

UNIVERSIDAD CATÓLICA ANDRÉS BELLO

FACULTAD DE INGENIERÍA

ESCUELA DE INGENIERÍA DE TELECOMUNICACIONES

TESIS
IP2006
S3
V.1

Diseño e Implementación de un Sistema para el Control, Supervisión y Gestión de Redes basadas en el protocolo IPv4.

TRABAJO ESPECIAL DE GRADO

presentado ante la

UNIVERSIDAD CATÓLICA ANDRÉS BELLO

Como parte de los requisitos para optar al título de

INGENIERO EN TELECOMUNICACIONES

REALIZADO POR

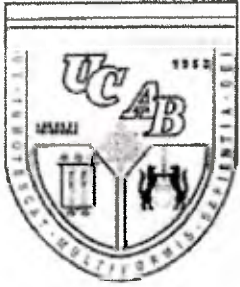
Miguel Ángel Sabal Matheus

PROFESOR GUÍA

Ing José Gregorio Cotua.

FECHA

Caracas, 13 Octubre de 2006.



UNIVERSIDAD CATÓLICA ANDRÉS BELLO

FACULTAD DE INGENIERÍA

ESCUELA DE INGENIERÍA DE TELECOMUNICACIONES

Diseño e implementación de un Sistema para el Control, Supervisión y Gestión de Redes basadas en el protocolo IPv4.

REALIZADO POR

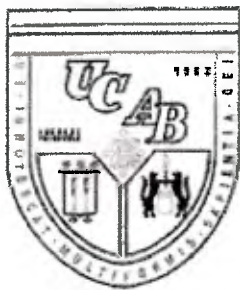
Miguel Ángel Sabal M.

PROFESOR GUÍA

Ing. José Cotua

FECHA

Caracas, 13 de octubre de 2006.

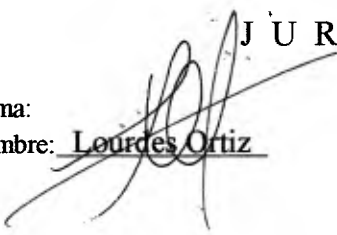


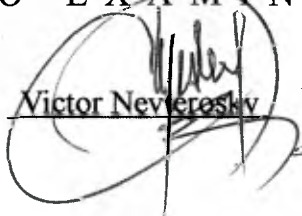
UNIVERSIDAD CATÓLICA ANDRÉS BELLO
FACULTAD DE INGENIERÍA
ESCUELA DE INGENIERÍA DE TELECOMUNICACIONES

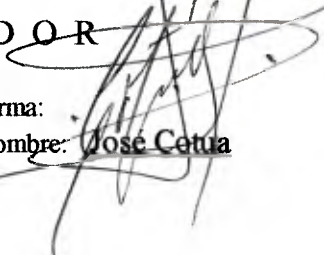
Diseño e implementación de un Sistema para el Control, Supervisión y Gestión de Redes basadas en el protocolo IPv4.

Este jurado; una vez realizado el examen del presente trabajo ha evaluado su contenido con el resultado: VEINTE (20) PUNTOS - MENCIÓN DE MENCION

JURADO EXAMINADOR

Firma: 
Nombre: Lourdes Ortiz

Firma: 
Nombre: Victor Nevleresky

Firma: 
Nombre: José Cótua

REALIZADO POR

Miguel Ángel Sabal Matheus

PROFESOR GUÍA

Ing. José G. Cótua

FECHA

Caracas, 13 de Octubre de 2006.

Dedicatoria.

...Después de tanto esfuerzo.. expresar en tan pocas líneas a quien dedico este Trabajo Especial de Grado, no es nada sencillo. Sin embargo creo que lo más sincero y justo es dedicárselo a las personas que más quiero en el mundo y que siempre estuvieron presentes durante mi carrera universitaria. Este trabajo se lo dedico a ustedes: MI FAMILIA.!!

...A mi papá que me ha acompañado en todas las “Aventuras” de mi vida, ha sido un gran ejemplo, una verdadera inspiración de la que siempre me sentiré orgulloso. A mi mamá, por haberme inculcado los valores que tengo, por darme ánimo, por consentirme y por valorar todo lo que hago. Mis hermanos: Carlucho, por enseñarme a disfrutar la vida y por estar pendiente de mi y Mary, mi hermanita linda a quien adoro y con quien cuento para todo. A mi tía Elvita, quien ha sido un verdadero apoyo, una gran madrina, profesora y amiga. Y finalmente a Meli, por estar conmigo en todo momento y ser una persona tan especial en mi vida.

Este trabajo es para ustedes y el esfuerzo que implica, es el reflejo de lo maravillosos son para mi.

Que lo disfruten...

Agradecimientos.

Quiero agradecer principalmente a Dios, por haberme acompañado en este largo camino. A mi familia, a quienes dediqué el Trabajo Especial de Grado y quienes también merecen el mayor de los agradecimientos.

Quisiera dedicar estas líneas para mencionar también a aquellas personas que fueron parte importante de mi vida universitaria y que significaron un gran apoyo en todo lo que hice:

Comienzo por Santi, quien quizás hubiese podido suscribir las páginas de este documento junto a mí, pero que la vida le dio una mejor oportunidad. Gracias amiga.

Alecita, aprendimos muchas cosas juntos, entre esas a no dejarnos vencer por nada... y nada nos venció, gracias.

A mis grandes Amigos: Majo, Mich, Miguel, Toto, Jenn, Geral, Eli, Marián, Lore, y todos aquellos que me falte mencionar, con quienes compartí momentos únicos. Creo que conocerlos a ustedes en esta universidad, ha sido uno de las mejores cosas que me han pasado.

Agradezco la oportunidad que me dio la vida de haber dedicado 3 años de mi carrera al Centro de Estudiantes de Ingeniería y de haber conocido a tanta gente valiosa.

A mi tutor, Ing. José Cotua, por acompañarme y estimularme en la elaboración de este proyecto.

Agradezco a Lourdes Ortiz, por se parte importante en mi formación como ingeniero, como persona, como amigo, gracias...

Agradezco a la Primera Promoción de Telecomunicaciones de Venezuela, por marcar pauta en el país.

A Mayerling Díaz, quien significó mucho para mí durante mis primeros años de la carrera y a quien aprecio y valoro mucho.

Finalmente, agradezco la confianza y el apoyo que me brindó siempre mi escuela de telecomunicaciones, a través de su Directora Mayra Narváez y aún más la Facultad de Ingeniería, a través de una persona a quien valoro y admiro mucho, el Decano Ing. Rafael Hernández.

Índice General.

Dedicatoria.....	i
Agradecimientos.....	ii
Índice General.....	iii
Índice de Figuras.....	v
Índice de Tablas.....	vi
Resumen.....	vii
Introducción.....	1
Capítulo I. Problema.....	3
I.1 Planteamiento del Problema.....	3
I.2 Objetivo General.....	4
I.3 Objetivos Específicos.....	4
I.4 Justificación.....	5
I.5 Alcance y Limitaciones.....	6
Capítulo II. Metodología.....	7
II.1 Investigación de Mercado.....	7
II.2 Estudio Comparativo.....	7
II.3 Recolección de Datos. Documentación Teórica.....	8
II.4 Diseño del sistema.....	8
II.5 Desarrollo del Fototipo. Programación orientada a eventos Y Orientada a objetos.....	8
Capítulo III. Marco Teórico.....	9
III.1 Arquitectura SNMP.....	9
III.2 Propósitos de la Arquitectura SNMP.....	12
III.3 Elementos de la Arquitectura.....	13
III.4 Especificaciones del Protocolo SNMP.....	15
III.4.1 Elementos de Procedimiento.....	16
III.4.2 Estructura de una PDU.....	18
III.4.3 GetRequest-PDU y GetNextRequest-PDU.....	18
III.4.4 SetRequest-PDU.....	19
III.4.5 GetResponse-PDU.....	19
III.4.6 Trap-PDU.....	21
Capítulo IV. Desarrollo.....	24
IV.1. Documentación Teórica Protocolo Simple de Gestión de Redes..	25
IV.2. Investigación de las Herramientas existentes en el Mercado.....	26
IV.3 Análisis de los Requerimientos Software.....	28
IV.4.Familiarización con los compiladores MIB y lenguaje ASN.1....	36
IV.5 Diseño del Sistema.....	39
IV.5.1 Módulo de Sesión y Configuración.....	40
IV.5.2 Módulo de Supervisión.....	41
IV.5.3 Módulo de Control.....	42
IV.5.4 Módulo de Reportes.....	43
IV.6 Desarrollo e Implementación del Prototipo.....	45

IV.7 Evaluación de Alternativas para generar reportes móviles.....	45
IV.8 Pruebas con datos contruidos y reales en una red TCP/IP de 10 estaciones	46
Capítulo V. Resultados.....	48
V.1 Documentación teórica	48
V.2 Investigación de Mercado.....	48
V.3 Requerimientos de Software.....	49
V.4 Diseño del Sistema.....	53
V.5 Prototipo.....	54
V.5.1 Módulo de Sesión y Configuración	54
V.5.2 Módulo de Supervisión.....	59
V.5.3 Módulo de Control.....	61
V.5.4 Módulo de Reportes.....	63
V.6 Pruebas.....	65
Capítulo VI. Conclusiones y Recomendaciones	68
Apéndices.....	76
I. Cuadro Resumen del Estudio de Mercado.....	77
II. Estudio Comparativo Lenguajes de Programación.....	78
III. Manual de Usuario.....	79
IV. Manuel del Sistema.....	80
Anexos.....	81

Índice de Figuras.

Figura 1: Arquitectura Protocolo SNMP.....	11
Figura 2: Extracto de imagen Window del Management Information Base (MIB).....	13
Figura 3: Modelo de funcionamiento del Protocolo SNMP.....	16
Figura 4: Modelo Relacional Etapas de Desarrollo TEG.....	25
Figura 5: Ejemplo de uso del MIB-COMPILER. Variable mostrada sysobjectId.....	38
Figura 6: Diseño General del Sistema de Control, Supervisión y Gestión.	53
Figura 7: Pantalla de Inicio del Prototipo. Módulo de Sesión y Configuración.....	55
Figura 8: Diagrama de Colaboración Fase1 del Módulo de Sesión y Configuración.....	56
Figura 9: Diagrama de Secuencia Fase 1 del Módulo de Sesión y Configuración.....	57
Figura 10: Diagrama de Secuencia Fase 2 del Módulo de Sesión y Configuración.....	58
Figura 11: Pantalla Estándar Buscador de MIB.....	60
Figura 12: Diagrama de Secuencia Módulo de Supervisión. Buscador MIB.....	60
Figura 13: Pantalla Estándar Módulo de Control.....	61
Figura 14: Pantalla Estándar Módulo de Control. Modificar Campo Remoto.....	61
Figura 15: Pantalla Estándar Módulo de Reporte.....	63
Figura 16: Diagrama de Secuencia Módulo Reporte.....	64
Figura 17: Prueba de Datos Reales. Datos Enviados Vs. Datos Recibidos	65
Figura 18: Prueba de Datos Reales. Unidades de Almacenamiento.....	66
Figura 19: Prueba de Datos Reales. SNMP MIB TABLE.....	67

Índice de Tablas.

Tabla 1: Cuadro Comparativo de las Herramientas Existentes en el Mercado.....	49
Tabla 2: Cuadro Comparativo Sistema Operativo Windows Vs. Linux.....	51
Tabla 3: Cuadro Comparativo de los Lenguajes de Programación.....	52
Tabla 4: Sinopsis de datos Promedios Supervisión de 2 estaciones (48 Horas).....	68

Resumen

Considerando la importancia que reviste la supervisión y gestión oportuna de las redes de telecomunicaciones en el mundo actual, se planteó el “Diseño e implementación de un sistema para el control, supervisión y gestión de redes basadas en el protocolo IPv4”. Basado en un estudio tanto teórico como empírico, se realizó la investigación del *Protocolo Simple de Gestión de Redes* (SNMP), sobre el cual se soporta el desarrollo del sistema de gestión. Para la determinación de las características innovadoras de la aplicación, se realizó un estudio de mercado donde se evaluaron las diferentes herramientas que ofrecen servicios de gestión y supervisión. Adicionalmente, se realizó un análisis técnico que permitió decidir sobre qué plataforma realizar el desarrollo: Sistema Operativo Windows, lenguaje de programación orientado a eventos, ambiente de desarrollo Borland Builder C++. La programación del prototipo se caracterizó en dos macro etapas: a) El desarrollo de las funcionalidades de supervisión a través de la implementación del protocolo SNMP y b) El desarrollo y evaluación del sistema de control, para la programación de las tablas de ruta de forma remota. Finalmente, se incluyó la funcionalidad de reportes de falla en tiempo real, a través de correo electrónico, después de realizado el estudio de alternativas existentes (sms, t-motion, email). El Trabajo Especial de Grado contiene la validación del Prototipo desarrollado en una red TCP/IP de 10 estaciones lo cual se presenta a través de las pantallas de resultados y los diferentes casos de uso.

Palabras Clave: *Sistema de Gestión, SNMP, Sistema de Supervisión, TCP/IP.*

INTRODUCCIÓN

El crecimiento vertiginoso de las redes de comunicación en la era moderna, es un acontecimiento que ocurre en forma acelerada a una escala global, cambiando drásticamente la forma en que se comunican tanto las personas como los dispositivos.

En la actualidad gran cantidad de negociaciones y diversas actividades, así como numerosas operaciones comerciales son realizadas en forma electrónica, desarrollando extensas redes de computadoras que día a día se multiplican. Gracias a las innumerables ventajas de las redes, se mantiene en constante movimiento y comunicación la sociedad moderna.

Los administradores de redes se enfrentan hoy en día con crecientes desafíos, que hacen cada día más populares la utilización de éstas. La sobrecarga intensa de las redes, limitan sus tiempos de respuesta, debido a esto, se ven en la necesidad de buscar caminos viables no solo para mejorar la latencia e incrementar el ancho de banda de la red, sino además para detectar cualquier comportamiento anómalo que la pueda estar afectando. Controlando así, los costos y mantenimientos de la plataforma de red con la que se trabaja.

Por lo general, quienes se encargan de la gestión de las redes de comunicación están conscientes del costo de una estación de trabajo y todo el hardware que implica, así como el costo que requiere el mantenimiento y servicio técnico de una red. Es allí donde se pone de manifiesto la importancia de desarrollar una herramienta que permita supervisar y controlar todos los eventos que ocurren en las estaciones que forman parte de las redes. Estos productos pueden aliviar la carga de trabajo, asumiendo y automatizando muchas tareas.

El presente Trabajo Especial de Grado, describe el proceso de desarrollo del Sistema, siguiendo la estructura que se presenta a continuación:

En el Capítulo I, se introduce al lector en el contexto del tema de la investigación “Diseño e Implementación de un Sistema para el Control, Supervisión y Gestión de redes basadas en el protocolo IPv4”, describiendo el problema planteado y los objetivos que se buscan alcanzar con el desarrollo de la investigación, así como las limitaciones y alcances que se establecieron en la planificación.

En el Capítulo II, se exponen los fundamentos teóricos del contexto de la investigación, describiendo las características, arquitectura y funcionamiento del protocolo SNMP, sobre el cual se soporta todo el desarrollo tecnológico de la aplicación.

En el Capítulo III, se define la metodología empleada para el desarrollo de la aplicación. Se describen las 5 estrategias que componen la fusión metodológica utilizada para alcanzar los objetivos planteados.

En el Capítulo IV se explica en detalle el desarrollo del Trabajo Especial de Grado, precisando algunos cambios que se realizaron durante la investigación y describiendo las actividades y procedimientos que dieron lugar al prototipo del sistema implementado.

El Capítulo V, presenta los resultados finales de este trabajo de grado, mostrando a través de hechos la veracidad de las actividades realizadas y la relación de cada resultados con los objetivos general y específicos que se plantearon.

El Capítulo VI, presenta las conclusiones y recomendaciones derivadas del trabajo realizado.

CAPÍTULO I.

PLANTEAMIENTO DEL PROBLEMA

I.1 Planteamiento del problema

Con el crecimiento del sector de las telecomunicaciones, las redes se han convertido en una herramienta esencial para la optimización de procesos y capital humano en cualquier organización. Los sistemas de control y supervisión de redes, resultan indispensables para todas las empresas e instituciones que tienen como objetivo explotar los canales de telecomunicaciones, pues se hace cada vez más importante conocer de forma oportuna, rápida y eficiente, la condición de los nodos que forman parte de la topología de una red, como enrutadores, conmutadores, enlaces microondas, ordenadores personales, entre otros.

Hoy en día, los sistemas de supervisión y control de redes que ofrece el mercado, son sumamente costosos y su valor se incrementa a medida que aumenta el número de estaciones. Aplicaciones reconocidas a escala global como ORION, desarrollada por el equipo de administración de redes de Solarwinds, tiene un costo que oscila entre \$ 1.945, para topologías de hasta 16 estaciones y \$ 20.000, para redes que lleguen a más de 1000 estaciones. Por esta razón, el uso de dichos sistemas se ha limitado a los Proveedores de Servicios de Internet (ISP) o a las instituciones con gran capacidad económica.

Además de la poca accesibilidad que tienen estas herramientas por la inversión económica que implican, la mayoría de los sistemas de supervisión son estáticos, es decir, no son capaces de tomar decisiones en cuanto al enrutamiento del tráfico en caso de existir congestión o pérdida del servicio en alguna estación de la red, siendo esto de vital importancia para organizaciones en las que se realizan numerosas operaciones a través de sus redes.

I.2 Objetivo General

Desarrollar un sistema de control y gestión, que permita la supervisión de todos los nodos que integran una red de cualquier topología basada en el protocolo de red IPv4, a través de la implementación del protocolo SNMP, que de forma dinámica, tenga la capacidad de tomar decisiones con respecto al enrutamiento del tráfico dentro de la red.

