PONTIFICIA UNIVERSIDAD CATÓLICA DEL ECUADOR

SEDE AMBATO

ESCUELA DE INGENIERÍA DE SISTEMAS

DISERTACIÓN DE GRADO PREVIA A LA OBTENCIÓN DEL TÍTULO DE INGENIERO DE SISTEMAS

"Sistema de Control de Estaciones de Trabajo en la Red de la Escuela de Ingeniería de Sistemas de la PUCESA"

Franklin Vicente Guevara Moncayo

Ofelia Alexandra López Camino

PONTIFICIA UNIVERSIDAD CATÓLICA DEL ECUADOR PROGRAMAS ACADÉMICOS AMBATO
BIBLIOTECA

DIRECTOR DE DISERTACIÓN: Ing. Janio Jadán

Ambato, 2002

PONTIFICIA UNIVERSIDAD CATÓLICA
DEL ECUADOR
AMBATO
INGENIERIA DE SISTEMAS
SECRETARIA

PONTIFICIA UNIVERSIDAD CATÓLICA DEL ECUADOR

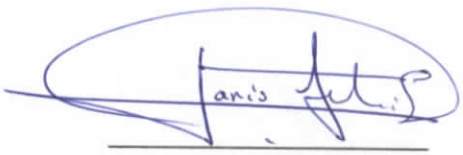
SEDE AMBATO

ESCUELA DE INGENIERÍA DE SISTEMAS

DISERTACIÓN DE GRADO PREVIA A LA OBTENCIÓN DEL TÍTULO DE INGENIERO DE SISTEMAS

"Sistema de Control de Estaciones de Trabajo en la Red de la Escuela de Ingeniería de Sistemas de la PUCESA"

DIRECTOR:

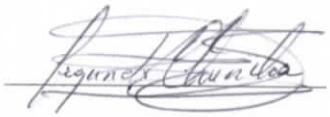


Ing. Janio Jadán

REVISORES:



Ing. Natasha Bayas



Ing. Víctor Chuncha

Franklin Vicente Guevara Moncayo

Ofelia Alexandra López Camino

Ambato, 2002

DEDICATORIA

Dedicado especialmente para mis padres Olmedo Guevara, Matilde Moncayo y hermanos Alexandra, Lino, Maricela, Walter quienes me apoyaron para culminar con éxito una de mis metas propuestas, quizás la más importante en la vida de una persona como es su educación.

Gracias por confiar en mi, nunca les fallaré.

Franklin

DEDICATORIA

A Dios creador de todo lo visible e invisible, que dio luz a mi inteligencia para poder seguir y culminar una carrera.

Dedicado a mis hermanos Fernando y Bolívar. A mis padres Jaime López y Blanca Camino porque con su buen ejemplo, paciencia y consejo oportuno supieron guiarme e inculcaron en mí un gran sentido de responsabilidad y ánimos de seguir adelante y por esta razón les dedico un acróstico denominado JAIME y BLANCA que dice así:

JAIME

Jamás te olvidaré padre querido,
Amigo fiel y padre bondadoso,
Iluminas mi vida con tu existencia,
Mi anhelo es siempre contar con tu presencia,
El gran creador de mis días.

BLANCA

Blanca luz que ilumina mi vida,
Lucero que brilla en la inmensidad,
Angel protector es mi madre querida,
Nada *ni* nadie podrá compararse a tí,
Carisma demuestras con intensidad,
Amistad por doquier a tú hija amada.

Ofelia

AGRADECIMIENTO

Quisiéramos dejar constancia y expresar nuestro profundo agradecimiento a todos los maestros que tuvimos a lo largo de nuestra vida estudiantil en la Pontificia Universidad Católica del Ecuador Sede Ambato, quienes con un sabio consejo supieron guiarnos por el sendero correcto y nos brindaron sus mejores conocimientos intelectuales.

De forma especial a nuestro director Ing. Janio Jadán y a nuestros revisores Ing. Víctor Chuncha e Ing. Natasha Bayas, quienes con el afán de vernos superados nos brindaron apoyo y han colaborado en el desarrollo de nuestra Tesis.

Franklin & Ofelia

ÍNDICE

CAPÍTULO I: ASPECTOS GENERALES

1.1. Introducción....1

1.2. Planteamiento del Problema....1

1.2.1. Descripción del Problema....1

1.2.2. Delimitación....2

1.3. Importancia y Justificación....3

1.4. Objetivos de la Investigación....4

1.4.1. Objetivo General....4

1.4.2. Objetivos Específicos....4

1.5. Hipótesis....5

1.6. Influencia del Software en la Evolución de la Entidad Educativa....5

CAPÍTULO II: REDES DE COMPUTADORES

2.1. Introducción....7

2.2. Redes....7

2.2.1. Arquitectura de una Red....10

2.2.1.1. Sistemas o Topologías de Red....10

2.2.1.2. Método de Acceso al Cable de una Red....15

2.2.1.3. Protocolos de Comunicación....16

2.3. Hardware de Red....16

2.3.1. Redes de Área Local (Local Area Network, LAN)....16

2.3.1.1. Razones para establecer una Red de Área Local (LAN)....17

2.3.1.2. Componentes de Hardware en una Red de Área Local (LAN) típica....19

2.3.1.3. Cableado Estructurado....22
2.3.2. Redes de Área Amplia (Wide Area Network, WAN)....28
2.3.2.1. Constitución de una Red de Área Amplia (WAN)....30
2.3.3. Intranets/Internet....31
2.3.3.1. Intranets....31
2.3.3.2. Internet....35
2.4. Software de Red....38
2.4.1. Protocolos de Comunicación....38
2.4.1.1. Protocolo TCP/IP (Transmission Control Protocol / Internet Protocol Protocolo de Control de Transmisiones/Protocolo Internet)....39
2.4.1.2. Protocolo NetBEUI....41
2.4.2. Interfaces y Servicios....43
2.4.2.1. Interfaces....43
2.4.2.2. Servicios....53
2.5. Modelos de Referencia....59
2.5.1. Modelo de Referencia OSI (Open System Interconnection – Interconexión de Sistemas Abiertos)....60
2.5.1.1. Capas o Niveles del Modelo de Referencia OSI....60
2.5.1.2. Funcionamiento del Modelo de Referencia OSI....62
2.5.2. Modelo de Referencia TCP/IP....63
2.5.2.1. Capas o Niveles del Modelo de Referencia TCP/IP....63
2.5.2.2. Funcionamiento del Modelo de Referencia TCP/IP....68
2.5.3. Estructura Cliente/Servidor....68

2.5.3.1. Sockets....68
2.5.3.2. Modelo Cliente-Servidor....70
2.6. Red de la Escuela de Ingeniería en Sistemas de la PUCESA....75
2.6.1. Información de la Red de la Escuela de Ingeniería en Sistemas de la PUCESA....75
2.6.1.1. Tipo de red....75
2.6.1.2. Topología de la Red....76
2.6.1.3. Componentes del Hardware de la Red de la Escuela de Ingeniería en Sistemas de la PUCESA....76
2.6.1.4. Componentes del Cableado Estructurado....80
2.6.2. Gráficos de la Red de la Escuela de Ingenieria en Sistemas de la PUCESA....81
CAPÍTULO III: AGENTES DE CAPTURA DE INFORMACIÓN
3.1. Introducción....89
3.2. ¿Qué son programas agentes?....89
3.2.1. Agentes....89
3.2.2. Corrientes que dominan la Investigación sobre Agentes....90
3.3. ¿Cómo capturar información?....93
3.4. Seguridades en programas agentes....95
CAPÍTULO IV:BITÁCORAS O ARCHIVOS DE AUDITORÍA(AUDIT LOGS)
4.1. Introducción....96
4.2. Administración de Bitácoras....96
4.2.1. Bitácoras....96
4.3. Propósitos, Tipos y Formato en el que se presentan los Audit Logs....97
4.3.1. Propósitos....97

PONTIFICIA UNIVERSIDAD CATOLICA DEL ECUADOR
PROGRAMAS ACADEMICOS AMBATO
BIBLIOTECA

4.3.2. Tipos.....98

4.3.3. Formato.....99

CAPÍTULO V: SOFTWARE DE DESARROLLO

5.1. Introducción.....100

5.2. Sistemas Operativos.....100

5.2.1. Linux.....100

5.2.1.1. Características del Sistema.....102

5.2.2. Windows.....104

5.2.2.1. Administrador de Programas.....105

5.2.2.2. Administrador de Archivos.....106

5.2.2.3. Windows 98.....107

5.3. Lenguaje de programación 109

5.3.1. Delphi.....110

5.3.1.1. Características de Delphi.....111

5.3.1.2. Conceptos Fundamentales.....111

5.3.2. Kylix.....115

5.3.2.1. El Compilador y la RTL.....116

5.4. Base de Datos 119

5.4.1. Base de Datos Relacional.....119

5.4.1.1. Sistema de Administración de Base de Datos Relacionales..119

5.4.2. MySQL.....120

5.4.2.1. Historia.....122

5.4.2.2. Características Técnicas de MYSQL.....124

5.4.2.3. Características principales de MYSQL.....124

5.4.2.4. Sistemas Operativos soportados por MYSQL.....125

5.4.2.5. Conectándose y desconectándose al Servidor MYSQL.....126
5.4.2.6. Visualización de las Bases de Datos existentes en el Servidor MYSQL.....127
5.4.2.7. Selección de una Base de Datos.....128
5.4.2.8. Creación de una Base de Datos.....128
5.4.2.9. Creación de Tablas.....129
5.4.2.10. Ingreso de Datos a las Tablas.....130
5.4.2.11. Recuperación de la Información de una Tabla.....131

