colocados los objetos administrados, estos grupos fueron definidos por dos razones: la primera, tener un organización de los objetos administrados acorde con su función. La segunda, proveer un método para la implementación de y saber cuales objetos deberán ser implementados. En el siguiente esquema se muestran los grupos de objetos definidos para **MIB-II**.

- System
- Interfaces
- Address Translation
- IP
- ICMP
- TCP
- UDP
- EGP

Figura 4.4 MIB-II Grupos de objetos

Cada parte de la jerarquía ha sido asignada a un nivel, un *nombre* esta definido por una secuencia de niveles que denotan subjeraquias. El nombre de la jerarquía más significativa se encuentra a la izquierda. Por ejemplo la variable **MIB** en la subjerarquía ip(4) que cuenta los datagramas **IP** que llegan (ipInReceives(3)), es nombrada como:

iso.org.dod.internet.mgmt.mib.ip.ipInReceives

Cuando se reciben o envían mensajes, **SNMP** no almacena los nombres de la variables como cadenas texto. **SNMP** usa una forma numérica de representación **ANSI.1** para representar cada nombre, ya que la representación numérica es más compacta que una representación textual, esto ahorra espacio en los paquetes de transmisión.
La forma numérica de representación **ANS.1** asigna un único entero a cada nivel de la jerarquía y representa a un nombre como una secuencia de enteros. Para la variable del ejemplo anterior la secuencia de números que la representa es:

1.3.6.1.2.1.4.3

Sintaxis. Todas las variables **MIB** deben definirse y ser referidas por medio de la **Abstract Syntax Notation 1** ***ASN.1*** de la **OSI** El **ASN.1** es un lenguaje formal que tiene dos características principales: una notación formal descrita en documentos que los usuarios pueden leer y una representación codificada compacta de la misma información empleada en los protocolos de comunicación. En ambos casos, la notación formal precisa suprime cualquier posible ambigüedad tanto de la representación como del significado, esta hecho es en especial importante cuando se trabaja con computadoras heterogéneas de las que no todas utilizan la misma representación para los datos.

Además de hacer que los documentos estándar estén libres de ambigüedades, **ASN.1** ayuda a simplificar la implantación de protocolos de administración de red y garantiza su interoperabilidad.

Por ejemplo pensemos en una tabla de direcciones **IP** como en un arreglo unidimensional, en el que cada elemento del arreglo consiste en una estructura (registro) que contiene cinco elementos: una dirección **IP**, el índice entero de una interfaz que corresponde a una entrada de información, una máscara de subred **IP**, un número máximo de datagrama que el ruteador rensamblará . El **MIB** proporciona un nombre para el arreglo, si existe y permite al software de administración transformar las referencias de la tabla en variables internas apropiadas.

Por ejemplo podemos definir una tabla *ipAddrTable* utilizando la notación **ASN.1** :

ipAddrtable :: = SEQUENCE OF ipAddrEntry

donde SEQUENCE OF son palabras reservadas de **ASN.1** que definen una "ipAddrtable" como un arreglo unidimensiomal de "ipAddrEntrys" definida como:

```
ipAddrEntry :: = SEQUENCE  {
        ipAdEnAddr
          IpAddress,
      ipAdEntNetMask
          INTEGER,
      ipAdEntNetMask
          ipAddress,
   ipAdEntReasmMaxSize
   INTEGER ( 0..065535 )
}
```

codificación. La codificación de un objeto es la representación del mismo cuando es transmitido sobre la red. Los objetos en la **MIB** son codificados usando las reglas básicas de codificación **BER** asociadas con **ASN.1** . Estas especificaciones describen un método para codificar valores de cada tipo **ASN.1** como una cadena de octetos.

4.2.4 Acceso a la información administrativa

SNMP basa todas sus funciones de administración en alteraciones o inspecciones a variables dentro de una **MIB**, estos accesos a variables son conocidos como funciones de propósito general y se dividen en tres :

Obtener: Un nodo administrador recupera el valor de cualquier objeto almacenado en la **MIB** de cualquier agente.
Modificar: Un nodo administrador actualiza el valor de cualquier objeto almacenado en la **MIB** de cualquier agente.
Trap: Un agente envía un valor de un objeto que no ha sido solicitado por un nodo administrador.

Para llevar a cabo los accesos mencionados anteriormente **SNMP** maneja cinco tipos de mensajes:

1. ***Get -Request***
2. ***Get-Response***
3. ***Get-Next-Request***
4. ***Set-Request***
5. ***trap***

El nodo administrador **SNMP** usa **Get-Request** para recuperar información de un objeto de cualquier agente en un elemento administrado. Dicho agente responde a la petición con un mensaje **Get-Response.** Ejemplos de la información que puede ser recuperada son: El nombre del elemento administrado, por cuanto tiempo el elemento ha estado trabajando, el número de interfaces de red con las que cuenta el elemento.

Get-Request y **Get-Next-Request** son usadas en conjunción para obtener los valores de un tabla de objetos. Por ejemplo, **SNMP** usa la conjunción de los dos mensajes para recuperar información dentro de una tabla que contenga los estados de operación de cada tarjeta de red en un elemento administrado.

SetRequest permite que el nodo administrador a través de **SNMP** realicé la configuración o actualización de parámetros en un elemento administrado. Tales parámetros pueden ser: el nombre del dispositivo, dirección **IP**, el acceso a puertos, por este medio puede también dar de baja remotamente al elemento.

Un **Trap SNMP** es un mensaje no solicitado que un agente envia al nodo administrador. Este mensaje informa a la máquina administradora sobre la ocurrencia de un evento especifico e inusual que puede representar un problema en la red. Por ejemplo los **Traps** son usados para avisar al nodo administrador que un conexión con un servidor a fallado, o que la capacidad de un disco esta próxima al 100%.

Es importante recalcar que este tipo de alteraciones y lecturas de la información no repercuten en la estructura de la **MIB**.

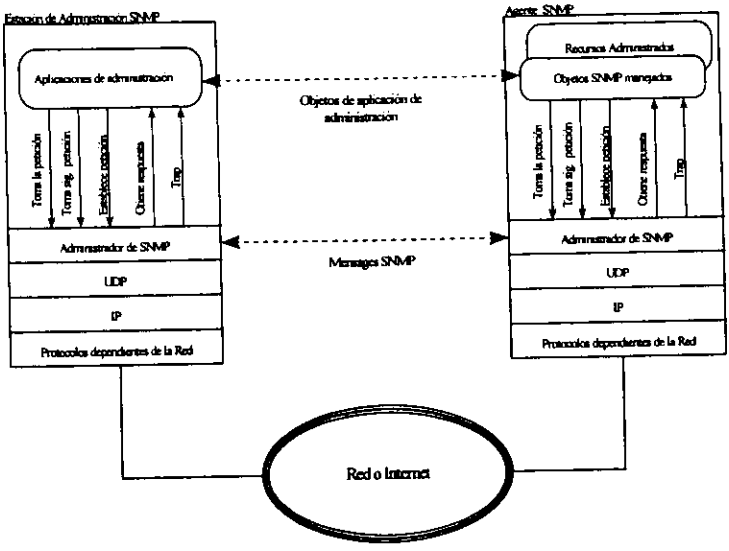


Figura 4.5 Esquema de intercambio de paquetes SNMP

4.2.5 Definición de relaciones administrativas

La administración de red involucra la interacción de varias entidades de aplicación, soportadas por un protocolo en este caso **SNMP**. Los elementos de la red administrada que se comunican con otros a través de **SNMP** son llamados *entidades de protocolo o entidades **SNMP***. Dichas entidades son: las aplicaciones de administración dentro del nodo administrador y el agente en el nodo administrado.

La interacción entre las entidades **SNMP** se puede definir como una relacion uno a muchos entre el nodo administrador y un conjunto de agentes en varios nodos administrados, varios nodos administradores pueden existir en una configuración, y estos a su vez sostener relaciones con varios conjuntos de nodos administrados. Cada agente controla su propia **MIB**, proporcionando la información requerida y realizando las modificaciones que los nodos administradores requieran. Por otro lado los nodos administradores tienen el derecho de hacer consultas y modificaciones a las **MIB** del conjunto de elementos que ellos administran. Con estas condiciones surge la necesidad de contar con alguna táctica que permita a los agentes protegerse así mismos y a sus **MIBs** de accesos no autorizados. De este hecho nace el concepto de *comunidades **SNMP*** y *nombres de comunidad.*

*Una comunidad **SNMP** es una relación entre un agente **SNMP** y un conjunto arbitrario de nodos administradores, la cual define reglas especificas que hace valida el acceso al agente y a su **MIB***

El concepto de comunidad es local y se define en el agente, este establece una comunidad por cada combinación con un nodo administrador. A cada comunidad le es asignado un nombre llamado *nombre de comunidad* el cual es unico dentro del agente. Como los nombres de comunidad son definidos en el agente, el mismo nombre puede ser usado por diferentes agentes. La definición de los nombres es irrelevante y no indica ninguna similitud entre las diferentes comunidades.

Al ser definidas las comunidades y los nombres de las mismas, el agente puede ejercer control sobre los accesos que los nodos administradores realizan a su **MIB** en tres aspectos:

Autenticación. Es el proceso mediante el cual se asegura que una comunicación entre el agente y el nodo administrador es autentica. En otras palabras todo paquete que es enviado desde un nodo administrador contiene un nombre de comunidad dentro de si, este nombre funciona como una contraseña, el paquete solo es admitido si el nodo administrador emisor reconoce este nombre. Con este servicio podemos excluir de el monitoreo y control sobre elementos de nuestra red a máquinas no deseadas.

Política de acceso. Un agente puede limitar el acceso a su **MIB** a un selecto grupo de nodos administradores. Para que no solo una comunidad pueda tener acceso a una **MIB** dentro de un agente, este puede dar diferentes categorias de acceso a la **MIB**. Esto se logra con la definición de subconjuntos de objetos dentro de una **MIB**, estos subconjuntos de variables pueden ser vistos o no por una comunidad **SNMP**, a cada subconjunto se le conoce como *ventana **MIB SNMP***. Las variables en cada ventana pueden tener dos diferentes *modos de acceso* para una comunidad: SOLO-LECTURA, LECTURA-ESCRITURA. La combinacion de una ventana **MIB** y un modo de acceso se conoce como ***SNMP comunity profile*** o *perfil de comunidad **SNMP***. Un perfil de comunidad **SNMP** es asociado con cada comunidad definida por un agente, a esta combinación se le conoce como política de acceso a una **MIB**.

*Servicio de **Proxies**.* El concepto de comunidad es también utilizable cuando se cuenta con proxies o delegados, recordemos que un **proxy** es un agente que actua como intermediario con otros dispositivos, comúnmente estos dispositivos son extraños es decir no soportan **TCP/IP** y por lo tanto tampoco soportan **SNMP.**

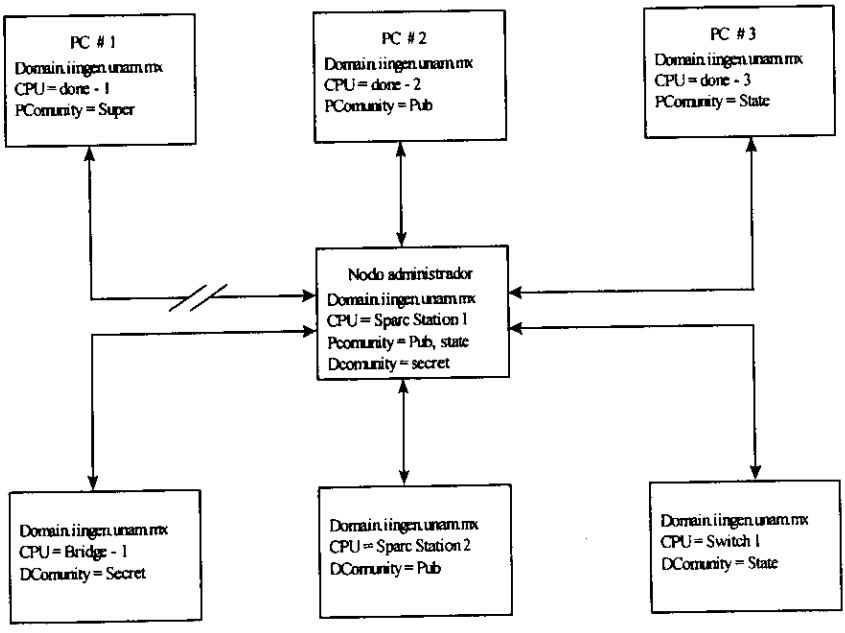


Figura 4.6 Ejemplo de una configuracion de administracion de red con el uso de comunidades y nombres de comunidades

4.2.6 Especificaciones del protocolo

La comunicación entre las entidades dentro de un esquema de administración de red, es realizada por el intercambio de mensajes **SNMP**, cada uno de los cuales es independientemente representado dentro de un solo datagrama **UDP** usuando las reglas básicas de codificación de **ASN.1**. Cada mensaje incluye un número de version, indicando la versión de **SNMP**, un nombre de comunidad, y uno de los cinco tipos de unidades de datos del protocolo(**PDU protocol data units**) los cuales le dan carácter a un mensaje, es decir si un **PDU** es un **Get-Request**, entonces el mensaje será un mensaje **Get-Request**. Dependiendo de cual sea el tipo de **PDU** en un mensaje **SNMP** se sabra que operación será ejercida sobre la información administrativa. En la siguiente figura se muestra el formato general de un mensaje SNMP.

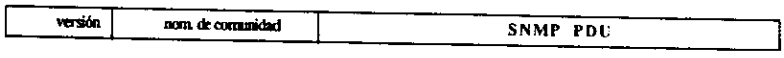


Figura 4.7 Formato general de un mensaje snmp

Existen varios pasos envueltos en la transmisión de un mensaje **SNMP** que una entidad debe realizar:

1. Dependiendo del tipo de operación que se ejercerá (**Get-Request, Get-Response, Get-Next-Request**, etc.) sobre la información administrativa, será asignado y construido un **PDU**
2. El protocolo entonces construye un mensaje, con numero de versión, el nombre de comunidad, e incluye el **PDU**.
3. El mensaje resultante es codificado, usando reglas de codificación especificas (en este caso **ASN.1**) entonces es pasado al servicio de transporte.

De la misma manera cuando un mensaje es recibido por una entidad **SNMP**, existen varias acciones que deben ser realizadas:

1. Una verificación básica de la sintaxis se lleva a cabo, descartando el mensaje si se encuentra un error en la verificación.
2. Una verificación del número de versión, si las versiones son diferentes, entonces el mensaje se descarta.
3. Se toma el nombre de comunidad, el **PDU**, las direcciones fuente y destino y son pasadas al servicio de autenticación de la máquina receptora.

    a) Si la autenticación falla, el mensaje es automaticamente descartado.
    b) Si la autenticación tiene éxito, entonces el servicio de autenticación regresa un **PDU** decodificado.

4. El protocolo realiza una verificación de sintaxis del **PDU**, si la sintaxis falla lo descarta.

Tipos de PDU

PDU Get-Request . El siguiente esquema muestra el formato de un **PDU Get-Request**.

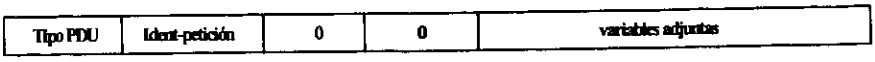


Figura 4.8 PDU Get-Request

Podemos observar en el esquema los siguientes campos:

- *Tipo de PDU* : Indicando que este es un **PDU Get-Request.**
- *identificador de petición (ident-petición):* La entidad administradora **SNMP** asigna números con los cuales cada petición a un mismo agente es identificada. Además el identificador de petición habilita a la entidad administradora para relacionar las peticiones a un agente con sus respectivas respuestas.
- *Variables adjuntas:* Recordemos que solo valores escalares pueden ser leidos en cada petición a una **MIB** (sea un objeto o una tabla), sin embargo se puede en una sola petición obtener un conjunto de valores. En todos los tipos de **PDU** existe el campo de variables adjuntas, en el se define una lista de variables de las cuales se requiere su contenido, no necesariamente tiene que ser una lista también puede ser una sola variable.

PDU Get-Response: La entidad receptora de un mensaje **Get-Request**, respondera a este con un mensaje **Get-Response**. El formato del **PDU Get-Response** es el mismo que para el PDU **Get-Request**, excepto por la inclusión de dos campos:

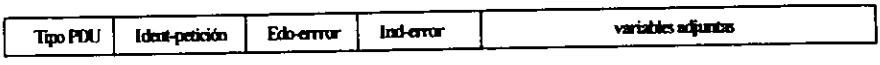


Figura 4.9 PDU Get-Response

Los campos *estado del error (edo-error)* e *indice del error (ind-error)* existen si hubo un error en la lectura o escritura de la o las variables solicitadas en el campo de variables adjuntas del **PDU Get-Request**. En el campo edo-error encontraremos el estado del error, es decir que causo que la variable o variables no pudiesen ser leidas, por ejemplo cuando el identificador de un objeto no existe el campo edo-error contendrá el mensaje *NoNombre*, si una variable no pudo ser leida por exceder el tamaño permitido, entonces el campo edo-error contendra el mensaje *DemasiadoGrande*. Ahora bien en el campo índice de error se encontrara el identificador final de la variable que causo el problema, por ejemplo si se desea obtener el valor de la variable 1.3.6.1.2.1.6.4.1 y por alguna razon no es posible accederla entonces el campo ind-error contendrá el valor 1.

En el campo identificador de petición este **PDU** tendrá el mismo valor que el correspondiente **PDU Get-Request**, si en el mensaje de petición esta definida una lista de variables adjuntas y la entidad reseptora esta en condición de ofrecer una respuesta a esta lista, entonces en el campo de variables adjuntas se encontraran los valores requeridos por la entidad transmisora.

PDU Get-Next-Request: El formato de un **PDU Get-Next-Request** es también idéntico al del **PDU Get-Request**. La unica diferencia es la siguiente: En el **PDU Get-Response** asociado a un **PDU Get-Next-Request** , tendrá en el campo de variables adjuntas el valor del objeto que es el siguiente en el orden, esto es, un valor miembro de una tabla.

PDU Set-Request: Nuevamente, el formato de un **PDU Set-Request** es el mismo al del **PDU Get-Request**. La diferencia es que un mensaje definido por un **PDU Set-Request** es usuado para escribir un objeto no solo para leerlo. Además el contenido de el campo variables adjuntas en este **PDU** incluye el identificador del objeto como el valor que va a ser asignado al mismo.
La entidad receptora de un mensaje **Set-Request** responde con un mensaje **Get-Response** si esta habilitada para realizar la escritura que se le ha pedido. En el mensaje de respuesta el **PDU Get-Response** contendrá el mismo valor del campo identificador de petición, y en el campo variables adjuntas contendrá el valor que ha sido escrito en el objeto dentro de la **MIB.**

PDU trap: El formato de este **PDU** es diferente a los descritos anteriormente:

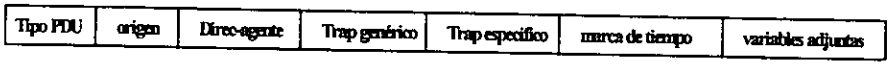


figura 4.10 Formato del PDU Trap

- *Tipo de **PDU*** : Indica que tipo de PDU es. Para este caso es **PDU Trap**
- *Origen:* Identifica a el nodo administrador que envió el **trap.**
- *Direccion del agente (direc-agente)*: Especifica la dirección **IP** del objeto generador del **trap**
- *trap-genérico:* Especifica un tipo predefinido de **trap.**
- *trap-especifico:* Un codigo que indica mas especificamente la nautaraleza del **trap**
- *marca de tiempo:* Es el tiempo comprendido desde que se inicializo la entidad y el tiempo que el **trap** ocurrio.
- *Variables adjuntas:* Información adicional relacionada con el **trap**

Los **traps** genéricos pueden ser:

1. ***ColdStart (0):*** Un trap de este tipo es generado cuando el objeto administrado se ha reinizializado por el mismo, debido a fallas importantes, gracias a esta reinicialización la configuración de la entidad **SNMP** en el objeto administrado pudo haber sido alterada.
2. ***warmStrat (1):*** Reinicialización debida a una falla grave, pero sin que la configuración de la entidad **SNMP** fuese alterada.
3. ***LinkDown (2):*** Señales de falla en la liga de comunicación con un agente dentro de un nodo administrado.
4. ***LinkUp (3):*** Mensaje que indica que una liga perdida con un agente ha sido restablecida.
5. ***Authentication failure (4):*** Mensaje que indica que la entidad transmisora del **trap** ha recivido un mensaje que no es autentico.
6. ***enterpriseSpecific (6):*** Señal que especifica que un evento especial especifico dentro de el nodo administrado ha ocurrido.

A diferencia de los otros **PDU** un mensaje **trap** no tiene como respuesta algún mensaje.

En la siguiente figura se muestra como los mensajes con diferentes **PDU** son contestados.

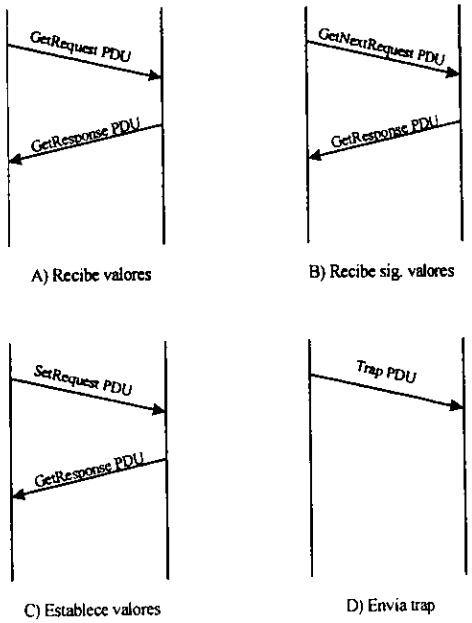


Figura 4.11 Los diferentes PDU y los correspondientes mensajes de respuesta.

Como mencionamos anteriormente, **SNMP** por si mismo no provee reconocimiento de paquetes ya que esta basado en **UDP**, esta inconveniente es manejado de una manera muy simple: En el caso de un mensaje **Get-Request**, y **Get-Next-Request**, el mensaje se da por perdido si la máquina receptora no recibe en un determinado período de tiempo una respuesta para su petición, entonces el nodo administrador puede enviar un o más veces su petición, después de un cierto período de tiempo y si no tiene respuesta del agente, el nodo administrador puede suponer que el nodo administrado en question puede estar abajo o puede tener problemas graves. En el caso de un **Set-Request**, si no se recibe un mensaje de respuesta confirmando que la acción pedida ha tenido lugar, entonces el nodo administrador envía un mensaje **Get-Request** con el fin de probar que la operación de escritura fue realizada, solo en el caso de encontrar que la escritura no fue llevada a cabo se vuelve a enviar el mensaje. Para el caso de los **Traps** es más difícil determinar si los **traps** fueron entregados. Dentro de un arquitectura **SNMP** un **trap** debe ser usado para obtener una señal temprana de alarma de un evento significativo; como respaldo, el nodo administrador deberá también realizar un **polling** periódicamente a los nodos administrados.

4.2.7 RMON (Remote Network Monitoring)

SNMP, es actualmente ampliamente utilizado, casi todos los fabricantes de computadoras personales, estaciones de trabajo, dispositivos de interconexión de red y dispositivos contra fallas eléctricas (**no-breaks**), incluyen en sus dispositivos a **SNMP**. Debido a lo anterior muchos desarrollos han surgido en base a las **MIBs** básicas de **SNMP**, el más importante de estos es sin duda **RMON**.

RMON es el mayor desarrollo creado para la administración de grandes redes, compuestas por varias subredes. **RMON** define una **MIB** de monitoreo remoto, que suplementa a la **MIBs** de SNMP para crear una sólida administración.

Para entender la filosofía de **RMON**, debemos primero explicar el concepto de monitoreo remoto para este contexto:

Con las **MIBs** básicas de **SNMP**, el nodo administrador puede obtener información individual de cada dispositivo. Consideremos una red, con un número determinado de subredes, conformadas por varios dispositivos, cada uno con un agente **SNMP**. Un nodo administrador SNMP puede tomar información acerca del trafico fuera de cada dispositivo, pero debido a las definiciones de las **MIBs** involucradas en este esquema, difícilmente el nodo administrador podrá tener información concisa y rápida del total de tráfico en cada subred. Debido a esta situación es necesaria la utilización de un dispositivo que se dedique a observar el trafico en la subred. Para propósitos de la administración de redes grandes, comúnmente se necesita un dispositivo observador por subred. El propósito de este dispositivo observador es el de capturar y analizar el trafico en la subred, este observador puede ser una estación de trabajo, una PC, o un enrutador. Estos dispositivos observadores son conocidos como *monitores remotos*. Los monitores remotos necesitan comunicarse con el o los nodos administradores y es aquí donde entra **RMON**.

*La especificación **RMON** es principalmente una definición de una **MIB**. Dicho más ampliamente **RMON** define funciones de monitoreo de red estándar, para la comunicación entre nodos administradores **SNMP** y monitores remotos.*

En términos generales, **RMON** provee un eficiente camino para monitorear el comportamiento de las subredes dentro de una red, reduciendo la carga en los demás agentes y los nodos administradores.

La MIB RMON

El grueso de la especificación **RMON** es dedicado a la definición de su base de datos de información administrativa (**MIB**).

La **MIB RMON** esta dividida en nueve grupos:

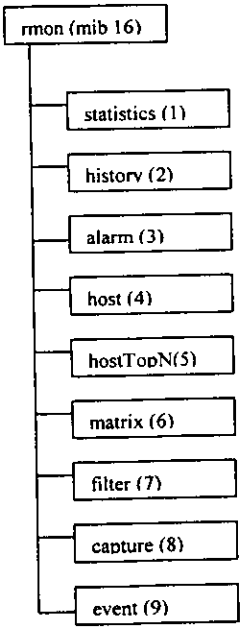


Figura 4.12 MIB RMON

1. *statistics(1).* Contiene las estadísticas de error y utilización, de cada subred monitoreada.
2. *history(2).* Guarda muestras, de las estadísticas del grupo 1.
3. *alarm(3).* Permite especificar valores específicos y umbral para definir alarmas.
4. *host(4).* Contiene contadores de trafico para los **hosts** conectados a la subred.
5. *hostTopN(5).* Lista de **hosts** con trafico mayor
6. *matrix(6).* Muestra información de error y utilización en forma de matriz.
7. *filter(7).* Permite al monitor observar paquetes de un filtro. El monitor puede capturar todos los paquetes o solamente realizar estadísticas con ellos.
8. *packet capture(8).* Indica como serán pasados los datos al nodo administrador.
9. *event(9).* Contiene todos los eventos generados por el monitor los dispositivos.

Cada grupo es utilizado para almacenar la información colectada por el monitor remoto. Un monitor remoto puede tener conectada una o más interfaces de red, por lo tanto puede estar conectado a uno a mas subredes. La información almacenada en cada grupo puede representar datos obtenidos de una o más subredes, dependiendo de como el monitor remoto esta configurado.

Todos los grupos de la **MIB RMON**, son opcionales, sin embargo existen algunas dependencias entre ellos:

- El grupo alarm requiere la existencia del grupo events
- El grupo hostTopN requiere la presencia del grupo host

La implantación de un monitoreo que incluya **RMON** es tan rica en paquetes administrativos, que se corre un riesgo real de sobrecargar la subred entre el monitor remoto y el nodo administrador, el monitor remoto, o el nodo administrador. Una solución ampliamente usada, es la que permite que el monitor remoto realice casi todos los análisis de la información colectada y solo pase los resultados al nodo monitor.

Puede ser poco practico incluir un agente **SNMP** en todos y cada uno de los elementos de la red, en este caso **RMON** da una buena opción al incorporar un nodo monitor en cada subred, de esta manera ofrece una visión global de la misma y no se gasta recursos al incluir un agente **SNMP** por elemento.

Cualquier **hardware**, puede usar **RMON**, por supuesto este hardware debe soportar **SNMP**. El nodo monitor puede ser una **PC** o estación de trabajo dedicada a las funciones **RMON**, pero también se puede utilizar un dispositivo no dedicado, como es el caso de los dispositivos de interconexión, por ejemplo un puente o un enrutador.

4.3 SNMP Protocolos de seguridad.

SNMP proporciona multiples ventajas referentes a su fácil implementación y su gran funcionalidad, pero desafortunadamente no provee elementos de seguridad. **SNMP** no es capaz de autentificar la fuente de un mensaje de administración, el uso de un nombre de comunidad puede no proveer seguridad ya que un agresor puede observar un mensaje y averiguar dicho nombre y usarlo para su beneficio.

En 1992 fueron desarrollados varios protocolos llamados *protocolos de seguridad* ***SNMP*** . Estos protocolos en su conjunto resuelven los problemas de seguridad que presenta **SNMP**. Los protocolos de seguridad no son totalmente compatibles con **SNMP**, los formatos de los encabezados de los mensajes son diferentes y muchos de los procedimientos que SNMP aplica fueron modificados, sin embargo el formato de los PDU es el mismo.

4.3.1 Servicios de seguridad que proveen los protocolos de seguridad SNMP.

En general los protocolos de seguridad proveen los siguientes servicios que incrementan la seguridad en un esquema de administración y monitoreo de red.

- *Integridad en los datos.* Asegura que todo mensaje sea recibido o enviado, sin duplicación, interrupción, intersección o modificado.
- *Autenticación del origen de los datos.* Reconoce el origen de cualquier mensaje enviado.
- *Confidencialidad en los datos.* Asegura que la información no este disponible para entidades o procesos no autorizados.

La integridad en los datos y la autenticación del origen de los mismos, son proporcionados por un mismo mecanismo, la confidencialidad en los datos es un servicio opcional que puede ser adicionado a los otros dos servicios en la misma implementación.

4.3.2 Mecanismos de seguridad

Para proveer los servicios de seguridad listados anteriormente, los protocolos de seguridad deberan incluir los siguientes mecanismos.

- Para garantizar la integridad de los datos, un algoritmo de resumen de mensaje es requerido. Dicho algoritmo es usado para calcular un resumen de 128 bits de un porción apropiada del mensaje. Este resumen es incluido como parte del mensaje enviado a la maquina receptora, para asegurar que dicho mensaje no sufrió modificación. El algoritmo de resumen de mensajes que los protocolos de seguridad **SNMP** usan es el **MD5**[1]. Una marca de tiempo es incluida en cada mensaje generado, el valor de la marca de tiempo esta basado en los relojes de sincronización de los nodos administradores y los agentes. Un receptor de mensaje evalua la marca de tiempo para determinar si dicho mensaje es reciente, o si el mensaje esta relacionado con otros que el mismo recipiente ha recibido. En conjunción con otra información disponible en el mensaje (por ejemplo: el identificador de petición), la marca de tiempo también indica si el mensaje es una respuesta de un mensaje previo.
- Para garantizar la integridad de los datos y la autenticación del origen de los mismos, la porción del mensaje que es resumida es primero reconstruida con un valor secreto compartido por la maquina originadora del mensaje y el recipiente.
- Para garantizar la confiabilidad de los datos, un algoritmo simetrico de encriptación es requerido. Una porción apropiada del mensaje es encriptada. Los protocolos de seguridad usan el algoritmo de encriptación **DES**[2].

4.3.3 Modelo administrativo (relaciones administrativas)

Los protocolos de seguridad **SNMP** están basados en un nuevo modelo administrativo que reemplaza el concepto de comunidad. Recordando el modelo administrativo **SNMP** en el cual se definia un nombre de comunidad que podia ser usado por dos o más entidades SNMP como un tipo de contraseña. Para proveer los servicios de seguridad, los protocolos de seguridad realizaron un cambio importante en el modelo tradicional administrativo: *La definición de piezas **SNMP**.*

Desde el punto de vista de la seguridad, cada entidad **SNMP** se comporta de diferente manera, dependiendo de la equivalencia con las otras entidades **SNMP** que estan involucradas en la arquitectura de administración, por otro lado el papel de la entidad SNMP depende del contexto de su operación, así es que cada entidad SNMP puede estar asociada a un nombre dependiendo de su acción. Dicho nombre es conocido como *pieza **SNMP**, la cual esta definida por el contexto de ejecución de una entidad **SNMP**.*

[1] El algoritmo **MD5** fue desarrollado por Ron Rivest en el **MIT** . El algoritmo toma como entrada un mensaje de longitud arbitraria y produce como salida un resumen del mensaje de 128 bits. Para mayor información referirse al RFC 1321

[2] **DES Data Encryption Standar** . Es uno de los esquemas de encriptación más utilizados, el cual utiliza una llave de 56 bits para encriptar datos en bloques de 64 bits

Cada pieza **SNMP** debe ser representada como un objeto en las **MIBS** de cada entidad **SNMP** que participe en la arquitectura de administración.

Cada mensaje transmitido debe identificar tanto a la entidad fuente como a la entidad destino, así es para cualquier mensaje su encabezado deberá tener 2 campos con las piezas **SNMP** asociadas a la entidad transmisora como a la entidad receptora, en vez de un solo campo con un nombre de comunidad.

4.3.4 Especificaciones de los protocolos

Como mencionamos anteriormente, los protocolos de seguridad **SNMP** utilizan dos métodos para proveer sus servicios de seguridad: un algoritmo de resumen de mensajes **MD5** y un esquema de encriptación **DES**. Para que un servicio de seguridad sea brindado, estos métodos deben interactuar con dos diferentes protocolos: *El protocolo de autenticación-resumen y el protocolo simétrico de privacia.*

La conjunción de estos dos protocolos y los respectivos métodos ofrecen los mecanismos que a su vez proporcionan todos los servicios de seguridad mencionados.

Cuando se brinda el servicio de autenticación del origen de los datos, por ende se brinda el servicio de integridad de los mismos. El servicio de confidencialidad puede ser incluido en la implementación. De esta manera una arquitectura de administración de red basada en estos protocolos puede proveer varios niveles de seguridad, es decir los datos pueden ser enviados sin ningún servicio de seguridad o bien solo con el servicio de autenticación (incluyendo el servicio de integridad de los datos), o solo con el servicio de confidencialidad, o finalmente con todos los servicios.

Para los protocolos de seguridad **SNMP**, la información entre nodos administradores y nodos administrados es intercambiada en forma de mensajes de la misma manera que en el esquema **SNMP** tradicional. Cada mensaje incluye un encabezado y uno de los 5 tipos de **PDU**, dichos **PDU** son los mismos que los definidos para **SNMP**.

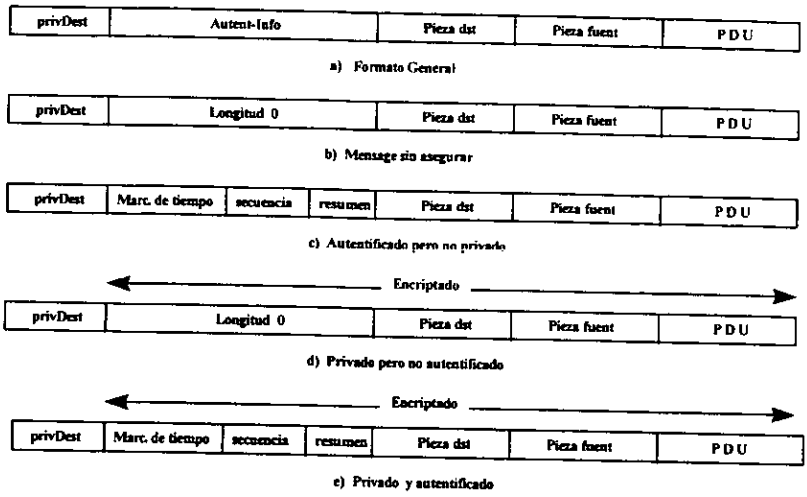


Figura 4.13 Diferentes tipos de mensajes de los protocolos de seguridad SNMP

El encabezado del mensaje consiste en cuatro campos: los campos *Pieza dst* y *Pieza fuent* que contienen los nombres de las piezas **SNMP** para el nodo fuente y destino del mensaje, el campo *Autent-Info* contiene información que concierne al protocolo que se encarga de realizar la autenticación del mensaje, el campo *privDest* contienen también el nombre de la pieza **SNMP** del nodo destino.

La figura 4.13 también provee encabezados de diferentes mensajes dependiendo del contexto de seguridad en el que son transmitidos. Si el mensaje no es seguro (sin ningún servicio de seguridad), entonces el campo Auten-Info consiste en una cadena de codificada en **ASN.1** de longitud 0 este encabezado se muestra en la figura 4.13 b). Si el mensaje es autentificado pero no tiene el servicio de confidencialidad, entonces el campo Auten-Info consiste en tres subcampos, como se muestra en la figura 4.13 c), dichos subcampos son: *Marca de tiempo*, el cual representa el tiempo de generación de este mensaje, *Secuencia* el cual representa un número de secuencia cuando un mensaje esta relacionado con otros mensajes transmitidos y todos ellos forman parte de una secuencia, *Resumen,* contiene el resumen realizado sobre un porción del mensaje.

Las figuras d) y e) de la figura 4.13 muestran el formato del mensaje cuando el servicio de confidencialidad es provisto, en este caso el mensaje (incluyendo el encabezado y el **PDU**) con excepción del campo privDest, es encriptado. El campo privDest no es encriptado con la finalidad que la entidad receptora pueda reconocer la pieza **SNMP** fuente del mensaje.

Transmisión de un mensaje

En el siguiente diagrama de flujo se muestran los pasos que una entidad transmisora realiza antes de enviar un mensaje

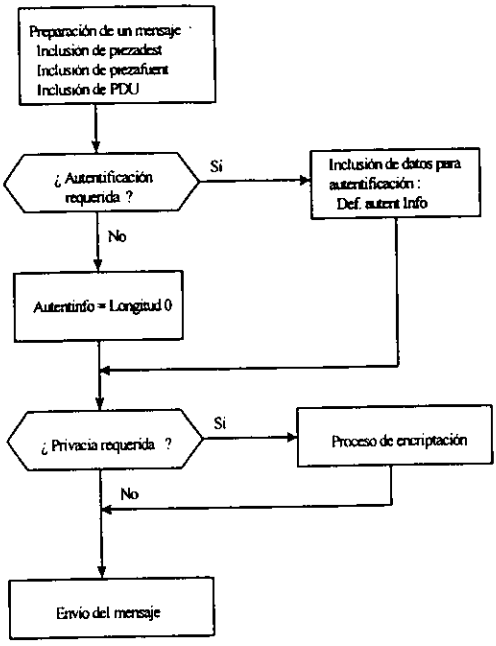


Figura 4.14 Diagrama genérico de el proceso de transmisión de un mensaje

En el siguiente diagrama se muestran los pasos generales que una entidad receptora realiza cuando toma un mensaje.

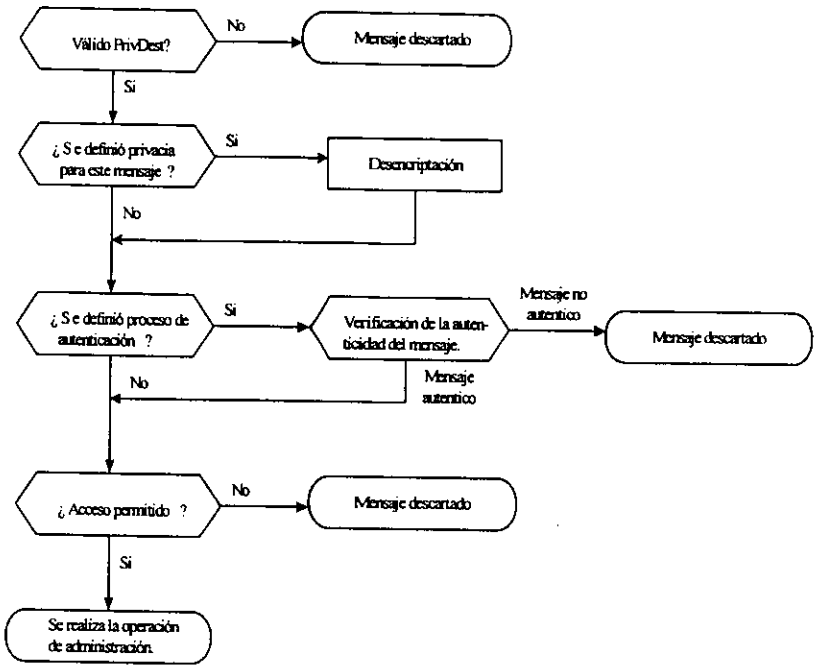


figura 4.16 Diagrama genérico de recepción de un mensaje por una entidad SNMP

Protocolos de seguridad

Protocolo de autenticación - resumen. Este protocolo junto con el algoritmo **MD5** proporciona el mecanismo para la autenticación del origen e integridad de los datos. En esencia el proceso de autenticación es el siguiente: Un resumen de mensaje es ejercido sobre el mensaje que será enviado, usando el algoritmo **MD5**. Dicho mensaje mas el resumen es transmitido, cuando el mensaje llega a la entidad receptora, de nueva cuenta se realiza un resumen del mensaje recibido y si este resumen es igual al resumen que esta incluido en el mensaje, entonces el mensaje recibido es declarado autentico e integro.

Protocolo simétrico de privacía. Este protocolo provee protección a los datos de tal forma que solo la entidad fuente y la entidad destino puedan leer el mensaje. El método para proveer dicha protección es la encriptación, la cual requiere que el destino y la fuente compartan la misma llave de encriptación. El algoritmo usado para la encriptación es **DES**.

4.4 SNMP versión 2 (SNMPv2)

En 1988, se supo que la administración de red era una necesidad critica, entonces se desarrollo el protocolo simple de administración de red **SNMP**, el cual resolvió varios problemas. **SNMP** provee un adecuado servicio a muchas configuraciones de red, pero su mayor deficiencia es que carece de métodos para proveer seguridad, los protocolos de seguridad **SNMP** solucionan estos problemas, pero dichos protocolos tienen otras desventajas relacionadas con el desempeño y funcionalidad. A finales de 1992 un grupo de desarrolladores se reunió para realizar una adecuada actualización a **SNMP**, esta actualización esta basada en los desarrollos anteriores (**SNMP** y los protocolos de seguridad **SNMP**) y generó un nuevo protocolo: **SNMPv2**, la versión dos de **SNMP**. **SNMPv2** provee la alta funcionalidad de **SNMP** y adopta los desarrollos de seguridad que los protocolos de seguridad **SNMP** alcanzaron.

Los desarrollos y modificaciones que **SNMPv2** trae consigo entran en las siguientes categorias:

- Estructura de la información de administración **SMI**
- Operación del protocolo
- Comunicación entre nodos adminstradores
- Seguridad.

4.4.1 La arquitectura SNMPv2

La arquitectura para **SNMPv2** es básicamente la misma que para **SNMPv1**, implícita en ella se encuentran nodos administrados, nodos administradores y el protocolo de administración que se encarga de comunicar la información administrativa a través de dichos nodos.

SNMPv2 esta bien capacitado para adoptar cualquiera de las arquitecturas de sistemas de administración de red que existen (centralizada, distribuida o mixta), pero incluye una nueva y poderosa capacidad que fortalece los sistemas distribuidos: La comunicación entre nodos administradores, más tarde hablaremos con detalle de este hecho.

El concepto de *entidades de protocolo o entidades SNMPv2* se sigue aplicando a los elementos de la red administrada que se comunican a través del protocolo **SNMPv2**.
Una entidad **SNMPv2** puede operar en el papel de administrador o en el papel de agente. Una entidad **SNMPv2** actúa en el papel de agente cuando esta realiza operaciones de administración de red en respuesta a mensajes **SNMPv2** (diferentes de mensajes de notificación) o bien cuando este envía mensajes de notificación o otras entidades. Una entidad **SNMPv2** actúa en un papel de administrador cuando genera mensajes que están destinados a otros nodos con el fin de ejercer sobre ellos tareas de administración, o bien cuando estas entidades realizan tareas de administración en respuesta a mensajes de notificación. Una entidad **SNMPv2** puede actuar en los dos papeles, dependiendo de su configuración e implementación. Las entidades **SNMP** pueden también actuar como **proxies.**

4.4.2 Estructura de la información administrativa (SMI)

La estructura de la información administrada **SMI** para **SNMPv2** esta basada en la **SMI** para **SNMPv1**[3] .**SMI** para **SNMPv2** provee especificaciones más elaboradas para los objetos administrados y **MIBs.**

La **SMI** para la versión dos de **SNMP** puede dividirse en :

- Definición de objetos
- Tablas
- Definición de notificaciones
- Modulos de información

Definición de objetos. Como en **SMI** para **SNMP** la definición de objetos es usada para describir objetos administrados. El **OBJECT-TYPE** de **ASN.1** da la sintaxis y semantica para representar a todos los objetos administrados.
En esta **SMI** varios tipos fueron modificados o incluidos: Hay restricción para los enteros de 32 bits, se incrementaron valores de cadenas de bits enumeradas, se incluyo un contador de 64 bits ademas del tradicional de 32 bits, también se incluyo el tipo **Uinteger** para representar enteros del rango 0 a 2 a la 32 menos 1.

Tablas. Dentro de **SNMPv2** la información más compleja puede ser representada como una tabla.

[3] Con el propósito de esta exposición llamaremos al desarrollo **SNMP** como **SNMPv1**

Esencialmente existen dos categorias de tablas permitidas en **SNMPv2**:

1. Tablas que prohiben la creación de renglones y borrado por un administrador. Estas tabals son controladas completamente por el agente. En muchos casos la tabla completa consiste solo de objetos con permiso solo de lectura.
2. Tablas que permiten la creación y borrado por un administrador. Estas tablas pueden ser inciadas con 0 renglones y un administrador se encargará de su posterior llenado. O bien un administrador y el propio agente pueden encargarse de la construcción de la tabla.

Definición de notificaciones. Estas definiciones son usadas para describir la información enviada por una entidad **SNMPv2** cuando un evento excepcional ocurre en dicha entidad. La macro **NOTIFICATION-TYPE** de **ASN.1** es usada para definir las notificaciones.

Módulos de información. **SNMPv2** incluye el concepto de módulo de información, el cual especifica un grupo de definiciones (objetos administrados) relacionadas. La macro **MODULE-IDENTITY** de **ASN.1** es usada para definir módulos.

4.4.3 SNMPv2 : Base de datos de administración (MIB)

La **MIB** de **SNMPv2** define objetos que describen el comportamiento de una entidad **SNMPv2**. Esta **MIB** consiste en 5 grupos:

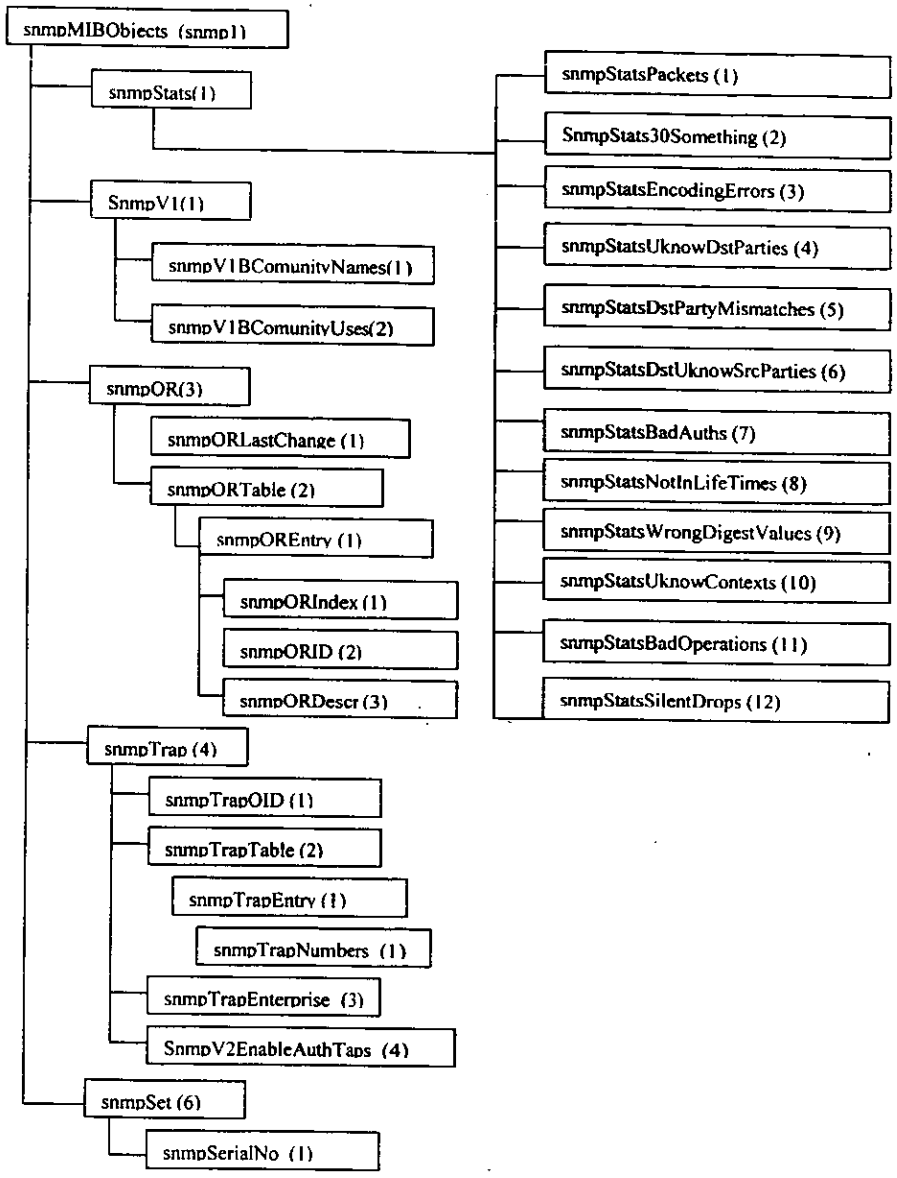


Figura 4.16 a MIB para SNMPv2

1. *Grupo estadístico SNMPv2 (snmpStats(1)):* Los objetos en este grupo representan información básica relacionada con la operación del protocolo **SNMPv2.**

2. *Grupo estadístico SNMPv1 (snmpV1(2)):* **SNMPv2** provee la facilidad de interactuar con entidades que implementen a **SNMP** como protocolo de administración. Los objetos en este grupo proveen información básica de trafico relacionado con la operación de **SNMP** en una entidad **SNMPv2** que también implemente **SNMP**. Esencialmente este grupo registra el numero de veces que los mensajes **SNMP** son rechazados usando nombres de comunidades.

3. *Grupo de objetos asociados recursos (snmpOR (3)):* Este grupo es utilizado por una entidad **SNMPv2** actuado como un agente, para describir objetos que esta puede controlar y que estan sujetos a la configuración y control de un nodo administrador.

4. *Grupo de notificaciones (snmpTrap (4)):* Una colección de objetos que permiten a una entidad **SNMPv2**, cuando actua como agente, ser configurada para generar **SNMPv2 traps.**

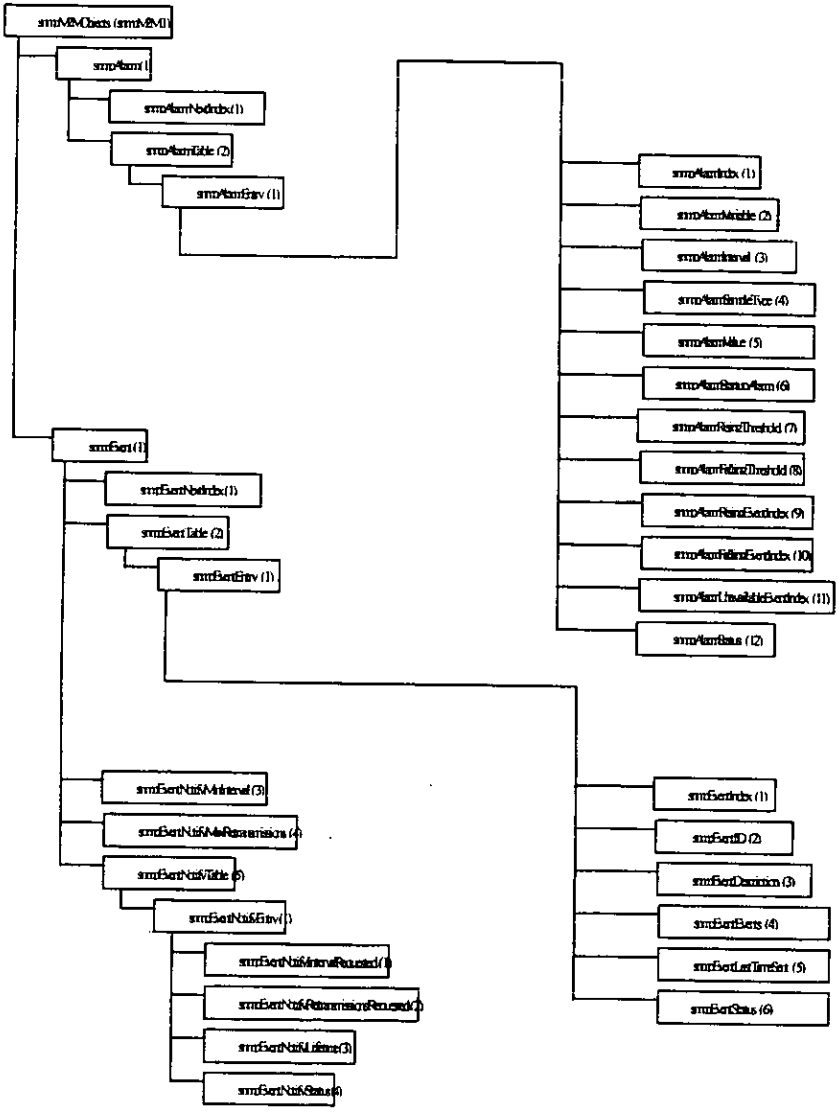
5. *Grupo de control (snmpSet (6)):* En este grupo se encuentra un solo objeto, que a través de el muchos nodos administradores pueden realizar acciones de control (definición y modificación) sobre esta **MIB**

MIB Administrador - Administrador

La **MIB** administrador - administrador para **SNMPv2** consiste de un conjunto de objetos que describen el comportamiento de una entidad **SNMPv2** que actúa como en el papel de administrador. La utilización de esta **MIB**, habilita un servicio muy importante que es uno de los desarrollos más significativos que provee la versión 2 de **SNMP**: la comunicación entre dos nodos administradores. Esta **MIB** consiste de dos grupos:

1. *Grupo de alarma:* Una colección de objetos que permiten la descripción y configuración de alarmas para una entidad actuando en ambos papeles.
2. *Grupo de eventos:* Una colección de objetos que permiten la descripción y configuración de eventos para una entidad **SNMPv2** actuando e ambos papeles.

En la siguiente figura se ilustra la estructura de dicha **MIB**



4.17b Estructura de la MIB administrador - administrador

4.4.4 Acceso a la información administrativa

Tres tipos de acceso a la información administrada son proporcionados por **SNMPv2**:

- *Nodo administrador - agente, petición - respuesta:* Cuando una entidad **SNMPv2** actúa en el papel de administrador, envía una petición a una entidad **SNMPv2** actuando en el papel de agente, entonces la entidad que actúa como agente responde a la petición. Este tipo de acceso es usado para recuperar o modificar información administrativa asociada con un nodo administrado.
- *Nodo administrador - nodo administrador, petición - respuesta:* Una entidad **SNMPv2** actuando como administrador envia una petición a una entidad **SNMPv2** actuando también como administrador, esta ultima entidad responde a la petición. Este tipo de acceso es usado para que una entidad administradora pueda conocer la información administrativa de otra entidad actuando también como administrador.
- *Agente - Nodo administrador, interacción no confirmada.* Una entidad **SNMPv2** actuando como agente envía un mensaje no solicitado llamado **trap** a una entidad **SNMPv2** actuando como administrador, ningún mensaje de respuesta es enviado por la entidad adminstradora. Este tipo de acceso es usuado para notificar un evento excepcional a una entidad administradora por una entidad agente.

Para llevar a cabo los accesos **SNMPv2** maneja los siguientes mensajes:

1. ***Get - Request***
2. ***Get - Next - Request***
3. ***Get - Bulk - Request***
4. ***Set - Request***
5. ***Response***
6. ***SNMPv2 - Trap***
7. ***Inform - Requets***

Todas las entidades actuando como agente están habilitadas para generar los siguientes mensajes: **SNMPv2 - Trap** y **Response**. Estas entidades están habilitadas para recibir los siguientes mensajes: **Get - Request, Get - Next - Request, Get - Bulk - Request** y **Set - Request**.
Todas las entidades actuando como administradores están habilitadas para generar los siguientes mensajes: **Get - Request, Get - Next -Request, Get - Bulk - Request, Set - Request, Inform - Request** y **Response**. Estas entidades estan habilitadas para recibir : **Response, SNMPv2 -Trap** y **Inform - Request.**

4.4.5 Especificaciones del protocolo

La información administrativa que maneja **SNMPv2** es transportada dentro de mensajes, el formato general de un mensaje **SNMPv2** es el siguiente:

figura 4.17 Formato general de un mensaje SNMPv2

El formato general de los mensajes **SNMPv2** es similar al formato del mensaje de los protocolos de seguridad **SNMP**, los primeros cinco campos del mensaje conforman el encabezado del mismo, y el ultimo campo es el que representa al **PDU**. Por el momento nos enfocaremos en la explicación de los **PDU.**
SNMPv2 es una extensión de **SNMPv1**, como en **SMNPv1** los **PDU** de **SNMPv2** son encapsulados en un mensaje, el carácter de dicho mensaje esta dado por el tipo de **PDU** que este contenga. Existen 7 tipos de **PDU** definidos por **SNMPv2**

Tipos de PDU

PDU Get-Request. El siguiente diagrama muestra el formato de un **PDU Get - Request**

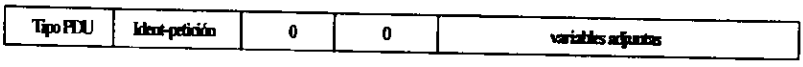


figura 4.18 PDU Get - Request

El **PDU Get - Request** contiene los siguentes campos:

- *Tipo de PDU.* Indica el tipo de **PDU**
- *Identificador de petición (ident-petición):* El identificador de petición es un número con el cual cada petición a un mismo agente es identificada.
- *Variables adjuntas.* Lista de variables que la entidad generadora del mensaje desea saber.

Un **PDU Get - Request** es generado y transmitido por un nodo administrador cuando hace una petición. El **PDU Get - Request** es identico en forma y semantica al **PDU Get-Request** de **SNMPv1**, la única diferencia se encuentra en la forma en que son manejados los mensajes de respuesta a esta petición. En **SNMPv1**, si una o más variables dentro de un mensaje **Get - Request** no son soportadas por la entidad receptora, entonces esta ultima entidad genera un mensaje de repuesta con una misiva de error a la entidad generadora de la petición. En contraste con **SNMPv1** en **SNMPv2** una lista de variables siempre es preparada y entregada por una entidad receptora auque esta no soporte una o más variables de la lista. El hecho de que se permita una respuesta parcial a una lista requerida en una mensaje **Get - Request** es una importante mejora que **SNMPv2** introduce.

PDU Get-Next-Request. El formato de un **PDU Get - Next - Request** es el mismo que el de un **PDU Get - Request** y también es identico en forma y semantica al **PDU Get-Next-Request** de **SNMPv1**. El **PDU Get - Next - Request** es también utilizado en **SNMPv2** por la entidad administradora para recuperar valores de una tabla en una entidad agente. El **PDU Get-Next- Request** puede ser contestado también con una respuesta parcial como en el caso del **PDU Get - Request**.

PDU Get-Bulk-Request. Un **PDU Get-Bulk-Request** es generado y transmitido por una entidad actuando en un papel de administrador. El proposito de dicho **PDU** es que la entidad administradora realice una petición de una gran cantidad de datos, en otras palabras, la entidad administradora puede tener una eficiente y rápida recuperación de tablas muy grandes. Con este tipo de **PDU** se minimiza el número de paquetes intercambiados por el protocolo requeridos para recuperar una gran cantidad de información administrativa. En el siguiente diagrama se muestra el formato de un **PDU Get-Bulk-Request**

figura 4.19 Formato de un PDU Get-Bulk-request

Existe un mapeo uno a uno entre las variables contenidas en la lista de los mensajes petición (**Get-Request, Get-Next-Request, Set-Request PDUs)** con las variables en la lista del mensaje respuesta (**Response PDU**), es decir para cada variable pedida será asignado un valor (este definida o no dicha variable en la entidad receptora) en el mensaje de respuesta. El **PDU Get-Bulk-Request** rompe con esta regla, ya que puede generar cero o más valores de respuesta.
En este **PDU** se adicionan dos campos conocidos como: **non-repeaters** y **max-repetitions**, los cuales son utilizados para calcular el número de datos requeridos. El primer segmento de información del mensaje de respuesta, esta asociado con las primeras N variables adjuntas en la petición y las posteriores M variables son resultado del requerimiento de las R variables adjuntas restantes en la petición, consecuentemente el número total de variables requeridas esta dado por: N + (M * R), donde N es el minimo de variables requeridas en un petición y el valor de **non-repeaters**, y M es el valor de **max-repetitions.**

PDU Set-Request. Un **PDU Set-Request** es generado y transmitido por una entidad administradora para hacer una petición. El formato de este mensaje es el mismo que el del **PDU Get-Request**, además el **PDU Set-Request** es idéntico en forma y semántica al **PDU Set-Request** de **SNMPv1**. Este mensaje es utilizado por la entidad administradora para modificar o definir un objeto en la **MIB** de una entidad actuando como agente.

PDU Response. El **PDU Response** es generado y transmitido por una entidad **SNMPv2** si ha recibido anteriormente un mensaje **Get-Request, Get-Next-Request, Get-Bulk-Request, Set-Request** o **InformRequest**. El formato del **PDU response** es el siguiente:

figura 4.20 Formato del PDU Response

El **PDU Response** de **SNMPv2** es idéntico en forma y semántica al **PDU Response** de **SNMP**, Si los campos *edo-error* e *ind-error* son diferentes de cero, entonces un error se presento en la petición que genero el **PDU Response**. En el campo edo-error se encuentra el tipo de error que causo que las variables requeridas no fuesen leidas o modificadas, el campo ind-error contendrá el identificador de la variable o variables que causaron el problema.

PDU SNMPv2 Trap. Un **PDU SNMPv2 Trap** es generado y transmitido por una entidad **SNMPv2** actuando en el papel de agente, cuando una situación excepcional ocurre. Este **PDU** realiza el mismo papel que el **PDU Trap** para **SNMP** pero con diferente formato. El formato del **PDU SNMPv2 Trap** es el mismo que el del **PDU Get-Request** para facilitar su recepción por la entidad a quien va dirigida.

PDU Inform-Request. El **PDU Inform-Request** es enviado por una entidad **SNMPv2** actuando como administrador a otra entidad **SNMPv2** actuando en el mismo papel. Este tipo de **PDU** es utilizado para comunicar información administrativa entre dos entidades administradoras. El formato de este **PDU** es el mismo que el **PDU Get-Request.**
Cuando un **PDU Inform-Request** es recibido, la entidad receptora valida la petición y envía la respuesta en un **PDU Response.**

SNMPv2 necesita también un potocolo de transporte para el envio de sus paquetes. Los protocolos en los que puede basarse para hacer la entrega de paquetes son los siguientes:

- **UDP**
- El protocolo de transporte del modelo **OSI**
- **IPX**
- **Appletalk**

4.4.6 SNMPv2: Seguridad

Las herramientas con las que cuenta **SNMPv2** para proveer seguridad en el manejo de información administrativa son un desarrollo importante que ha alcanzado este protocolo. Para entender en que se basa y como funciona el esquema de seguridad en **SNMPv2** deberemos primeramente entender el modelo administrativo **SNMPv2.**

4.4.6.1 Modelo administrativo.

El modelo administrativo entre las entidades **SNMPv2** tiene sus orígenes en el modelo administrativo basado en nombres de comunidad de **SNMP** pero se asemeja más al modelo administrativo empleado por los protocolos de seguridad **SNMP**.

El modelo administrativo de **SNMPv2** mejora el modelo basado en un esquema de nombres de comunidad y soporta un modelo de control de acceso más conveniente.

Los elementos de este modelo administrativo serán descritos a continuación :

Pieza SNMPV2

El concepto de pieza **SNMPv2** es el mismo que se aplica al concepto de pieza en los protocolos de seguridad **SNMP**. Recordando este concepto: desde el punto de vista de seguridad, cada entidad puede comportarse de diferente manera dependiendo del papel que le toca desempeñar cuando esta interactuando con otra entidad **SNMPv2**, dicho de otra forma el papel de una entidad **SNMPv2** depende del marco de su operación. Cada papel que desempeña una entidad **SNMPv2** pueden ser representado por una *pieza* y cada pieza puede realizar accesos y acciones a un conjunto de información en la entidad **SNMPv2**, una entidad puede incluir varias piezas. Cuando una entidad **SNMPv2** esta actuando con una pieza particular, la operación de dicha entidad esta restringida a las operaciones que están definidas para esa pieza particular.
Tomemos como definición formal de una pieza **SNMPv2** la siguiente:

*Una pieza **SNMPv2** es un ambiente de ejecución virtual cuya operación esta restringida a un subconjunto de operaciones de una entidad **SNMPv2** en particular.*

Una pieza **SNMPv2** debe comprender:

- Un identificador único para cada pieza.
- Una localización lógica en la red.
- Una porción de **MIB** disponible para la pieza

Ventana MIB

Para establecer la definición de una ventana **MIB**, debemos primero definir que es un *árbol de ventana*. Un árbol de ventana es simplemente un nodo en la estructura jerárquica de la MIB, de dicho nodo nacen varios objetos subordinados.
Una ventana **MIB** es definida como una colección de árboles de ventana. El concepto de ventana **MIB** es importante ya que para cada pieza **SNMPv2** definida debe existir una ventana **MIB** que represente la información que dicha pieza puede manejar.

Contexto SNMPv2

Un contexto **SNMPv2** es una colección de objetos administrados accesible por una entidad **SNMPv2**. Los objetos identificados por un contexto pueden ser accedidos localmente: *contexto local*, o remotamente: ***contexto remoto***.

Un contexto **SNMPv2** local, se refiere a objetos que son accedidos directamente de un administrador a un agente. Este tipo de objetos están representados y contenidos en una ventana **MIB**.
Un contexto **SNMPv2** remoto, se refiere a objetos que son accedidos vía una relación que involucra a un agente actuando como **proxie**.

El contexto es un concepto que relaciona al control de acceso a ventanas **MIB**. Cuando un nodo administrador interactua con un agente para tener acceso a información administrativa dentro de dicho agente, la interacción se lleva a cabo entre la pieza definida para el nodo administrador (pieza administradora) y la pieza definida para el agente (pieza agente) con respecto a un contexto seleccionado; los privilegios de control de acceso se aplican sobre la ventana **MIB** definida para ese contexto. Cuando un nodo administrador hace uso de un agente actuando como un **proxie** para tener acceso a información administrativa relativa a una entidad que no soporte **SNMPv2**, la interacción se lleva acabo entré la pieza del nodo administrador y la pieza del agente que esta actuando como **proxy**. En este caso los privilegios de control de acceso son expresados en terminos de la relación **proxy** - nodo administrador.
Las principales motivaciones para introducir el concepto de contexto son para clarificar las relaciones envueltas en el acceso de información de administración y para minimizar los requerimientos de almacenamiento y procesamiento en un agente.