I.3 Objetivos Específicos

- Realizar un estudio detallado de las herramientas existentes en el mercado que permitan el control y supervisión de redes, con el objeto de precisar sus funcionalidades, ventajas y desventajas para verificar la factibilidad de desarrollar una aplicación competitiva y que cumpla con los requerimientos establecidos.
- Determinar en qué plataforma tecnológica realizar el diseño, Microsoft o Linux, tomando en cuenta para ello factores como los requerimientos planteados, complejidad del diseño y posibilidad de desarrollo en cada plataforma.
- Investigar de forma detallada el protocolo de control de redes SNMP (*Simple Network Management Protocol*), para su implementación en el sistema de gestión.
- Implementar la aplicación para generar reportes vía correo electrónico de los eventos que ocurran en cualquier estación de la red en gestión.
- Evaluar las alternativas existentes para generar reportes móviles en tiempo real (SMS, T-MOTION, VIPER, entre otros), precisando cuál de

ellas resulta más adecuada para garantizar una supervisión que proporcione ventajas de movilidad.

- Realizar el diseño del sistema de gestión y supervisión implementando un prototipo que cumpla con las funcionalidades establecidas y las ventajas que presenta este diseño ante las herramientas existentes en el mercado.
- Verificar el funcionamiento del prototipo a través de distintas pruebas sobre una red TCP/IP de al menos 10 nodos.

I.4 Justificación

Dado el gran auge que han tenido las redes en los últimos años, surge la necesidad de tener un control preciso de las actividades que se llevan a cabo en las mismas. Al implementar un sistema que permita obtener en forma rápida y veraz estadísticas y reportes del estado de la red, se puede detectar con mayor exactitud cualquier irregularidad que ocurra de forma muy eficiente. Esto podrá lograrse a través de un prototipo de usuario final que le permita a éste realizar consultas haciendo uso de una interfaz gráfica amigable, logrando un alto nivel de flexibilidad para la obtención de información, permitiendo confirmar cualquier sospecha que pueda existir con respecto a cierto comportamiento de la red. Al desarrollar la aplicación de esta forma, se puede cubrir prácticamente cualquier solicitud que el usuario requiera sin importar el nivel de detalle al cual desee llegar.

Las facilidades que proporciona este sistema con respecto a los existentes en el mercado, además de los costos evidentemente, es el hecho de poder realizar un control dinámico de los eventos que ocurren en las estaciones de trabajo, solucionando de forma casi inmediata, remota y automática congestiones y caídas del sistema.

I.5 Alcance y Limitaciones.

El Sistema se limitará a redes basadas en el protocolo IPv4. Su diseño se realizará para operar hasta un máximo de 60 nodos. Además, deberá ser capaz de proveer la siguiente información en tiempo real:

1.- Disponibilidad: Si un nodo se encuentra activo o no.

2.- Ancho de Banda utilizado por cada interfaz del nodo: tanto en términos absolutos como porcentuales.

3.- Errores de Hardware: como tarjetas de red.

4.- Consumo de memoria y del CPU.

Deberá ser un sistema escalable y amigable, poseer la capacidad de generar reportes estadísticos en cuanto a fallas e interrupciones del servicio y ser capaz de presentar reportes históricos de todas las características mencionadas anteriormente. Finalmente, en cuanto al tipo de decisiones que el sistema pueda tomar, éstas se limitarán al enrutamiento del tráfico en caso de falla o congestión. El diseño se realizará con nodos redundantes para respaldo que permitan la redefinición de las rutas por defecto en caso de falla o congestión.

El sistema debe ser capaz de desarrollar una supervisión especializada para determinar si las acciones tomadas, han alcanzado los resultados esperados.

CAPÍTULO II.

METODOLOGÍA

El presente Trabajo Especial de Grado contempla un estudio tanto teórico como empírico que abarca la utilización de distintas estrategias de recolección y análisis de datos, así como la implementación tecnológica del Protocolo Simple de Gestión de Redes SNMP. Para ello, se empleó una metodología compuesta, basada en cinco estrategias de investigación y desarrollo:

III.1.- Investigación de Mercado:

Es una herramienta que permite conocer, comparar y evaluar las características del mercado en términos de demanda, calidad de los servicios ofrecidos, vida útil de los productos existentes, relación costo-beneficio, debilidades, oportunidades, amenazas y fortalezas propias y de la competencia, entre otras variables relevantes. *(Hernández, 2006)*

En el ámbito tecnológico, la Investigación de Mercado provee información útil para la definición del alcance y análisis de factibilidad de los proyectos de desarrollo, facilitando la estimación de resultados y promoviendo la generación de ideas competitivas.

III.2.- Estudio Comparativo:

Para que la toma de decisiones esté fundamentada en argumentos sólidos, se utiliza el estudio comparativo como estrategia de documentación y soporte de las estrategias y acciones que se tomen en la ejecución de un determinado proyecto.

El estudio comparativo pretende hacer uso de las diferencias y similitudes encontradas en los elementos en proceso de análisis, para facilitar su comprensión y poder emitir juicios asertivos en relación a las ventajas y desventajas que ofrecen cada uno de ellos.

III.3.- Recolección de Datos. Documentación Teórica:

A través de la investigación aplicada y descriptiva, “que consiste en la descripción de algunas características fundamentales de fenómenos o procesos homogéneos”(Sabino, 1986) se realizará la documentación teórica para la obtención de los contenidos y conocimientos que soportan el desarrollo del proyecto.

III.4.- Diseño del Sistema

La utilización de mapas conceptuales, diagramas de clases y descripción de métodos, permite una aproximación lo más exacta posible a las necesidades y requerimientos del proyecto en desarrollo (Ortiz, 2006). El Diseño del Sistema, se basa en la utilización de las herramientas mencionadas anteriormente, entre otras, para generar un modelo inicial a partir del cual se implementará la aplicación.

III.5.- Desarrollo del Prototipo:

Programación Orientada a Eventos y Orientada a Objetos.

La metodología a utilizar para el desarrollo del Prototipo se basa en el Paradigma de la Programación Orientada a Objetos, la cual define la implementación en términos de objetos que son entidades que combinan estado, comportamiento e identidad (Wikipedia, 2006).

Adicionalmente, para la realización de una interfaz gráfica amigable, se empleará la Programación Orientada a Eventos, que tiene como ventajas que proporciona al usuario la posibilidad de construir aplicaciones utilizando interfaces gráficas sobre la base de ocurrencia de eventos.

A través de la convergencia de la programación Orientada a Objetos y la Orientada a Eventos se garantiza el desarrollo de aplicaciones robustas, eficientes, escalables y sobre todo amigables.

CAPÍTULO III.

MARCO TEÓRICO

Hoy en día se hace uso de distintas herramientas para la administración, supervisión y gestión de redes, según los resultados que generan estas herramientas, el administrador puede identificar fallas dentro de la red, prevenir congestión, enrutar el tráfico y hasta prevenir ataques informáticos.

Existen varios protocolos utilizados para la gestión de redes basadas en TCP/IP; los más conocidos son el SNMP (*Simple Network Management Protocol*), descrito en el RFC 1157 y el CMIP (*Common Management Information Protocol*), descrito en el RFC 1189.

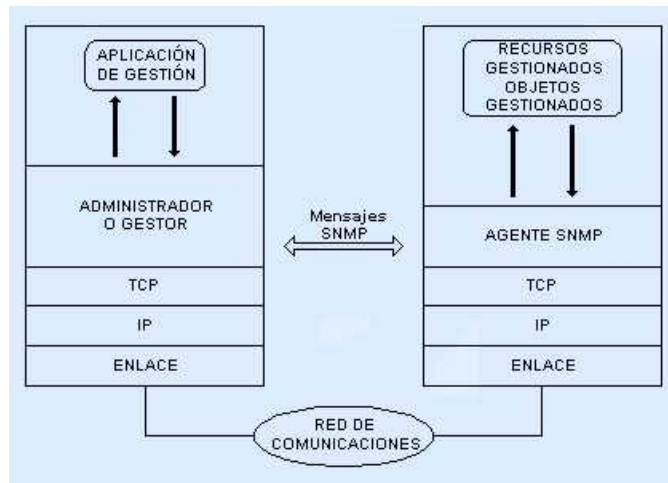
SNMP en sus distintas versiones, es un conjunto de aplicaciones de gestión de red que emplea los servicios ofrecidos por TCP/IP, y que ha llegado a convertirse en un estándar. Surge a raíz del interés mostrado por la IAB (*Internet Activities Board*) en encontrar un protocolo de gestión que fuese válido para la red Internet, dada la necesidad del mismo debido a las grandes dimensiones que estaba tomando.(Huidobro, 2004)

III.1. Arquitectura SNMP

Implícita en el modelo de arquitectura del SNMP existe una colección de estaciones de gestión de red y de elementos de red. Las estaciones de gestión de red ejecutan aplicaciones de gestión que supervisan y controlan los elementos de red. Los elementos de red son dispositivos como hosts, gateways, servidores de terminal, etc., que poseen agentes de gestión para realizar las funciones solicitadas por las estaciones de gestión de red. El SNMP es usado para comunicar información de gestión entre las estaciones de gestión de red y los agentes en los elementos de red. En la figura 1 se muestra un esquema de la arquitectura del

protocolo SNMP (*RFC 1157,1999*). En el ANEXO 1 se encuentra el RFC 1157 que define las especificaciones del protocolo

SNMP es un protocolo del nivel de aplicación que utiliza como protocolo de transporte UDP. Define una relación cliente/servidor entre el gestor de red (que actúa de cliente) y los elementos gestionados (que son los servidores y reciben el nombre de "agentes SNMP").



*Figura 1.- Arquitectura Protocolo SNMP
Fuente: HUIDOBRO, J.*

De esta forma, para el protocolo SNMP la red constituye un conjunto de elementos básicos integrados por los *Administradores o Management Stations* ubicados en los equipos de gestión de red y los *Agentes* (elementos pasivos ubicados en los nodos -host, routers, modems, multiplexores, etc.- a ser gestionados), siendo los segundos los que envían información a los primeros, relativa a los elementos gestionados, por iniciativa propia o al ser interrogados (*polling*) de manera secuencial, apoyándose en los parámetros contenidos en sus MIB (*Management Information Base*).

Agentes y gestor manejan una base de datos de información gestionable, el MIB o *Management Information Base* que contiene, organizados de forma jerárquica, un conjunto de informaciones estadísticas y valores de control que pueden ser

estándar o ser extensiones propias de los fabricantes de los diferentes agentes/gestores (TOBAL, J. 1999)

El contenido del MIB de un agente concreto se define usando la ASN.1 (Abstract Syntax Notation) lo que permite al software gestor incorporar a su base de datos la información de gestión de cualquier nuevo agente.

Manejando este MIB un gestor SNMP puede:

- Modificar las tablas de rutado de un router
- Conocer las estadísticas de funcionamiento de un servidor
- Desconectar una estación de trabajo de la red
- Ver los paquetes que circulan por una subred
- Conocer la temperatura de funcionamiento de un concentrador

A través del MIB se tiene acceso a la información para la gestión, contenida en la memoria interna del dispositivo en cuestión. MIB es una base de datos completa y bien definida, con una estructura en árbol, adecuada para manejar diversos grupos de objetos (información sobre variables/valores que se pueden adoptar), con identificadores exclusivos para cada objeto.

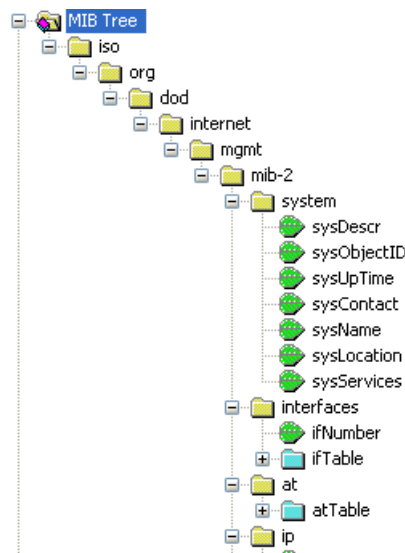


Figura 2.- Extracto de Imagen del Management Information Base (MIB)
Fuente: MG-SOFT LAB

III.2 Propósitos de la Arquitectura de SNMP

El SNMP explícitamente minimiza el número y complejidad de las funciones de gestión realizadas por el propio agente de gestión. Esta meta es atractiva al menos en cuatro aspectos:

1. El coste de desarrollo del software del agente de gestión necesario para soportar el protocolo se reduce acordeamente.
2. El grado de funciones de gestión soportado remotamente se incrementa, posibilitando un uso completo de los recursos de Internet en la tarea de gestión.
3. El grado de funciones de gestión soportado remotamente se incrementa, imponiendo así las mínimas restricciones posibles en la forma y sofisticación de herramientas de gestión.
4. Los conjuntos simplificados de funciones de gestión son fácilmente entendibles y usados por los creadores de herramientas de gestión de red.

(LACUNZA, F. MAGAÑA, E. MARTÍNEZ, A)

Un segundo objetivo del protocolo es que el paradigma funcional para supervisar y controlar sea lo suficientemente flexible como para posibilitar aspectos de gestión y operación de la red adicionales y posiblemente no anticipados.

Un tercer propósito es que la arquitectura sea en lo posible independiente de la arquitectura y mecanismos de hosts o gateways particulares.

III.3. Elementos de la arquitectura

La arquitectura SNMP formula una solución al problema de gestión de redes en términos de:

a. Alcance de la información de gestión

El alcance de la información de gestión transmitida por operaciones del SNMP es exactamente el representado por casos de todos los tipos de objetos no agregados, definidos en el estándar MIB de Internet, o definidos en cualquier otro sitio de acuerdo a las convenciones expuestas en el estándar SMI de Internet.

b. Representación de la información de gestión

La información de gestión se representa según el lenguaje ASN.1, que es especificado para la definición de tipos no agregados en el SMI. El SNMP utiliza un subconjunto bien definido de dicho lenguaje, incluyendo un subconjunto más complejo para la descripción de objetos gestionados y para describir las unidades de datos de protocolo (PDU's) utilizadas para gestionar esos objetos. Asimismo, solo se utiliza un subconjunto de las reglas básicas de codificación del ASN.1, esto es, todas las codificaciones utilizan la forma de longitud definida.

Con el deseo de facilitar una futura transición a protocolos de gestión de redes basados en OSI, se procedió a la definición en el lenguaje ASN.1 de un SMI standard de Internet y de un MIB.

c. Operaciones soportadas por la información de gestión

El SNMP modela las funciones del agente de gestión como lecturas (get) o escrituras (set) de variables. Esta estrategia posee al menos dos consecuencias positivas: Limita el número esencial de funciones de gestión realizadas por el agente de gestión a dos y evita introducir el soporte de comandos de gestión imperativos en la definición del protocolo.

La estrategia se basa en que la monitorización del estado de la red se puede basar a cualquier nivel de detalle en el sondeo (poll) de la información apropiada en la parte de los centros de monitorización. Un número limitado de mensajes no solicitados (traps) guían el objetivo y la secuencia de sondeo.

Las funciones de los pocos comandos imperativos actualmente soportados pueden ser fácilmente implementadas en este modelo de modo asíncrono.

c. Forma y significado de los intercambios

La comunicación de la información de gestión entre entidades de gestión se realiza en el SNMP por medio del intercambio de mensajes de protocolo.

El intercambio de mensajes SNMP sólo requiere un servicio de datagramas poco fiable, y todo mensaje se representa por un único datagrama de transporte.

d. Forma y significado de las referencias a objetos gestionados

El SMI requiere que la definición de un protocolo de gestión contemple:

1.- Resolución de referencias MIB ambiguas

Debido a que el alcance de cualquier operación SNMP está conceptualmente confinado a los objetos relevantes a un único elemento de red, y ya que todas las referencias SMI a objetos MIB son por medio de nombres de variables únicos, no hay posibilidad de que una referencia SNMP a cualquier tipo de objeto definido en el MIB se pueda resolver entre múltiples casos de ese tipo.

2.- Resolución de referencias entre versiones MIB

El objeto referenciado por cualquier operación SNMP es exactamente el especificado como parte de la operación de petición, o en el caso de una operación get-next su sucesor en el conjunto de MIB. En particular, una referencia a un objeto como parte de una versión del MIB estándar de Internet, no se aplica a ningún objeto que no sea parte de dicha versión, excepto en el caso de que la operación sea get-next, y que el nombre del

objeto especificado sea el último lexicográficamente entre los nombres de todos los objetos presentados como parte de dicha versión.

3.- Identificación de los casos de objetos

Cada caso de un tipo de objeto definido en el MIB se identifica en las operaciones SNMP por un nombre único llamado su "nombre de variable". En general, el nombre de una variable SNMP es un identificador de objeto de la forma x.y, donde x es el nombre del tipo de objeto no agregado definido en el MIB, e y es un fragmento de un identificador de objeto que de forma única para dicho tipo de objeto, identifica el caso deseado. Esta estrategia de denominación admite la completa explotación de la semántica de la PDU GetNextRequest, dado que asigna nombres para variables relacionadas de forma que sean contiguas en la ordenación lexicográfica de todas las variables conocidas en el MIB. (LACUNZA, F. MAGAÑA, E. MARTÍNEZ, A)

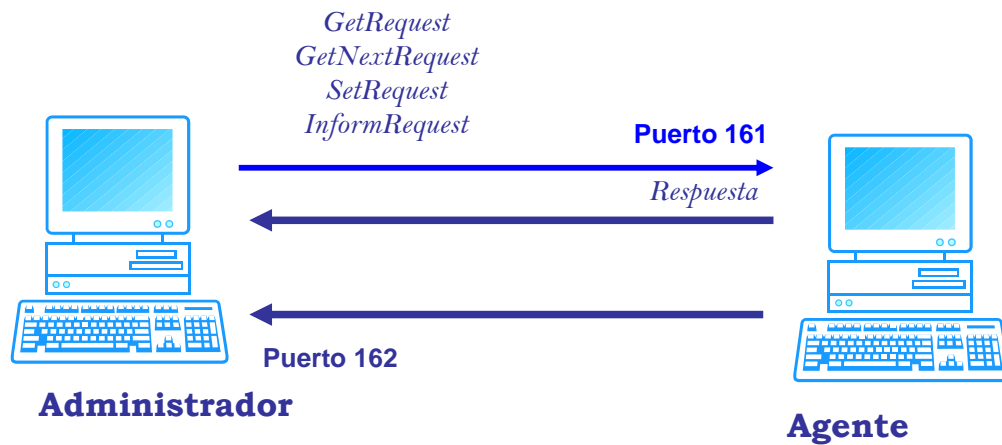
III.4. Especificaciones del Protocolo SNMP

El protocolo de administración de red es un protocolo de aplicación por el que las variables del MIB de un agente son inspeccionadas o alteradas.