CAPÍTULO VI: SEGURIDAD DE LA INFORMACIÓN

6.1. Introducción.....132
6.2. Seguridad de la Información.....132
6.2.1. Seguridad.....132
6.2.2. Seguridad de Datos.....135
6.3. Medidas de Seguridad.....136
6.3.1. Seguridad Legal.....137
6.3.2. Seguridad Física.....138
6.3.3. Seguridad Lógica.....139
6.3.3.1. Características de la Seguridad Lógica de la Información....141

CAPÍTULO VII: METODOLOGÍA DE DESARROLLO DEL SOFTWARE

7.1. Introducción.....144
7.2. Análisis Orientado a Objetos.....145
7.2.1. Conceptos y Principios Orientados a Objetos.....145
7.2.1.1. Identificación de Clases y Objetos.....146
7.2.1.2. Especificación de Atributos.....151
7.2.1.3. Definición de las Operaciones.....153

7.2.1.4. Comunicación entre Objetos.....155
7.2.2. Modelo Conceptual.....157
7.2.2.1. Diagramas de Clase.....158
7.2.2.2. Diagramas de Interacción (Diagramas de Secuencia).....160
7.3. Diseño.....164
7.3.1. Diseño de la Base de Datos.....165
7.3.1.1. Diseño Lógico.....166
7.3.1.2. Diseño Físico.....169
7.3.2. Diseño Orientado a Objetos.....171
7.3.2.1. Diagramas de Interacción (Diagramas de Colaboración).....171
7.3.2.2. Definir como se hace cada operación del sistema.....174
7.4. Programación OO.....180
7.4.1. Código Fuente.....180
Conclusiones.....181
Recomendaciones.....184
Bibliografía.....186
Anexo A – Manual del Usuario.....189
Anexo B – Manual Técnico.....209
Glosario.....224

LISTADO DE FIGURAS

Figura 2.1.: Red de Computadoras....8

Figura 2.2.: Red de Internet....36

Figura 2.3.: Configuración de la Red Linux Windows....44

Figura 2.4.: Configuración de la Red Linux – Máquina 1....45

Figura 2.5.: Configuración de la Red Linux Máquina 2....46

Figura 2.6.: Configuración del Hostname del Servidor Linux – Names....47

Figura 2.7.: Configuración del Hostname del Servidor Linux – Hosts....48

Figura 2.8.: Configuración del Hostname del Servidor Linux – Interfaces....48

Figura 2.9.: Configuración del Hostname del Servidor Linux Routing....49

Figura 2.10.: Configuración del Cliente Windows – Red (Configuración)....51

Figura 2.11.: Configuración del Cliente Windows – Red (Identificación)....52

Figura 2.12.: Servidores Concurrentes – Comunicación Orientada a Conexión....73

Figura 2.13.: Servidores Concurrentes – Comunicación mediante Datagramas....74

Figura 2.14.: Switch Central....82

Figura 2.15.: Hub IBM....84

Figura 2.16.: Hub Laboratorista....85

Figura 2.17.: Hub COMPAQ....86

Figura 2.18.: Hub Dirección....88

Figura 4.1.: El encabezado y la información de los campos de la Tabla de movimientos(mvitacora) – formato de la bitácora Primera Parte....99

Figura 4.2.: El encabezado y la información de los campos de la Tabla de movimientos(mvitacora) – formato de la bitácora Segunda Parte....99

Figura 6.1.: Medidas de Seguridad....136

Figura 7.1.: Clases.....147
Figura 7.2.: Atributos para la Clase Usuario.....152
Figura 7.3.: Atributos para la Clase Intervalo.....152
Figura 7.4.: Atributos para la Clase Estación.....152
Figura 7.5.: Atributos para la Clase Bitácoras.....152
Figura 7.6.: Atributos para la Clase Aplicación.....153
Figura 7.7.: Operaciones para la Clase Usuario.....154
Figura 7.8.: Operaciones para la Clase Intervalo.....154
Figura 7.9.: Operaciones para la Clase Estación.....154
Figura 7.10.: Operaciones para la Clase Bitácoras.....154
Figura 7.11.: Operaciones para la Clase Aplicación.....155
Figura 7.12: Paso de Mensajes entre Objetos.....156
Figura 7.13.: Paso de Mensajes.....157
Figura 7.14.: Diagrama de Clases.....159
Figura 7.15.: Diagrama de Secuencia del escenario de registro del usuario en el programa servidor de agentes.....161
Figura 7.16.: Diagrama de Secuencia del escenario de registro del intervalo de tiempo en el programa servidor de agentes.....161
Figura 7.17.: Diagrama de Secuencia del escenario para la verificación de la contraseña de desactivación del programa agente.....162
Figura 7.18.: Diagrama de Secuencia del escenario para la verificación del intervalo de tiempo con el que el programa agente envía información a la base de datos agentes.....162
Figura 7.19.: Diagrama de Secuencia del escenario de registro de la estación en el programa agente.....163

Figura 7.20.: Diagrama de Secuencia del escenario de registro de bitácoras en el programa agente.....163
Figura 7.21.: Diagrama de Secuencia del escenario de registro de la aplicación en el programa agente.....164
Figura 7.22.: Diseño Lógico de la Base de Datos.....170
Figura 7.23.: Diseño Físico de la Base de Datos.....170
Figura 7.24.: Diagrama de Colaboración del escenario de registro del usuario en el programa servidor de agentes.....172
Figura 7.25.: Diagrama de Colaboración del escenario de registro del intervalo de tiempo en el programa servidor de agentes.....172
Figura 7.26.: Diagrama de Colaboración del escenario para la verificación de la contraseña de desactivación del programa agente.....172
Figura 7.27.: Diagrama de Colaboración del escenario para la verificación del intervalo de tiempo con el que el programa agente envía información a la base de datos agentes.....173
Figura 7.28: Diagrama de Colaboración del escenario de registro de la estación en el programa agente.....173
Figura 7.29.: Diagrama de Colaboración del escenario de registro de bitácoras en el programa agente.....173
Figura 7.30.: Diagrama de Colaboración del escenario de registro de la aplicación en el programa agente.....174
Figura A.1.: Pantalla Principal del Sistema Servidor de Agentes.....189
Figura A.2.: Botón Salir para cerrar el Sistema Servidor de Agentes.....190
Figura A.3.: Menú Mantenimiento Eliminación.....191
Figura A.4.: Menú Mantenimiento Parámetros Usuarios - Altas Bajas.....192

Figura A.5.: Menú Mantenimiento – Parámetros – Usuarios – Consulta.....192

Figura A.6.: Menú Mantenimiento – Parámetros – Intervalo – Altas Bajas.....194

Figura A.7.: Menú Mantenimiento – Parámetros – Intervalo – Consulta.....194

Figura A.8.: Menú Útiles – Mensajes.....195

Figura A.9.: Menú Útiles – Mensajes – Botón Apagar PC.....195

Figura A.10. Menú Útiles – Estadísticas.....196

Figura A.11. Menú Útiles – Estadísticas – Botón Estaciones.....197

Figura A.12. Menú Útiles – Estadísticas – Botón Aplicaciones.....198

Figura A.13. Menú Útiles – Reportes.....200

Figura A.14. Menú Útiles – Reportes – Por Dirección IP.....201

Figura A.15. Menú Útiles – Reportes – Por Dirección y Rango de Fechas.....202

Figura A.16. Menú Útiles – Reportes – Por Dirección, Fecha y Rango de Horas.....203

Figura A.17. Menú Útiles – Colores.....204

Figura A.18. Menú Ayuda – Ayuda del Software.....205

Figura A.19. Menú Ayuda – Acerca.....205

Figura A.20. Enviar Programa Agente al Registro de Windows.....206

Figura A.21. Bandeja de Entrada – Plataforma Windows.....207

Figura A.22. Pantalla de Contraseña de desactivación del Sistema Agente.....207

Figura A.23. Escritorio – Plataforma Linux.....208

LISTADO DE TABLAS

Tabla 2.1.: Ventajas y Desventajas de la Topología Anillo/Bus....13

Tabla 2.2.: Ventajas y Desventajas de la Topología Estrella....14

Tabla 2.3.: Ventajas y Desventajas de la Topología de Árbol....14

Tabla 2.4.: Ventajas y Desventajas del Protocolo NetBEUI....43

Tabla 2.5.: Opciones del Comando Ping....57

Tabla 2.6.: Hub 24 puertos....79

Tabla 2.7.: Hub 16 puertos....80

Tabla 2.8.: Switch Central 24 puertos....80

Tabla 2.9.: Switch Central....81

Tabla 2.10.: Hub IBM....83

Tabla 2.11.: Hub Laboratorista....85

Tabla 2.12.: Hub COMPAQ....87

Tabla 2.13.: Hub Dirección....88

Tabla 7.1.: Identificación de Clases y Objetos....149

Tabla 7.2.: Características Selectivas de la Identificación de Clases....151

PREFACIO

Hoy en dia la informática se ha constituido en una herramienta de trabajo en muchas de nuestras actividades diarias. El uso de tecnologia y la aplicación de la informática a nivel empresarial en nuestro país, es inminente. El uso de software ofimático y de gestión ayudan a la automatización de procesos de una entidad, que a su vez ayudan a brindar un mejor servicio o simplemente agilitar tareas que anteriormente se hacian en forma manual. Pese a esta automatización en la Escuela de Ingenieria de Sistemas de la PUCESA, no existe una real utilización de estas herramientas para agilitar el trabajo diario.