Control de acceso

Clases de comunicación administrativa SNMPv2

Las clases de comunicación administrativa son las operaciones, definidas en términos de unidades de datos de protocolo **PDUs**, que una entidad tiene permiso para realizar sobre otra entidad.
Una clase de comunicación administrativa corresponde a un tipo especifico de **PDU**. Estas clases están definidas por un entero de **ASN.1**. La lista que relaciona a las clases con los **PDUs**, se presenta a continuación:

Get	1
GetNext	2
Response	4
Set	8
unused	16
GetBulk	32
Inform	64
SNMPv2-Trap	128

Políticas de control de acceso

Las políticas de control de acceso, son restricciones en la admisión de operaciones de administración sobre la información administrativa de una entidad. Dichas políticas están especificadas en términos del contexto y de las clases de comunicación. Una política de acceso tiene cuatro elementos:

1. *Target*: Piezas **SNMPv2** que pueden realizar operaciones de administración sobre su información, que fueron requeridas por otras piezas.
2. *Sujeto*: Piezas **SNMPv2** requiriendo operaciones de administración sobre otras piezas.
3. *Recursos:* La información administrativa sobre la cual la operación requerida se puede realizar, esta información es conocida como el contexto.
4. *Privilegios:* Los privilegios especifican las clases de comunicación que son permitidas.

Para un par dado de piezas target y sujeto, existen multiples políticas de control de acceso, una por cada contexto. El contexto es comunicado por el sujeto al target en el encabezado del mensaje **SNMPv2**. Este desarrollo elimina la necesidad de definir un único par de piezas sujeto/target por cada política de control de acceso, y habilita a una sola pieza target para desempeñar una variedad de contextos por una pieza sujeto dada.

El uso de los elementos del modelo administrativo provee una parte importante del esquema de seguridad del protocolo **SNMPv2**, ya que gracias a la definición de piezas, contexto y políticas de acceso es posible tener control y restricciones sobre la información administrativa en un esquema de administración de red.

La seguridad de un esquema de administración de red no solo esta basada en la aplicación de los elementos de un modelo administrativo, además de esto se necesita servicios de seguridad anexos, los cuales son implementados por mecanismos de seguridad específicos.

4.4.6.2 Objetivos de seguridad que cumple SNMPv2

Los objetivos de seguridad que provee **SNMPv2**, son los siguientes:

- El protocolo debe proveer la certeza que cada mensaje **SNMPv2** recibido no ha sido modificado durante su transmisión a través de la red.
- El protocolo debe proveer la verificación de la identidad de la fuente de cualquier mensaje **SNMPv2**
- El protocolo deberá proveer, cuando sea necesario, que el contenido de cada mensaje no este disponible.

4.4.6.3 Servicios de Seguridad

Los servicios de seguridad necesarios para soportar los objetivos anteriormente mencionados, son los siguientes:

- *Integridad en los datos.* Asegura que todo mensaje sea enviado o recibido, sin duplicación, interrupción, intersección o modificación.
- *Autenticación del origen de los datos.* Corrobora el origen de cualquier mensaje.
- *Confidencialidad en los datos.* Asegura que los datos de un mensaje no esten disponibles a individuos, entidades o procesos no autorizados.

Los protocolos que proveen seguridad a **SNMPv2**, requieren el uso de la integridad en los datos y la autenticación del origen de los mismos todo el tiempo para cumplir con sus objetivos. Para estos protocolos no es posible realizar la integridad de los datos sin la autenticación del origen de los mismos y viceversa.
La confidencialidad es un servicio opcional, pero tampoco es realizable sin la existencia de los dos restantes servicios.

4.4.6.4 Mecanismos de seguridad

Los mecanismos en los que los servicios de seguridad se apoyan para la realización satisfactoria de sus objetivos son los siguientes:

- Para soportar la integridad de los datos, un algoritmo de resumen de mensaje es requerido. Un resumen es calculado sobre una porción apropiada del mensaje, este resumen es incluido como parte del mensaje enviado a la máquina receptora.
- Para soportar la autenticación del origen de los datos y la integridad de los mismos, la porción de mensaje que fue resumido es primero reconstruido con un valor secreto compartido por la entidad origen y la entidad destino.
- Una marca de tiempo es incluida en cada mensaje generado, el valor de la marca de tiempo esta basado en los relojes sincronizados de los nodos administradores y los agentes. Un receptor del mensaje evalúa la marca de tiempo para determinar si dicho mensaje es reciente, o si esta relacionado con otros que el mismo mensaje ha recibido, en conjunción con otra información disponible en el mensaje, la marca de tiempo también indica si el mensaje es una respuesta de un mensaje previo.
- Para soportar al servicio de confidencialidad, un algoritmo simétrico de encriptación es requerido. Una porción del mensaje es encriptada antes de ser transmitido.

De los cuatro mecanismos de seguridad citados anteriormente, dos de ellos necesitan basarse en algoritmos externos al protocolo. El algoritmo de resumen de mensaje que es requerido para soportar la integridad de los datos es **MD5**. Y el algoritmo simétrico de encriptación que es requerido para soportar el servicio de confidencialidad es **DES Data Encryption Strandar**.

4.4.6.5 Protocolos de seguridad

Al igual que en los protocolos de seguridad **SNMP** en el esquema de seguridad **SNMPv2**, se necesita una conjunción entre los mecanismos de seguridad externos y los protocolos de seguridad para proveer los servicios de seguridad que el esquema proporciona.

Los protocolos de seguridad **SNMPv2**, son dos: *Protocolo de autenticación - resumen* y el *protocolo simétrico de privacia.*.

- *Protocolo de autenticación - resumen.* Con este protocolo es posible realizar los servicios de integridad de un mensaje y autenticación del origen de los mismos. La integridad de un mensaje es provista gracias al resumen de una porción apropiada del mensaje mediante el algoritmo **MD5**. El resumen es computado por la entidad origen del mensaje, transmitido junto con el mensaje mismo y verificado por la entidad destino. La autenticación del origen esta implícita en la verificación del resumen por una entidad receptora, ya que un valor secreto conocido solo por la entidad origen y la entidad receptora del mensaje es pre - construido y puesto en el mensaje antes que el resumen se realice, dicho valor secreto es también resumido y verificado posteriormente.

- *Protocolo simétrico de privacia.* Este protocolo implementa junto con **DES**, el servicio de confidencialidad en los datos. Una porción apropiada del mensaje es encriptada acorde a una llave de encriptación secreta, conocida solo por la entidad origen y destino del mensaje .

Los protocolos de seguridad, los mecanismos de seguridad que interactuan con ellos, y el modelo administrativo, forman el esquema completo de seguridad para **SNMPv2**. En este esquema existen diferentes niveles de seguridad como veremos a continuación.

En la siguiente figura se muestran los formatos de los encabezados de diferentes mensajes **SNMPv2**, los cuales pueden incluir o no diferentes tipos de servicios de seguridad.

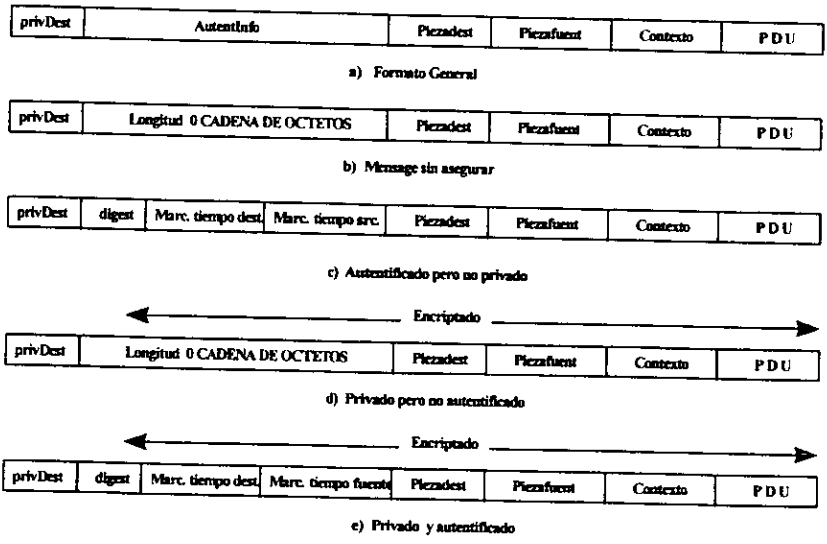


figura 4.21 Formatos de mensajes con diferentes niveles de seguridad

Los formatos de los mensajes **SNMPv2**, son muy parecidos a los formatos de los mensajes de los protocolos de seguridad **SNMP**, con la diferencia que en estos mensajes existe un cambio adicional: la inclusión de un campo destinado a la definición de un contexto.

El encabezado general del mensaje consiste de cinco campos: los campos *Piezadest* y *Piezafuent*, los cuales contienen los nombres de las piezas **SNMPv2** destino y fuente del mensaje, el campo *Autentinfo* contiene información que concierne al protocolo que se encarga de realizar la autenticación del mensaje, el campo *privDest*, que contiene también el nombre de la pieza destino **SNMPv2**, por ultimo el campo *Contexto* donde se define el tipo de contexto para este mensaje. Los campos anteriores pertenecen al encabezado del mensaje, además del encabezado también existe el **PDU**, el cual representa la operación de adminitración deseada.

La figura 4.20 también muestra encabezados de diferentes mensajes dependiendo del nivel de seguridad en el que son transmitidos. Si el mensaje no es seguro (sin ningun servicio de seguridad) 4.20 b), entonces el campo autentinfo consiste de una cadena de longitud 0 representada en **ASN.1**. Si el mensaje es autenticado pero no privado, entonces el campo autentinfo consiste de tres subcampos, dos de ellos (*marca de tiempo fuente y destino*), son marcas de tiempo que representan el momento en que se genero el mensaje en la entidad fuente y destino, cuando el mensaje es transmitido y recibido se realiza un computo en la entidad destino que verifica si el tiempo de generación del mensaje determinado por la entidad fuente concuerda con el tiempo de generación computado por la entidad destino; además de las marcas de tiempo otro subcampo es el que contiene al resumen realizado sobre un porción apropiada del mensaje, con el establecimiento de estos parametros se llega a concluir si el mensaje es o no autentico.
Las figuras d) y e) de la figura 4.20 muestran el formato del mensaje cuando el servicio de confidencialidad es provisto, en este caso el mensaje (incluyendo el encabezado y el **PDU**) con excepción del campo privDest, es encriptado, el campo privdest no es encriptado con la finalidad que la entidad receptora pueda reconocer la pieza **SNMPv2** fuente del mensaje.

Transmisión de un mensaje.

En el siguiente diagrama de flujo se muestran los pasos que una entidad fuente realiza antes de enviar un mensaje

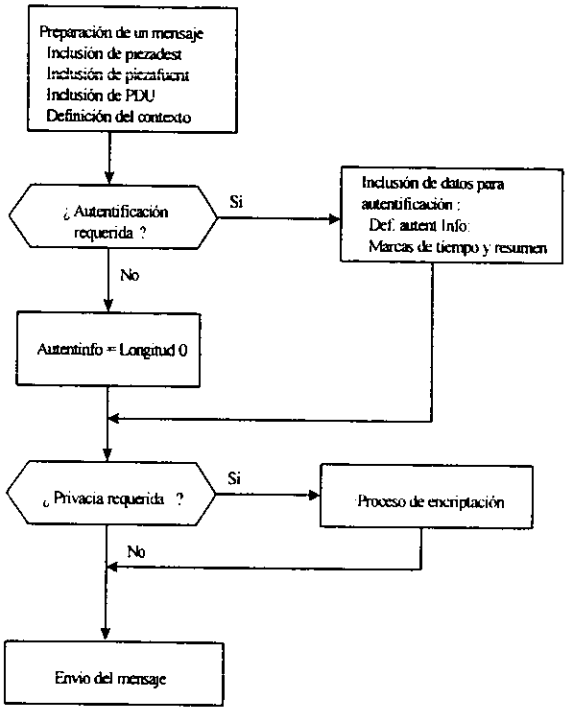


Figura 4.22 Diagrama genérico del proceso de transmisión de un mensaje.

En el siguiente diagrama se muestran los pasos que una entidad destino realiza cuando recibe un mensaje.

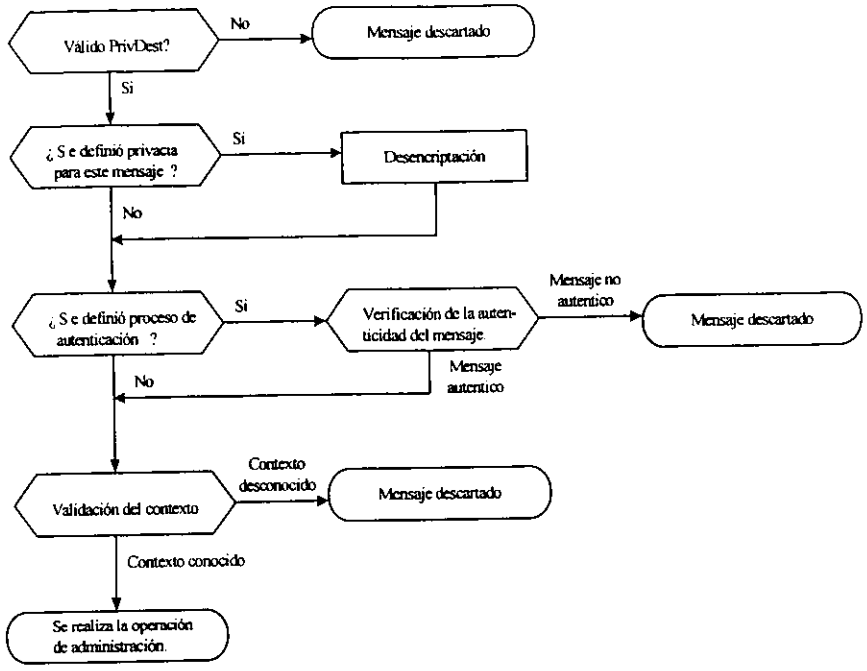


figura 4.23 Diagrama genérico de la recepción de un mensaje.

4.4.7 Coexistencia con SNMP

Una de las más importantes caracteristicas, de **SNMPv2** es la coexistencia con **SNMP**, este es un punto primordial, ya que no todos los dispositivos que conforman una red son capaces de soportar las nuevas versiones de los protocolos, en el caso de la familia **SNMP** muchos elementos solo soportan la primera versión y otros elementos están capacitados para soportar nuevas versiones de este protocolo, entonces tenemos aquí un ejemplo de una esquema de administración no heterogeneo.

La base fundamental de **SNMPv2** es **SNMP**. La evolución de **SNMP** a **SNMPv2** se intento que fuese lo menos complicada posible tanto para los desarrolladores como para los usuarios, el adelanto que provee finalmente esta característica, es que las entidades **SNMPv2** puedan mantener relaciones con entidades administradoras **SNMPv2**, agentes **SNMPv2** y agentes **SNMP**.

Son dos las áreas que son importantes mencionar en un ambiente donde se maneje **SNMPv2** y **SNMP**:

1. La información administrativa
2. La operación de los protocolos

4.4.7.1 Información administrativa

La estructura de la información administrativa **SMI** de **SNMPv2** es muy parecida a la estructura de la información administrativa de **SNMPv1**, de hecho la primera se basa en la segunda. Sin embargo hay cambios y adiciones en la definiciones de dicha información entre ambos protocolos.

Las definiciones de una **MIB** usuando **SMI** de **SNMPv1** pueden continuar siendo usadas con **SNMPv2**. Para adaptar estas definiciones a **SMI** de **SNMPv2** algunos cambios son necesarios. Sin embargo es importante recalcar que estos cambios son necsarios solo para la adaptación a la estructura de la información administrativa **SNMPv2**, pero no lo son para la coexistencia, es decir es posible que un agente mantenga una **MIB SNMP** y siga coexistiendo en una ambiente **SNMPv2 - SNMP**.

4.4.7.2 Operación de los protocolos

El protocolo **SNMPv2** es casi identico al protocolo **SNMP**, usan esencialmente los mismos formatos **PDU**. El mayor cambio es una extensión del conjunto de **PDU**, lo que incluye el **PDU GetBulkRequest** y el **PDU InformRequest**.

La coexistencia a nivel transferencia de información administrativa, puede darse de dos maneras:

Comportamiento Agente - Proxy

El camino más fácil para la coexistencia entre dos entidades **SNMPv2** y **SNMP**, es a través de un **proxy**. Una entidad **SNMPv2** actuando en un papel de agente puede ser implementada y configurada para actuar en el papel de **proxy** entre agentes **SNMP** y entidades administradoras **SNMPv2**.
El **proxy** necesita realizar dos mapeos de **PDUs**: Los **PDUs** que vienen desde la entidad administradora son convertidos a **PDUs SNMP** para ser enviados a un agente **SNMP** (**SNMPv2** a **SNMPv1**) y los **PDUs** que vienen desde un agente **SNMPv1** deberan ser convertidos a **PDUs SNMPv2** para ser enviados a una entidad **SNMPv2** (**SNMPv1** a **SNMPv2**). Este manejo de **PDUs** por el **proxy** es ilustrado en la siguiente figura:

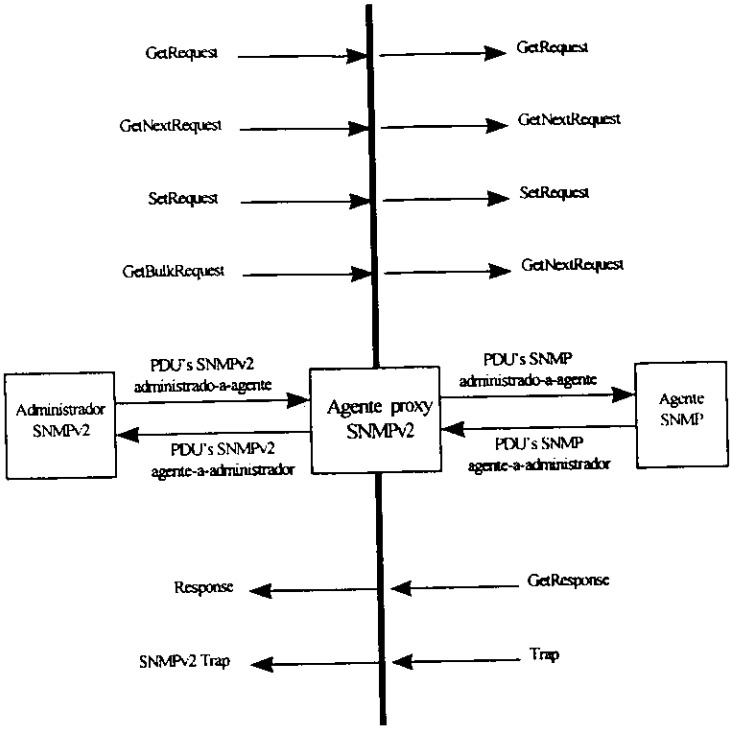


Figura 4.24 Coexistencia a través de un agente proxy

El mapeo de **PDUs SNMPv2** a **SNMPv1**, se realiza conforme a las siguientes reglas:

- Los **PDU GetRequest, GetNextRquest** y **SetRequest** son pasados sin cambio
- El **PDU GetBulkRequest** es convertido a un **PDU GetNextRequest** representando la lista de variables adjuntas.

El mapeo de **PDUs SNMPv1** a **SNMPv2**, se realiza conforme a las siguientes reglas:

- Un **PDU GetResponse** es pasado sin cambio
- Un **PDU Trap** es convertido en un **PDU Trap SNMPv2**

Comportamiento Administrador Bilingüe

Un camino alternativo para realizar la coexistencia entre los protocolos **SNMPv1** y **SNMPv2** es la implementación de un nodo administrador que *hable* ambos protocolos. En la sigiuiente figura se muestra un implementación de este tipo:

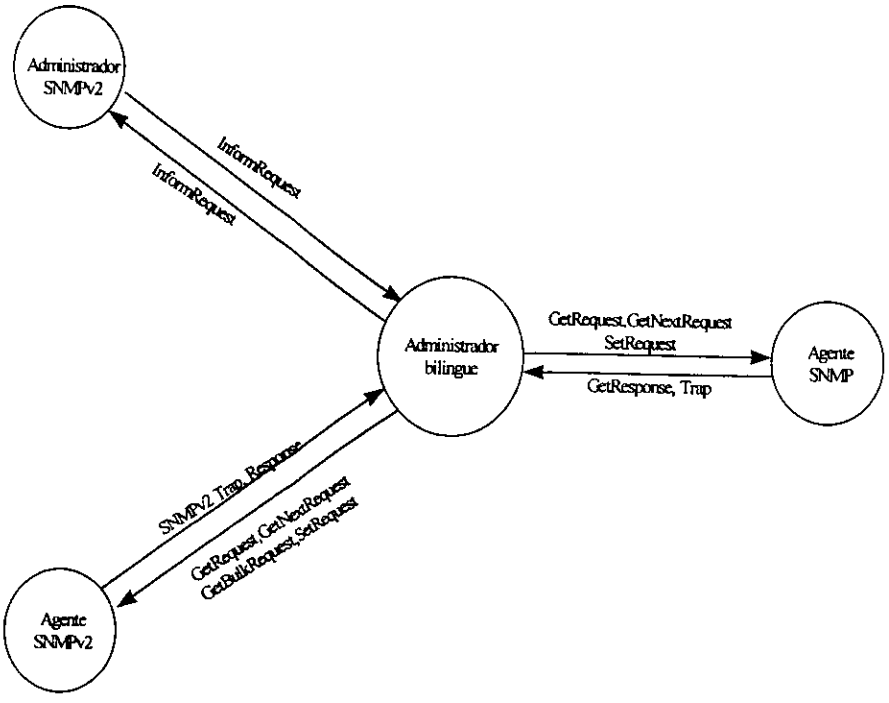


Figura 4.25 Coexistencia por un nodo administrador bilingüe

Cuando una aplicación de administración en el nodo administrador necesita contactar a una entidad de protocolo actuando en el papel de agente, la entidad administradora a nivel protocolo en el nodo administrador comunicará **PDUs SNMPv1** y **SNMPv2** basada en información dentro de una base de datos local.

4.5 CMIS/CMIP

4.5.1 Administración de red sobre OSI

De todas las áreas de **OSI**, el conjunto de estándares de administración de red, es el más voluminoso y complejo. Los sistemas de administración de **OSI** están definidos por un conjunto de estándares realizados conjuntamente por **ISO** y **CCITT**[4].

La administración de red sobre **OSI** se basa en un conjunto servicios, los cuales fueron diseñados para proveer un esquema de administración robusto y aplicable a cualquier dispositivo de red. Este conjunto esta dividido en dos:

- **CMIS Common Management Information Service** o servicio común de información administrativa, el cual define los servicios con los cuales puede ejercerse la administración de red.
- **CMIP Common Management Information Protocol** o protocolo común de información administrativa, el cual se encarga de la transferencia de la información administrativa requerida por los servicios de **CMIS**.

4.5.1.1 CMIS/CMIP dentro del modelo de capas de protocolo OSI

El protocolo de administración de red **OSI** basa su funcionamiento en el modelo de capas de protocolo **OSI**, dicho protocolo de administración se encuentra en la capa de aplicación.

[4] International Consultative Committe on Telegraphy and Telephony

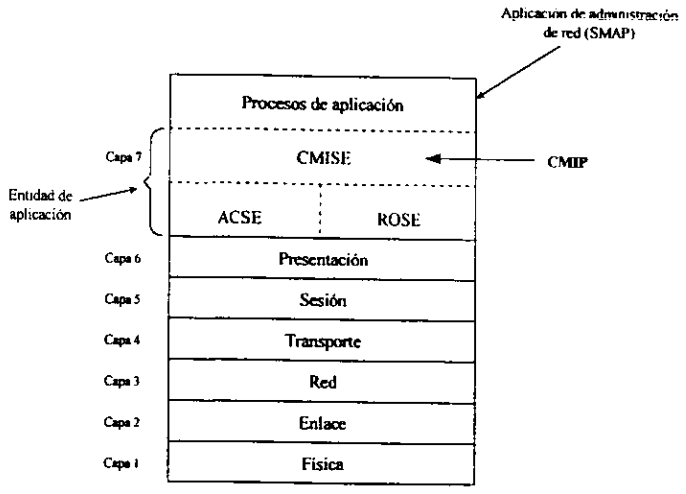


Figura 4.26 CMIP dentro del modelo de capas OSI

Como se puede observar en el diagrama, en el modelo de capas de protocolo **OSI** la capa de aplicación es más compleja, cada programa de aplicación colocado en dicha capa es conocido como *proceso de aplicación*, la comunicación entre procesos de aplicación se realiza por medio de una *entidad de aplicación*. Las entidades de aplicación son responsables del intercambio de información entre procesos de aplicación contenidos en diferentes nodos y representan las funciones de comunicación de un proceso de aplicación. Una entidad de aplicación contiene un conjunto de elementos que le permiten llevar a cabo sus tareas, llamados *elementos de servicio de aplicación*, estos elementos son un conjunto de funciones integradas. **ACSE** y **ROSE** , son ejemplos de estos elementos de servicio de aplicación. Cuando una comunicación es requerida entre dos entidades de aplicación, una o más asociaciones entre los elementos de servicio son establecidas. Finalmente, un protocolo de capa de aplicación es para el modelo **OSI** una entidad de aplicación, la cual cumple sus objetivos mediante un interrelación de los elementos de servicio de aplicación.

Los elementos de servicio de aplicación más importantes envueltos en la comunicación de información administrativa son:

- ***ACSE:*** El elemento de servicio de control de asociación **ACSE**, fue diseñado para administrar conexiones a nivel de la capa de aplicación, estas conexiones son llamadas *asociaciones*. **ACSE** permite intercambiar información entre dos entidades de aplicación que permite identificar cuales son las entidades que estan involucradas, además de establecer que otros elementos de servicio de aplicación pueden ser usados sobre la asociación.
- ***ROSE:*** El elemento de servicio de operación remota **ROSE**, es el equivalente en **OSI** a las llamadas a procedimientos remotas **RPC**. **ROSE** realiza las siguientes operaciones:
    1. Invocación de una operación para ser realizada en un sistema remoto
    2. Regresar el resultado de la operación invocada.
    3. Si es el caso, regresar mensajes de error a la entidad invocadora.
    4. Re - ejecutar la operación
- ***CMISE:*** El elemento de servicio de información administrativa común **CMISE**, es el elemento de servicio de aplicación que provee todos los servicios de administración de red. **CMISE** utiliza a **ACSE** y a **ROSE** para cumplir sus objetivos.

4.5.1.2 La arquitectura de los servicios de administración de red en OSI

Los elementos de un nodo participante en un esquema de administración de red **OSI**, son los siguientes:

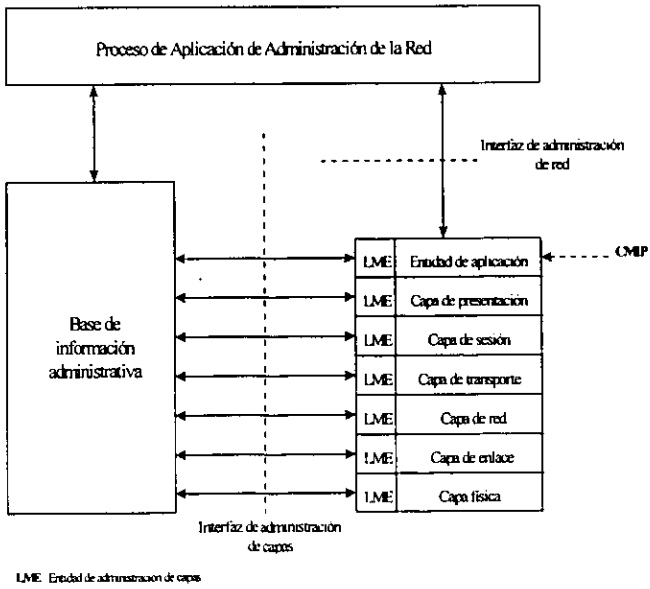


Figura 4.27 Elementos de un nodo en un esquema OSI de administración.

- *Proceso de aplicación de administración de red (SMAP [5]).* **Software** local dentro de un nodo que es responsable de la ejecución de las funciones de administración de red.

- *Entidad de aplicación de administración de red (SMAE [6]).* Esta entidad que recide en la capa de aplicación y es responsable del intercambio de información administrativa con otras **SMAE** en otros nodos, un protocolo de aplicación de red estandar es utilizado para este proposito.

- *Entidad de administración por capa.* Entidad alojada en cada capa de la arquitectura **OSI** para proporcionar funciones de administración de red en cada capa.

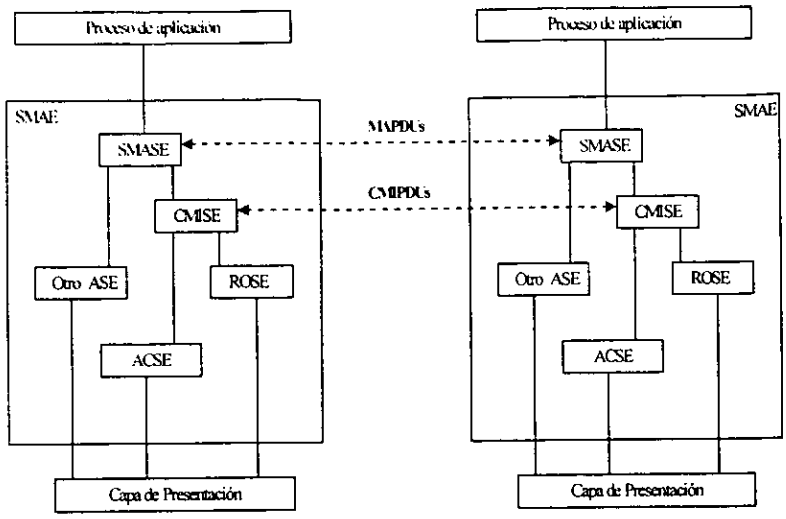
- *Base de datos de administración.* Colección de información de administración.

Los módulos de aplicaciones de administración de red **SMAP** dentro de un nodo, están habilitados para actuar tanto en el papel de administrador como en el papel de agente. La definición de los papeles administrador y agente en la arquitectura de administración de red **OSI**, no difiere de la definición empleada en la arquitectura **SNMP**, así es que un **SMAP** actúa como administrador cuando el nodo que lo contiene tiene a su cargo la realización de tareas de administración de red sobre varios elementos, por otro lado un **SMAP** actúa como agente cuando el nodo que lo contiene es un nodo administrable y por lo tanto tiene que responder a tareas de administración de red.

Un elemento muy importante en cualquier esquema de administración de red, es el protocolo que comunica la información administrativa entre los nodos pertenecientes a dicho esquema. En el esquema **OSI** el protocolo de administración de red se concibe como una entidad de aplicación (**SMAE**), como ya vimos anteriormente una entidad de aplicación esta definida como un conjunto de elementos de servicio de aplicación, en este caso dos de los elementos (**ACSE** y **ROSE**) han sido desarrollados para uso general en una gran variedad de aplicaciones. Pero existen mas elementos de servicio de aplicación involucrados en **SMAE**, en la siguiente figura se muestra la estructura detallada de la entidad de aplicación de administración de red.

[5] Por sus siglas en inglés: Systems-Management application process

[6] Por sus siglas en inglés: Systems-Management application entity



MAPDU: Unidad de datos del protocolo de aplicación de administración.
CMIPDUs: Unidad de información común del protocolo de administración.

Figura 4.28 Estructura de la entidad de aplicación de administración de red

Dos elementos de servicio de aplicación que son específicos para la administración de red son: **CMISE** y **SMASE** [7] . **SMASE** provee varios servicios disponibles al proceso de aplicación, los cuales implementan funciones básicas de administración sobre todas las áreas que cubre la administración de red (administración de fallas, administración de acceso, administración de desempeño, administración de configuración, etc.). Para las funciones que requieren comunicación con otros sistemas, **SMASE** se basa en **CMISE** el cual a su vez se basa en **ACSE** y **ROSE**, para juntos proveer el ambiente y las funciones propicias para la comunicación entre nodos.

La administración de red en la arquitectura **OSI** requiere que todos los sistemas ilustrados en las figuras 4.27 y 4.28, existan en cada uno de los sistemas administrables.

En **OSI** igualmente que en **TCP/IP** las tareas de administración son posibles por la manipulación de objetos administrados. Cada elemento contiene un número determinado de objetos, cada uno de los cuales es una estructura de datos que corresponde a una entidad que será administrada. El uso de los principios de orientación a objetos para definir a la información administrativa es una de las más importantes aportaciones de la administración de red en **OSI**. Un objeto es definido por los atributos que este contiene, operaciones que puede realizar, notificaciones que puede emitir, y sus relaciones con otros objetos.

[7] Por sus siglas en inglés: Systems-Management aplication-service

4.5.2 Estructura de la información administrativa (SMI) y base de datos de información administrativa (MIB).

La estructura de la información administrativa **SMI** para los servicios de administración de red **OSI**, usan **ASN.1** y la filosofía del diseño orientado a objetos para definir la información administrativa. Cada recurso que es monitoreado y controlado por los servicios de administración de red de **OSI** es representado como un *objeto administrado*. Una **MIB** puede definirse entonces como una colección estructurada de objetos administrados.

Ya que la estructura de la información administrativa SMI utiliza el diseño orientado a objetos, existe un modelo que permite definir en base a dicho diseño la información administrativa, este modelo es conocido como *Modelo de información de administración.*

4.5.2.1 Modelo de información administrativa

Conceptos básicos

Por principio de cuentas, debemos aclarar que el termino objeto administrado tiene su parte correspondiente en lo que se conoce como *objeto* en el diseño orientado a objetos, a su vez el termino *clase de objeto administrado* tiene su comtraparte en el mismo diseño en lo que se conoce como *clase*.

En el modelo de información administrativa un *objeto administrado* es definido en terminos de los atributos que éste posee, operaciones y notificaciones que puede realizar, y sus relaciones con otros objetos administrados. Cada objeto administrado es un elemento de *una clase de objetos administrados*. Una clase de objetos administrados es un modelo o "machote" para los objetos administrados que comparten los mismos atributos, notificaciones, y operaciones de administración. Las siguientes caracteristicas definen a una clase:

- Atributos visibles en el objeto administrado
- operaciones de administración de red que pueden ser aplicadas al objeto administrado
- comportamiento mostrado por el objeto administrado en respuesta a las operaciones de administracion de red
- notificaciones que pueden ser emitidas por el objeto administrado
- posición de el objeto administrado en la jerarquia

A continuación se describirán conceptos del diseño orientado a objetos adaptados a la definición de la información administrativa en **OSI.**

Encapsulamiento. La encapsulación es una caracteristica fundamental del diseño orientado a objetos. En el contexto de la administración de red la encapsulación tiene el siguiente significado: Cada tipo de recurso a ser administrado es representado por un clase de objeto administrado. Los datos administrativos relacionados con cada recurso y los procedimientos de administración aplicables a dicho recurso son "enpacados" (encapsulación) juntos en un objeto correspondiente. Las aplicaciones de administración tiene acceso al recurso a traves del objeto para ejercer acciones de control y monitoreo.

Atributos. Los elementos de datos contenidos en un objeto administrado son llamados atributos. Cada atributo representa una propiedad del recurso que el objeto representa, estas propiedades pueden ser: caracteristicas operacionales, estado actual, condiciones de operacion. Los atributos son mayormente usados para el monitoreo.

Clase de objeto administrado y herencia. Una clase de objeto administrado es un modelo que define las operaciones de administración, atributos, notificaciones y comportamiento de un tipo particular de objeto. Todos los objetos que comparten estos mismos elememtos son miembros de la misma clase. De estas clases se pueden derivar una o mas subclases las cuales *heredan* las operaciones, atributos, notificaciones y comportamiento de la clase por la cual se derivaron. De esta manera se puede crear una estructura jerárquica de clases de objetos, con esto, se tiene la conveniencia de definir una gran variedad de tipos de objetos con un minimo de texto.

Operaciones. Las operaciones de administración de red se aplican directamente a los atributos de un objeto o bien al objeto como un todo. La operación de administración puede llevarse a cabo sobre el objeto si y solo si el elemento que envia la operacion tiene un acceso autorizado al objeto.

Comportamiento. Un objeto administrado exhibe cierto comportamiento, esto es, como el objeto reacciona a operaciones realizadas en el. El comportamiento de un objeto administrado ocurre en respuesta a estímulos interno o externos. Los estimulos externos son operaciones de administración que llegan al objeto en forma de mensajes CMIP. Los estimulos internos son eventos dentro del objeto administrado y el recurso asociado a dicho objeto, un ejemplo de estimulo internos es un contador. Todos los objetos pertenecientes a una misma clase exiben el mismo comportamiento.
El comportamiento define:

- La semántica de los atributos, operaciones y notificaciones
- La respuesta a las operaciones de administración
- Las circunstancias en las cuales las notificaciones deberan ser omitidas
- Los efectos de las relaciones con otros objetos.

Notificaciones. Los objetos administrados estan capacitados para emitir notificaciones cuando algún evento extraordinario ocurra al objeto. Las notificaciones pueden ser transmitidas al exterior por medio de mensajes de protocolo.

Principios de contención y estructura de la MIB

Hemos visto que la facilidad de crear subclases en el diseño orientado a objetos permite la creación de una estructura jerarquica, la cual refleja las relaciones de varios tipos de objetos. Sin embargo dicha estructura jerárquica no refleja a la estructura real de la **MIB**, esta ultima estructura es definida a traves del uso de la facilidad de *contención* del diseño orientado a objetos.

La contención es un estructura de relaciones para objetos administrados, en la cual la existencia de un objeto administrado es dependiente de la existencia de otro objeto administrado. A un objeto que tiene uno o mas objetos dependientes se le llama *objeto contenedor* y un objeto que es dependiente se le llama *objeto contenido*. Un objeto dependiente o subordinado puede estár contenido en solo un objeto administrado superior, esto es con el fin de cumplir con la regla de que la estructura de una **MIB** debe ser un estructura de arbol. Un objeto contenedor, puede a su vez estar contenido en otro objeto.

La siguiente figura muestra la estrucura general de la MIB para la información administrativa en **OSI.**

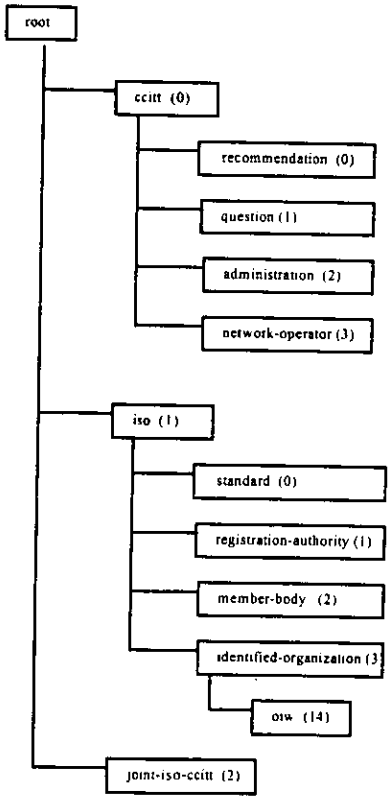


Figura 4.29 MIB CMIS/CMIP

De esta estructura básica se generan varios nodos que son los objetos administrados los cuales se basan en el principio de contenimiento del diseño orientado a objetos, pero se debe tener en mente que este árbol es la estructura tangible de la **MIB**, sin embargo existe una estructura jerárquica aparte donde se involucran las relaciones de las clases de los objetos administrados, esta estructura de clases es tomada en cuenta en el momento de la definicion de cada objeto.

En el caso de la familia **SNMP** la estructura basica de la **MIB OSI**, es en si toda la estructura de la MIB SNMP, ya que todos los objetos en dicha **MIB** son escalares o tablas y no tienen atributos, ni se define un comportamiento especifico a cada objeto, y mucho menos existen las clases.

4.5.3 Acceso a la información administrativa

El acceso a la información en el modelo de administración de red OSI se basa en la ejecución de operaciones de administración sobre la información administrativa representada por objetos.
Estas operaciones son realizadas por las entidades administradoras y llegan a las entidades administradas por medio de paquetes definidos por el protocolo de administración de red. Las operaciones de administración pueden ser divididas en dos categorías: Las que se aplican a los atributos de los objetos administrados y las que se aplican a las instancias de los objetos.

4.5.3.1 operaciones sobre los atributos

Las siguientes operaciones aplican a los atributos de un objeto administrado

- Obtener un valor de un atributo
- Remplazar una valor de un atributo
- Modificar o establecer el valor de un atributo
- Adicionar un miembro a un atributo
- Remover una miembro a un atributo

4.5.3.2 operaciones sobre los instancias de los objetos administrados

- Crear
- Borrar
- Acción

4.5.4 Los elementos fundamentales de la administración de red OSI: CMIS/CMIP

La función principal dentro del esquema de administración OSI es el intercambio de información administrativa entre dos entidades (agente y nodo administrador), por medio de un protocolo. Esta funcionalidad es referida como **CMISE Common Management Information Service Element** o elemento de servicio común de información de administración.

La definición de **CMISE** esta dividida en dos partes:

1. La interface con el usuario, especificando los servicios proveídos, conocido como **CMIS**

2. El protocolo **CMIP**, gracias al cual se pueden llevar a cabo los servicios proveídos por **CMIS.**

CMISE a través de **CMIS** provee varios servicios que permiten realizar las operaciones de administración en **OSI**. Para el intercambio de información administrativa entre nodos **CMISE** emplea a **CMIP** el cual define las unidades de datos para realizar dicha transferencia, a su vez **CMIP** se basa en los elementos de servicio de aplicación **ACSE** y **ROSE**, para llevar a cabo sus funciones.

4.5.4.1 Servicio común de información administrativa: CMIS

El servicio común de información administrativa **CMIS**, define los servicios de administración de red **OSI**. Estos servicios son evocables por aplicaciones de administración de red a fin de comunicarse remotamente y ejercer las tareas de administración. Estos servicios son especificados en termino de *primitivas*, las cuales pueden ser vistas como comandos o llamadas a procedimientos. **CMIS** tiene definidas tres clases de servicios:

Servicio de asociación. Estos servicios son usados primordialmente para el establecimiento y realización de conexiones a nivel capa de aplicación entre sistemas bajo un esquema de administración de red. El control sobre la inicialización, terminación y comportamiento anormal de las conexiones es manejado por los siguientes primitivas:

- **M-INITIALIZE**
- **M-TERMINATE**
- **M-ABORT**

La primitiva **M-INITIALIZE** inicia una asociación entre dos sistemas. La primitiva **M-TERMINATE** se encarga de los parámetros de terminación de una sesión, finalmente **M-ABORT** se encarga de la manipulación de conexiones que se desarrollaron de una manera anormal. Todos estos servicios de asociación asumen el uso de **ACSE** para su operación.

Servicios de notificación administrativa. El segundo tipo de servicio que **CMIS** provee es la notificación administrativa. De la misma manera que los **traps SNMP** proveen información de eventos extraordinarios en la red, los servicios de notificación administrativa de **CMIS** proporcionan una función similar para **OSI**. La primitiva que realiza estas notificaciones es **M-EVENT-REPORT**.

Servicios de operación administrativa. Las primitivas de operación administrativa son los siguientes:

- **M-GET**
- **M-SET**
- **M-ACTION**
- **M-CREATE**
- **M-DELETE**

El servicio **M-GET** es usado para que un sistema pueda recuperar información administrativa, este servicio es similar a **Get-Request** de **SNMP**.
El servicio **M-SET** permite modificar la información administrativa y es similar a **Set-Request** de **SNMP** el cual permite la modificación de la información administrativa en un elemento de la red administrada.
El servicio **M-ACTION** permite la realización de una acción sobre diversos elementos de la red. Dicha acción es relativa a el tipo de elemento al cual se le este realizando la petición.
El servicio **M-CREATE** es usado para crear una instancia de un objeto administrado. Cada objeto administrado tiene asociada una instancia, **CMIS** permite muchas instancias de el mismo objeto, pero solo una definición del objeto por si mismo. Una ejemplo del uso de **M-CREATE** puede ser cuando en un nodo administrador se tienen registrados diferentes dispositivos de interconexión de red, cuando un nuevo elemento es adicionado (por ejemplo un **bridge**), este debe ser registrado en el nodo administrador, entonces con **M-CREATE** puede realizarse esta tarea.
El ultimo de los servicios de operación administrativa es **M-DELETE**, este servicio permite borrar una instancia de un objeto administrado.

4.5.4.2 Protocolo común de información administrativa CMIP

Mientras **CMIS** define los servicios de administración de red, **CMIP** define los procedimientos y las unidades de datos para la transmisión de información administrativa.

Existen 11 **PDUs** definidos para el intercambio de información con **CMIP**. La siguiente tabla lista dichos **PDUs**

1. **m-EventReport**
2. **m-EventReport-Confirmed**
3. **m-Get**
4. **m-linked-Reply**
5. **m-Set**
6. **m-Set-Confirmed**
7. **m-Action**
8. **m-Action-Confirmed**
9. **m-Create**
10. **m-Delete**
11. **m-Cancel-Get-Confirmed**

Existen tres tipos de información en cada unidad de datos: en el *registro de argumentos*, en el cual se definen los argumentos o parámetros que serán accedidos en una máquina remota y son entregados por **CMIS** para ser empaquetados en los **PDU**. Los *registros de resultados y errores* contienen información acerca de el resultado de la operación de administración.

Para cada primitiva que proporciona los diferentes servicios **CMIS** existe un **PDU CMIP** definido que transportará los parámetros solicitados en una petición y cuando la petición se haya realizado habrá otro **PDU** asociado para transferir los datos respuesta.
CMIP se basa en **ACSE** y **ROSE** para realizar sus funciones.

4.6 CMOT

El servicio común de información administrativa y protocolo sobre **TCP/IP** comúnmente llamado **CMOT**, propone la implementación de los servicios y el protocolo (**CMIS/CMIP**) de administración de red que ofrece **OSI** en la cima del modelo de capas de protocolo **TCP/IP**.

4.6.1 CMOT dentro del modelo de capas de protocolo TCP/IP

CMOT, esta basado en los modelos, servicios y protocolos desarrollados por **OSI** para la administración de red. **CMOT** demuestra como la administración de red **OSI** puede ser aplicada a un ambiente **TCP/IP** y usada para administrar objetos en una red **TCP/IP**.

El objetivo de **CMOT** es mapear el protocolo de administración de red de **OSI** dentro de ambientes **TCP/IP**. **CMOT** sigue el modelo **OSI** en la capa de aplicación, mientras que usa los protocolos **Internet** en la capa de transporte y de red. Los elementos de servicio de aplicación usados por **CMOT** son **ACSE, ROSE, CMISE**. En vez de implementar estos elementos en la cima del modelo de capas de protocolo **ISO**, la información de **ACSE, ROSE** y **CMISE** es transportada usando los protocolos de transporte **Internet: UDP** y **TCP**. En la siguiente figura se muestra el modelo de capas de protocolo para la implementación de **CMOT**.

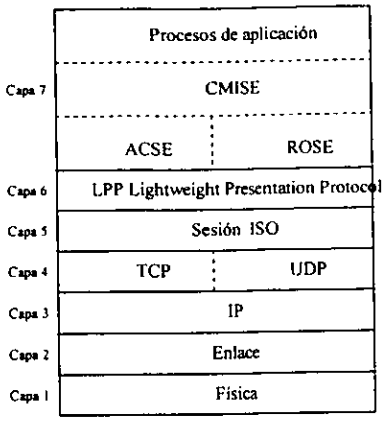


Figura 4.30 CMOT dentro del modelo de capas TCP/IP

La transportación de la información de **ACSE**, **ROSE** y **CMISE** se lleva a cabo gracias al protocolo **LPP Lightweight Presentation Protocol** el cual comunica a **ACSE** y **ROSE** con **TCP/UDP/IP**.

Ya que los elementos de servicio de aplicación (**ACSE, ROSE** y **CMISE**) requeridos por la administración de red, no requieren el uso de todos los servicios de la capa de presentación **OSI**, es posible definir una capa de presentación que provea solo los servicios requeridos por los elementos de servicio de aplicación. Dicho protocolo de presentación es **LPP**, el cual permite el uso de algunos servicios de la capa de presentación sobre **UDP** y **TCP**. Es importante mencionar que todos los servicios provistos por **LPP** son realmente servicios **OSI** de capa de presentación.
Los servicios de capa de presentación permiten a la aplicación de administración seleccionar cualquiera de los dos protocolos de capa de transporte (**UDP** y **TCP**) de **Internet** .

El uso de los protocolos **Internet** es transparente a las aplicaciones de administración de red, ya que ellas interactuan directamente con servicios reales de administración de red **ISO**.

4.6.2 El modelo de información

Existen dos diferentes estructuras de información administrativa **SMI** que son importantes para **CMOT**. La primera es la **SMI** definida por **ISO**, esta estructura de información es importante para **CMOT** ya que **CMIP** fue diseñado teniendo en mente la **SMI** de **OSI**. La segunda **SMI** de importancia es la que define la **MIB Internet** , esta **SMI** es importante ya que los objetos administrados definidos en las redes **TCP/IP** que serán usados por **CMOT** son definidos en términos de esta. Es necesario para que **CMIP** realice sus funciones sobre **TCP/IP**, que la información de administración definida en la **MIB Internet** sea mapeada a información que se apegue a los manejos de **CMIP**.

4.7 LMMP

El **IEEE 802.1b LAN Man Management Protocol** (**LMMP**) intenta proveer una solución de administración de red para ambientes **LAN**. **LMMP** es conocido como **CMIP/CMIS** sobre **802 Logical Link Control (CMOL)** de la **IEEE**. **LMMP** reside directamente en la cima del estándar **802 IEEE**, como se muestra en la siguiente figura.

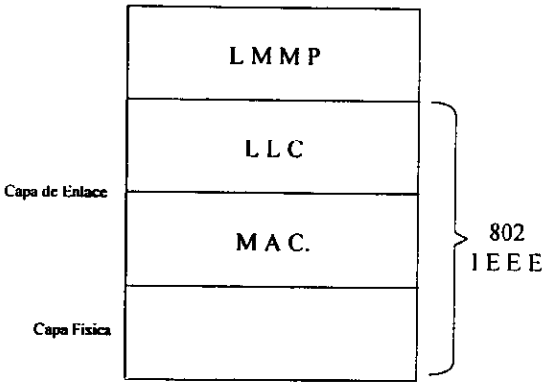


figura 4.31 LMMP sobre el estándar 802 IEEE

La cima del **802 IEEE**, es **LLC Logical Link Layer** y al residir sobre ella se elimina la necesidad de la utilización de los protocolos **OSI**, y de ningún protocolo de capa de red, en otras palabras **LMMP** puede ser implementado sin ningún protocolo de capa de red, pero por supuesto, los mensajes **LMMP** no pueden atravesar ruteadores, por eso es un protocolo de administración que solo se puede implementar en ambientes **LAN**.

4.8 Evaluación de los protocolos descritos y elección del o los mas adecuados

Como hemos visto a través de este capítulo, podemos dividir a los protocolos descritos. en dos grandes familias: la familia **SNMP**, la cual comprende a **SNMPv1** y **SNMPv2**; y la familia de servicios y protocolo de administración de **OSI**, la cual comprende a **CMIS/CMIP** y las adaptaciones de este esquema de administración en **TCP/IP** que es **CMOT** e **IEEE** que es **LMMP**. La evaluación de estas dos grandes familias se realizará en base en una comparación de sus virtudes y defectos.

Las ventajas de la familia SNMP

- La mayor ventaja de la familia **SNMP**, es que su diseño es simple, esto hace posible que sea fácil de implementar en cualquier red, desde una pequeña **LAN** hasta una red de mayores dimensiones. Esto se traduce en:

    1. El costo del software de administración basado en **SNMP** se reduce.
    2. Las tareas de administración que provee **SNMP**[8] son simples, gracias a este hecho los usuarios de **SNMP** y herramientas de administración basadas en el, pueden entender fácilmente cada una de ellas.
    3. La implementación de una arquitectura **SNMP** no toma demasiado tiempo.

- La implementación de **SNMP** no incrementa la carga de trabajo que se traduce en esfuerzo en la red que se va a administrar.
- **SNMP** es altamente portable, es decir cualquier agente **SNMP** puede ser implementado en casi todos los dispositivos de red, y la implementación de la arquitectura completa puede ser instalada sobre casi cualquier plataforma operativa, como **MSDOS, Windows95, UNIX, NetWare, VMS**, etc.
- No es necesaria una gran cantidad de recursos para su implementación.
- **SNMP** es ampliamente usado. El resultado de esto, es que muchas herramientas de administración estan basadas en **SNMP**, así como múltiples compañías de fabricación de dispositivos de interconexión de red construyen agentes **SNMP** que son integrados en los dispositivos que fabrican, este es el caso de **3Com**, **Cabletron**, **Cisco**, etc.
- La facilidad para la expansión de los protocolos es otra ventaja de la familia **SNMP**. Debido a la simplicidad de su diseño los protocolos de la familia **SNMP** pueden ser fácilmente actualizados, como ejemplo podemos citar a **SNMPv2**.

[8] En esta evaluación nos referiremos a la familia **SNMP** simplemente como **SNMP**

Las desventajas de la familia SNMP

- Los protocolos de la familia **SNMP**. no incluyen por si mismos mecanismos de seguridad que protejan la información administrativa que ellos manejan. Si se desea seguridad en un esquema **SNMP** hay que implementar esquemas anexos basados en mecanismos y protocolos que provean seguridad. **SNMPv1** no fue diseñado pensando en la seguridad, posteriormente se desarrollaron los protocolos de seguridad **SNMP**, que pueden proveer un esquema de seguridad bastante aceptable. Sin embargo **SNMPv2** si fue diseñado pensando en el problema de la seguridad, y al mismo tiempo en que se trabajo en la funciones primarias de protocolo, también se trabajo en el esquema de seguridad.
- La definición de la información en la familia **SNMP** no permite manejar grandes cantidades de información, ni información en extremo detallada. Esto es importante cuando se administra una red bastante grande que este extendida a través de ciudades o países. Sin embargo en la versión 2 de **SNMP**, se desarrollo una especificación más detallada de las variables, incluyendo la mejora de las estructuras de datos llamadas tablas que optimizan la recuperación de un número grande de datos.

Las ventajas de la familia CMIP

- La mayor ventaja de **CMIP** se encuentra en la definición de la información administrativa. La definición de variables en **CMIP** es mas compleja y sofisticada, lo cual se traduce en que la estructura de la información en **CMIP** puede cubrir un número mayor de datos y se incrementa el detalle con el que son descritos, esta habilidad permite que cuando un evento anormal ocurre pueda determinarse su naturaleza más rápida y fidedignamente. Esto incrementa la eficiencia de las herramientas de administración.
- **CMIP** fue diseñado previendo los problemas de seguridad que podrían existir, así es que la implementación de **CMIP** no necesita adiciones de mecanismos de seguridad anexos.
- El desarrollo **CMIP** esta respaldado por poderosas compañías y entidades gubernamentales, esto significa un gran panorama de desarrollo para esta tecnología.
- **CMIP** utiliza un protocolo de transporte orientado a la conexión, esto asegura confiabilidad en la entrega de sus mensajes.

Las desventajas de la familia CMIP

- **CMIP** necesita un gran numero de recursos para su implementación. Pocos son los sistemas de cómputo existentes actualmente que podrían manejar por completo la implementación **CMIP** sin modificaciones masivas del ambiente de red y de los propios sistemas, la compra de mucha más memoria, capacidad de almacenamiento de datos y nuevos protocolos de red, serían ejemplos de las modificaciones necesarias para dicha implementación. El único camino para su utilización es el decrecimiento de todo el esquema adaptando sus especificaciones a los protocolos estándar existentes, tal es el caso de **CMOT** y **LMMP**. Con **CMOT** que es **CMIS/CMIP** sobre **TCP/IP** el problema es que su funcionalidad no esta del todo probada y muchos de los desarrolladores de herramientas de administración de red, no quieren invertir en una tecnología que es la mitad **OSI** y la mitad **TCP/IP**. Por otro lado **LMMP** que es **CMIS/CMIP** sobre **802 IEEE** actualmente esta siendo desarrollada y es una tecnología que es en teoría, no aun en la practica.
- Otra fuerte desventaja de **CMIP** es que sus funciones y tareas de administración se basan en una estructura de información compleja fundada en el diseño y programación de objetos, lo que se traduce en que solo personas con un grado de conocimiento de este diseño y programación pueden entenderlas del todo y aprovechar al máximo su potencial.
- No existen muchos fabricantes de herramientas de administración de red ni de dispositivos de interconexión que soporten **CMIP** actualmente.

La familia SNMP vs La familia CMIP

Las principales diferencias entre estas dos familias de protocolos de administración son:

- *El transporte de sus mensajes.* **CMIP** basa la entrega de sus mensajes en protocolos orientados a conexión, esto asegura una entrega de mensajes confiable. Mientras que **SNMP** se basa en un protocolo que no es orientado a conexión y no puede proporcionar una entrega de mensajes realmente confiable. La entrega de mensajes confiable es una característica que puede ser una "arma de doble filo", ya que el objetivo de un protocolo de administración de red es ayudar a la detección, corrección y prevención de problemas en la red, entonces cuando nuestra red se encuentra en un colapso, con un gran número de paquetes yendo y viniendo a través de la red le aumentamos paquetes con reconocimiento que son generados por el protocolo de administración el problema puede incrementarse. Puede pensarse que es una exageración preocuparse por un mensaje más con reconocimiento generado por el protocolo de administración, ya que muchos de nuestros servicios utilizan mensajes de reconocimiento para su transmisión, pero cuando la red se encuentra en un problema esto puede traducirse en un bajo rendimiento del protocolo cuando más se le necesita.
- *La definición de la información administrativa.* La definición de la información que maneja **CMIP** es compleja pero mucho más poderosa, y puede abarcar gran cantidad de información e información mas detallada de los objetos administrados. Por otro lado **SNMP** emplea una definición de la información mucho más simple y su alcance es menor, pero no por eso deja de ser poderosa y funcional.
- *Las operaciones sobre la información administrativa.* El principal axioma de **SNMP** es la simplicidad, y tiene un conjunto pequeño pero funcional y bien organizado de operaciones que ejerce sobre la información administrativa. El conjunto de operaciones de **CMIP** es mayor.
- *Seguridad.* **SNMP** necesita adaptar mecanismos y protocolos anexos para proveer un esquema de seguridad en la información que maneja, **CMIP** no tiene esta necesidad.
- *Portabilidad.* **SNMP** fue diseñado para ser ampliamente portable, **CMIP** intenta ser portable, pero no nació bajo esa filosofía.
- *Necesidad de recursos para su implantación.* **CMIP** necesita un gran número de recursos, como se explico en el punto anterior, por su parte **SNMP** no necesita demasiados recursos para su implementación, y no hay que realizar casi ninguna modificación en la red en donde se va implementar.
- *Amplio uso.* **SNMP** es ampliamente usado actualmente, lo que hace que muchos fabricantes de herramientas de administración de red y de componentes de red puedan construir agentes **SNMP** para sus dispositivos y productos, así es que con esto también se logra un gran soporte de los fabricantes a los usuarios de sus productos. **CMIP** no tiene esta ventaja.
- *Capacidad de expansión.* Aunque las dos familias de protocolo han sido desarrolladas desde hace ya varios años, **SNMP** por su simplicidad es más propenso a la actualización. **CMIP** por su parte se ha actualizado y adaptado al mercado existente, pero cualquier desarrollo o incremento en la familia se ve frenado por el grado de complejidad de la misma.

El protocolo de administración ideal para la red del Instituto de Ingeniería REDII

Considerando la información de todo este capítulo, podemos ver que ambas familias de protocolos de administración tiene ventajas y desventajas. Sin embargo la decisión en la elección de alguna de ellas se basa en su implementación. Para la implementación de cualquiera de los protocolos de estas dos familias se debe tomar en cuenta los siguientes puntos adaptados a las necesidades y recursos de **REDII**.

- *Simplicidad de implantación.* Para **REDII** se busca un protocolo de administración que sea robusto y confiable y que su implementación no deposite una pesada carga a los administradores. Esto se logra con un protocolo que sea sencillo de entender tanto en su funcionamiento como en la definición de información administrativa, además de que este envuelto un ambiente operativo de red que no sea desconocido para los administradores de **REDII**. Estos requerimientos los cumple la familia **SNMP**, ya que su axioma es la simplicidad en su diseño y en su manera de manejar la información administrativa, pero esto no quiere decir que no sea eficiente. El problema con la familia **SNMP** es el manejo de grandes cantidades de información en redes muy grandes **WAN** de alcance nacional e internacional, sin embargo existen actualmente implementaciones de sistemas de administración de red basados en **SNMP** que cubren ciudades y países, y trabajan de una manera eficiente. Por otro lado aunque nuestra red es una red **LAN** grande (que cuenta con 450 nodos entre computadoras personales y estaciones de trabajo), la estructura de información de la familia **SNMP** es altamente adaptable a las necesidades de **REDII**. Además la familia **SNMP** nació en el ambiente operativo de red en el cual se basa **REDII**: **TCP/IP**. *Por las razones anteriores los protocolos que cumplen con este punto son: **SNMPv1 y SNMPv2**.*
- *Necesidad mínima de recursos para su implementación.* En el Instituto de Ingeniería, se esta en disposición de otorgar recursos para la implementación de un sistema de administración y monitoreo de **REDII**, sin embargo no se esta en la posibilidad ni es el objetivo transformar el ambiente de cómputo y de red para adaptarse al sistema de administración, al contrario se busca adaptar un sistema de administración a la infraestructura de computo con la que se cuenta actualmente. *Es por esto que este punto lo satisface los protocolos **SNMPv1 y SNMPv2**.* En este punto cabe hacer la aclaración de que se hace mención tanto a **SNMPv1** como **SNMPv2** ya que algunos de nuestros elementos en la red, no son capaces soportar la versión 2 de **SNMP**, ya que la construcción de estos elementos fue años antes de que **SNMPv2** se desarrollara. Otro punto importante para pensar en la familia **SNMP**, es que **SNMPv2** puede interactuar con **SNMPv1**.

- *Facilidad en su manejo.* La facilidad en el manejo es muy importante, ya que esto se traduce en una facilidad de aprendizaje del protocolo. Esto representa un punto que se debe considerar ya que uno de los objetivos del Instituto de Ingeniería es la formación de personal capacitado en todas la áreas de la ingeniería, es por esto que los administradores de la red son estudiantes de la licenciatura o maestría, y su estancia en la institución puede ser temporal y si se adopta un protocolo de administración complejo esto representará un problema de funcionalidad en la delegación del cargo de un administrador a otro. *Una vez mas los protocolos que cumplen este punto son: **SNMPv1 y SNMPv2**.*
- *Soporte de los proveedores.* El soporte que puedan dar los fabricantes de herramientas de administración y de dispositivos de red que tiene integrado un agente del protocolo de administración, es muy importante para cualquier administrador, ya que si una falla operativa en el producto ocurre es vital tener un respaldo del fabricante para que se solucione lo más pronto posible el problema. Este punto lo cumple tanto **CMIP** como **SNMP**, pero esta ultima familia tiene un más amplio espectro de fabricantes que han decidido adaptar agentes a sus dispositivos o implementar herramientas de administración basados en ella. *Elección: **SNMPv1 y SNMPv2**.*
- *Capacidad de expansión.* **REDII** ha crecido mucho en los últimos años, y el auge de las redes esta aumentando, es necesario que el protocolo de administración para **REDII** sea capaz de adaptarse y cubrir los nuevos desarrollos de la tecnología computacional. Por su simplicidad la familia **SNMP** esta más propensa a la actualización. *Elección: **SNMPv1 y SNMPv2***
- *Seguridad.* La seguridad de la información administrativa es importante para cualquier red y **REDII** no es la excepción. Este es el único punto en donde **CMIP** gana por completo, pero se puede construir un esquema de seguridad, tanto para **SNMPv1** y **SNMPv2**, adicionando mecanismos y protocolos para este fin. Teniendo en cuenta las necesidades anteriores que fueron cumplidas por **SNMPv1** y **SNMPv2** y que si se puede construir un esquema de seguridad para estos protocolos la *elección es nuevamente **SNMPv1 y SNMPv2**.*

Conclusión

A continuación se muestra las tablas de evaluación de las familias, tomando como referencia el método de puntos, mencionado en la metodología.

Criterio	Calif.	Explicación del criterio
Excelente	5	Se asigna a aquellos casos en que el funcionamiento en el punto evaluado, supera en gran medida las expectativas deseadas.
Bueno	4	Satisface los criterio estándar e incluye algunas características especiales.
Suficiente	3	Su función o características son las esperadas.
Pobre	2	Escaso cumplimiento en las funciones o características esenciales.
Inaceptable	1	Es seriamente deficiente.

Tabla 4.1 Criterios de calificación

Característica	Familia SNMP	Familia CMIS/CMIP	Ponderador
Necesidad mínima de recursos para su implantación	5	1	25
Capacidad de expansión	5	4	20
Simplicidad implantación	5	2	20
Seguridad	3	5	15
Facilidad en su manejo	5	2	10
Soporte de los proveedores	4	2	10
Calificación	4.5	2.7	

Tabla 4.2 Calificaciones de las familias evaluadas

Como conclusión de esta evaluación, podemos decir que para el ambiente de cómputo y red del Instituto de Ingeniería, los protocolos de administración que satisfacen sus necesidades, son: **SNMPv1** y **SNMPv2**.

Decimos **SNMPv1** y **SNMPv2** ya que algunos de los dispositivos de **REDII** que se desea administrar, no pueden soportar la implementación de agentes **SNMPv2**. Como **SNMPv2** brinda la posibilidad de convivencia con **SNMPv1** es por eso que se ha elegido ambos protocolos como protocolos de administración para **REDII**.

Capítulo 5

Alternativas y evaluación de diferentes sistemas de administración y monitoreo de red

5.1 Introducción

El trabajo de un administrador de red puede ser descrito de una manera sencilla: Mantener la red de su organización lista y funcionando las 24 hrs. del día, los 7 días de la semana, los 365 días del año, a su máximo desempeño. Si el administrador hace un buen trabajo, nadie sabrá que el administrador existe, pero si se cae la red, esto podría costarle mucho dinero a su organización en productividad, y el administrador tendría un tarea muy grande y a veces difícil para encontrar el motivo de tal caída.