Las entidades de protocolo se comunican entre sí mediante mensajes, cada uno formado únicamente por un datagrama UDP. Cada mensaje está formado por un identificador de versión, un nombre de comunidad SNMP y una PDU (Protocol Data Unit - Unidad de datos de protocolo). Estos datagramas no necesitan ser mayores de 484 bytes, pero es recomendable que las implementaciones de este protocolo soporten longitudes mayores.

Todas las implementaciones del SNMP soportan 5 tipos de PDU:

- GetRequest-PDU
- GetNextRequest-PDU
- GetResponse-PDU
- SetRequest-PDU
- Trap-PDU



*Figura 3.- Modelo de funcionamiento del Protocolo SNMP
Fuente: Elaboración Propia*

III.4.1. Elementos de procedimiento

Se describirán a continuación las acciones que realiza una entidad de protocolo en una implementación SNMP. Se define *dirección de transporte* como una dirección IP seguida de un número de puerto UDP (Si se está usando el servicio de transporte UDP).

Cuando una entidad de protocolo envía un mensaje, realiza las siguientes acciones:

1. Construye la PDU apropiada como un objeto definido con el lenguaje ASN.1

2. Pasa esta PDU, junto con un nombre de comunidad y las direcciones de transporte de fuente y destino, a un servicio de autenticación. Este servicio generará en respuesta otro objeto en ASN.1
3. La entidad construye ahora un mensaje en ASN.1 usando el objeto que le ha devuelto el servicio de autenticación y el nombre de comunidad
4. Este nuevo objeto se envía a la entidad destino usando un servicio de transporte. (LACUNZA, F. MAGAÑA, E. MARTÍNEZ, A)

Cuando una entidad de protocolo recibe un mensaje, realiza las siguientes acciones:

1. Hace un pequeño análisis para ver si el datagrama recibido se corresponde con un mensaje en ASN.1. Si no lo reconoce, el datagrama es descartado y la entidad no realiza más acciones.
2. Observa el número de versión. Si no concuerda descarta el datagrama y no realiza más acciones.
3. Pasa los datos de usuario, el nombre de comunidad y las direcciones de transporte de fuente y destino al servicio de autenticación. Si es correcto, este devuelve un objeto ASN.1. Si no lo es, envía una indicación de fallo. Entonces la entidad de protocolo puede generar una trampa (trap), descarta el datagrama y no realiza más acciones.
4. La entidad intenta reconocer la PDU. Si no la reconoce, descarta el datagrama. Si la reconoce, según el nombre de comunidad adopta un perfil y procesa la PDU. Si la PDU exige respuesta, la entidad iniciará la respuesta ahora.

III.4.2 Estructura de una PDU

Los datos que incluye una PDU genérica son los siguientes:

- **RequestID:** Entero que indica el orden de emisión de los datagramas. Este parámetro sirve también para identificar datagramas duplicados en los servicios de datagramas poco fiables.

- **ErrorStatus:** Entero que indica si ha existido un error. Puede tomar los siguientes valores, que se explicarán posteriormente:
 - noError (0)
 - tooBig (1)
 - noSuchName (2)
 - badValue (3)
 - readOnly (4)
 - genErr (5)

- **ErrorIndex:** entero que en caso de error indica qué variable de una lista ha generado ese error.

- **VarBindList:** Lista de nombres de variables con su valor asociado. Algunas PDU quedan definidas sólo con los nombres, pero aún así deben llevar valores asociados. Se recomienda para estos casos la definición de un valor NULL.

III.4.3. GetRequest-PDU y GetNextRequest-PDU

Son PDU's que solicitan a la entidad destino los valores de ciertas variables. En el caso de GetRequest-PDU estas variables son las que se encuentran en la lista VarBindList; en el de GetNextRequest-PDU son aquellas cuyos nombres son sucesores lexicográficos de los nombres de las variables de la lista. Como se

puede observar, `GetNextRequest-PDU` es útil para confeccionar tablas de información sobre un MIB. (LACUNZA, F. MAGAÑA, E. MARTÍNEZ, A)

Siempre tienen a cero los campos `ErrorStatus` y `ErrorIndex`. Son generadas por una entidad de protocolo sólo cuando lo requiere su entidad de aplicación SNMP.

Estas PDU's siempre esperan como respuesta una `GetResponse-PDU`.

III.4.4. SetRequest-PDU

Ordena a la entidad destino poner a cada objeto reflejado en la lista `VarBindList` el valor que tiene asignado en dicha lista. Es idéntica a `GetRequest-PDU`, salvo por el identificador de PDU. Es generada por una entidad de protocolo sólo cuando lo requiere su entidad de aplicación SNMP. Espera siempre como respuesta una `GetResponse-PDU`.

III.4.5. GetResponse-PDU

Es una PDU generada por la entidad de protocolo sólo como respuesta a `GetRequest-PDU`, `GetNextRequest-PDU` o `SetRequest-PDU`. Contiene o bien la información requerida por la entidad destino o bien una indicación de error. Cuando una entidad de protocolo recibe una `GetRequest-PDU`, una `SetRequest-PDU` o una `GetNextRequest-PDU`, sigue las siguientes reglas:

1. Si algún nombre de la lista (o el sucesor lexicográfico de un nombre en el caso de `GetNextRequest-PDU`) no coincide con el nombre de algún objeto en la vista del MIB al que se pueda realizar el tipo de operación requerido ("set" o "get"), la entidad envía al remitente del mensaje una `GetResponse-PDU` idéntica a la recibida, pero con el campo `ErrorStatus` puesto a 2 (`noSuchName`), y con el campo `ErrorIndex` indicando el nombre de objeto en la lista recibida que ha originado el error.

2. De la misma manera actúa si algún objeto de la lista recibida es un tipo agregado (como se define en el SMI), si la PDU recibida era una GetRequest-PDU.
3. Si se ha recibido una SetRequest-PDU y el valor de alguna variable de la lista no es del tipo correcto o está fuera de rango, la entidad envía al remitente una GetResponse-PDU idéntica a la recibida, salvo en que el campo ErrorStatus tendrá el valor 3 (badValue) y el campo ErrorIndex señalará el objeto de la lista que ha generado el error.
4. Si el tamaño de la PDU recibida excede una determinada limitación, la entidad enviará al remitente una GetResponse-PDU idéntica a la recibida, pero con el campo ErrorStatus puesto a 1 (tooBig).
5. Si el valor de algún objeto de la lista no puede ser obtenido (o alterado, según sea el caso) por una razón no contemplada en las reglas anteriores, la entidad envía al remitente una GetResponse-PDU idéntica a la recibida, pero con el campo ErrorStatus puesto a 5 (genErr), y el campo ErrorIndex indicando el objeto de la lista que ha originado el error.

Si no se llega a aplicar alguna de estas reglas, la entidad enviará al remitente una GetResponse-PDU de las siguientes características:

- Si es una respuesta a una GetResponse-PDU, tendrá la lista varBindList recibida, pero asignando a cada nombre de objeto el valor correspondiente.

- Si es una respuesta a una GetNextResponse-PDU, tendrá una lista varBindList con todos los sucesores lexicográficos de los objetos de la lista recibida, que estén en la vista del MIB relevante y que sean susceptibles de ser objeto de la operación "get". Junto con cada nombre, aparecerá su correspondiente valor.
- Si es una respuesta a una SetResponse-PDU, será idéntica a esta, pero antes la entidad asignará a cada variable mencionada en la lista varBindList su correspondiente valor. Esta asignación se considera simultánea para todas las variables de la lista.
- En cualquiera de estos casos El valor del campo ErrorStatus es 0 (noError), igual que el de ErrorIndex. El valor del campo requestID es el mismo que el de la PDU recibida.

III.4.6. Trap-PDU

Es una PDU que indica una excepción o trampa. Es generada por una entidad de protocolo sólo a petición de una entidad de aplicación SNMP. Cuando una entidad de protocolo recibe una Trap-PDU, presenta sus contenidos a su entidad de aplicación SNMP. (FOROUZAN, 2002)

Los datos que incluye una Trap-PDU son los siguientes:

- **enterprise**: tipo de objeto que ha generado la trampa.
- **agent-addr**: dirección del objeto que ha generado la trampa.
- **generic-trap**: entero que indica el tipo de trampa. Puede tomar los siguientes valores:
 - coldStart (0)
 - warmStart (1)
 - linkDown (2)

- linkUp (3)
 - authenticationFailure (4)
 - egpNeighborLoss (5)
 - enterpriseSpecific (6)
-
- **specific-trap**: entero con un código específico.

 - **time-stamp**: tiempo desde la última inicialización de la entidad de red y la generación de la trampa.

 - **variable-bindings**: lista tipo varBindList con información de posible interés.

Dependiendo del valor que tenga el campo generic-trap, se iniciarán unas u otras acciones:

- **Trampa de arranque frío** (coldStart): La entidad de protocolo remitente se está reiniciando de forma que la configuración del agente o la implementación de la entidad de protocolo puede ser alterada.

- **Trampa de arranque caliente** (warmStart): La entidad de protocolo remitente se está reiniciando de forma que ni la configuración del agente ni la implementación de la entidad de protocolo se altera.

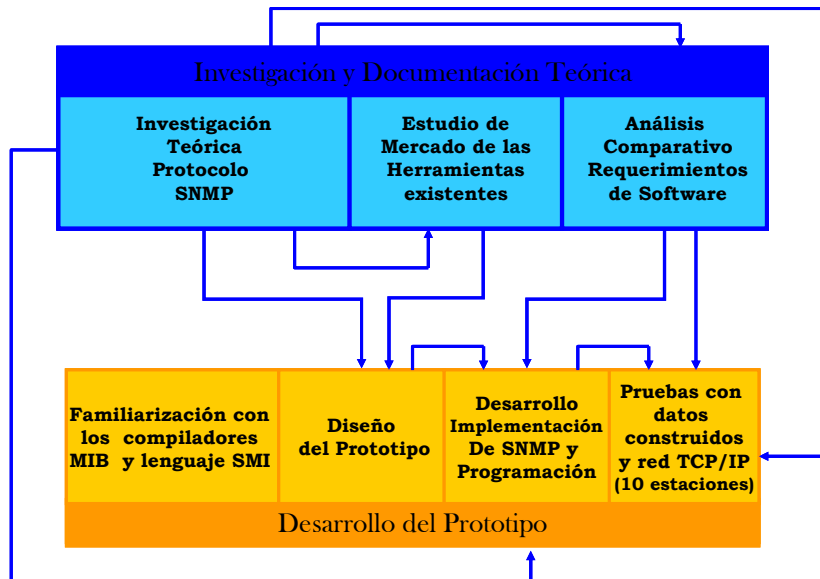
- **Trampa de conexión perdida** (linkDown): La entidad de protocolo remitente reconoce un fallo en uno de los enlaces de comunicación representados en la configuración del agente. Esta Trap-PDU contiene como primer elemento de la lista variable-bindings el nombre y valor del interfaz afectado.

- **Trampa de conexión establecida** (linkUp): La entidad de protocolo remitente reconoce que uno de los enlaces de comunicación de la configuración del agente se ha establecido. El primer elemento de la lista variable-bindings es el nombre y el valor del interfaz afectado.
- **Trampa de fallo de autenticación** (authenticationFailure): La entidad de protocolo remitente es la destinataria de un mensaje de protocolo que no ha sido autenticado.
- **Trampa de pérdida de vecino EGP** (egpNeighborLoss): Un vecino EGP con el que la entidad de protocolo remitente estaba emparejado ha sido seleccionado y ya no tiene dicha relación. El primer elemento de la lista variable-bindings es el nombre y el valor de la dirección del vecino afectado.
- **Trampa específica** (enterpriseSpecific): La entidad remitente reconoce que ha ocurrido algún evento específico. El campo specific-trap identifica qué trampa en particular se ha generado.

CAPÍTULO IV. DESARROLLO

El presente proyecto se dividió en dos (2) macro etapas que engloban todas las actividades que condujeron al desarrollo del Sistema de Control, Supervisión y Gestión de Redes basadas en el protocolo IPv4. Dichas etapas se estructuraron en tareas específicas, cada una de ellas asociada a los objetivos planteados.

En la *figura 4* se presenta un modelo relacional de las etapas de desarrollo del Sistema.



*Figura 4.- Modelo Relacional Etapas de Desarrollo TEG
Fuente: Elaboración Propia*

Como se observa en el Modelo Relacional, existe una real interdependencia entre los procedimientos y tareas específicas utilizadas para el desarrollo del Sistema de Control y Supervisión de Redes.

A continuación se describen cada una de las actividades llevadas a cabo durante la ejecución del proyecto, así como el procedimiento detallado para la implementación del prototipo.

IV.1 Documentación Teórica Protocolo Simple de Gestión de Redes SNMP.

La documentación teórico-práctica del protocolo que sustenta el Sistema desarrollado (SNMP), sirvió como base para todas las actividades sucesivas que formaron parte del proyecto. De hecho, la investigación del protocolo fue un proceso considerablemente dinámico, ya que el diseño e implementación del sistema exigió de actualizaciones continuas y de un dominio absoluto de la arquitectura y funcionamiento de SNMP.

Como se describió en el Capítulo II, SNMP definido en el RFC 1157- es un protocolo sencillo de comprender, pues su funcionamiento se basa en apenas 6 mensajes: *Get-Request*, *Set-Request*, *GetNext-Request*, *GetBulk*, *GetInform* y *Traps*, pero muy complejo de implementar por la gran cantidad de variables que pueden ser gestionadas a través de él. El MIB (*Management Information Base*) de una estación que preste servicios de aplicación únicamente (como un computador personal, IP Host), puede contener más de mil identificadores de objetos (OID's) ubicados en una estructura jerárquica en forma de árbol cuya interpretación requiere de un estudio profundo y detallado, además de demandar el dominio del lenguaje ASN 1, que por convención internacional, describe todos los componentes de gestión y administración de redes.

Las razones expuestas anteriormente, sustentan la importancia que reviste el dominio exhaustivo de SNMP para la implementación del sistema de gestión, así como la familiarización con el manejo de los MIB que contienen todas las variables gestionables por este protocolo. Posteriormente se describirán las

actividades y herramientas utilizadas para alcanzar el dominio de estas bases de datos.

Para la implementación del módulo de control del sistema, es decir, aquel que permite tomar decisiones en cuanto al enrutamiento del tráfico de la red, se había pensado utilizar algún protocolo de acceso remoto como Telnet o SSH, que facilitara la modificación instantánea de las tablas de ruta o de las interfaces de salida de las estaciones gestionadas. Sin embargo, después de conocer con profundidad el protocolo SNMP, se ajustó la planificación inicial, ya que éste es capaz no sólo de obtener el estado de todos los procesos activos de un determinado nodo, sino también permite escribir y modificar el valor de cualquiera de estas variables o procesos. De esta forma, se excluyó la investigación teórica de esos protocolos de acceso remoto y se consolidó la utilización de SNMP tanto para funciones de supervisión como de control.

IV.2 Investigación de las Herramientas existentes en el Mercado

Realizar el estudio y el desarrollo de un sistema de gestión de redes es una tarea ambiciosa si se piensa en la gran cantidad de productos similares que existen en el mercado y en la importancia que ha ido adquiriendo la supervisión oportuna y eficiente de las estaciones que la conforman. Por lo tanto, antes de iniciar el diseño y determinar las características específicas del sistema a desarrollar, se consideró indispensable realizar un análisis de las herramientas de supervisión que se ofrecen en el mercado con el fin de presentar un prototipo competitivo y capaz de introducir elementos innovadores en los futuros desarrollos.

Como estrategia para organizar la investigación y obtener una información lo más completa y confiable posible, se utilizó la Internet como herramienta de búsqueda y se realizó una lista de todos los sistemas de gestión y supervisión de redes basadas en el protocolo IPv4, adicionalmente, se agregaron sus fuentes

electrónicas para su fácil estudio. En el APÉNDICE I se presenta la tabla con los 29 sistemas homólogos encontrados.

Una vez finalizada la búsqueda, se seleccionaron 5 de los sistemas que se consideraron más completos, evaluando para esa selección no sólo el nivel de reportes que proporcionaba cada uno, ni la robustez del desarrollo, sino también parámetros de estimación de mercado como los comentarios emitidos por los usuarios de estos sistemas en los foros de discusión.

Los 5 sistemas que se estudiaron con detalle fueron los siguientes:

- JFFNMS (*Just For Fun Network Management System*). Desarrollado por Cisco System.
- OPENNMS (*Open Network Management System*). Basado en código abierto (Open source), implementado por los mismos creadores de Wikipedia.
- Fidelia Netvigil: Desarrollado por Network General.
- IP-MONITOR. Desarrollado por IPMonitor Corporation. Sistema de Gestión con certificado Microsoft.
- ORION NPM. (Network Performance Monitor). Desarrollado por Solarwinds empresa líder en sistemas de Gestión y Supervisión de Redes IP.

Con las características generales de cada uno de los sistemas en estudio, se elaboró un cuadro comparativo-descriptivo que se presenta en el capítulo V.

IV.3 Análisis de los Requerimientos de Software.

Conocidas las características mínimas que se requieren para generar un producto competitivo, se procedió a realizar el análisis comparativo de la plataforma tecnológica sobre la cual se debía realizar el diseño y la implementación del sistema.

Se inició el análisis evaluando los dos Sistemas Operativos por excelencia, el soportado por Microsoft: Windows XP, y el que se basa en la tendencia del software libre, LINUX: en cualquiera de sus versiones.