El problema anterior se debe principalmente a la falta de conciencia en muchos de los empleados o estudiantes, conciencia necesaria para el desarrollo de la entidad. Frente a este problema se han desarrollado algunos mecanismos de control de personal, tales como, control por parte de profesores o administradores de la red, etc. Pese a éstos controles existen lamentablemente aún muchos usuarios de computadoras que mal utilizan estos recursos, por ejemplo, ejecutan juegos de computadora, ingresan a páginas restringidas en Internet, realizan tareas personales o simplemente no hacen nada, en otras palabras alimentan al gran mounstruo de la burocracia, que dia a dia nos agobia.

Pero, ¿Cómo saber lo que hace cada usuario?. Si se podría de alguna manera controlar las actividades de los usuarios de una determinada estación y obtener estadísticas resumidas de sus actividades, un administrador del centro de cómputo o un directivo de mandos medios podria hacer una toma de decisiones y redistribuir o cambiar los recursos humanos o técnicos para hacer el trabajo más eficiente.

En el sector educativo también existe la mala utilización de los equipos, de los laboratorios y centros de cómputo, por lo que se ve necesario recurrir a herramientas automatizadas de control que detecten problemas en el desempeño de una red, así como de los equipos de cómputo; y proporcionen al administrador información que permita verificar el estado general o particular de la red y de igual manera el estado de los equipos y en caso de presentarse problemas en alguno de ellos se notificará al administrador responsable del equipo.

Con objeto de asistir a la administración de la red, la propuesta del presente proyecto es la de abarcar el Control de las Estaciones de Trabajo de la Escuela de Ingeniería de Sistemas de la PUCESA, con el desarrollo de un software, el cual estará integrado como una aplicación a la Red el mismo que proporcionará al administrador información de las actividades de los usuarios en las estaciones de trabajo.

Este sistema deberá identificar las aplicaciones utilizadas e identificar el tipo de información obtenida en Internet. Para evitar que el sistema sea una carga al administrador y más bien provea ayuda de control, simplificando la interacción con el administrador de la red, éste debe tener una interfaz amigable y debe tener la opción de generar avisos y reportes estadísticos.

Específicamente la finalidad del sistema es ayudar y proveer a los administradores de sistemas de un mecanismo que les permita ejecutar tareas de control de manera automática, que puedan servir para la toma de decisiones en lo que respecta a la asignación y redistribución de recursos.

En la actualidad existe una amplia variedad de controladores de red usados en la fabricación de tarjetas de red Fast Ethernet como son de National, DEC, AMD, Intel, Fujitsu, Realtek, así como programas de monitoreo de propósito general, tales como, Netmonitor, Microsoft Network Monitor y otros. Pero muchos de ellos no se ajustan a nuestras necesidades en nuestro medio, por tal motivo pretendemos desarrollar nuestra propia aplicación basándonos en las existentes.

Por otro lado, no todos los sistemas de monitoreo fueron hechos con fines favorables, por mencionar algunos, Back Orifice y Netbus, son dos de los principales programas para el control remoto de computadoras o sistemas de monitoreo, los cuales permiten controlar una máquina mediante un enlace a través de un medio de comunicación, ya sea en una Red de Área Local o en "INTERNET", los cuales pueden ser mal utilizados por piratas informáticos.

CAPÍTULO I

ASPECTOS GENERALES

1.1. INTRODUCCIÓN

El propósito del presente capítulo es permitir que cualquier persona pueda acceder a la información, que constituye una explicación detallada de los aspectos generales más importantes.

En este capítulo se encontrará el planteamiento del problema, así como su descripción y su delimitación, la importancia y su justificación, los objetivos de la investigación, así como sus objetivos generales y sus objetivos específicos, la hipótesis y la influencia del software en la evolución de la entidad educativa en nuestro caso el objeto de estudio que es la Pontificia Universidad Católica del Ecuador Sede Ambato.

1.2. PLANTEAMIENTO DEL PROBLEMA

1.2.1. Descripción del Problema

El problema que estamos tratando de resolver es:

Actualmente los laboratorios de la Escuela de Ingeniería de Sistemas de la PUCESA posee un software de monitoreo y administración en forma remota. Con este software se puede controlar las estaciones de trabajo en línea, pero esto requiere que el

administrador o alguno de sus ayudantes esté monitoreando todo el tiempo o por lo menos cada cierto tiempo.

Como sub-problemas del software de monitoreo E/POP que posee actualmente la Escuela de Ingeniería de Sistemas de la PUCESA se detallan a continuación:

- Únicamente trabaja bajo la plataforma Windows.
- El E/POP puede ser descargado de Internet, una versión de prueba que caduca a los 30 días. Para evitar esto se debe comprar la licencia, costo que representa una carga para la PUCESA.
- No posee un registro histórico de información de cada estación de trabajo.

1.2.2. Delimitación

Se propone como tema de disertación el desarrollo de un Sistema de Control de Estaciones de Trabajo en la Red de la Escuela de Ingeniería de Sistemas de la PUCESA mediante programas agentes, para analizar el comportamiento de usuarios en estaciones de trabajo, que sea multiplataforma con el fin de que se pueda monitorear estaciones en la Plataforma Windows y Linux, es decir proporcionaremos un software que brinde información al administrador de red en el momento que requiera.

Se lo delimita:

- Investigación y desarrollo de captura de actividades o procesos por medio de bitácoras, ya que en ellas se registran los eventos que ocurren en el sistema operativo, es decir eventos que el administrador pasaría inadvertidos sin el respaldo de las bitácoras.

- Crear una Base de Datos adecuada que permita almacenar y extraer información que servirá para obtener los informes estadísticos.
- Controlar el personal en los terminales de la Escuela de Ingeniería de Sistemas de la PUCESA.
- Obtener informes estadísticos adecuados para la correcta toma de decisiones para el uso de los recursos de la PUCESA.
- Aplicar seguridades a la información capturada de las actividades realizadas y al software desarrollado en las estaciones de trabajo de la Red de la Escuela de Ingeniería de Sistemas de la PUCESA.

Al finalizar el desarrollo del trabajo, el administrador de la Red de la PUCESA tendrá la opción de realizar un mejor control de la red con eficiencia y en menor tiempo, gracias a las oportunidades que brinda la informática.

1.3. IMPORTANCIA Y JUSTIFICACIÓN

Para evitar constantes pérdidas de recursos, los sistemas de monitoreo de redes son extremadamente importantes para la correcta administración de redes. A medida que avanza la tecnología nuestros recursos quedan vulnerables a cualquier intento de infracciones o mal uso.

Nuestra propuesta se justifica a fin de que el administrador de la Red de la Escuela de Ingeniería de Sistemas de la PUCESA pueda controlar y obtener informes estadísticos de las distintas estaciones de trabajo en tiempos **reales**, estableciendo una alternativa para potenciar el nivel académico, evitando de esta **manera** que tanto empleados como

estudiantes malgasten su tiempo en acceder a programas no permitidos, o tareas fuera del ámbito académico.

El software de Control de Estaciones de Trabajo en la Red de la Escuela de Ingeniería de Sistemas de la PUCESA, así como las seguridades que se aplicarán en este proyecto será el inicio de nuevos controles que están llamados a crecer y consolidarse en aras de la tecnología, para que nuestra entidad educativa abra las puertas al resto de entidades brindando conocimiento del poderoso alcance que se puede tener mediante esta herramienta.

1.4. OBJETIVOS DE LA INVESTIGACIÓN

1.4.1. Objetivo General

Desarrollar un software de Control de Estaciones de Trabajo en la Red de la Escuela de Ingeniería de Sistemas de la PUCESA utilizando programas agentes, en multiplataforma.

1.4.2. Objetivos Específicos

- Investigar la mejor alternativa para el almacenamiento de la información capturada por los agentes.
- Analizar las diferentes seguridades que se deben aplicar para mantener la información bajo el control del administrador de Red de la Escuela de Ingeniería de Sistemas de la PUCESA.

- Obtener reportes estadísticos por tiempo y recursos de los sitios Web, estaciones de trabajo, programas, tareas en días u horas de mayor uso de los recursos.

1.5. HIPÓTESIS

La realización de un Sistema de Control de Estaciones de Trabajo en la Red de la Escuela de Ingeniería de Sistemas de la PUCESA mediante programas agentes para el envío de información al servidor de la Base de Datos, proporcionará una gran ayuda en la obtención de resultados confiables e información y al tiempo requerido, permitiendo de esta manera tomar decisiones al administrador de la red, para mejorar el uso de recursos en las estaciones de trabajo.

1.6. INFLUENCIA DEL SOFTWARE EN LA EVOLUCIÓN DE LA ENTIDAD EDUCATIVA

El Software en desarrollo en la evolución de la Entidad Educativa influye en los siguientes aspectos:

- Creará un amplio sentido de responsabilidad tanto en empleados como estudiantes en no malgastar su tiempo en acceder a programas no permitidos o tareas fuera del ámbito académico y de esta manera el administrador de la Red de la Escuela de Ingeniería de Sistemas de la PUCESA podrá controlar las distintas estaciones de trabajo basándose en las bitácoras; estableciendo una alternativa para potenciar el nivel académico.