Los sistemas de administración de red nacen a partir del desarrollo de los protocolos de administración y se basan en estos para transmitir y manejar la información administrativa. Como se menciono en el capítulo 5, *un sistema de administración de red es un conjunto de herramientas (**hardware y software**) que proporcionan monitoreo y control sobre una red para incrementar la eficiencia y productividad en la misma.* Los sistemas de administración de red son de gran ayuda a los administradores en la búsqueda de los motivos que propician una caída o un mal funcionamiento de la red, además de ayudar en la toma de decisiones para modificaciones y mejoras de la red.

En este capítulo se describirán diferentes sistemas de administración de red, y en base a sus características y a las necesidades de **REDII**, se seleccionará el más adecuado. Es importante mencionar que se enfatizará la búsqueda en los sistemas de administración de red que se ejecuten sobre **UNIX**, por el sistema operativo de mayor poder con el que cuenta **REDII**.

5.2 Características que se buscan en un sistema de administración de red.

Las siguientes son características básicas que se debe cumplir un sistema de administración para su implementación en el Instituto de Ingeniería :

1) El sistema deberá proveer una interfaz gráfica que pueda producir una estructura jerárquica de la red (mapa) y permitir conexiones lógicas entre los diferentes niveles de la jerarquía. Esta interfaz gráfica podrá contar con : Herramientas de descubrimiento de elementos de red. Esta herramienta proporciona la facilidad de agregar automáticamente elementos de red al mapa, estos elementos son agregados gracias a una búsqueda que la herramienta realiza sin necesidad de la intervención del administrador.
2) Herramientas que permitan medir el desempeño de los nodos administrados, pueden incluirse analizadores de trafico.
3) Generadores de alarmas y reporte de eventos inusuales en la red. Estas herramientas deberán cubrir :
   - Prevención de excedencia. Por ejemplo, esta herramienta deberá ser capaz de activar una alarma que alerte al administrador cuando un disco duro de un servidor esta por llenarse.
   - No - disponibilidad de cualquier nodo.
   - Reporte de errores ocurridos
4) Herramientas de análisis y decodificación de protocolos. Esta herramienta habilita la decodificación y el análisis de todas las capas de protocolo de **TCP/IP**.
5) Análisis de datos y herramientas de graficación.
   - Se deberá contar con un almacenamiento adecuado de los datos generados, mediante una base de datos que puede ser relacional.
   - Se deberá contar con una herramienta que facilite la busqueda de datos en la base
   - Las gráficas generadas por la o las herramientas de graficación deberán ser acumulativas y concisas.
6) El sistema deberá basarse en protocolos de administración de red estándares y actuales.
   - Deberá contar con una definición de **MIB** específica, y deberá contar con la capacidad de edición de la **MIB** para incorporar nuevos elementos administrados.
   - De preferencia el sistema puede incluir un convertidor de los datos **MIB** a formatos simples que cualquier persona no preparada en el lenguaje **ASN.1** pueda entender.
7) El sistema deberá ser fácil de implantar y expandir. Es decir el sistema deberá ser adaptable a cualquier tipo de red que se tenga, e igualmente si la red tiene o no la misma plataforma operativa, además de facilitar la adición de aplicaciones y desarrollos requeridos por el administrador.
8) Capacidad para la creación de desarrollos hechos por el administrador y posibilidad de integrarlos en el sistema.
9) Documentación clara y concisa.
10) Un mínimo de equipo adicional. Esto es, mucho del software y hardware requerido por el sistema de administración deberá ser encontrado en el equipo existente.

5.3 Descripción de sistemas de administración de red

La primera compañía en desarrollar un sistema de administración de red capaz de monitorear y controlar redes consistentes en miles de nodos, fue **IBM**. El sistema que desarrollo en los años ochenta esta compañía fue llamado **NetView**, el cual en nuestros días conserva su nombre. **NetView** abrió un camino de desarrollo para sistemas de administración de red que hoy en día está calificado como uno de los mercados más actuales de la industria de las comunicaciones de datos, el cual incluye competidores como **Digital Equipment Corporation, Hewlett Packard y Sun Microsystems**.
Este naciente campo de aplicaciones de administración de red ha cambiado significativamente a través de la última década, creando nuevas posibilidades, las cuales permiten a los administradores de red ejercer monitoreo y control sobre su red.

5.3.1 HP OpenView

HP OpenView es una solución integrada enfocada a la administración de una red con diferentes plataformas y equipos como : estaciones de trabajo, servidores y PC's. HP OpenView da a los administradores de red, las herramientas para monitorear y controlar en forma centralizada todos los dispositivos y recursos de la red reduciendo de esta manera los costos de operación y administración de los sistemas.

HP OpenView es una solución integral diseñada de manera modular, lo que permite que dependiendo la complejidad de la red se tome la decisión de cuales son los módulos que cada empresa necesita dependiendo de la complejidad de su red.

Hoy en día, **OpenView** es capaz de monitorear y controlar recursos de una localidad remota sin importar que estos recursos sean de diferentes proveedores.

HP OpenView brinda un punto de control centralizado al emplear una interfaz de usuario común, una arquitectura de aplicación común, una base de datos de información común y un mecanismo común para monitorear y controlar los dispositivos de la red.

La estructura modular del **HP OpenView** da la posibilidad de que se adquieran únicamente las funciones de administración que realmente necesita la empresa, además con la plataforma SNMP es posible que aplicaciones de monitoreo o administración más complicadas puedan ser integradas posteriormente.

HP OpenView, desde su aparición en 1989 ha establecido los estándares de administración para TCP/IP y SNMP. Puede administrar todos los dispositivos de la red a los cuales se les pueda asociar una dirección IP como equipos HP, PC's ,puentes, ruteadores, **hubs** y **hardware** de otros proveedores.
HP OpenView puede ser empleado en una red de área local sencilla (**LAN**) o bien en una red compuesta por varias **LANs** interconectadas.

HP OpenView cumple con las características citadas en el punto 5.2 de la siguiente manera

- *Punto 1. Interfaz gráfica.* **HP OpenView** proporciona una detallada vista de la red administrada, por medio de :
  - ✓ **Consola SNM.** Es una interfaz gráfica para el usuario y a través de ella puede ejercer múltiples tareas de administración.
  - ✓ Herramienta de descubrimiento. Automáticamente descubre y agrega al mapa general de la red todos los segmentos y nodos de red que pueda encontrar.

- *Punto 2. medición del desempeño.* **HP OpenView** no cuenta con herramientas especificas para la medición del desempeño, por supuesto tampoco cuenta con analizadores de trafico de red. Sin embargo con la ayuda de la información otorgada por herramientas orientadas a otras funciones se puede lograr un calculo aceptable del desempeño de los nodos administrados.

- *Punto 3. generación de alarmas y reporte de eventos.* Configura reportes automáticos de eventos especiales. Las notificaciones de problemas pueden ser observadas en una ventana en la pantalla o bien con una notificación remota vía **módem.** Asigna prioridades a los diferentes eventos con lo que permite al administrador enfocarse a los problemas más críticos.

- *Punto 4. herramientas de análisis y decodificación de protocolos.* **HP Openview,** no realiza ninguna de estas tareas.

- *Punto 5, análisis de datos y herramientas de graficación.*
  - ✓ Gráfica cualquier dato de forma inmediata sin necesidad de programar, la cual puede ser personalizada fácilmente.
  - ✓ Maneja información histórica con lo que se pueden detectar tendencias que sirven para anticipar problemas antes de que ocurran.

- *Punto 6. protocolos de administración de red actuales y estándares.* **HP OpenView,** basa sus funciones en **SNMP**

- *Punto 7, el sistema deberá ser fácil de implementar y expandir.* La estructura modular de **HP OpenView,** le permite adicionar nuevos desarrollos fácilmente.

- *Punto 8, oportunidad de creación de desarrollos para los usuarios.* **HP OpenVIew** provee esta herramienta, a través de la adición y automática ejecución de **UNIX Shell Scripts.**

- *Punto 9. documentación clara y concisa* La documentación de este producto es muy buena, desde los papeles informativos hasta los mismos manuales. Las hojas de especificaciones pueden encontrarse en varios formatos : Impresas en papel, en **Internet** a través de hojas con formato **HTML**, y en CD - ROM.

- *Punto 10, un mínimo de equipo adicional.* Los requerimientos de **hardware** y **software** para la instalación de **HP OpenView**, son los siguientes:

*Estación de trabajo **HP***
Hardware
HP 9000 serie 700 o serie 800
96 MBytes RAM
300 Mbytes en disco duro
software
HP-UX 9.0
Plataforma SNMP HP OpenView 3.2
Ingres RDBMS versión 6.4/03
X11 Windows System X11R5
LAN/9000
Servicios ARPA

*Estación de trabajo **SUN***

SunOs 4.1.2/4.1.3
Solaris 2.1 o posterior
NCS versión 1.5.1 o DCE RPC
Servicios ARPA
Agente **SNMP** con **MIB I** o **MIB II**

5.3.2 NetView

NetView es el más antiguo sistema de administración de red comercial, el cual brinda operaciones que cubren todas las áreas de la administración de red. **NetView** Puede administrar redes **TCP/IP** y redes **SNA**, de una manera eficiente y amigable.

De las características citadas en el punto 5.2 **NetView** cubre las siguientes:

- *Punto 1, Interfaz gráfica.* Una poderosa interfaz, que permite la representación de todos los dispositivos que sean accesibles vía **SNMP** y sus conexiones. También cuenta con herramientas de descubrimiento automatico de elementos.

- *Punto 2, medición del desempeño.* **NetView** no cuenta con herramientas especificas para la medición del desempeño, por supuesto tampoco cuenta con analizadores de trafico de red. Sin embargo con la ayuda de la información otorgada por herramientas orientadas a otras funciones se puede lograr un calculo aceptable del desempeño de los nodos administrados.

- *Punto 3, generación de alarmas y reporte de eventos.* **NetView** es capaz de generar alarmas que prevengan al administración de posibles fallas o bien de problemas que representen ya una falla real. Además genera reportes de eventos, que son almacenados en bases de datos para su posterior análisis.

- *Punto 4, herramientas de análisis y decodificación de protocolos.* **NetView,** no realiza ninguna de estas tareas.

- *Punto 5, análisis de datos y herramientas de graficación.* **NetView** cuenta con herramientas de análisis de datos y herramientas de graficación completas y efectivas.

- *Punto 6, protocolos de administración de red actuales y estándares.* **NetView**, puede basarse en los protocolos de administración de red que actualmente estan a la vanguardia : **SNMP** y **CMIP**. En las hojas de especificación del producto no se menciona ninguna herramienta para la manipulación de la **MIB**.

- *Punto 7, el sistema deberá ser fácil de implementar y expandir.* **NetView** es un producto diseñado para la fácil instalación y manipulación por los usuarios. Además de ello al ser un sistema de administración con una ya larga vida, ha sufrido modificaciones y mejoras, todas ellas de una muy buena calidad ya que estan respaldadas por una gran experiencia. **NetView** pertenece a una familia de desarrollos modulares que permiten al administrador de red, implementar varias herramientas que le proveerán gran ayuda en sus tareas.

- *Punto 8, oportunidad de creación de desarrollos para los usuarios.* **NetView** incluye un ambiente de desarrollo de herramientas para uso especifico que el administrador puede crear. Este ambiente de desarrollo es **API (Application Program Interfaces)**.

- *Punto 9, documentación clara y concisa* La documentación de este producto no es muy buena en cuanto a papeles informativos se refiere. Para el usuario que esta en búsqueda de información de este sistema, no hay muchas oportunidades de encontrarla. La información encontrada a este respecto fue mínima y se encontró : Impresas en papel, en **Internet** a través de hojas con formato **HTML**. Los manuales del sistema no se tuvieron a la mano para realizar una evaluación de ellos.

- *Punto 10, un mínimo de equipo adicional.* **NetView**, necesita una Estación de trabajo **IBM RS/6000.**

5.3.3 SunNet Manager (SNM) 2.2.2

Mas de 20,000 usuarios actualmente usan **SNM**, para monitorear y controlar sus redes. Usando protocolos de administración de red estándares, **SNM** es una poderosa herramienta que es tanto plataforma de administración de red como ambiente de desarrollo en la misma área. Con las herramientas que cuenta, **SNM** puede proveer tareas de administración de red sobre las siguientes áreas : administración de fallas, configuración, acceso, desempeño, y seguridad. En adición, un ambiente de desarrollo flexible y robusto se ha integrado a **SNM**, con la finalidad de habilitar a los administradores en la construcción de sus propias herramientas adaptadas a sus propios requerimientos.
SNM esta basado en la familia **SNMP** para proveer monitoreo y control sobre recursos conectados a través de redes **TCP/IP** básicamente. Con otros productos **SunSoft**, **SNM** puede administrar diversos ambientes, incluyendo redes **DECnet** y **FDDI** y puede también interactuar con **NetView** de **IBM**.
Los más comunes usos de **SNM** son :

- Analizar desempeño de los recursos en una red
- Identificar y resolver fallas
- simplificar y automatizar tareas de administración

Características generales

Las tres más importantes características de **SNM** son :

- Herramientas de usuario. Las herramientas de usuarios de **SNM**, habilitan operaciones de monitoreo y control de la red y de sus recursos, las interfaces gráficas son sencillas de usar reduciendo así los requerimientos de capacitación para uso del producto.
- Arquitectura distribuida. **SNM** esta basado en una arquitectura de sistema de administración de red distribuida, la cual proporciona a los usuarios la habilidad de administrar redes que integren elementos de diferentes fabricantes, y que varían en tamaño y complejidad. **SNM** puede administrar desde 10 nodos hasta miles de nodos.
- Interfaces de programación de aplicación **API**. **SNM** provee herramientas de desarrollo para construir poderosas herramientas que permiten al administrador complementar la funcionalidad de **SNM** mediante la creación de herramientas que estén orientadas a resolver problemas específicos.

SNM cumple con las características citadas en el punto 5.2 de la siguiente manera

- *Punto 1, Interfaz gráfica.* **SNM** proporciona una detallada vista de la red administrada, por medio de :

  - ✓ **Consola SNM**. La consola **SNM**, es una aplicación central de administración. Es una interfaz gráfica para el usuario y a través de ella puede ejercer múltiples tareas de administración como son: configuración, identificación y diagnostico de fallas, Monitoreo y control de los dispositivos de la red.
  - ✓ **Topology Map Display.** Esta herramienta representa los elementos de la red en forma de objetos y las conexiones de red existentes entre ellos, conformando así la topología de la red. Además puede incluir mapas de países u otras imágenes para ser desplegados detrás de los objetos y conexiones proveyendo un punto visual de referencia.
  - ✓ Ventanas jerárquicas. Todas las ventanas pueden ser arregladas en una jerarquía para representar varios niveles del ambiente de red.
  - ✓ Ventanas perspectivas. En adición los usuarios pueden crear ventanas a su gusto, de tal manera que se pueda tener una vista de la red como se desea. Por ejemplo, una sola ventana puede desplegar todos los ruteadores administrados en la red.
  - ✓ **Discovery tool**. Esta herramienta automáticamente crea la base de datos y construye la representación gráfica de los elementos encontrados en la red y las conexiones entre ellos. Esta herramienta ahorra considerablemente tiempo en la configuración del mapa de representación de red y la base de datos. **Discovery tool,** busca detalladamente todas las direcciones **IP** y los elementos con **SNMP** en la red.

- *Punto 2, medición del desempeño.* **SNM** no cuenta con herramientas especificas para la medición del desempeño, por supuesto tampoco cuenta con analizadores de trafico de red. Sin embargo con la ayuda de la información otorgada por herramientas orientadas a otras funciones se puede lograr un calculo aceptable del desempeño de los nodos administrados. Las principales herramientas son:

  - ✓ **Auto-Management**. Esta herramienta permite a los usuarios monitorear recursos sin la necesidad de especificar operaciones de administración iniciales. **Auto-management**, empieza el monitoreo de los recursos con peticiones predefinidas. Los recursos que pueden ser monitoreados con esta herramienta deberán soportar uno de los siguientes agentes: **SNMPv1** o **SNMPv2, hostperf , ping.**
  - ✓ **Request Management.** Con esta herramienta, el usuario podrá adecuar a sus necesidades de monitoreo cada petición hecha por el nodo administrador a los objetos administrados.

- *Punto 3, generación de alarmas y reporte de eventos.* La herramienta con la que cuenta SNM para proveer la generación de alarmas, es un demonio de producción de **Traps** basado en SNMP. Una herramienta muy eficiente que permite visualizar de una manera resumida los reportes de eventos generados es la ventana de eventos.

- *Punto 4, herramientas de análisis y decodificación de protocolos.* **SNM,** no realiza ninguna de estas tareas.

- *Punto 5, análisis de datos y herramientas de graficación.*

  - ✓ **Browser tool.** Esta herramienta provee la habilidad de recuperar, organizar y desplegar la información administrativa. Los usuarios pueden generar reportes, revisar detalladamente la información de sus recursos administrados, y realizar análisis de datos comparativos con esta herramienta.
  - ✓ **Grapher tool.** Esta herramienta permite la generación de gráficas en 2 y 3 dimensiones. Esta es una herramienta muy útil para diagnosticar posibles problemas en la red. Por ejemplo la gráficas del % de utilización de **CPU** de los servidores, pueden ser utilizadas para comparar el rendimiento de cada servidor y si este esta trabajando a niveles adecuados con respecto a las aplicaciones que corre.

- *Punto 6, protocolos de administración de red actuales y estándares.* **SNM** esta habilitado para trabajar con **SNMPv1** y **SNMPv2,** además contiene agentes **proxies** desarrollados por la tecnología **SNM.** Una herramienta muy útil es **mib2schema utility** la cual traduce el estándar **SNMP MIB** en un formato fácil de entender, así es que los usuarios no necesitan descifrar las definiciones de los objetos administrados en **ASN.1**

- *Punto 7, el sistema deberá ser fácil de implementar y expandir.* **SNM** es un producto diseñado para la fácil instalación y manipulación por los usuarios. **Sun Microsystems,** la compañía con más desarrollos y ventas en el mundo **UNIX-INTERNET** actualmente, respalda totalmente la expansión de su producto.

- *Punto 8, oportunidad de creación de desarrollos para los usuarios.* **SNM** provee un ambiente de desarrollo para construir poderosas herramientas que permiten al administrador complementar la funcionalidad de **SNM** mediante la creación de herramientas que estén orientadas a resolver problemas específicos. Esto se logra a través de **API (Application Programming Interfaces)**

- *Punto 9, documentación clara y concisa* La documentación de este producto es muy buena, desde los papeles informativos hasta los mismos manuales. Las hojas de especificaciones pueden encontrarse en varios formatos : Impresas en papel, en **Internet** a través de hojas con formato **HTML,** y en CD - ROM.

- *Punto 10, un mínimo de equipo adicional.* Los requerimientos de **hardware** y **software** para la instalación de **SNM**, son pocos como se muestra a continuación:

Hardware
SPARCstation o SPARCserver
X86/Pentium
Software
Solaris 2.4 para x86
Solaris 2.3 o posterior
Solaris 1.1.1 SunOs 4.1.3
Ambiente de Ventanas
OpenWindows 3.1 o posterior
configuración del sistema
32 Mbytes RAM
400 Mbytes de disco duro

5.3.4 Spectrum 4.0

Spectrum es una plataforma de administración de **Cabletron Systems**, que permite el control y monitoreo sobre redes, cuyos elementos pueden ser de origen variado (haciendo referencia al fabricante de dichos elementos). **Spectrum** introduce la tecnología **IMT** (por sus siglas en ingles **Inductive Modeling Technology**), con **IMT**, **SPECTRUM** crea un modelo de cada entidad en la red, incluyendo cables físicos, dispositivos de interconexión de red, **PC's**, servidores y aplicaciones.

Características generales

- Provee capacidades de administración para todos los ambientes de red : **LAN**, **WAN**, **SNA**, **PBX**, **ATM**.
- Provee alarmas que pueden minimizar el tiempo requerido para la localización de una falla.
- Provee herramientas que ayudan a un pronto aislamiento de fallas de **software** y **hardware**.
- Puede ejercer acciones correctivas para asistir a los administradores de red en la solución de problemas.
- Cuenta con herramientas de descubrimiento de elementos de red
- Tiene capacidades de colección y análisis de datos.
- Formula recomendaciones a los usuarios.

Spectrum cumple con las características citadas en el punto 5.2 de la siguiente manera

- *Punto 1, Interfaz gráfica.* **Spectrum** proporciona una detallada vista de la red administrada, por medio de :

  - ✓ **SpectroGRAPH**. Es una interfaz gráfica, que presenta a la red que será administrada en un modelo con una alta calidad gráfica. Además proveen a los usuarios con un fácil acceso a las aplicaciones de **Spectrum**, por medio de "clicks" de ratón.
  - ✓ Ambiente Multi - ventanas. **Spectrum** ofrece un ambiente de ventanas orientadas a la presentación de información específica , como la topología y el estado de la red, el lugar físico de cada dispositivo, y una estructura jerárquica basada en la importancia de los dispositivos.
  - ✓ **AutoDiscovery**. Esta herramienta permite descubrir elementos de red automáticamente, y los integra al mapa establecido previamente, o bien, el administrador cuando inicia la operación de **Spectrum** puede recurrir a esta herramienta para automáticamente crear un mapa de la red que va a ser administrada.

- *Punto 2, medición del desempeño.* **Spectrum** no cuenta con herramientas especificas para la medición del desempeño, por supuesto tampoco cuenta con analizadores de trafico de red. Sin embargo con la ayuda de la información otorgada por herramientas orientadas a otras funciones se puede lograr un calculo aceptable del desempeño de los nodos administrados.

- *Punto 3. generación de alarmas y reporte de eventos.* Las herramientas con las que cuenta **Spectrum** para proveer estos servicios, son la ventanas de alarmas, que muestran las alarmas activas y sus posibles causas, además de contar con la generación de reportes de error que se almacenan en archivos " log " y se muestran también en ventanas.

- *Punto 4, herramientas de análisis y decodificación de protocolos.* **Spectrum,** no realiza ninguna de estas tareas.

- *Punto 5, análisis de datos y herramientas de graficación.* **Spectrum** cuenta con herramientas de análisis de datos y herramientas de graficación completas y efectivas.

- *Punto 6, protocolos de administración de red actuales y estándares.* **Spectrum**, puede basarse en los protocolos de administración de red que actualmente estan a ala vanguardia : **SNMPv2** y **CMIP**. En las hojas de especificación del producto no se menciona ninguna herramienta para la manipulación de la **MIB**.

- *Punto 7, el sistema deberá ser fácil de implementar y expandir.* **Spectrum** es un producto diseñado para la fácil instalación y manipulación por los usuarios. La firma **Cabletron Systems** respaldada totalmente la expansión de su producto, actualmente existen 4 actualizaciones de este producto desde su introducción al mercado

- *Punto 8, oportunidad de creación de desarrollos para los usuarios.* **Spectrum** provee esta herramienta, a través de la adición y automática ejecución de **UNIX Shell Scripts.**

- *Punto 9, documentación clara y concisa* La documentación de este producto es muy buena, desde los papeles informativos hasta los mismos manuales. Las hojas de especificaciones pueden encontrarse en varios formatos : Impresas en papel, en **Internet** a través de hojas con formato **HTML**, y en CD - ROM.

- *Punto 10, un mínimo de equipo adicional.*

5.4 Conclusión

A continuación se muestra las tablas de evaluación de los sistemas de administración, tomando como referencia el método de puntos, mencionado en la metodología.

Criterio	Calif.	Explicación del criterio
Excelente	5	Se asigna a aquellos casos en que el funcionamiento en el punto evaluado, supera en gran medida las expectativas deseadas.
Bueno	4	Satisface los criterio estándar e incluye algunas características especiales.
Suficiente	3	Su función o características son las esperadas.
Pobre	2	Escaso cumplimiento en las funciones o características esenciales.
Inaceptable	1	Es seriamente deficiente.

Tabla 5.1 Criterios de calificación

Características	HP OpenView	NetView	Spectrum	SunNet Manager	Ponderador
Interfaz gráfica	5	5	5	4	10
Medición del desempeño	3	3	3	3	10
Generación de alarmas y reporte de eventos	5	5	5	5	10
Herramientas de análisis y decodificación de protocolos	2	2	3	2	10
análisis de datos y herramientas de graficación	5	4	5	4	10
Protocolos de administración de red actuales y estándares	5	5	5	5	10
facilidad de implantación y expansión	5	5	4	5	10
Creación de desarrollos por los usuarios	4	5	4	5	10
Documentación	5	4	4	5	10
Mínimo de equipo adicional	5	3	4	5	10
Calificación	4.4	4.1	4.2	4.3	

Tabla 5.2 Calificación de los sistemas de administración de red.

Aunque en la tabla anterior **SunNet Manager**, no obtuvo la calificación más alta, se eligió como el sistema de administración adecuado a **REDII**, por sus características anteriormente descritas, además de no ser necesaria la dedicación de una gran cantidad de recursos para su implantación. Dichos recursos existen actualmente en **REDII**. Otro punto importante para la elección de **SunNet**, es que la mayoría de las estaciones de trabajo con las que cuenta el Instituto de Ingeniería son de la marca **Sun Microsystems**, de esta manera se puede aprovechar muchas de las aplicaciones que **SunNet** incluye, que son especificas para servidores **SUN**, además de poder implantar dichas aplicaciones a los demás servidores. La siguiente tabla muestra la distribución de servidores por marca en el Instituto de Ingeniería.

Marca	No. de Estaciones	Porcentaje
Silicon Graphics	3	5 %
IBM	5	8%
HP serie 700	9	17 %
Sun	31	66 %

Tabla 5.3 Estaciones de trabajo en REDII

Otro punto que se considero para la elección fue el costo de **SunNet Manager,** el cual es menor a los sistemas restantes. Además **SunNet Manager** es un sistema robusto que cubre todas las áreas de la administración de red, de una manera amigable y fácil de aprender. La característica más importante para **REDII** que se encontró en **SunNet** es que provee un ambiente para el desarrollo de propias herramientas, haciendo con esto que se incremente la funcionalidad del sistema de administración de red.

Capítulo 6

Implantación

6.1 Introducción

Después de la selección de la mejor alternativa en sistemas de administración de red para el Instituto de Ingeniería, el siguiente paso es su implantación.

En este capítulo se describe la instalación del sistema de administración seleccionado, el equipo donde dicho sistema operará, así como su configuración y mantenimiento.

Se muestra que fue necesario el uso de algunas herramientas de administración adicionales, enfocadas a tareas específicas, las cuales ayudaron a la integración de todo el sistema de administración en la red del Instituto de Ingeniería.

6.2 Arquitectura del sistema de monitoreo y administración en REDII

Utilizando los conceptos empleados en el capítulo 3 de esta tesis, podemos decir que la arquitectura del sistema de administración de red implantado en la red del Instituto de Ingeniería, es centralizada. Se decidió utilizar este tipo de arquitectura en base a las características de REDII[1].

Con esta arquitectura se pretende:

- Tener un control estricto sobre el sistema de administración el cual sea responsabilidad de una o dos personas a lo sumo.
- Tener un solo lugar físico y lógico donde se operare el sistema, con el fin de actuar rápidamente en caso de contingencia.
- Desarrollar un buen conocimiento de las aplicaciones de administración y en general de todo el sistema por los encargados del mismo.
- Ahorro de recursos de computo y aprovechamiento de los recursos humanos.

En la siguiente figura se muestra la arquitectura del sistema de administración para REDII.

[1] Las características de REDII son descritas ampliamente en el capítulo 1

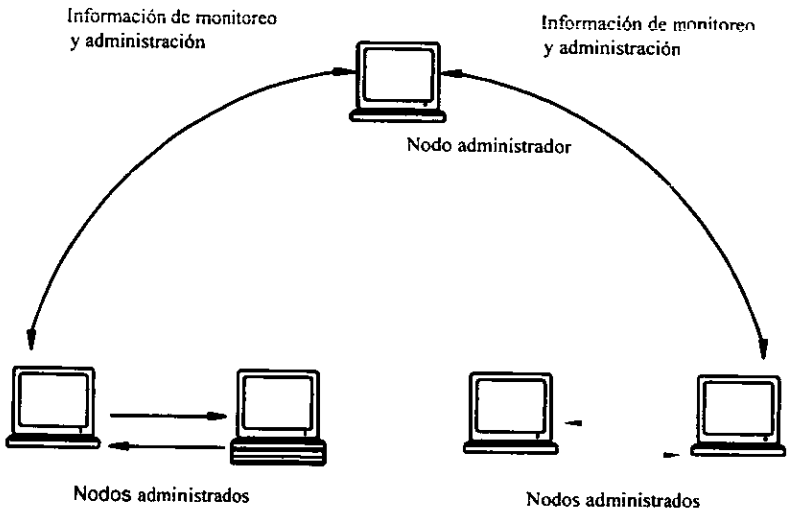


figura 6.1 arquitectura del sistema de administración y monitoreo de REDII

El nodo administrador contendrá el software de administración de red (SAR) y agentes. Sin embargo, para cubrir todas las necesidades de administración y monitoreo de REDII fue necesario utilizar herramientas distintas a **SunNet Manager**, las cuales residen en los nodos administrados, sin embargo los datos generados por estas herramientas son almacenados y analizados en el nodo administrador, por lo tanto se sigue respetando la arquitectura centralizada. Más adelante en este capítulo hablaremos más a detalle de dichas herramientas.

6.3 Planeación

Antes de empezar con la instalación del sistema de administración y monitoreo, debemos dedicar tiempo para definir a donde queremos llegar con nuestro sistema y que es lo que debemos tomar en cuenta para cubrir nuestros objetivos.

Para realizar lo anterior se establecieron las siguientes preguntas:

- *¿Cuál es la topología de la red ?* Debemos conocer la topología de nuestra red para así poder definir un mapa de la misma.
- *¿Cuales son los nodos de la red que deseamos monitorear ?* Es necesaria la identificación de los nodos que son de nuestro interés, deberemos tomar en cuenta sus funciones y características con el fin de poder visualizarlos como elementos de nuestro sistema.

- *¿Cuales son los nodos críticos?* Se deberá identificar los dispositivos que tengan mayor importancia dentro de la red con la finalidad de establecer una jerarquía dentro de nuestro sistema.
- *¿Cuantas vistas deseamos tener en el mapa de la red ?* Esta pregunta es necesaria ya que hay que definir como será el mapa de la red tomando en cuenta la topología. Por ejemplo, se podrían definir todos los nodos críticos en una sola vista, o bien hacer un mapa que represente fidedignamente la ubicación física de cada nodo, o bien se puede agrupar todos los nodos de un tipo en una sola vista, etc.
- *¿Que tipo de información nos interesa obtener y como conseguirla?* Recordando el capítulo 3 de esta tesis, clasificamos a la información obtenida por el monitoreo en: Información estática, dinámica, y estadística, es necesario saber cual de estos tipos de información nos será más útil, además de conocer cuales métodos debemos emplear para obtenerla. Recordando también del capítulo 3, tenemos dos métodos principales para obtener información: **Polling** y reporte de eventos.

Al aplicar las preguntas anteriores a nuestro entorno, obtuvimos lo siguiente:

- *¿Cuál es la topología de la red ?* REDII esta configurada para emplear una topología física en estrella, la cual se basa en una tecnología **Ethernet** conmutado a 10 Mbps. Se emplea cable par trenzado categoría 3 y 5, además de fibra óptica como medios de transmisión, y los protocolos de comunicación que utiliza son **TCP/IP**.
- *¿ Cuales son los nodos de la red que deseamos monitorear ?* Los nodos que se desean monitorear, son aquellos que tienen cierta importancia dentro de cada coordinación que conforma al Instituto de Ingeniería.

	Coordinación	Nodo (Workstation)	Dirección IP
1	Sistemas de Cómputo	[illegible]	[illegible]
2	Sistemas de Cómputo	Tonatiuh, SS20	132.248.53
3	Sistemas de Cómputo	Altair,	132.248.156.21
4	Sistemas de Cómputo	Tlaloc, Sparcclassic	132.248.156.69
5	Automatización	Automatización, SS10	132.248.156.49
6	Sismología e inst. sísmica	Gea SparcServer 670	132.248.53.24
7	Hidráulica	Vortex (Hp)	132.248.53.212
8	Hidráulica	Quetzal (Hp)	132.248.53.242
9	Ingeniería sismológica	Runasimi (Sun)	132.248.154.97
10	Estructuras y materiales	Esma (S.G)	132.248.154.250
11	Mecánica aplicada	Leviatan (Sun)	132.248.53.92
12	Geotecnia	Merlin (Hp)	132.248.53.93
13	Geotecnia	Sacks (IBM)	132.248.53.164
14	Hidráulica	Cefm (Sun)	132.248.155.246
15	Ingeniería sismológica	Haskell (S.G)	132.248.154.245
16	Hidráulica	Hidro (Sun)	132.248.53.210
17	Ingeniería sismológica	Inti (Sun)	132.248.154.246
18	Ingeniería sismológica	Hermes (IBM)	132.248.53.76

Tabla 6.1 Servidores principales del Instituto de Ingeniería

Edificio	Nodo (Dispositivos de Interconexión y UPS)	Dirección IP	Coordinación
1	Concentrador SEH-24 (Cabletron)	132.248.153.2	Sistemas de Cómputo
	Concentrador SEH-24 (Cabletron)		
	Concentrador SEHI-34 (Cabletron)		
2	MMAC-M5FNB con salida IRBM (Cabletron)	132.248.154.1	Sistemas de Cómputo
3	Concentrador SEHI-34 (Cabletron)	132.248.154.2	Sistemas s de Cómputo
4	MMAC-M3FNB con salida IRBM (Cabletron)		Sistemas de Cómputo
5	MMAC-M3FMB con salida IRBM (Cabletron)	132.248.155.1	Sistemas de Cómputo
	Concentrador HP48J2602A (HP)	132.248.155.4	
	Concentrador MRXI (Cabletron)	132.248.155.2	
	2 concentradores SEH-24 (Cabletron)	132.248.155.3	
6	Concentrador MRXI (Cabletron)	132.248.156.2	Sistemas de Cómputo
12	MMAC-M8FNB con salida IRBM (Cabletron)	132.248.156.2	Sistemas de Cómputo
4	Switch	132.248.153.1	Sistemas de Cómputo
1	UPS	132.248.153.4	Sistemas de Cómputo
2	UPS	132.248.154.3	Sistemas de Cómputo
3	UPS	132.248.154.4	Sistemas de Cómputo
4	UPS	132.248.153.3	Sistemas de Cómputo
5	UPS	132.248.155.5	Sistemas de Cómputo
6	UPS	132.248.156.4	Sistemas de Cómputo
12	UPS	132.248.156.3	Sistemas de Cómputo
	Gateway 53.254	132.248.53.254	Externo a II, REDUNAM
	Gateway 156.254	132.248.156.254	Externo a II, REDUNAM
	Gateway 153.254	132.248.153.254	Externo a II, REDUNAM
	Gateway 154.254	132.248.154.254	Externo a II, REDUNAM
	Gateway 155.254	132.248.155.254	Externo a II, REDUNAM

Tabla 6.2 Dispositivos de interconexión de red, UPS y Gateways

- *¿Cuales son los nodos críticos?* Todos los nodos en REDII tienen importancia, pero existen nodos que destacan por sus funciones y una falla en ellos puede costar la operación de la red. A continuación se muestra una tabla que cita los nodos críticos en REDII.

REDII, Nodos criticos
Pumas
Tonatiuh
Altair
Tlaloc
Gea
Vortex
Esma
Leviatan
Inti
Hermes
Todos los dispositivos de interconexión, el principal el Switch
UPS en cada edificio
Los Gateways a REDUNAM

Tabla 6.3 Nodos críticos

- *¿Cuantas vistas deseamos tener en el mapa?* Para representar a REDII en el mapa desde donde se controlará la administración y el monitoreo, se decidió hacer una representación de los edificios y dentro de estos definir a los nodos, dando así la ubicación exacta de cada uno de ellos dentro de REDII.

- *¿Que tipo de información nos interesa obtener y como conseguirla?* Consideramos que para nuestro esquema de monitoreo y administración, los tres tipos de información (estática, dinámica y estadística) son útiles, para obtener dicha información deberemos emplear los dos métodos de recolección: **polling** y reporte de eventos. La información que debemos obtener de nuestro esquema, esta basada en las áreas que un sistema de administración y monitoreo debe cubrir (Administración de fallas, desempeño, acceso, seguridad y configuración). A continuación se muestra una tabla donde se cita la información que nos interesa obtener dividida en áreas, clasificada por tipo, y tomando en cuenta que método debemos usar para obtenerla. Como mencionamos en el capítulo 3 de la tesis, el área de administración de seguridad no se incluye en este desarrollo, ya que dicha área requiere de un trabajo más específico que sale de los objetivos de esta tesis.

Administración de fallas	Administración de desempeño	Administración de acceso	Administración de configuración
Conectividad D, E	Disponibilidad ESTD, P	Identificación de usuario D, P	Configuración gral. de los dispositivos EST,P
Integridad de protocolos ESTD, P, E	Tiempo de respuesta ESTD, P	Identificación de hosts D, P	
Saturación de conexión ESTD, E, P	Precisión ESTD, P	No. de paquetes o bytes transmitidos D, P	
Saturación de recursos ESTD, E, P	Rendimiento ESTD, P	Recursos utilizados por el usuario D, P	
Funcionalidad del protocolo de monitoreo D, ESTD, P, E	Utilización ESTD, P		

Tipos de información		Métodos de acceso	
Estática	EST	**Polling**	P
Dinámica	D	Reporte de eventos	E
Estadística	ESTD		

Tabla 6.4 Tipos de información y métodos de acceso.

6.4 Software seleccionado

6.4.1 SunNet Manager 2.2.2

Como vimos en el capítulo anterior, el sistema de administración que mejor se apega a las necesidades de REDII es **SunNet Manager** (snm); la versión que es utilizada en este trabajo es **SunNet** 2.2.2.

SunNet Manager es un sistema de administración compuesto por varias herramientas, las cuales permiten al usuario realizar tareas en diversas áreas de la administración de red. **SunNet Manager** es también una plataforma expansible que permite desarrollar aplicaciones propias, para que el usuario pueda adecuar el sistema a sus necesidades.

El paquete **SunNet Manager** 2.2.2 representa lo que se definió en el capítulo 3 como " ***software de administración de red o SAR*** ". **SunNet Manager** provee: software de presentación, software que realiza las tareas de administración, software de soporte de administración de red, estos tres paquetes conforman como tal el SAR, además de esto, **SunNet Manager** incluye agentes.

6.4.1.1 Arquitectura de SunNet Manager

Al ser considerado como una plataforma de administración, **SunNet Manager** puede catalogarse dentro del modelo administrador/agente definido por **OSI**[2], además de utilizar como estándar de comunicación a **TCP/IP**. La siguiente figura presenta un diagrama de bloques funcional de **SunNet Manager**.

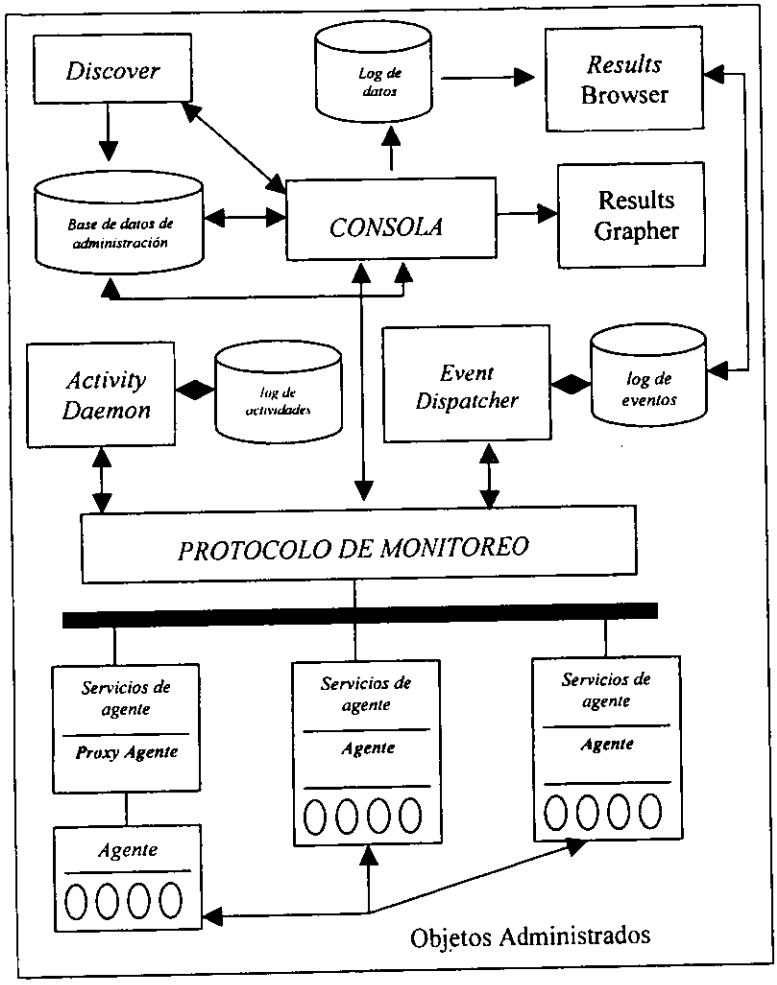


Figura 6.2 Diagrama funcional de SunNet Manager

[2] Descrito a través de todo el capítulo 3 de esta tesis

A continuación se describirán los siguientes componentes del diagrama:

- La *consola* y herramientas: ***Discovery, Results Browser*** y ***Result Grapher***
- La base de datos de administración
- Agentes y Proxies.
- Demonios auxiliares (***activity*** y ***event dispatcher***)

Consola y herramientas

La consola de **SunNet Manager** es la aplicación de administración central en el paquete. Es el lugar donde el usuario inicia las tareas de administración y es en donde la información generada por dichas tareas es presentada al usuario. En términos de lo definido en el capítulo 3, la consola representa al software de presentación en la arquitectura del SAR.

La consola **SunNet Manager** presenta una interfaz gráfica orientada a objetos. Dicha consola fue desarrollada bajo el ambiente **OPEN LOOK**, el cual corre bajo **OpenWindows 3.0**[3] o posterior. **OpenWindows** soporta el protocolo **X11**, el cual permite a la consola y a otras ventanas de aplicación ser desplegadas en un ambiente de red. La consola **SunNet Manager** y todas las ventanas generadas por ella son totalmente compatibles con el estándar **X11** del **MIT**.

SunNet Manager soporta múltiples instancias de la consola ejecutándose en la misma maquina al mismo tiempo. Cada instancia es asociada al nombre del usuario que la invoco.

Todo nodo administrado puede ser definido gráficamente en la consola **SunNet Manager**, por medio de un símbolo que lo identifique, además del símbolo, se asocian características del nodo como nombre, dirección ip, etc. También puede definirse gráficamente el ambiente en el que el nodo administrado reside, por ejemplo un edificio, una red de computo, una ciudad, etc. Esto permite realizar mapas de la ubicación de los nodos administrados, proporcionando al usuario una visualización completa del esquema administrado.

La consola provee mecanismos para iniciar peticiones de *reporte de datos* y de *reporte de eventos*, mediante dichas peticiones los agentes son dirigidos a realizar tareas de administración.
Los reportes de datos indican a los agentes que deben colectar información del nodo administrado en una base de tiempo. Los reportes de eventos indican a los agentes que deben enviar información solo cuando condiciones especificas en el nodo administrado sean encontradas.
Cada petición de reporte que es hecha en la consola contiene información del nodo administrado y el periodo de tiempo o bajo que condiciones el agente debe reportar la información.

[3] Interface gráfica provista por **Solaris**

Otra de las funciones de la consola es desplegar la información generada por los reportes, así como la ocurrencia de eventos, generando alarmas audibles y visuales.

SunNet Manager incluye varias herramientas que pueden ser desplegadas desde la consola:

- ***Discovery tool***. La finalidad de esta herramienta es realizar una búsqueda automática de todos los dispositivos que puedan ser monitoreados a través de **SNMP** dentro de una red determinada. Cada elemento que ***Discovery tool*** encuentra es agregado al mapa descriptivo de la red, asignándole una imagen y una posición jerárquica dentro del mapa. Es también función de esta herramienta crear el mapa físico de la red con la finalidad de tener la ubicación exacta de cada elemento. ***Discovery tool*** es muy útil cuando se inicia por primera vez SNM ya que se encarga de definir automáticamente la red que se va a monitorear.
- ***Results Browser***. Esta herramienta permite examinar y organizar los archivos *log* de SNM
- ***Results Grapher***. La finalidad de esta herramienta es realizar gráficas de la información almacenada en los archivos *log*.

Base de datos de administración

La consola y otras aplicaciones de administración, se basan para su operación en las descripciones de la base de datos de administración (***MDB** Management Data Base)*, la cual contiene definiciones de los elementos que están siendo administrados, los agentes que están disponibles, y las peticiones que se han hecho a dichos agentes.

La base de datos de administración contiene:

- *Definiciones de cada tipo de elemento que puede ser representado en la consola.* Esta definición especifica el nombre del tipo de elemento (por ejemplo: SS10 o SPARCStation 10) y el glifo o icono asociado a el. **SunNet Manager** incluye un gran variedad de tipos de elementos pre-definidos, pero el usuario puede agregar sus propios tipos.
- *Definiciones de instancias de cada tipo de elemento.* Esta definición representa un elemento particular en la red (el nombre del dispositivo, ejemplo: Pumas), además de definir que agentes pueden ser usados para administrar a dicho elemento.
- *Definiciones de los agentes que pueden ser usados para administrar elementos.* Cada agente puede manipular diferentes conjuntos de información o atributos. El conjunto de atributos que puede ser manipulado por cada agente esta definido en un archivo llamado archivo ***schema***. Al menos un archivo **schema** para cada agente debe ser instalado en el nodo administrador.
- *Definiciones de la peticiones que se han hecho a cada agente.* La peticiones que permiten el monitoreo o la administración de cada elemento en nuestra red, son también definidas y almacenadas en la MDB, de esta manera cada vez que se re-incide SNM esta peticiones reanudaran sus tareas.

Cuando un usuario ejecute **SunNet Manager** por primera vez, una MDB se generará para la representación de la red administrada que defina dicho usuario, esta base de datos por usuario es conocida como *runtime database*.

Agentes y proxies.

En la arquitectura de **SunNet Manager** existen dos tipos de agentes los cuales se diferencian por el tipo de acceso que tienen a los objetos administrados: acceso directo e indirecto.

Agentes de acceso directo. Están instalados solamente en **workstations** Sun y tienen acceso directo a los objetos administrados en dicho nodo. Por ejemplo, el agente *hostmen*, que usa el mismo mecanismo que el comando de **Solaris** *netstat -m* para obtener la utilización de la memoria

Agentes de acceso indirecto. Estos agentes tienen la habilidad de administrar objetos que residan en otro tipo de **workstations** diferentes a las Sun o bien en otros dispositivos (ups, impresoras, dispositivos de interconexión de red etc.). Este tipo de agentes son conocidos como agentes **proxy**, los nodos donde residen dichos agentes son llamados **sistemas proxy**.

SunNet Manager provee un conjunto de agentes muy variado, algunos de los cuales generan información muy parecida a comandos de **Solaris** y **SunOs**. Los agentes que vienen con SNM, están desarrollados solamente para plataformas **Sparc**.

A continuación se muestra una tabla de los agentes incluidos en SNM versión 2.2.2.

Nombre del agente	Descripción	Comando **UNIX** relacionado	Tipo
diskinfo	Reporta información de uso de disco	df	Agente
etherif	Estadísticas de las interfaces **Ethernet** (**SunOS**)	-	Agente
etherif2	Estadísticas de las interfaces **Ethernet** (**Solaris**)	-	Agente
hostif	Monitoreo de paquetes IP	netstat -i	Agente
hostmem	Utilización de memoria (**SunOS**)	netstat -m	Agente
hostmem2	Utilización de memoria (**Solaris**)	netstat -m	Agente
hostperf	Información del desempeño del sistema	rup y perfmeter	Agente **proxy**
iostat	Estadísticas de entrada y salida (**SunOS**)	iostat	Agente
iostat2	Estadísticas de entrada y salida (**Solaris**)	iostat	Agente
ippath	Información de paquetes IP	-	Agente **proxy**
iproutes	Tablas de enrutamiento y estadísticas	netstat -r	Agente
layers	Estadísticas de las diferentes capas de protocolo (**SunOS**)	netstat -rs netstat -s	Agente
layers2	Estadísticas de las diferentes capas de protocolo (**Solaris**)	netstat -rs netstat -s	Agente
lpstat	Estado de impresoras	lpq y lpstat	Agente **proxy**
ping	Conectividad IP	Ping	Agente **proxy**
rpcnfs	Estadísticas de **RPC** y **NFS**	Nfsstat	Agente
snmp	Información basada en la definición **MIB** I para **SNMP**	-	Agente **Proxy**
snmp-mibII	Información basada en la definición **MIB** II para **SNMP**	-	Agente **proxy**
sun-snmp	Información basada en la definición **MIB** I para **SNMP** con extensiones para monitorear **workstations** Sun	-	Agente **proxy**
snmpv2	Para administrar dispositivos **SNMPv2**	-	Agente **proxy**
sync	Monitorea líneas seriales sincronas	Syncstat	Agente
traffic	Analizador básico de trafico **Ethernet**	-	Agente **Proxy**

Tabla 6.5 Agentes incluidos en SunNet Manager 2.2.2

Demonios auxiliares

Para su ejecución **SunNet Manager** incluye dos demonios auxiliares: ***activity*** (na.activity) y ***event dispatcher*** (na.event). El primer demonio es un proceso que asegura que las peticiones hechas continúen siendo servidas por los agentes.

El demonio **event dispatcher** es un proceso que se encarga de tomar los eventos reportados por los agentes y encaminarlos a sus destinos (por ejemplo la consola).

6.4.1.2 Instalación

Para la ejecución de **SunNet Manager** 2.2.2 se eligió una maquina **Sparcclassic**, ya que **SunNet** necesita la plataforma operativa **Solaris** 2.3 o mayor. El sistema operativo **Solaris** de **Sun Microsystems** fue desarrollado para arquitecturas de hardware basadas en procesadores **SPARC**[4], la estación de trabajo **Sparcclassic** pertenece a este tipo de arquitectura.

La estación de trabajo seleccionada cuenta con los siguientes elementos en su configuración:

- Procesador Sparc a MHZ
- Unidad de disco duro de 535 Mbytes
- 32 Mbytes de memoria RAM
- Interface de red **Ethernet**
- Unidad de disco flexible

El sistema operativo de la maquina es **Solaris** 2.4, distribuido en cinco sistemas de archivos[5]:

Sistema de archivos	Partición	Espacio en Mbytes
/	/dev/dsk/c0t3d0s0	18.05 MB
/usr	/dev/dsk/c0t3d0s3	230.23 MB
/var	/dev/dsk/c0t3d0s4	8.20 MB
/opt	/dev/dsk/c0t3d0s5	137.27 MB
/home	/dev/dsk/c0t3d0s6	20.23 MB

[4] Los procesadores SPARC fueron también desarrollados por Sun Microsystems

[5] Para el sistema operativo un sistema de archivos es una partición de disco a la cual se le a dado un formato en bloques de datos y que contiene una estructura de tablas las cuales definen direcciones de archivos y directorios. Para el usuario del sistema operativo un sistema de archivos es una colección de archivos y directorios usados para almacenar y organizar información.

Este esquema de particiones permite que en el sistema de archivos " / " quede almacenada toda la información referente a *root*, en " /usr " los binarios del sistema operativo, en " /var " se almacenara archivos *log*[6] del sistema, " /opt " es el espacio reservado para el almacenamiento de paquetes que no son propiamente del sistema operativo, por ejemplo **SunNet Manager** 2.2.2, finalmente " /home " es el espacio reservado para los usuarios de la maquina. Adicionalmente la maquina fue configurada con 96 Mbytes de **Swap**

La estación de trabajo fue incorporada a REDII, dándole una dirección IP, un nombre, y también se le configuro como cliente de DNS.

El paquete **SunNet Manager** se debe instalar en o los nodos administradores, en este caso por ser una arquitectura centralizada, el paquete se instalará en un solo nodo. La instalación del paquete es muy sencilla, y se realiza a través del comando de **Solaris** *pkgadd*, el cual es una interfaz texto muy amigable que lleva prácticamente " de la mano " al usuario. A continuación se muestra la manera de ejecutar el comando pkgadd y la pantalla de inicio del mismo instalando **SunNet Manager**.

Dentro del directorio donde se encuentran los paquetes a instalar, se ejecuta:

```
hosts % pkgadd -d .
```

Después de ejecutar el comando anterior la siguiente pantalla se despliega:

```
The following packages are available:
1  SUNWabsnm     SunNetManager 2.2.2 AnswerBook (for Solaris 2.3)
                 (all) 38.3.4
2  SUNWsnmag     SunNetManager Agents & Libraries
                 (sparc) 2.2.2
3  SUNWsnmct     SunNetManager Core Tools
                 (sparc) 2.2.2
4  SUNWsnmpd     SunNetManager SNMP daemon
                 (sparc) 2.2.2

Select package(s) you wish to process (or 'all' to process
all packages). (default: all) [?,??,q]:
```

Para el nodo administrador se deberán elegir todas las opciones que instalan el paquete **SunNet Manager** completo, además se debe instalar el agente **SNMP**, la instalación de dicho agente requiere los siguientes parámetros: el nombre del administrador de la maquina, la ubicación de la misma, y lo más importante el *read-comunity-name* y el *write-comunity-name*, estos son nuestros passwords de acceso para escritura y lectura en nuestro nodo, si los omitimos, la instalación los configurará como "*public*" , lo cual implica que todo nodo con una gente **SNMP** instalado podrá solicitar y modificar información de administración a nuestro nodo.

[6] En los archivos *log* de un sistema UNIX, se guarda cualquier acontecimiento detectado en la maquina

Para los nodos administrados es necesaria la instalación de todos los agentes que provee **SunNet Manager**, con el fin de que el nodo administrador pueda ejercer acciones de administración sobre ellos, la instalación se hace igualmente con el comando pkgadd, pero únicamente seleccionando como software a instalar los agentes.

A continuación se muestran los archivos de configuración de **SunNet Manager** como del agente **SNMP** en el nodo administrador.

```
# Copyright 1994 Sun Microsystems, Inc. All Rights   Reserved.
# Copyright 1993 Sun Microsystems, Inc. All Rights Reserved.
#
# @(#)snm.conf    2.36 07 Jul 1994 - SunNetManager configuration file
# Copyright (c) 1990,1993 by Sun Microsystems Inc.

# Site-specific configuration information

### Keywords for the SNMP proxy agent:
# Directory list for SNMP schema files. Separate each directory
#  with a colon.
na.snmp.schemás /opt/SUNWconn/snm/agents
# File name of default MIB (in schema format)
na.snmp.default-schema /opt/SUNWconn/snm/agents/snmp-mibII.schema
# File name of per-host schema files
na.snmp.hostfile /home/snm/log/snmp.hosts
# SNMP request timeout
na.snmp.request_timeout 5
# Maximum number of SNMP polling attempts per reporting interval
na.snmp.max_attempts 3
# SNM report timeout
na.snmp.report_timeout 5
# subprocess acknowledgement timeout
na.snmp.ack_timeout 15
# Maximum number of requests a subprocess will perform
na.snmp.max-requests 50
# Maximum number of subprocesses handling requests
na.snmp.max-subprocs 20
# Send trap if no response from device
na.snmp.trap-if-no-response true
# Exit if not performing requests
na.snmp.exit-if-no-requests true

### Keywords for the SNMP proxy agent:
# File name of default MIB (in schema format)
na.snmpv2.default-schema /opt/SUNWconn/snm/agents/snmpv2-MIBII.schema

### Keywords for the hostperf proxy agent:
# rstat(3R) request timeout
na.hostperf.request_timeout 5
# SNM report timeout
na.hostperf.report_timeout 5
# subprocess acknowledgement timeout
na.hostperf.ack_timeout 15
# Maximum number of requests a subprocess will perform
na.hostperf.max-requests 50
# Maximum number of subprocesses handling requests
na.hostperf.max-subprocs 20
# Send trap if no response from device
na.hostperf.trap-if-no-response true
```

```
### Keywords for the ping proxy agent:
# Number of ICMP packets for each 'reach' ping
na.ping.reach-packets 3
# Number of ICMP packets for each 'stats' ping
na.ping.stats-packets 5
# ping request timeout
na.ping.request_timeout 1
# SNM report timeout
na.ping.report_timeout 5
# subprocess acknowledgement timeout
na.ping.ack_timeout 15
# Maximum number of requests a subprocess will perform
na.ping.max-requests 50
# Maximum number of subprocesses handling requests
na.ping.max-subprocs 20

### Keywords for the SNMP trap proxy:
# default file name of per-enterprise traps
na.snmp-trap.default-trapfile /home/snm/log/snmp.traps
#
# Flag used to indicate whether the trap daemon should also send
# raw oid/value in trap report, default is false.
na.snmp-trap.raw false

### Where the link map file lives
# used by discover and Console (and others?) for link management
linkmap           /home/snm/databases/linkmap

### Keywords for the various loggers
# File name for na.activity logs
activity-log    /home/snm/log/activity.log
# File name for na.event logs
event-log       /home/snm/log/event.log
# File name for na.logger logs
monitor-log     /home/snm/log/monitor.log
# File name for agent request logs
request-log     /home/snm/log/request.log

#database location for anyone who cares
snmdb-directory   /home/snm/databases
.....
# Copyright 1994 Sun Microsystems, Inc. All Rights Reserved.
# Copyright 1993 Sun Microsystems, Inc. All Rights Reserved.
# @(#)snmpd.conf  2.16 07 Jul 1994 (c) 1991 SMI

# See below for file format and supported keywords

sysdescr        Sun SNMP Agent, SPARCclassic
syscontact      Coordinacion de Sistemas de Computo
sysLocation     Edif. 12 Instituto de Ingenieria, Unam
#
system-group-read-community     mngt
system-group-write-community    mngt
#
read-community  iider
write-community iider
#
trap            localhost
trap-community  SNMP-trap
#
```

```
#kernel-file      /vmUNIX
#
#Managers         lvs golden
#Managers         swap

############################

# File Format:

# Each entry consists of a keyword followed by a parameter string,
# terminated by a newline.  The keyword must begin in the first
# position.  The parameters are separated from the keyword (and from
# one another) by whitespace.  All text following (and including) a '#'
# character is ignored.  Case in keywords is ignored, but case in
# parameter strings is NOT ignored.

# Supported Keywords:

# sysdescr        String to use for sysDescr.
# syscontact            String to use for sysContact.
# syslocation           String to use for sysLocation.
#
# system-group-read-community Community name needed for read access
#                           to the system group.
# system-group-write-community      Community name needed for write
access
#                           to the system group.
# read-community  Community name needed for read access
#                       to the entire MIB.
# write-community Community name needed for write access
#                       to the entire MIB (implies read access).
#
# trap                  Host names where traps should be sent.
#                       A maximum of 5 hosts may be listed.
# trap-community  Community name to be used in traps.
#
# kernel-file           Filename to use for kernel symbols.
#
# Managers        Hosts that can send SNMP queries.
#                 Only five hosts may be listed on any one line.
#                 This keyword may be repeated for a total of 32 hosts.
```

Iniciación de la consola y creación del mapa

Como mencionamos anteriormente la consola de **SunNet Manager**, es donde el usuario podrá ejercer tareas de administración y observar el comportamiento de la red administrada.

Para dar inicio a la consola **SunNet Manager** es necesario ejecutar el siguiente comando:

```
host% snm &
```

Cuando se ejecuta por primera vez la consola **SunNet Manager**, no existen bases de datos creadas y automáticamente se ejecuta la ventana ***Quick Start*** la cual se muestra a continuación

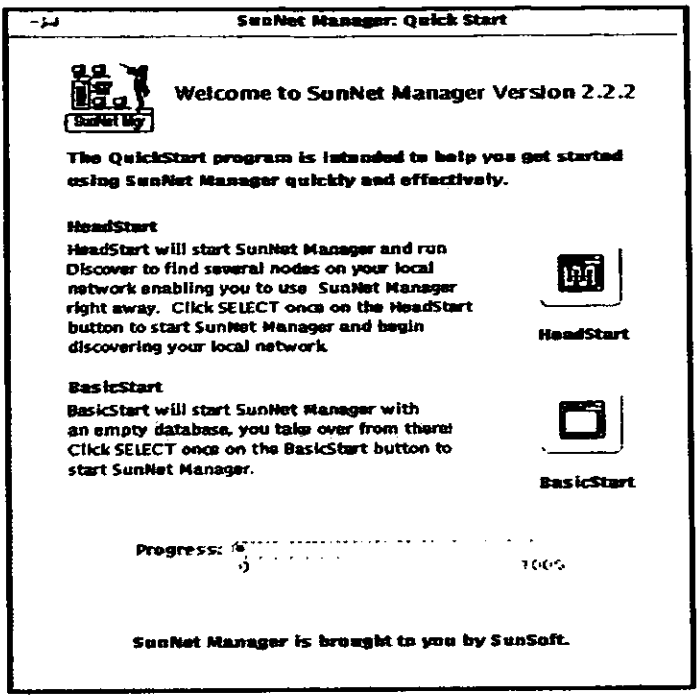


figura 6.3 La ventana Quick Start

En esta ventana se presentan dos botones (***Head Start*** y ***Basic Start***), al elegir el botón **Head Start**, la consola es iniciada y también la herramienta **Discovery** que automáticamente empieza una búsqueda de nodos en la red y va creando en base a la información encontrada un mapa con nodos en la consola. Por otro lado cuando el botón **Basic Start** es elegido solamente la consola es iniciada, dando oportunidad a que el usuario defina el mapa y cada nodo a monitorear.
Cuando se ejecutó por primera vez la consola **SunNet Manager** en nuestro nodo administrador se eligió el botón **Head Start**, el cual realizo una mapa de REDII con casi todos los nodos a administrar, pero no fue lo que se esperaba ya que la disposición de los nodos dentro del mapa no era la que necesitábamos, y ninguno de los nodos estaba personalizado. La opción **Head Start** puede simplificar mucho el trabajo, pero en nuestro caso preferimos usar la opción **Basic Start** ya que nos permitió hacer el mapa de REDII tal cual lo necesitábamos y definir cada nodo de manera personalizada.

Al iniciar la sesión **SunNet** con la opción Basic **Start** de la ventana **Quick Start**, la consola será desplegada de la siguiente manera:

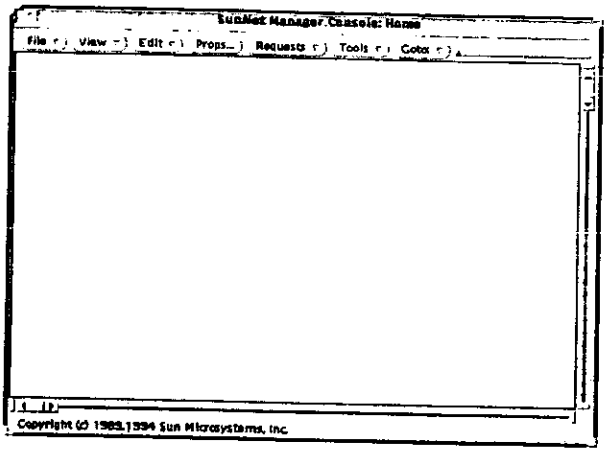


figura 6.4 Consola SunNet Manager

Después de desplegarse la consola, se puede empezar a crear el mapa de la red administrada. Para realizar dicha tarea se tendrá que utilizar el submenú **Edit➤Create**, al cual se tiene acceso desde el menú principal de la consola. Por este medio **SunNet Manager** nos permite crear elementos de diferentes tipos para representar vistas, redes, **workstations**, servers, conexiones, dispositivos de interconexión, etc. e ir creando el mapa de la red administrada.

Cuando se utiliza el submenú **Edit➤Create**, la ventana *create object* aparece.

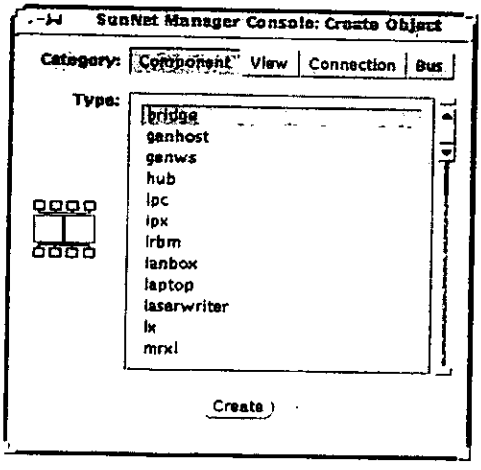


figura 6.5 ventana "Create Object"

Desde esta ventana, se puede seleccionar que tipo de elemento se desea crear, los tipos que **SunNet** tiene definidos son: Componentes, vistas, conexiones, y buses. Dentro del tipo componentes existen elementos como **workstations**, **servers**, dispositivos de interconexión, etc. dentro del tipo vistas se definen glifos que representan edificios, redes, etc., en los tipos conexión y buses, se encuentran definidos diferentes patrones de conexión por ejemplo, un cable RS232. Después de seleccionar el tipo de elemento, aparecerá una lista de opciones existentes para el tipo de elemento seleccionado, (en la figura 6.5 se muestra la selección componente y como se puede observar hay una lista de elementos pre definidos) se deberá elegir el que más se apegue al elemento que deseamos crear.
Una vez seleccionado el elemento que necesitamos aparecerá la ventana de propiedades, la cual nos permitirá definir las propiedades especificas del elemento que se esta creando. En esta ventana se puede definir: nombre del elemento, dirección IP, contacto (administrador de la maquina), localización física, descripción, *read-comunity-name* y *write-comunity name*, agentes para administrar al elemento, y color del glifo que lo representará. A continuación se muestra una ventana de propiedades para un elemento de tipo componente:

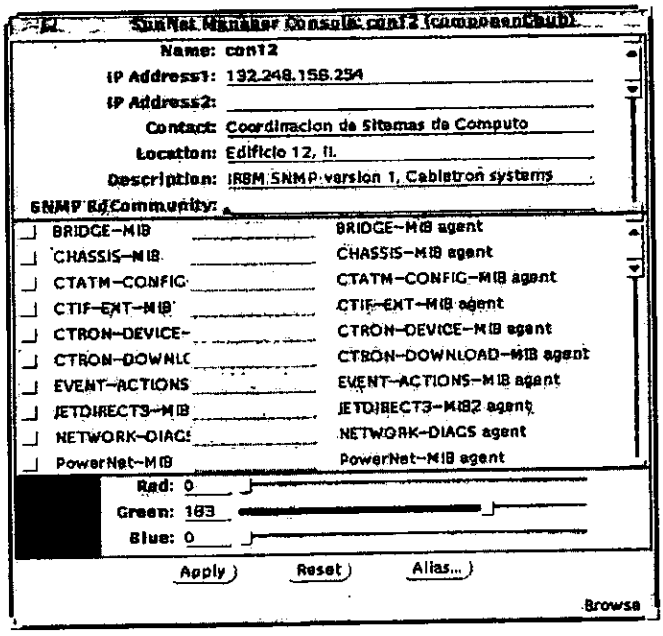


figura 6.6 ventana de propiedades en la creación de un elemento

De esta manera se va creando el mapa y los elementos (nodos), que lo conforman.
El mapa que describe a REDII, fue creado siguiendo los pasos mencionados anteriormente, como resultado obtuvimos un mapa con vistas jerárquicas que se apega a la distribución física de la red en los edificios del Instituto de Ingeniería. A continuación se muestra el mapa de REDII, con algunas de sus vistas jerárquicas.