Para soportar la documentación, se realizó una investigación detallada de ambos sistemas y se realizó la comparación en ciertos elementos específicos, seleccionados previamente, que se consideraron trascendentales para la toma de decisiones:

- *Instalación:* se tomó en consideración la complejidad de configurar y ejecutar su programa de inicialización. A través de esta variable se puede estimar la aceptación de los usuarios del Sistema Operativo y la dimensión de uso que pueda tener por su sencillez o dificultad, al momento de instalación.
- *Compatibilidad:* se evaluó la compatibilidad de ambos Sistemas con los dispositivos de hardware existentes en el mercado. El Sistema de Gestión y Supervisión de redes, no se limita a estaciones que ofrecen servicios de aplicación únicamente (como ordenadores personales) sino a cualquier dispositivo que se comunique a través del protocolo IPv4 y que tenga un servicio activo SNMP, por esta razón la compatibilidad con el Sistema Operativo es una característica fundamental

- *Interfaz gráfica:* la facilidad para desarrollar aplicaciones amigables resultó también premisa relevante en el análisis comparativo. Los Sistemas de Gestión de Redes que pululan en el mercado son considerablemente atractivos al usuario, utilizan gráficos para mostrar los reportes del comportamiento de las estaciones, tienen ventanas dinámicas de datos, entre otras características, por lo tanto, la plataforma sobre la que se decidiera realizar el diseño debía ofrecer facilidades de programación visual.
- *Administración de Redes:* la forma de gestionar el tráfico IP, el acceso al ancho de banda de las interfaces de entrada y salida, los *firewalls* que ofrece cada plataforma, son elementos determinantes para la programación e implementación del protocolo SNMP, por esta razón se consideró oportuno evaluar con detalle el alcance que posee cada Sistema Operativo en la administración de las redes.
- *Estabilidad:* dependiendo de la composición del Kernel y de la comunicación directa que puedan tener algunos procesos con el núcleo del mismo, los Sistemas Operativos se hacen más o menos estables. Para efectos de la implementación del prototipo, la estabilidad no representó una premisa definitiva para la toma de decisiones, pues ambas plataformas, en sus distintas versiones, ofrecen servicios robustos y considerablemente estables, al menos para las exigencias de capacidad de procesamiento que exige este desarrollo.
- *Facilidad:* si bien la facilidad de uso no es un elemento cuantificable pues depende de la afinidad que desarrolle cada usuario con el sistema operativo, se tomó muy en cuenta la simplicidad de algunas herramientas (como el servicio SNMP de Windows) que ofrece cada plataforma, eso garantiza que el uso de los recursos fuese eficiente en el tiempo estimado para el desarrollo.

Tomando en cuenta las variables descritas anteriormente, se elaboró un cuadro resumen, donde se sintetizan los aspectos más importantes, producto de la investigación documental y referencial realizada durante el estudio comparativo.

Resultó bastante complejo tomar una decisión de cuál plataforma tecnológica utilizar para el diseño e implementación del Sistema. En primer lugar, Windows es un Sistema Operativo amigable, fácil de instalar y resulta mucho más sencillo desarrollar interfaces gráficas de calidad bajo este entorno. Sin embargo, en términos generales la tendencia del Software Libre es cada vez mayor y Linux está teniendo gran auge debido a que resulta un poco más estable y es más potente para la administración de redes. Basado en la realización de esta investigación y bajo la premisa de la familiarización que existe con el entorno Microsoft, la facilidad de desarrollar ambientes gráficos amigables y la compatibilidad con los diversos fabricantes de hardware, se decidió realizar el desarrollo haciendo uso del Sistema Operativo Windows XP.

Definido el Sistema Operativo sobre el cual soportar el desarrollo, se procedió al análisis del lenguaje de programación a utilizar para el diseño y posterior implementación del mismo.

Se seleccionaron 3 lenguajes de programación estándares, Visual Basic, JAVA y C++ para su evaluación y comparación, y de la misma manera como se realizó con el Sistema Operativo, se seleccionaron 3 variables que de forma objetiva, sustentaran una evaluación enmarcada en las necesidades propias del sistema en desarrollo y no, en los paradigmas y argumentaciones tradicionales de los programadores.

Los elementos que se consideraron indispensables para el estudio fueron los siguientes:

- *Portabilidad:* la selección del lenguaje de programación tenía que ser cónsona con el Sistema Operativo a utilizar y aunque las tres posibles alternativas (Visual Basic, JAVA y C++), facilitan el desarrollo en ambas plataformas, considerar que el sistema pueda gestionar estaciones

con características de software diferentes, significa un valor agregado importante. La portabilidad se pensó, respondiendo al objetivo de que la supervisión y control se realizará sobre cualquier dispositivo que se comunique en lenguaje IPv4 y que tenga activo un servicio SNMP, independientemente del Sistema Operativo sobre el cual se soporte el agente en gestión.

- *Interfaz Gráfica:* Similar a la evaluación de los requerimientos de software utilizados para la selección del Sistema Operativo, la premisa de que las herramientas de gestión de redes que existen en el mercado sean tan amigables y con recursos visuales tan atractivos, exigían que la implementación permitiese desarrollar un prototipo competitivo no solo a nivel de funcionalidad, sino también en cuanto a interfaz gráfica se refiere. En este aspecto en específico los ambientes de desarrollo son muy variables y la complejidad en el acceso a la API de Windows varía considerablemente en cada uno de ellos.
- *Gestión de Redes:* Siendo el objetivo primordial del sistema en desarrollo la supervisión y control de las redes, los lenguajes en estudio debían ser capaces de soportar y proveer un módulo, DLL o librería de comunicación IP, y de ofrecer estos servicios todos los ambientes de desarrollo, verificar cual de éstos se adaptaba más a las características del protocolo a implementar.

Realizado el análisis comparativo-descriptivo de las características mencionadas anteriormente en cada uno de los lenguajes de programación y considerando la inminente necesidad de realizar un desarrollo robusto, amigable y competitivo, se decidió inicialmente utilizar el lenguaje C++. Si bien es uno de los lenguajes más complicados de implementar, es también uno de los más poderosos y portables a nivel mundial, el que presenta mayor cantidad de librerías, compila en cualquier sistema operativo y existe mucha documentación. Visual Basic es un lenguaje

sencillo, no es tan portable, pues no tiene la capacidad de ejecutarse a menos que se instalen todas las librerías utilizadas en cada una de las máquinas donde se quiera utilizar la aplicación y solo responde al entorno Microsoft, pero presenta grandes ventajas por los módulos SNMP y las librerías que soportan este protocolo. JAVA, por su parte, también es un lenguaje poderoso y portable, sin embargo, no se encontraron fuentes que exaltaran su efectividad en la administración de redes.

Como se describió anteriormente, C++ se destaca en dos de las tres características que fueron objeto de estudio: es portable y además facilita la supervisión de redes a través de la implementación de sockets. Se buscó entonces un compilador (ambiente de programación) que facilitara el diseño de elementos gráficos atractivos y cuya implementación no requiriera procedimientos complejos. Se evaluaron las distintas alternativas: DEV-C++, VISUAL C++ y BUILDER C++, siendo el último el que más se aproximaba a los requerimientos buscados. En el APÉNDICE II se presenta el resultado de la investigación comparativa de los tres entornos.

Durante las dos fases posteriores a la evaluación de los requerimientos de Software, es decir, en el diseño y posterior implementación del sistema, se presentaron situaciones que motivaron a la reconsideración del lenguaje y el ambiente de programación seleccionado.

En la fase de diseño, se ratificó la idea de que el objetivo principal del desarrollo del sistema no era programar el protocolo SNMP al más bajo nivel, sino por el contrario, implementar el mismo, haciendo uso de las librerías que soportan su funcionamiento. BUILDER C++ contiene una librería nativa (es decir, no es importada) denominada *winsnmp.h* que no era compatible con la mayoría de los sistemas operativos actuales que operan bajo el entorno Microsoft. Al compilarla, surgía un error de enlace (Link error) que sólo se solucionó al correr el ambiente de desarrollo en un Sistema Operativo poco vigente (Windows 98).

Existían cuatro posibles alternativas:

- 1.-Mantener el lenguaje de programación y trabajar con cualquiera de los otros dos ambientes de desarrollo. (DEV o VISUAL C++).
- 2.-Utilizar Builder C++ y desarrollar el protocolo SNMP al más bajo nivel, interpretando el lenguaje ASN 1.
- 3.-Utilizar Builder C++ para la programación del prototipo, pero realizando llamadas al sistema de otro compilador (por ejemplo Visual C++) cuyas librerías *snmp* si son compatibles.
- 4.-Evaluar nuevamente las alternativas existentes y, con una mayor argumentación de las necesidades de software exigidas por el diseño del sistema, cambiar el lenguaje de programación.

Todas las alternativas se evaluaron e inclusive se pusieron en práctica hasta encontrar alguna argumentación que motivara a su descarte. Lo primero que se realizó fue proseguir el desarrollo en C++, en el ambiente VISUAL C++. La librería *snmp.h* contenía las 7 funciones básicas de comunicación a través de sockets (puertos 161 y 162), sin embargo desarrollar aplicaciones gráficas bajo este entorno resulta sumamente complicado pues se programa directamente a través de la API de Windows y resultó mayor la exigencia de la programación de la interfaz, que la del mismo sistema, por tanto, se descartó la alternativa.

Después se retomó la idea de usar Builder C++ y realizar la programación al más bajo nivel del protocolo SNMP; sin embargo sólo el desarrollo del protocolo requería de meses de estudio y aplicación y no tenía sentido alguno, cuando la mayoría de los lenguajes de programación ofrecen módulos completos para la implementación de SNMP. (De hecho, Builder C++ también lo ofrece, pero no era compatible con los Sistemas Operativos actuales).

La tercera alternativa parecía la solución más aproximada al problema, sin embargo, se consideró que realizar llamadas al sistema (*system-call*) cada vez que se requería aplicar alguna función SNMP, iba a hacer del prototipo una aplicación demasiado ineficiente con un consumo excesivo de procesamiento.

El último planteamiento resultó el más conveniente. Se evaluaron nuevamente los lenguajes de programación y se seleccionó Visual Basic 6.0 como la mejor opción para la implementación del sistema.

Ante los paradigmas existentes de la poca fortaleza del lenguaje seleccionado, se realizó una investigación documental y experimental que permitiera blindar cualquier tipo de argumentación contra el nuevo ambiente de desarrollo. Tomando en consideración las premisas utilizadas para la primera evaluación de los lenguajes (Portabilidad, Interfaz Gráfica y Gestión de Redes), sólo en uno de los aspectos Visual Basic 6.0 presenta una debilidad considerablemente importante: sólo trabaja bajo entorno Microsoft.. Sin embargo, dicha debilidad se atenúa ante las herramientas que ofrece el lenguaje para la programación gráfica y el paradigma de Programación Orientada a Objetos y Eventos.

Posteriormente se profundizó en las posibles consecuencias negativas de realizar el desarrollo en un lenguaje no compatible con entornos LINUX, y se concluyó que no representa ninguna limitación que el Sistema de Gestión corra exclusivamente en ambiente Microsoft, ya que el protocolo SNMP es tan poderoso que puede supervisar y controlar cualquier dispositivo de red basado en IPv4 y que tenga un servicio SNMP activo (independientemente del Sistema Operativo), de tal forma que la única limitación se reduce a que el administrador de la red deberá cargar la aplicación sobre Windows. Las estaciones gestionables (switches, host, router, enlaces punto a punto, bridge, entre otros) pueden correr bajo cualquier entorno.

Finalmente, se encontró un módulo comercial para Visual Basic 6.0 con el desarrollo de todas las funciones SNMP, denominado *PowerSNMP*.

Dicho módulo, además de excelente documentación teórica de los RFC 1213- y 1157-, que describen el funcionamiento de SNMP y que define los valores de los MIB, provee las principales funciones del protocolo a través de un .DLL que se agrega como componente al proyecto de Visual Basic 6.0.

El componente permite la utilización de un objeto denominado *Manager*, que posee los siguientes atributos y métodos:

- *Manager.Open*: abre la sesión SNMP (abre el puerto 161 o 162, dependiendo del proceso, y establece un socket con los agentes en supervisión o una vez que ocurra un evento).
- *Manager.Close*: cierra la sesión SNMP.
- *Manager.Message.Variable*: atributo que recibe el identificador de objeto (OID) de la variable en proceso de gestión. La inicialización de este atributo se realiza pasando como parámetro el OID en notación punto decimal, es decir, directamente como se lee en el MIB (e.j. 1.1.2.3.4.25.1.1). El RFC-1213 que define las características de todas las variables gestionables, contiene la descripción de 1003 variables que pueden ser interpretadas por el protocolo SNMP, es decir el árbol que representa la base de datos de los elementos de una estación contiene 1003 hijos. El dominio de este atributo es de vital importancia para la implementación del protocolo, ya que dependiendo del reporte que se desee generar, se pasa el valor del OID correspondiente.
- *Manager.AgentName*: atributo que se refiere a la dirección IP en notación decimal de la estación destino.

- *Manager.Type*: como se describió en el capítulo II, existen 7 tipos de mensajes SNMP: *GetRequest*, *SetRequest*, *GetNextRequest*, *GetBulk*, *GetInform*, *GetResponse* y las capturas conocidas como *Traps*. La inicialización de este atributo se realiza pasando como parámetro el nombre del mensaje, precedido por la palabra *snmp* y añadiendo al final del mismo el número 1 o 2. (e.j. *snmpGetNext1*). El número que se agrega representa la versión de SNMP que se utiliza. El sistema desarrollado soporta tanto la versión 1 como la 2.
- *Manager.Send*. Una vez inicializado el objeto con los atributos descritos anteriormente, se llama al método *enviar* cuya función es compilar el PDU del mensaje SNMP y por el socket que se encuentra abierto enviarlo al agente destino. Cuando la estación responde se invoca automáticamente un método denominado *Manager_Response()*.

A través del uso del objeto *Manager* que ofrece el módulo *PowerSNMP* de Visual Basic 6.0 y estableciendo estrategias para la familiarización con el contenido del MIB de cada estación, se garantizaron los requerimientos de software necesarios para el desarrollo del prototipo del Sistema de Control, Supervisión y Gestión de Redes basados en el protocolo IPv4.

IV.4 Familiarización con los Compiladores MIB y Lenguaje ASN.1

Tanto en el Capítulo II como en la descripción del desarrollo del proyecto se ha mencionado frecuentemente la importancia del dominio de la información contenida en el *Management Information Base* (MIB) para la implementación del Sistema. La razón por la cual se ha redundado tanto en este tema es porque todas las variables que solicita el protocolo SNMP se encuentran contenidas en dicha

base datos que organiza sus valores según un árbol jerárquico cuyas “ramas” están caracterizadas en distintas Macro-Divisiones de los parámetros de supervisión, en otras palabras, una “rama” contiene todos los valores relacionados con las características generales del sistema (e.j. nombre, ubicación, contacto, descripción); otra contiene los valores del estado de las interfaces de entrada y salida (e.j. ancho de banda, errores transmitidos, fabricante, etc.). y así con más de 1000 valores que se pueden obtener de forma remota a través del protocolo SNMP.

Evidentemente, la supervisión de estaciones no necesariamente implica el procesamiento de todos los valores contenidos en el MIB, sin embargo, es indispensable conocer el significado de cada uno de ellos y la ubicación espacial en el árbol jerárquico en el cual se encuentran organizados, de lo contrario, sería imposible solicitar el valor de un campo determinado.

Todos los dispositivos de red basados en IPv4, por convención tienen uno o más archivos *.mib* en las carpetas del sistema. Sin embargo, la dificultad se presenta en cómo interpretar la información contenida en esos archivos. Procurar abrirlos a través de un editor de texto resultaría un fracaso, pues están codificados en lenguaje ASN.1 imposibles de interpretar sin una herramienta especializada.

Debido a la inminente necesidad de conocer a profundidad la información contenida en el MIB, su estructura y significado, se utilizó un compilador denominado MIB Compiler, desarrollado por MG-SOFT LAB, empresa norteamericana especializada en sistemas de supervisión de redes.

Existen muchos software que se utilizan para este fin, ninguno de los cuales está escrito en español. MIB Compiler se empleó como herramienta de consulta durante todo el desarrollo del proyecto.

El compilador se basa en una especie de traductor que decodifica cualquier archivo *.mib* y despliega de forma detallada el significado de cada OID, su tipo de acceso (lectura o escritura) entre otros. En la *figura 5* se muestra una imagen del software y la forma en que presenta la información al usuario.