- Con el control de las estaciones de trabajo se podrá evitar constantes pérdidas de recursos. Ya que a medida que avanza la tecnología nuestros recursos quedan vulnerables a cualquier intento de infracciones o mal uso.

CAPÍTULO II

REDES DE COMPUTADORES

2.1. INTRODUCCIÓN

El propósito del presente capítulo es permitir que cualquier persona pueda acceder a la información, que constituye una explicación detallada de los conceptos más importantes en los que se fundamenta la comunicación entre ordenadores, tanto desde un punto de vista teórico como práctico.

En este capítulo se encontrará conceptos sobre redes, así como la arquitectura de una red, hardware de red, así como las Redes de Área Local (LAN), Redes de Área Amplia (WAN) y Intranets/Internet, software de red, así como los protocolos de comunicación, interfaces y servicios, modelos de referencia, así como el modelo OSI, el modelo TCP/IP y la estructura cliente/servidor, además información relacionada con la Red de la Escuela de Ingeniería de Sistemas de la PUCESA.

2.2. REDES

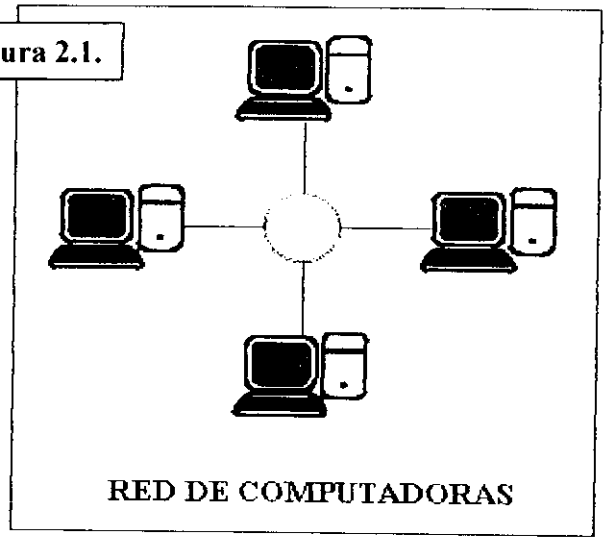
Una red de computadoras es un conjunto de conexiones físicas y programas de protocolo, empleados para conectar dos o más computadoras. En realidad la red la componen el hardware y el software.

El hardware lo constituyen los cables tales como cables par trenzados UTP, cables FTP, fibra óptica y adaptadores de red que permiten la comunicación entre dos o más computadoras, por medio de transmisión de información en códigos binarios.

El conjunto de software utilizado para la transmisión de datos de las conexiones físicas se denominan protocolos de comunicación, éstos permiten la sincronización en la transferencia de datos por las redes hasta llegar hasta su destino.

Un gráfico de **Red de Computadoras** se muestra a continuación en la **(Figura 2.1.)**.

Figura 2.1.



Desde el punto de vista de su alcance existen las redes que tienen gran extensión y las máquinas se pueden encontrar a grandes distancias unas de otras (Wide Area Network, WAN) y las redes donde las máquinas conectadas están relativamente cerca o redes locales (Local Area Network, LAN).

Las redes Ethernet son **LAN**s (Local Area Networks) que mediante interconexiones a través de **bridges** (puentes), **routers** (enrutadores) y **gateways** (pasarelas) forman lo

que se conoce como **WAN**s (Wide Area Networks). La necesidad de intercambiar información entre todas estas LANs y WANs genera un esquema de direccionamiento y enrutado estándar para todas ellas: **Internet** agrupa al mayor número de redes actualmente (no todas son Ethernet) y todas ellas utilizan el protocolo de comunicación **TCP/IP.**

Las redes que hoy efectivamente, manejan cualquier tipo de información, han evolucionado mucho gracias a los importantes desarrollos realizados en el campo de las comunicaciones digitales, tanto de voz como de datos. Desde las primeras centrales telefónicas analógicas y computadoras centralizadas, muy limitadas en sus facilidades de comunicaciones, y por tanto, con muy bajo impacto en las empresas, hasta los actuales Sistemas de Información que proporcionan la competitividad necesaria para el desarrollo del negocio.

Hoy, las redes incorporan voz, datos, texto e imagen, y necesitan de flexibilidad, disponibilidad y calidad, además de una alta relación costo/servicio.

En los últimos 10 años, se han explotado nuevos productos, tecnologías y servicios dando lugar a una demanda de mayor ancho de banda, capaz de soportar una red global de comunicaciones.

Partiendo de cubrir la necesidad básica de cualquier empresa la comunicación de voz, pasando por las tradicionales de transmisión de datos, hasta llegar a las más sofisticadas, como puede ser el establecimiento de comunicaciones gráficas o de video (que demandan mayor ancho de banda), vemos que en todos los casos los medios han ido

evolucionando para satisfacerlas, con la incorporación de nuevos servicios y tecnologías.

2.2.1. Arquitectura de una Red

La arquitectura de una red viene definida por:

- Su Topología.
- El Método de Acceso a la Red.
- Los Protocolos de Comunicación.

2.2.1.1. Sistemas o Topologías de Red

Los nodos de red (las computadoras) necesitan estar conectados para comunicarse. A la forma en que están conectados los nodos se le llama *topología*.

Topologías de Red Estándar

Las tres topologías de red estándar son:

- Topología Bus.
- Topología Estrella.
- Topología Anillo.

Topología Bus.- En una topología de bus, cada computadora está conectada a un segmento común de cable de red. El segmento de red se coloca como un bus lineal, es decir, un cable largo que va de un extremo a otro de la red, y al cual se conecta cada

nodo de la red. El cable puede ir por el piso, por las paredes, por el techo, o puede ser una combinación de éstos, siempre y cuando el cable sea un segmento continuo. El servidor y todas las estaciones están conectadas a un cable general central. Todos los nodos comparten este cable y éste necesita acopladores en ambos extremos.

Cada nodo verifica las direcciones de los paquetes que circulan por la red para ver si alguna coincide con la suya propia. El cable puede extenderse de cualquier forma por las paredes y techos de la instalación. Las señales y los datos van y vienen por el cable, asociados a una dirección destino.

La topología bus usa una cantidad mínima de cable y el cable es muy fácil de instalar, ya que puede extenderse por un edificio en las mejores rutas posibles. Así el cable debe ir de equipo en equipo.

Topología Estrella.- Se utiliza un dispositivo como punto de conexión de todos los cables que parten de las estaciones de trabajo. El dispositivo central puede ser el servidor de archivos en sí o un dispositivo especial de conexión como es un concentrador (HUB).

El diagnóstico de problemas es fácil, debido a que las estaciones de trabajo se comunican a través del equipo central. Los fallos en el nodo central son fáciles de detectar y es fácil cambiar los cables. La colisión entre datos es imposible, ya que cada estación tiene su propio cable y resulta fácil ampliar el sistema.

En algunas empresas tienden a agruparse los cables en la unidad central lo cual puede ocasionar errores de gestión.

Topología Anillo.- En una topología de anillo, cada computadora se conecta en forma de anillo a la red. Las topologías de anillo casi siempre son lógicas con topología física de estrella.

Las señales viajan en una única dirección a lo largo del cable en forma de un bucle cerrado. En cada momento, cada nodo pasa las señales a otro nodo.

Con la topología en anillo, las redes pueden extenderse a menudo a largas distancias, y el coste total del cableado será menor que en una configuración en estrella y casi igual a la de bus. Una rotura del cable hará caer el sistema. Actualmente existen sistemas alternativos que evitan que esto ocurra.

Combinaciones de Topologías

También hay combinaciones de más de una topología como son:

- Topología Estrella/Bus o Topología de Árbol.
- Topología Estrella/Anillo.

Topología Estrella/Bus o Topología de Árbol.- Es la combinación de las topologías de Estrella/Bus. Muchos concentradores de las redes Ethernet con topología física de estrella también tienen un conector en la parte trasera que enlaza al concentrador a una red de topología de bus.

Topología Estrella/Anillo.- Físicamente parece una topología estrella pero el tipo de concentrador utilizado, la MAU se encarga de interconectar internamente la red en forma de anillo.

Existe un conector central. Las estaciones de trabajo se extienden a partir de este conector para incrementar las distancias permitidas.

Comparación de las Ventajas y Desventajas de las Topologías

En la **(Tabla 2.1. Ventajas y Desventajas de la Topología Anillo/Bus)** que se muestra a continuación.

(Tabla 2.1. Ventajas y Desventajas de la Topología Anillo/Bus)

Ventajas	Desventajas
• Costos de materiales iniciales más bajos. • Costos de mano de obra iniciales más bajos. • Flexibilidad en aplicaciones específicas. • Facilidad en bus de añadir y quitar estaciones.	• Un fallo en el anillo inhabilitaría a todas las estaciones. • En Bus al fallar un tramo del bus, varias estaciones quedan desconectadas a la red.

En la **(Tabla 2.2. Ventajas y Desventajas de la Topología Estrella)** que se muestra a continuación.