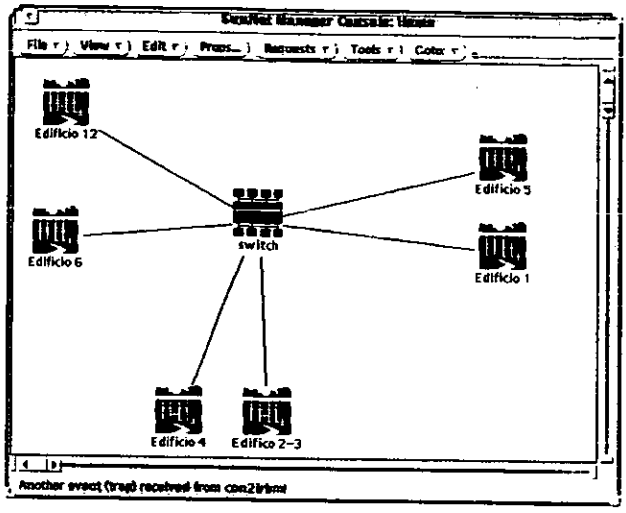


Figura 6.7 Mapa de REDII en la consola SunNet Manager

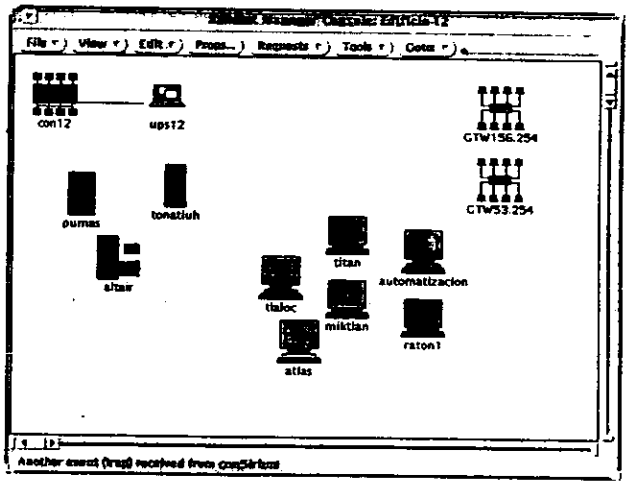


Figura 6.8 vista jerárquica del mapa de REDII (edificio 12)

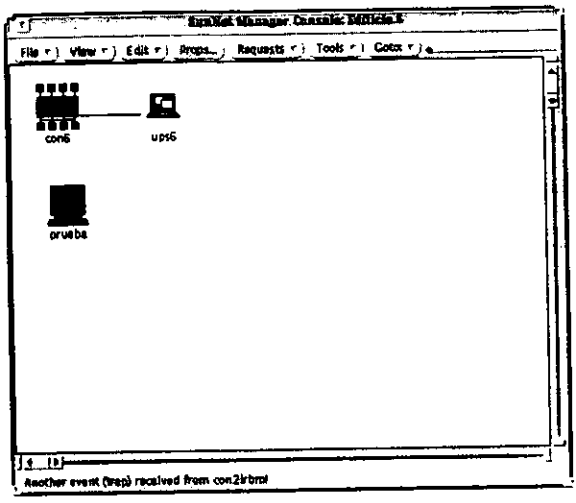


Figura 6.9 vista jerárquica del mapa de REDII (edificio 6)

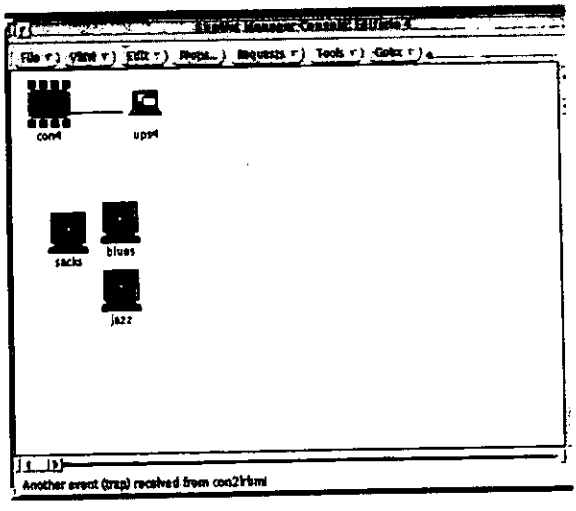


Figura 6.10 vista jerárquica del mapa de REDII (edificio 4)

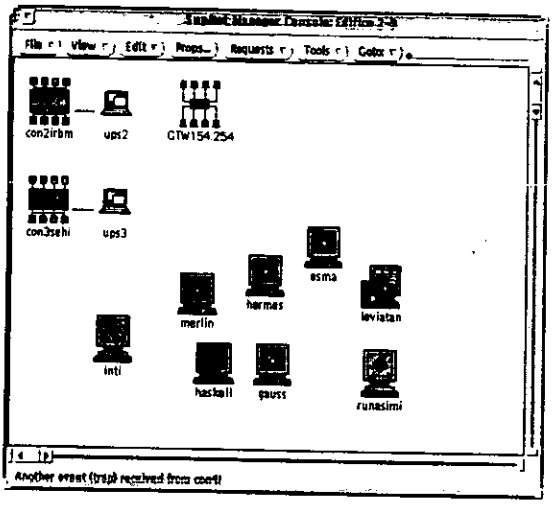


Figura 6.11 vista jerárquica del mapa de REDII (edificio 2-3)

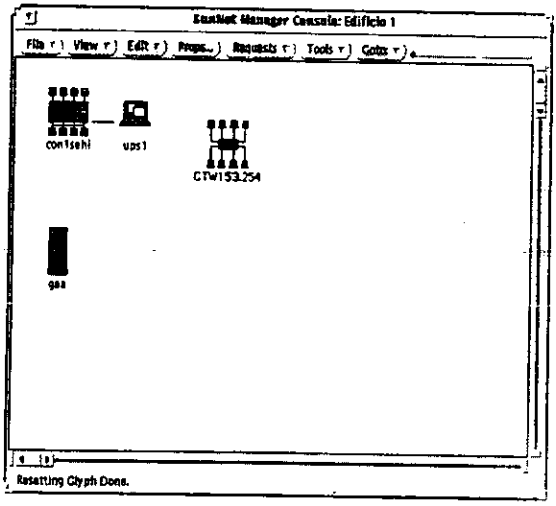


Figura 6.12 vista jerárquica del mapa de REDII (edificio 1)

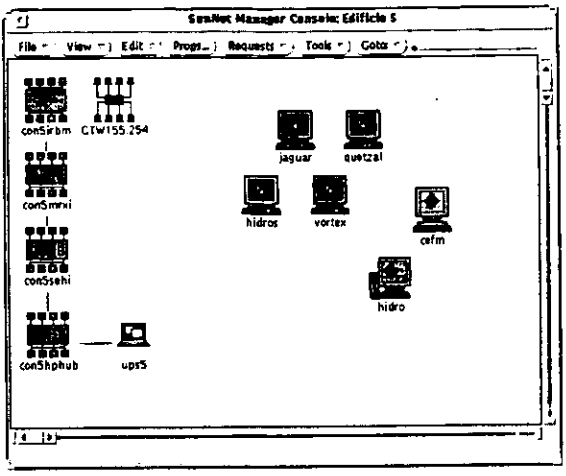


Figura 6.13 vista jerárquica del mapa de REDII (edificio 5)

6.4.1.3 Puesta en operación

La puesta en operación de **SunNet Manager** es muy sencilla, consiste en definir apropiadamente las peticiones de monitoreo a los nodos administrados. Sin embargo para esta implantación fue necesario agregar MIBs para poder monitorear ciertos dispositivos que no estaban pre-definidos, además fue necesario realizar **scripts** que interactúan con las peticiones de **SunNet**, con el objeto de adecuar tareas de administración a situaciones especificas.
Para que pueda comprenderse mejor como fueron agregadas las diferentes **MIBs** que se necesitaron, empezaremos por explicar como funciona el agente **SNMP** incluido **SunNet Manager 2.2.2**

Operación del agente SNMP incluido en SunNet Manager 2.2.2

SunNet Manager provee a un agente **proxy** que soporta **SNMP**, este agente permite obtener información de administración así como configurar atributos de dispositivos que son administrados vía **SNMP**. Este agente corre en estaciones de trabajo **Sun** (Si es el nodo administrador o cualquier otro nodo dentro de la red administrada)

Las aplicaciones de administración de **SunNet Manager** (ejemplo: la consola), se comunican con el agente **SNMP** usando el mismo protocolo basado en RPC, como lo hacen todos los demás agentes que incluye SunNet. El agente **SNMP** de **SunNet** se comunica con otros dispositivos **SNMP** usando las definiciones estándar de protocolo que se precisan en el RFC 1157. A continuación se muestra la manera en que la consola SunNet, puede comunicarse con un dispositivo **SNMP** a través del agente **SNMP** de SunNet.

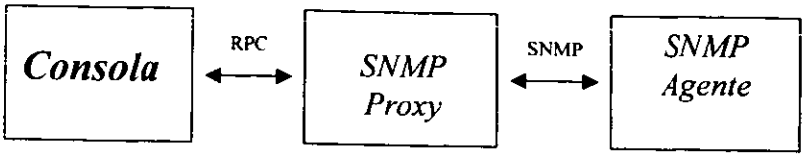


Figura 6.14 Comunicación Consola - dispositivo SNMP vía agente proxy SNMP de SunNet Manager

El agente **SNMP** de **SunNet** permite el manejo de cualquier **MIB SNMP**, dentro de cada **MIB** se puede definir y manipular objetos o enterprise-specific-objects, con el objeto de administrar de formar particular ciertos dispositivos.

El agente **SNMP** de **SunNet** utiliza un archivo **schema** para hacer un mapeo de los objetos definidos en una **MIB** a objetos cuya definición sea entendida por las aplicaciones **SunNet** Manager. En otra palabras el archivo tipo **schema** que utiliza el agente **proxy SNMP** es una representación de la **MIB** usando definiciones que sean entendidas por las aplicaciones **SunNet**.

SunNet Manager provee cuatro archivos **schema SNMP**:

- *snmp-schema*, describe la **MIB I**, definida en el RFC 1156
- *snmp-MIBII.schema*, describe la **MIB II**, definida en el RFC 1213
- *snmpv2-MIBII.schema*, describe la **MIB II**, para la versión SNMPV2
- *sun-snm.schema*, describe la **MIB II** definida en el RFC 1213, pero con extensiones para monitorear estaciones de trabajo **Sun**.

La siguiente figura muestra como interactúan la consola, el agente **proxy SNMP** incluido con **SunNet** y un agente **SNMP** en un cierto dispositivo, consultado al archivo **schema** y a la **MIB** respectivamente.

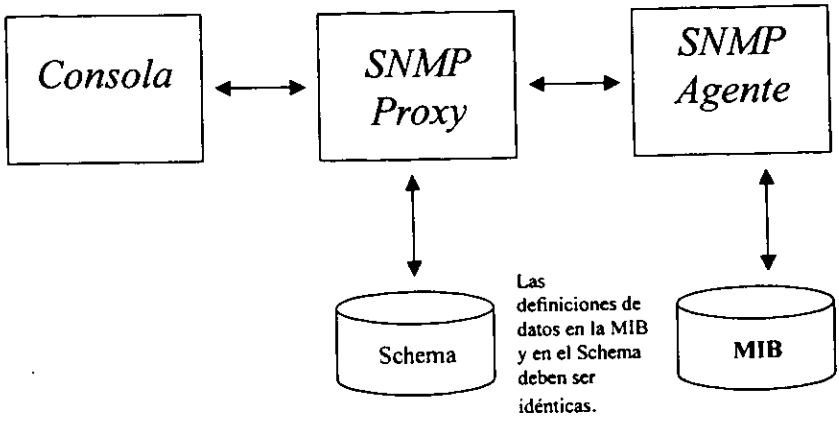


Figura 6.15 Agente SNMP de SunNet Manager y archivo schema.

Traps

Los agentes **SNMP** pueden generar reportes no solicitados o no esperados llamados ***traps***. El demonio que se encarga de manejar los **traps** (*na.snmp-trap*), es instalado junto con los demás agentes, este demonio se encarga de traducir el trap recibido a una forma que pueda ser interpretada por **SunNet Manager**, entonces lo envía al demonio **event dispatcher** para que el **trap** sea manejado como un evento más. Muchas **MIB** incluyen su propia definición de **traps**, es necesario entonces realizar la traducción de estas definiciones para que cada **trap** que la **MIB** maneje sea interpretado apropiadamente por **SunNet Manager**.

La traducción de **MIBs** y **traps** en definiciones que **SunNet Manager** pueda entender y manipular (archivo **schema**), se realizar a través de un programa incluido en la versión de **SunNet Manager** que se este utilizando. Para traducir **MIBs** definidas para **SNMP** se utiliza el programa MIB2schema, para traducir **MIBs** definidas para **SNMPv2** se utiliza v2MIB2schema. Un ejemplo del uso del programa MIB2schema se muestra a continuación.

```
hosts% MIB2schema cabletron-mrxi.mib
```

Si alguna definición en la **MIB** no es correcta, MIB2schema saldrá de ejecución reportando un error y no realizando la traducción, entonces se tendrá que revisar la sintaxis de la **MIB** que se esta traduciendo. Una vez finalizada la traducción de la **MIB**, los traps serán traducidos y se generará un archivo de traps que tendrá que ser adicionado al **archivo** de definiciones de **traps** de **SunNet Manager**.

Uno de los tipos de datos soportado por **SunNet Manager** es el identificador de objeto (***Object identifier OID***), este tipo de dato esta envuelto en las definiciones de cada **MIB**, dado que es un número (por ejemplo 1.2.3.4.5), una base de datos es manejada por **SunNet Manager** para que la consola pueda desplegar cada **OID** usando una cadena más sencilla y manejable, para el usuario. La base de datos **OID** provee la información que necesita la consola para realizar el mapeo entre **OIDs** y cadenas manejables. Al agregar una nueva **MIB** al grupo de schemas, nuevos **OIDs** aparecerán y será necesario agregarlos a la base de datos **OID**, existe una herramienta para realizar esta función: build_oid. Esta herramienta deberá ser ejecutada terminado la traducción de cada **MIB**, su ejecución es muy simple, solo basta con correr el comando build_oid y será el paso final en la traducción de una **MIB**.
Para nuestro esquema de monitoreo y administración fue necesario al adición de la siguientes **MIBs** para monitorear distintos dispositivos.

- Dispositivos de interconexión cabletron.
  cabletron-traps.txt
  actions-mib.txt
  ctbridge-mib.txt
  ctdevice-mib.txt
  ctdownload-mib.txt
  ctmib2-ext-mib.txt
  chassis-mib.txt
  csi-irbm.txt
  csi-mrxi.txt

- **UPS APC**
  powernet.mib

Como administrar REDII desde SunNet Manager

La consola **SunNet Manager** provee cuatro métodos para obtener información de administración de los agentes:

- ***Quick Dumps***. Son reportes de datos de un solo intervalo de tiempo, es decir, suceden una sola vez. Estos reportes entregan toda la información de un grupo de atributos definido en una **MIB** dentro de un agente.
- *Reportes de datos*. Estos reportes entregan el valor de atributos seleccionados sobre un período de tiempo dado. Este tipo de reporte es útil para realizar análisis comparativos o para observar tendencias en el comportamiento de los elementos monitoreados.
- *Reportes de eventos*. Estos reportes realizan una notificación de los cambios en los valores de ciertos atributos basados en condiciones pre-definidas (valores de umbral). Este tipo de reporte es útil para reportar condiciones anormales en un elemento monitoreado que puedan generar una falla en el mismo.
- *Traps*. Como se definió anteriormente, los traps son reportes informativos no solicitados que se generan cuando una condición dada se presenta en un elemento monitoreado.

Los tres primeros métodos (**Quick Dump,** reporte de datos y eventos), se basan en el método de **polling** para obtener información. Los **traps** utilizan el método de reporte de eventos.

En la configuración de **SunNet Manager** que se implanto para el Instituto de Ingeniería, los reportes de eventos y datos son los más utilizados para obtener información de administración en los nodos monitoreados. A continuación se muestra la manera en que se configuran dichos reportes.

Desde la consola cuando se elige la opción **Request➢Send Request** sobre un elemento especifico, aparece la ventana ***Request Builder*** en donde es posible seleccionar el tipo de reporte (datos o eventos) que deseamos crear, además de seleccionar el agente y el grupo de atributos al que se le va hacer la petición. En la siguiente figura se muestra la ventana ***Request Builder***

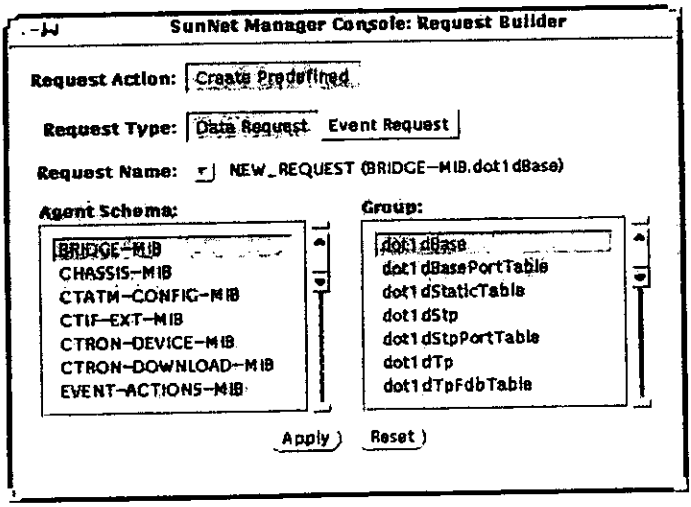


figura 6.16 Ventana Request Builder

Cuando en la ventana ***Request Builder*** se elige crear un reporte de datos, la ventana ***Data Request*** aparece, en esta ventana es posible definir los parámetros necesarios para configurar un reporte de datos.

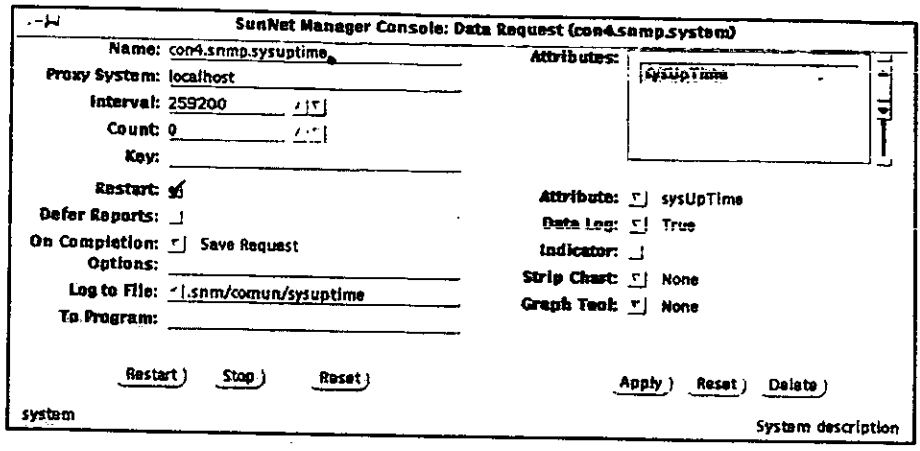


Figura 6.17 Ventana Data Request

Los campos en el lado izquierdo de la ventana **Data Request**, especifican las caracteristicas del reporte. Cada campo será descrito a continuación:

Name: En este campo se puede asignar un nombre opcional a la petición de datos. Si este campo no es llenado, un nombre es asignado automáticamente, usando el siguiente formato.

<agente>.<grupo>.<numero> (ejemplo, snmp.system.0)

Proxy System: El nombre del sistema al que el reporte va a ser enviado.

Interval: Especifica el intervalo de tiempo en segundos sobre el cual se llevara a cabo la petición de datos.

Count: Especifica el número de veces que se enviará la petición de datos. Un contador (**Count**) con valor cero especifica que la petición se realizará una vez por intervalo hasta que dicha petición sea parada.

Key: Identifica un renglón en una tabla especifica dentro de la **MIB**, lo cual permite monitorear solamente un objeto dentro del nodo administrado (Ejemplo, la tarjeta le0 dentro de una estación de trabajo).

Restart: Indica que la petición de datos deberá ser re iniciada, cuando el agente termina su operación inesperadamente, la consola se re inicia, o si nodo al que se le esta haciendo la petición ha tenido un **reboot**.

Defer Reports: Si este campo esta habilitado, el agente recibe la orden de colectar los datos y mantenerlos en "cache", hasta que el usuario a través de la consola le da la orden de enviar los datos almacenados.

On completion: Si este campo esta habilitado, la petición seguirá dada de alta en el resumen de peticiones aun cuando esta haya terminado.

Options: En este campo se puede especificar opciones para los agentes que las acepten.

Log to File: Especifica el archivo donde los datos de los reportes serán almacenados.

To program: En este campo se puede especificar, un programa que será ejecutado cada que se obtengan datos, estos datos serán enviados a la entrada estándar del programa.

Los campos en la lado derecho de la ventana **Data Request**, son usados para especificar atributos.

Attributes. Lista que contiene los atributos a los que se les puede hacer una petición de datos.

Data log. Especifica los datos obtenidos deberán escribirse en el log de datos de **SunNet Manager**.

Indicator. Especifica si los datos obtenidos deberán desplegarse como indicador. Un indicador muestra el ultimo valor reportado para un atributo en particular.

Strip Chart. Especifica si los datos obtenidos deberán desplegarse en un **Strip Chart**. Un **Strip Chart** es una gráfica pequeña la cual ajusta su escala a los valores del atributo que esta siendo monitoreado. Esta gráfica aparece junto a el nodo administrado.

Graph Tool. Especifica si los datos obtenidos deberán enviarse a **Graph tool**, para que sean graficados.

Desde la ventana **Request Builder** mostrada en la figura 6.13, también se puede elegir crear un reporte de evento, en ese momento aparecerá la ventana **Event Request**.

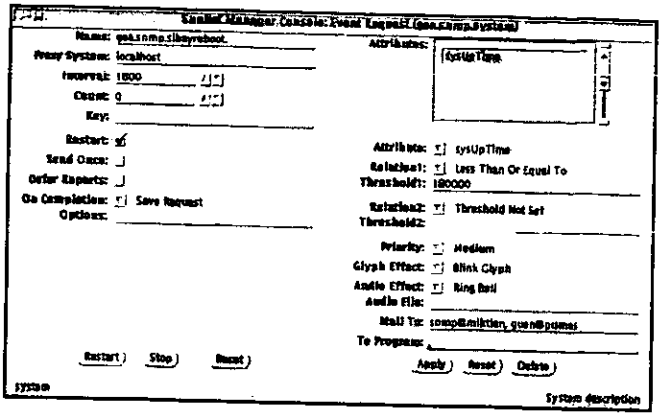


Figura 6.18 Ventana Event Request

Los campos de lado izquierdo de esta ventana son los mismos que los que fueron descritos para la ventana **Data Request**, en los campos de lado derecho también se elige al atributo monitoreado además de aparecer nuevos campos. Los campos Relation1-Threshold1, Relation2-Threshold2, definen los valores de umbral que se deben presentar para que se pueda generar un evento. También del lado derecho, se define la manera en que el evento será presentado en la consola **SunNet Manager**, la prioridad que el evento tiene, si cuando se presente tendrá efecto audible, si cuando se presente el evento deberá de notificarse a alguien vía e-mail la presencia del mismo y si cuando se presente el evento se ejecutará un programa.

SunNet Manager, también provee la posibilidad de tener control sobre los nodos administrados, a través del cambio de los valores de los atributos en dichos nodos. Este cambio se lleva a cabo desde la consola por medio de la herramienta ***set tool.***

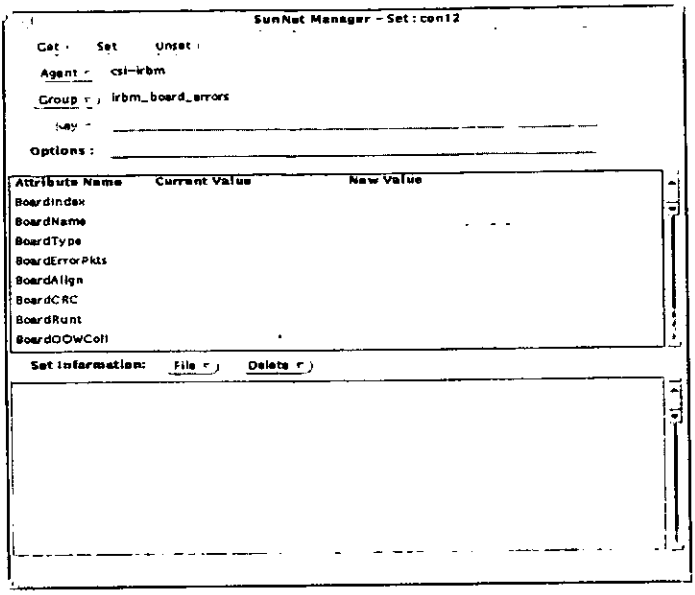


Figura 6.19 Ventana Set tool.

Como la figura muestra, desde la ventana **Set tool**, se puede definir el agente y el grupo de atributos que se pueden modificar, el proceso es muy sencillo, solo basta con definir los nuevos valores y aplicar la operación de cambio. Esta herramienta nos permite tener un control en nuestros dispositivos de manera remota, por ejemplo, en un dispositivo de interconexión se desea deshabilitar un puerto, para restringir el acceso a la red de una determinada maquina, desde esta herramienta se puede realizar esta tarea, cambiando el atributo de disponibilidad para dicho puerto.

A continuación se mostraran las definiciones de reportes de eventos y reportes de datos que se llevaron a cabo para monitorear REDII.

El primer reporte que se mostrara, es un reporte de eventos, el cual fue definido para verificar conectividad y fue aplicado para servidores, dispositivos de interconexión y **Gateways** de REDUNAM.

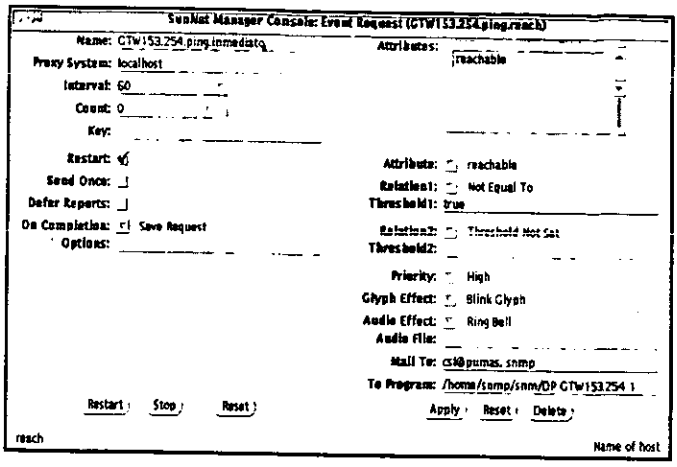


Figura 6.20 Reporte de evento definido para verificar conectividad con los Gateways de REDUNAM

Un caso especial se presenta al aplicar el reporte de eventos anterior al servidor de DNS. Con el fin de verificar la existencia de una falla en la funcionalidad del servidor de DNS cuando una falla de conectividad se presenta se incluyo la ejecución de un programa que verifica que el servidor de DNS este funcionando adecuadamente a través de una prueba petición - respuesta. La definición de este reporte de eventos se muestra a continuación.

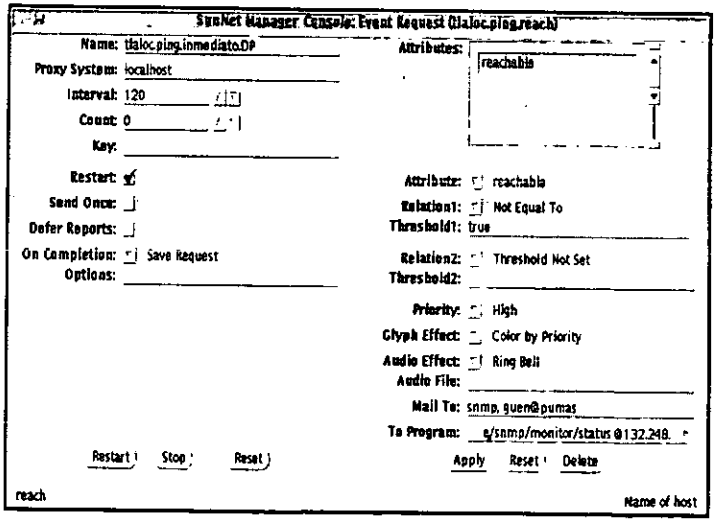
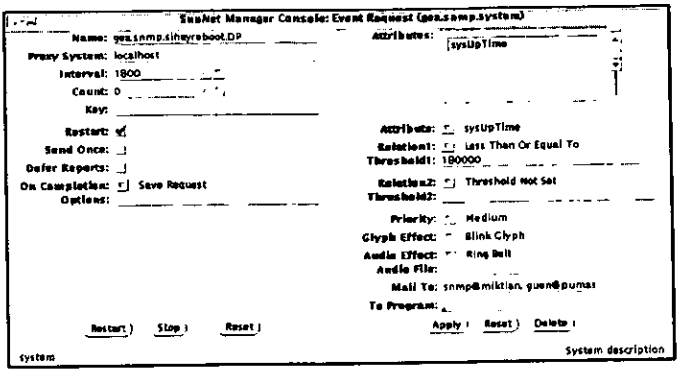


Figura 6.21 Reporte de evento definido para verificar la funcionalidad del servidor DNS

El siguiente reporte también es un reporte de eventos, fue definido para detectar procesos de reboot en los nodos administrados. Se aplico en servidores y dispositivos de interconexión.



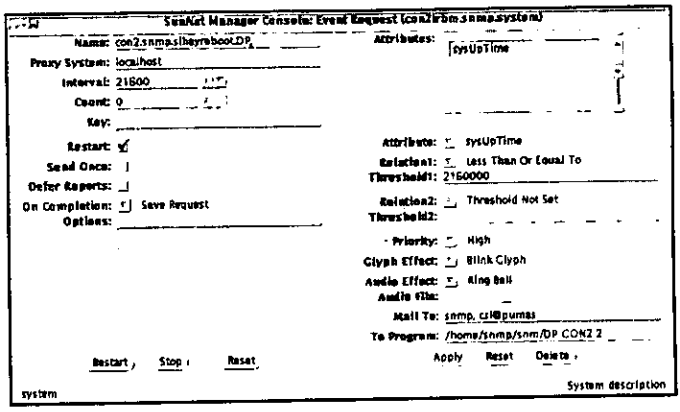


Figura 6.22 y 6.23 Reportes de eventos que detectan procesos de reboot

El siguiente reporte de eventos, fue definido para monitorear la carga de los **UPS**, cuando la carga baja al 15%, se genera un evento con la finalidad de advertir que el **UPS** esta próximo a descargarse totalmente

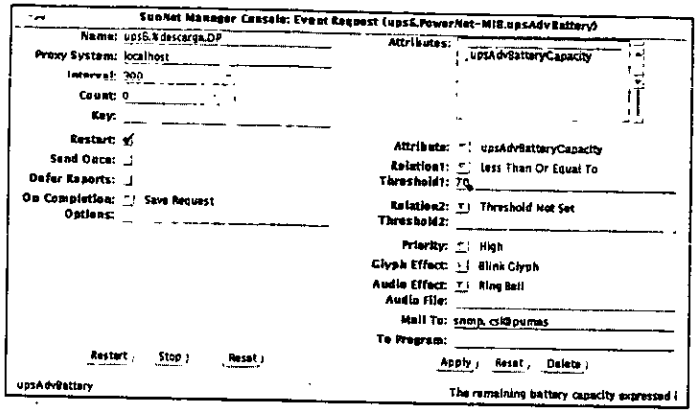


Figura 6.24 Reporte de eventos que monitorea la carga en los UPS

El siguiente reporte de eventos, fue definido para monitorear la utilización de los sistemas de archivos en los servidores. Cuando un sistema de archivos se llena puede traer consecuencias graves en la operación de los servidores, más aun cuando se trata de sistemas de archivos donde el sistema operativo tiene espacios reservados. En este reporte de eventos se definieron valores umbral que marcan el limite aceptable en la utilización del sistema de archivos, cuando alguno de los sistemas de archivos pase del 95% de utilización se generará un evento.

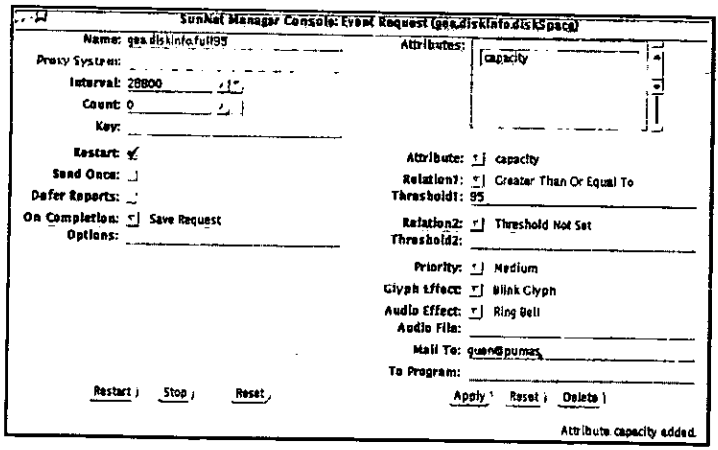


Figura 6.25 Reporte de eventos que monitorea la utilización de los sistemas de archivos

El siguiente reporte es un reporte de datos y monitorea cuanto tiempo ha estado arriba el sistema operativo de los nodos administrados. Este reporte se aplico a los servidores y los dispositivos de interconexión.

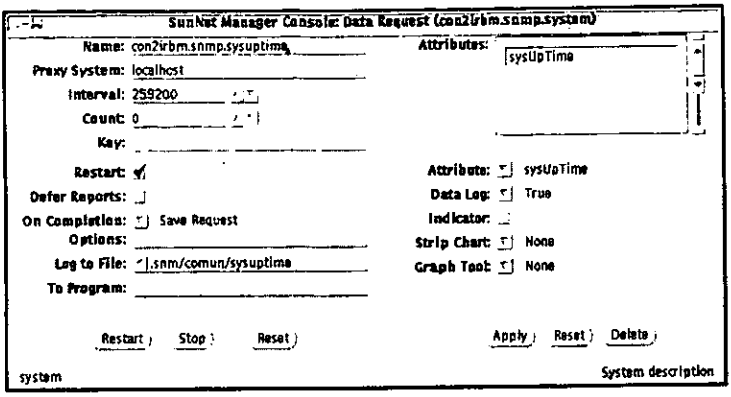


Figura 6.26 Reporte de datos que monitorea disponibilidad en los nodos administrados

El siguiente reporte de datos monitorea el porcentaje de colisiones que se presentan en los servidores.

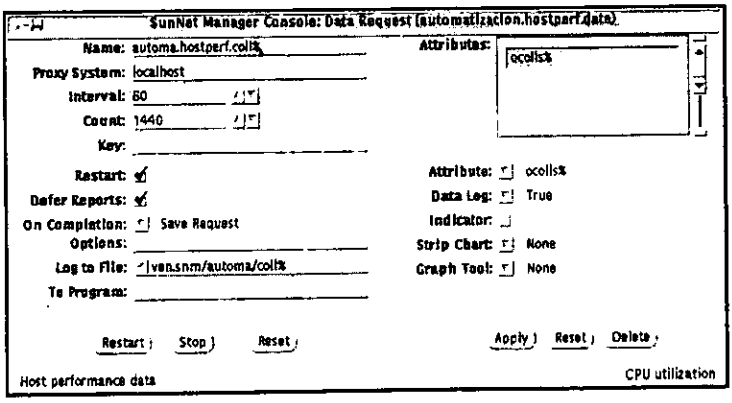


Figura 6.27 Reporte de datos que monitorea el porcentaje de colisiones en los servidores

El siguiente reporte de datos monitorea el porcentaje de colisiones y paquetes erróneos en las tarjetas de puertos de los dispositivos de interconexión.

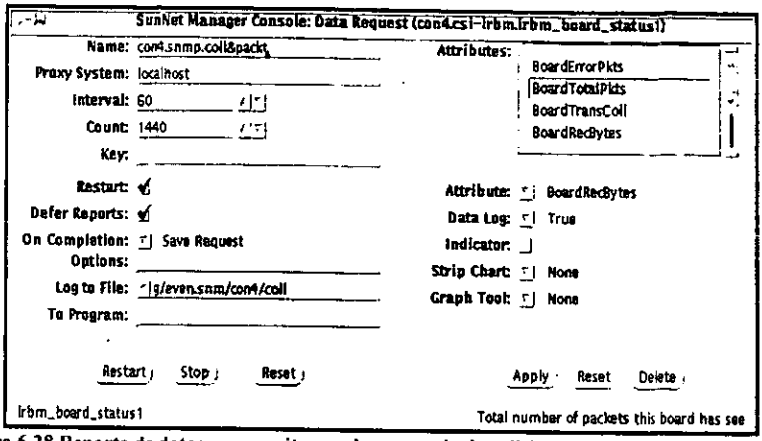


Figura 6.28 Reporte de datos que monitorea el porcentaje de colisiones y paquetes erróneos en los dispositivos de interconexión.

El siguiente reporte de datos monitorea el desempeño de los accesos de entrada y salida en los discos de los servidores.

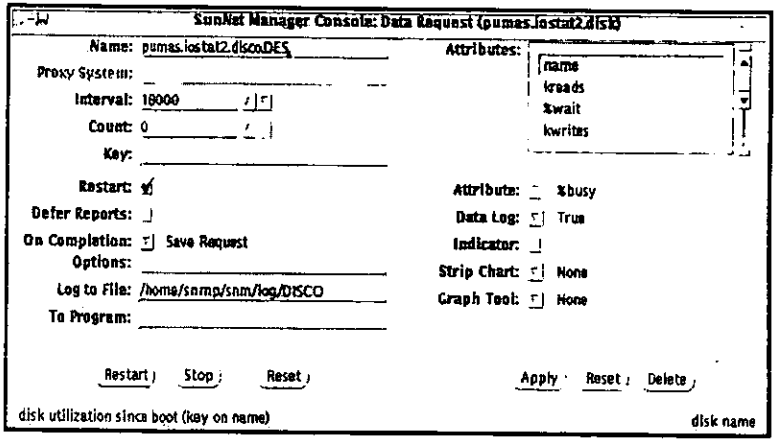


Figura 6.29 Reporte de datos que monitorea el desempeño de los discos

Al realizar el proceso de monitoreo **SunNet Manager** y otras aplicaciones, dejan la información recopilada en archivos log que se almacenan en el disco duro del nodo administrador, esto genera la necesidad de depurar dichos archivos periódicamente, este hecho nos obligo a establecer procedimientos de mantenimiento que debíamos aplicar al nodo administrador. Para realizar dicha tarea, se definieron reportes de eventos que monitorearan el estado de los sistemas de archivos en los que se guardan los archivos log , en estos reportes se definieron valores umbral, cuando alguno de estos valores es rebasado

se genera un reporte de evento, el cual a su vez ejecuta un programa que depura automáticamente los sistemas de archivos involucrados. A continuación se muestran las definiciones de los reportes de eventos y los programas que hacen la depuración.

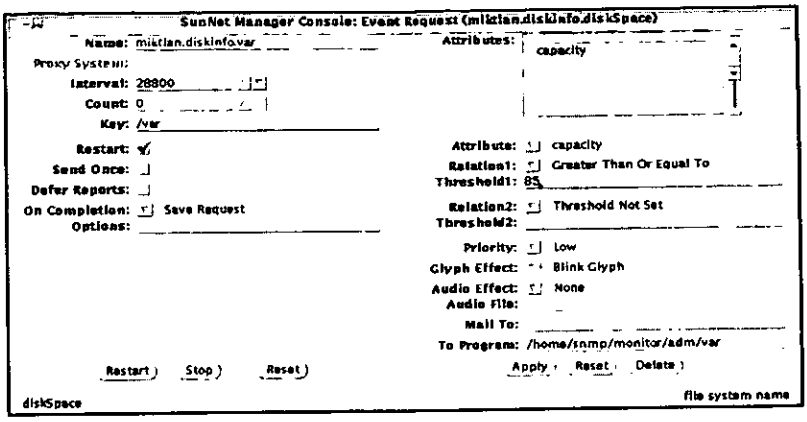


Figura 6.30 Reporte de eventos definido para depurar el sistema de archivos /var en el nodo administrador

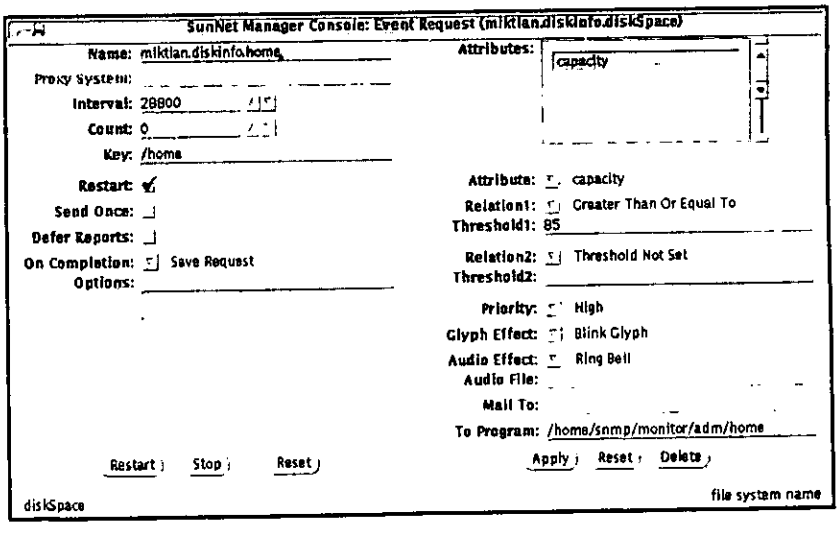


Figura 6.31 Reporte de eventos definido para depurar el sistema de archivos /home en el nodo administrador

Los programas que realizan la depuración, están hechos en **Shell**[7], en especifico **Born Shell**, en ellos se hace un copiado de los archivos log del nodo administrador hacia un servidor con más capacidad en disco y se guardan por fecha, posteriormente los archivos log son borrados del nodo administrador y re creados vacíos para que puedan seguir siendo utilizados. A continuación se presentan dichos programas.

```
#################################################################
### Shell script que realiza la depuración de /var en miktlan
### es ejecutado por SunNet Manager
#################################################################

#!/bin/sh
path1=/var/adm/sa
path2=/home/snmp/snm/log/even.symbel/miktlan
date=`date '+%d.%m.%y.%H:%M'`
cp $path1/monitor.log $path2/$date
> $path1/monitor.log
```

```
#############################################################
#### Shell script que realiza la depuración de /home en
#### miktlan, ejecutado por SNM
#############################################################

#!/bin/sh
path1=/home/snm/log
path2=/home/snmp/snm/log/even.snm
date=`date '+%d.%m.%y.%H:%M'`
cp $path1/event.log $path2/$date
> $path1/event.log
```

[7] Shell. Es el interprete de comandos de UNIX

6.4.2 Otros programas implementados

SunNet Manager, es un software de monitoreo y administración de red robusto, sin embargo, se tuvo la necesidad de implementar y elaborar otros programas con el objeto de complementar las funciones de **SunNet Manager**.

En primer termino se describirán los programas que fueron elaborados en la Coordinación de Sistemas de Cómputo, para satisfacer tareas especificas de monitoreo. Posteriormente se describirán el software de monitoreo que fue obtenido de Internet y se explicará su configuración.

6.4.2.1 Programas de monitoreo elaborados en la Coordinación de Sistemas de Cómputo.

Como se menciono anteriormente estos programas fueron creados para satisfacer necesidades especificas de monitoreo, que ni **SunNet Manager** ni ningún otro programa de dominio publico cubría.

Estos programas fueron hechos en **Born Shell**, aprovechando la facilidad de programación de este interprete de comandos, y también considerando que los programas en **Shell** son portables para cualquier ambiente **UNIX**.

El primero de estos programas, monitorea la disponibilidad del servidor de **DNS** a través de un programa de dominio público llamado **DIG**.

DIG (Domain Information Gopher), es una herramienta de línea de comando muy flexible, la cual es usada para obtener información de servidores de **DNS**. **DIG** es muy parecido al comando de **UNIX** *nslookup*, en este caso se opto por **DIG**, ya que su uso en línea de comandos es más sencillo y el tiempo de petición de información al servidor de **DNS** es más rápido. A continuación se muestra el programa en **Shell** para monitorear la funcionalidad del servidor de **DNS**.

```
#########################################################################
## Shell script, que monitorea disponibilidad de servidores de DNS
## a través de DIG (Domain Information Gopher)
#########################################################################

#!/bin/sh

/opt/USCdig/bin/dig $1 Pumas.iingen.unam.mx >/home/snmp/monitor/dns/res
grep NOERROR /home/snmp/monitor/dns/res >/home/snmp/monitor/dns/vl
if [ -s /home/snmp/monitor/dns/vl ]; then
  echo " * `date` * Status DNS SERVER  $1 OK> " >>/home/snmp/monitor/dns/reporte
else
  /usr/ucb/Mail guen@Pumas guen@titan snmp@miktlan </home/snmp/monitor/dns/men
  echo " * `date` * Status DNS SERVER $1 NO OK> " >>/home/snmp/monitor/dns/reporte
fi
```

Con este programa no solo se monitorea la disponibilidad del servidor de **DNS** del Instituto de Ingeniería, también se monitorean los dos principales servidores de **DNS** de REDUNAM.

El monitoreo para el servidor **DNS** de REDII se realiza de 10:00 a 20:00 con intervalos de 15 minutos, para los servidores de REDUNAM el monitorea se realiza en el mismo horario pero con intervalos de 20 y 30 minutos. Para la ejecución periódica de este programa se utilizo el comando de **UNIX** *cron* como muestra a continuación la salida del comando *crontab -l*.

```
0,15,30,45,59 10,12,14,16,18,20 * * * /home/snmp/monitor/dns/status @tlaloc
0,20,40,59 10,12,14,16,18,20 * * * /home/snmp/monitor/dns/status @132.248.10.2
0,30,59 10,12,14,16,18,20 * * * /home/snmp/monitor/dns/status @132.248.204.1
```

El segundo programa desarrollado tiene como tarea detectar la existencia de direcciones IP duplicadas. Este programa esta hecho también en Born Shell, utiliza los comandos **UNIX** "ping" y "arp" para detectar los nodos activos de la red, asocia la dirección IP con la dirección MAC de cada uno de dichos nodos y posteriormente valida estas direcciones en una base de datos preestablecida. A continuación se muestra el programa en **shell** que detecta direcciones IP duplicadas en REDII.

```
##############################################################################
##############################################################################
###  Programa que detecta direcciones IP duplicadas.
###

#!/bin/sh

#### Definicion de Constantes ###############
valor="1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25
26 27 28 29 30 31 32 33 34 35 36 37 38 39 40 41 42 43 44 45 46 47 48 49
50 51 52 53 54 55 56 57 59 59 60 61 62 63 64 65 66 67 68 69 70 71 72 73
74 75 76 77 78 79 80 81 82 83 84 85 86 87 88 89 90 91 92 93 94 95 96 97
98 99 100 101 102 103 104 105 106 107 108 109 110 111 112 113 114 115 116
117 118 119 120 121 122 123 124 125 126 127 128 129 130 131 132 133 134
135 136 137 138 139 140 141 142 143 144 145 146 147 148 149 150 151 152
153 154 155 156 157 158 159 160 161 162 163 164 165 166 167 168 169 170
171 172 173 174 175 176 177 178 179 180 181 182 183 184 185 186 187 188
189 190 191 192 193 194 195 196 197 198 199 200 201 202 203 204 205 206
207 208 209 210 211 212 213 214 215 216 217 218 219 220 221 222 223 224
225 226 227 228 229 230 231 232 233 234 235 236 237 238 239 240 241 242
243 244 245 246 247 248 249 250 251 252 253 "
valor2=" 53. 154. 155. 156. "
vivo=alive

date >>reporte
echo " Programa verificador de IP's y MacS"
echo " Elige la opcion que deseas realizar"
echo " "
echo " "
echo " 1) Monitoreo de Ip's en maquinas vivas "
```

```
echo " 2) Busqueda de direcciones Ip para encontrar una Mac y viceversa
que ya esten dentro de la base"
echo " "
echo " "
echo " opcion>>>>> \c"
read opcion
if [ $opcion = 1 ]; then
  echo " escoje la subred que deseas monitorear"
  echo " "
  echo " 1) 132.248.53.X "
  echo " 2) 132.248.154.X "
  echo " 3) 132.248.156.X"
  echo " 4) Toda la red "
  echo " "
  echo " opcion >> \c"
  read opcion2
  case $opcion2 in
     1) red=53.;;

     2) red=154.;;

     3) red=156.;;

     4) red=777;;

  esac
  if [ $red = 777 ]; then
    for i in $valor2
      do
         for j in $valor
           do
             ping 132.248.$i$j 10 |cut -d' ' -f3 >valida
           datos=`more valida`
             if [ $datos = $vivo ]; then
               grep 132.248.$i$j base >valida
            if [ -s valida ]; then
              veri1=`arp 132.248.$i$j |cut -d' ' -f4`
              veri2=`grep 132.248.$i$j base |cut -d' ' -f2`
                  if [ $veri1 = $veri2 ]; then
                   echo " REPORTE: la direccion 132.248.$i$j esta O.K con
su direccion MAC" >>reporte
                  else
                   echo " ERROR: la direccion 132.248.$i$j tiene una
direccion MAC DIFERENTE!!" >>reporte
                   echo "        la direccion MAC que deberia tener es:
$veri2, y la que tiene es:$veri1" >>reporte
                Mail snmp@miktlan <mesaje
                Mail csl@Pumas <mesaje
               fi
            else
               arp 132.248.$i$j |cut -d' ' -f1,4 >>base
            fi

             fi

           done
      done
  else
     for j in $valor
       do
            ping 132.248.$red$j 10 |cut -d' ' -f3 >valida
```

```
              datos=`more valida`
              if [ $datos = $vivo ]; then
                grep 132.248.$red$j base >valida
             if [ -s valida ]; then
               veri1=`arp 132.248.$red$j |cut -d' ' -f4`
               veri2=`grep 132.248.$red$j base |cut -d' ' -f2`
                   if [ $veri1 = $veri2 ]; then
                    echo " REPORTE: la direccion 132.248.$red$j esta O.K
con su direccion MAC" >>reporte
                   else
                    echo " ERROR: la direccion 132.248.$red$j tiene una
direccion MAC DIFERENTE!!" >>reporte
                    echo "         la direccion MAC que deberia tener es:
$veri2, y la que tiene es:$veri1" >>reporte
                 Mail csl@Pumas <mesaje
                 Mail snmp@miktlan <mesaje
               fi
             else
                arp 132.248.$red$j |cut -d' ' -f1,4 >>base
             fi

              fi

       done

  fi
  # fin del if red 777

else
   echo " Dame la direccion (MAC o IP ) que deseas buscar "
   echo " "
   echo " Direccion \c"
   read direccion
   grep $direccion base >valida
   if [ -s valida ]; then
      echo " En la base IP-MAC se encontro el siguiente resultado para tu
peticion: "
      echo " "
      echo " ***"
      more valida
      echo " ***"
      echo " "
   fi

fi
```

6.4.2.2 PowerNet SNMP Adapter 2.2

Este producto permite el monitoreo de los **UPS**, a través de un adaptador que contiene el agente **SNMP**. Este producto no es de dominio público, es un producto comercial desarrollado por la compañía **APC (American Power Conversion)**.

Debido a la cada vez más fuerte aceptación de **SNMP** como el protocolo de monitoreo de RED estándar, **APC** desarrollo un dispositivo que soportara **SNMP** para sus **UPS** (Smart-**UPS** y Matrix **UPS**). El adaptador esta basado en un procesador Intel x 86 e incorpora el agente **SNMP** para hacer monitoreable los **UPS APC**, el adaptador es un dispositivo externo al **UPS**, en los nuevos modelos de **UPS** contienen este adaptador internamente.

Los requerimientos del adaptador son los siguientes:

- Un ambiente de red **TCP / IP**
- Una conexión de red 10 Base-T
- Un nodo administrador basado en **SNMP**
- **UPS APC**
- Una terminal o una PC con emulación de terminal para configurar el software del adaptador

Instalando y configurando el adaptador

La instalación del adaptador esta dividida en las siguientes secciones:

Instalación física del adaptador
- Determinar la dirección IP y la máscara de la subred para el adaptador
- Instalar el **UPS**
- Conectar el adaptador al **UPS**

Configurar e iniciación del adaptador
- Configurar la terminal o la PC con la emulación de terminal
- Apagar y prender el adaptador para que el proceso de inicialización del mismo de comienzo
- Establecer comunicación con el adaptador vía terminal
- Entrar al menú de configuración básica y configurar la dirección IP
- Definir *el read-comunity-name* y el *write-comunity-name* del adaptador
- Especificar los nodos que están habilitados para recibir los **traps** que el adaptador genere.
- Arrancar el agente en el adaptador a través del menú principal

Configurar el nodo administrador

- Agregar la dirección IP del adaptador al archivo /etc/hosts del nodo administrador
- Cargar la **MIB PowerNet** en el nodo administrador, esta **MIB** permitirá el monitoreo **SNMP** del adaptador
- Adicionar un objeto que represente al adaptador en el mapa de la red administrada en el nodo administrador.

6.4.2.3 HTTP-ANALYZE 2.0

Http-analyze 2.0 es un programa de dominio público (también existe un versión que se puede comprar), el cual analiza el archivo log generado por servidores de **Web** y crea estadísticas detalladas de acceso y las gráfica perfectamente.

El **http-analyze**, soporta todos los formatos comunes de archivos log (**Common Logfile Format CLF, Combined Logfile Format DLF, Extended Logfile Format ELF**) de los más populares servidores de **Web** como son **Netscape Enterprise Server, NCSA httpd, Apache, Ximati** y muchos otros.

Esta versión de **http-analyze** ha sido altamente optimada para procesar archivos de log muy grandes a máxima velocidad en un mínimo de tiempo.

Existen 2 modos de operación con diferentes niveles de detalle en las estadísticas que se generan:

- *Estadísticas cortas*. En este modo **http-analyze** genera un pequeño resumen del uso del servidor por día, y los reportes son generados en muy poco tiempo
- *Estadísticas totales*. En este modo **http-analyze** genera un reporte completo mensual a detalle.

Todos los reportes que genera **http-analyze** son generados en **HTML**.

Http-analyze, es uno de los analizadores de log de **WWW** en Internet, por sus excelentes gráficos y su rápido procesamiento de información.

Instalación

La distribución de **http-analyze** para **UNIX**, esta comprimida con formato *gzip* y *tar* , se deberá descomprimir el software de la siguiente manera:

```
# gzip http-analyze2.01-sol.tar.gz | tar xvf -
```

Al descomprimir la distribución de **http-analyze**, se genera un directorio que contiene la documentación del producto y su instalación, y los códigos fuentes del producto.

Para compilar a **http-analyze**, es necesaria la biblioteca GD para la creación de los gráficos GIF. Antes de compilar **http-analyze** se debe instalar dicha biblioteca creada por Thomas Boutell.

Cuando **GD** esta debidamente instalada en el sistema, se puede comenzar con la compilación de **http-analyze**, para realizar dicha tarea, se deben definir las opciones de configuración apropiadas en los archivos config.h y Makefile, por ejemplo se define la ruta de la biblioteca **GD**, el tipo de compilador y las opciones de compilación, etc. Posterior a estas definiciones se ejecuta el comando make para realizar la compilación.

La compilación generará los archivos binarios de **http-analyze**, los cuales deberán ser colocados en /usr/local/bin , o en donde el usuario establezca.

6.4.2.4 GWFstats 1.1

GWFstats, es un programa de dominio público, el cual fue desarrollado en **Perl** por Jens Elker; este programa, fue desarrollado en base a los programas **FTPWeblog** 1.0.2 y **GraphFTPWeblog** 1.0 desarrollados por Benjamin Snowhare Franz.

GWFstats analiza los archivos log que dejan los servidores de **WWW** y **FTP**, en el Instituto de Ingeniería este programa es aplicado sobre los archivos log del servidor de **FTP**. El programa **GWFstats**, es capaz de generar reportes por día, mes y año, presentándolos textualmente o de manera gráfica; tiene la capacidad de almacenar los reportes de manera histórica. Los reportes gráficos son generados en formato HTML.

Instalación

Para ejecutar el programa **GWFstats**, son requeridos los siguientes paquetes:

- Se debe contar con **Perl** 5.001 o mayor para generar reportes gráficos, para los reporte tipo texto es necesario al menos **Perl** 4.036.
- GD.pm de Lincoln Stein, es la interfaz en **Perl** de la biblioteca GD
- El archivo *Country-Code*, que contiene una traducción de las siglas usadas en Internet para identificar países, ejemplo mx - México

Ya que **GWFstats** esta desarrollado en **Perl**, no hay necesidad de compilarlo, pero para su correcta ejecución se deberán definir los siguientes parámetros:

- Definir en el programa **GWFstats** la ruta donde esta ubicado **Perl**
- Definir en el programa **GWFstats** la ruta donde esta ubicado el archivo log a analizar
- Identificar el directorio donde se guardaran los reportes generados por **GWFstats**

GWFstats, es un programa sencillo pero muy poderoso, puede manejar archivos *log* muy extensos, procesándolos en muy corto tiempo. Es muy fácil de utilizar y se puede adecuar fácilmente a los requerimientos del usuario.

6.4.2.5 Multi Router Traficc Grapher (MRTG)

MRTG es una herramienta para monitorear el trafico en nodos de red. **MRTG** genera paginas HTML que contienen gráficas las cuales proveen una representación del trafico por intervalos de tiempo.

MRTG esta hecho en **Perl** y en C y trabaja bajo **UNIX** o Windows NT. **MRTG** consiste básicamente en un programa **Perl** el cual usa **SNMP** para leer el trafico de los nodos en una red, posteriormente un programa en C procesa los datos colectados y genera las gráficas que representan el trafico monitoreado, estas gráficas son incluidas en una pagina **HTML**, la cual puede ser vista por cualquier *browser* de **Web**. Las gráficas que se generan representan el trafico por día, mes y año.

MRTG no esta limitado a monitorear trafico de red, puede monitorear cualquier variable **SNMP**, también se puede utilizar cualquier programa externo y los datos que genere pasarlos a **MRTG** para que juntos realicen el monitoreo, de esta manera **MRTG** puede monitorear la carga del sistema, uso de recursos del sistema por los usuarios, **sendmail**, etc.

Características

1. **MRTG** corre en las más importantes plataformas **UNIX** y Windows NT
2. Tiene una implementación portable de **SNMP**, escrita totalmente en **Perl**, por lo que no es necesario la inclusión de una agente **SNMP** para **MRTG**.
3. El archivo *log* de **MRTG** no crece demasiado, gracias a un algoritmo de consolidación de datos.
4. Las rutinas de **MRTG** para procesar datos que pueden consumir mucho tiempo están escritas en C.
5. Las gráficas generadas necesitan la biblioteca GD de Thomas Boutell
6. Las paginas HTML generadas, son totalmente configurables por el usuario.
7. **MRTG** es de dominio publico.

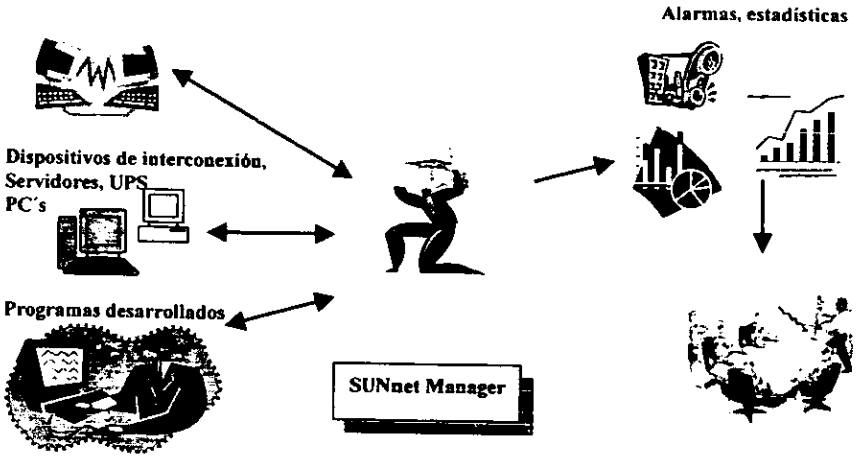
Instalación

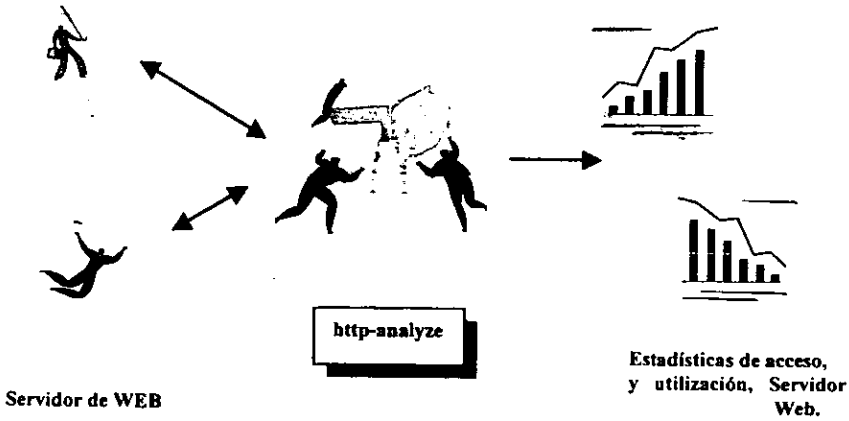
- Es necesaria la biblioteca GD de Thomas Boutell
- Es necesario **Perl** 5.003 o posterior
- Editar el archivo Makefile, para definir la opciones de configuración
- Decidir en que directorio **MRTG** generará y almacenará las paginas HTML.
- Adecuar el archivo de configuración de **MRTG**, al sistema que se va a monitorear
- Integrar la ejecución de **MRTG** en el *crontab*, para obtener la muestra de los datos en un intervalo de tiempo especifico.

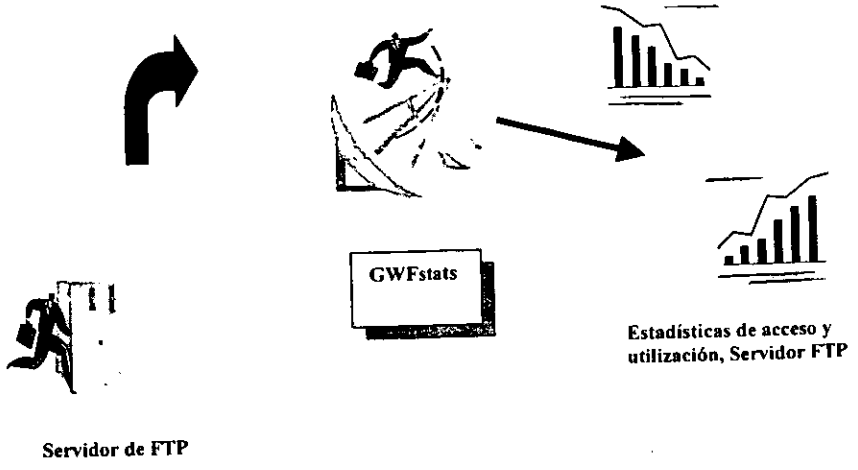
MRTG es utilizado en el Instituto de Ingeniería para monitorear el flujo de **e-mails** que se reciben y se generan en nuestro servidor de e-mail, para este fin se emplea el programa *mailstats* de **UNIX** que interactúa con **MRTG**.
Para obtener información más completa de donde se reciben **e-mails** y adonde se envían, **MRTG** no fue suficiente, fue necesaria la utilización de ***smtpstats*** que es un programa en **Born Shell**, que analiza la información que deja ***Sendmail*** en el archivo ***syslog*** y genera un listado mostrando cuantos e-mail han sido enviados desde que dominios, y cuantos han sido enviados a que dominios.

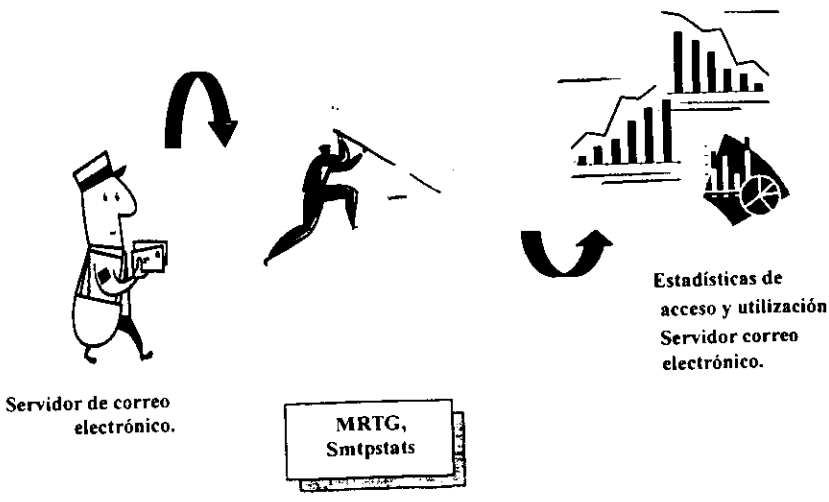
MRTG es una poderosa herramienta de monitoreo, es fácil de usar, sencillo de configurar y altamente confiable.

6.5 Esquema de aplicaciones que integran el sistema de monitoreo y administración del Instituto de Ingeniería






GWFstats
Estadísticas de acceso y utilización, Servidor FTP
Servidor de FTP


Estadísticas de acceso y utilización Servidor correo electrónico.
Servidor de correo electrónico.
MRTG, Smtpstats

6.6 Mantenimiento del sistema

El mantenimiento del sistema implementado, envuelve los procedimientos comunes de mantenimiento para cualquier equipo **UNIX**.