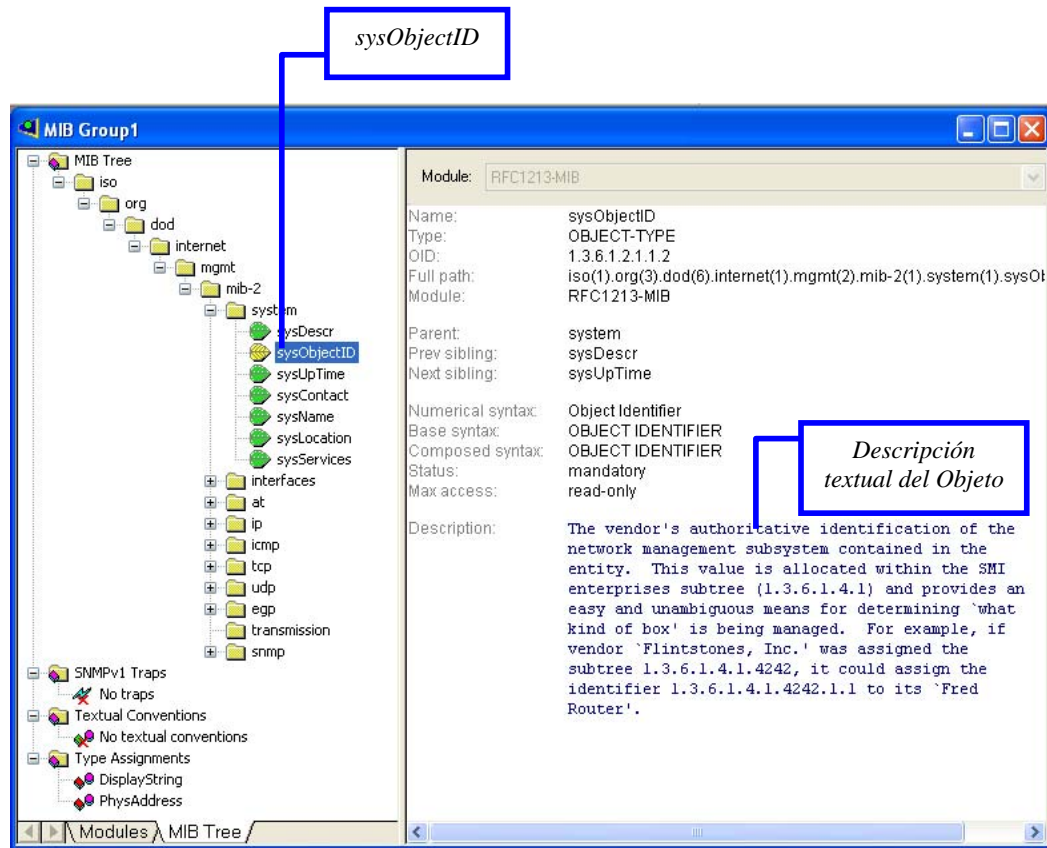


Figura 5. Ejemplo de uso del MIB-COMPILER. Variable mostrada sysObjectID.
Fuente: MG-SOFT MIB COMPILER

En el ejemplo representado en la *Figura 5* se muestra la descripción del objeto *sysObjectID*, que pertenece al grupo de valores del sistema. A la derecha de la pantalla se desplegó la descripción textual y las características, tales como la ruta completa de acceso (*full path*), escrita tanto en notación punto decimal como en cadena de caracteres.

Para el desarrollo del sistema se visualizaron casi la totalidad de las variables contenidas en el MIB, tomando las que se consideraron necesarias para generar

los reportes, que surgen como producto de los procesos de supervisión que ofrece el sistema desarrollado y que serán descritos en el Capítulo V.

IV.5 Diseño del Sistema.

Para el diseño del Sistema de Control, Supervisión y Gestión de Redes basadas en el protocolo IPv4 se aplicó el concepto de la Teoría General de Sistemas, el cual define a éstos como conjuntos de elementos que guardan estrechas relaciones entre sí, que lo mantienen directa o indirectamente unidos de modo más o menos estable y cuyo comportamiento global persigue, normalmente, algún tipo de objetivo (*Arnold, 1998*).

Para completar la definición de la metodología utilizada, se trabajó con la concepción de Sistemas Abiertos, en donde queda establecida como condición para la continuidad sistémica el establecimiento de un flujo de relaciones con el entorno.

Basado en el paradigma de la Programación Orientada a Objetos y explotando las fortalezas del ambiente de desarrollo de facilitar la Programación Orientada a Eventos, se realizó el diseño del Sistema dividiéndolo en 4 módulos funcionales que interactúan entre sí. La clasificación se realizó sobre la base del alcance del proyecto, considerando todos los requerimientos establecidos en la planificación inicial, e incluidas algunas novedades que surgieron como parte del aporte innovador y diferencial que se le dio al prototipo durante su desarrollo.

Los 4 módulos funcionales son los siguientes:

- Sesión y Configuración
- Supervisión
- Control
- Reportes

Los módulos funcionales son interdependientes e interactúan con una lista de objetos “Agentes” que presentan todos los atributos de las estaciones que se encuentran en supervisión y los métodos que hacen posible la comunicación Administrador-Estación.

IV.5.1 Módulo de Sesión y Configuración.

El diseño del sistema se soporta en los parámetros de inicialización que se definen en el presente módulo, el cual incluye todos los procesos de activación tanto de software como de hardware de los componentes que interactúan en la aplicación.

En el módulo de Sesión y Configuración, se realiza la apertura de los puertos de envío y recepción de mensajes SNMP, a través de los cuales se lleva a cabo el intercambio de información entre el administrador y los agentes en gestión.

Adicionalmente, tiene un atributo troncal representado por la *Lista de Objetos Agentes* que se inicializa en la carga del sistema y se actualiza periódicamente dependiendo de las estaciones que se vayan integrando a la topología de la red administrada.

Es en esta fase del sistema donde se configuran los parámetros de gestión y control de la aplicación, indicados por el modo de supervisión, dimensiones de las escalas temporales de los reportes gráficos, dirección de correo electrónico del administrador de la red, entre otros.

El Sistema desarrollado funciona bajo la topología cliente-servidor. El software de administración de la red envía mensajes de solicitud de información y cada estación remota tiene un agente SNMP corriendo que se comporta como servidor. En este módulo se realiza el primer

intercambio de mensajes de solicitud y se cargan los valores iniciales de cada agente.

Todos los valores inicializados en el módulo de sesión y configuración servirán como parámetros para los métodos implementados en los demás módulos que forman parte del sistema.

IV.5.2 Módulo Supervisión

Los métodos y procedimientos que forman parte del presente módulo representan la columna vertebral de todo el sistema. Una vez configurados los parámetros generales en el módulo anterior y cargada la lista de agentes que se encuentran activos, el sistema inicia un proceso de encuesta periódica (*polling*) a través de mensajes *GetRequest* y *GetNextRequest*. Cada agente activo tiene un tiempo global asignado para solicitar todos los valores y cada mensaje de solicitud también tiene asignado un tiempo determinado. En otras palabras, el objeto *Agente* tiene un atributo que define cuándo le corresponde realizar su encuesta (como una globalidad) y otro que determina cuándo solicitar cada dato específico.

Esa distribución temporal tan compleja se debe a que por el puerto SNMP 161 se puede establecer una sola comunicación simultánea, por tanto el esquema de encuestas periódicas a través de las cuales se van a generar los reportes gráficos, debe garantizar una asignación de tiempos para cada estación que se encuentre activa.

Adicionalmente, en el módulo de supervisión se almacena el registro de los mensajes tipo TRAPS, conocidos como capturas. Dichas alertas son generadas por las estaciones cuando ocurre algún evento inesperado y son escuchadas por el objeto *Manager* a través del puerto 162. Existe una correlación muy importante entre este módulo y el de Reportes ya que cada que vez que se recibe un mensaje TRAPS, se invocan funciones específicas del módulo mencionado que serán detalladas posteriormente.

Por último, el módulo de supervisión incluye un elemento innovador que se agregó al diseño después de haber iniciado y que no estaba planificado en el alcance del proyecto, que se basa en la implementación de las funciones necesarias para el desarrollo de un *Buscador MIB*, esto es, una herramienta con características similares al MIB-COMPILER, utilizado como soporte para la familiarización de los valores gestionables por los mensajes SNMP.

El objetivo del desarrollo de esta herramienta es complementar el proceso de supervisión realizado por los administradores de la red facilitando el acceso a todos los elementos contenidos en el MIB de las estaciones activas.

IV.5.3 Módulo de Control

Una de las características que hacen del Sistema en desarrollo una herramienta novedosa frente a las alternativas que ofrece el mercado actual, es que además de supervisar y gestionar las estaciones activas en una red, es capaz de modificar las tablas de ruta de los nodos en caso de existir alguna falla, caída del sistema o congestión. Para ello, el módulo de control utiliza el protocolo SNMP para modificar los valores de las rutas por defecto que tiene cada estación.

Como se describió en el apartado IV.2 del presente capítulo, inicialmente se había pensado que el desarrollo del módulo de control se realizaría a través de la implementación de algún protocolo de acceso remoto como Secure SHell (SSH). Sin embargo, la función *SetRequest* del protocolo SNMP permite modificar el valor de cualquier variable contenida en el MIB de una estación, siempre y cuando ésta sea de lectura y escritura. De esta forma, dependiendo de los eventos que hayan generado la alarma, el sistema de forma automática está en capacidad de modificar el valor de las rutas por defecto de cada nodo en supervisión.

Sin embargo, el módulo de control no se limitó exclusivamente a la parte de enrutamiento dinámico, sino que se incluyó la posibilidad de que el administrador de la red modifique los valores de cualquier otro parámetro contenido en el MIB de la estación y que sea lectura y escritura tales como, el tamaño máximo de los datagramas que pueden ser recibidos y transmitidos por una interfaz determinada, el ancho de banda máximo de una estación, el nombre del sistema, la ubicación, entre muchos otros parámetros.

Tanto el módulo de supervisión como el de control, tienen como función compartida la gestión de los TRAPS recibidos. Como se describió anteriormente, durante el proceso de supervisión se registran todas las alertas que llegan al sistema en un archivo de texto, mientras que en el proceso de control se evalúan esas alertas y el sistema, si se encuentra en modo de supervisión automático toma las decisiones que posiblemente solventen de manera temporal la causa que generó dicha alarma. En caso de que el modo de gestión sea manual, el administrador de la red tomará las decisiones que considere conveniente, modificando el valor del parámetro adecuado.

IV.5.4 Módulo Reportes.

Para poder ofrecer una información amigable al administrador de la red se incluyó dentro del diseño un módulo de reportes que presenta los valores obtenidos en el proceso de supervisión a través de diversos formatos como tablas, gráficas, listas, entre otros.

El módulo de reportes es la interfaz directa con el usuario, sin embargo es producto del procesamiento de datos y mensajes entre los tres módulos anteriores.

Incluye también las funciones especializadas que permiten que el administrador de la red reciba correos electrónicos con el reporte de las alertas detectadas por el sistema en tiempo real.

Además de los 4 módulos funcionales, el diseño del sistema incluye la iteración con el *Objeto Agente*, mencionado anteriormente en reiteradas ocasiones, que es el elemento que ejecuta todas las funciones descritas en los módulos funcionales, una vez recibidas las instrucciones de éstos. Dichas funciones las implementa a través de dos recursos fundamentales:

- *Los elementos del entorno*: todos los componentes nativos del lenguaje de programación que proveen las herramientas operativas para el cumplimiento de los métodos invocados. Algunos de los componentes utilizados son: MSCHAR, para la elaboración de gráficos, MSAPI para la generación de reportes a través del correo electrónico, los TIMER para la distribución temporal de funciones de supervisión y finalmente, WINSOCK para la habilitación y establecimiento de sockets entre el sistema de administración y los nodos.
- *Objeto Manager*: Es provisto por la librería comercial *PowerSNMP*, a través de la cual se ejecutan las 7 funciones del protocolo SNMP. El Objeto Manager se caracterizó en el diseño general del sistema en tres partes denominadas Conexión, Supervisión y Control. La primera subdivisión se refiere a las funciones relacionadas a la apertura de los puertos de comunicación, que interactuando con el módulo de configuración y haciendo uso del Winsock establece los canales de comunicación con las estaciones. La segunda subdivisión se refiere a las funciones Get de la librería, es decir, todas aquellas que se utilizan durante el proceso de encuesta secuencial del módulo de supervisión. Y por último, se hace referencia a las funciones Set, es decir aquellas que permiten modificar valores de forma remota, utilizadas principalmente en la modificación de tablas de ruta del módulo de control.

IV.6 Desarrollo e implementación del Prototipo

Después de realizado el diseño del Sistema de Control, Supervisión y Gestión de redes basadas en el protocolo IPv4, se procedió al desarrollo de un prototipo capaz de brindar las facilidades descritas en los apartados anteriores, a través de una interfaz amigable y basados en los paradigmas de Programación Orientada a Objetos y Orientada a Eventos que ofrece el ambiente de desarrollo empleado.

Para modelar el funcionamiento del prototipo se presentaron las secuencias dinámicas de acción entre los distintos grupos de objetos de la aplicación, para lo cual se emplearon diagramas distintos, denominados Diagrama de Colaboración y Diagrama de Secuencia.

Un Diagrama de Colaboración muestra una interacción organizada basándose en los objetos que toman parte en la interacción y los enlaces entre los mismos. Un diagrama de Secuencia muestra una interacción ordenada según la secuencia temporal de eventos. *(González y Muñoz, 2004)*

En los resultados de la investigación, descritos en el Capítulo V, se explicarán con detalle los procedimientos implementados y el comportamiento de la aplicación, caracterizando de esta forma cada módulo funcional.

IV.7 Evaluación de alternativas para generar reportes móviles.

Uno de los objetivos que se plantearon inicialmente durante el proceso de formulación del presente proyecto de investigación y desarrollo, fue la evaluación de las alternativas para generar reportes móviles en tiempo real. En otras palabras, investigar la necesidad y conveniencia de agregar un módulo que ofreciera ventajas de movilidad, para el desarrollo futuro de una versión más innovadora del prototipo implementado.

Se realizó un análisis económico y técnico, considerando la real utilidad de los sistemas de Gestión de Redes existentes en el mercado y la relación costo-beneficio de incluir una arquitectura de comunicación que ofreciera estas ventajas; se derivó en que recibir el reporte de un evento o estado de una estación que forme parte de la topología de una red, a través de un SMS, no genera ningún valor agregado en cuanto a solución del problema de refiere. Los administradores de red no pudiesen modificar a través del teléfono celular la configuración de las estaciones en gestión, por lo tanto la movilidad, más allá de ser un elemento innovador, propio de las nuevas tendencia de comunicación, no representa un factor determinante en los reportes que genera un sistema de Control, Supervisión y Gestión de Redes.

IV.8 Pruebas con datos contruidos y reales en una red TCP/IP de 10 estaciones.

Con la finalidad de validar y verificar el funcionamiento del prototipo implementado, se realizaron pruebas con datos contruidos y datos reales.

La primera etapa de prueba se basó en un híbrido entre datos reales y simulados en una red privada, enlazada a través de un enrutador inalámbrico LinkSyS, formada por cuatro estaciones de computadoras personales con Sistema Operativo Windows XP, un ordenador portátil con Sistema Operativo Linux, versión Ubuntu y un teléfono VoIP Linksys. Los datos contruidos se generaron automáticamente a través de la simulación de una red virtual de seis estaciones. En el capítulo V se presentan algunos gráficos que se obtuvieron en la primera sesión de pruebas.

Posteriormente se instaló el prototipo en un laboratorio de 16 estaciones. Se realizaron las pruebas midiendo las mismas características que en la primera fase, pero este caso con todos los datos reales. Adicionalmente, se verificó el funcionamiento del módulo de control del prototipo que se encarga tanto del enrutamiento del tráfico de la red, al existir falla o congestión, como de la

modificación de forma remota de las variables de lectura y escritura contenida en el MIB de cualquier estación en gestión.

Se capturaron las pantallas de reporte generadas por el sistema durante el proceso de validación y pruebas, las cuales serán presentadas en el capítulo V, correspondiente a los resultados del proyecto.

CAPÍTULO V

RESULTADOS

Después de la realización de todas las actividades planificadas para el diseño e implementación del Sistema de Control, Supervisión y Gestión de redes basadas en el protocolo IPv4, se obtuvieron los siguientes resultados, caracterizados por etapa de desarrollo.

V.1 Documentación Teórica.

A través del proceso de investigación tanto aplicada como descriptiva realizada en la primera etapa de desarrollo del Trabajo Especial de Grado, se obtuvieron los basamentos teóricos del funcionamiento del Protocolo Simple de Gestión de Redes SNMP, necesarios para el diseño e implementación del Sistema de Control, Supervisión y Gestión de Redes basadas en el Protocolo IPv4.

V.2 Investigación de Mercado

La realización del estudio de las herramientas existentes en el mercado, derivó en la definición de las características mínimas que debía contener el sistema en desarrollo para considerarse una herramienta competitiva. Dichas características son:

- a) Tener la capacidad de generar reportes en tiempo real del ancho de banda, consumo de memoria del CPU, datos transmitidos y recibidos por cada interfaz, errores transmitidos y recibidos, latencia, disponibilidad.

- b) Presentar una interfaz amigable, que sea operable no sólo por personas en capacidad de administrar la red, sino por cualquier usuario en general con conocimientos medios de interoperabilidad y redes de comunicación.

- c) Generar reportes estadísticos en forma de gráficos de barra y de líneas, que permitan visualizar el comportamiento de la red a través de una escala dinámica de tiempo.

En la *Tabla 1* se presenta el cuadro comparativo que se obtuvo como resultado de la investigación de mercado.

*Tabla 1.- Cuadro Comparativo de las Herramientas Existentes en el Mercado
Fuente: Elaboración Propia.*

V.3 Requerimientos de Software.

La definición de los requerimientos de software se dividió en dos partes fundamentales. La primera de ellas referida a la selección del Sistema Operativo sobre el cual soportar el diseño y posterior implementación de aplicación y la segunda, relacionada con la determinación del lenguaje de programación a utilizar.

Como se mencionó en el Capítulo IV, el Sistema Operativo seleccionado fue Windows XP, por sus ventajas en cuanto a compatibilidad con los fabricantes de hardware, facilidad para desarrollar entornos gráficos amigables, sus opciones para gestionar redes de comunicación y un aspecto importante, la familiarización que existe con esa plataforma. El producto resultante del estudio comparativo se sintetiza en la *Tabla 2*.

Por su parte, la selección del lenguaje de programación resultó un poco más compleja. En el desarrollo del Trabajo Especial de Grado se explicaron las causas por las cuales se decidió sustituir el lenguaje C++ y el compilador Borland Builder, por Visual Basic 6.0. Dicha decisión está soportada por el cuadro comparativo que se presenta en la *Tabla 3*.