(Tabla 2.2. Ventajas y Desventajas de la Topología Estrella)

Ventajas	Desventajas
• Mejor acceso a la red. • Mejor confiabilidad. • Flexibilidad para adicionar, mover o cambiar.	• Costos de materiales iniciales más altos. • Costos iniciales de mano de obra más altos.

En la **(Tabla 2.3. Ventajas y Desventajas de la Topología de Árbol)** que se muestra a continuación.

(Tabla 2.3. Ventajas y Desventajas de la Topología de Árbol)

Ventajas	Desventajas
• Cableado punto a punto para segmentos individuales. • Soportado por multitud de vendedores de software y de hardware.	• La medida de cada segmento viene determinada por el tipo de cable utilizado. • Si se viene abajo el segmento principal todo el segmento se viene abajo con él. • Es más difícil su configuración.

2.2.1.2. Método de Acceso al Cable de una Red

El método de acceso al cable de una red describe como una estación de trabajo establece el acceso a medios compartidos y así puede transmitir información.

Cuando la tarjeta de la interfaz de red obtiene el acceso al cable, comienza el envío de paquetes de información en un formato de trama como flujos de bits sobre la red.

Los sistemas de cable lineales utilizan el método de Acceso Múltiple con Detección de Portadora/Detección de Colisión (CSMA/CD, Carrier Sense Multiple Access/Collision Detection), donde una estación de trabajo accede al cable pero se retira si otra estación de trabajo intenta el acceso al mismo tiempo.

Una estación de trabajo trasmite una señal y cada uno de los otros nodos de la red la oye, pero sólo el nodo direccionado presta atención. Si dos nodos trasmiten al mismo tiempo, se produce una colisión y ambos se retiran, esperan una cantidad aleatoria de tiempo y lo intentan de nuevo.

Las prestaciones se degradan cuando el tráfico de la red es pesado, debido a estas colisiones y a las retransmisiones.

El método de prioridad bajo demanda (demand on priority) utiliza un concentrador central que determina que estación puede acceder al cable y puede otorgar prioridades a unas estaciones sobre otras, dependiendo de la sensibilidad al tiempo de los datos que se quiere trasmitir.

2.2.1.3. Protocolos de Comunicación

Los protocolos de comunicación es un conjunto de reglas, para el intercambio de información de un proceso dentro de una red.

Lo que hemos expuesto aquí acerca de protocolos de comunicación es únicamente una introducción, ya que se hablará con más detenimiento en el punto de Software de Red del **Capítulo II** en el punto **2.4.1. Protocolos de Comunicación.**

2.3. HARDWARE DE RED

2.3.1. Redes de Área Local (Local Area Network, LAN)

Redes de Área Local (LAN) son redes que interconectan equipos dentro de un entorno físico reducido con el fin de compartir recursos y permitir la transmisión de datos entre ellos. En general no se extiende más allá de un edificio.

El propietario de una LAN no tiene que preocuparse de utilizar al máximo los canales, ya que son baratos en comparación con su capacidad de transmisión (los cuellos de botella en las LAN suelen estar en el SOFTWARE). Por tanto, no es tan crítica la necesidad de esquemas muy eficientes de multiplexado y multidistribución. Además, como las redes de área local que residen en un mismo edificio, la topología tiende a ser más ordenada y estructurada, con configuraciones en forma de bus, anillo o estrella.

2.3.1.1. Razones para establecer una Red de Área Local (LAN)

Una de las razones originales para establecer una red fue la necesidad de compartir recursos muy caros como los discos, las impresoras y los plotters. Aunque este aspecto ha perdido importancia porque ahora estos equipos no son generalmente tan caros, han surgido otros factores con alguna importancia.

Trabajo en Grupo.- El trabajo en redes ayuda al trabajo en grupos que posean información común a compartir. Linux y Windows permiten el trabajo en grupo, pues puede crearse un grupo y a éste se pueden asignar cualquier número de miembros, después los derechos y facilidades de trabajo pueden asignarse al grupo en lugar de a los usuarios uno a uno. Un grupo también puede ser usado para enviarle mensajes a todos sus miembros a la vez.

Actualización de Software.- La actualización de software es mucho más fácil si un paquete se almacena centralmente en el servidor, en lugar de actualizar individualmente cada información de los usuarios uno a uno.

Disponibilidad de Software para Redes.- Una buena selección de la calidad del software multiusuario que pueda satisfacer la necesidad del usuario, puede ser una razón importante para seleccionar una red local. Puede darse la situación que al conocer todas las posibilidades que brinda un sistema multiusuario, el diseñador comprenda que es ese producto el que necesita realmente.

Beneficios en la Manipulación y Salva de los Datos.- Como los datos se encuentran centralizados en el servidor, es mucho más fácil manipularlos, así como realizar la salva de los mismos. Los ficheros pueden ser transferidos entre los usuarios dentro de la red lo que es mejor que la transferencia usando floppies.

Beneficios en la Organización.- Las facilidades en la manipulación de los datos puede mejorar la organización de la información. Por ejemplo los miembros de un grupo no tienen necesariamente que sentarse cerca unos de otros para compartir los ficheros.

Compartir Impresoras de Alta Calidad.- El compartir recursos siempre ha sido uno de los principales beneficios de una red local y mientras unos equipos no son tan caros, otros si lo son, tanto como para justificar la necesidad de una red local; por ejemplo: impresoras de láser de alta calidad y plotters sofisticados.

Procesamientos Distribuidos.- En muchas compañías ya existen los llamados procesos distribuidos. Si en una oficina hay muchas microcomputadoras instaladas, esas máquinas representan la base para la instalación de una red. Muchas veces la decisión de instalar una red se toma al contar con las microcomputadoras suficientes para la red y comprender la necesidad de compartir la información lo que puede dar lugar a la creación de un sistema distribuido de la información, que incluya el uso de correo electrónico.

Extender el Uso de la Red a través de Estaciones baratas.- Cuando una red ya está establecida, el costo para añadir nuevas estaciones es muy bajo. Estas estaciones,

pueden ser PC sin discos duros e incluso sin torres de floppies. Estos sistemas pueden ser de bajo costo y de poderosas características.

Seguridad.- La seguridad en el almacenamiento de datos y programas usando el bloqueo físico de equipos y el bloqueo de software es un factor que se debe tener en cuenta a la hora de decidir establecer una red.

2.3.1.2. Componentes de Hardware en una Red de Área Local (LAN) típica

Los cinco componentes básicos del hardware de una red son:

- Los Servidores.
- Las Estaciones de Trabajo.
- Tarjetas Interfaces de la Red (Network Interface Cards, NIC).
- Los Cables (incluyendo los conectores y otros componentes especiales).
- Concentradores (Hubs).

El Servidor

El servidor es una computadora principal, que se encarga de administrar los recursos de la red y el flujo de información utilizando el sistema operativo de red que se carga en el disco del servidor junto con los utilitarios y demás herramientas necesarias para el trabajo en red.

Muchos de los servidores son "dedicados", es decir, están realizando tareas específicas, por ejemplo:

- Servidor de impresión, sólo para imprimir.
- Servidor de comunicaciones, sólo para controlar el flujo de los datos, etc.

Para que una máquina sea un servidor, es necesario que sea una computadora de alto rendimiento en cuanto a velocidad y procesamiento y gran capacidad en disco duro u otros medios de almacenamiento.

Las Estaciones de Trabajo

Las estaciones de trabajo son conectadas al servidor a través de las tarjetas interfaces de la red y los cables. Las estaciones pueden ser microcomputadoras comunes (IBM o compatibles) con discos duros o sin ellos y con torres de floppy o sin ellas.

Sistemas Operativos para las Estaciones.- Independientemente que las estaciones puedan ser de un tipo u otro el sistema operativo debe ser consecuente con el tipo de red que se vaya a instalar. Una posibilidad de trabajo en red es que se quiera acceder a una base de datos y después pasar a una aplicación local que haga otro tipo de procesamiento. Esto puede hacerse con el empleo de sistemas multitareas como el OS/2, PC-MOS, Windows 95, Windows 98, Windows NT, etc.

Tarjetas Interfaces de la Red (NETWORK INTERFACE CARDS, NIC)

Estas tarjetas son las que permiten conectar el cableado entre servidores y estaciones de trabajo. En la actualidad existen numerosos tipos de placas que soportan distintos tipos de cables y topologías de red.

Las placas contienen los protocolos y órdenes necesarios para soportar el tipo de red al que está destinada. Muchas tienen memoria adicional para almacenar temporalmente los paquetes de datos enviados y recibidos, mejorando el rendimiento de la red.

La compatibilidad a nivel físico y lógico se convierte en una cuestión relevante cuando se considera el uso de cualquier placa de red. Hay que asegurarse que la placa pueda funcionar en la estación deseada y de que existen programas controladores que permitan al sistema operativo enlazarlo con sus protocolos y características a nivel físico.

Los Cables en la Red

Después de que todas las máquinas (servidores y estaciones) tengan instalada la tarjeta para conectar una con otra se usan los cables. El tipo de cable a utilizar depende de muchos factores que deben tenerse en cuenta a la hora de seleccionarlo. Lo más importante es saber que el cable que se seleccione es el adecuado para la red que se desee instalar y que éste sea compatible con la tarjeta interface.

Concentradores (Hub's)

Dispositivo que interconecta un computador (host) dentro de una red. Es el dispositivo de interconexión más simple que existe.