Ni **SunNet Manager**, ni ninguna aplicación que conforman el sistema de monitoreo y administración de REDII requiere procedimientos específicos de mantenimiento, sin embargo todos ellos basan sus actividades en análisis de archivos **log**, estos archivos por su naturaleza tienden a crecer de manera considerable en poco tiempo, de esta situación surge la necesidad de contar con procesos automatizados que permitan al administrador del sistema depurar estos archivos de una manera sencilla. En Miktlan se han realizado estos procedimientos con la ayuda de **SunNet Manager** , estableciendo valores umbral en el crecimiento de sistemas de archivos del servidor donde se almacenan dichos archivos **log**, cuando un valor umbral es rebasado, SunNet Manager lo detecta y a través de pequeños programas en **shell** depura los sistemas de archivos afectados.

Algunas aplicaciones que conforman el sistema de monitoreo y administración de REDII, necesitan ser ejecutados periódicamente, por lo que se recurrió a la utilería *cron*, de sistema operativo para poder realizar este proceso.

Una parte muy importante en el mantenimiento del sistema son los respaldos de la información generada por cada aplicación, además de los archivos de configuración de la aplicaciones. Se ha establecido la realización de respaldos totales cada 20 días de la configuración de las aplicaciones como del sistema operativo, con la finalidad de contar con copias de seguridad de todo el sistema de monitoreo y administración de REDII que nos permitan restablecer la configuración del sistema en caso de que una contingencia se presente. Por otro lado se realizan respaldos totales semanales de la información generada por las aplicaciones una vez depurados estos datos.

Capítulo 7

Presentación de resultados

7.1 Introducción

En este capitulo se muestran los resultados obtenidos en este proyecto. Dichos resultados han sido recopilados por espacio de varios meses de estudio, trabajando con las herramientas que integran el sistema de monitoreo y administración de la red del Instituto de Ingeniería.

Los temas que se expondrán en este capítulo siguen el orden: comportamiento de los dispositivos de interconexión de la red, comportamiento de los principales servidores en REDII, estudio de los servicios de Internet **Web**, **FTP** y **e-mail**.

7.2 Comportamiento de los dispositivos de interconexión de red.

REDII cuenta con varios dispositivos de interconexión, presentar el estudio de cada uno de estos dispositivos en este capitulo sería muy extenso y se perdería objetividad, por lo cual solo presentaremos aquí el estudio de los más representativos.

Los dispositivos de interconexión más representativos de REDII son, el concentrador del edificio 12 y el concentrador del edificio 2, debido a que manejan la carga mas fuerte de la red en cuanto a usuarios y a transmisión de datos.

El concentrador del edificio 2 es un modelo **MMAC-M5FNB**, con 4 tarjetas de puertos y una tarjeta de conexión a red **IRBM** de Cabletron, en este dispositivo hay 2 tarjetas de puertos que presentan mas carga (2 y 4), se tomaron los resultados obtenidos de estas dos tarjetas para mostrarlos en este trabajo, además de considerar también a la tarjeta **IRBM**. El concentrador del edificio 12 es modelo **MMAC-M8FNB** con 6 tarjetas de puertos y una tarjeta de conexión a red **IRBM**, en este dispositivo la tarjeta mas cargada es la tarjeta 6 en la cual nos enfocaremos junto con la tarjeta **IRBM**.

Para llevar acabo esta parte del estudio nos apoyamos en el software **SunNet Manager 2.2.2**, mediante el uso de las **MIB** correspondientes a cada dispositivo. Con el uso de

dichas **MIB**, se puede tener acceso a información diversa de los dispositivos, pero nos enfocamos a dos partes esenciales: el porcentaje de colisiones y el total de errores presentados por los dispositivos.

SunNet Manager a través de la **MIB** no proporciona el porcentaje de colisiones en cada tarjeta de los concentradores, solamente nos presentaba las colisiones y paquetes transmitidos, para llegar a los resultados que necesitábamos se tuvieron que hacer cálculos anexos. Otro Problema que se presento fue el formato en que **SunNet Manager** presenta los resultados ya que grafica la muestra de datos en un período de tiempo establecido, permitiendo manipular la gráfica generada con cierta flexibilidad pero no es lo suficientemente robusto para hacer una manipulación de datos que permita generar estadísticas como medias, medianas etc. Explicando un poco mas lo anterior, una grafica de **SunNet Manager** puede presentar sus datos en valores absolutos, delta y acumulativos, puede mezclar varias muestras de períodos de tiempo distintos en la misma grafica, pero no llega a mas su flexibilidad, además de que no permite exportar los datos en formato **SunNet Manager** a texto para hacer una manipulación posterior de los mismos.

Un punto muy importante a considerar fue el establecimiento de la frecuencia en la que se iba a realizar el **polling**, ya que este no puede ser determinado a la ligera porque puede generar diversos problemas a la red y a los nodos administrados tales como un incremento en el tráfico de la red, incremento en procesos de los servidores que están siendo monitoreados, lo cual se refleja en la baja del desempeño de estos componentes.
Casi toda la información obtenida por el nodo administrador es generada a través del método **polling**, lo que hace necesaria una política para establecer el tiempo que debe pasar entre un **polling** y otro. Esta política esta relacionada con el tamaño de la red y el número de agentes que pueda manejar de una manera efectiva el nodo administrador, es difícil definir lo anterior ya que el desempeño del nodo administrador dependerá de la capacidad de procesamiento que tenga, la velocidad y el nivel de congestionamiento en la red, y otros factores. Debemos aclarar que cada petición a un nodo administrado genera un agente el cual esta dedicado a la atención de las peticiones del nodo administrador únicamente al nodo administrado en cuestión. El nodo administrador puede manejar solo un agente al mismo tiempo, esto es cuando el nodo administrador realiza una petición **polling** a un agente en particular, este no puede realizar otra petición a cualquier otro agente hasta que la tarea haya terminado con el agente en cuestión. La petición de **Polling** hecha a cualquier agente puede envolver una sola transacción o bien una serie de transacciones.

Existe una ecuación sencilla que engloba lo explicado anteriormente:

$$N \leq T / \Delta$$

Donde:

N = Número de agentes.
T = Lapso de tiempo deseado entre **pollings** sucesivos del mismo agente.
Δ = Tiempo requerido para realizar un solo **polling**.

Despejando T de la ecuación anterior tenemos que:

$$T \leq N \cdot \Delta$$

Aplicando la ecuación anterior a nuestro ambiente de monitoreo, definimos lo siguiente:

Número de agentes (N). Para nuestro esquema de monitoreo tenemos definidos 24 servidores, a los cuales se les pueden aplicar hasta 6 diferentes peticiones de datos, en el caso de que se tuviesen todas las peticiones a todos estos servidores activas, tendríamos 144 agentes activos en el nodo administrador. Tenemos 21 dispositivos más contando los dispositivos de interconexión de la red, los **gateways** de red UNAM y los **ups**, a los cuales se les pueden aplicar hasta 3 diferentes peticiones, en el caso que se activaran todas las peticiones a estos dispositivos, tendríamos 63 agentes activos mas, lo que nos da un total de 207 agentes activos, en nuestro esquema de monitoreo. Redondeando este número tomaremos para la ecuación 210 agentes activos.

Tiempo requerido para realizar un solo **polling** (Δ). Se realizo una medida del tiempo que al nodo administrador le tomaba ejecutar un solo **polling** hacia un nodo administrador, encontrando un tiempo promedio de 3.90 segundos. Esta medida se realizo con el cronometro que el mismo sistema operativo provee, ejerciéndolo hacia varios nodos monitoreados.

Finalmente, sustituimos lo anterior en la ecuación y tenemos lo siguiente:

$$T \leq 210 \cdot 3.90 \text{ sec.}$$
$$T \leq 819 \text{ sec.}$$

Ya que 900 segundos son 15 minutos, decidimos redondear el valor de T = 900, para poder manejar minutos cerrados.

Antes de establecer este período entre **pollings** como valido, se tomaron varias muestras con períodos de tiempo que variaron entre 1 minuto y 15 minutos en muestras de 24 horas. Se consideró como, cada variación de tiempo influía tanto en la carga de la red como en los datos de la muestra misma, observando que el período de 15 minutos entre cada **polling** era el optimo. Ya establecido el valor anterior, también se vario el tiempo entre muestras, yendo desde cada 24 hrs., 48 hrs., 72 hrs. y 120 hrs. no encontrando variación importante en

los datos, el problema que se nos presento con las variaciones de tiempo mayores a 24 hrs. fue que al ejecutar las peticiones de monitoreo, muchas de ellas se interrumpian antes de concluir dejando segmentada la muestra. Por las razones anteriores se llego a la conclusión que un período de tiempo de 15 minutos entre **pollings** en muestras de 24 horas, era lo optimo para evitar demasiada carga en la red así como para que los datos reflejaran valores reales.

Cabe mencionar que estos períodos de tiempo establecidos se tomaron como base para casi todos los monitoreos a casi todos los nodos en nuestro esquema, sin embargo en algunos casos bajo circunstancias especiales se definieron otros intervalos entre **pollings**.

Finalmente en los meses de febrero y marzo de 1999, se llevaron a cabo monitoreos en los dispositivos de interconexión de red, con un período entre **pollings** de 15 minutos generando muestras de 24 horas. Específicamente en el mes de febrero se realizó el monitoreo en las tarjetas de puertos de los concentradores, mientras que en el mes de marzo se llevo a cabo el monitoreo de las tarjetas **IRBM**.

Como se menciono anteriormente, los resultados obtenidos al monitorear los dispositivos de interconexión de REDII, no los proporcionó directamente **SunNet Manager**, fue necesario realizar cálculos posteriores basados en los datos proporcionados por dicha aplicación. A continuación expondremos como se obtuvieron estos resultados.

7.2.1 Porcentaje de colisiones en cada tarjeta de los dispositivos seleccionados.

Comenzaremos por explicar la definición del porcentaje de colisiones, en la cual los cálculos están basados.

El porcentaje de colisiones en un dispositivo esta definido como el número de colisiones de salida divididos entre el número de paquetes de salida multiplicado por cien.

*Número de Colisiones / Número de paquetes * 100 = porcentaje de colisiones*

El primer paso que se realizo, fue recopilar los datos a través de **SunNet Manager** en los períodos de tiempo establecidos. Diariamente se obtuvieron gráficas en valores delta que mostraban las colisiones y los paquetes transmitidos. A continuación se muestran dos de las gráficas obtenidas por **SunNet Manager** de la tarjeta **IRBM** del concentrador 12, que representan las colisiones y los paquetes transmitidos en un período de 24 horas.

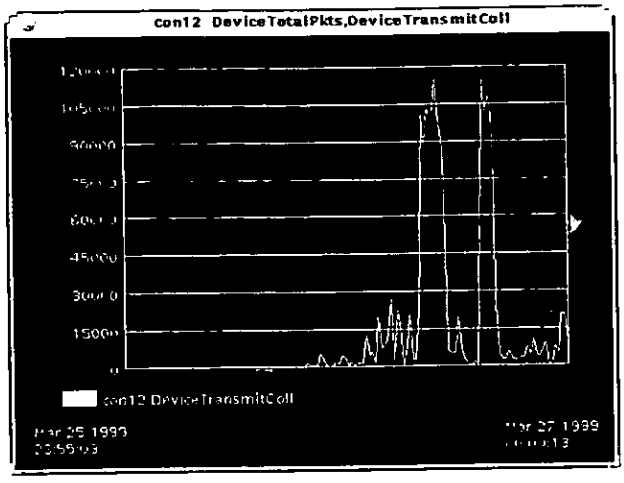


Figura 7.1 Gráfica del total de colisiones transmitidas por la tarjeta IRBM del concentrador 12, en una muestra de 24 hrs.

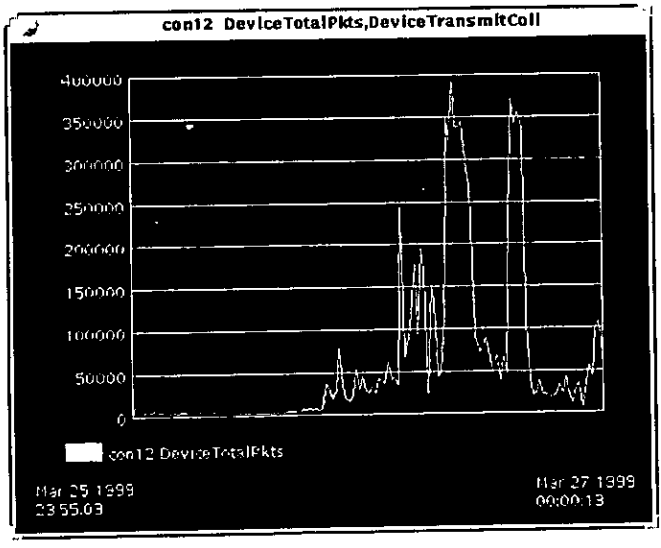


Figura 7.2 Gráfica del total de paquetes transmitidos por la tarjeta IRBM del concentrador 12, en una muestra de 24 hrs.

Una vez obtenidas las gráficas del total de paquetes y de colisiones en periodos de 24 horas, se obtuvieron de cada una de estas los valores de los datos en intervalos de 30 minutos desde las 8:00 a.m. a las 10:00 p.m., con el fin de acotar la muestra a las horas de mayor actividad dentro de REDII.

De esta manera se genero un rango de valores de colisiones y paquetes por cada 24 horas, a estos valores se les aplico la formula de porcentaje de colisiones descrita anteriormente, obteniendo el porcentaje de colisiones cada 30 minutos en muestras de 24 horas.

Con estos datos pudimos crear gráficas diarias donde poder observar las variaciones del porcentaje de colisiones en el transcurso de los días en las horas de mas actividad en REDII.

Nos interesaba obtener valores representativos de las muestras diarias, con el fin de generar graficas que mostraran valores característicos del porcentaje de colisiones a través de una semana. Para obtener estos valores, se obtuvo la mediana de cada rango de datos diarios en una hora específica (por ejemplo: 8:30, 9:00 o 22:00).

También se generaron graficas semanales donde se muestran las variaciones de las colisiones y los paquetes transmitidos. De igual manera para crear estas graficas, se obtuvieron los valores medios de cada rango de datos diarios en una hora especifica para las colisiones y paquetes. Se manejo la mediana en vez de la media aritmética con el fin de eliminar el efecto de los valores extremos en las muestras.

En el apéndice A, se muestran los resultados obtenidos para los concentradores de los edificios 2 y 12, tomando en cuenta las tarjetas con mas carga y las tarjetas **IRBM**.

7.2.2 Errores en cada tarjeta de los dispositivos seleccionados.

De manera similar al proceso anterior, el primer paso que se realizo, fue recopilar los datos a través de **SunNet Manager** en los periodos de tiempo establecidos. Diariamente se obtuvieron gráficas en valores delta que mostraban los errores presentados por el dispositivo monitoreado, a continuación se muestra una gráfica generada por **SunNet Manager** donde se muestran los errores presentados en un periodo de 24 horas.

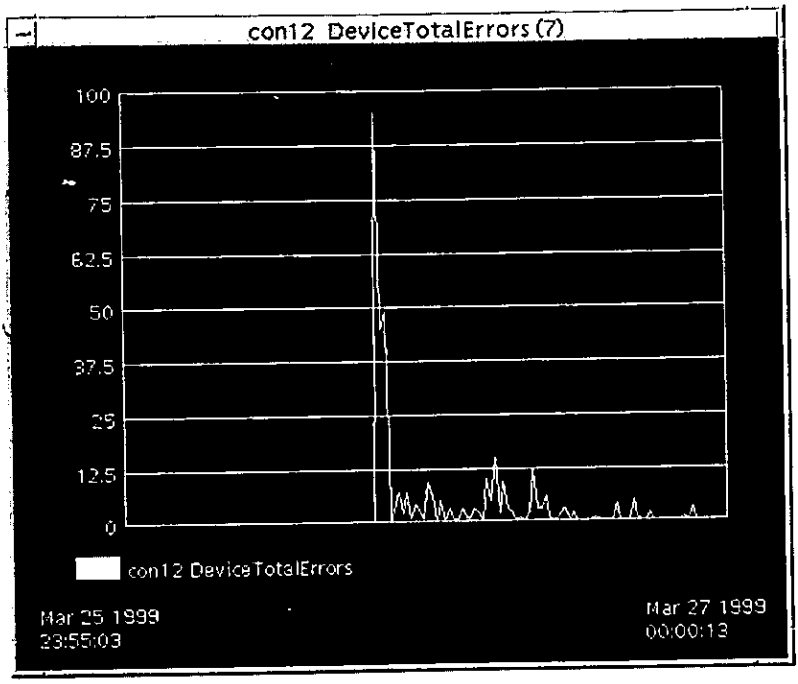
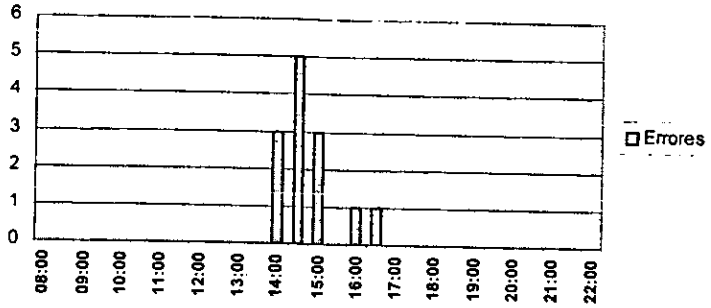


Figura 7.3 Gráfica del total de errores en la tarjeta IRBM del concentrador 12, en una muestra de 24 hrs.

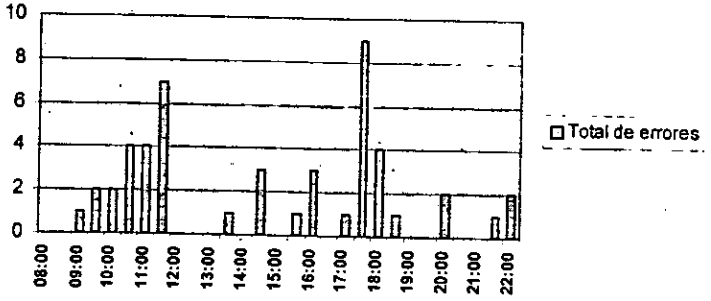
Con el fin de acotar las muestras diarias a los periodos de tiempo de mas actividad en REDII, se tomaron los valores contenidos dentro del periodo de tiempo de las 8:00 a.m. a las 10:00 p.m.

En este caso también nos interesaba generar gráficas semanales, por lo que también se obtuvieron los valores medios de los rangos de datos diarios en una hora especifica, para poder obtener dichas gráficas.

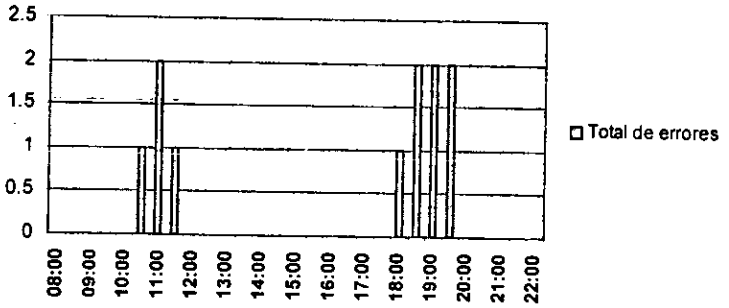
A continuación se presentan algunas gráficas representativas del total de resultados obtenidos al monitorear los errores generados por las tarjetas con más carga y las tarjetas **IRBM** de los concentradores de los edificios 2 y 12.



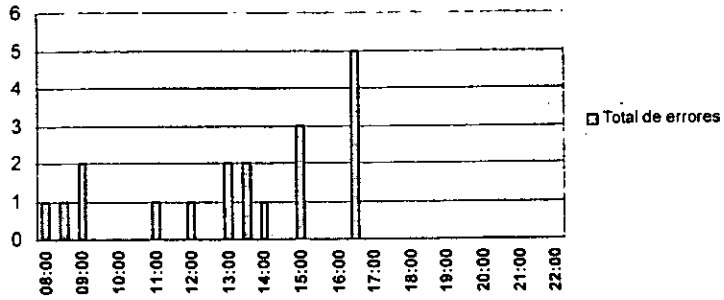
Concentrador 12,tarjeta5, tercera semana de Febrero 1999.



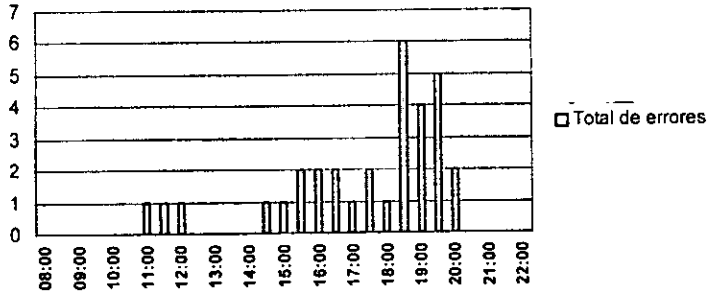
Concentrador 12 IRBM segunda semana de Marzo 1999.



Concentrador 2, Tarjeta 2, primera semana de Febrero 1999.



Concentrador 2, tarjeta 4, segunda semana de Febrero 1999.



Concentrador 2 IRBM, segunda semana de Marzo 1999.

7.2.3 Conclusiones

En primer lugar, se explicaran las conclusiones a las que se llego después de considerar los resultados obtenidos en los dispositivos de interconexión de REDII, con respecto al porcentaje de colisiones encontrado en ellos.

Si más de un nodo en la red intenta transmitir en el canal **Ethernet** en el mismo momento, entonces se presenta una *colisión*. Todos los nodos involucrados son notificados que una colisión ocurrió, e inmediatamente redisponen la transmisión de su paquete usando un algoritmo especial que hace que cada nodo involucrado espere una cantidad de tiempo aleatoria para realizar la retransmisión.

Desafortunadamente el diseño original de **Ethernet** empleo la palabra "colisión" para nombrar este aspecto del control de acceso al medio, si hubiera sido llamado de otra manera como por ejemplo acontecimiento estocástico del arbitraje al medio (AEAM), entonces nadie se preocuparía por la ocurrencia de AEAM en una red **Ethernet**; sin embargo el termino "colisión", suena como si algo malo estuviese pasando en la red. Por esto mucha gente cree que las colisiones en la red son una indicación de falla. La realidad es que las colisiones son eventos absolutamente normales y esperados en una red **Ethernet**, e indican que el protocolo de acceso al medio (**CSMA / CD**) esta funcionando para lo que fue creado.

Si se incrementa el número de nodos en una red, se incrementará el trafico en la misma y también el número de colisiones crecerá como parte de la operación normal de la red **Ethernet**. La mayoría de las colisiones presentes en una red que no esta sobre cargada serán resueltas en microsegundos o milisegundos, las colisiones empiezan a considerarse un problema cuando una red esta sobrecargada ya que al no haber suficiente ancho de banda para dar salida a los paquetes que los números nodos en la red desean transmitir, las colisiones se generan una y otra vez, lo que puede causar la perdida de los paquetes que esperan ser transmitidos. De esta manera, un porcentaje de colisiones elevado puede ser una indicación de una red sobrecargada.

Los siguientes son valores del porcentaje de colisiones que deben ser considerados en la operación de una red **Ethernet**:

- 5% - 10% de colisiones. Porcentaje normal de colisiones en una red medianamente cargada.
- 10% - 30% de colisiones. Las colisiones empiezan a interferir con el buen desempeño de la red.
- 30% - 70% de colisiones. El rendimiento en la red o segmento de red, se ve severamente afectado.
- 70 % de colisiones. Si este limite es sobrepasado, es altamente probable que en el segmento en donde se presenta esta condición sea imposible la transferencia de datos.

En el caso del REDII, se puede observar en las gráficas, que el período de actividad en esta red es de las 9:00 a.m. a las 22:00 p.m., presentándose picos de actividad de 10:00 a.m. a 15:00 p.m. y de 17:00 p.m. a 22:00 p.m.

Se puede apreciar en las gráficas de las tarjetas del concentrador 2 (tarjeta 2 y 4), que los valores del porcentaje de colisiones sobrepasan el 10% en varias ocasiones llegando hasta el 30% de colisiones en los períodos de mayor actividad en la red. Los porcentajes de colisiones observados en la tarjeta **IRBM** del mismo concentrador van de igual manera del 10% al 30% en el mismo período de actividad. Estos niveles en el porcentaje de colisiones, decrementan el desempeño de la red en este edificio, de manera que en las horas pico de actividad la velocidad de la red se ve afectada lo que genera un impacto en los usuarios.

En el caso de la tarjeta 5 en el concentrador del edificio 12, se puede observar altos niveles de porcentaje de colisiones, que van desde el 10% alcanzando niveles hasta del 60%, en tanto que los porcentajes de colisiones en la tarjeta **IRBM** del concentrador del edificio 12 van del 10% hasta el 35% o 40 %. Estos porcentajes de colisiones decrementan el desempeño de la red para este edificio de una manera parecida a la del edificio 2, pero en este caso la problemática esta acentuada debido a que la actividad en esta parte de REDII es mayor.

De lo anterior podemos concluir, que en REDII, existen niveles altos de porcentaje de colisiones que deben ser considerados ya que afectan el desempeño de la red y esto interfiere con el trabajo del personal que labora en la institución al generar retardos en el acceso a fuentes de información de importancia en el Internet y dentro de REDII.

Estos niveles en los porcentajes de colisiones en REDII, son generados debido a una sobre carga en la red. Antes de llegar a esta conclusión se consideraron otros factores por los cuales se pudieran presentar estos niveles de colisiones en REDII. El primer factor considerado fue, problemas de hardware, en algunas ocasiones un daño a nivel hardware en alguno de los puertos de las tarjetas de los dispositivos de interconexión de red, de las estaciones de trabajo o de las computadoras personales puede generar que un paquete sea retransmitido en varias ocasiones lo cual genera niveles de colisiones altos, se verifico que esto no estuviese ocurriendo en ningún dispositivo. Otro factor considerado fue la segmentación de la red. Es recomendable que al presentarse altos niveles de colisiones, la red sea configurada en varios segmentos físicos con la finalidad de reducir la contención de canal sobre la misma red, la forma de lograr esto es mediante el uso de conmutadores o **switches**, tecnología que opera sobre la capa 2 del modelo OSI. Por ejemplo se tiene una red la cual se encuentra paralizada debido a un trafico excesivo generado por un gran número de nodos que han sido conectados a esta red, es posible dividir la red en dos segmentos con lo que se reduce la carga de cada segmento a la mitad, si el tráfico continua siendo un problema, se podrá dividir la red en cuatro o seis segmentos y así sucesivamente. El conmutador manipula el tráfico entre los nodos de cada segmento, si un nodo situado en un segmento necesita comunicarse con otro nodo perteneciente a otro segmento, el dispositivo de conmutación actúa como puente y establece un circuito temporal entre los segmentos. REDII está configurada para emplear una topología física en estrella, la cual se

basa en una tecnología **Ethernet** conmutado a 10 Mbps. Una característica importante de la topología de REDII, es que como punto medular se utiliza un conmutador **Ethernet**, lo que convierte a REDII en una red segmentada físicamente, es decir la segmentación para reducir el trafico que se menciono anteriormente REDII la tiene ya implementada. Los segmentos de REDII fueron creados y configurados en base a un estudio minucioso previo.

REDII a crecido mucho en los últimos años, y cada día se incrementa este crecimiento, el ir agregando mas nodos en la red conlleva al aumento en el porcentaje de colisiones y aun retraso en la transmisión de información entre nodos, afectando directamente a cada componente (servidores, computadoras personales, etc.) ya que deben competir por el medio de transmisión, presentándose de esta manera un desempeño deficiente, el cual es un problema considerable pero no ha llegado al limite de dejar inoperable la red. Sin embargo si se considera que las aplicaciones que los usuarios de la red requieren, cada vez son más complejas y envuelven el manejo de cantidades de información mucho mas grandes (texto, audio, imágenes y video), pronto el problema de la sobre carga de REDII puede tornarse muy crítico.
Ahora se expondrá el significado de los resultados obtenidos en cuanto a los errores encontrados. Las gráficas muestran los paquetes de salida que no fueron transmitidos en los dispositivos analizados a causa de ciertos errores. Estos errores pueden ser de los siguientes tipos:

- Errores de CRC[1] o Alineación. Normalmente un paquete a nivel capa física (**frame**) es transmitido con un valor CRC el cual es estándar, cuando el paquete llega al nodo receptor este recalcula el contenido de **frame** para verificar que el valor CRC de este coincida con el valor CRC esperado. Si se encuentra alguna diferencia entre estos valores es posible que el **frame** este corrupto y un error CRC se generará. Los errores CRC y de alineación, en muchos casos son recopilados como si se tratara del mismo tipo de error, esto es ya que los dos realmente advierten problemas de corrupción en los **frames** a niveles de bytes. Si el nivel de este tipo de errores excede el 2 % o 3% del total del trafico, se considera de un nivel excesivo. Se debe verificar que el cableado no tenga ninguna falla si estos errores se presentan, así como las tierras de los equipos, o alguna falla en tarjetas de red o dispositivos transmisores – receptores de red conocidos también como **transceivers**.
- **Jabbers**. En algunas ocasiones las interfaces de red y **transceivers** externos pueden generar problemas llamados **Jabbers**[2]. Esto se presenta cuando bits de datos truncados o mutilados son emitidos dentro de una secuencia de **frames** dentro de una transmisión continua. Los **Jabbers** cuando se presentan puede incrementar el trafico en al red generando problemas graves en la transmisión de datos. Si el nivel de este tipo de errores excede el 2% o 3% del total del trafico se considera un nivel excesivo. Este tipo de errores se presenta por fallas en las tarjetas de red en nodos y dispositivos de interconexión.

[1] Cyclical Redundant Check

[2] Nombre en ingles sin traducción

El nivel de paquetes erróneos con relación al trafico de salida de los concentradores considerados, se mostrarán a continuación. Se tomo en cuenta todas las tarjetas antes mencionadas para cada concentrador además de las tarjetas **IRBM**.

	Valor Mínimo	Valor Máximo	Valor Medio
Concentrador 2	0.001%	0.013%	0.004%
Concentrador 12	0.0003%	0.03%	0.004%

Tabla 7.1, valores representativos de los errores encontrados en los concentradores de los edificios 2 y 12 en el mes de Marzo.

Como se puede observar, los niveles de paquetes erróneos en cada concentrador no sobrepasan el 1% ni aun en sus valores máximos, el nivel medio en ambos concentradores es del 0.004%, el cual es un nivel altamente aceptable. Con esto podemos concluir que los errores generados por causas físicas (problemas con cableado, tarjetas de red dañados, problemas con dispositivos mal polarizados o mala instalación de tierra física), no generan conflictos en REDII.

7.3 Comportamiento de los servidores del Instituto de Ingeniería.

El Instituto de Ingeniería, cuenta con mas de 20 servidores **UNIX** distribuidos en las diferentes áreas que lo conforman. Estos servidores son herramientas muy útiles en las investigaciones que en la institución se realizan. En esta etapa de la tesis presentaremos los resultados obtenidos de los monitoreos de los 3 servidores más importantes.

7.3.1 Servidor PUMAS.

PUMAS es el servidor principal del Instituto de Ingeniería, debido a que en el se ejecutan los principales servicios de Internet : Correo electrónico, servidor de **Web**, servidor de **FTP**, además de fungir como servidor de archivos, también tiene instalados lenguajes como C, C++, y Fortran permitiendo a algunos usuarios realizar desarrollos en estos lenguajes.

PUMAS es una SPARCstation 20, que cuenta con 2 procesadores a 60 MHZ, 136 MB de memoria, dos discos duros de 1.05 GB, y ochos discos más de 2.1 GB. El sistema operativo de pumas es **Solaris** 2.5.1. Ocho de los 10 discos de pumas, están configurados con la tecnología de **RAID**[3], en cuatro diferentes volúmenes.

El monitoreo del servidor PUMAS, se llevo a cabo en los meses de Febrero y Marzo de 1999, en muestras de 24 horas con intervalos entre **pollings** de 15 minutos.

[3] Redundant Arrays of Inexpensive Disks (RAID). Grupo de discos que aparecen ante el sistema como discos virtuales también conocidos como volúmenes, los cuales usan parte de su capacidad de almacenamiento para guardar información duplicada de los datos almacenados en ellos. Esta información duplicada hace posible regenerar los datos si se presenta una falla en algún disco.

A continuación presentaremos. los resultados del monitoreo hacia el servidor PUMAS, en el área de administración de desempeño.

7.3.1.1 Procesadores.

En las gráficas que se presentan a continuación podemos observar la utilización de los procesadores de pumas. No se muestra la utilización de cada uno de los procesadores ya que la **MIB** que se utilizo para realizar este monitoreo, presenta la utilización de todos los procesadores del servidor como una entidad.

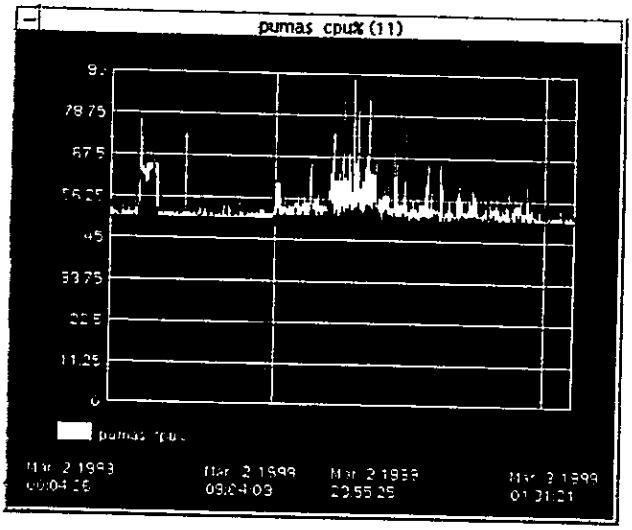


Figura 7.4 Gráfica que muestra la utilización de los procesadores del servidor PUMAS en una muestra de 24 promedio, indicando el periodo de más actividad.

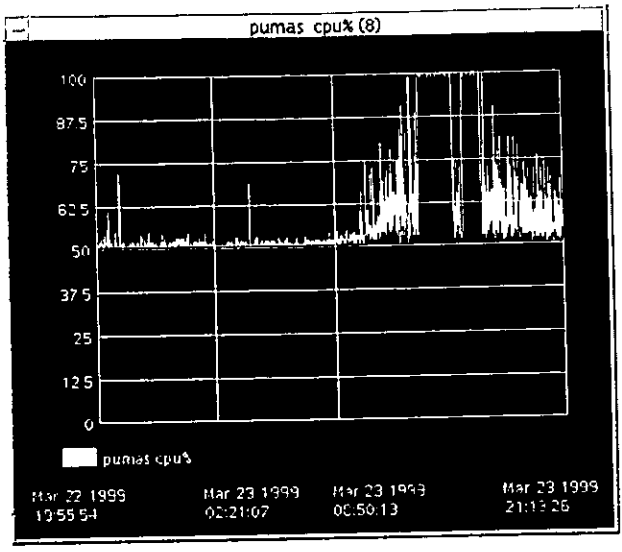


Figura 7.5 Gráfica que muestra la utilización de los procesadores del servidor PUMAS en una muestra promedio de 24 horas, indicando las horas de menor actividad, también pudiéndose apreciar las horas de mayor actividad.

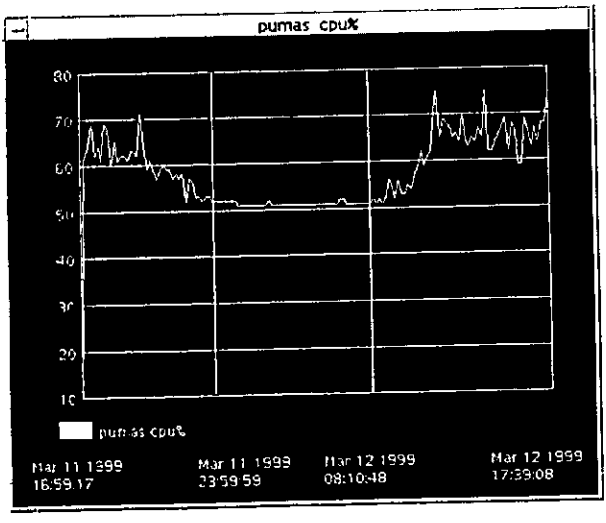
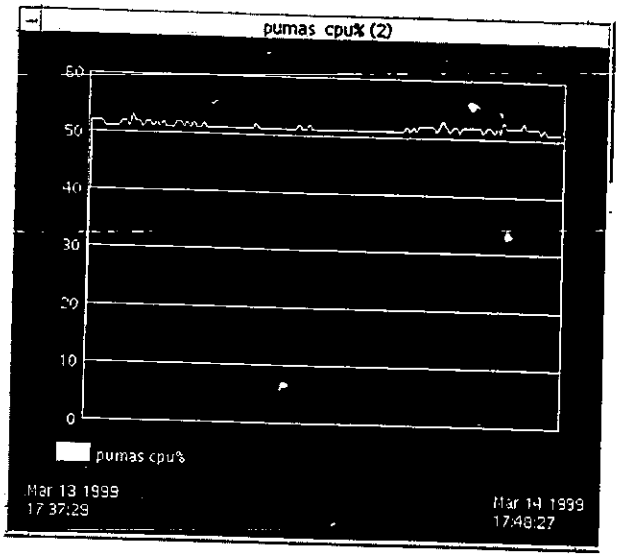


Figura 7.6 Gráfica que muestra la utilización de los procesadores del servidor pumas en una muestra promedio de 24 hrs., indicando las horas de menor actividad.



7.8 Gráfica que muestra la utilización de los procesadores del servidor PUMAS, en una muestra de 24 horas en fin de semana.

Las gráficas anteriores fueron extraídas del total de muestras tomadas durante el mes de Marzo y Febrero de 1999. Los resultados encontrados en este periodo de monitoreo respecto a la utilización de los procesadores de PUMAS, son los siguientes:

- PUMAS presenta una utilización promedio del 50% en sus procesadores dentro de las 24 horas del día aun en los días no laborables (sábado y domingo).
- El período de mayor actividad se encuentra de las 9:00 a.m. a las 12:00 a.m., este período puede dividirse de las 9:00 a.m. a las 15:00 p.m. y de las 17:00 p.m. a las 12:00 a.m.
- La utilización de los procesadores, también presenta picos hasta del 100%.

7.3.1.2 Área de almacenamiento de respaldo (Swap)

El monitoreo del área de **swap** es uno de los puntos medulares en la administración de desempeño orientada a servidores, debido a que nos permitirá saber de que manera se esta utilizando la memoria física primaria.

La **MIB** que utilizamos para monitorear el área de **swap** de nuestros servidores, nos presento la cantidad de paginas intercambiadas entre la memoria primaria y el área de **swap**.

Para entender mejor, los parámetros y la terminología utilizada anteriormente, explicaremos brevemente los principios de operación del sistema de memoria de los servidores **UNIX**, basándonos en el sistema operativo **Solaris**.

La principal característica del sistema de memoria de los servidores **UNIX** es el uso del sistema de memoria virtual **VM**[4]. Uno de los objetivos de este sistema es permitir la existencia de objetos de memoria, mayores que la memoria física disponible. Esto permite a los procesos tener más memoria disponible no solo la memoria física (**RAM**), de esta manera los procesos pueden utilizar una lenta pero más grande memoria secundaria o área de **swap** que reside en una o más particiones de disco, esta memoria secundaria es utilizada como almacenamiento de respaldo. EL sistema de **VM** maneja de manera transparente el almacenamiento virtual entre la **RAM** y el **swap**. Debido a que la **RAM** es significativamente más rápida que el acceso al disco (aproximadamente 100,000 veces más rápida), el trabajo del sistema **VM**, es mantener las porciones de memoria más frecuentemente referenciadas en la memoria primaria. En el caso de que la memoria primaria se termine, el sistema **VM** transfiere la porción de memoria menos utilizada al área de **swap**.
Existen dos tipos de sistemas de **VM** usados en la mayoría de sistemas operativos, estos son: ***swapping*** y ***demand paged***. Con el sistema de **swapping** si existe una escasez de memoria primaria, el proceso menos activo es intercambiado de la memoria primaria hacia el área de **swap**, liberando memoria para los otros procesos. Este método es fácil de implementar para el sistema operativo, pero el desempeño se ve seriamente afectado ya que todo el proceso es sacado de la memoria primaria, por lo que cuando el proceso entra en actividad es necesario moverlo o "subirlo" completo a la memoria primaria. El modelo **demand paged**, utiliza pequeños pedazos de memoria conocidos como *páginas* (un proceso puede estar referenciado por varias páginas). Mientras el modelo **swapping** transfiere todo un proceso al área de almacenamiento de respaldo, el modelo **demand paged**, transfiere las páginas menos utilizadas, lo cual permite a los procesos continuar mientras una parte inactiva de ellos es transferida al área de **swap. Solaris**, utiliza una combinación de los dos modelos descritos anteriormente. El modelo **demand paged** es usado bajo circunstancias normales, mientras que **swapping** es utilizado solamente como ultimo recuso.

Uno de los parámetros que indica escasez de memoria primaria en un servidor, es el intercambio constante de páginas de la memoria primaria al área de **swap**. El encontrar esta condición en un sistema no es suficiente para determinar que se necesita agregar más memoria primaria, es necesario hacer un estudio de las aplicaciones que se están ejecutando en ese servidor antes de emitir un juicio ya que una aplicación mal diseñada o adaptada al sistema o bien un programa mal hecho puede generar un consumo excesivo de memoria, si este fuera el caso es necesario el análisis de cada aplicación y llevar acabo mejoras, antes de agregar más memoria primaria. Al descubrir un alto intercambio entre la memoria primaria y el área de **swap**, estamos encontrando un problema de desempeño que hay que analizar.

[4] Por sus siglas en ingles: Virtual Memory (**VM**)

A continuación se presentarán gráficas que muestran el comportamiento del área de respaldo del servidor PUMAS.

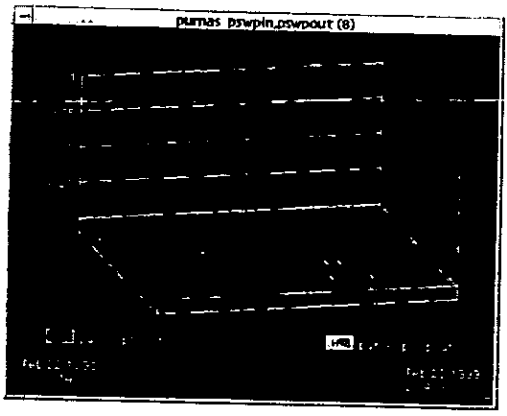


Figura 7.9 Comportamiento del área de almacenamiento de respaldo del servidor PUMAS en una muestra promedio, en el mes de Febrero de 1999.

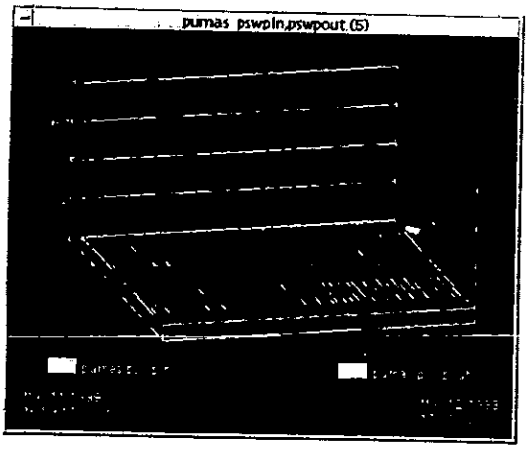


Figura 7.10 Comportamiento del área de almacenamiento de respaldo del servidor PUMAS en una muestra promedio, en el mes de Marzo de 1999.

Las gráficas anteriores fueron extraídas del total de muestras obtenidas en los meses de Febrero y Marzo de 1999. Del período de monitoreo pudimos observar que la transferencia de paginas entre la memoria primaria y el área de almacenamiento de respaldo es nula, lo que quiere decir que los procesos que se ejecutan en pumas, son manejados apropiadamente por la memoria primaria.

7.3.1.3 Disco

Como se mencionó anteriormente el servidor PUMAS, cuenta con 10 discos, a continuación se presentarán las gráficas que muestran la utilización de cada uno de estos discos.

Las gráficas presentan a cada disco con sus respectivas particiones, mostrando el total de Mbytes asignado, la cantidad de espacio usado y disponible por partición en Mbytes.

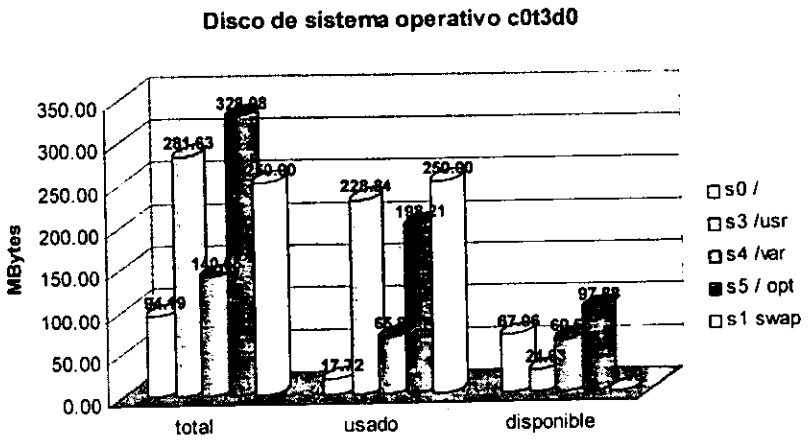
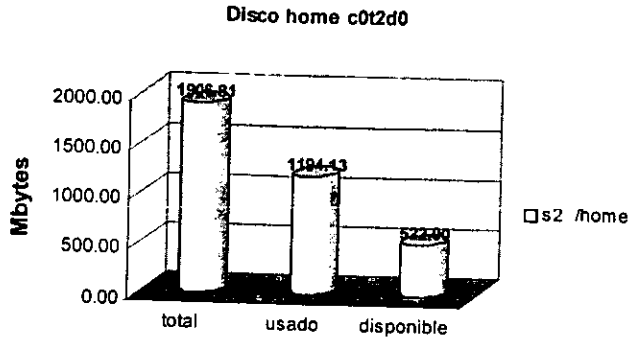
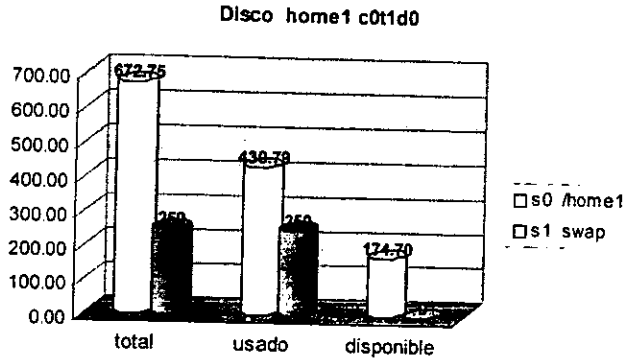
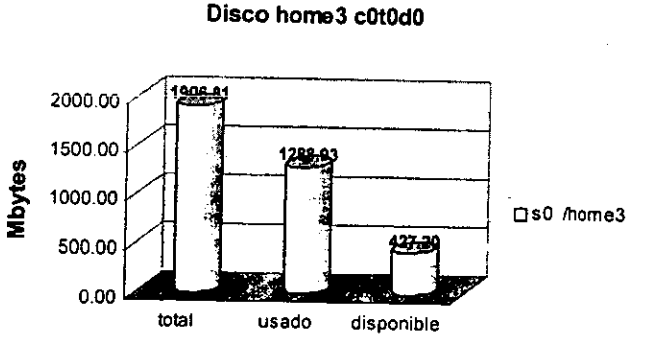


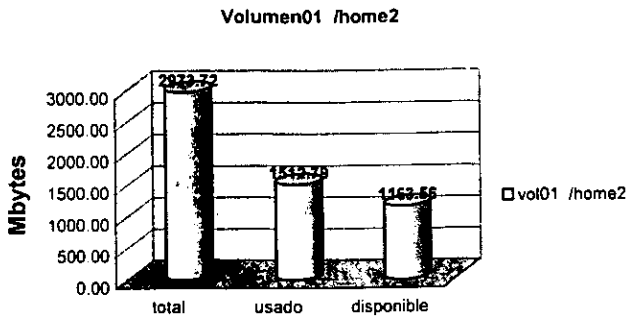
Figura 7.11 Utilización del disco de sistema operativo del servidor PUMAS.

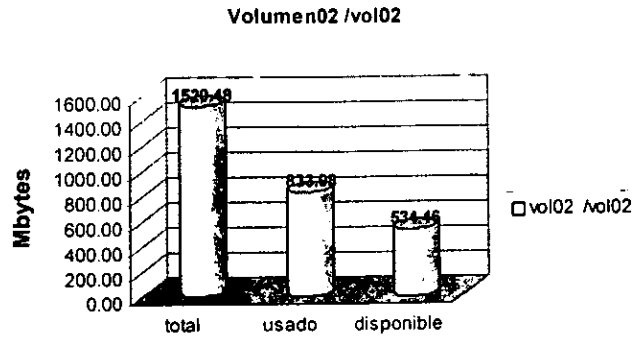


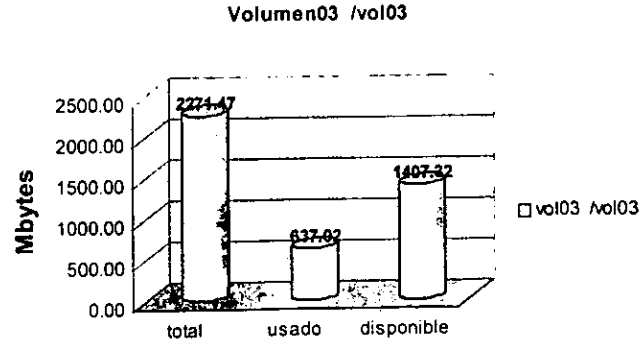




Figuras 7.12, 7.13, 7.14 Utilización de discos del servidor Pumas







Figuras 7.15, 7.16, 7.17 Utilización de los volúmenes del servidor PUMAS

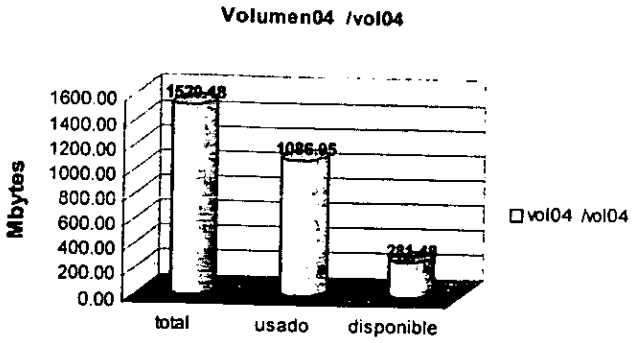


Figura 7.18 Utilización de volumen /vol04 en el servidor PUMAS.

Disco o Volumen	Total usado	Total disponible	Uso
.c0t3d0 Sistema Operativo	75 %	25 %	Sistema operativo y una partición destinada a swap
.c0t1d0 /home1 y swap	80%	20%	Directorios home de la mayoría de usuarios, una partición de swap
.c0t2d0 /home	70%	30%	Directorios de usuarios
.c0t0d0 /home3	76%	24%	Directorios de usuarios que son compartidos a través de NFS, información del servicio de **Web**
.vol01 /home2	57%	43%	Información del servicio de **FTP**.
.vol02 /vol02	61%	39%	Correo electrónico.
.vol03 /vol03	32%	68%	Información de usuarios.
.vol04 /vol04	80%	20%	Información de monitoreo.

NOTA: Las particiones definidas como área de **swap,** son mostradas como totalmente utilizadas, debido a que este espacio no se puede utilizar para almacenar información, sin embargo la utilización real de las áreas de **swap** es de un 2% promedio.

Tabla 7.2 Información de la utilización de todos los discos y volúmenes del servidor PUMAS.

La información de todos los discos se obtuvo por medio de **SUNnet Manager** a través de una **MIB** apropiada, con estos datos se realizaron gráficas en Excel para la presentación de los resultados en este capítulo, debido a que las gráficas generadas por **SunNet Manager** no son tan explicitas.

7.3.1.4 Red: Colisiones.

En las gráficas que se presentan a continuación, se muestra los niveles de colisiones que se presentan en el servidor PUMAS.

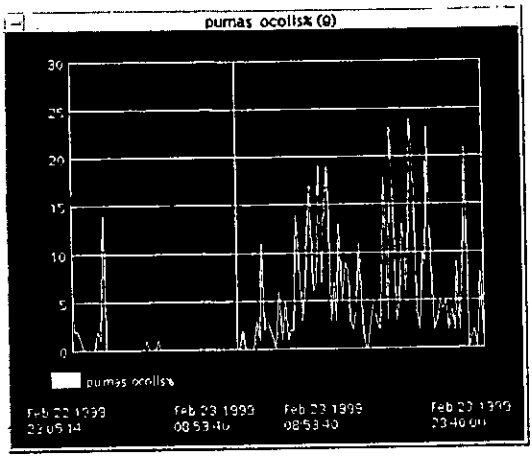


Figura 7.19 Gráfica que muestra las colisiones en la interfaz de red del servidor PUMAS. Muestra promedio 24 horas, con indicación de período de mayor actividad.

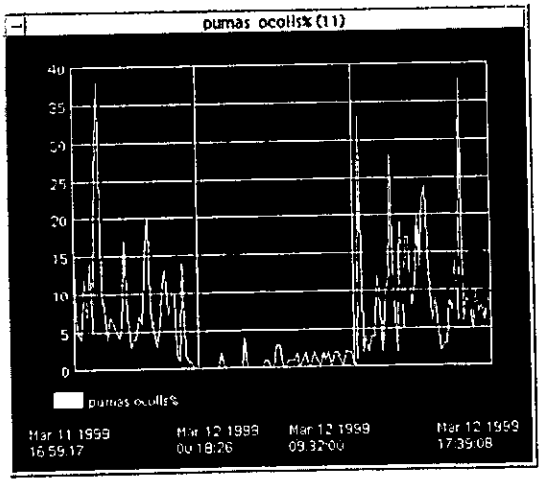


Figura 7.20 Gráfica que muestra las colisiones en la interfaz de red del servidor PUMAS. Muestra promedio 24 horas, con indicación de período de menor actividad.

Estas gráficas fueron tomadas del total de muestras obtenidas en el periodo de monitoreo comprendido en los meses de Marzo y Febrero de 1999. De este período de monitoreo obtuvimos los siguientes resultados:

- El porcentaje de colisiones promedio encontrado en la interfaz de red del servidor pumas es de 5% - 10%.
- Se presentan picos de colisiones entre 15% a 40%
- El período de mayor actividad de transferencia de paquetes de red en este servidor es de 9:00 a.m. a 12:00 a.m., sin embargo se presentan actividad durante las 24 horas del día.

7.3.1.5 Conclusiones.

El primer punto que se trato en cuanto al desempeño del servidor PUMAS, fue la utilización de sus procesadores, como pudimos observar el valor promedio de utilización es del 50% para las 24 horas del día, encontrándonos con picos de hasta el 100% de utilización. El desempeño del servidor es bueno con estos niveles de utilización de los procesadores, sin embargo hay un impacto cuando se presentan los picos de más de 80%. PUMAS cuenta con más de 550 usuarios, la mayoría de ellos lo utilizan como servidor de correo electrónico, sin embargo PUMAS ofrece además del correo electrónico una amplia variedad de servicios. Los usuarios de PUMAS se incrementan día con día y sus necesidades también. Por el momento el desempeño del servidor no se ve afectado seriamente por los niveles de utilización de sus procesadores, sin embargo podría representar un problema grave en un futuro cercano. Es muy recomendable en primera instancia crecer en velocidad los procesadores, también es aconsejable crecerlos en número, sin embargo esto no es posible debido a que la arquitectura de la SPARCstation 20 no lo permite, de manera que incrementando la velocidad de procesamiento se puede prever un problema futuro con un costo menor.

El segundo punto tratado, fue la utilización del área de almacenamiento de respaldo o **swap**. Las gráficas nos muestran que el intercambio entre la memoria primaria y el área de **swap** es casi nulo, por lo que podemos concluir que PUMAS cuenta con la cantidad de memoria **RAM** suficiente para poder manejar todos los procesos que en ella se ejecutan.

El tercer punto tratado, fue el espacio en disco. Como podemos observar en la tabla 7.2 donde se resume la utilización de los discos de este servidor, casi todos los discos y volúmenes presentan una utilización mayor del 50%, sin embargo no hay una exigencia para incrementar el número de discos ya que los usuarios a los que se les permite tener un directorio en uno o más sistemas de archivos del servidor, tienen definidas cuotas limite de uso de disco lo cual asegura que ningún sistema de archivos será llenado con la información de los usuarios. Por otro lado, existe información generada por el monitoreo la cual reside en el volumen vol04, debido a que el espacio en disco del nodo administrador fue insuficiente, esta información esta en el servidor PUMAS de manera temporal, por lo que al liberar el volumen en cuestión se tendrá disponible 1.5 GB de espacio en disco para poder distribuir entre las diferentes necesidades que se presenten. Una ventaja es que PUMAS cuenta con un arreglo de discos SSA 210, el cual tiene capacidad para 24 discos más de 2.1 GB cada uno, esto facilita el incremento de discos cuando se requiera.

Finalmente se mostró el porcentaje de colisiones del servidor, en las gráficas pudimos observar que el nivel de colisiones presenta valores promedio del 5% al 10%, sin embargo se presentan picos que rebasan el 20% hasta llegar al 40%, estos niveles de porcentaje de colisiones son generados por una sobre carga en la red como ya explicamos en la sección 7.2

7.3.2 Servidor TONATIUH

El servidor TONATIUH, es uno de los servidores más importantes de la coordinación de sistemas de computo y del Instituto de Ingeniería debido a que en el se desarrollan diversos sistemas de información. TONATIUH es un servidor **SPARCstation** 20, con sistema operativo **Solaris** 2.5.1., cuenta con dos procesadores a 50 MHz, 128 Mbytes de memoria RAM, con cuatro discos, dos de ellos son 535 Mbytes y los dos restantes tienen capacidad de 2.1 Gbyte.
El monitoreo de este servidor se llevo a cabo también en los meses de Febrero y Marzo de 1999, recopilando muestras de 24 horas con periodos de **polling** de 15 minutos. Mostraremos a continuación el resultado de dicho monitoreo siguiendo el orden que se llevo con el servidor PUMAS.

7.3.2.1 Procesadores

En las gráficas que a continuación se presentan se puede observar la utilización de los procesadores de este servidor como una entidad, esto es debido a la **MIB** que se utilizo para realizar este monitoreo.

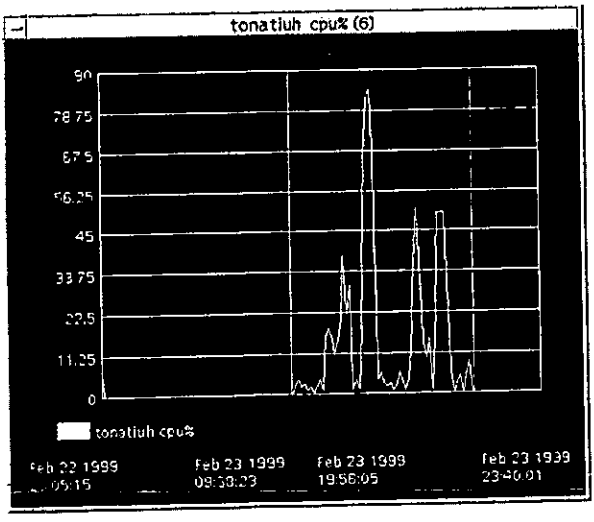


Figura 7.21 Utilización de procesadores del servidor TONATIUH, muestra promedio 24 horas, indicando las horas de mayor actividad

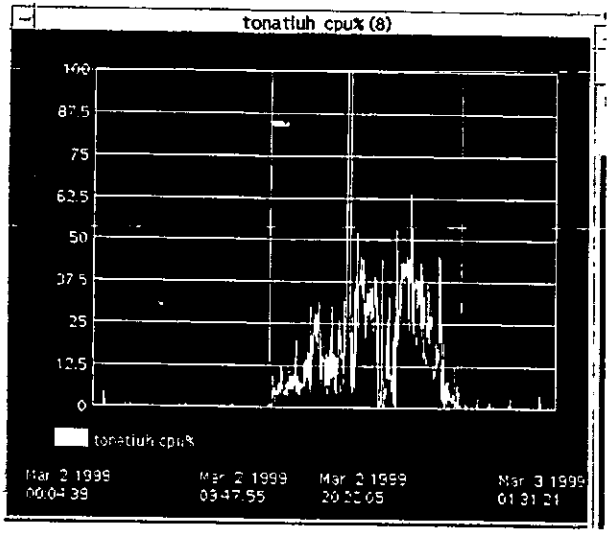


Figura 7.22 Utilización de los procesadores del servidor TONATIUH, muestra promedio 24 horas, indicando el periodo de mayor actividad.

Las gráficas anteriores fueron extraídas del total de muestras generadas en el período de monitoreo antes mencionado. A través de dicho período de monitoreo pudimos observar:

- El lapso de tiempo donde se presenta actividad en los procesadores del servidor es: de 9:30 a.m. a 20:30 p.m. Detectando que de las 12:00 p.m. a las 15:00 p.m. se ve incrementada la utilización.
- Se observan picos de utilización que van de un 80% a un 100%, en pequeños lapsos de tiempo.
- La utilización promedio de los procesadores de este servidor es de 25 %

7.3.2.2 Área de almacenamiento de respaldo (swap)

Como comentamos anteriormente, la utilización del área de **swap** nos puede proporcionar un parámetro confiable de la manera en que se esta utilizando la memoria primaria **RAM**.

A continuación se presentarán algunas gráficas que muestran el comportamiento del área de almacenamiento **swap** de este servidor.

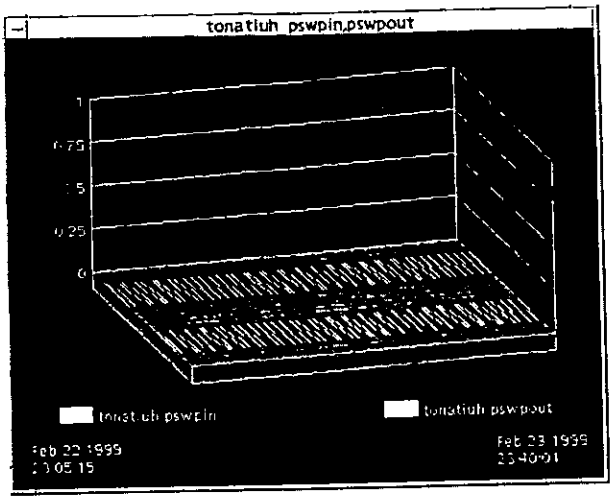


Figura 7.23 Comportamiento del área de almacenamiento de respaldo del servidor TONATIUH, en una muestra promedio, en el mes de Febrero de 1999.

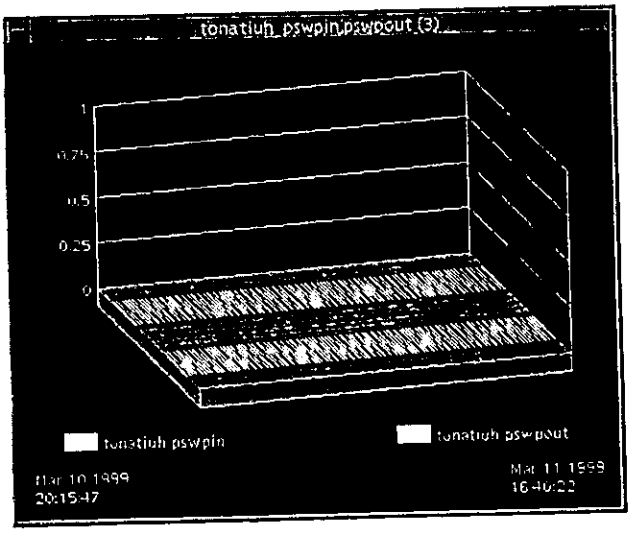


Figura 7.24 Comportamiento del área de almacenamiento de respaldo del servidor TONATIUH, en una muestra promedio, en el mes de Marzo 1999

Las gráficas anteriores fueron extraídas del total de muestras obtenidas en el período de monitoreo anteriormente mencionado. De las gráficas obtenidas en todo el período pudimos observar que la transferencia de páginas entre la memoria primaria **RAM** y el área de almacenamiento de respaldo es nula, lo que significa que los procesos son manejados apropiadamente por la memoria primaria del servidor TONATIUH.

7.3.2.3 Disco

EL servidor TONATIUH cuenta con cuatro discos, a continuación se presentarán las gráficas que muestran la utilización de cada uno de estos discos. Las gráficas representan a cada disco con sus respectivas particiones, mostrando el total de espacio asignado, usado y disponible por partición en Mbytes.

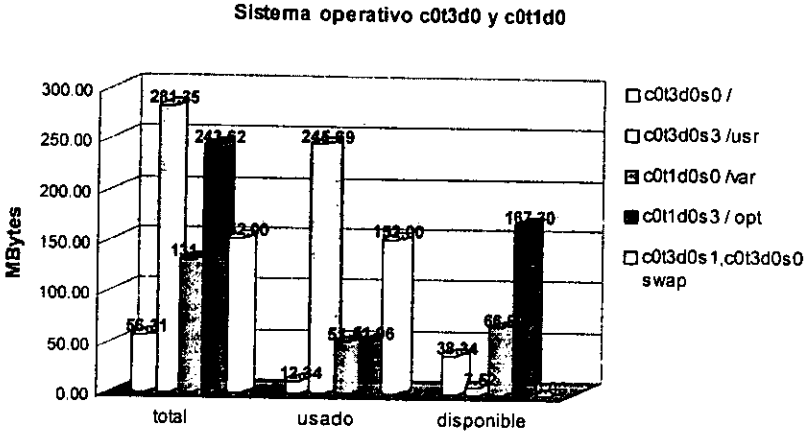
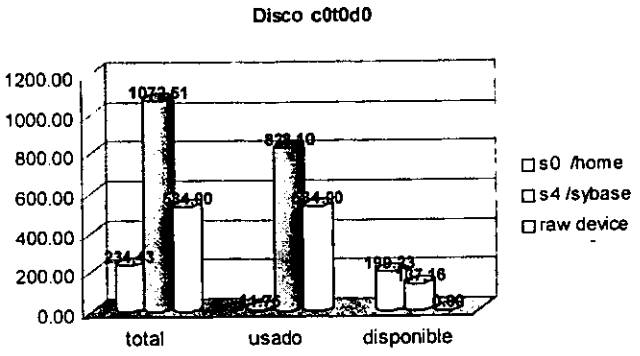


Figura 7.25 Utilización de los discos asignados para el sistema operativo del servidor TONATIUH



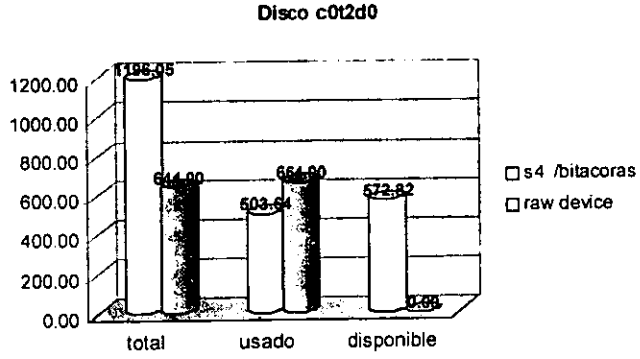


Figura 7.26, 7.27 Utilización de discos del servidor TONATIUH

Disco	Total usado	Total disponible	Uso
.c0t3d0 y c0t1d0, sistema operativo y swap	70%	30%	Sistema operativo y una partición de cada disco para swap.
.c0t0d0 /home, /Sybase y raw device	82%	18%	Usuarios, instalación del producto Sybase, espacio en raw device para la base de datos
.c0t2d0 /bitacoras y raw device	69%	31%	Bitácoras, espacio de disco en raw device para la base de datos.