Teniendo Visual Basic 6.0 un solo ambiente de desarrollo (a diferencia del lenguaje C++ para el que se utilizan varios entornos de programación) el estudio comparativo realizado entre los distintos compiladores de lenguaje C++, donde se analizaron las ventajas y desventajas de Dev, Visual y Borland Builder, se conservó como fuente importante de información para el conocimiento detallado de los lenguajes evaluados y se puede consultar en el APÉNDICE II

*Tabla 2.- Cuadro Comparativo Sistema Operativo Windows Vs. Linux
Fuente: Elaboración Propia*

*Tabla 3.- Cuadro Comparativo Lenguajes de Programación
Fuente: Elaboración Propia.*

V.4 Diseño del Sistema

En la *Figura 6*, se muestra el diseño del sistema y la relación existente entre cada uno de los módulos funcionales, el Objeto Agentes, los métodos del ambiente de programación provenientes de la librería importada *PowerSNMP* y los elementos del entorno que permiten la interconexión a nivel de software de todos los factores que intervienen.

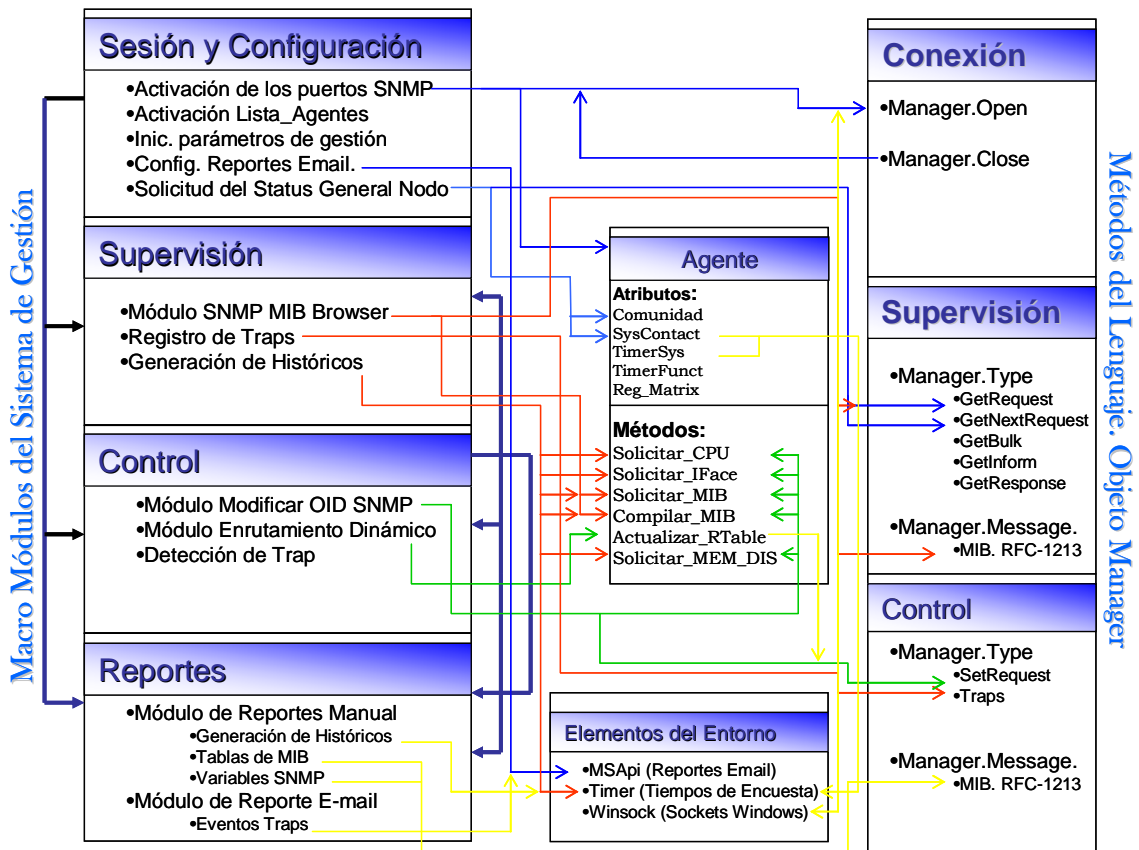


Figura 6. Diseño General del Sistema de Control, Supervisión y Gestión
Fuente: Elaboración Propia

El diseño general presentado, modela el funcionamiento completo del sistema. Los 4 módulos funcionales, Sesión, Supervisión, Control y Reportes, invocan los métodos del Objeto Agentes y éstos a su vez utilizan las funciones específicas del lenguaje de programación y de la librería *PowerSNMP*.

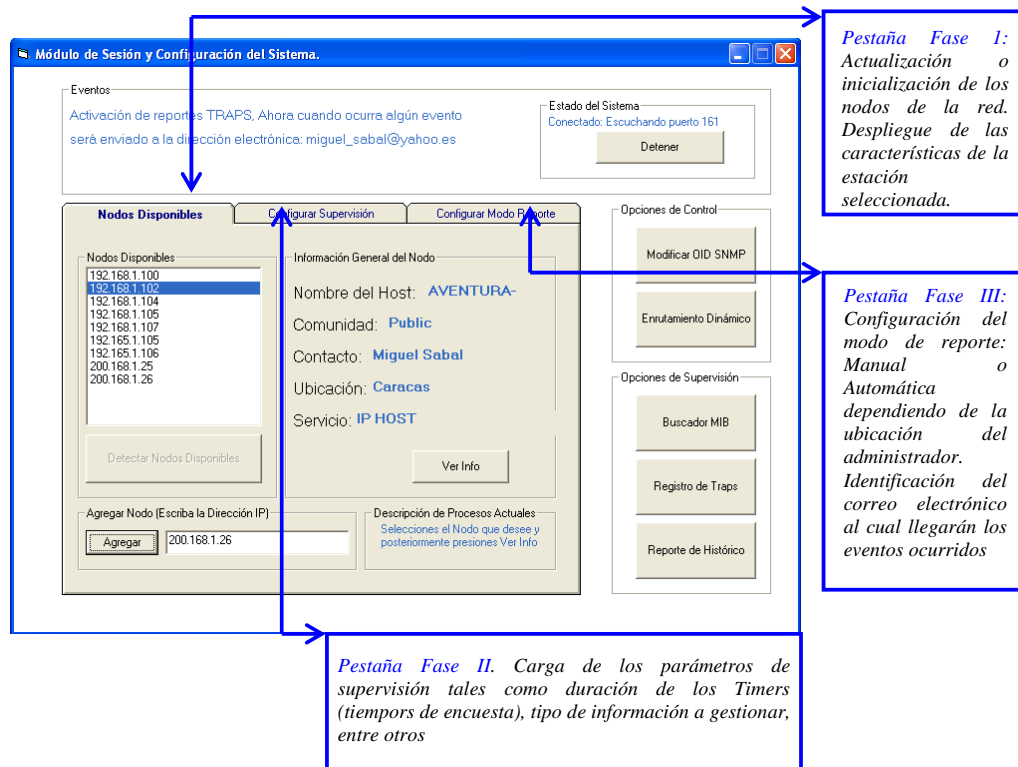
V.5 Prototipo.

Una vez finalizado el diseño general del sistema, se procedió a realizar el desarrollo de la aplicación. En el capítulo IV, se describieron las características de cada módulo; en el presente capítulo se presentará el resultado final de la programación del prototipo, diferenciado por módulo funcional.

V.5.1 Módulo de Sesión y Configuración.

El Módulo de Sesión y Configuración se divide en 3 fases. La primera relacionada con la inicialización o actualización de la lista de agentes disponibles y la presentación de las características generales de cada estación. La segunda, se refiere a la sesión de los parámetros de supervisión, tales como tiempos de encuesta, tipos de información; finalmente, la tercera fase se refiere a la configuración del modo de reporte que realizará el sistema.

En la *Figura 7* se presenta la pantalla de inicio del prototipo, que contiene el menú de opciones que ofrece la aplicación (a través del cual se invocan los otros tres módulos definidos en el sistema) y la información relacionada con las tres fases que componen el Módulo de Sesión y Configuración. En la *Figura 7* se encuentra activa únicamente la pestaña que gestiona la información de la *Fase I*, es decir, la relacionada a la inicialización o actualización de la lista de agentes activos en la red y el despliegue de la información general de la estación seleccionada por el usuario. En el APÉNDICE III se muestran las pantallas y el funcionamiento de las demás pestañas que componen el módulo de Sesión y Configuración.



*Figura 7. Pantalla de Inicio del Prototipo. Módulo de Sesión y Configuración
Fuente: Elaboración propia*

En la figura 8 y 9 se presenta el diagrama de colaboración y de secuencia respectivamente de la *Fase 1*, donde se puede apreciar que una vez que el usuario decide activar los puertos de comunicación SNMP (161 y 162), y se inicialice el sistema, se pueden llevar a cabo dos acciones, la primera es agregar una estación a la red de forma manual a través de la dirección IP de la misma y la segunda, solicitar al prototipo envíe un Broadcast, espere la respuesta de las estaciones activas y actualice la lista de nodos en gestión que se despliega por pantalla. De igual forma en la Fase 1, a través del uso de la función `GetNextRequest` del protocolo SNMP, se solicitan los valores generales del nodo seleccionado y se presentan por pantalla. Estos valores iniciales que se solicitan a cualquiera de las estaciones en gestión son:

- Nombre del Sistema
- Comunidad
- Contacto
- Ubicación
- Tipo de servicio

La *Fase 1* no está concebida dentro del módulo de supervisión, sin embargo se realiza la solicitud de las variables contenidas en el “ramal sistema” del MIB, para introducir al usuario las características esenciales de cada uno de los nodos que se encuentran activos y así identificar de forma rápida, oportuna y eficiente alguna estación en específico que se sospeche esté generando conflictos en la topología de la red.

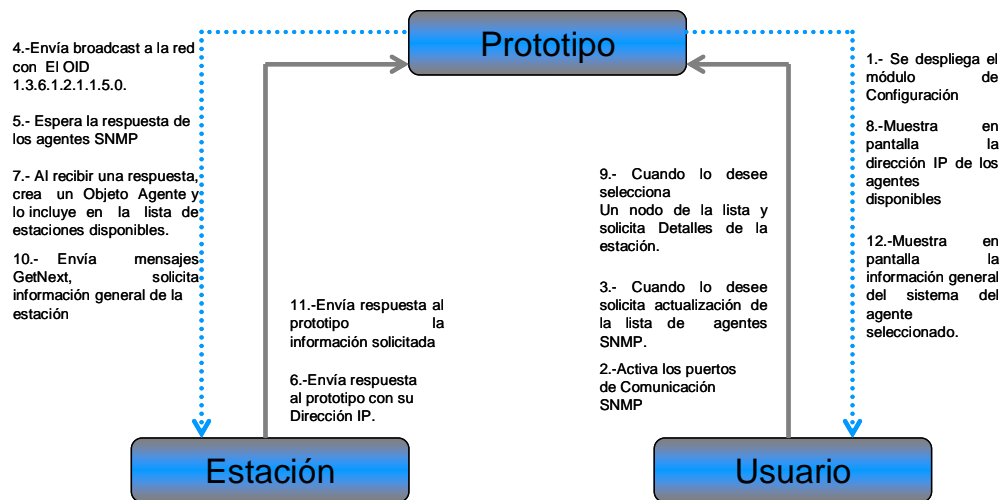


Figura 8. Diagrama de Colaboración Fase 1 del Módulo de Sesión y Configuración Fuente:

Elaboración propia

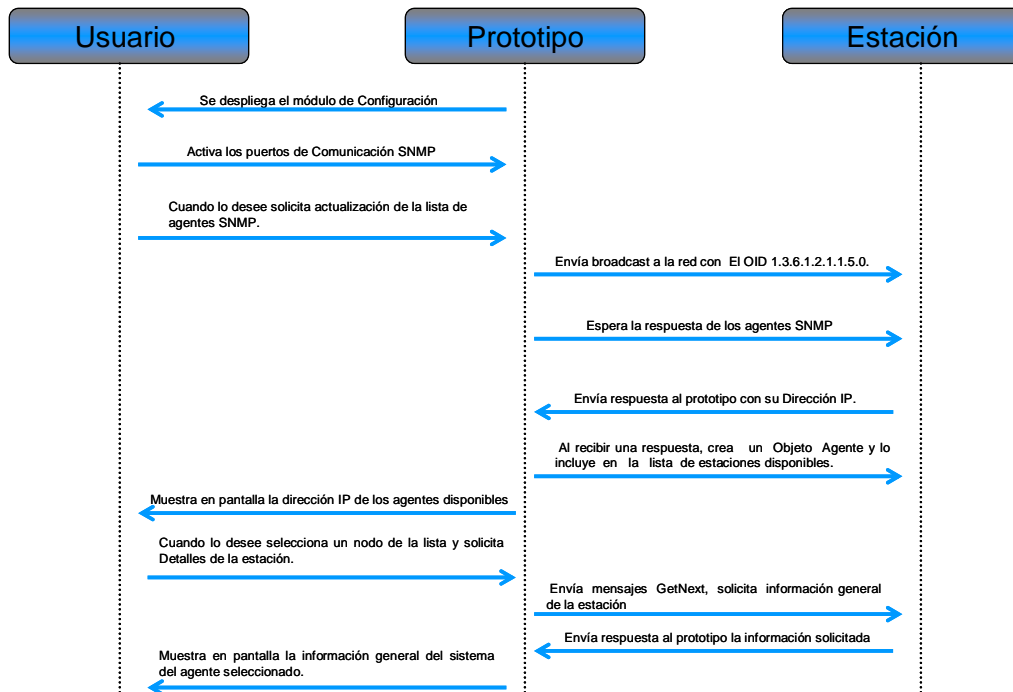
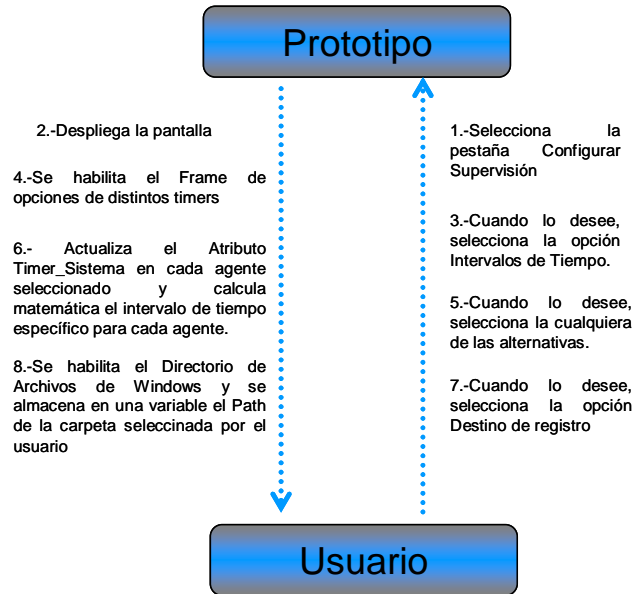


Figura 9. Diagrama de Secuencia Fase 1 del Módulo de Sesión y Configuración
Fuente: Elaboración propia

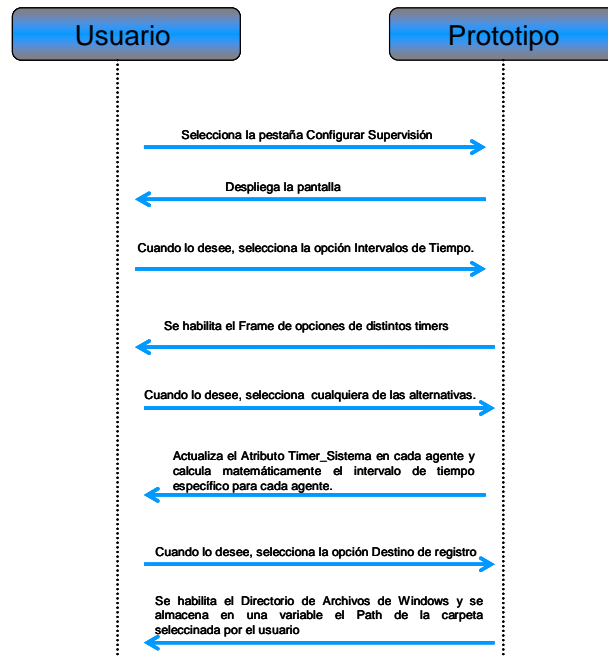
Una vez actualizada o inicializada la lista de agentes disponibles, el prototipo habilita las demás pestañas pertenecientes al Módulo de Sesión y Configuración. La *Fase 2*, está representada en los diagramas de colaboración y secuencia que se muestran en las Figuras 10 y 11. Esta fase no interactúa con la estación, pues como se puede apreciar, en ésta sólo se definen los parámetros generales que enmarcarán el proceso de supervisión, descrito en el próximo módulo.

El primer valor que se inicializa es el tiempo asignado a cada *Objeto Agente* de la lista de estaciones disponibles para realizar su encuesta. Como se explicó en el capítulo IV, cada *Objeto Agente* tiene un intervalo de tiempo para realizar distintas peticiones a la estación que representa dentro del modelo y a su vez, cada petición (GET) debe realizarse en otro intervalo de tiempo específico. Esa compleja asignación temporal que soporta al módulo de supervisión del prototipo, se define en esta fase 2 del Módulo de Sesión y Configuración.

Adicionalmente, en esta fase se selecciona el *Path* o ruta de acceso del archivo de texto que almacena los eventos Traps que se generan durante el proceso de supervisión.