Sus principales características son:

- Se trata de un armario de conexiones donde se centralizan todas las conexiones de una red, es decir un dispositivo con muchos puertos de entrada y salida.
- No tiene ninguna función aparte de centralizar conexiones.

- Se suelen utilizar para implementar topología en estrella física, pero funcionando como un anillo o como un bus lógico.

Aunque el aspecto externo de un hub puede coincidir con el de un switch, y efectivamente ambos distribuyen señal entre segmentos de red, hay diferencias sustanciales entre ellos. La más significativa es que, mientras que en el hub el ancho de banda de la máquina es compartido por todos los puentes mediante una multiplexación en el tiempo (sólo una estación puede transmitir de un puerto a otro en cada instante), en el switch el ancho de banda está por encima del ancho de banda de cada uno de los puertos.

2.3.1.3. Cableado Estructurado

Cableado estructurado es tender cables de señal en un edificio de manera tal que cualquier servicio de voz, datos, video, audio, tráfico de Internet, seguridad, control y monitoreo este disponible desde y hacia cualquier roseta de conexión (Outlet) del edificio.

Un sistema de cableado estructurado es un diseño de arquitectura abierta ya que es independiente de la información que se trasmite a través de el. También es confiable porque está diseñado con una topología de estrella, la que en caso de un daño o desconexión, éstas se limitan sólo a la parte o sección dañada y no afecta al resto de la red.

Un sistema de cableado estructurado permite mover personal de un lugar a otro o agregar servicios a ser transportados por la red sin la necesidad de incurrir en altos costos de recableado.

Un sistema de cableado estructurado permite transmisión de datos de hasta 100 megabytes por segundo.

Un "sistema de cableado abierto", es un sistema de cableado estructurado que está diseñado para ser independiente del proveedor y de la aplicación a la vez.

Características del Cableado Estructurado

Las características claves de un sistema de cableado abierto son que todos las outlets (salidas para conexión) del área de trabajo son idénticamente conectados en estrella a algún punto de distribución central, usando una combinación de medio y hardware que puede aceptar cualquier necesidad de aplicación que pueda ocurrir a lo largo de la vida del cableado (10 años).

Estas características del sistema de cableado abierto ofrecen tres ventajas principales al dueño o usuario:

- Debido a que el sistema de cableado es independiente de la aplicación y del proveedor, los cambios en la red y en el equipamiento pueden realizarse por los mismos cables existentes.
- Debido a que los outlets están cableados de igual forma, los movimientos de personal pueden hacerse sin modificar la base de cableado.

- La localización de los hubs y concentradores de la red en un punto central de distribución, en general un closet de telecomunicaciones, permite que los problemas de cableado o de red sean detectados y aislados fácilmente sin tener que parar el resto de la red.

Tipos de Cable utilizados

Hay tres tipos de cable que pueden ser utilizados en un diseño de cableado estructurado y son:

- UTP (Unshielded Twisted Pair).
- FTP (Foil Screened Twisted Pair).
- Fibra Óptica.

El cableado estructurado UTP (Unshielded Twisted Pair par trenzado) en categoría 5 es el tipo de cableado más solicitado hoy en día.

Cable UTP.- Es el cable más usado y provee una infraestructura a través de la cual la mayoría de los productos pueden ser conectados. El diseño de un Sistema de cableado UTP tiene una configuración de estrella, todas las rosetas de conexión (outlets) están conectadas a un panel central de cables (Patch Panel Central) y los HUB's son utilizados para conectar a un servicio. Para la conexión entre el gabinete y la roseta de conexión el largo máximo aceptado es de 100 metros.

El cable consiste en 4 pares torcidos y existen 5 categorías siendo las tres más importantes (3, 4 y 5) utilizadas en transmisión de datos. El cable Categoría 5 soporta transmisión de datos hasta 100 Megabytes por segundo.

Las ventajas más importantes del cable UTP son:

- Soporta un amplio rango de sistemas y protocolos.
- Fácil reubicación de dispositivos.
- Bajo costo.

Cable FTP.- Es utilizado en aplicaciones en donde el ruido puede ser un problema. Cuando es instalado correctamente permite la utilización de cableado estructurado en un ambiente que anteriormente fue critico por ruidos en la red.

El cable FTP puede ofrecer un alto nivel de protección sin aumentar los costos significativamente. La instalación de cable FTP minimiza la sensibilidad en el diseño de la ruta (proximidad a emisores EMI) pero agrega complejidad desde el punto de vista de la calidad de las conexiones y conexión a tierra.

La clave para utilizar cable FTP es hacer el sistema compatible lo más posible con el diseño, la instalación y el mantenimiento UTP estándar. Esto minimizará el impacto de este medio en la disponibilidad del sistema. Un ejemplo es utilizar cable de 100 ohm

nominal, que es la intensidad que circula por el cable UTP, de tal manera que los equipos diseñados para UTP puedan trabajar bien con cable FTP. Esto permite por ejemplo, usar cable FTP en la fábrica y cable UTP en las oficinas cuando éstas se encuentran en el mismo edificio.

La incorrecta conexión a tierra puede resultar en un pobre rendimiento por lo que es conveniente que dicha conexión sea correctamente instalada en forma inicial y sea mantenida posteriormente.

Fibra Óptica.- Se utiliza principalmente para Servicios de Datos ya que su ancho de banda y alta velocidad es ideal para ese propósito.

Un filamento de vidrio sumamente delgado diseñado para la transmisión de la luz. Las fibras ópticas poseen capacidades de transmisión enormes, del orden de miles de millones de bits por segundo. Además, y a diferencia de los pulsos eléctricos, los impulsos luminosos no son afectados por interferencias causadas por la radiación aleatoria del ambiente.

Componentes del Cableado Estructurado

Entre los componentes más importantes del cableado estructurado tenemos los siguientes:

- Ruteadores (Routers).
- Switch.
- Patch Panel.

Ruteadores (Routers).- Son dispositivos electrónicos complejos que permiten manejar comunicaciones entre redes que se encuentran a gran distancia, utilizando vínculos provistos por las empresas prestatarias del servicio telefónico (líneas punto a punto), líneas de datos (ARPAC), enlaces vía satélite, etc.

Poseen avanzadas funciones de negociación del enlace y conversión de protocolos de transmisión. Se utilizan por lo general en empresas que manejan muchas sucursales, tales como Bancos y Supermercados. Están relacionados con sistemas bajo Unix y TCP-IP.

Switch.- El switch es un conmutador que tiene funciones de nivel 2 de OSI (Open System Interconnection) y que, por tanto, se parece a un puente (bridge) en cuanto a su funcionamiento.

Sin embargo, tiene algunas características que lo distinguen:

- El switch es siempre local.
- Conecta segmentos de red en lugar de redes, aunque en estos niveles inferiores no es fácil diferenciar un caso de otro.

- La velocidad de operación del switch es mayor que la del bridge (puente), que introduce mayores tiempos de retardo.

- En un switch se puede repartir el ancho de banda de la red de una manera apropiada en cada segmento de red o en cada nodo, de modo transparente a los usuarios. Esto proporciona facilidades para la construcción de redes virtuales.

- Gran parte de los modelos comerciales de conmutadores son apilables y por tanto, fácilmente escalables, por lo que les da una flexibilidad semejante a los repetidores, pero con la funcionalidad de los puentes en cuanto a la gestión del tráfico de la red se refiere.

- Algunos conmutadores de muy alto rendimiento se conectan en forma modular a un bus de muy alta velocidad (backplane) por el que producen su conmutación.

Patch Panel.- Están formados por un soporte, usualmente metálico y de medidas compatibles con rack de 19", que sostiene placas de circuito impreso sobre la que se montan: de un lado los conectores RJ45 y del otro los conectores IDC para block tipo 110.

Se proveen en capacidades de 12 a 96 puertos (múltiplos de 12) y se pueden apilar para formar capacidades mayores.

2.3.2. Redes de Área Amplia (Wide Area Network, WAN)

Cuando se llega a un cierto punto deja de ser poco práctico seguir ampliando una LAN.

A veces esto viene impuesto por limitaciones físicas, aunque suele haber formas más adecuadas o económicas de ampliar una red de computadoras. Dos de los componentes importantes de cualquier red son la red de teléfono y la de datos. Son enlaces para grandes distancias que amplían la LAN hasta convertirla en una red de área extensa (WAN). Casi todos los operadores de redes nacionales (como DBP en Alemania o British Telecom en Inglaterra) ofrecen servicios para interconectar redes de computadoras, que van desde los enlaces de datos sencillos y a baja velocidad que funcionan basándose en la red pública de telefonía hasta los complejos servicios de alta velocidad (como frame relay y SMDS-Switched Multimegabit Data Services) adecuados para la interconexión de las LAN. Estos servicios de datos a alta velocidad suelen denominarse conexiones de banda ancha. Se prevee que proporcionen los enlaces necesarios entre LAN para hacer posible lo que han dado en llamarse autopistas de la información.

Una WAN se extiende sobre un área geográfica amplia, a veces un país o un continente; contiene una colección de máquinas dedicadas a ejecutar programas de usuario (aplicaciones), estas máquinas se llaman Hosts. Los hosts están conectados por una subred de comunicación. El trabajo de una subred es conducir mensajes de un host a otro. La separación entre los aspectos exclusivamente de comunicación de la red (la subred) y los aspectos de aplicación (hosts), simplifica enormemente el diseño total de la red.