NOTA: Las particiones definidas como área de **swap**, son mostradas como totalmente utilizadas, debido a que este espacio no se puede usar para almacenar información, sin embargo la utilización de las áreas de **swap** es de un 1% promedio.

Tabla 7.3 Información de la utilización de todos los discos del servidor TONATIUH

La información referente a los discos se obtuvo por medio de un **MIB** apropiada, como en el caso de PUMAS se tuvo que realizar las gráficas en Excel debido que las gráficas que generaba **SUNnet Manager** con esta información eran poco explicitas.

7.3.2.4 Red: Colisiones.

En el período de monitoreo Febrero – Marzo 1999, se recopilo información referente a la interfaz de red del servidor TONATIUH. En esta sección del capítulo mostraremos el nivel de porcentaje de colisiones encontrados.

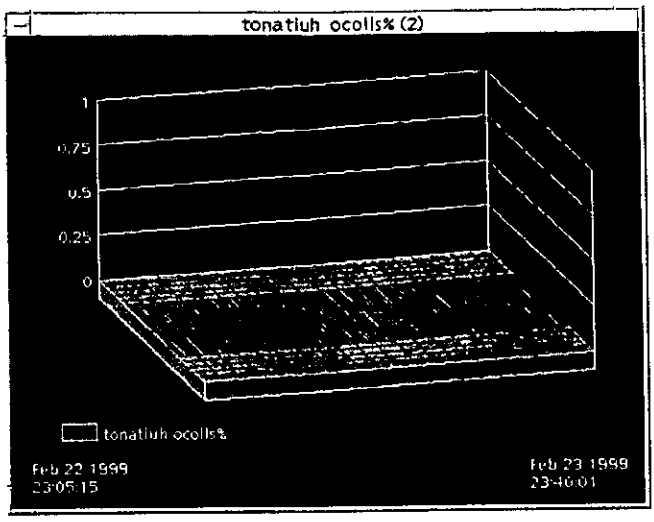


Figura 7.28 Gráfica que muestra las colisiones en la interfaz de red del servidor TONATIUH, muestra promedio

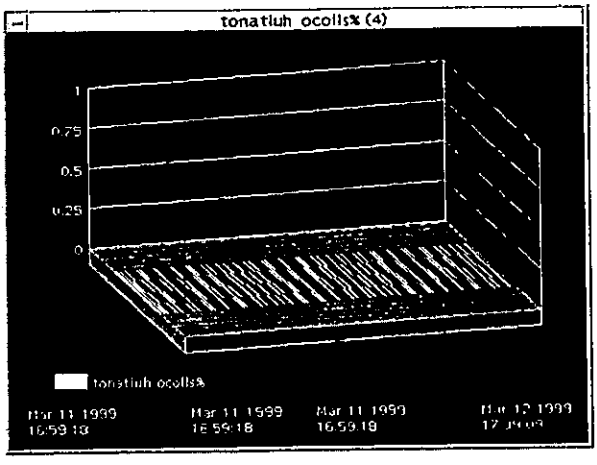


Figura 7.29 Gráfica que muestra el nivel de colisiones en la interfaz de red del servidor TONATIUH, muestra promedio.

Estas gráficas fueron tomadas del total de muestras obtenidas en el período de monitoreo establecido. De todas la muestras obtenidas en este período se observo que el porcentaje de

colisiones promedio en la interfaz del servidor TONATIUH es del 0% a cualquier hora del día incluso en las horas donde este servidor tiene más carga de usuarios.

7.3.2.5 Conclusiones

El primer parámetro de desempeño del servidor TONATIUH que se examino fue la utilización de sus procesadores, de los resultados obtenidos podemos concluir que el 25 % de uso promedio en los procesadores es un nivel aceptable, no necesitando un incremento ni en velocidad ni en número de los mismos a pesar de que en algunos casos el uso de los procesadores llega a ser de 100%, esto no representa ningún problema en el desempeño debido a que no esta condición no se presenta con regularidad.

Otro punto examinado fue la utilización del área de almacenamiento de respaldo o **swap**, en donde pudimos observar que el intercambio entre la memoria primaria y el área de **swap** es nulo, con lo que podemos concluir que el servidor TONATIUH cuenta con la memoria suficiente para atender todos sus procesos.

En cuanto a la utilización de los discos del servidor TONATIUH, encontramos un problema ya que todos los discos presentan menos de 30% de espacio disponible. En el servidor TONATIUH se desarrollan bases de datos, lo cual demanda un incremento en espacio de disco, es aconsejable aumentar el número de discos, considerando las necesidades futuras de las bases de datos.

El porcentaje de colisiones encontrado en el servidor es de 0%, el cual es un buen parámetro de desempeño.

7.3.3 Desempeño de servidores críticos del Instituto de Ingeniería.

En los meses de Febrero y Marzo de 1999 además de realizar el monitoreo de los servidores PUMAS y TONATIUH, también se llevo a cabo el monitoreo del resto de los servidores del Instituto de Ingeniería. A continuación presentaremos una tabla que resume los resultados obtenidos del monitoreo de servidores críticos en el período antes mencionado.

Servidor Coordinación	Utilización de procesadores.	Utilización de área de swap.	Porcentaje de colisiones	Observaciones
Gea, sismología e instrumentación sísmica	Utilización promedio del 15%, con picos de 50% al 100%	Intercambio entre memoria primaria y área de swap nulo.	Promedio de 15%, con picos que rebasan el 30%	Gea es un servidor con un buen desempeño, sin embargo es recomendable realizar una actualización de todo el servidor debido a que es una arquitectura antigua, con esta actualización se podrá abatir los picos altos de utilización de procesador. También es recomendable el incremento del espacio en disco.
Tlaloc, Coordinación de sistemas de cómputo	Utilización promedio del 1%, las	Intercambio entre memoria primaria y área de swap nulo.	Promedio de 0%, con picos de 10% a 25% que no afectan el desempeño.	El servidor Tlaloc, no tiene problemas de desempeño.Es recomendable implementar nuevos servicios en este servidor con el fin de aprovecharlo mejor.
Esma, Estructuras y materiales.	Utilización promedio del 15% con picos del 75% a 100%	Intercambio entre memoria primaria y área de swap constante.	Promedio de colisiones 15%.	En este servidor se observa una gran actividad entre la memoria primaria y el área de swap es recomendable analizar las aplicaciones o programas desarrollados por los usuarios, si el problema persiste se debe incrementar la memoria del servidor en base a los requerimientos de las aplicaciones o programas.
Leviatán, Mecánica aplicada	Utilización promedio del 10%, con picos el 50% a 100%en los períodos de máxima actividad	Intercambio entre memoria primaria y área de swap nulo.	Promedio de colisiones 20%, con picos mayores al 30%	Es recomendable el incremento de la velocidad del procesador, así como el incremento en el espacio en disco.
Inti, Ingeniería sismológica	Utilización promedio del 25%	Intercambio entre memoria primaria y área de swap nulo	Promedio de colisiones 15% con picos hasta de 30%	Servidor con un buen desempeño.
Hermes, Ingeniería sismológica	Utilización promedio del 19% con picos de hasta el 60 %	Intercambio entre memoria primaria y área de swap nulo	Promedio de colisiones 0%	Servidor con un buen desempeño.
Vortex, Hidráulica	Utilización promedio del 0%	Intercambio entre memoria primaria y área de swap nulo	Promedio de colisiones 0%	Servidor con un buen desempeño

Tabla 7.4 Desempeño de los servidores críticos del Instituto de ingeniería

7.4 Comportamiento del servidor de Web del Instituto de Ingeniería.

El estudio del servidor de **Web** del Instituto de Ingeniería, esta enfocado a conocer el nivel de acceso o utilización de los usuarios tanto internos como externos, los resultados serán parámetros fiables para evaluar la utilidad de este servicio.

Para llevar a cabo esta parte del estudio, utilizamos el software **http- analyze** versión 2.0, el cual fue descrito en el capitulo anterior.

Explicaremos brevemente como funciona un servidor de **Web** antes de exponer los resultados obtenidos con el fin de que se entiendan mejor.

Un servidor de **Web** es un programa que esta corriendo permanentemente en una maquina conectada a una red (puede ser **Internet** o una Intranet[5]), dicho programa esta en espera de conexiones externas para servir ciertos documentos requeridos por un navegador. Para comunicarse el servidor de **Web** y el navegador usan un protocolo llamado HyperText Transfer Protocol mejor conocido como HTTP, este protocolo funciona de la siguiente manera:

1. - El usuario inicializa un navegador e introduce en el una dirección URL[6]
2. - El navegador se conecta con la maquina en donde esta corriendo el servidor de **Web**.
3. - El servidor de **Web** maneja la petición del navegador y le envía una respuesta.
    a. Si el documento existe, el servidor de **Web** lo proporciona al navegador.
    b. Si el documento no existe, entonces el servidor envía al navegador un mensaje de error.

El documento que el servidor da como respuesta a la petición del navegador contiene objetos *en línea*, estos objetos en línea (un documento, una imagen, un programa, un archivo audio – video, etc.) son sencillamente hipervínculos apuntando a otros recursos en el mismo servidor o en otros.
El tipo de comunicación que establece un servidor **Web** con un navegador, se le conoce como asíncrona, ya que el navegador envía muchas peticiones para los objetos en línea de un documento en una sola petición, usando diferentes canales.

Es necesario explicar también algunos términos que los servidores de **Web** manejan, y que **http – analyze** utiliza para presentar los resultados obtenidos de los archivos log del servidor de **Web**. Por lo que se elaboró la tabla que se presenta a continuación.

[5] Una red es llamada Intranet, cuando provee los mismos servicio que Internet a una institución o empresa, pero de manera aislada.

[6] URL significa Localizador de recursos uniforme. Una dirección URL, es una codificación del destino de una liga virtual o hipervínculo. Por ejemplo: http://pumas.iingen.unam.mx/index.html.

Término	Significado
Hits	Un **hit**, es cualquier respuesta del servidor a una petición enviada por un navegador. Por ejemplo si una pagina HTML en el servidor incluye dos imágenes, el servidor generará tres **hits** si esta pagina es requerida por un navegador.
Files	Si un usuario desde un navegador pide al servidor un documento y este se lo envía satisfactoriamente, entonces esta respuesta es contada como un **File**. Cualquier tipo de archivo enviado es tomado como un **File**
Code 304	Un **Code 304** (Código 304 = No modificado) es generado por el servidor de **Web** cuando un documento no ha sido modificado desde la ultima vez que este fue requerido por un usuario, por lo tanto no hay necesidad de enviar los archivos que conforman este documento. Esta técnica es usada para reducir el trafico en la red.
Pageviews	Los archivos que tienen el sufijo .html o .text, o los archivos índices de un directorio, son contabilizados como pageviews. Estos pageviews no contienen imágenes, programas CGI, java applets, o cualquier otro objeto HTML. EL número de pageviews permite estimar el número de documentos reales transmitidos por el servidor de **Web**.
Kbytes sent	Total de datos transferidos por el servidor, en un período de tiempo dado.
Sessions	Es la suma de todos los nodos únicos, que tuvieron acceso al servidor en una ventana de tiempo dada.

Tabla 7.5 Nomenclatura utilizada por http-analyze.

A continuación explicaremos los resultados obtenidos para el servidor de **Web** del Instituto de Ingeniería. Http – analyze, nos permitió analizar el log del servidor de **Web** en un período de 9 meses, el cual comprende desde el mes de Agosto de 1998 hasta el mes de abril de 1999. Dividimos este período de tiempo en dos: 1998 y 1999. En este capítulo, solo se presentarán los meses con mas carga de cada uno de los períodos antes mencionados, debido a que son los más significativos, estos meses son: Septiembre de 1998 y Marzo de 1999. El análisis para cada uno de los períodos, esta dividido en 6 etapas:

- **WWW Access Statistics for 1998 o 1999**. Se muestran los resultados obtenidos por mes en el servidor de **Web** del II. Presentando el total del **Hits**, **Files**, **Pageviews**, **Sessions** y **Kbytes sent**.
- **Hits by day**. A partir de esta etapa todos los resultados se enfocan al mes elegido como el más representativo para cada período. En ellos podemos observar una gráfica, que muestra los valores (**Hits**, **Files**, **Pageviews**, **Sessions** y **Kbytes sent**.) por día. También podemos observar una tabla con el resumen de los valores mencionados anteriormente en cada día del mes de septiembre 1998 y marzo 1999.

- **Average load.** En esta etapa se exponen gráficas que representan el porcentaje de **Hits**, **Files** y **Pageviews** por día y en la ultima semana del mes en cuestión. Se presenta una tabla con los días más representativos del mes, mostrando los valores de **Hits**, **304´s**, **Kbytes sent** para cada uno de estos días. Una gráfica muy importante es la que muestra el porcentaje de **Hits** por hora en el mes, con esta gráfica podemos la distribución de la carga del servidor de **Web** en un día promedio. En esta etapa se agregan también tablas que muestran las 24 horas en las cuales se observo mas carga en todo el mes, así como los 5 minutos y 5 segundos mas cargados del mes.
- **Hits by Country.** En esta etapa se presenta un gráfica tipo pie en donde se muestra el porcentaje de **hits** realizados por país. Junto a esta grafica también se incluye una tabla que lista los países de donde se han hecho accesos al servidor ordenados de manera decreciente por número de **hits**, **304´s**, generados y **Kbytes** transmitidos.
- **The Top 30 items/URLS.** En esta etapa se presenta una gráfica tipo pie, donde se muestran las direcciones URL visitadas dentro del servidor. Se expone una tabla con las 30 direcciones URL mas visitadas ordenas de la mas visitada a la menos visitada. También se incluye una tabla con las 10 direcciones URL menos visitadas.
- **The Top 30 client domains.** Se presenta una gráfica de pie donde se muestran los dominios desde donde se han hecho accesos al servidor de **Web**, estos son representados junto con los porcentajes de acceso que han realizado. También se incluye una tabla que lista los 30 dominios desde donde más se han hecho accesos al servidor de **Web** del Instituto, ordenados de manera decreciente por número de **hits**, **304´s**, generados y **Kbytes** transmitidos.

En el apéndice B se muestran los resultados obtenidos del servidor del **Web** del Instituto de Ingeniería.

7.4.1 Conclusiones.

Después de observar las estadísticas generadas por el software http-analyze, podemos llegar a las siguientes conclusiones:

- Las horas que presenta mas actividad el servidor de **Web** del instituto de Ingeniería, son: 8:30 a.m. a 16:00 p.m. y de las 17:00 a las 21:00. Sin embargo se pueden observar accesos al servidor en todas horas del día y la noche.

- El servidor de **Web**, transmite un promedio de 657757 Kbytes, al mes.

- El servidor de **Web**, es visitado desde más de 30 países en el mundo. Los países que han realizados más accesos son: México, Estados Unidos, España, Colombia, Francia, Argentina, Chile, Canadá, Italia, Japón. También se han tenido accesos de lugares remotos como Israel, Nueva Zelanda, Finlandia, Taiwán, Grecia, Sudáfrica, Turquía.

- Los dominios que más realizan accesos al servidor son: net.mx, unam.mx, pemex.com, com.mx itesm.mx, acnet.net, gob.mx, imp.mx, com.mx, dec.com. Como se puede observar estos dominios pertenecen a México, siendo las instituciones educativas y de investigación las que realizan más accesos, pero también hay compañías comerciales que se interesan en las actividades del Instituto.

- Las áreas del servidor de **Web** que son mas visitadas: Información general, Áreas de Investigación, Proyectos, Área de consultas, La torre de Ingeniería, Directorio del Instituto de Ingeniería, Estudios de Postgrado.

El servidor de **Web** del Instituto de Ingeniería de la UNAM, es una ventana de divulgación del las actividades y proyectos del Instituto de Ingeniería muy importante. El servidor de **Web** crece día con día, como podemos observar en las estadísticas presentadas el aumento de accesos denota un claro incremento desde los meses monitoreados en 1998 hasta los meses monitoreados en 1999. Lo que nos lleva a concluir que el servidor de **Web** del Instituto de Ingeniería es una poderosa herramienta de divulgación de las actividades y proyectos que realiza esta institución, además de servir como vinculo con otras instituciones de investigación y comerciales en el país y en varias partes del mundo.

7.5 Comportamiento del servidor de FTP del Instituto de Ingeniería.

Otro de los servicios electrónicos que el Instituto de Ingeniería provee, es el servidor de **FTP**. Esta parte del estudio esta enfocado a medir la utilización de dicho servidor, para esto nos apoyamos en el software **GWFstats** versión 1.1 descrito en el capítulo anterior.

GWFstats crea estadísticas alrededor de 2 tópicos principales: los **Hits** y el número de Kbytes transmitidos. En este caso el número de **Hits** se refiere al número de archivos que son accedidos por un usuario a través del servicio de **FTP**.

Los resultados que **GWFstats** genera están divididos en 14 etapas:

- **Index**. Presenta el índice de los resultados obtenidos por un período de tiempo dado.
- **Summary**. Es un resumen de las estadísticas generadas en un período de tiempo dado.
- **Daily number of Hits.** Gráfica que muestra el número de accesos (Hits), por día en el mes en cuestión.
- **Daily volume transferred.** Gráfica que muestra el volumen de Kbytes transferidos por día en el mes en cuestión.
- **Cumulative number of hits by hour of Day.** Gráfica que muestra la distribución del número de Hits por hora en un día promedio del mes.
- **Cumulative volume transferred by hour of day.** Gráfica que muestra la distribución del volumen de datos transferido por hora en un día promedio del mes.
- **Top 10 top level domains by number of hits.** Gráfica que muestra una lista de los 10 dominios desde donde se hacen mas accesos (hits) al servidor de **FTP**, ordenados de manera descendente.
- **Top 10 top level domains by volume transferred.** Gráfica que muestra a los dominios que transfieren más volumen de datos vía el servidor de **FTP**. Estos dominios son ordenados de manera descendente.
- **Top 10 archives by number of hits.** Gráfica que muestra los 10 directorios que presentan mas accesos, ordenados de manera descendente.
- **Top 10 archives by volume transferred.** Gráfica que presenta los 10 directorios de donde se han transmitidos mas datos vía servicio de **FTP**.
- **Top 10 files by number of hits.** Gráfica que muestra los 10 archivos que presentan mas accesos, ordenados de manera descendente.
- **Top 10 files by volume transferred.** Gráfica que presenta los 10 archivos de tamaño más grande que se han transferido vía servidor de **FTP**.
- **Top 10 hosts by number of hits.** Gráfica que muestra a los 10 nodos de red que han realizado más accesos al servidor, ordenados en forma descendente.
- **Top 10 hosts by volume transferred.** Gráfica que muestra los 10 nodos de red que haciendo uso del servicio de **FTP** han transferido más datos.

Con **GWFstats** se analizo el log que dejo el servidor de **FTP** del Instituto de Ingeniería, desde el mes de octubre de 1998 hasta el mes de abril de 1999. En el apéndice C se muestran los resultados obtenidos en el mes de marzo de 1999, ya que este mes es el más representativo de todo el período anteriormente descrito.

7.5.1 Conclusiones

Después de observar las estadísticas generadas por el software **GWFstats**, podemos concluir lo siguiente:

El servidor de **FTP** del Instituto de ingeniería es una herramienta muy útil. El horario en el que este servicio presenta actividad es de las 8:00 a.m. las 23:00 p.m., presentándose las horas pico entre las 14:00 a las 18:00. El servidor transfiere 654456.22 Kbytes y 4129 archivos promedio por mes. Aunque se cuenta con un servicio de **FTP** anónimo, podemos observar que los accesos más numerosos los realiza el personal que labora en el Instituto de Ingeniería, los archivos que más comúnmente se transfieren son de aplicaciones de utilidad como antivirus, navegadores de **Web**, aplicaciones para Windows, aplicaciones para **UNIX** etc., y archivos de datos de los usuarios. **GWFstats** constituye una herramienta muy útil para el administrador del servicio de **FTP**, ya que con ella puede dar seguimiento de quien esta haciendo uso indebido del servicio.

7.6 Comportamiento del servidor de correo electrónico del Instituto de Ingeniería.

Uno de los servicios electrónicos más importantes dentro de REDII es el correo electrónico. Para conocer el grado de utilización de este servicio nos apoyamos en dos programas de monitoreo: **MRTG** y **smtpstats**.

MRTG se basa en los datos estadísticos generados por la utilería de sistema operativo mailstats para generar sus resultados. Mailstats obtiene información varia acerca del servidor de correo electrónico (Sendmail), pero los datos que son utilizados por **MRTG** son el número de correos electrónicos enviados y recibidos por el servidor, estos datos son re-direccionados a **MRTG** cada 5 minutos. Con la información anteriormente descrita **MRTG** genera 4 graficas dentro de su reporte:

- **Gráfica Diaria (5 Minutos Promedio).** Esta gráfica se actualiza cada 5 minutos y en ella se muestra el comportamiento de la entrada y salida de correos electrónicos del servidor, se puede apreciar la distribución de los datos por hora. También se presenta el promedio de mensajes enviados y recibidos en los últimos 5 minutos, así como el número máximo de mensajes y la cantidad de correos electrónicos enviados y recibidos en el ultimo intervalo de tiempo monitoreado.
- **Gráfica Semanal (30 Minutos Promedio).** Esta gráfica se actualiza cada 30 minutos y nos presenta el comportamiento de la entrada y salida de correos electrónicos del servidor a través de los días de la semana. Se muestra también el número máximo de mensajes enviados y recibidos así como el promedio de correos electrónicos en los últimos 30 minutos además de la cantidad de correos en el último período monitoreado.
- **Gráfica Mensual (2 horas Promedio).** Esta gráfica nos muestra el comportamiento de la entrada y salida de correos electrónicos del servidor durante las últimas 6 semanas y se actualiza cada 2 horas. De la misma manera que en los puntos anteriores también se muestra el promedio de correos en las ultimas 2 horas y el número máximo de correos, así como la cantidad de correos en el ultimo período de monitoreo actualizado.

- **Gráfica Anual (1 Día promedio).** Finalmente esta gráfica, se actualiza diariamente y presenta la distribución de correos por mes, proporcionando también el valor promedio de los correos que entran y salen del servidor en un día, así como los valores máximos encontrados y los valores encontrados en el último período de monitoreo actualizado.

MRTG, nos proporciona información de mucha utilidad que nos permiten visualizar de manera gráfica el comportamiento de los correos electrónicos que maneja el servidor en varios intervalos de tiempo. Sin embargo surgió la necesidad de contar con datos que nos dieran mas información acerca del servidor de correo electrónico, es por eso que se uso el programa de dominio público **smtpstats**, el cual proporciona información mas especifica. Smtpstats genera un reporte en el que muestra el número total de mensajes manejados por el servidor, el número bytes y recipientes[7] manejados, este reporte esta dividido en tres partes:

- **Parte I (Mail relayed from).** En esta parte del reporte se presenta una lista de las computadoras que enviaron e-mail a través de nuestro servidor de correo.

- **Parte II (Mail sent from).** En esta parte del reporte se presenta una lista de los servidores desde donde fueron enviados correos electrónicos a nuestro servidor. Los servidores están listados de manera descendente desde el servidor que más correos envió hasta el que menos.

- **Parte III (Mail sent to).** En esta parte del reporte se presenta una lista de los servidores a los que fueron enviados correos electrónicos desde nuestro servidor. Los servidores están listados de manera descendente desde el servidor que más correos recibió hasta el que menos.

A continuación se muestran los resultados obtenidos con **MRTG** y **smtpstats** en el mes de Marzo de 1999, para ejemplificar de que manera se realiza el monitoreo del correo electrónico mensualmente.

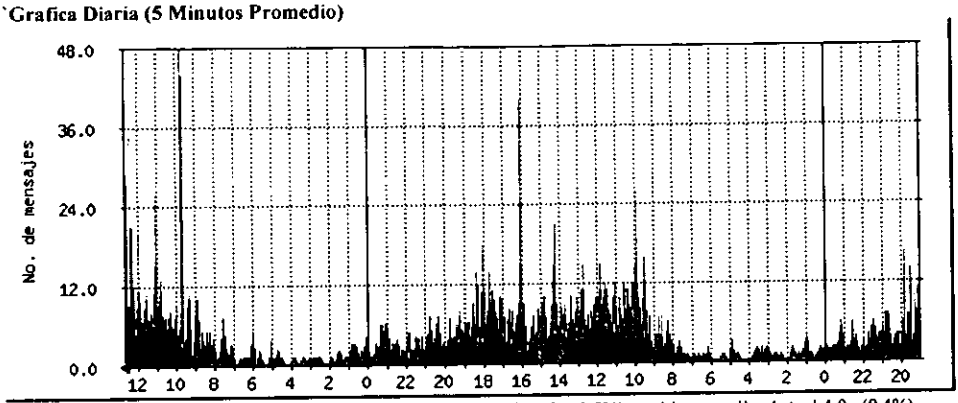
[7] Un recipiente es una dirección electrónica desde donde se ha recibido o enviado un correo electrónico

Instituto de Ingeniería UNAM, Coord. de Sistemas de Computo

Sistema: PUMAS
Arquitectura: SS20. solaris 2.5.1
Velocidad Maxima: 1250.0 kBytes/s (Ethernet-Csma/Cd)

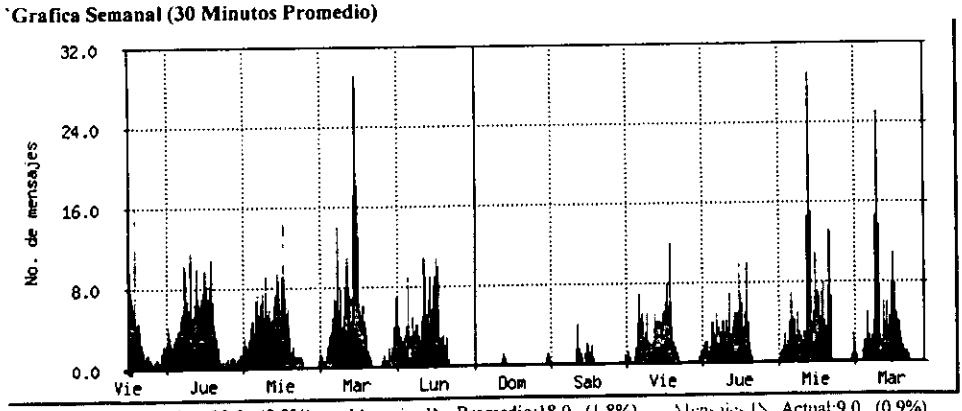
Las estadisticas se han actualizado al **Viernes, 19 Marzo 1999 a las 12:40**

`Grafica Diaria (5 Minutos Promedio)



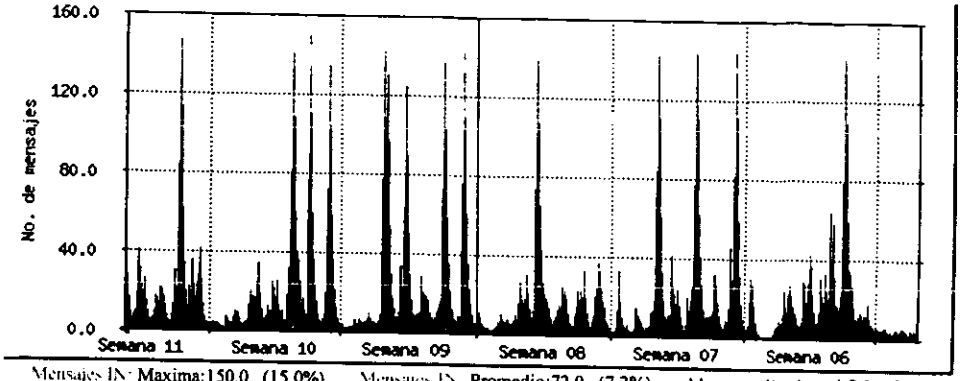
Mensajes IN Maxima:48.0 (4.8%) Mensajes IN: Promedio:5.0 (0.5%) Mensajes IN Actual:4.0 (0.4%)
Mensajes OUT: Maxima:15.0 (1.5%) Mensajes OUT: Promedio:3.0 (0.3%) Mensajes OUT: Actual:4.0 (0.4%)

`Grafica Semanal (30 Minutos Promedio)



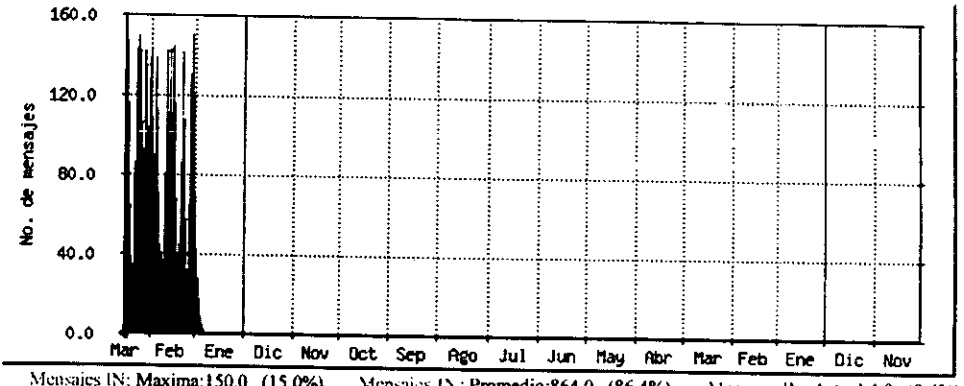
Mensajes IN Maxima:29.0 (2.9%) Mensajes IN Promedio:18.0 (1.8%) Mensajes IN Actual:9.0 (0.9%)
Mensajes OUT: Maxima:29.0 (2.9%) Mensajes OUT: Promedio: 6.0 (0.6%) Mensajes OUT: Actual:7.0 (0.7%)

`Grafica Mensual (2 Horas Promedio)



Mensajes IN: Maxima:150.0 (15.0%) Mensajes IN: Promedio:72.0 (7.2%) Mensajes IN Actual:7.0 (0.7%)
Mensajes OUT: Maxima:147.0 (14.7%) Mensajes OUT: Promedio:24.0 (2.4%) Mensajes OUT: Actual:3.0 (0.3%)

Grafica Anual (1 Dia Promedio)



Mensajes IN: Maxima:150.0 (15.0%) Mensajes IN: Promedio:864.0 (86.4%) Mensajes IN Actual:4.0 (0.4%)
Mensajes OUT: Maxima:150.0 (15.0%) Mensajes OUT: Promedio:288.0 (28.8%) Mensajes OUT: Actual:3.0 (0.3%)

VERDE ### Trafico de Entrada en Bytes por segundo

AZUL ### Trafico de Salida en Bytes por segundo

VERDE OSCURO### Maximo 5 Minutos de Trafico de Entrada

MAGENTA### Maximo 5 Minutos de Trafico de Salida

MULTI ROUTER TRAFFIC GRAPHER

2.5.1.sp- Tobias Oetiker <oetiker@ee.ethz.ch> and Dave Rand <dlr@bungi.com>
1997/10/24

Reporte generado con SMTPSTATS en el mes de marzo de 1999.

```
Total messages   handled:  32738
Total recipients handled:  31122
Total bytes      handled:  1808.09M
-------------------------------------------------------------
Part I -- Mail relayed from:
-------------------------------------------------------------
4011 pumas
1640 clubdelphi.com
1008 euro.planet.com.mx
 584 boreas.planet.com.mx
 470 tardis-delek-fast.ee.ethz.ch
 463 servidor.unam.mx
 297 [132.248.155.48]
 252 [132.248.156.163]
 228 wetheliving.com
 223 mezcal.super.unam.mx
 190 uxmcc2.iimas.unam.mx
 185 [132.248.155.114]
 181 [132.248.156.211]
 151 listproc3.pcworld.com
 146 eb0.ciateq.mx
 124 [132.248.155.191]
 125 m4.egroups.com
 120 peyote-asesino.nuclecu.unam.mx
 114 goku.telmex.net.mx
  99 screamer.xnet2.com
  85 home.ease.lsoft.com
  82 titan.iingen.unam.mx
  73 [132.248.155.151]
  71 md.egroups.com
  70 euler.iingen.unam.mx
  65 maelstrom.stjohns.edu
  53 sankara.midgard.kth.se
  51 [132.248.155.140]
  49 cicl.iimas.unam.mx
  49 [132.248.237.27]
  46 [132.248.155.140]
  45 sivalinga.midgard.kth.se
  39 sol.ucdavis.edu
  37 datasys.com.mx
  35 mango.ease.lsoft.com
  35 [132.248.155.136]
  33 lists.village.virginia.edu
  32 cs7-10.modems.unam.mx
  31 zephyr.isi.edu
  11 [132.248.156.34]
  10 ica.com.mx
  10 cicl.iimas.unam.mx
   9 abnt.abits.com
   9 [207.248.1.20]
```

```
Part II -- Mail sent from:
------------------------------------------------------------------
6188 pumas.iingen.unam.mx
4011 pumas
1640 clubdelphi.com
1426 foros.planet.com.mx
 470 list.ee.ethz.ch
 439 hotmail.com
 433 servidor.unam.mx
 330 tipworld.com
 325 yahoo.com
 312 home.ease.lsoft.com
 294 NULL sender
 281 returns.egroups.com
 228 wetheliving.com
 223 listas.unam.mx
 151 athena.nuclecu.unam.mx
 146 cideteq.mx
 141 screamer.xnet2.com
 121 cicl.iimas.unam.mx
 107 dfl.telmex.net.mx
  95 maelstrom.stjohns.edu
  73 uxmcc2.iimas.unam.mx
  73 lists.village.virginia.edu
  57 b.xoom.com
  56 euro.planet.com.mx
  46 aom.kth.se
  41 boreas.planet.com.mx
  39 infonavit.gob.mx
  37 datasys.com.mx
  10 abits.com
   7 news.newswire.microsoft.com
   7 chollian.net
```

Part III -- Mail sent to:		Avg delay	Max delay
15679	pumas.iingen.unam.mx	64.96 secs	8.31 mins
3011	pumas	32.60 secs	5.60 mins
502	servidor.unam.mx	22.48 mins	13.20 hrs
290	yahoo.com	28.60 mins	16.57 hrs
202	cem.iingen.unam.mx	1.33 mins	1.58 mins
172	merlin.iingen.unam.mx	16.56 secs	4.48 mins
147	dfl.telmex.net.mx	1.27 mins	1.78 hrs
120	gea.iingen.unam.mx	17.59 secs	37.00 secs
104	mail.internet.com.mx	8.45 mins	22.77 mins
102	euler.iingen.unam.mx	6.57 secs	15.00 secs
78	cibnor.mx	2.38 mins	10.82 hrs
73	hotmail.com	11.84 mins	35.80 mins
72	infosel.net.mx	6.13 mins	8.55 mins
67	tuxcom.net.mx	2.30 mins	8.33 mins
59	syscase.com.mx	41.52 mins	11.99 hrs
58	bah.com	2.36 mins	4.52 mins
54	int.mri.gouv.qc.ca	3.04 mins	17.35 mins
53	vortex.iingen.unam.mx	2.26 mins	9.87 hrs
38	usa.net	9.11 mins	12.18 mins
36	blues.iingen.unam.mx	3.7 secs	6.00 secs
36	aol.com	32.19 mins	9.70 hrs
31	quetzal.iingen.unam.mx	2.00 mins	3.90 hrs
30	t-online.de	32.3 secs	48.00 secs
25	mri.gouv.qc.ca	2.35 mins	14.10 mins
20	energia.gob.mx	44.6 secs	10.00 secs
20	usf.com.mx	16.25 secs	29.00 secs
17	compuserve.com	6.55 mins	50.68 mins
17	xanum.uam.mx	1.14 mins	6.68 mins
13	jazz.iingen.unam.mx	17.15 secs	3.37 mins
13	data.net.mx	11.09 mins	1.24 hrs
7	tonatiuh.igeofcu.unam.mx	3.43 secs	17.00 secs
5	ibm.net	7.00 secs	15.00 secs

7.6.1 Conclusiones.

Después de considerar las estadísticas generadas por **MRTG** y **smtpstats** podemos concluir lo siguiente:

- En el mes de Marzo de 1999 el servidor manejó 32738 mensajes, en donde se transmitieron 1808.9 Mbytes.

- El servidor de correo electrónico del Instituto de Ingeniería maneja un promedio de 1152 mensajes al día. Con un promedio de 62.6 Mbytes transmitidos.

- Existe actividad en el servidor desde las 24 horas del día, pero la mayor carga se presenta entre las 7:00 hrs. y las 23:30 hrs.

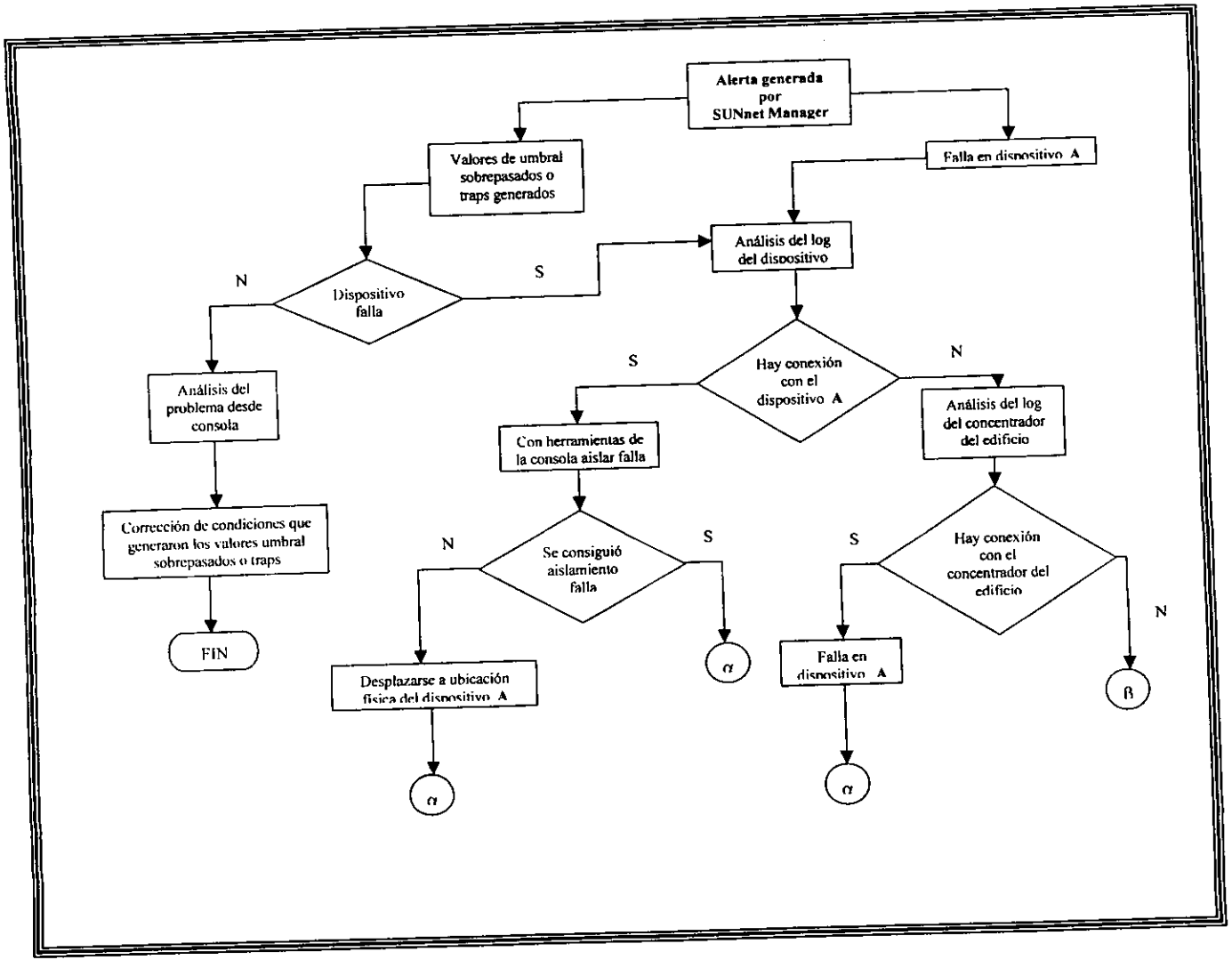
- Los servidores desde donde recibimos y enviamos más correo son: foros.planet.com.mx, clubdelphi.com, servidor.unam.mx, list.ee.ethz.ch, yahoo.com, dfl.telmex.net.mx y desde nuestro propio servidor de correo (pumas).

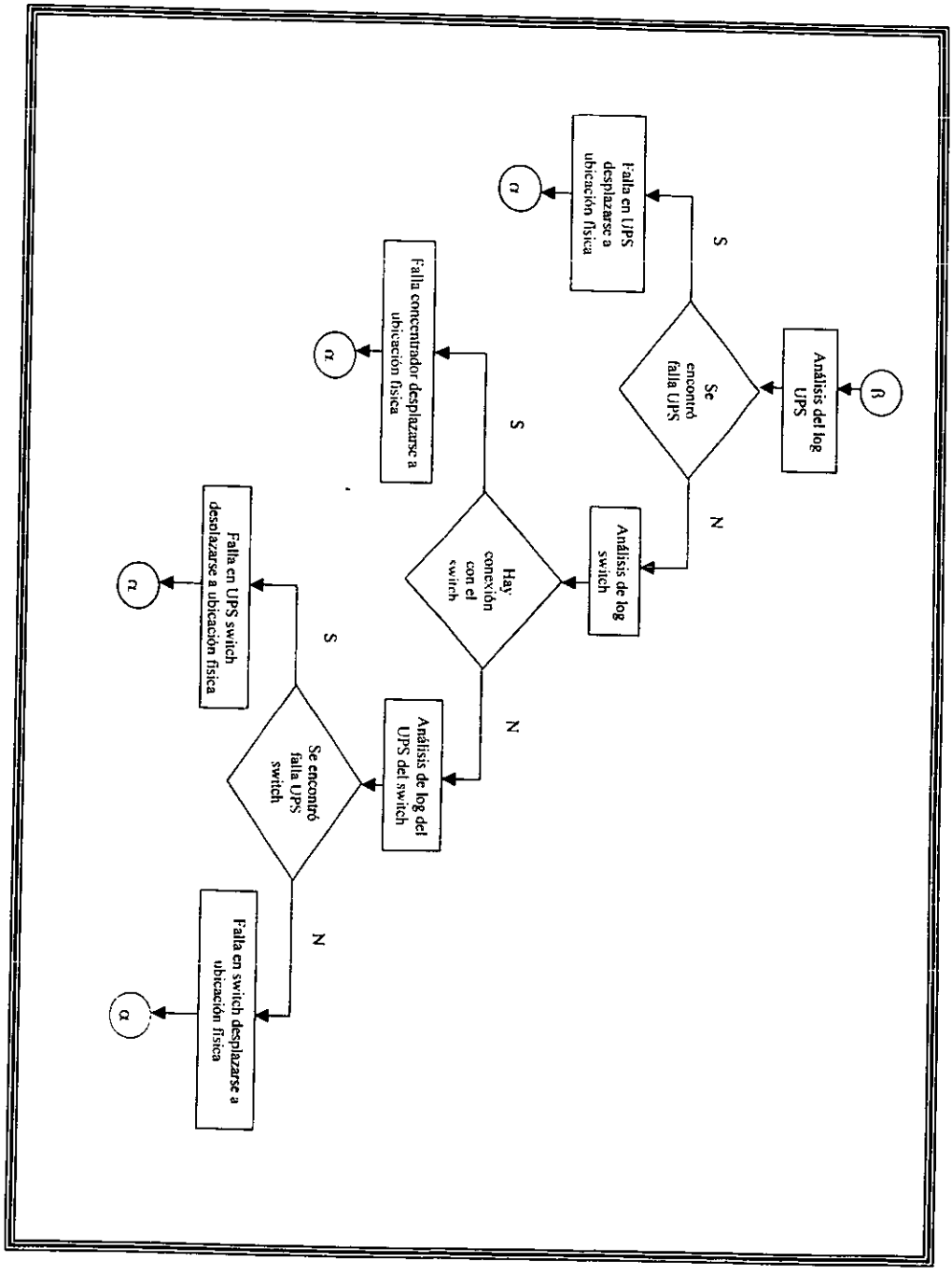
El servidor de correo electrónico es la herramienta electrónica más importante dentro del Instituto de Ingeniería, debido a que es un medio de comunicación fácil de usar, económico y rápido. Cada día se dan de alta más cuentas de correo en nuestro servidor para cubrir las necesidades de la comunidad del Instituto (Investigadores, becarios, etc.), siendo esta herramienta de gran ayuda en las investigaciones y en todas las actividades que se llevan a cabo.

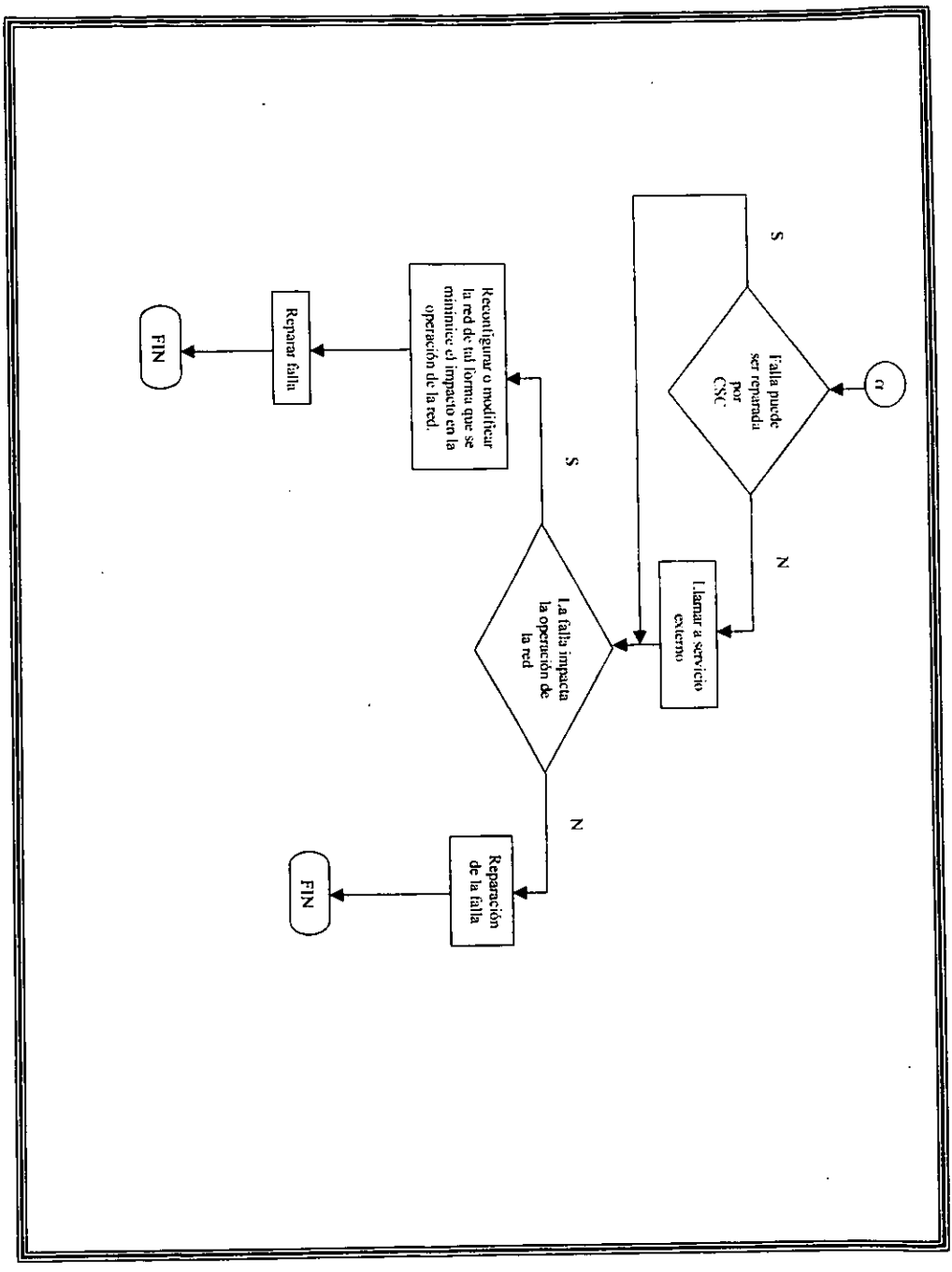
7.7 Procedimiento de detección y corrección de fallas

Antes de la implantación del sistema de monitoreo y administración de REDII, existía un procedimiento de detección y corrección de fallas el cual fue explicado en el capitulo 2, después de la implantación de dicho sistema fue necesaria la creación de un nuevo procedimiento, debido a que las condiciones en las que se detecta una falla cambiaron.

La tarea de detectar y corregir una falla podría parecer trivial, sin embargo es una de las prioridades en la administración de fallas. Si no se cuenta con un procedimiento definido para realizar esta tarea, las cosas pueden dificultarse y tener repercusión en la rapidez con la que se debe detectar, aislar y corregir una falla. El siguiente diagrama de flujo muestra el procedimiento de detección y corrección de fallas en el Instituto de Ingeniería con las herramientas de monitoreo implantadas.


Alerta generada por SUNnet Manager
Valores de umbral sobrepasados o traps generados
Falla en dispositivo A
Análisis del log del dispositivo
Dispositivo falla
N
S
Análisis del problema desde consola
Corrección de condiciones que generaron los valores umbral sobrepasados o traps
FIN
Hay conexión con el dispositivo A
S
N
Con herramientas de la consola aislar falla
Análisis del log del concentrador del edificio
Se consiguió aislamiento falla
N
S
Desplazarse a ubicación física del dispositivo A
α
Hay conexión con el concentrador del edificio
S
N
Falla en dispositivo A
α
ß


β
Análisis del log UPS
Se encontró falla UPS
S
Falla en UPS desplazarse a ubicación física
α
N
Análisis de log switch
Hay conexión con el switch
S
Falla concentrador desplazarse a ubicación física
α
N
Análisis de log del UPS del switch
Se encontró falla UPS switch
S
Falla en UPS switch desplazarse a ubicación física
α
N
Falla en switch desplazarse a ubicación física
α


a
Falla puede ser reparada por CSC
S
N
Llamar a servicio externo
La falla impacta la operación de la red
S
N
Reconfigurar o modificar la red de tal forma que se minimice el impacto en la operación de la red.
Reparar falla
FIN
Reparación de la falla
FIN

7.8 Perspectivas de desarrollo

El sistema de monitoreo y administración implantado cubre las necesidades actuales de REDII en este aspecto, sin embargo será necesario realizar variaciones en su arquitectura y en los elementos que lo conforman en un futuro con la finalidad de adaptarse a los cambios e incrementos que continuamente esta sufriendo REDII con la finalidad de cubrir las necesidades de sus usuarios.

También debe tomarse en cuenta la compatibilidad con el año 2000 del software **SUNnet Manager**, para que este software sea compatible deberá de realizarse la actualización a la versión **SUNnet/Site/Domain Manager** versión 2.3 revisión B, instalar el parche 104018-05.

En esta parte del capítulo se plantearan alternativas de desarrollo en el sistema de monitoreo y administración de REDII. Primeramente presentaremos el desempeño del nodo administrador basado en monitoreos realizados con **SUNnet Manager**.

7.8.1 Desempeño del nodo administrador Miktlan

El período en el que se realizo el monitoreo en el nodo administrador, fue el mes de Febrero y Marzo de 1999, al igual que en el resto de los servidores se obtuvieron muestras promedio de 24 horas, con intervalos entre **pollings** de 15 minutos. Las gráficas que se mostrarán a continuación fueron extraídas del total de gráficas obtenidas en el período de monitoreo antes mencionado, y muestran el comportamiento promedio en los diferentes tópicos de desempeño analizados.

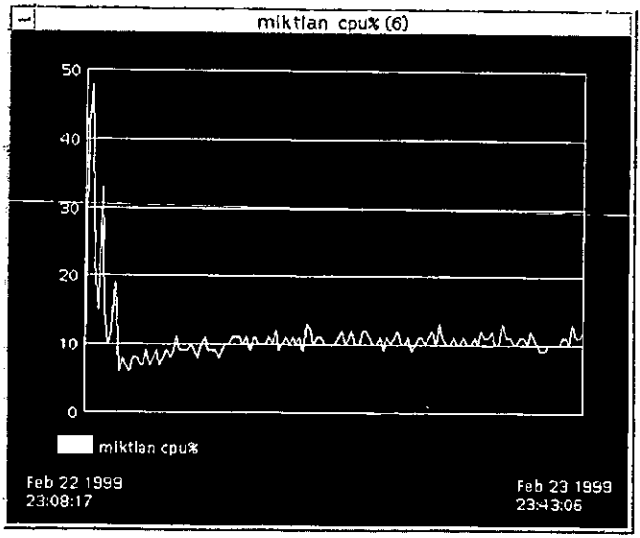


Figura 7.30 Gráfica que muestra la utilización del procesador del nodo administrador Miktlan.

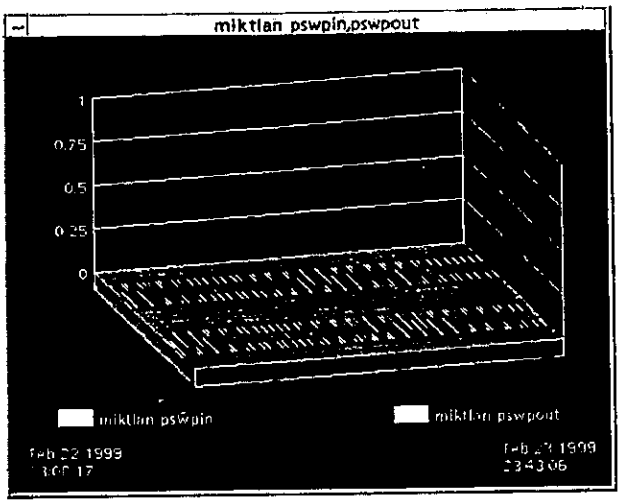


Figura 7.31 Utilización del área de swap del nodo administrador Miktlan.

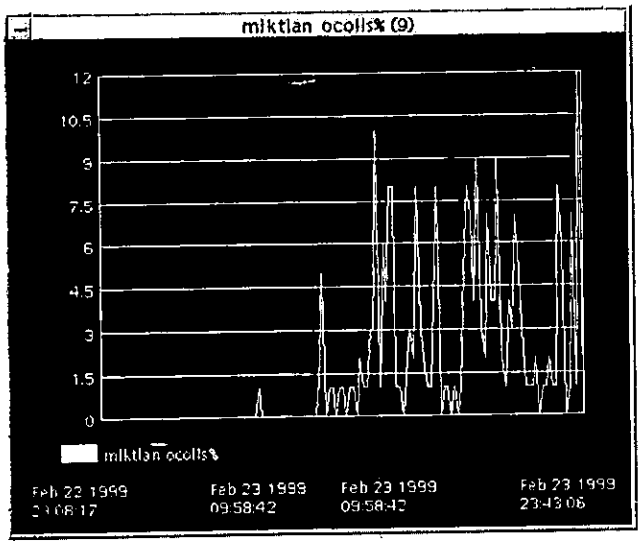


Figura 7.32 porcentaje de colisiones en la interfaz de red del nodo administrador Miktlan.

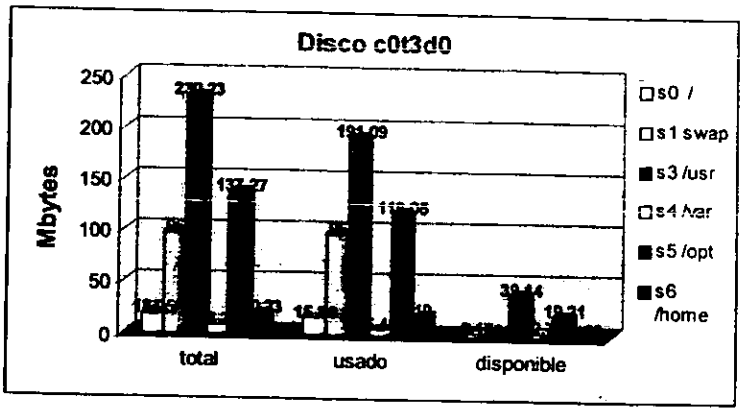


Figura 7.33 Utilización del disco del nodo administrador Miktlan

El porcentaje de utilización del procesador de este nodo es del 12.5%, presentando picos hasta del 50% que no impactan la operación ni el desempeño del mismo. Por otro lado la memoria del nodo es suficiente para manejar todos los procesos generados por la aplicación de monitoreo. El porcentaje de colisiones promedio en el nodo administrador es del 6% alcanzando picos hasta del 20%. Existe un problema grave con el espacio en disco, debido a la naturaleza de la información que se almacena en el nodo resulta insuficiente el espacio en disco, es recomendable adicionar un disco de 1.05 GB como mínimo para cubrir las necesidades existentes.

En general el desempeño del nodo administrador es bueno, sin embargo se realizaron pruebas de estrés para determinar hasta cuantas peticiones de monitoreo puede manejar el nodo administrador encontrando que con un incremento del 80% del total de las peticiones existentes, el nodo administrador tenía un impacto en su desempeño reflejándose en la velocidad en la que procesaba dichas peticiones además de que la mayoría de ellas se terminaban abruptamente mucho antes de que se cumpliera el tiempo establecido por muestra, dejando los datos en cada muestra truncados.

Actualmente el desempeño del nodo administrador es bueno y cubre las necesidades de procesamiento que las aplicaciones requieren para las peticiones de monitoreo hasta ahora establecidas.

7.8.2 Propuestas de desarrollo

Como pudimos observar a lo largo de esta tesis, un sistema de monitoreo y administración de Red no puede establecerse sobre la base de una sola aplicación, es necesaria la adecuación de diversas aplicaciones para cubrir el total de necesidades de un administrador o encargado de la red. De lo anterior se origina las siguientes propuestas de desarrollo del sistema de monitoreo y administración de REDII.

La arquitectura actual del sistema de monitoreo y administración de REDII es centralizada, para desarrollos futuros será necesario el cambio a la arquitectura distribuida, debido a que hay que cubrir más nodos administrados y para prevenir cuellos de botella en el nodo administrador.

La primera propuesta es una arquitectura distribuida que cuente con un nodo administrador principal donde este instaladas todas las aplicaciones de monitoreo y administración, además de los datos generados por dichas aplicaciones, también deberá contar con nodos administradores secundarios que únicamente tengan instalado el agente **snmp**, y que sirvan como **proxies**, estos nodos deberán estar instalados en diferentes subredes a la del nodo administrador. El objetivo de este esquema es que los nodos administradores secundarios actúen como un filtro del trafico de paquetes generados por el monitoreo, para evitar que todos los paquetes sean manejados por el nodo administrador principal.

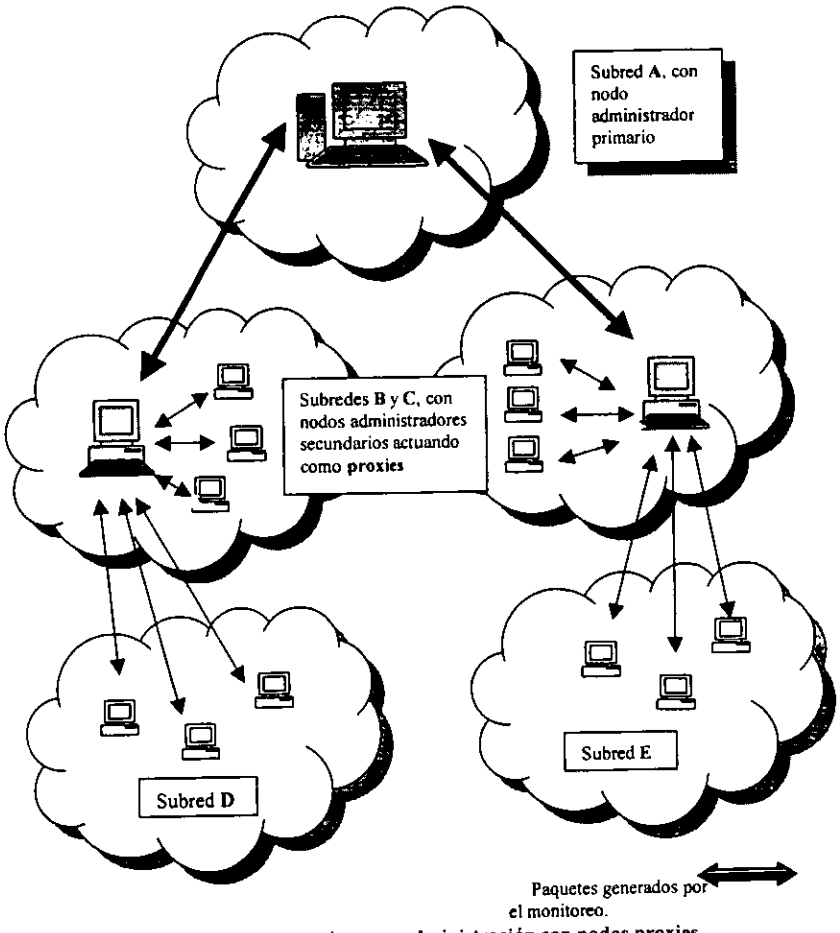


Figura 7.33 Sistema de monitoreo y administración con nodos proxies.

La segunda propuesta consiste en un sistema de monitoreo y administración con una arquitectura distribuida, que contenga dos o tres nodos administradores primarios en donde se incluyan tanto las aplicaciones de monitoreo y administración como los datos que estas generen. El objetivo de esta configuración es que los nodos administradores se distribuyan el trafico generado por el monitoreo, así como también los datos y el control de los nodos administrados. Es conveniente que uno de estos nodos administradores sea designado el nodo directivo y que desde este se pueda tener acceso al resto de los nodos administradores (información, peticiones de monitoreo, etc.) con el fin de establecer una base operativa dentro del sistema.

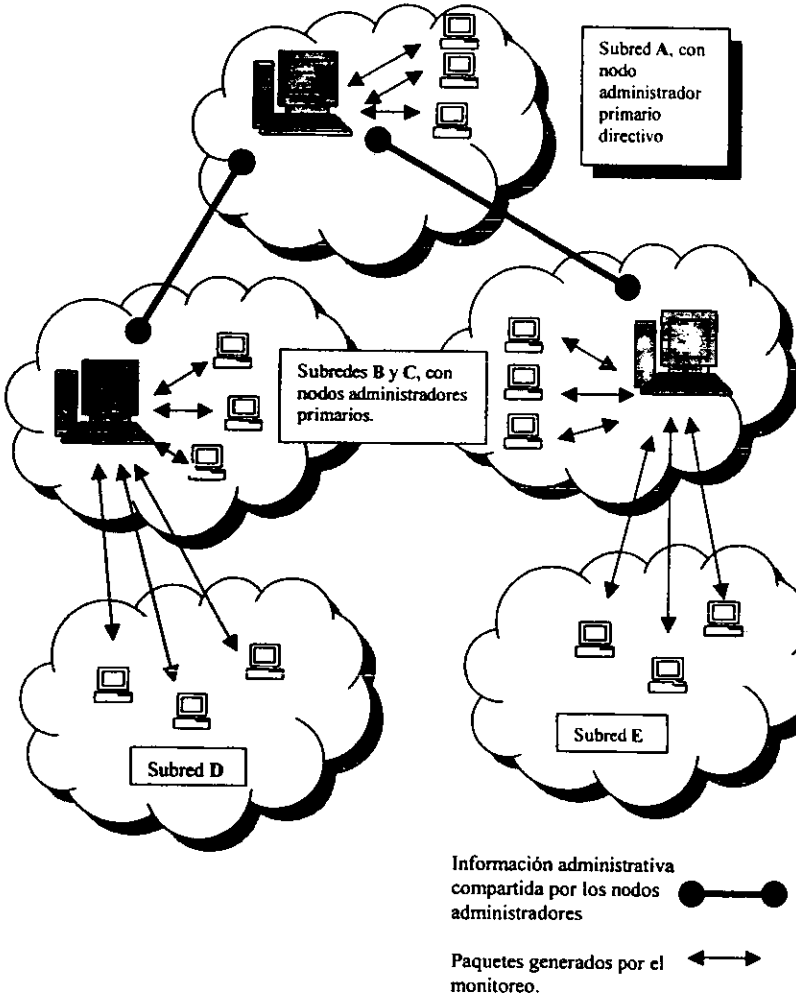


Figura 7.34 Sistema de administración y monitoreo, arquitectura distribuida con dos nodos primarios y un nodo primario directivo.

Una adición importante al sistema de monitoreo y administración sería el aviso de eventos y condiciones de falla a través de un radio localizador, y tendría que ser implementada en cualquiera de los dos esquemas propuestos.

Para implantar cualquiera de las anteriores propuestas será necesario considerar los servidores que serán los nodos administradores. Como vimos en los resultados obtenidos del monitoreo del servidor Miktlan, su desempeño es bueno. La segunda propuesta se podría implementar utilizando este tipo de servidores que existen en la actualidad en el Instituto de Ingeniería, será necesario agregar más espacio en disco a cada uno de los nodos administradores. La primera propuesta no necesita la utilización de otros dos servidores, debido a que cualquier servidor que tenga instalado el agente **snmp** puede fungir como agente **proxy**, sin embargo en este caso también será necesario incrementar el espacio en disco del nodo administrador primario.

Se debe tener en cuenta que el tipo de maquina que actualmente se esta usando como nodo administrador (**SPARClassic de SUN Microsystems**), puede crecer hasta los siguientes parámetros:

Número de procesadores	1 de 50 Mhz.
Memoria	96 Mbytes
Capacidad de disco	Un disco interno 1.05, y varios discos externos dependiendo del arreglo que se le instale

A continuación se plantean algunas propuestas para mejorar las maquinas que actuarían como nodos administradores.

- Aumentar los recursos con los que cuentan actualmente las maquinas **SPARClassic** ya existentes en el Instituto de Ingeniería
- Adquirir equipos con mayor capacidad
- Utilizar equipos con mayor capacidad ya existentes en el Instituto de Ingeniería reasignando las tareas que en ellos se ejecutan.

Conclusiones

La red del Instituto de Ingeniería ha ido creciendo no solo físicamente sino también en importancia, se ha convertido en la espina dorsal de los sistemas de información y comunicación los cuales son herramientas esenciales para realizar las labores que en la institución se llevan a cabo.