*Figura 10. Diagrama de Colaboración. Fase 2 Módulo de Sesión y Configuración
Fuente: Elaboración Propia.*



*Figura 11. Diagrama de Secuencia. Fase 2 Módulo de Sesión y Configuración
Fuente: Elaboración Propia.*

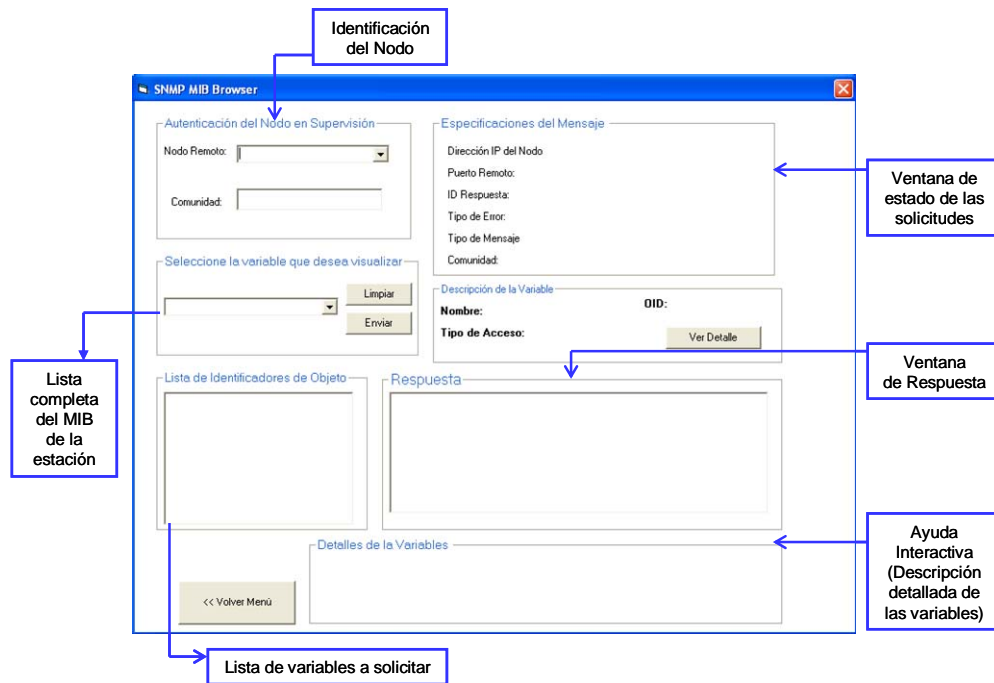
La *Fase 3* del Módulo de Sesión y Configuración se encarga de inicializar el modo de reporte que genera el sistema: Manual o Automático. La explicación detallada de las características así como el diagrama de Colaboración y Secuencia, se pueden consultar en el APÉNDICE IV (MANUAL DEL SISTEMA)

V.5.2 Módulo de Supervisión

El Módulo de Supervisión se representa en el prototipo a través de dos aplicaciones. La primera, estipulada en el alcance inicial del proyecto, facilita la obtención de diversas variables que se procesan y en conjunto con el módulo de reportes, generan gráficas amigables que permiten visualizar el comportamiento de la red. La segunda, surgió como valor agregado durante la implementación del sistema y consiste en un “Buscador MIB”, que es una herramienta similar a la utilizada durante el desarrollo del proyecto (MG-SOFT LAB), que permite conocer el significado y el valor de los distintos identificadores de objetos (OID’s) contenidos en el MIB de cualquier estación.

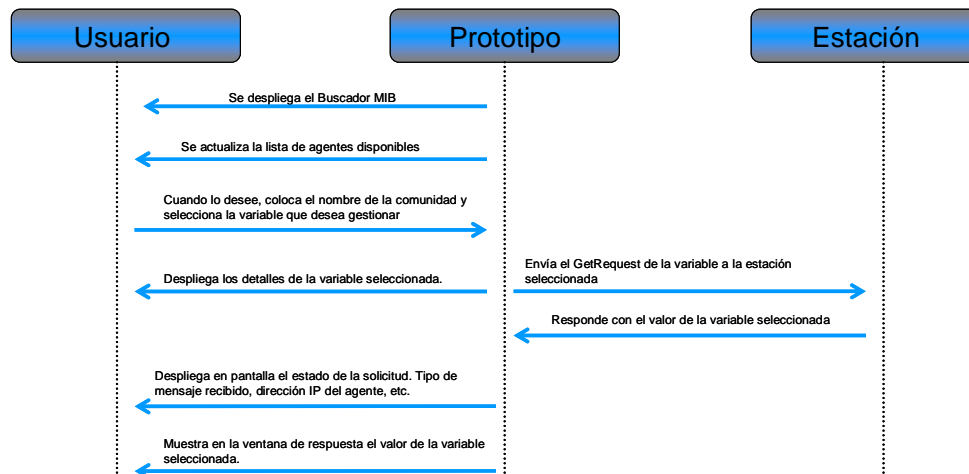
La pantalla correspondiente a la generación de gráficas producto de las encuestas secuenciales que realiza cada Objeto Agentes sobre la estación que representa se podrá visualizar como resultado del módulo de reportes, a pesar de que tenga un componente de participación del módulo de supervisión muy elevado

En la *Figura 12* se presenta el estándar de la pantalla que representa el Buscador MIB del prototipo, en la cual se distingue una lista de todas las variables que pueden ser gestionadas a través de SNMP, una ventana que reporta el status del mensaje de solicitud y un espacio para visualizar el detalle (descripción del significado) de cada uno de los Objetos contenidos en la lista. El Buscador MIB desarrollado, según la investigación realizada, es la única herramienta existente que presenta la descripción de los Identificadores Objetos en español, por lo tanto, este resultado además de sobrepasar las expectativas planteadas inicialmente, porque no se encontraba en el alcance, genera un aporte importante en la gestión de redes.



*Figura 12. Pantalla Estándar Buscador MIB
Fuente: Prototipo*

El diagrama de secuencia del módulo de supervisión se presenta en la Figura 13 cuya explicación detallada se podrá consultar en el Manual del Sistema, ubicado en el APÉNDICE IV, así como los detalles de la implementación técnica.

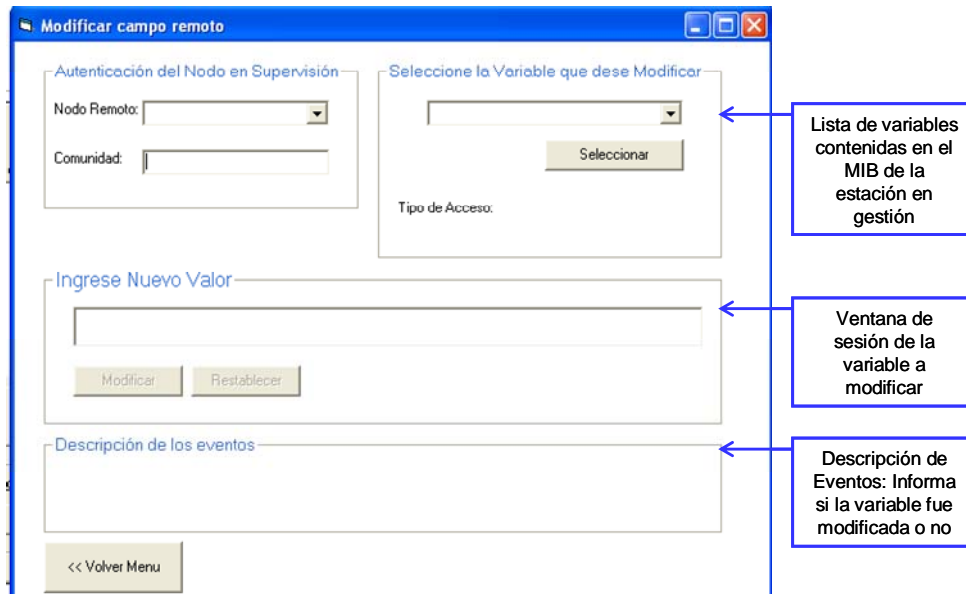


*Figura 13. Diagrama de Secuencia Módulo de Supervisión. Buscador MIB
Fuente: Elaboración Propia*

V.5.3 Módulo de control

El módulo de control también ofrece dos grandes funcionalidades. La primera de ellas, que se modela a través de la pantalla de *Figura 14*, permite modificar el valor de cualquier variable de las estaciones activas, siempre que éstas sean de lectura y escritura. Y la segunda, que no se puede visualizar a través de ninguna pantalla, es la toma de decisiones que realiza el sistema en cuanto al enrutamiento del tráfico en caso de ocurrir cualquier falla o congestión de una estación, basado en la modificación de las rutas por defecto.

Así como el Buscador MIB resultó un valor agregado en el desarrollo de la aplicación, la primera función mencionada anteriormente también representa una innovación que surgió durante la programación del prototipo. SNMP permite modificar cualquier valor de forma remota, desde el nombre que tiene el sistema (que lo asigna por defecto el sistema operativo durante la instalación del mismo) hasta el tamaño de los datagramas que pueden ser enviados o recibidos de una interfaz. Eso hace mucho más robusto el módulo de control que no se limita al enrutamiento dinámico como se planteó inicialmente.



*Figura 14. Pantalla Estándar Módulo de Control.
Fuente: Prototipo*

En la *Figura 14* se visualizó el estándar de la pantalla que permite modificar el valor de las variables de forma remota. En la esquina superior izquierda, se muestra el frame de autenticación del nodo en gestión, a través del cual se selecciona la dirección IP y se escribe el nombre de la comunidad; a su lado se encuentra la lista de variables precompiladas del MIB de la estación y una etiqueta que indica si el objeto seleccionado es de lectura, de escritura o ambas. Finalmente, en la parte inferior de la pantalla se encuentra el cuadro de texto para que el usuario coloque el nuevo valor que desea asignar. La ayuda interactiva que acompaña al usuario en todo el prototipo le indicará si el valor fue modificado o no. En el APÉNDICE III se presentan los casos de uso de la pantalla y las instrucciones detallada de su funcionamiento.

El diagrama de secuencia del módulo de control se presenta en la *Figura 15*. La caracterización y explicación detallada de éste se localiza en el Manual del Sistema desarrollado en el APÉNDICE IV

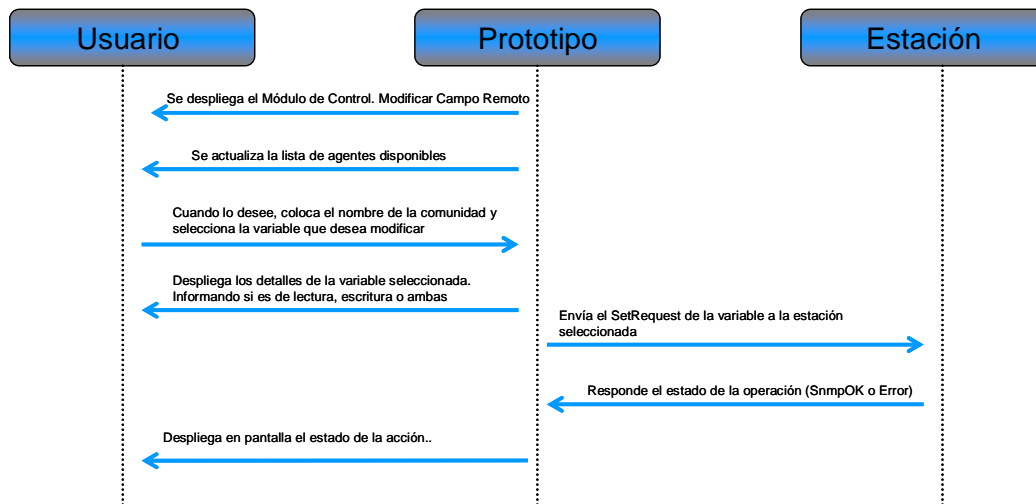


Figura 15. Pantalla Estándar Módulo de Control. Modificar Campo Remoto
Fuente: Elaboración Propia

V.5.3 Módulo de Reportes

El módulo de reportes representa la interfaz más importante de todo el prototipo ya que a partir de éste se hace la demostración de todos los procesos de supervisión y control ejecutados por los módulos anteriores. La pantalla estándar que se muestra en la *Figura 16* presenta un menú de opciones en la columna izquierda. Cada botón se refiere un tipo de reporte: Interfaces de Red, Memoria Física y Dispositivos, Latencia, Consumo de CPU y el último que se refiera al SNMP MIB Table. Al seleccionar alguna de las alternativas, se despliega un sub-menú debajo de los botones, que especifica aún más las características del reporte solicitado. En el centro de la pantalla se encuentra la Ventana de Reportes en la cual aparecerán las gráficas de históricos y las listas de dispositivos, procesos o programas según sea el caso. La pantalla estándar muestra también el espacio para la ayuda interactiva, presente en todos los módulos del prototipo. La descripción de todos los reportes y funcionamiento de cada uno de los botones se encuentra especificada en el Manual de Usuario desarrollado en el APENDICE III

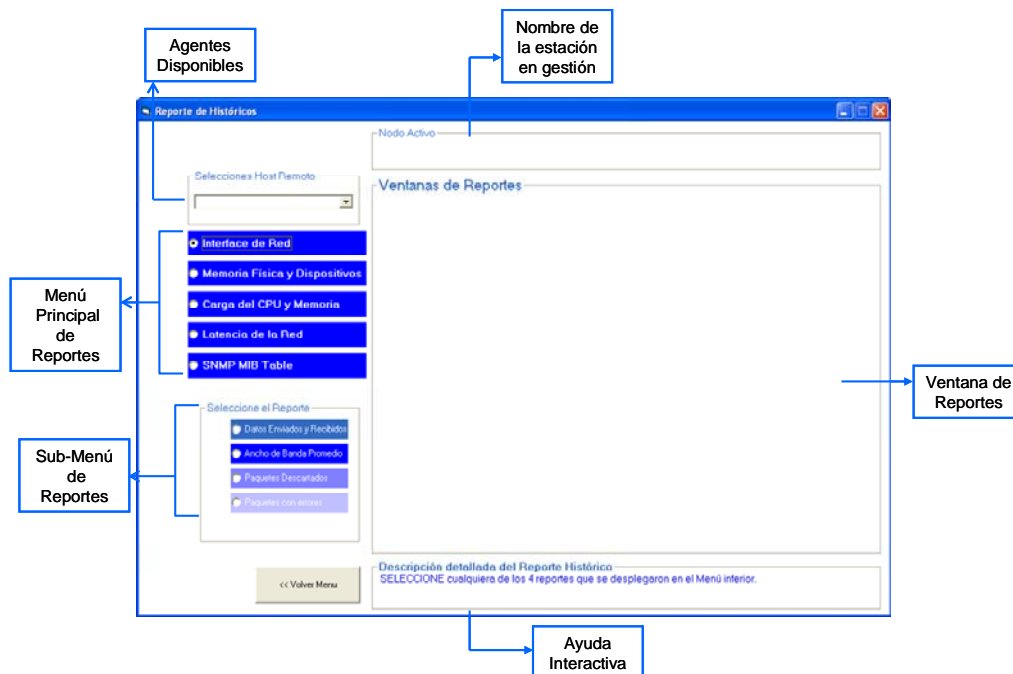


Figura 16. Pantalla Estándar Módulo de Reporte
Fuente: Prototipo

El diagrama de secuencia del módulo de reportes se presenta en la *Figura 17* y la explicación del funcionamiento se presenta en el APÉNDICE IV, Manual del Sistema.

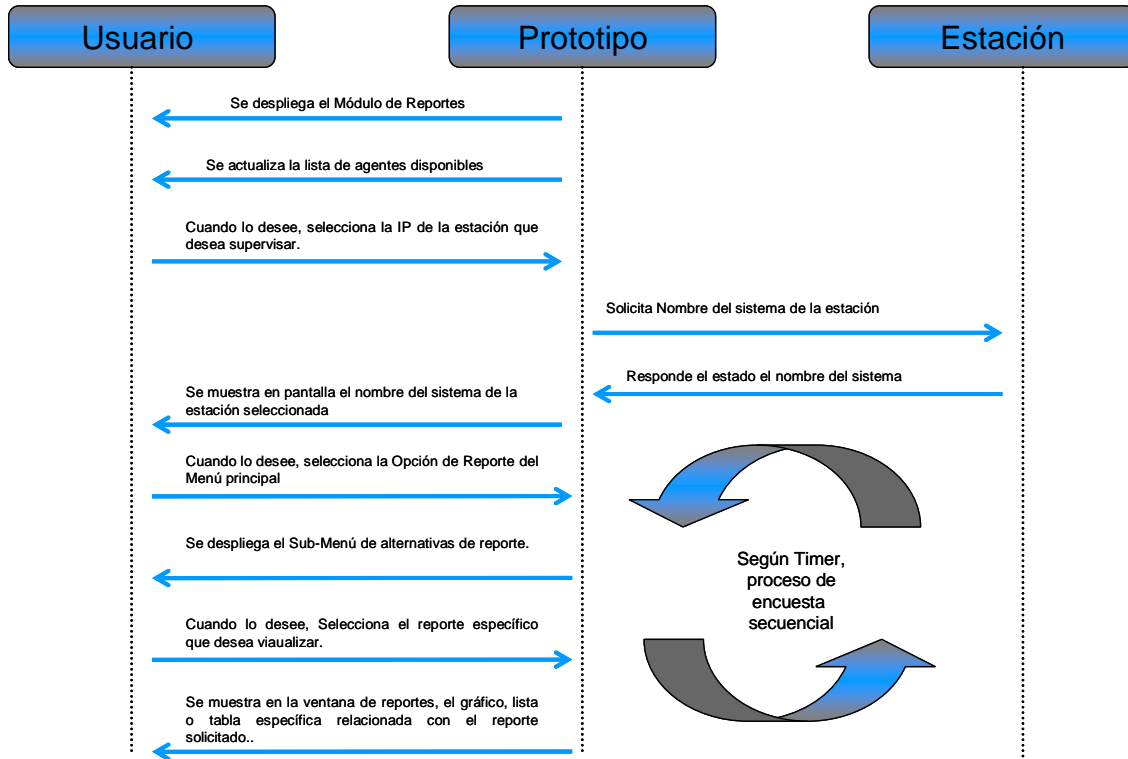


Figura 17. Diagrama de Secuencia Módulo Reporte
Fuente: Elaboración Propia

V.6. Pruebas

Después de culminado el desarrollo del prototipo, se realizaron pruebas con datos contruidos (simulados) y datos reales un una red TCP/IP de 10 nodos. En las Figuras 18, 19 y 20 se muestran algunos de los resultados de las pruebas realizadas. En la tabla 4 que se muestra al final del presente capítulo, se aprecia un resumen de las pruebas realizadas con 2 estaciones durante 48 horas. Los valores que allí se muestran son promedios calculados manualmente.