En muchas redes de área amplia, la subred tiene dos componentes distintos: las líneas de transmisión y los elementos de conmutación. Las líneas de transmisión (también llamadas circuitos o canales) mueven los bits de una máquina a otra.

Los elementos de conmutación son computadoras especializadas que conectan dos o más líneas de transmisión. Cuando los datos llegan por una línea de entrada, el elemento de conmutación debe escoger una línea de salida para enviarlos. Como término genérico para las computadoras de conmutación, les llamaremos enrutadores.

Redes WAN son aquellas que surgen para satisfacer las necesidades de transmisión de datos a distancias superiores a unos pocos kilómetros. Este tipo de redes permite conexiones entre múltiples usuarios y dispositivos de todo tipo.

Las redes WAN más comunes son las *Redes Públicas de Telecomunicaciones*, que existe en casi todos los países. Las redes WAN conocidas como redes de área extensa o redes de banda ancha.

La estructura de las WAN tiende a ser más irregular, debido a la necesidad de conectar múltiples terminales, computadoras y centros de conmutación. Como los canales están alquilados mensualmente (a un precio considerable), las empresas y organizaciones que los utilizan tienden a mantenerlos lo más ocupados posible. Para ello, a menudo los canales "serpentean" por una determinada zona geográfica para conectarse a los ETD allí donde estén. Debido a eso la topología de las WAN suele ser más irregular.

2.3.2.1. Constitución de una Red de Área Amplia (WAN)

La red consiste en ECD (computadoras de conmutación) interconectadas por canales alquilados de alta velocidad (por ejemplo, líneas de 56 kbit/s). Cada ECD utiliza un protocolo responsable de encaminar correctamente los datos y de proporcionar soporte a

los computadores y terminales de los usuarios finales conectados a los mismos. La función de soporte ETD (Terminales/computadoras de usuario). La función soporte del ETD se denomina a veces PAD (Packet Assembly/Disasembly – ensamblador / desensamblador de paquetes). Para los ETD, el ECD es un dispositivo que los aisla de la red. El centro de control de red (CCR) es el responsable de la eficiencia y fiabilidad de las operaciones de la red.

2.3.3. Intranets/Internet

Internet ha traído consigo, tanto el principio de una nueva era de comunicaciones globales como la estandarización de redes públicas. Representa una revolución tecnológica tan importante como la propiciada por el uso masivo de computadoras personales.

2.3.3.1. Intranets

Historia de la Red Intranet

El entusiasmo en torno a Internet ha llevado a las empresas a evaluar la forma de obtener beneficios utilizando Intranets – es decir, usando la tecnología Internet dentro de sus organizaciones.

Las Intranets tiene un gran auge en estos momentos, pues permite bajar los costos que demandaría la implantación de una red y la comunicación, no importa cual sea la magnitud. De otro lado permite que instituciones pequeñas (Pequeña y Mediana Empresa - PYMES) e instituciones con poca capacidad económica, tengan acceso a los

recursos compartidos por sistemas públicos, gremios y asociaciones que implementan este tipo de redes y servicios de información.

De otro lado, los servicios WEB sirven para el acceso gráfico a la Red INTERNET, siendo hoy día, el prototipo de comunicación del futuro (un futuro muy cercano), que incorpora servicios de acceso a eventos gráficos, Sistema MULTIMEDIA y las más importantes, el acceso a Sistemas y Aplicaciones propias del centro institucional (a través de pequeñas aplicaciones llamadas APPLETS), que bajo el formato de programas que operan en el ambiente de Cliente/Servidor, permite a un usuario institucional, hacer uso de los sistemas de Información de la Empresa u organización.

¿Qué es Intranet?

La definición de una Red INTRANET como una Red Corporativa o Red Institucional, que es implementada sobre la base de la infraestructura de una Red Global como INTERNET, constituye una solución tentadora. En otras palabras, haciendo uso de la Red de INTERNET, montada por los proveedores de este servicio de Red Global, en el país y en el exterior, las empresas o instituciones públicas o privadas, enlaza a sus diversas dependencias, para constituir una Red Corporativa de alcance nacional o mundial.

Una Red INTRANET, incorpora todos los Servicios que presta una Red Global como INTERNET, a los Sistemas de Información propios de la Corporación o Institución, siendo posible establecer:

- Acceso de los Bancos de Datos de las instituciones centralmente organizadas.
- Permite la coordinación a través del servicio de mensajería y transmisión de documentos. Sobre este concepto, se está construyendo todo un tipo de software orientado a la gestión de documentos y coordinación de actividades, como el Lotus Notes, Info Central y otros, que generan un ambiente común para los Grupos de Trabajo.

Asi, el Correo Electrónico y transferencias de mensajes y documentos, se convierten en herramientas indispensables, entre las Dependencias y Usuarios. De esta forma, un usuario que tiene acceso a la Red de una Institución, puede acceder a través de un servicio privado como Winview) o público (como INFOVIA) a los Servicios de Información de su Institución.

Por otro lado, el Acceso a Bases de Datos y Banco de Datos de la institución, conlleva a peligros originados por los hackers, que como siempre, desean visitar y en algunos casos destruir la información estratégica de una empresa o institución. Es por ello que los Centros de Operación de la Red Institucional, deben implantar programas de seguridad que protejan a la Red Institucional, de accesos no debidos (uso de software de protección y seguridad llamados FIREWALL), que operan en los gateways del servidor de Internet o en una máquina dedicada entre el ruteador (router) y la red, protegiendo así a la red.

¿Para qué sirve las Intranets?

Las empresas están interesadas en las Intranets para:

- Simplificar el control interno de la información y mejorar la comunicación dentro de las organizaciones, ofreciendo ayudas sumamente sencillas, pero poderosas, como las derivadas del uso de hipervínculos.
- Fomentar la colaboración real entre empleados, permitiendo el acceso generalizado a información permanente por medio de herramientas de análisis fáciles de utilizar.
- Establecer líneas de acción para cada proceso de negocios y para las tareas administrativas, mediante la difusión de procedimientos en torno a las aplicaciones existentes referentes a los sistemas de producción.

Las expectativas de los usuarios finales de una Intranet son simplemente la facilidad de uso, la velocidad y la confiabilidad. Como con los demás sistemas de producción, los responsables de la administración de la información necesitan asegurarse de que sus Intranets sean seguras, efectivas en relación con su costo y de fácil manejo. Como Internet, en un principio las Intranets fueron diseñadas pensando en la distribución de la información, pero con el tiempo permitiendo ofrecer muchas otras posibilidades que permiten reducir costos.

Algunos aspectos que deben tomarse en cuenta en el desarrollo de las Intranets son:

- La evolución de Internet y la terminología de uso común.
- La forma en que su empresa puede beneficiarse de las Intranets, de los sitios dinámicos, y de las tecnologías del futuro. En especial en relación con la disminución de costos.

- La forma en que Microsoft puede ayudarle - Microsoft está adoptando y ampliando las tecnologías relativas al Internet con miras a proporcionar un valor agregado a las inversiones existentes en las empresas.

2.3.3.2. Internet

Historia de la Red Internet

Internet surgió hace más de 30 años pero no con el nombre de Internet sino como **ARPANET**, un producto de investigaciones del departamento de defensa de EEUU "US Departament of Defense" para la construcción de redes informáticas en tiempos de guerra. Uno de los principales requerimientos de esta red era que debía estar descentralizada, es decir que toda la información no debía estar almacenada en un sólo punto y así, si este punto era destruido por algún ataque militar, no se perdería toda su vital información. Se necesitaba de una red que pudiera resistir daños parciales y aún así seguir funcionando.

Esta tecnología fue aprovechada y mejorada en 1990 por la Red de la Fundación Nacional para la Ciencia **"NSFNET"** que se encontraba en la necesidad de compartir información con los demás institutos de investigación. Se comenzó entonces a construir lo que conocemos hoy como **INTERNET.**

Internet crece a un ritmo vertiginoso. Constantemente se mejoran los canales de comunicación con el fin de aumentar la rapidez de envío y recepción de datos. Cada día que pasa se publican en la Red miles de documentos nuevos y se conectan por primera vez miles de personas. Con relativa frecuencia aparecen nuevas posibilidades de uso de

Internet, y constantemente se están inventando nuevos términos para poder entenderse en este nuevo mundo que no para de crecer.

La Red de Internet está compuesta de millones de computadoras que se comunican entre si utilizando diferentes medios.

La **Red de Internet** se muestra en la **(Figura 2.2.)** que se encuentra a continuación.

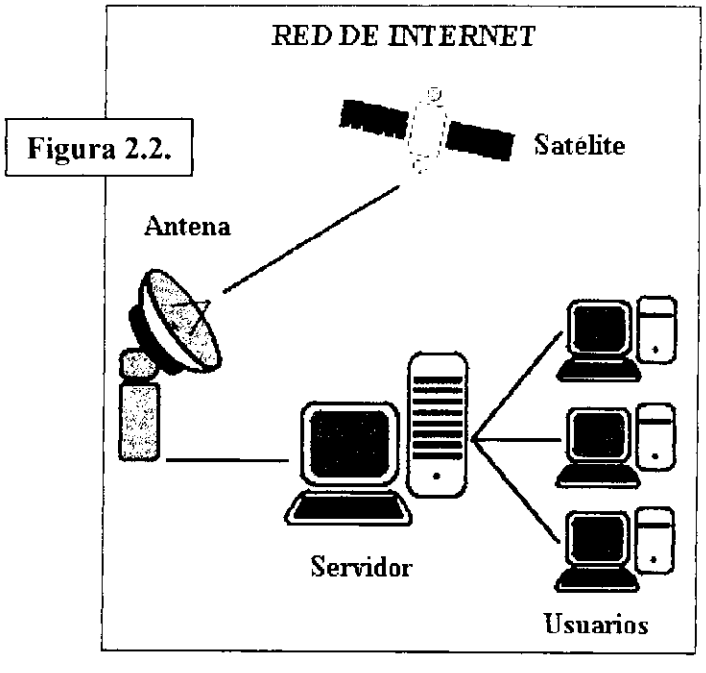


Figura 2.2.