Al crecer la red del Instituto de Ingeniería REDII, se ha hecho más compleja soportando un mayor número de aplicaciones y usuarios. Con este crecimiento dos hechos se hacen cada vez más evidentes :

1. La red, sus recursos y servicios asociados se convirtieron en herramientas de trabajo indispensables para la institución

2. Existen más parámetros que pueden provocar fallas en la red, afectando seriamente su desempeño o bien en el caso extremo deshabilitando una porción o la totalidad de ella.

Una red grande no puede ser administrada únicamente por un humano, aunque este invierta todo su tiempo y esfuerzos en ello. La complejidad de las redes en la actualidad requieren el uso de herramientas automatizadas para el monitoreo y la administración de las mismas.

El objetivo principal de esta tesis fue lograr un alto nivel de confiabilidad, disponibilidad y eficiencia en REDII a través de la implantación de un sistema de monitoreo y administración el cual provea una fácil detección y corrección de fallas.

Para el desarrollo de este proyecto, se realizo el estudio de la situación de REDII antes de la implantación, así como una investigación bibliográfica extensa que nos permitiera comprender los estándares del monitoreo y la administración de una red, también se realizo la evaluación de varias aplicaciones comerciales de monitoreo y administración, y se instalaron algunas herramientas de dominio público para completar el sistema y adecuarlo a las necesidades especificas de REDII.

Finalmente se implanto un sistema de monitoreo y administración que cubre todos los recursos y servicios de REDII, basado en el protocolo de monitoreo estándar para **TCP/IP**: **SNMP**.

Los beneficios que se obtuvieron con la implantación del sistema mencionado anteriormente puede ser resumidos de la siguiente forma:

- Implantación de un procedimiento que permite detectar fallas en dispositivos de REDII de manera más rápida.

- Ahora se cuenta con herramientas que permitan aislar las fallas de manera remota.

- Se estandarizo el protocolo de monitoreo (**SNMP**) en REDII.

- Con la ayuda de las herramientas de monitoreo y administración se pueden detectar posibles fallas lo que nos lleva a realizar acciones pro-activas para evitar que esas posibles fallas se vuelvan reales.
- Se cuenta con historias estadísticas de los dispositivos, servicios y eventos de REDII.

- Se fortalece la relación entre el usuario final y el administrador o administradores de REDII, los cuales trabajan en pro de la comunidad del Instituto para proporcionar un mejor servicio.

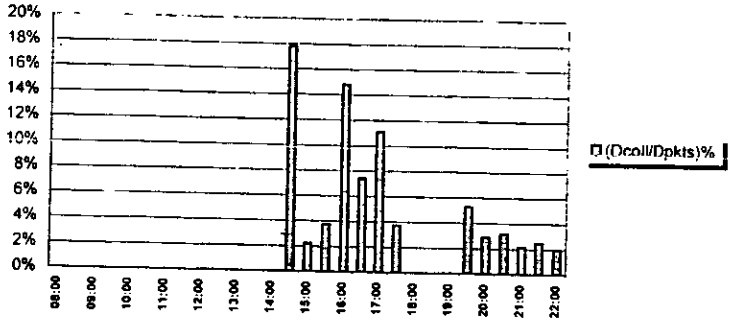
Con lo anterior se cumplieron los objetivos planteados al principio de este proyecto además de sentar las bases para que cada cambio en REDII sea ejecutado en base al estudio estadístico del comportamiento del dispositivo o configuración en cuestión.

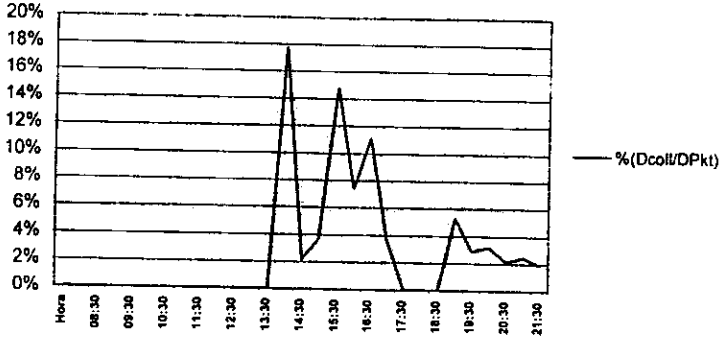
Este proyecto también da a conocer a las nuevas generaciones de administradores de red del Instituto de Ingeniería los diferentes protocolos de monitoreo existentes, así como a los principios teóricos de los sistemas de administración y monitoreo de redes.

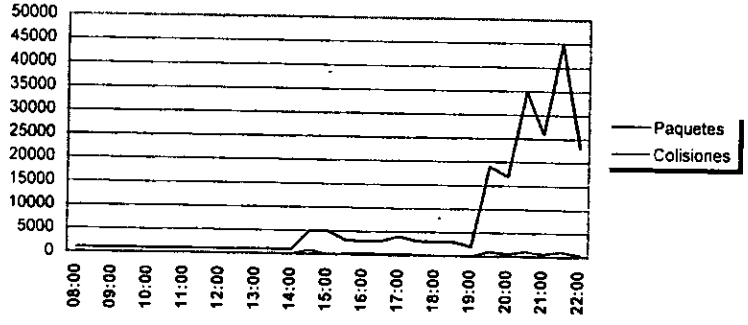
Todo proyecto de cómputo debido a sus características debe estar en continua evolución con la finalidad de mejorar los servicios que ofrece, es por ello que al establecer este sistema no se puede decir que el desarrollo del mismo ha finalizado. La implantación de este sistema, los mecanismos y las herramientas que lo conforman marcan la pauta para el desarrollo y la implantación de nuevas y más complejas herramientas en este ámbito dentro del Instituto de Ingeniería.

Apéndice A

Concentrador 12 Tarjeta 5, primera semana de Febrero 1999

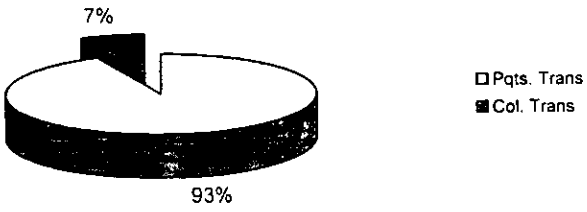




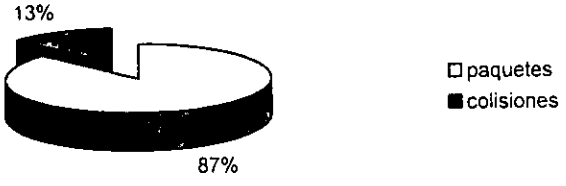


Concentrador 12 Tarjeta 5, primera semana de Febrero 1999

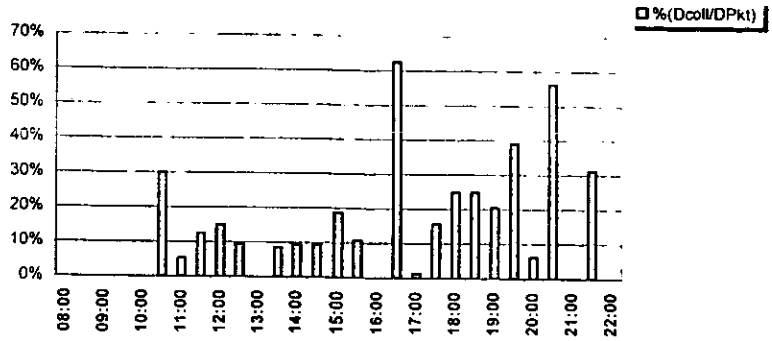
Porcentaje de paquetes y colisiones transmitidos

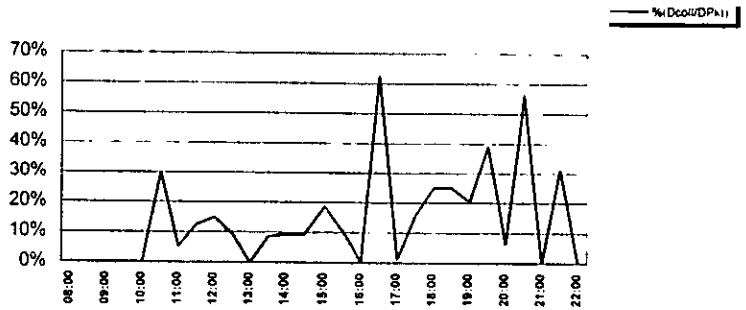


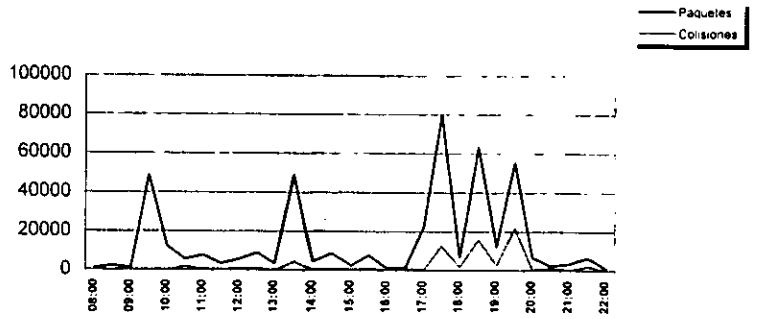
% de pqts.y col. cuando se presenta un máximo de col.



Concentrador 12 Tarjeta 5, segunda semana de Febrero 1999

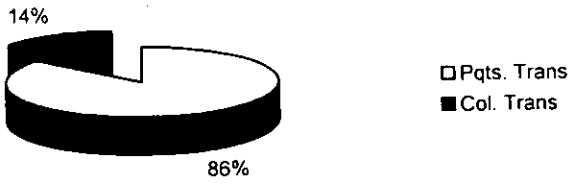






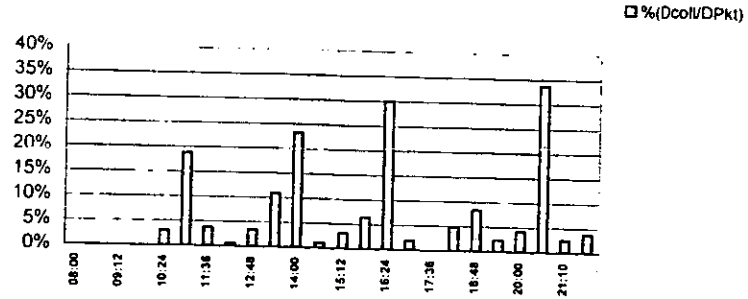
Concentrador 12 Tarjeta 5, segunda semana de Febrero 1999

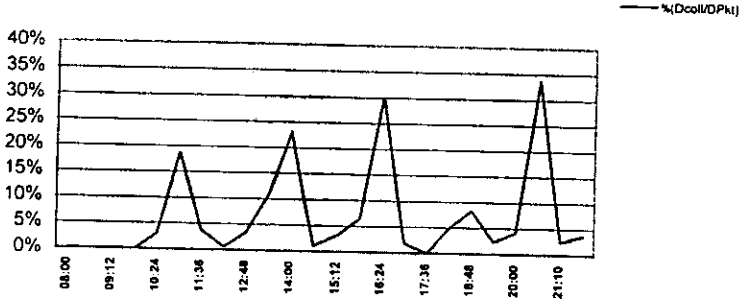
Porcentaje de paquetes y colisiones transmitidos

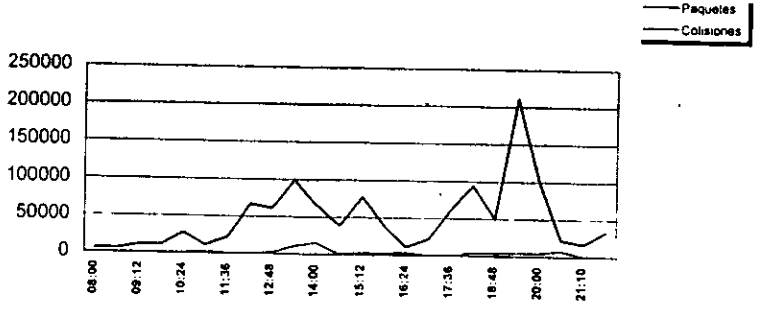


% de pqts. y col. cuando se presenta un máximo de col.

Concentrador 12, tarjeta 5, tercera semana de Febrero 1999

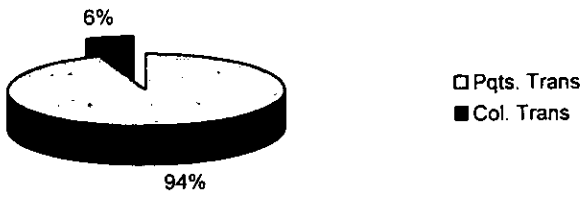




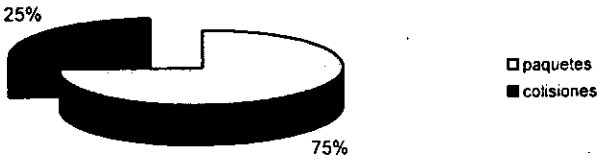


Concentrador 12,tarjeta 5, tercera semana de Febrero 1999

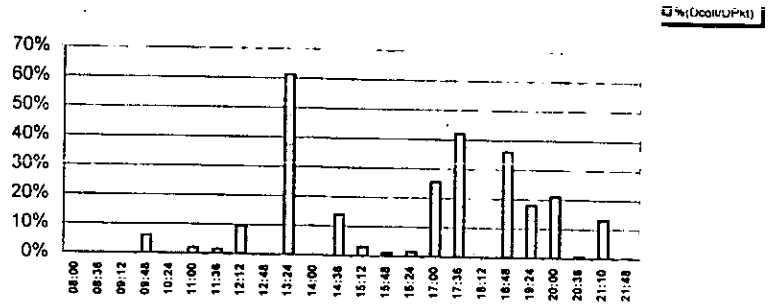
Porcentaje de paquetes y colisiones transmitidos

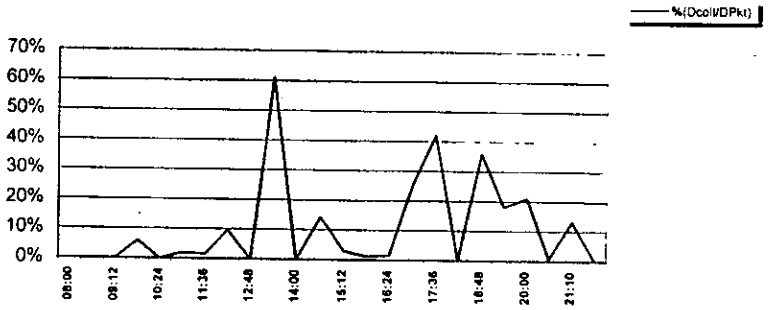


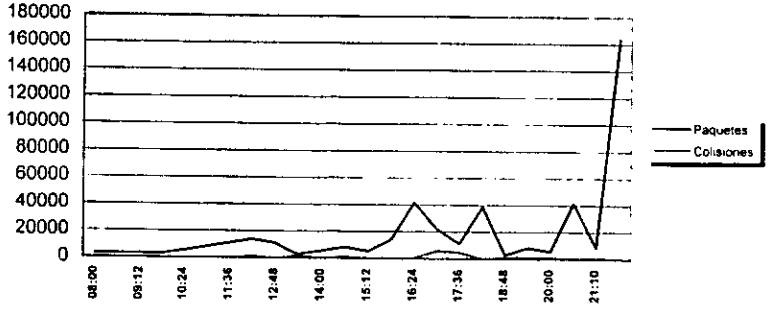
% de pqts. y col. cuando se presenta un máximo de col.



Concentrador 12 Tarjeta 5, cuarta semana de Febrero 1999

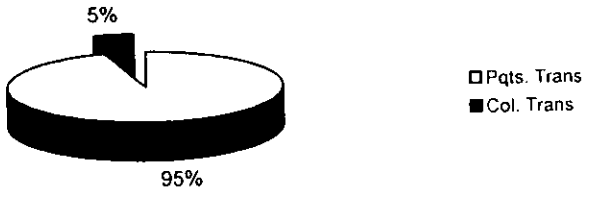




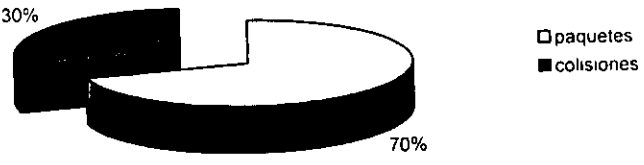


Concentrador 12 Tarjeta 5, cuarta semana de Febrero 1999

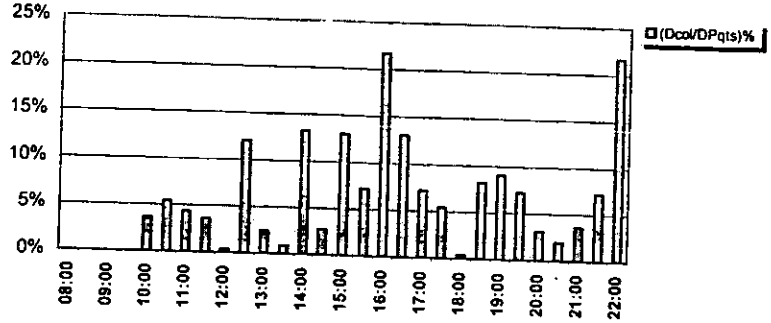
Porcentaje de paquetes y colisiones transmitidos

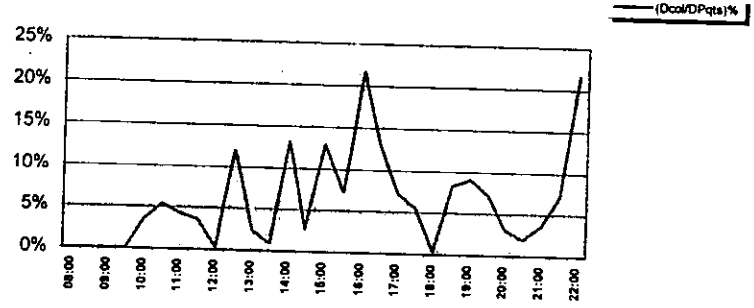


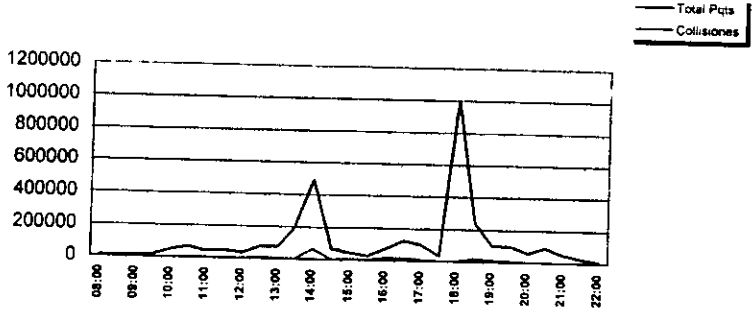
% de pqts. y col. cuando se presenta un máximo de col.



Concentrador 12 IRBM, primera semana Marzo 1999

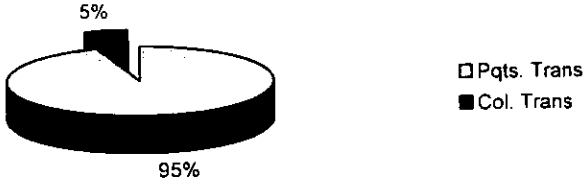




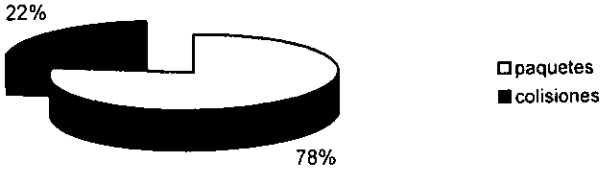


Concentrador 12 IRBM, primera semana Marzo 1999

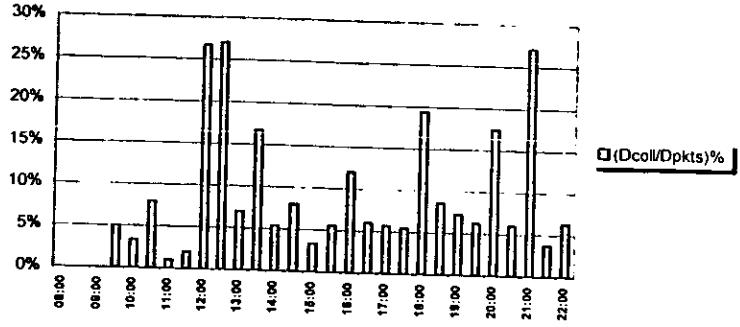
Porcentaje de paquetes y colisiones transmitidos

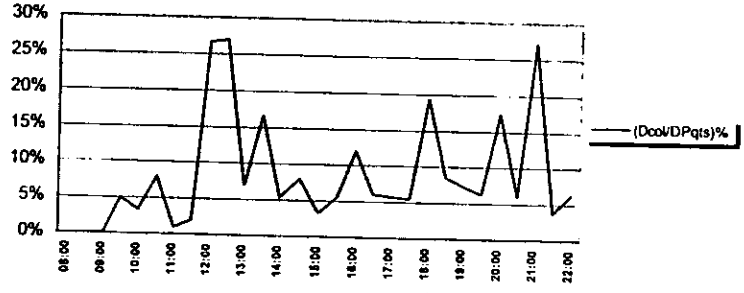


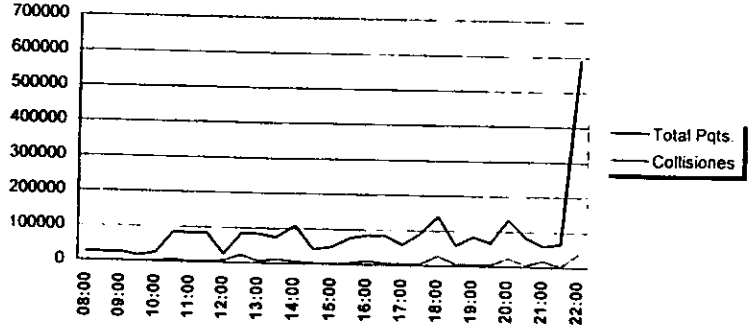
% de pqts. y col. cuando se presenta un máximo de col.



Concentrador 12 IRBM, segunda semana de Marzo de 1999

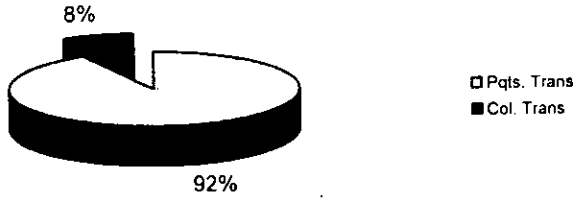




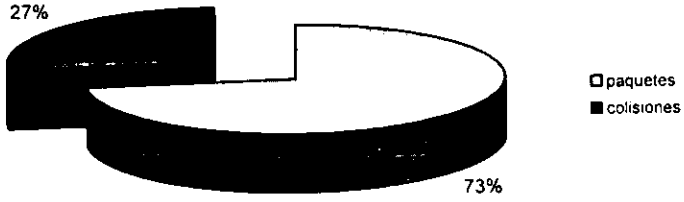


Concentrador 12 IRBM, segunda semana de Marzo de 1999

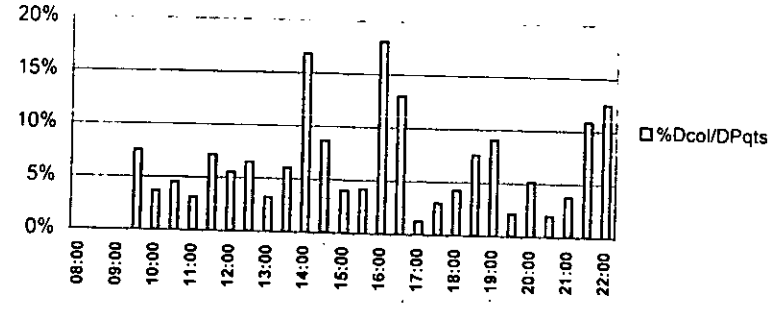
Porcentaje de paquetes y colisiones Transmitidos

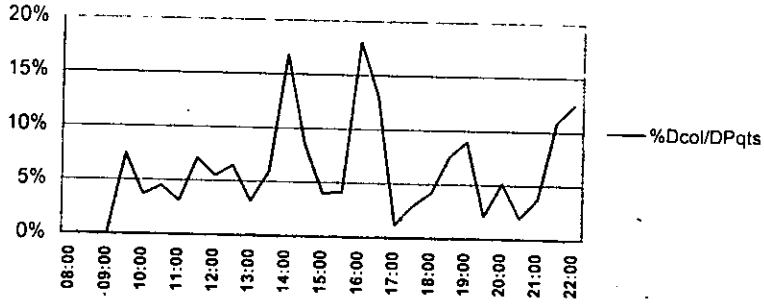


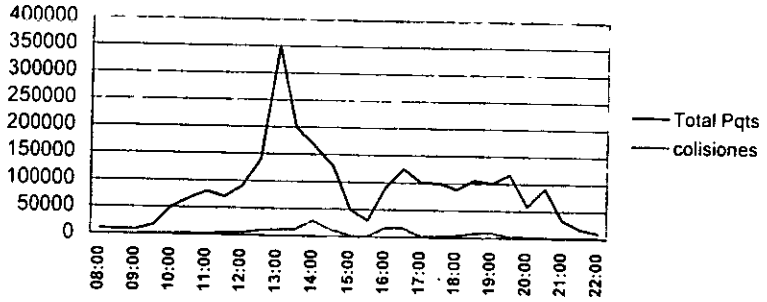
% de pqts. y col. transmitidos cuando se presenta el máximo de col.



Concentrador 12 IRBM, tercera semana de Marzo de 1999

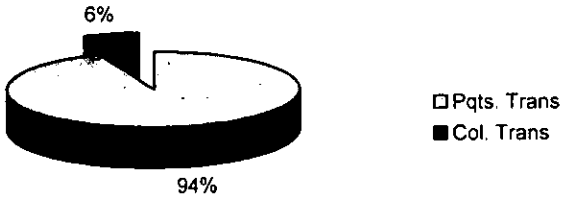




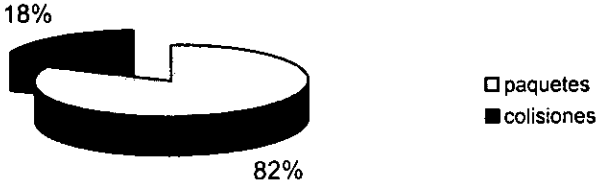


Concentrador 12 IRBM, tercera semana de Marzo de 1999

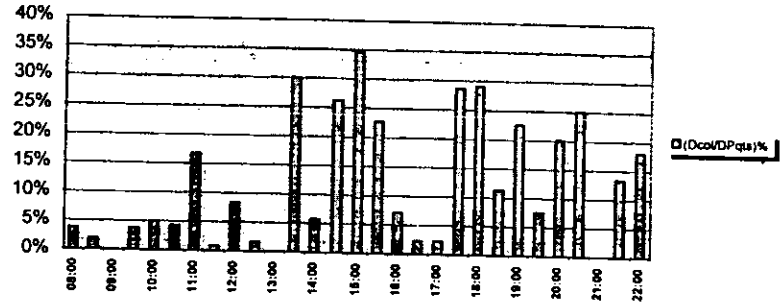
Porcentaje de paquetes y colisiones transmitidos

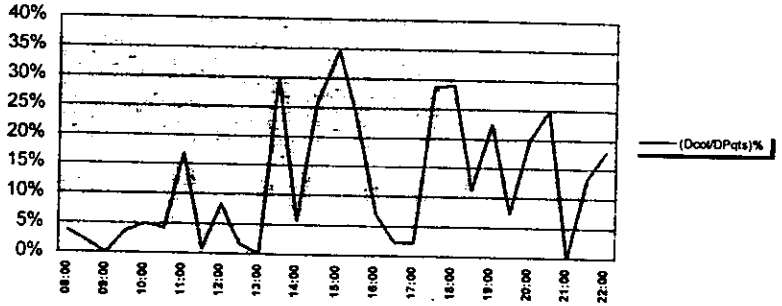


% de pqts. y col. cuando se presenta un máximo de col.



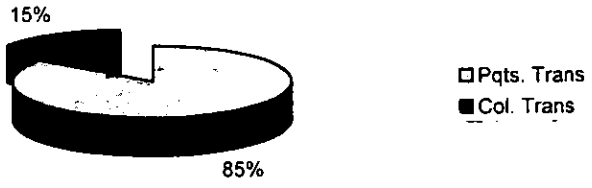
Concentrador 12 IRBM, cuarta semana de Marzo 1999



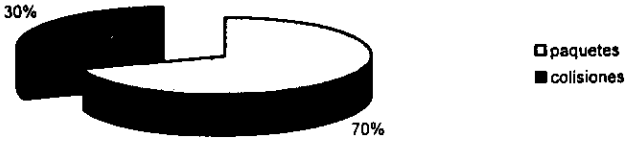


Concentrador 12 IRBM, cuarta semana de Marzo 1999

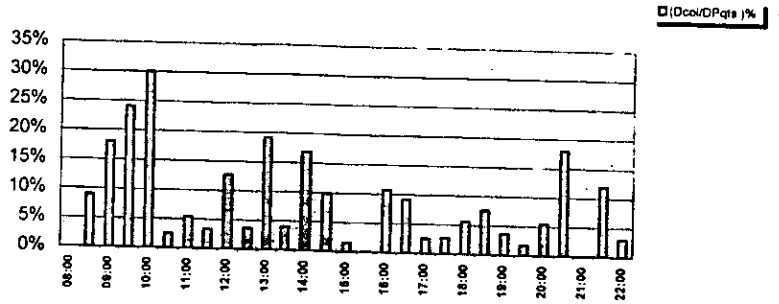
Porcentaje paquetes y colisiones transmitidos

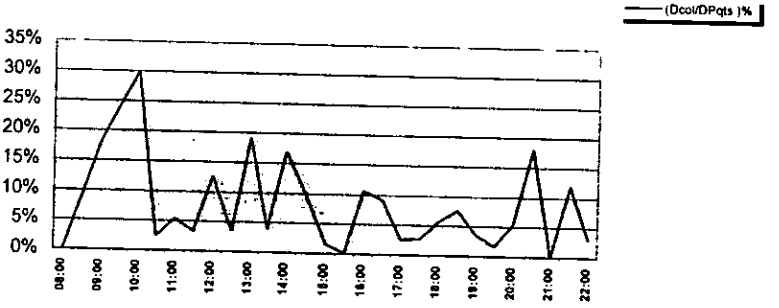


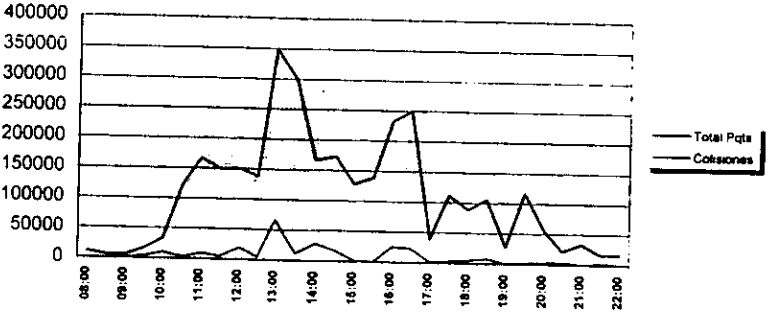
% de pqts. y col. cuando se presenta un máximo de col.



Concentrador 12 IRBM, quinta semana de Marzo 1999

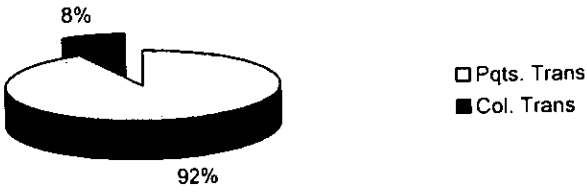




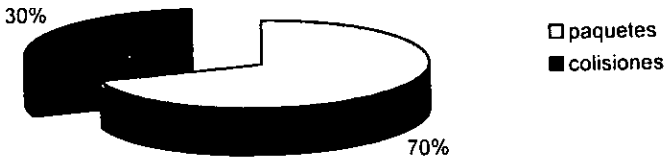


Concentrador 12 IRBM, quinta semana de Marzo 1999

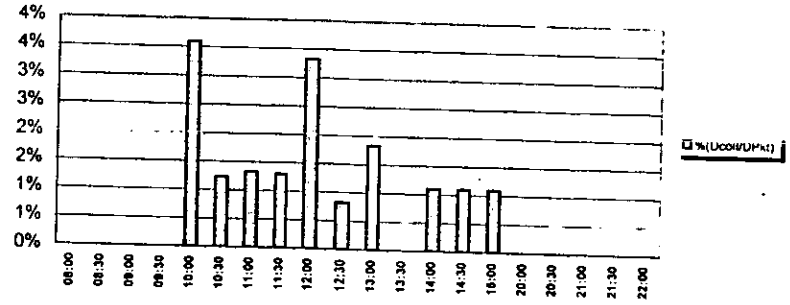
Porcentaje de paquetes y colisiones transmitidos

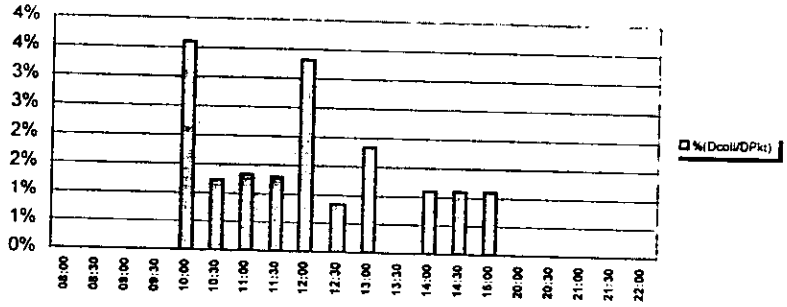


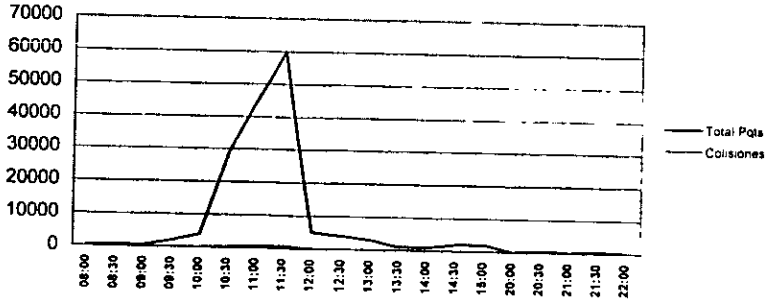
% de pqts. y col. cuando se presenta un máximo de col.



Concentrador 2, Tarjeta 2 Primera semana de febrero 1999



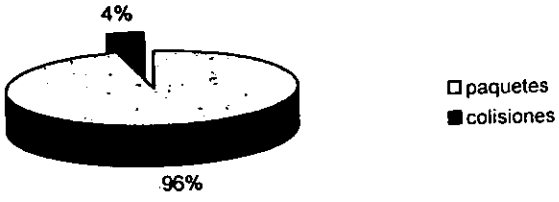




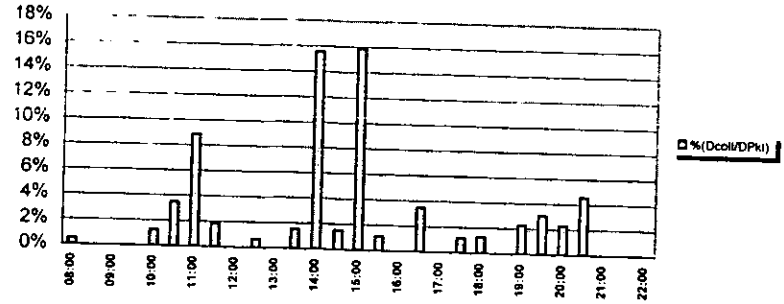
Concentrador 2, Tarjeta 2 Primera semana de febrero 1999

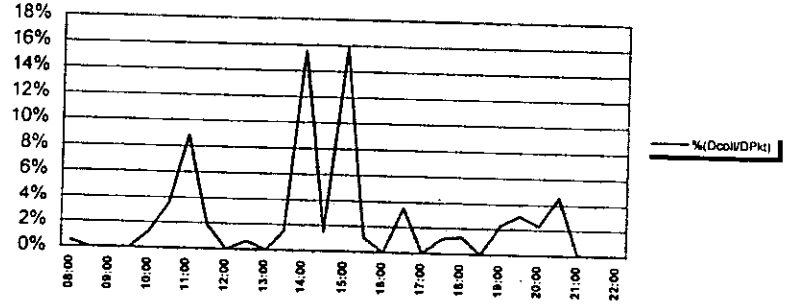
Porcentaje de paquetes y colisiones transmitidos

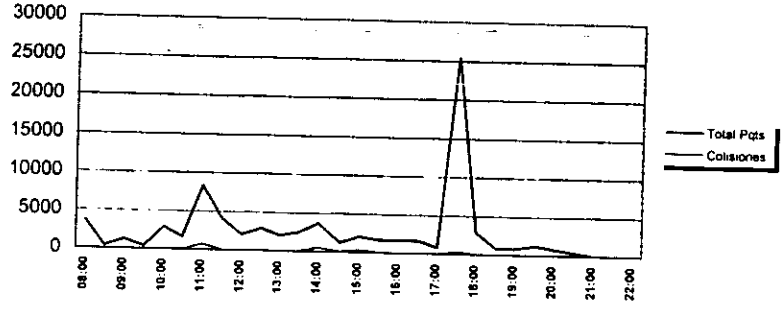
% de pqts. y col. cuando se presenta un máximo de col.



Concentrador 2 Tarjeta 2, segunda semana de Febrero 1999



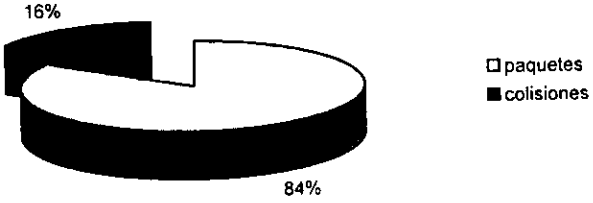




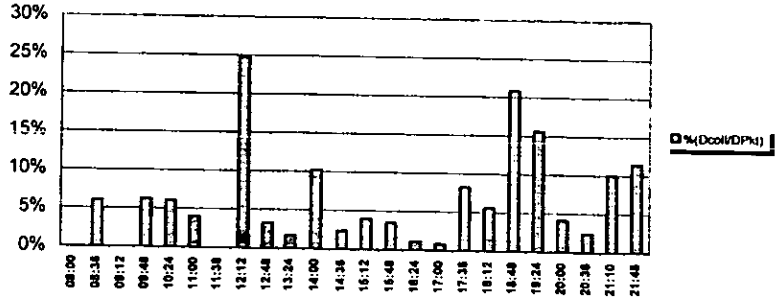
Concentrador 2 tarjeta 2, segunda seman de Febrero 1999

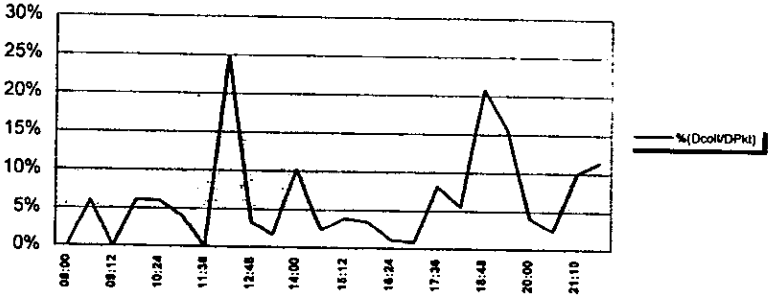
Porcentaje de paquetes y colisiones transmitidos

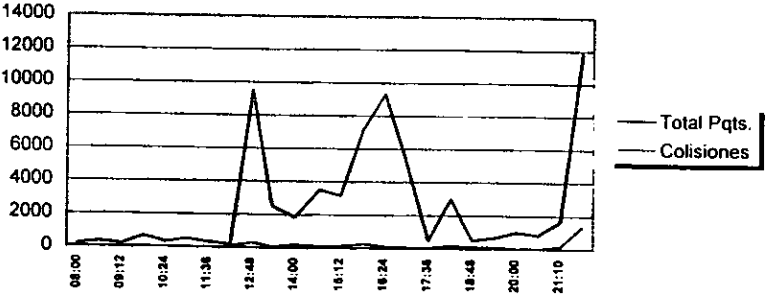
% de pqts. y col. cuando se presenta un máximo de col.



Concentrador 2 tarjeta 2, tercera semana de Febrero 1999



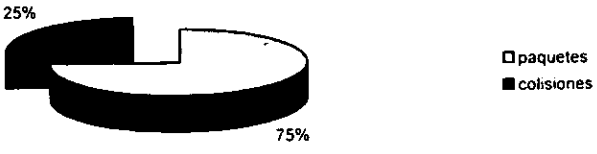




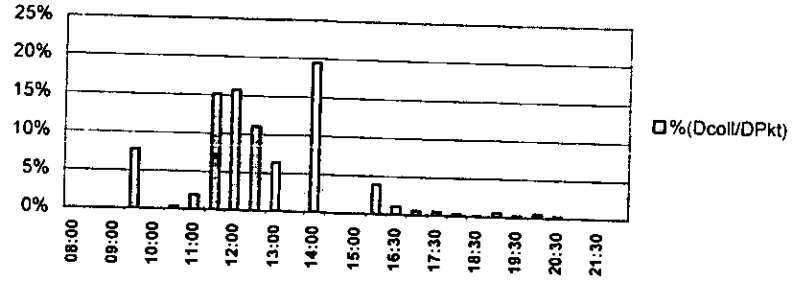
Concentrador 2 tarjeta 2, tercera semana de Febrero 1999

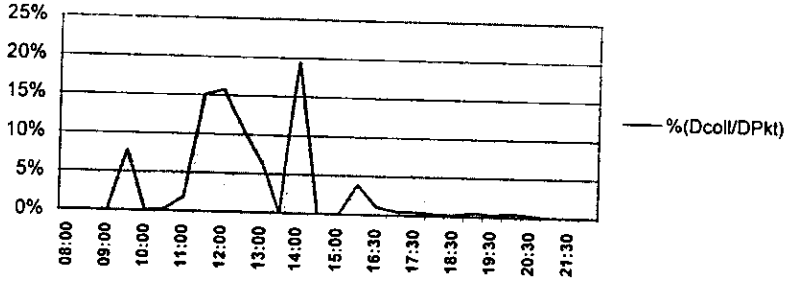
Porcentaje de paquetes y colisiones transmitidos

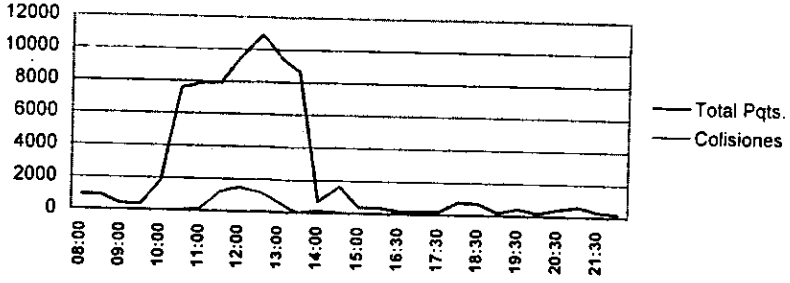
% de pqts.y col. cuando se presenta un máximo de col.



Concentrador 2 tarjeta 2, cuarta semana de Febrero 1999



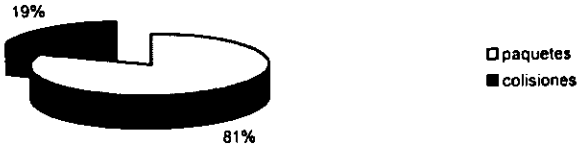




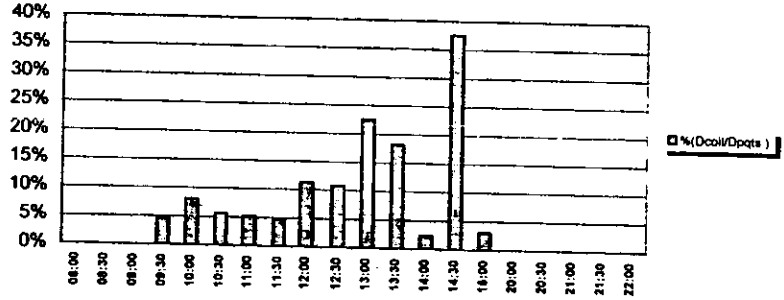
Concentrador 2 tarjeta 2, cuarta semana de Febrero 1999

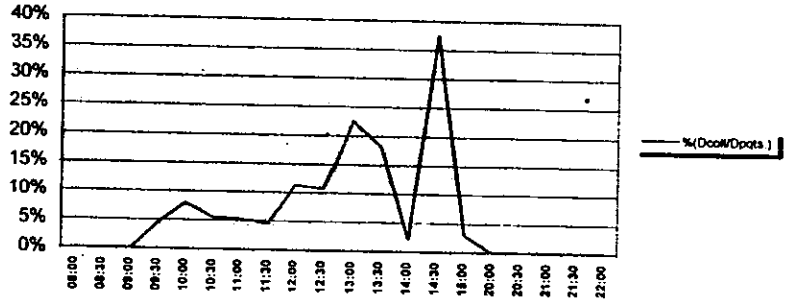
Porcentaje de colisones y paquetes transmitidos

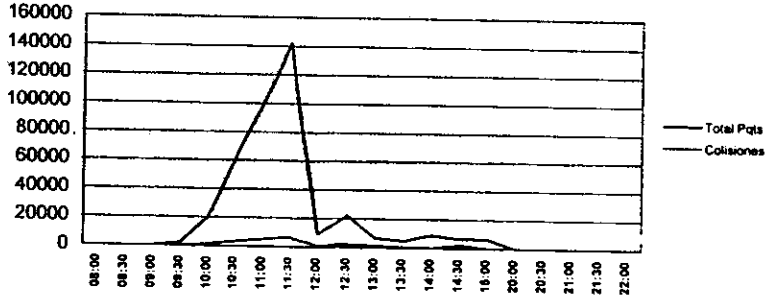
% de pqts. y col. cuando se presenta un máximo de col.



Concentrador 2 tarjeta 4, primera semana de Febrero 1999

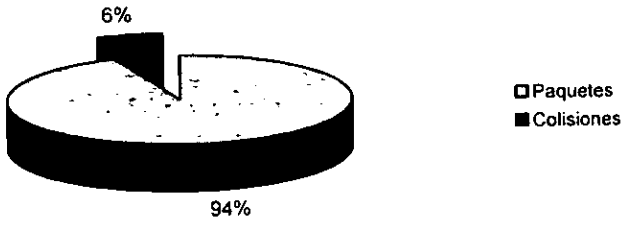






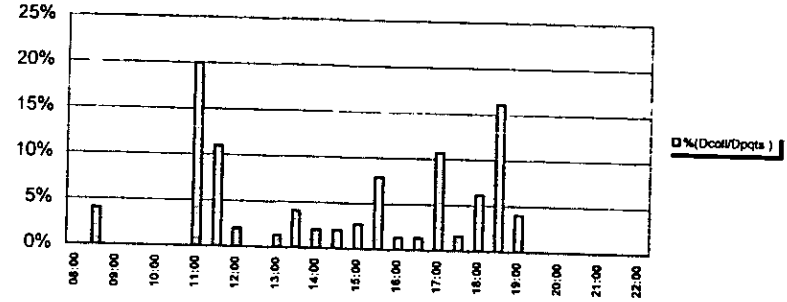
Concentrador 2 tarjeta 4, primera semana de Febrero 1999

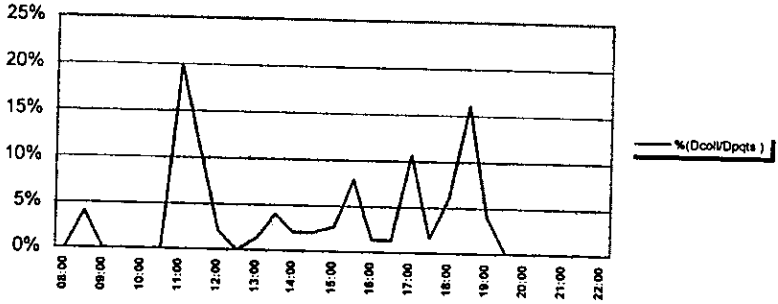
Porcentaje de paquetes y colisiones transmitidos

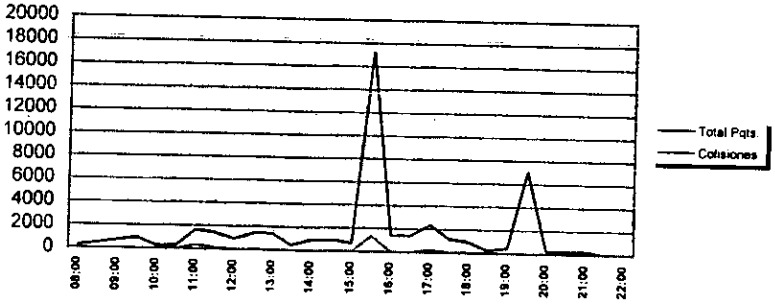


% de pqts. y col. cuando se presenta un máximo de col.

Concentrador 2 Tarjeta 4, segunda semana de Febrero 1999

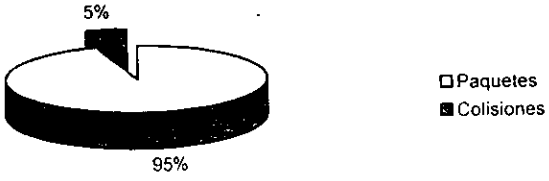




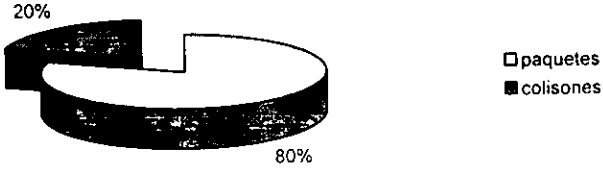


Concentrador 2 Tarjeta 4, segunda semana de Febrero 1999

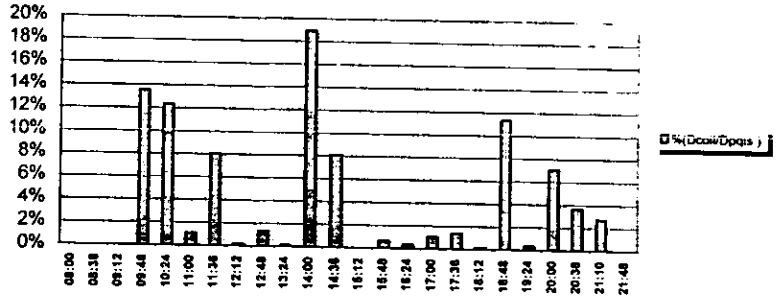
Porcentaje de paquetes y colisiones transmitidos

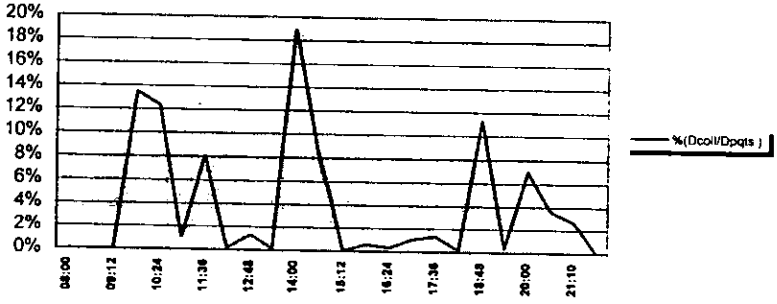


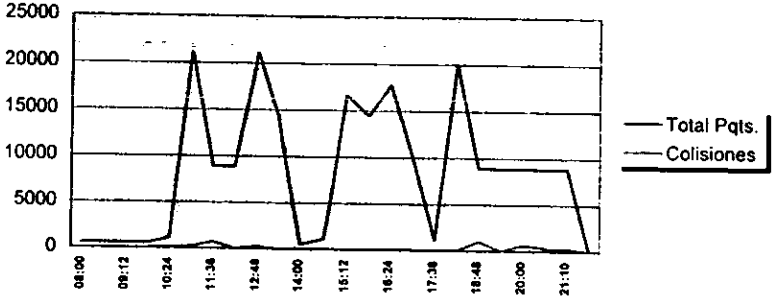
% de pqts.y col. cuando se presenta un máximo de col.



Concentrador 2 Tarjeta 4, Tercera semana de Febrero 1999

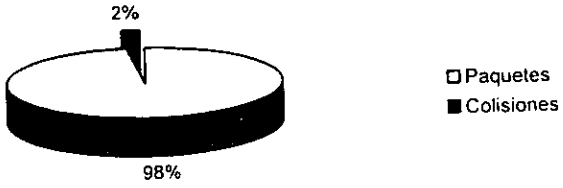




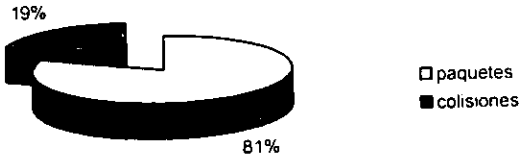


Concentrador 2 Tarjeta 4, Tercera semana de Febrero 1999

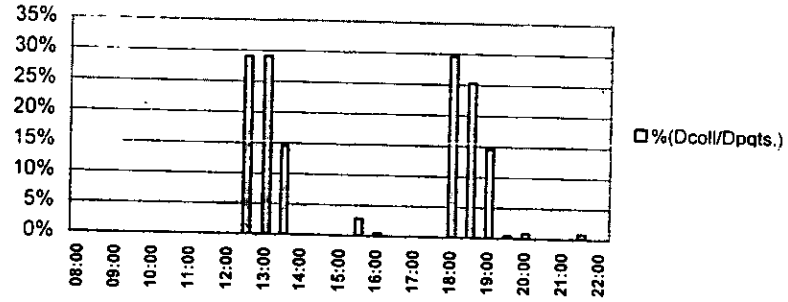
porcentaje de paquetes y colisiones transmitidos

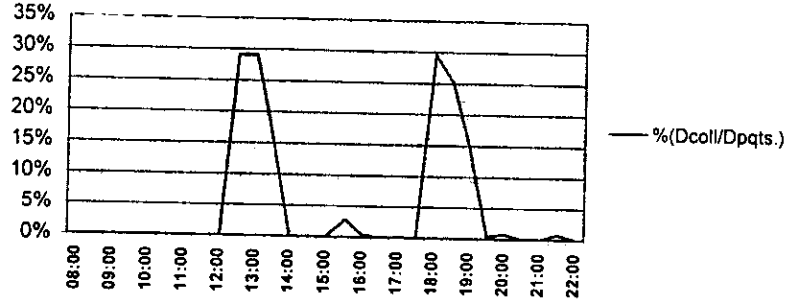


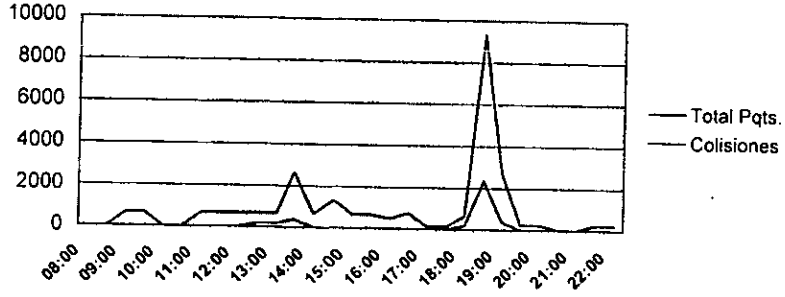
% de pqts.y col. cuando se presenta un máximo de col.



Concentrador 2 Tarjeta 4, Cuarta semana de Febrero 1999

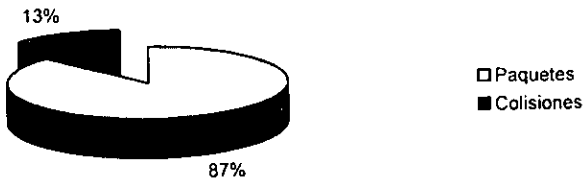




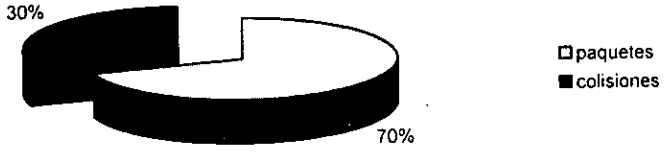


Concentrador 2 Tarjeta 4, Cuarta semana de Febrero 1999

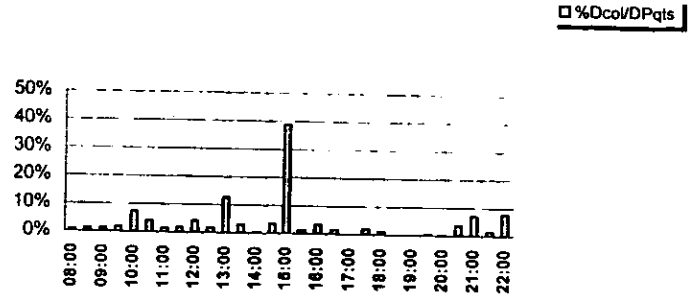
Porcentaje de paquetes y colisiones transmitidos

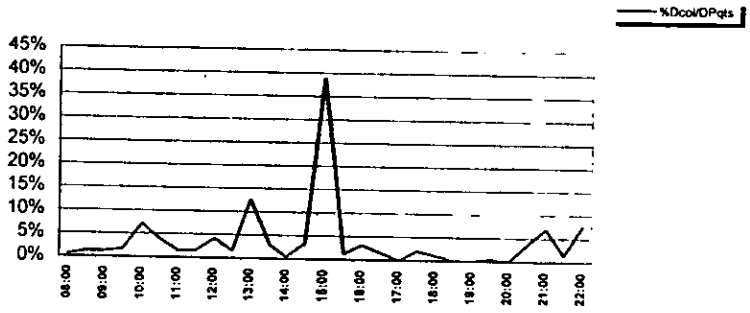


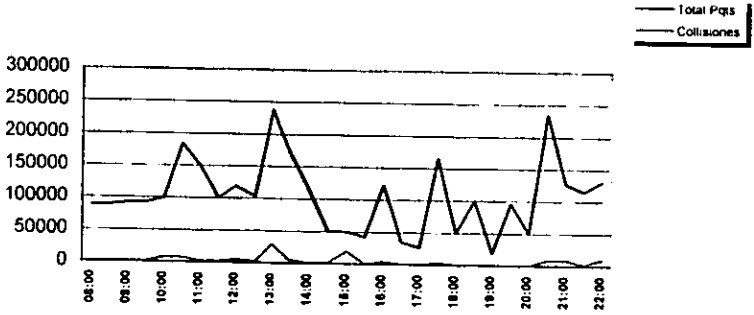
% de pqts.y col. cuando se presenta un máximo de colisiones



Concentrador 2 IRBM, primera semana de Marzo 1999.





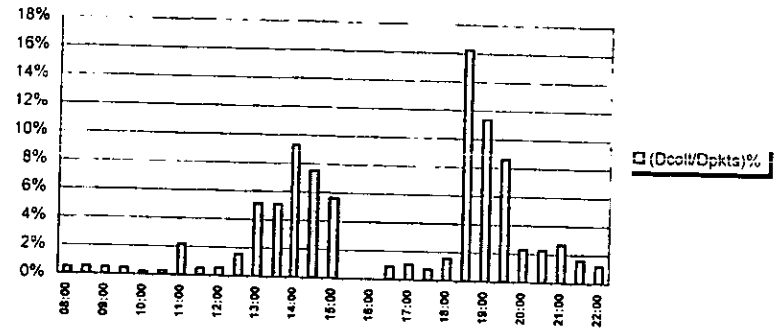


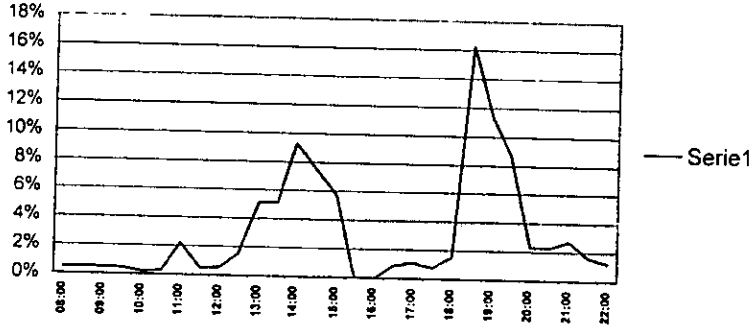
Concentrador 2 IRBM, primera semana de Marzo 1999.

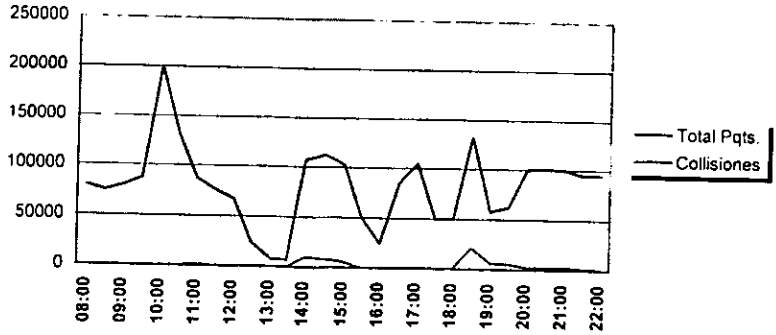
Total de paquetes y colisiones transmitidos

% de col. y pqts. cuando se presenta un máximo de col.

Concentrador 2 IRBM, segunda semana de Marzo 1999

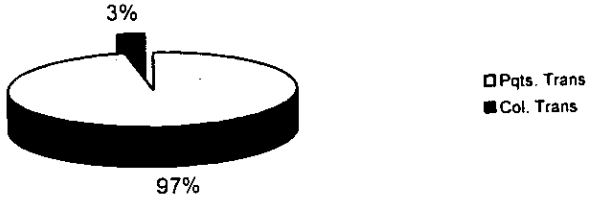




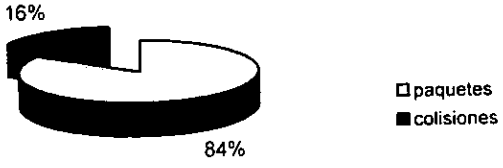


Concentrador 2 IRBM, segunda semana de Marzo 1999

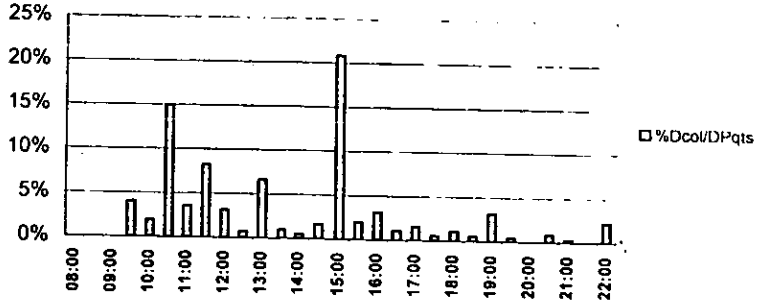
Total Paquetes y colisiones Transmitidos

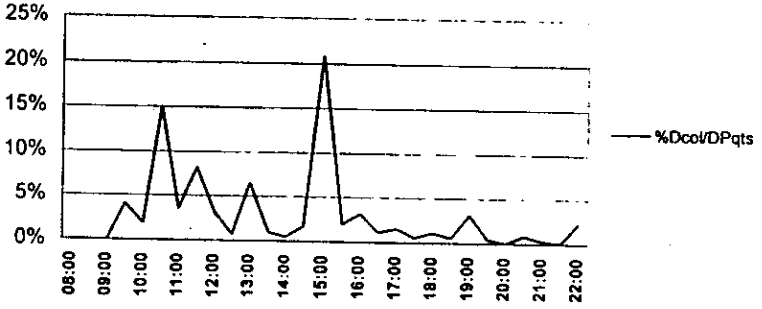


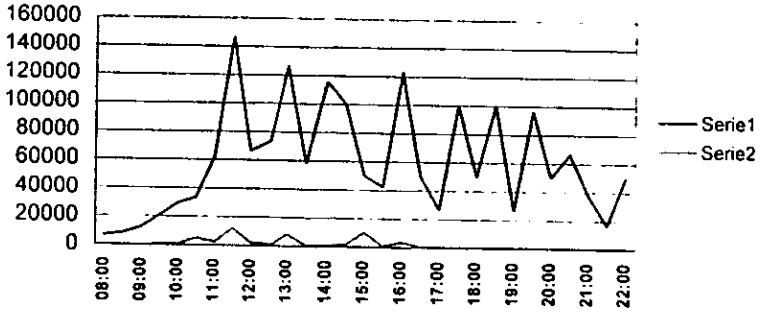
%de col. y pqts. cuando se presenta un máximo de col.



Concentrador 2 IRBM, tercera semana de Marzo 1999



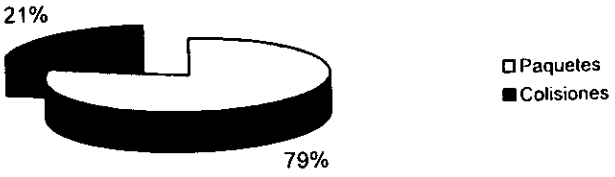




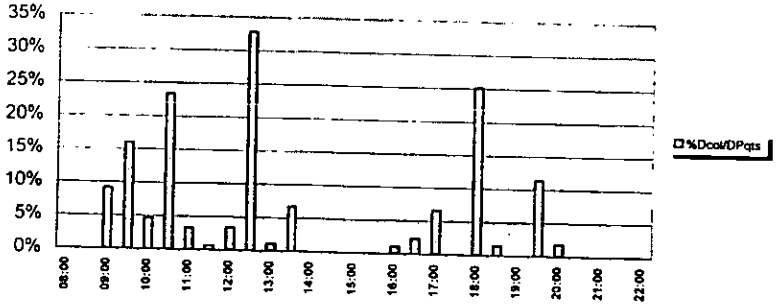
Concentrador 2 IRBM, tercera semana de Marzo 1999

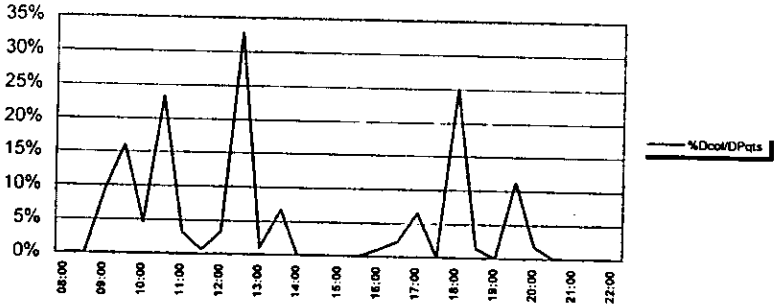
Total de paquetes y colisiones transmitidos

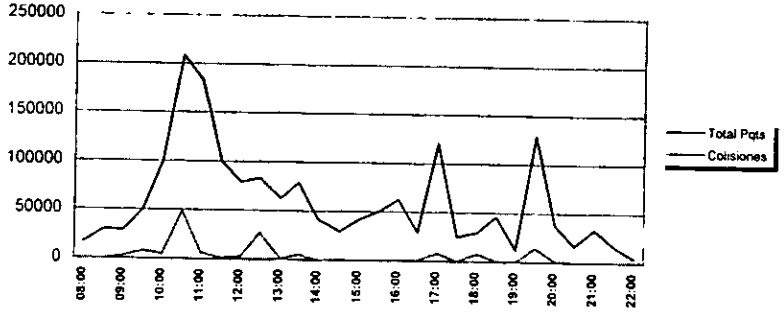
% de col. y pqts. cuando se presenta un máximo de col.