La Figura 18 presenta el reporte de los Datos Enviados Vs. Datos Recibidos (medido en cantidad de paquetes), de una estación cuyo proceso de supervisión tuvo una duración de 8 horas (desde la 5 de la tarde hasta las 12 de la noche). En la gráfica se puede apreciar claramente una línea asociada al tráfico entrante y otra al saliente, así como los picos que se generan en las horas que hubo mayor utilización de la interfaz para el intercambio de información

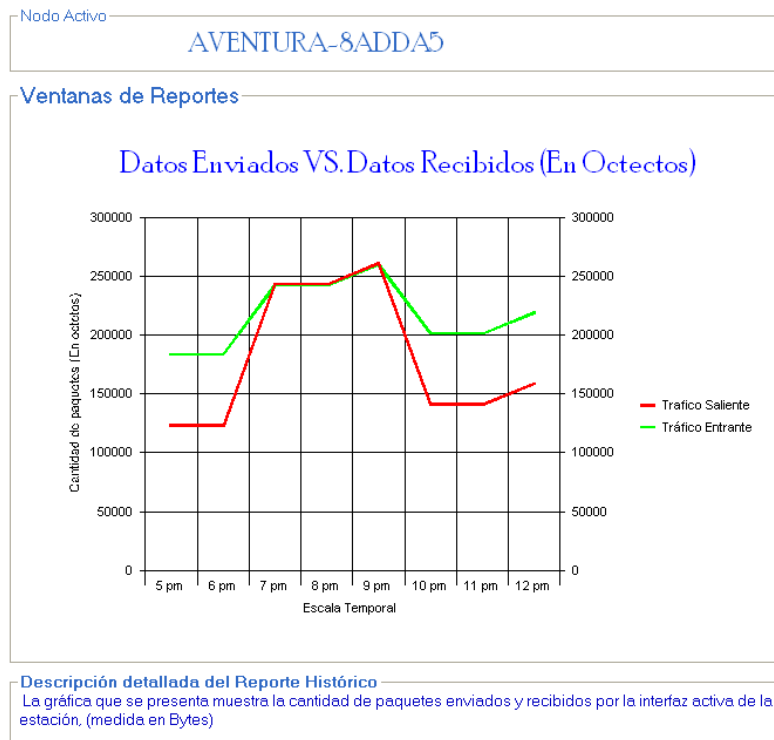


Figura 18. Prueba datos Reales. Datos Enviados Vs. Datos Recibidos
Fuente: Prototipo

En la *Figura 19* se despliega una lista de las unidades de almacenamiento disponibles en el mismo nodo. Una de las ventajas más resaltantes del prototipo es la capacidad de detallar la información actual de las estaciones. Como se observa en el resultado presentado en la *Figura 19*, en la lista se detalla inclusive el nombre del CD que se encuentra dentro de la unidad D del nodo en supervisión, garantizando esto características muy poderosas de control y gestión.

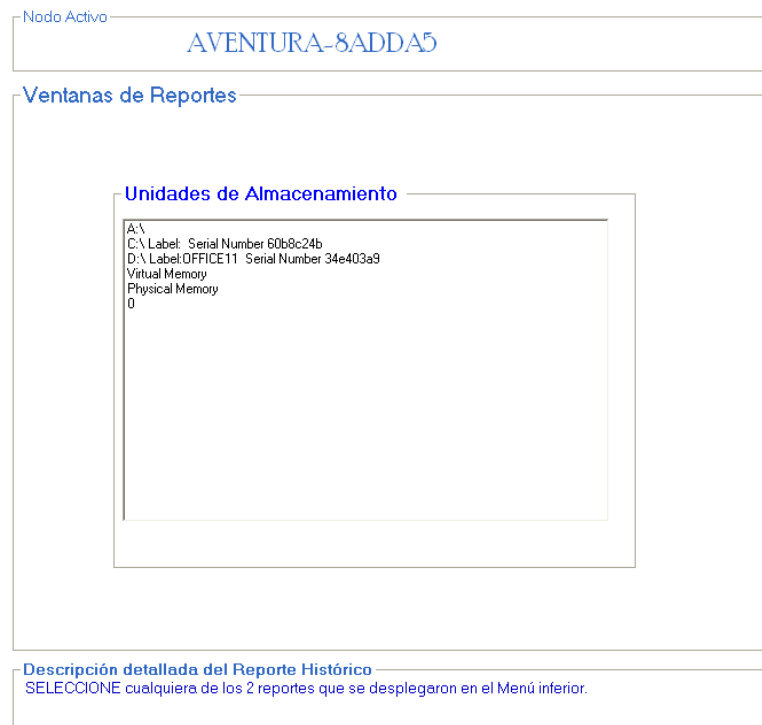


Figura 19. Prueba datos Reales. Unidades de Almacenamiento
Fuente: Prototipo

Finalmente en la *Figura 20* se presenta el resultado del reporte SNMP MIB Table, que no es más que la tabla de objetos MIB del nodo en supervisión. En este caso, la lista que se presenta se refiere a las variables de sistema (nombre del equipo, locación, contacto y descripción del Sistema Operativo). Sin embargo, la aplicación permite gestionar los datos relacionados a interfaces, enrutamiento ip, usuario, entre otros.

VI. CONCLUSIONES Y RECOMENDACIONES

Una vez realizado el diseño y desarrollo del Sistema de Control, Supervisión y Gestión de Redes basadas en el protocolo IPv4, a través de la implementación del protocolo SNMP, se obtuvieron las siguientes conclusiones.

En la investigación teórica de SNMP se determinó que la ventaja fundamental de este protocolo es que su diseño es simple por lo que su implementación es sencilla en grandes redes y la información de gestión que se necesita intercambiar ocupa pocos recursos de la red. Además, permite al usuario elegir las variables que desea supervisar. Es el protocolo más extendido y la popularidad la ha conseguido al ser el único protocolo que existió en un principio y por ello casi todos los fabricantes de dispositivos como bridges, host y enrutadores diseñan sus productos para soportar SNMP.

A través de la realización del estudio de mercado, se apreció con claridad que los sistemas de gestión existentes son muy poderosos y competitivos. En la mayoría de los casos, tienen la capacidad de proveer información en tiempo real del estado de los nodos: disponibilidad, latencia, ancho de banda utilizado, consumo de memoria y CPU, entre otros. Algunos tienen la ventaja de determinar el rendimiento general de una estación y de presentar históricos que permiten visualizar el comportamiento de la red en un lapso de tiempo determinado. Sólo uno de los sistemas estudiados realiza alertas en tiempo real que van directamente a la persona encargada de administrar la red a través de correo electrónico. Sólo algunos de los sistemas, a parte de la supervisión, controlan la red en caso de presentarse caídas del sistema, congestión o fallas en general, siendo esto una de las características diferenciales e innovadoras del prototipo desarrollado

En la determinación de los requerimientos de software, resultó bastante complejo tomar una decisión de cuál plataforma tecnológica utilizar para el diseño e implementación del Sistema. En primer lugar, Windows es un Sistema Operativo

amigable, fácil de instalar y resulta mucho más sencillo desarrollar interfaces gráficas de calidad bajo este entorno. Sin embargo, en términos generales la tendencia del Software Libre es cada vez mayor y Linux está teniendo gran auge debido a que resulta un poco más estable y es más potente para la administración de redes. Basado en la realización de esta investigación y bajo la premisa de la familiarización que existe con el entorno Microsoft, la facilidad de desarrollar ambientes gráficos amigables y la compatibilidad con los diversos fabricantes de hardware, se decidió realizar el desarrollo haciendo uso del Sistema Operativo Windows XP, decisión que sin duda fue asertiva y contribuyó en definitiva a que la implementación se realizara en el tiempo planificado.

Al realizar el análisis de las consecuencias de utilizar Visual Basic 6.0 para el desarrollo de la aplicación, lenguaje no compatible con entornos Linux, se concluyó que no representa ninguna limitación que el Sistema de Gestión corra exclusivamente en ambiente Microsoft, ya que el protocolo SNMP es tan poderoso que puede supervisar y controlar cualquier dispositivo de red basado en IPv4 y que tenga un servicio SNMP activo (independientemente del Sistema Operativo), de tal forma que la única limitación se reduce a que el administrador de la red deberá cargar la aplicación sobre Windows. Las estaciones gestionables (switches, host, router, enlaces punto a punto, bridge, entre otros) pueden correr bajo cualquier entorno. Adicionalmente, Visual Basic 6.0 es un lenguaje de fácil interpretación, lo que garantiza que cualquier estudiante de Ingeniería de Telecomunicaciones, con conocimientos básicos de programación Orientada a Objetos, está en capacidad de continuar el desarrollo del prototipo para la realización de versiones aún más competitivas, con la posible implementación de SNMP V3, que aún se encuentra en etapa de evaluación.

Como conclusión de la primera etapa del Trabajo Especial de Grado, relacionada con el levantamiento de información, se puede afirmar que la documentación teórica realizada a través de una investigación aplicada y descriptiva del protocolo Simple de Gestión de Redes SNMP, sirvió como base para casi todas las actividades asociadas al desarrollo del proyecto; El estudio de Mercado, por su

parte, resultó una herramienta esencial para el diseño conceptual del prototipo y para precisar las características que hicieron del sistema un producto competitivo; mientras que el análisis comparativo de los requerimientos de software permitió que la programación de la aplicación se hiciese en un ambiente de desarrollo poderoso, pero a la vez amigable y sencillo de utilizar.

El análisis de las alternativas para generar reportes móviles, es decir, aquellos que se reciben en dispositivos inalámbricos, derivó en la conclusión que los sistemas de gestión y supervisión de redes deben proveer información oportuna, pero sobre todo completa y sustentada. Un reporte de una falla a través de un mensaje de texto, no arrojaría ningún detalle de la situación problemática y menos aún, permitiría al administrador de la red la solución del conflicto. Sin embargo, la generación de reportes vía correo electrónico, provee de cualquier detalle que se requiera y también ofrece características de movilidad, esto es: cada vez es mayor la penetración de la tecnología EVDO que permite el acceso a correos electrónicos a través del teléfono celular, dándole mayor peso a la decisión tomada de desarrollar la aplicación para reportar las fallas a través de este medio.

En cuanto al prototipo desarrollado se puede concluir que sus funcionalidades sobrepasaron las expectativas planteadas inicialmente. Además de proveer herramientas para la supervisión oportuna de las estaciones de la red y tener las funciones de control que permiten el enrutamiento dinámico del tráfico en caso de fallas o congestión, se incluyeron dos módulos adicionales que permiten complementar los procesos antes descritos, a través de las facilidades que ofrece el protocolo SNMP.

El primer módulo que se desarrolló como valor agregado al prototipo fue “*Modificar Variable Remota*”, que permite cambiar cualquier variable contenida en el MIB de una estación siempre que ésta sea de lectura y escritura. Esta herramienta es de gran utilidad porque permite modificar valores como el tamaño máximo de los paquetes que pueden pasar por una determinada interfaz, lo cual representa una acción muy útil para optimizar el ancho de banda utilizado por

algunas estaciones en determinado momento. El segundo módulo agregado fue el *Buscador MIB*, que es una herramienta similar a la utilizada como software de apoyo que permite solicitar el valor de cualquier variable SNMP y despliega por pantalla el significado de la variable solicitada.

Basado en las descripciones anteriores, se puede afirmar que el prototipo desarrollado cumple con las características planificadas inicialmente, es decir, permite el **Control**: a través del enrutamiento dinámico del tráfico de la red, **La Supervisión**: a través del reporte del estado de los nodos y las alertas que se generan vía correo electrónico de los eventos ocurridos y la **Gestión**: que se alcanza a través de la sinergia de los dos conceptos anteriores, al permitir administrar de forma remota los parámetros de funcionamiento de las estaciones que forman la topología de la red.

Se soluciona entonces la situación problemática que motivó la realización del presente proyecto, a través del desarrollo de un sistema que permite la supervisión de los nodos que integran una red basada en protocolo IPv4, que no representa una inversión mayor a los \$19 (costo de la librería SNMP utilizada) y que además ofrece la facilidad de controlar de forma dinámica el tráfico de la red en caso de existir falla o congestión.

El prototipo desarrollado es escalable, por tanto se recomienda la continuidad de su implementación técnica y evaluación. Según las pruebas realizadas y tal como se planteó inicialmente, soporta la supervisión de redes de hasta 100 estaciones. Un próximo Trabajo Especial de Grado desarrollado por un estudiante aspirante al título de Ingeniero en telecomunicaciones, puede ser la evaluación del funcionamiento del prototipo y la inclusión de mecanismos de “Seguridad Computacional” a la herramienta, a través de la implementación de algoritmos de encriptamiento o haciendo uso de SNMP v3 que aún está en proceso de evaluación, pero que ofrece considerables ventajas en el tema de seguridad.

Debido a la importancia que tiene el protocolo SNMP en el estudio y gestión de redes, se recomienda:

- Abrir líneas de investigación de los Sistemas de Gestión y Administración de Redes ya que las nuevas tendencias de las comunicaciones en Venezuela y en el mundo, se mueven velozmente hacia el entorno IP siendo los sistemas de gestión y supervisión cada vez más necesarios.
- Incluir SNMP en el plan de estudios de la carrera de Ingeniería de telecomunicaciones, específicamente en las cátedras de Telemática y Sistemas Telemáticos, debido a que representa el protocolo por excelencia de gestión y administración de redes.

BIBLIOGRAFÍA

Libros consultados

- RADCOM. (2002). *Guía Completa de Protocolos de Telecomunicaciones*. McGraw-Hill.
- FOROUZAN. (2002) *Transmisión de Datos y Redes de Comunicaciones*. McGraw-Hill.
- LEÓN, A. WIDJAJA, I.(2002). *Redes de Comunicación*. McGraw-Hill.
- [11] Booch, G. (1994). *Análisis y Diseño Orientado a Objetos con Aplicaciones* (2da ed.). España: Addison-Wesley.

Citas de Charlas, Clases Magistrales o Talleres:

- HERNANDEZ (2006). *Mercadotecnia para Ingenieros. Clase Magistral* Importancia del estudio de mercado en el desarrollo de nuevos productos. UCAB, Caracas.
- Centro de Investigación y Desarrollo de Ingeniería (CIDI), Escuela de Ingeniería de Telecomunicaciones. (2005). *Taller: Informe de Avance del Trabajo Especial de Grado de Ingeniería de Telecomunicaciones*. Trabajo no publicado.
- Sabino, C. (1986). *El proceso de la Investigación*. Caracas. Editorial Panapo

Fuentes Electrónicas

- HUIDOBRO, J. *SNMP. Un protocolo Simple de Gestión*. <http://www.coit.es/publicac/publbit/bit102/quees.htm>. (2005).
- TOBAL, J. *SNMP*. (30 Agosto 1999). <http://www.arrakis.es/~tobal/snmp.htm>. (Diciembre 2005).
- Network Working Group RFC 1157. (Mayo 1999). <http://www.ietf.org/rfc/rfc1157.txt>. (Diciembre 2005).
- LACUNZA, F. MAGAÑA, E. MARTÍNEZ, A. *Arquitectura de SNMP*. <http://www.arrakis.es/~gepetto/redes/rog08p3.htm>. (Diciembre 2005)

- LACUNZA, F. MAGAÑA, E. MARTÍNEZ, A. *SNMP. Especificaciones del Protocolo.* <http://www.arrakis.es/~gepetto/redes/rog08p4.htm>. (Diciembre 2005)
- MANUAL DE REFERENCIA DE RED HAT LINUX 9. *Versiones del Protocolo SSH.* <http://linux-cd.com.ar/manuales/rh9.0/rhl-rg-es-9/ch-ssh.html> (Enero 2006).
- MANUAL OFICIAL DE REFERENCIA DE RED HAT LINU 7.3. *Protocolo SSH.* <http://www.tu-chemnitz.de/docs/lindocs/RH73/RH-DOCS/rhl-rg-es-7.3/s1-ssh-protocol.html>. (Enero 2006).
- PÉREZ, C.A. (2003). *Windows Vs Linux. Mitos y Realidades* <http://www.rinconsolidario.org/linux/Win-Lin/Win-Lin.html> (Febrero, 2006)
- VILLANUEVA, P .(1999). *Windows Contra Linux.* <http://www.microsoft.com/spanish/msdn/comunidad/mtj.net/voices/art184.asp> (Febrero, 2006)
- Noticias 3D (2005). *Windows Vs. Linux en el 2005 Parte I* <http://noticias3d.com/articulo.asp?idarticulo=450&pag=9> (Febrero, 2006)
- FONTANET, F. (1999). *Borland C++ Builder Vs. Microsoft Visual C++.* http://www.arrakis.es/~rporcar/Builder_vs_Visual.pdf. (Febrero, 2006)
- JINMÉNEZ, C. *Programación grafica en DEV-C++ usando la librería no estándar WINBGIM.* <http://usuarios.lycos.es/charlytospa/charlytospa/dev.htm> (Febrero, 2006).
- CIAO (2006), http://www.ciao.es/bloodshed_Dev_C_4_0_72222 (Febrero, 2006)
- CIAO (2006). http://www.ciao.es/Borland_C_Builder_72191 (Febrero, 2006)
- CIAO (2006). http://www.ciao.es/Microsoft_Visual_C_6_0_72202 (Febrero, 2006).
- CARRANZA, R. (2005). *Cuadro Comparativo C++, PASCAL, JAVA.* <http://virtual.uav.edu.mx/tmp/26892139140901.doc> (Febrero, 2006)

- CIAO (2006). Opinión de lenguaje C++. http://www.ciao.es/Lenguajes_Orientado_a_Objeto_C_Opinion_524009. (Febrero, 2006)
- *¿Qué lenguaje elegir para programar?*. <http://www.gamedevwiki.com.ar/tiki-index.php?page=QueLenguajeElegir>. (Febrero, 2006).
- CIAO (2006). Opinión de JAVA. http://www.ciao.es/Java_72200 (Febrero,2006)
- MSDN (Noviembre, 2002). *Preguntas más frecuentes de Visual Basic*. <http://www.microsoft.com/spanish/msdn/vbasic/productinfo/vbasic03/faq/default.asp> (Febrero, 2006)