Concentrador 2 IRBM, cuarta semana de Marzo 1999

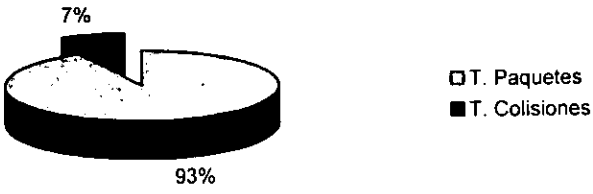




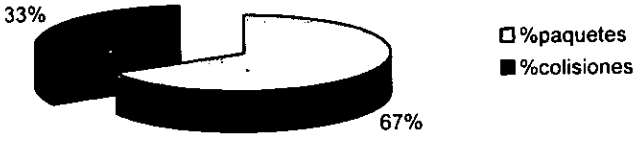


Concentrador 2 IRBM, cuarta semana de Marzo 1999

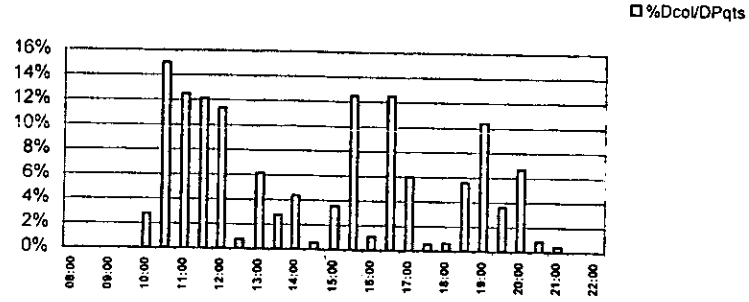
Total de paquetes y colisiones transmitidos

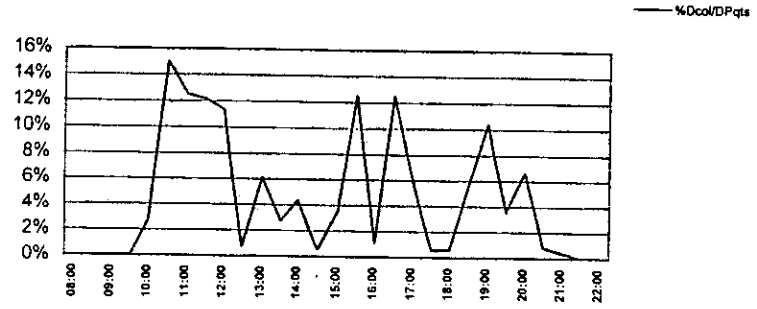


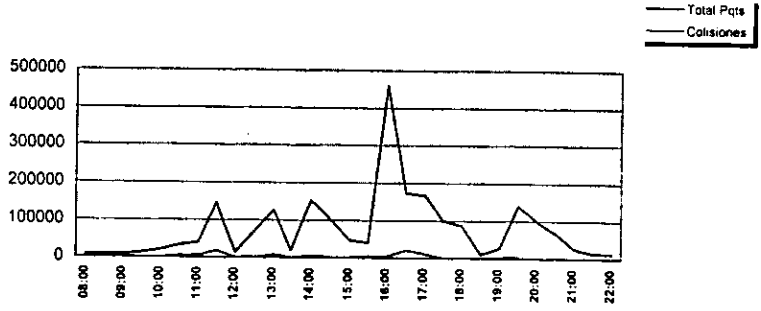
% de col. y pqts. cuando se presenta un máximo de col.



Concentrador 2 IRBM, quinta semana de Marzo 1999.

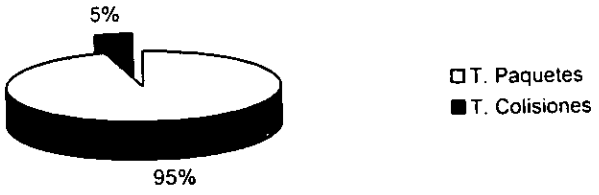




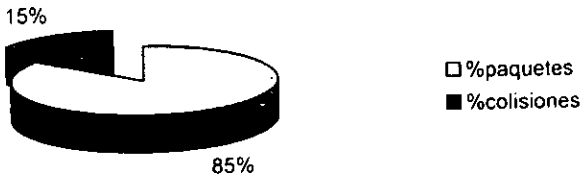


Concentrador 2 IRBM, quinta semana de Marzo 1999.

Total paquetes y colisiones transmitidos

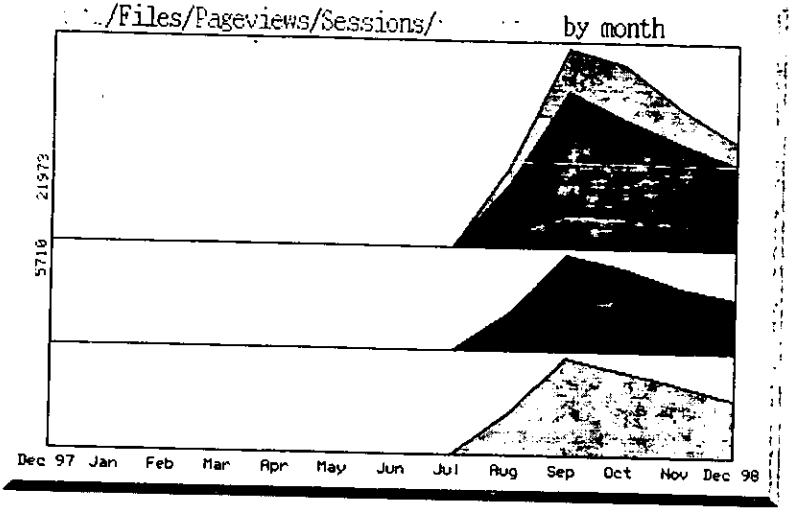


% de col. y pqts. cuando se presenta un máximo de col.



Apéndice B

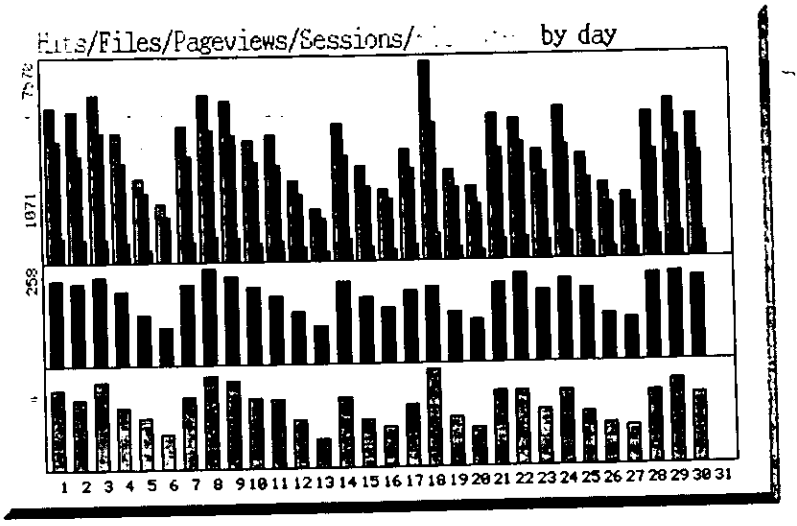
WWW Access Statistics for 1998



Month	Hits	Files	Pageviews	Sessions	KBytes sent
December 1998	74130	58812	13683	3134	443289
November 1998	96340	72611	17702	3750	562995
October 1998	126830	89206	21461	4890	669684
September 1998	137767	107251	21973	5710	771853
August 1998	56197	44273	8966	2187	318921
July 1998	0	0	0	0	0
June 1998	0	0	0	0	0
May 1998	0	0	0	0	0
April 1998	0	0	0	0	0
March 1998	0	0	0	0	0
February 1998	0	0	0	0	0
January 1998	0	0	0	0	0
Total	491264	372153	83785	19671	2766740
Average	40938	31012	6982	1639	230561

Back to the Main Page
Evaluation version - please register your copy

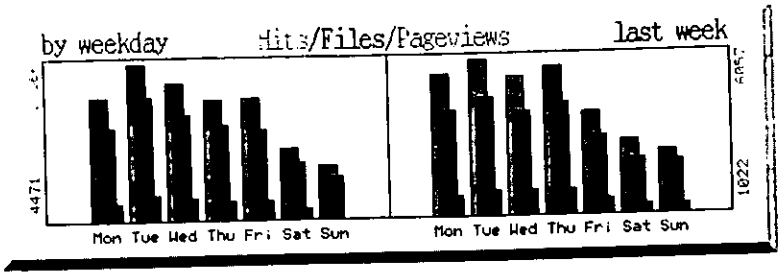
Hits by day



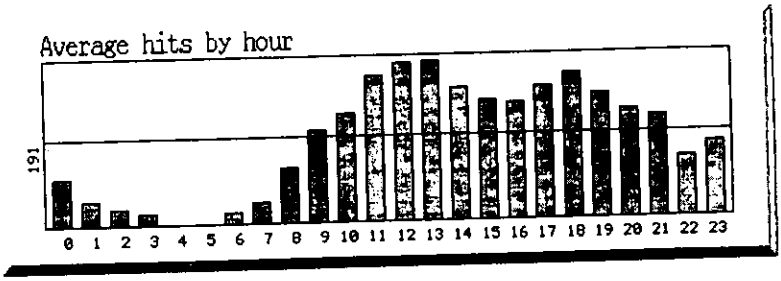
Day	Hits		Files		Pageviews		Sessions		KBytes sent	
1	5900	4.28%	4678	4.36%	1002	4.56%	232	4.06%	32312	4.19%
2	5785	4.20%	4099	3.82%	939	4.27%	222	3.89%	28739	3.72%
3	6387	4.64%	4957	4.62%	955	4.35%	238	4.17%	35254	4.57%
4	4964	3.60%	3822	3.56%	784	3.57%	199	3.49%	25236	3.27%
5	3189	2.31%	2674	2.49%	490	2.23%	138	2.42%	21027	2.72%
6	2234	1.62%	1770	1.65%	324	1.47%	104	1.82%	14352	1.86%
7	5190	3.77%	4060	3.79%	793	3.61%	219	3.84%	29178	3.78%
8	6412	4.65%	4990	4.65%	971	4.42%	258	4.52%	37593	4.87%
9	6148	4.46%	4850	4.52%	922	4.20%	239	4.19%	35382	4.58%
10	4647	3.37%	3798	3.54%	779	3.55%	210	3.68%	28255	3.66%
11	4842	3.51%	3716	3.46%	825	3.75%	185	3.24%	27832	3.61%
12	3082	2.24%	2573	2.40%	563	2.56%	141	2.47%	19812	2.57%
13	2032	1.47%	1641	1.53%	361	1.64%	104	1.82%	11768	1.52%
14	5286	3.84%	4080	3.80%	848	3.86%	223	3.91%	28301	3.67%
15	3629	2.63%	2834	2.64%	588	2.68%	181	3.17%	19782	2.56%
16	2726	1.98%	2361	2.20%	445	2.03%	151	2.64%	16526	2.14%
17	4240	3.08%	3491	3.25%	627	2.85%	195	3.42%	25406	3.29%
18	7570	5.49%	5269	4.91%	1071	4.87%	204	3.57%	39089	5.06%
19	3473	2.52%	2777	2.59%	529	2.41%	137	2.40%	20285	2.63%
20	2764	2.01%	2160	2.01%	369	1.68%	117	2.05%	15397	1.99%
21	5569	4.04%	4254	3.97%	815	3.71%	212	3.71%	30500	3.95%
22	5368	3.90%	4498	4.19%	896	4.08%	240	4.20%	30422	3.94%
23	4173	3.03%	3336	3.11%	734	3.34%	193	3.38%	22526	2.92%
24	5787	4.20%	4356	4.06%	1022	4.65%	222	3.89%	30581	3.96%
25	3980	2.89%	3110	2.90%	658	2.99%	197	3.45%	21724	2.81%
26	2910	2.11%	2160	2.01%	426	1.94%	131	2.29%	17039	2.21%

Day	Hits		Files		Pageviews		Sessions		KBytes sent	
27	2469	1.79%	2126	1.98%	383	1.74%	116	2.03%	15447	2.00%
28	5536	4.02%	4111	3.83%	855	3.89%	234	4.10%	29478	3.82%
29	6057	4.40%	4641	4.33%	1014	4.61%	240	4.20%	34293	4.44%
30	5418	3.93%	4059	3.78%	985	4.48%	228	3.99%	28326	3.67%
Total	137767	100.00%	107251	100.00%	21973	100.00%	5710	100.00%	771853	100.00%

Average load

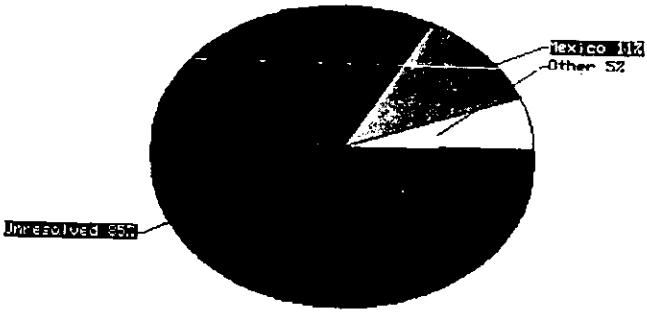


The Top 7 days of the period						
No.	Hits		304's		KBytes sent	Date
1	7570	5.49%	1740	10.86%	39089	18/Sep/1998
2	6412	4.65%	801	5.00%	37593	08/Sep/1998
3	6387	4.64%	695	4.34%	35254	03/Sep/1998
4	6148	4.46%	537	3.35%	35382	09/Sep/1998
5	6057	4.40%	684	4.27%	34293	29/Sep/1998
6	5900	4.28%	575	3.59%	32312	01/Sep/1998
7	5787	4.20%	946	5.90%	30581	24/Sep/1998



The Top 24 hours of the period						
No.	Hits		304's		KBytes sent	Date/Time
1	995	0.72%	635	3.96%	2395	28/Sep/199812:XX:XX
2	900	0.65%	290	1.81%	4758	18/Sep/199821:XX:XX
3	825	0.60%	51	0.32%	4419	09/Sep/199813:XX:XX
4	810	0.59%	137	0.85%	4319	18/Sep/199814:XX:XX
5	792	0.57%	240	1.50%	3765	11/Sep/199813:XX:XX
6	754	0.55%	114	0.71%	3326	02/Sep/199813:XX:XX
7	699	0.51%	261	1.63%	3722	18/Sep/199820:XX:XX
8	693	0.50%	108	0.67%	3493	30/Sep/199813:XX:XX
9	671	0.49%	98	0.61%	4349	08/Sep/199819:XX:XX
10	662	0.48%	61	0.38%	3694	14/Sep/199813:XX:XX
11	654	0.47%	80	0.50%	4027	21/Sep/199818:XX:XX

Hits by Country



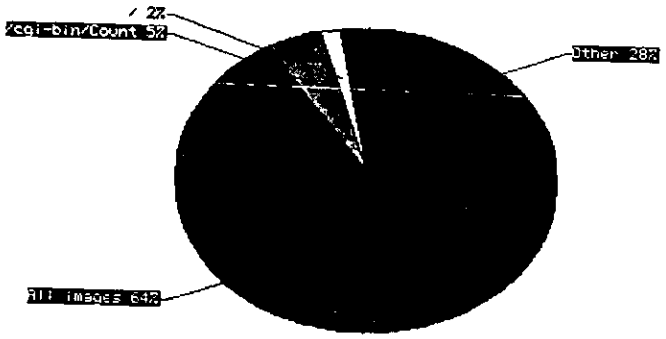
Hits by Country

100% = 137767 hits

No.	Hits		304's		KBytes sent	Country
1	116712	84.72%	14804	92.39%	645551	Unresolved
2	14566	10.57%	880	5.49%	88917	Mexico
3	1825	1.32%	71	0.44%	10531	US Commercial
4	1463	1.06%	102	0.64%	8112	Network
5	521	0.38%	13	0.08%	2690	US Educational
6	460	0.33%	20	0.12%	2635	France
7	385	0.28%	14	0.09%	2231	Spain
8	248	0.18%	6	0.04%	1558	Chile
9	225	0.16%	1	0.01%	1415	Canada
10	158	0.11%	2	0.01%	904	Netherlands
11	148	0.11%	26	0.16%	896	United Kingdom
12	144	0.10%	56	0.35%	986	Bolivia
13	139	0.10%	0	0.00%	923	Argentina
14	100	0.07%	1	0.01%	419	Panama
15	97	0.07%	30	0.19%	376	Australia
16	89	0.06%	1	0.01%	568	Germany
17	75	0.05%	0	0.00%	449	Switzerland
18	72	0.05%	2	0.01%	325	Colombia
19	45	0.03%	0	0.00%	421	Japan
20	43	0.03%	0	0.00%	271	Austria
21	43	0.03%	0	0.00%	156	Guatemala
22	38	0.03%	1	0.01%	216	Uruguay
23	37	0.03%	0	0.00%	274	Ecuador
24	34	0.02%	1	0.01%	263	Peru
25	28	0.02%	0	0.00%	40	Brazil
26	22	0.02%	0	0.00%	155	El Salvador

No.	Hits		304's		KBytes sent	Country
27	15	0.01%	0	0.00%	304	Non-Profit Organization
28	11	0.01%	0	0.00%	59	US Military
29	9	0.01%	0	0.00%	81	Italy
30	5	0.00%	0	0.00%	48	New Zealand (Aotearoa)
31	3	0.00%	0	0.00%	6	Israel
32	3	0.00%	0	0.00%	37	Finland
33	2	0.00%	0	0.00%	9	Cuba
34	2	0.00%	0	0.00%	32	Ireland

The Top 30 items/URLs



The Top 30 Items/URLs
100% = 137767 hits

More details

No.	Hits		304's		KBytes sent	URL
1	88535	64.26%	12732	79.46%	626705	All images
2	7363	5.34%	0	0.00%	2847	/cgi-bin/Count.cgi
3	2981	2.16%	975	6.08%	4565	/
4	1289	0.94%	186	1.16%	2890	/homeesp.html
5	680	0.49%	86	0.54%	1955	/infogrl/infogrl.html
6	612	0.44%	576	3.59%	10	/torre-ii/camara.html
7	598	0.43%	36	0.22%	1933	/areas/areas.html
8	545	0.40%	27	0.17%	1649	/proyectos/proyectos.html
9	495	0.36%	29	0.18%	1152	/informes/info94/info1994.html
10	479	0.35%	35	0.22%	1579	/consulta/consulta.html
11	405	0.29%	32	0.20%	1544	/share/share.html
12	340	0.25%	57	0.36%	2155	/directorio/directorio.html
13	331	0.24%	18	0.11%	837	/estudios/estudios.html
14	299	0.22%	18	0.11%	831	/ccco/ccco.html
15	277	0.20%	9	0.06%	847	/informes/info95/map-idx.html
16	275	0.20%	52	0.32%	915	/torre-ii/menu3.htm
17	275	0.20%	48	0.30%	268	/torre-ii/index.htm
18	268	0.19%	44	0.27%	138	/torre-ii/inicio2.htm
19	227	0.16%	18	0.11%	2385	/torre-ii/_fpclass/fphover.class
20	221	0.16%	7	0.04%	729	/asocia/secive.html
21	215	0.16%	0	0.00%	1580	/cgi-bin/ws.exe/websql.dir/webbdaii/tesbdaii.hts
22	213	0.15%	0	0.00%	328	/cgi-bin/ws.exe/websql.dir/webbdaii/datper3.hts
23	204	0.15%	18	0.11%	371	/torre-ii/_fpclass/fphoverx.class
24	199	0.14%	93	0.58%	15	/velec/velec.css

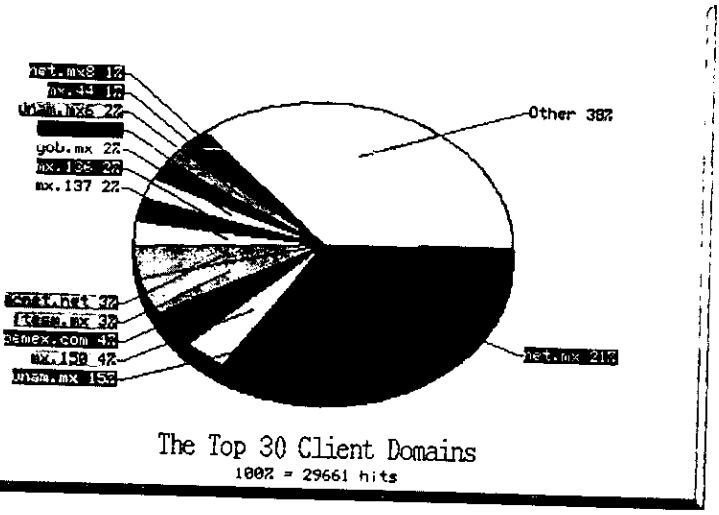
No.	Hits		304's		KBytes sent	URL
25	177	0.13%	7	0.04%	622	/cecco/cursos.html
26	175	0.13%	2	0.01%	182	/informes/info94/estructu.html
27	163	0.12%	31	0.19%	438	/asocia/asocia.html
28	157	0.11%	14	0.09%	133	/informes/info94/mecterfl.html
29	148	0.11%	2	0.01%	439	/informes/mformes.html
30	136	0.10%	7	0.04%	4162	/informes/info94/b2invest.html

The 10 least frequently accessed items/URLs

More details

No.	Hits		304's		KBytes sent	URL
10	1	0.00%	0	0.00%	3	/torre-ii/etapa-cf1.htm
9	1	0.00%	0	0.00%	1	/informes/direccion/info95/fig31.html
8	1	0.00%	0	0.00%	8	/torre-ii/empresas/ern/sismicidad.htm
7	1	0.00%	0	0.00%	4	/informes/direccion/info96/fig24.html
6	1	0.00%	0	0.00%	1	/torre-ii/corte-e1.htm
5	1	0.00%	0	0.00%	2	/cgi-bin/ws.exe/websql.dir/webbdaii/tesis.hts
4	1	0.00%	0	0.00%	18	/torre-ii/noticias/pp-prensa.htm
3	1	0.00%	0	0.00%	1	/informes/direccion/info95/fig16.html
2	1	0.00%	0	0.00%	1	/informes/direccion/info95/fig17.html
1	1	0.00%	0	0.00%	3	/informes/direccion/info96/fig15.html

The Top 30 client domains

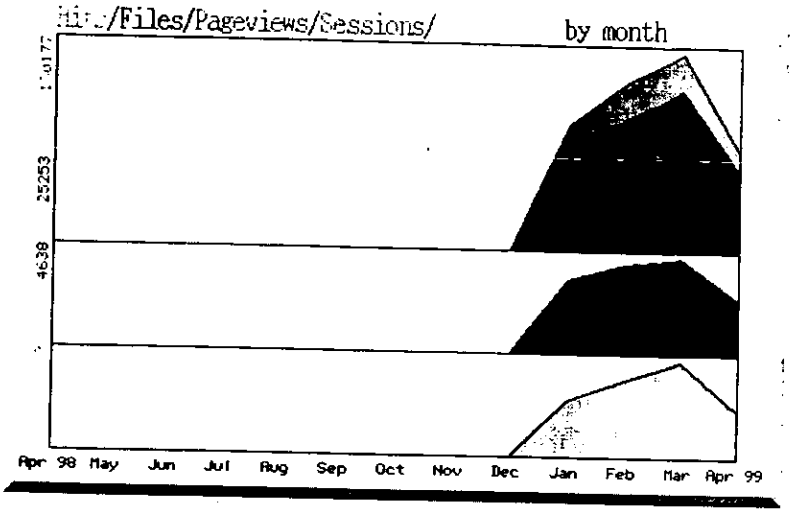


More details

No.	Hits		304's		KBytes sent	Domain
1	6223	20.98%	341	9.39%	37750	net.mx
2	4310	14.53%	304	8.37%	27011	unam.mx
3	1151	3.88%	922	25.39%	1443	mx.150
4	1150	3.88%	67	1.84%	6239	pemex.com
5	1000	3.37%	24	0.66%	6579	itesm.mx
6	806	2.72%	33	0.91%	4553	acnet.net
7	658	2.22%	254	6.99%	497	mx.137
8	571	1.93%	180	4.96%	1243	mx.136
9	570	1.92%	29	0.80%	3430	gob.mx
10	566	1.91%	165	4.54%	1190	mx.117
11	446	1.50%	25	0.69%	3481	unam.mx6
12	441	1.49%	71	1.95%	1095	mx.44
13	427	1.44%	381	10.49%	269	net.mx8
14	383	1.29%	18	0.50%	2254	udg.mx
15	291	0.98%	49	1.35%	1323	imp.mx
16	282	0.95%	22	0.61%	1324	com.mx
17	279	0.94%	30	0.83%	1414	ibm.net
18	189	0.64%	0	0.00%	915	uaem.mx
19	187	0.63%	15	0.41%	1335	net.mx93
20	181	0.61%	0	0.00%	1098	rediris.es
21	179	0.60%	0	0.00%	933	mx.168
22	170	0.57%	92	2.53%	409	mx.104
23	167	0.56%	36	0.99%	892	unam.mx0
24	155	0.52%	40	1.10%	1343	mx.161

No.	Hits		304's		KBytes sent	Domain
25	154	0.52%	4	0.11%	1180	net.mx1
26	152	0.51%	0	0.00%	1147	ipn.mx
27	148	0.50%	26	0.72%	896	ac.uk
28	147	0.50%	4	0.11%	584	uson.mx24
29	136	0.46%	2	0.06%	749	WAU.NL
30	132	0.45%	17	0.47%	1114	net.mx6

WWW Access Statistics for 1999

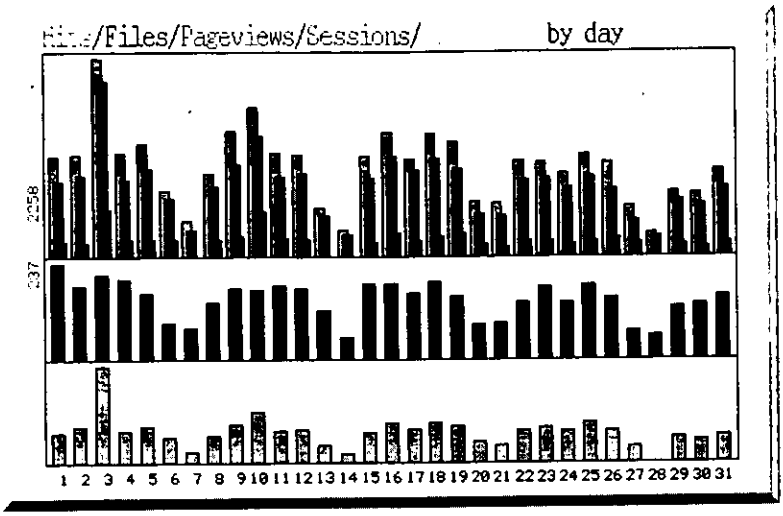


Month	Hits	Files	Pageviews	Sessions	KBytes sent
April 1999	65589	52090	10304	2621	401568
March 1999	130177	104739	25253	4638	833934
February 1999	109893	85963	19032	4314	657757
January 1999	82256	65751	14563	3533	491698
December 1998	0	0	0	0	0
November 1998	0	0	0	0	0
October 1998	0	0	0	0	0
September 1998	0	0	0	0	0
August 1998	0	0	0	0	0
July 1998	0	0	0	0	0
June 1998	0	0	0	0	0
May 1998	0	0	0	0	0
Total	387915	308543	69152	15106	2384956
Average	32326	25711	5762	1258	198746

Back to the Main Page

Evaluation version - please register your copy

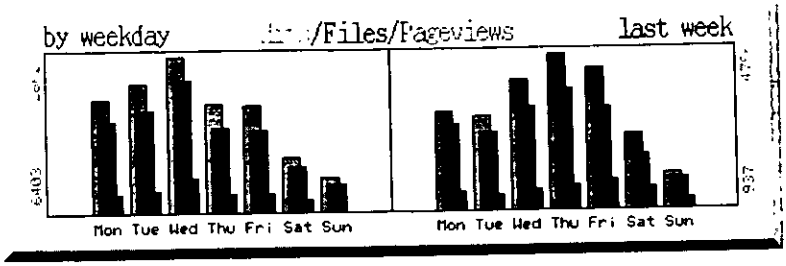
Hits by day



Day	Hits		Files		Pageviews		Sessions		KBytes sent	
1	4697	3.61%	3523	3.36%	729	2.89%	237	5.11%	26652	3.20%
2	4778	3.67%	3816	3.64%	692	2.74%	183	3.95%	31201	3.74%
3	9183	7.05%	8224	7.85%	2258	8.94%	212	4.57%	81933	9.82%
4	4831	3.71%	3575	3.41%	868	3.44%	200	4.31%	26997	3.24%
5	5316	4.08%	4120	3.93%	806	3.19%	167	3.60%	32233	3.87%
6	3119	2.40%	2741	2.62%	831	3.29%	90	1.94%	21946	2.63%
7	1699	1.31%	1300	1.24%	295	1.17%	80	1.72%	9991	1.20%
8	3899	3.00%	3309	3.16%	835	3.31%	143	3.08%	23444	2.81%
9	5848	4.49%	4344	4.15%	1005	3.98%	177	3.82%	33231	3.98%
10	6963	5.35%	5657	5.40%	2131	8.44%	171	3.69%	43290	5.19%
11	4849	3.72%	3773	3.60%	938	3.71%	186	4.01%	27993	3.36%
12	4743	3.64%	3872	3.70%	849	3.36%	178	3.84%	29023	3.48%
13	2325	1.79%	1901	1.81%	372	1.47%	121	2.61%	15112	1.81%
14	1239	0.95%	1041	0.99%	196	0.78%	58	1.25%	8139	0.98%
15	4734	3.64%	3671	3.50%	717	2.84%	188	4.05%	25560	3.06%
16	5820	4.47%	4673	4.46%	1118	4.43%	190	4.10%	34442	4.13%
17	4586	3.52%	4015	3.83%	753	2.98%	167	3.60%	29020	3.48%
18	5744	4.41%	4590	4.38%	971	3.85%	191	4.12%	34803	4.17%
19	5323	4.09%	4133	3.95%	1153	4.57%	158	3.41%	32148	3.85%
20	2586	1.99%	2030	1.94%	606	2.40%	88	1.90%	17501	2.10%
21	2483	1.91%	1920	1.83%	445	1.76%	93	2.01%	16084	1.93%
22	4473	3.44%	3639	3.47%	760	3.01%	143	3.08%	27759	3.33%
23	4388	3.37%	3653	3.49%	783	3.10%	180	3.88%	30562	3.66%
24	3870	2.97%	3208	3.06%	588	2.33%	142	3.06%	27462	3.29%
25	4786	3.68%	3762	3.59%	795	3.15%	184	3.97%	34126	4.09%
26	4392	3.37%	3165	3.02%	937	3.71%	153	3.30%	27586	3.31%

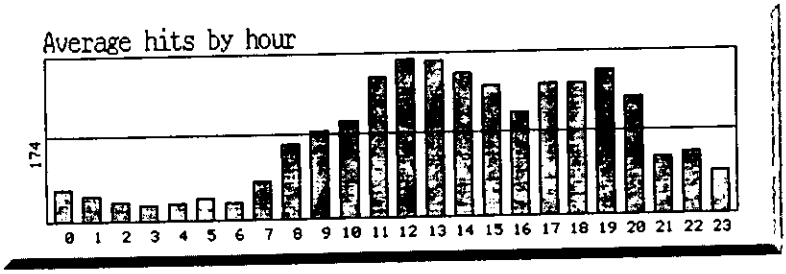
Day	Hits		Files		Pageviews		Sessions		KBytes sent	
27	2342	1.80%	1736	1.66%	699	2.77%	73	1.57%	14632	1.75%
28	1154	0.89%	1001	0.96%	347	1.37%	62	1.34%	6243	0.75%
29	3054	2.35%	2672	2.55%	598	2.37%	130	2.80%	22140	2.65%
30	2934	2.25%	2455	2.34%	505	2.00%	137	2.95%	19023	2.28%
31	4019	3.09%	3220	3.07%	673	2.67%	156	3.36%	23649	2.84%
Total	130177	100.00%	104739	100.00%	25253	100.00%	4638	100.00%	833934	100.00%

Average load



The Top 7 days of the period

No.	Hits		304's		KBytes sent	Date
1	9183	7.05%	413	2.50%	81933	03/Mar/1999
2	6963	5.35%	784	4.75%	43290	10/Mar/1999
3	5848	4.49%	1136	6.88%	33231	09/Mar/1999
4	5820	4.47%	836	5.06%	34442	16/Mar/1999
5	5788	4.45%	1136	6.88%	31256	17/Mar/1999
6	5744	4.41%	797	4.82%	34803	18/Mar/1999
7	5629	4.32%	581	3.52%	30127	19/Mar/1999



The Top 24 hours of the period

No.	Hits		304's		KBytes sent	Date/Time
1	1538	1.18%	17	0.10%	10285	03/Mar/199913:XX:XX
2	1355	1.04%	49	0.30%	15601	03/Mar/199919:XX:XX
3	1341	1.03%	560	3.39%	7115	09/Mar/199920:XX:XX
4	1230	0.94%	9	0.05%	16858	03/Mar/199914:XX:XX
5	1024	0.79%	27	0.16%	5725	03/Mar/199918:XX:XX
6	820	0.63%	260	1.57%	3363	15/Mar/199912:XX:XX
7	789	0.61%	151	0.91%	5325	16/Mar/199911:XX:XX
8	750	0.58%	203	1.23%	4248	05/Mar/199912:XX:XX
9	745	0.57%	82	0.50%	5489	10/Mar/199919:XX:XX
10	740	0.57%	173	1.05%	5618	18/Mar/199913:XX:XX
11	740	0.57%	263	1.59%	3957	10/Mar/199912:XX:XX
12	701	0.54%	28	0.17%	9850	03/Mar/199920:XX:XX

No.	Hits		304's		KBytes sent	Date/Time
13	692	0.53%	127	0.77%	3470	16/Mar/199912:XX:XX
14	686	0.53%	189	1.14%	4455	05/Mar/199917:XX:XX
15	684	0.53%	35	0.21%	6288	11/Mar/199919:XX:XX
16	631	0.48%	143	0.87%	3337	09/Mar/199919:XX:XX
17	630	0.48%	19	0.12%	7365	03/Mar/199915:XX:XX
18	626	0.48%	184	1.11%	3171	31/Mar/199914:XX:XX
19	606	0.47%	66	0.40%	3839	12/Mar/199911:XX:XX
20	602	0.46%	17	0.10%	3899	17/Mar/199912:XX:XX
21	599	0.46%	60	0.36%	5714	23/Mar/199914:XX:XX
22	584	0.45%	32	0.19%	3140	19/Mar/199911:XX:XX
23	573	0.44%	37	0.22%	4180	04/Mar/199915:XX:XX
24	573	0.44%	88	0.53%	2993	18/Mar/199912:XX:XX

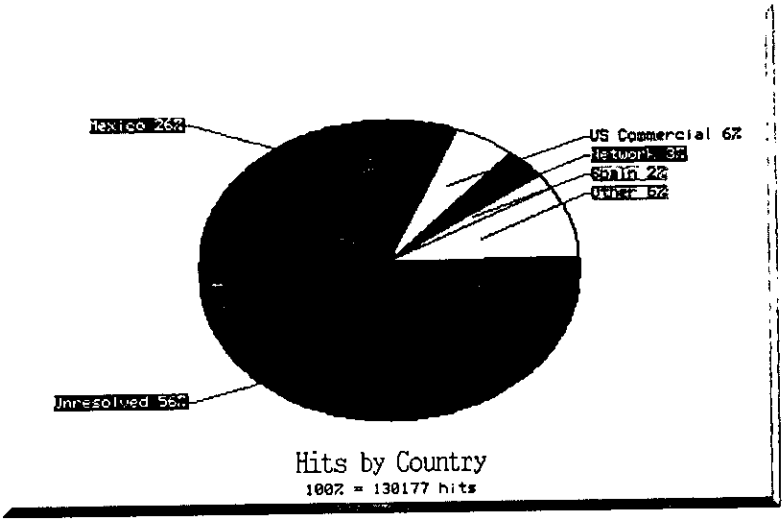
The Top 5 minutes of the period

No.	Hits		304's		KBytes sent	Date/Time
1	387	0.30%	359	2.17%	7	09/Mar/199920:36:XX
2	119	0.09%	46	0.28%	43	15/Mar/199912:24:XX
3	105	0.08%	0	0.00%	1545	03/Mar/199914:23:XX
4	93	0.07%	0	0.00%	726	03/Mar/199913:38:XX
5	92	0.07%	39	0.24%	398	10/Mar/199910:29:XX

The Top 5 seconds of the period

No.	Hits		304's		KBytes sent	Date/Time
1	39	0.03%	37	0.22%	1	09/Mar/199920:36:12
2	38	0.03%	36	0.22%	1	09/Mar/199920:36:11
3	32	0.02%	30	0.18%	1	09/Mar/199920:36:19
4	28	0.02%	26	0.16%	1	09/Mar/199920:36:18
5	24	0.02%	20	0.12%	2	18/Mar/199921:42:56

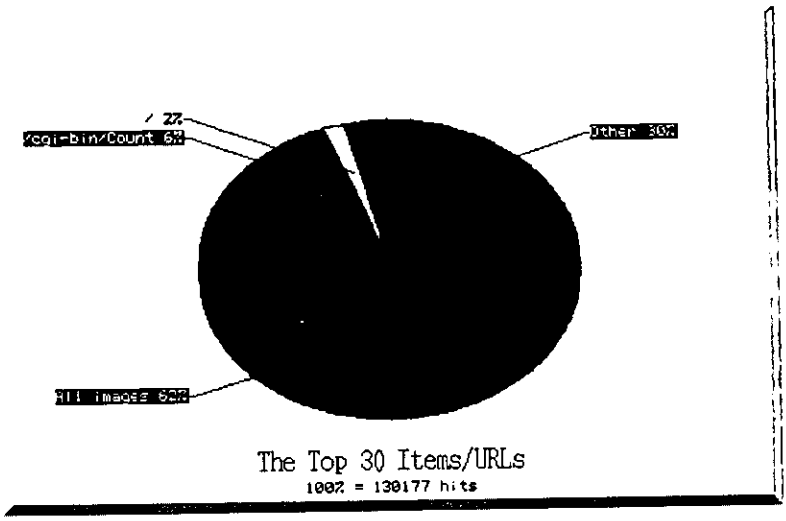
Hits by Country



No.	Hits		304's		KBytes sent	Country
1	73492	56.46%	11960	72.40%	447757	Unresolved
2	34315	26.36%	2638	15.97%	249672	Mexico
3	7595	5.83%	595	3.60%	41608	US Commercial
4	4343	3.34%	283	1.71%	26158	Network
5	2040	1.57%	259	1.57%	11240	Spain
6	1248	0.96%	80	0.48%	7958	Colombia
7	926	0.71%	48	0.29%	6786	US Educational
8	711	0.55%	163	0.99%	3769	France
9	612	0.47%	17	0.10%	5305	Argentina
10	459	0.35%	203	1.23%	1872	Italy
11	407	0.31%	34	0.21%	2347	Chile
12	290	0.22%	87	0.53%	1334	Netherlands
13	259	0.20%	2	0.01%	1416	Peru
14	244	0.19%	22	0.13%	2042	Germany
15	242	0.19%	3	0.02%	1630	Venezuela
16	242	0.19%	9	0.05%	1703	United Kingdom
17	205	0.16%	6	0.04%	1456	Costa Rica
18	203	0.16%	2	0.01%	890	Non-Profit Organization
19	196	0.15%	1	0.01%	1322	Brazil
20	162	0.12%	11	0.07%	821	Canada
21	157	0.12%	3	0.02%	935	Dominican Republic
22	143	0.11%	13	0.08%	1241	Uruguay
23	140	0.11%	2	0.01%	856	Bolivia
24	138	0.11%	0	0.00%	858	Nicaragua
25	136	0.10%	1	0.01%	2566	Belgium
26	103	0.08%	27	0.16%	1018	Malaysia

No.	Hits		304's		KBytes sent	Country
27	103	0.08%	3	0.02%	1476	Philippines
28	99	0.08%	0	0.00%	1124	Sweden
29	90	0.07%	0	0.00%	792	Old style Arpanet
30	74	0.06%	7	0.04%	572	Portugal
31	69	0.05%	0	0.00%	444	Czech Republic
32	68	0.05%	6	0.04%	919	Estonia
33	65	0.05%	0	0.00%	397	Denmark
34	61	0.05%	0	0.00%	456	United States
35	57	0.04%	0	0.00%	373	Japan
36	48	0.04%	0	0.00%	254	Ecuador
37	48	0.04%	6	0.04%	241	Turkey
38	44	0.03%	0	0.00%	317	Poland
39	42	0.03%	8	0.05%	369	El Salvador
40	35	0.03%	14	0.08%	72	South Africa
41	31	0.02%	2	0.01%	171	New Zealand (Aotearoa)
42	28	0.02%	0	0.00%	117	Korea (South)
43	28	0.02%	0	0.00%	169	Slovenia
44	27	0.02%	0	0.00%	152	Thailand
45	22	0.02%	0	0.00%	135	Cuba
46	22	0.02%	4	0.02%	87	Switzerland
47	19	0.01%	0	0.00%	99	Singapore
48	18	0.01%	0	0.00%	194	Australia
49	17	0.01%	0	0.00%	85	Indonesia
50	13	0.01%	0	0.00%	81	Paraguay
51	8	0.01%	0	0.00%	53	Taiwan
52	8	0.01%	0	0.00%	53	Bulgaria
53	8	0.01%	0	0.00%	53	International
54	5	0.00%	0	0.00%	13	Norway
55	4	0.00%	0	0.00%	24	US Government
56	3	0.00%	0	0.00%	30	Croatia (Hrvatska)
57	3	0.00%	0	0.00%	29	Ireland
58	2	0.00%	0	0.00%	29	Russian Federation

The Top 30 items/URLs



More details

No.	Hits		304's		KBytes sent	URL
1	80796	62.07%	12964	78.48%	637791	All images
2	7737	5.94%	0	0.00%	3094	/cgi-bin/Count.cgi
3	2786	2.14%	937	5.67%	8785	/
4	1460	1.12%	225	1.36%	3654	/homecsp.html
5	528	0.41%	36	0.22%	1791	/areas/areas.html
6	512	0.39%	70	0.42%	1657	/infogrl/infogrl.html
7	467	0.36%	400	2.42%	19	/torre-ii/camara.html
8	410	0.31%	37	0.22%	1356	/proyectos/proyectos.html
9	348	0.27%	30	0.18%	1078	/consulta/consulta.html
10	339	0.26%	42	0.25%	350	/torre-ii/index.htm
11	335	0.26%	52	0.31%	175	/torre-ii/inicio2.htm
12	324	0.25%	40	0.24%	1166	/torre-ii/menu3.htm
13	315	0.24%	39	0.24%	2074	/directorio/directorio.html
14	301	0.23%	9	0.05%	830	/estudios/estudios.html
15	288	0.22%	24	0.15%	1140	/share/share.html
16	270	0.21%	38	0.23%	2648	/torre-ii/_fpclass/fphover.class
17	259	0.20%	14	0.08%	725	/servicios/servicio.html
18	255	0.20%	40	0.24%	429	/torre-ii/_fpclass/fphoverx.class
19	248	0.19%	17	0.10%	741	/cecco/cecco.html
20	241	0.19%	52	0.31%	359	/cecco/iawq/
21	203	0.16%	8	0.05%	1242	/auto/
22	187	0.14%	0	0.00%	1402	/cgi-bin/ws.exe/websql.dir/webbdaii/tesbdaii.hts
23	173	0.13%	80	0.48%	279	/gch/publics.htm
24	169	0.13%	0	0.00%	158	/cgi-bin/ws.exe/websql.dir/webbdaii/datper3.hts

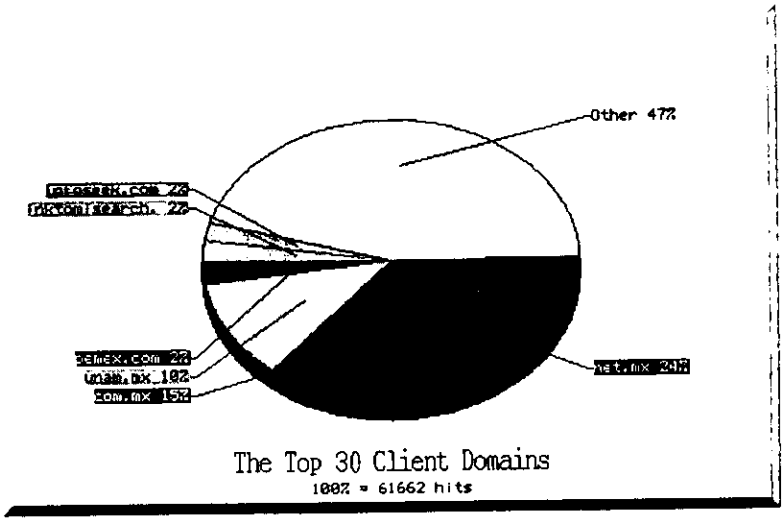
25	167	0.13%	4	0.02%	576	/ccco/cursos.html
26	166	0.13%	0	0.00%	1413	/cgi-bin/ws.exe/websql.dir/webbdaii/inlbdaii.hts
27	158	0.12%	25	0.15%	397	/ccco/iawq/home.html
28	151	0.12%	22	0.13%	554	/infogrl/reglamentos/
29	150	0.12%	23	0.14%	52	/ccco/iawq/index1.html
30	149	0.11%	19	0.12%	475	/mictlan/

The 10 least frequently accessed items/URLs

More details

No.	Hits		304's		KBytes sent	URL
10	1	0.00%	0	0.00%	3	//homeesp.html
9	1	0.00%	0	0.00%	2	/anes/seciv.html
8	1	0.00%	0	0.00%	8	/areas/../directorio/directorio.html
7	1	0.00%	0	0.00%	5	/asocia/secx.html
6	1	0.00%	0	0.00%	2	/areas/isi/../../torre-ii/index.htm
5	1	0.00%	0	0.00%	4	/areas/../areas/areas.html
4	1	0.00%	0	0.00%	3	/areas/isi/../../servicios/servicio.html
3	1	0.00%	1	0.01%	0	/anes/seciiic.html
2	1	0.00%	0	0.00%	4	/asocia/seciiic.html
1	1	0.00%	0	0.00%	4	/areas/isi/../../consulta/consulta.html

The Top 30 client domains



More details

No.	Hits		304's		KBytes sent	Domain
1	14563	23.62%	1397	27.15%	94326	net.mx
2	8982	14.57%	447	8.69%	80774	com.mx
3	6031	9.78%	302	5.87%	44074	unam.mx
4	1193	1.93%	56	1.09%	7017	pemex.com
5	1160	1.88%	0	0.00%	6755	inktomisearch.com
6	988	1.60%	310	6.02%	1945	infoseek.com
7	804	1.30%	148	2.88%	4654	itesm.mx
8	692	1.12%	4	0.08%	6460	gob.mx
9	559	0.91%	0	0.00%	3142	excite.com
10	546	0.89%	69	1.34%	3078	infoserv.net
11	504	0.82%	72	1.40%	2900	edu.co
12	492	0.80%	24	0.47%	2826	imp.mx
13	459	0.74%	78	1.52%	1778	edu.mx
14	445	0.72%	0	0.00%	2816	NET.co
15	440	0.71%	14	0.27%	4120	com.ar
16	436	0.71%	0	0.00%	1847	dec.com
17	428	0.69%	33	0.64%	2257	uu.net
18	388	0.63%	0	0.00%	1943	atext.com
19	329	0.53%	179	3.48%	848	retevision.es
20	314	0.51%	200	3.89%	458	polimi.it
21	298	0.48%	183	3.56%	707	uacam.mx
22	297	0.48%	25	0.49%	994	rubis.net
23	282	0.46%	63	1.22%	1826	unam.mx1
24	278	0.45%	8	0.16%	2103	com.co

No.	Hits		304's		KBytes sent	Domain
25	272	0.44%	47	0.91%	1308	acnet.net
26	269	0.44%	5	0.10%	3009	unam.mx71
27	263	0.43%	16	0.31%	2055	entelchile.net
28	229	0.37%	21	0.41%	1284	itam.mx
29	225	0.36%	161	3.13%	984	grh-informatique.fr
30	217	0.35%	6	0.12%	1243	ibm.net

Apéndice C

Graphic Pumas.iingen.unam.mx

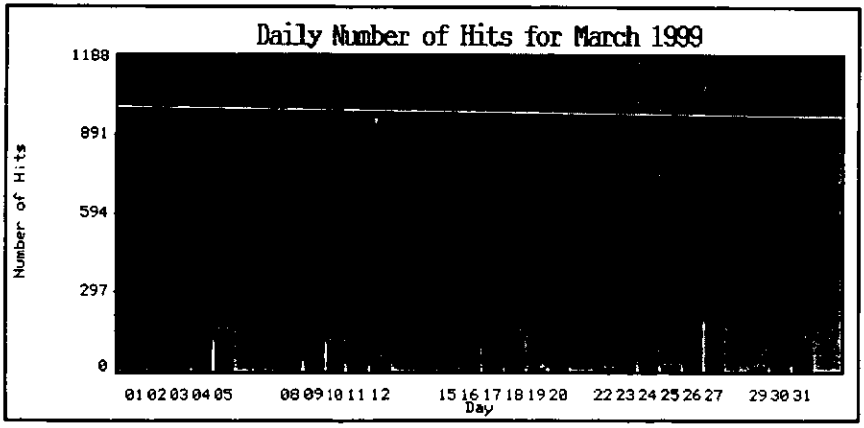
Index

Summary

Period Covered: 12:11:19 01 March 1999 to 22:05:05 31 March 1999
Total Files Transfered: 3,329
Total Bytes Transfered: 670,163,178
Unique Hosts: 45

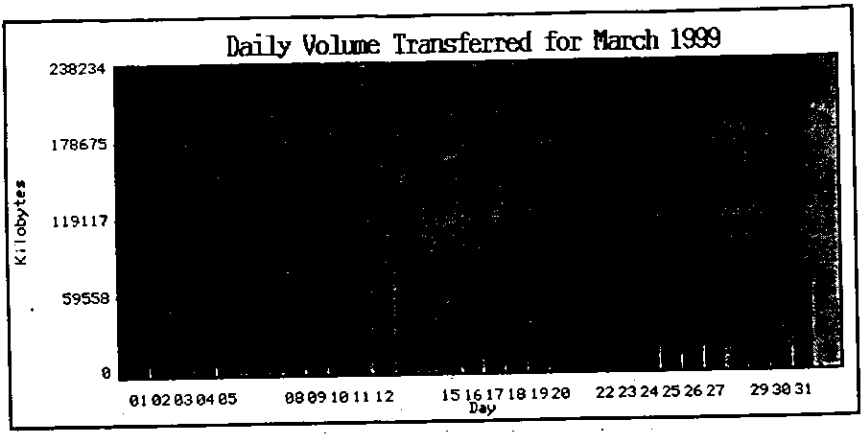
Return to Index

Daily Number of Hits



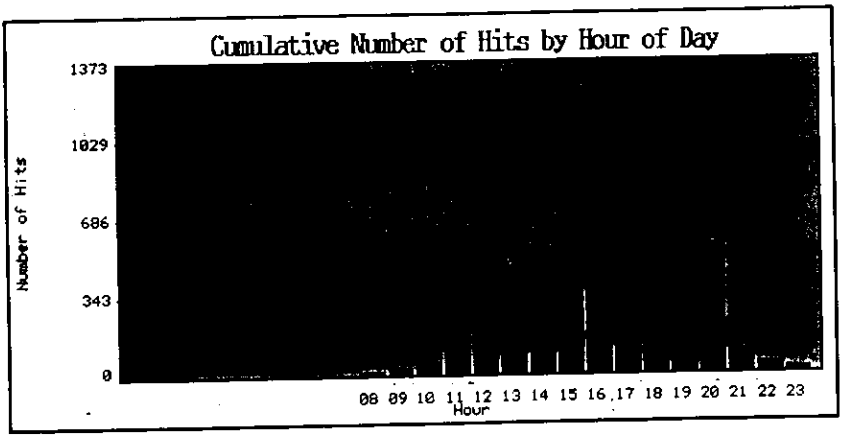
Return to Index

Daily Volume Transferred



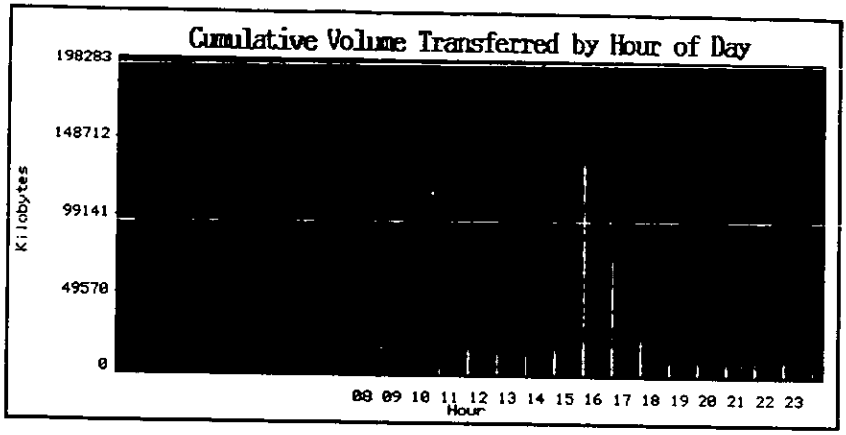
Return to Index

Cumulative Number of Hits by Hour of Day



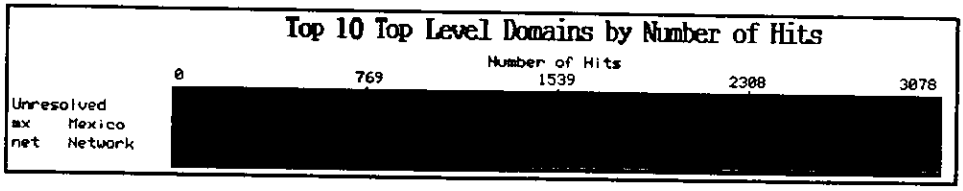
Return to Index

Cumulative Volume Transferred by Hour of Day



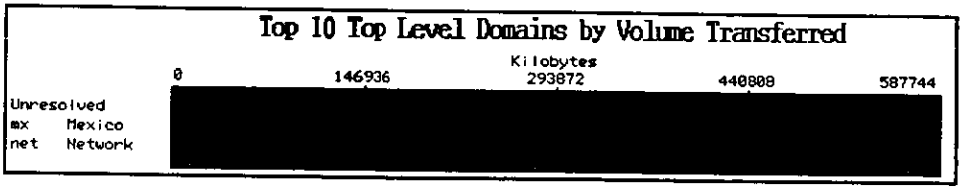
Return to Index

Top 10 Top Level Domains by Number of Hits



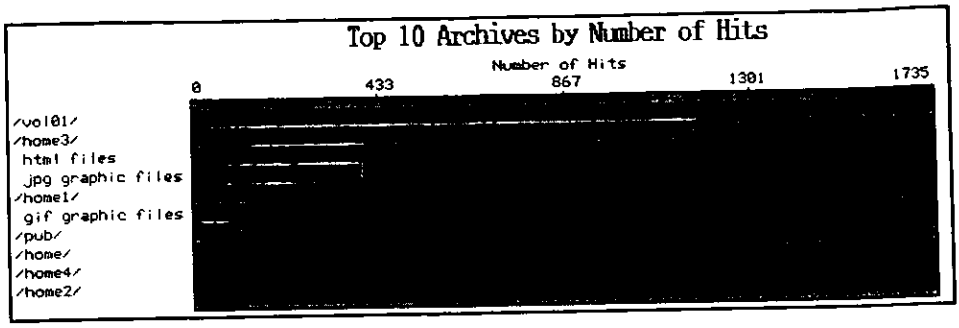
Return to Index

Top 10 Top Level Domains by Volume Transferred



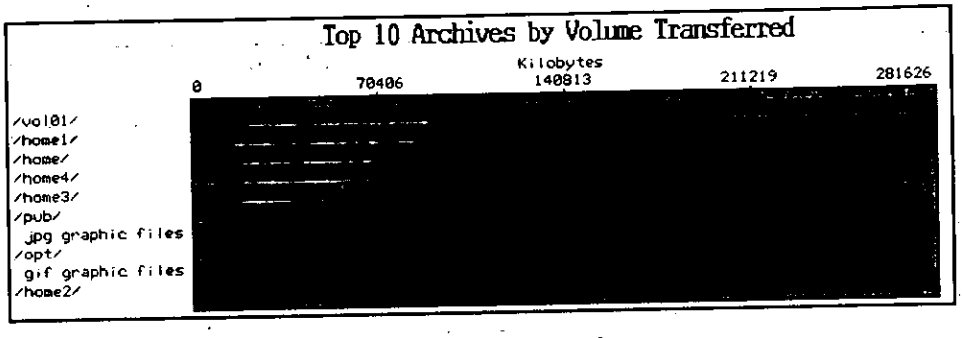
Return to Index

Top 10 Archives by Number of Hits



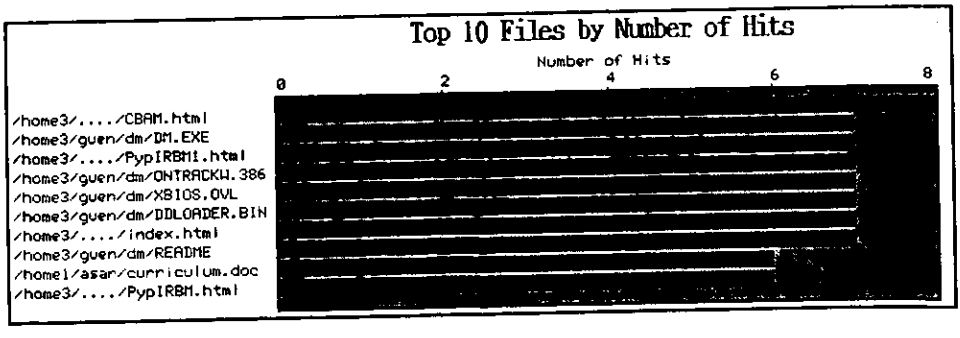
Return to Index

Top 10 Archives by Volume Transferred



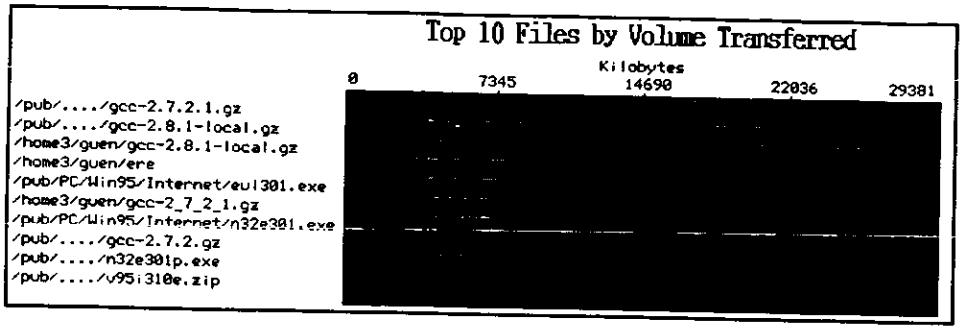
Return to Index

Top 10 Files by Number of Hits



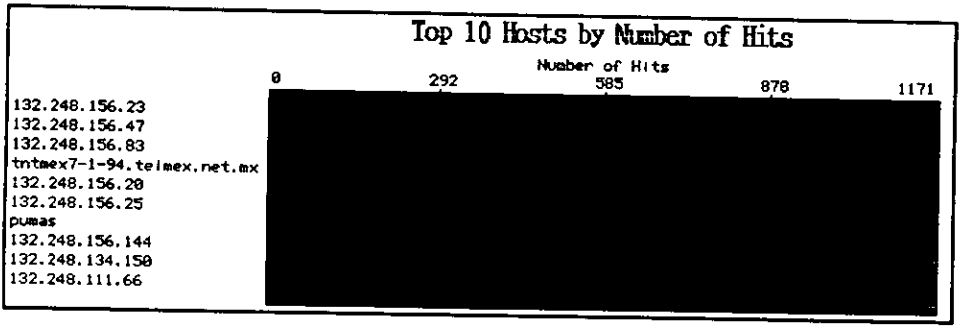
Return to Index

Top 10 Files by Volume Transferred



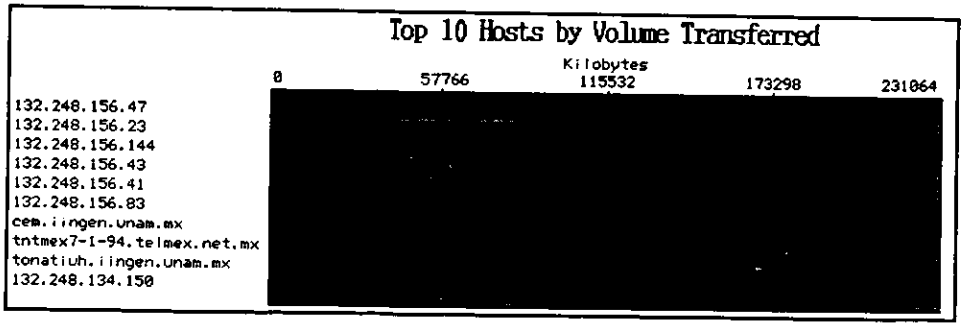
Return to Index

Top 10 Hosts by Number of Hits



Return to Index

Top 10 Hosts by Volume Transferred



Bibliografía

Libros

Black Uyless; OSI Model for Computer Communications Standards; Prentice Hall, 1992

Black Uyless; Network Management Standards: Snmp, Cmip, Mibs, and Objects Libraries; MacGraw – Hill, 1994

Cockcroft Adrian, Pettit Richard; Sun Performance and Tuning, Java and the Internet; Prentice Hall 1998

Comer Douglas E.; Internetworking with TCP/IP: Principles, Protocol, and Architecture. Volume I; Prentice Hall, 1996

Deitel Harvey M.; An introduction to Operating Systems; Addison – Wesley, 1990

Goscinski Andrzej; Distributed Operating Systems; Addison – Wesley, 1991

Halshall Alfred; Data Communications, Computer Networks & Open Systems; Addison – Wesley, 1992

Leinwand Allan, Fang – Conroy Karen; Network Management: A practical Perspective (Unix and Open Systems Series); Prentice Hall 1995

Rose T. Marshall; The Simple Book: An Introduction to Internet Management; Prentice Hall 1996

Tanenbaum Andrew S.; Computer Networks 3a ed.; Prentice Hall, 1996

Terplan Kornel, Abre Shaku; Effective Management of Local Area Networks, MacGraw Hill, 1992

Santifaller Michel; TCP / IP and NFS (Internetworking in Unix environment); Addison – Wesley, 1991

Stallings William; SNMP, SNMPv2 and CMIP, The Practical Guide to Network – Management Standards; Addison – Wesley, 1995

Tesis

Carnacho Palacios Gustavo, Ramirez Acevedo Artemia, Septien Nava Ricardo; Análisis y diseño de la infraestructura tecnológica para la Red de alta velocidad del Instituto de Ingeniería; tesis de licenciatura, UNAM 1997

Callejas Mancilla Isaías E., Guati Rojo Arenas Mariana; Desarrollo del modulo de interfase para integración a la red académica BITNET e implementación de servicios y políticas de administración bajo ambiente UNIX para el equipo Microvax 3400, como consecuencia de la sustitución del Mainframe IBM 4381; tesis de licenciatura, Facultad de Ingeniería, UNAM 1995.

Mendoza Romero Fernando, Martínez P. Rosa Alva; Facilidades de computo distribuido en el Instituto de Ingeniería: Implementación y desarrollo de mecanismos de información, procesamiento y administración; tesis de licenciatura, Facultad de Ingeniería, UNAM 1995.

Morchio Secul Javier E.; Metodología de preparación, presentación y evaluación de proyectos de equipamiento computacional; tesis de licenciatura, Universidad Católica de Chile, 1987.

Rosales Romero Federico, Muñoz Muñoz Mónica; Diseño e implementación del sistema de evaluación de productividad de una compañía afianzadora; tesis de licenciatura, Facultad de Ingeniería, UNAM 1995

Request for Comments (RFC)

1155 Structure and identification of management information for TCP/IP-based internets

1156 Management Information Base for network management of TCP/IP-based internets

1157 Simple Network Management Protocol (SNMP)

1158 Management Information Base for network management of TCP/IP-based internets: MIB-II

1212 Concise MIB definitions

1270 SNMP Communications Services

1441 Introduction to version 2 of the Internet-standard Network Management Framework

1445 Administrative Model for version 2 of the Simple Network Management Protocol (SNMPv2)

1446 Security Protocols for version 2 of the Simple Network Management Protocol (SNMPv2)

1452 Coexistence between version 1 and version 2 of the Internet-standard Network Management Framework

1514 Host Resources MIB

1552 The PPP Internetworking Packet Exchange Control Protocol (IPXCP)

1902 Structure of Management Information for Version 2 of the Simple Network Management Protocol (SNMPv2)

Direcciones Web

Proveedores

Cabletron Systems Inc.;
http://www.cabletron.com

Hewlett Packard Company;
http://www.hp.com

Sun Microsystems Inc.;
http://www.sun.com

Independientes

Abstract Syntax Notation One;
http://www.rad.co.il/web/networks/1995/snmp/asn1.htm

GWFstast 1.1;
http://irb.cs.uni-magdeburg.de/~elkner/webtools/gwfstats.shtml

HTTPD log analyzing tools, Uppsala University;
http://www.uu.se/Software/Analyzers/

MRTG Multi Router Traffic Grapher;
http://ee-staff.ethz.ch/~oetiker/webtools/mrtg/mrtg.html#WHAT

RFC Index, Carnegie Mellon
http://andrew2.andrew.cmu.edu/rfc/rfc-front.html

Sun Performance Information;
http://www.sun.com/sun-on-net/performance/