¿Qué es Internet?

El Internet es entonces un conjunto de computadoras distribuidas en todo el mundo e interconectadas entre si, a través de varios medios de comunicación y Servidores.

Por esto se la conoce como la Red de redes o en los medios de comunicación suelen llamar a Internet la super autopista de la información.

Se conecta por teléfono (a través de un modem) o por fibra óptica y transmite toda clase de información. Su importancia radica en que a través de ella podemos obtener información rápida y eficaz sobre diversos temas, sin moverse de casa o del lado de su computadora, esto entre infinidades de aplicaciones que podemos utilizar.

La palabra Internet es el resultado de la unión de dos términos: **Inter**, que hace referencia a enlace o conexión y **Net** (Network) que significa interconexión de redes. Es decir, Internet no es otra cosa que una conexión integrada de redes de computadores o redes interconectadas.

Por medio de todo este conjunto de componentes de hardware y software. Se crearon y continúan desarrollándose numerosos servicios, aplicaciones y usos de toda índole que son aprovechados para diferentes fines, los que conforman el infinito mundo Internet.

Existe una gran variedad de "lenguajes" que usan los ordenadores para comunicarse por Internet. Estos "lenguajes" se llaman **Protocolos**. Se ha establecido que en Internet, toda la información ha de ser transmitida mediante el Protocolo **TCP/IP**.

¿Para qué sirve Internet?

Es sin lugar a dudas un mundo de infinitas posibilidades sin moverse de su casa ni del lado de su computadora, con Internet puede entre tantas y tantas aplicaciones:

- Comunicarse con un amigo al otro lado del planeta, casi de forma instantánea.
- Obtener información rápida sobre diversos temas.
- Escuchar música y observar videos.

- Transmitir cualquier tipo de datos.
- Viajar virtualmente (es decir, no físicamente sino a través de su PC, sintiendo como si estuviera en ese lugar) de un país a otro en pocos minutos.
- Leer las noticias y artículos de los principales diarios y revistas del orbe.
- Hacer sus reservaciones de hotel cuando piense viajar.
- Conocer nuevos amigos interesados en sus temas a distancia.
- Comprar y vender productos y servicios.
- Realizar cursos y aprender diferentes temas a distancia.
- Grabar, imprimir y copiar información de audio, video y texto.
- Participar en video juegos con personas ubicadas en otros lugares de la tierra.
- Conseguir la pareja de sus sueños.
- E inclusive hasta pedir una pizza.

2.4. SOFTWARE DE RED

2.4.1. Protocolos de Comunicación

Los protocolos de comunicación son las reglas y procedimientos utilizados en una red para establecer la comunicación entre los nodos que disponen de acceso a la red. Los protocolos gestionan dos niveles de comunicación distintos. Las reglas de alto nivel definen como se comunican las aplicaciones, mientras que las de bajo nivel definen como se trasmiten las señales por el cable.

Un *protocolo* es el "lenguaje" que un equipo utiliza. Los equipos deben utilizar el mismo protocolo para comunicarse.

Los protocolos que se utilizan en las comunicaciones son una serie de normas que deben aportar las siguientes funcionalidades:

- Permitir localizar un ordenador de forma inequívoca.
- Permitir realizar una conexión con otro ordenador.
- Permitir intercambiar información entre ordenadores de forma segura, independiente del tipo de máquinas que estén conectadas.
- Abstraer a los usuarios de los enlaces utilizados (red telefónica, radioenlaces, satélite) para el intercambio de información.
- Permitir liberar la conexión de forma ordenada.

Debido a la gran complejidad que conlleva la interconexión de ordenadores, se ha tenido que dividir todos los procesos necesarios para realizar las conexiones en diferentes niveles. Cada nivel se ha creado para dar una solución a un tipo de problema particular dentro de la conexión. Cada nivel tendrá asociado un protocolo, el cual entenderán todas las partes que formen parte de la conexión. Diferentes empresas han dado diferentes soluciones a la conexión entre ordenadores, implementando diferentes familias de protocolos, y dándole diferentes nombres (DECnet, TCP/IP, IPX/SPX, NETBEUI, etc.).

2.4.1.1. Protocolo TCP/IP (Transmission Control Protocol/Internet Protocol – Protocolo de Control de Transmisiones/Protocolo Internet)

TCP / IP fue desarrollado y demostrado por primera vez en 1972 por el departamento de defensa de los Estados Unidos, ejecutándolo en el ARPANET una red de área extensa del departamento de defensa.

El protocolo TCP/IP se utiliza para conectarse a Internet y a redes de área extensa (WAN).

TCP/IP, se relaciona automáticamente como el protocolo sobre el que funciona la red Internet. Esto, en cierta forma es cierto, ya que se le llama TCP/IP, a la familia de protocolos que nos permite estar conectados a la red Internet. Este nombre viene dado por los dos protocolos estrella de esta familia:

- El protocolo TCP, funciona en el nivel de transporte del modelo de referencia OSI, proporcionando un transporte fiable de datos.
- El protocolo IP, funciona en el nivel de red del modelo OSI, que nos permite encaminar nuestros datos hacia otras máquinas.

Pero un protocolo de comunicaciones debe solucionar una serie de problemas relacionados con la comunicación entre ordenadores, además de los que proporciona los protocolos TCP e IP.

Arquitectura de protocolos TCP/IP

Para poder solucionar los problemas que van ligados a la comunicación de ordenadores dentro de la red Internet, se tienen que tener en cuenta una serie de particularidades sobre las que ha sido diseñada TCP/IP:

- Los programas de aplicación no tienen conocimiento del hardware que se utilizará para realizar la comunicación (modem, tarjeta de red).
- La comunicación no está orientada a la conexión de dos máquinas, eso quiere decir que cada paquete de información es independiente y puede viajar por caminos diferentes entre dos máquinas.

- La interfaz de usuario debe ser independiente del sistema, así los programas no necesitan saber sobre que tipo de red trabajan.
- El uso de la red no impone ninguna topología en especial (distribución de los distintos ordenadores).

De esta forma, podremos decir, que dos redes están interconectadas, si hay una máquina común que pase información de una red a otra. Además, también podremos decir que una red Internet virtual realizará conexiones entre redes, que a cambio de pertenecer a la gran red, colaboraran en el tráfico de información procedente de una red cualquiera, que necesite de ella para acceder a una red remota. Todo esto independiente de las máquinas que implementen estas funciones y de los sistemas operativos que éstas utilicen.

2.4.1.2. Protocolo NetBEUI

El interfaz utilizador extendido NETBIOS (NetBEUI) primero fue introducido por IBM en 1985. NETBIOS, parte integral del programa piloto del protocolo de NetBEUI, es un interfaz de programación que pone muchas funciones de la capa en ejecución de sesión.

Es el protocolo nativo de Windows, normalmente no rutable. Es el sistema que debe utilizarse, por seguridad, cuando la red está constituida exclusivamente por máquinas Windows. Solo se puede transmitir por Internet si se encapsula sobre IP.

NetBEUI es un protocolo que puede utilizar para conectarse a servidores de Windows NT, Windows para Trabajo en Grupo o LAN Manager. El protocolo NetBEUI es utilizado sobre todo por Microsoft Networks para la comunicación sobre una red

pequeña. Para una variedad de razones es inadecuado para el uso en la red del campus de UGA.

El protocolo NetBEUI es relativamente limitado con respecto a sus aplicaciones y puede no poder comunicarse con los dispositivos fuera de su LAN departamental. NetBEUI es instalado por el valor por defecto cuando usted agrega al cliente para las redes de Microsoft y con el podremos acceder tanto a ficheros como a carpetas, discos, impresoras y todo aquello que sea compartido.

NetBEUI se optimiza para el rendimiento muy alto cuando está utilizado en segmentos departamentales de LANs o del LAN. Para el tráfico dentro de un segmento del LAN, NetBEUI es típicamente el protocolo más rápido. NetBEUI es un protocolo pequeño, eficiente, y rápido con gastos indirectos bajos.

Mientras que NetBEUI es rápido en LANs pequeño, no es tan eficaz en redes grandes porque tiene un esquema de dirección pobre. NetBEUI no permite nombres de ordenador duplicados en la misma red. Esto evita que una red tenga dos ordenadores con mismo nombre-algo difícil de eliminar en una red grande.

Las Ventajas y las Desventajas del Protocolo NetBEUI

Se resume en la **(Tabla 2.4.)** que se muestra a continuación:

(Tabla 2.4. Ventajas y Desventajas del Protocolo NetBEUI)

Ventajas	Desventajas
• Se templa para la comunicación pequeña del LAN y por lo tanto muy rápidamente en LANs. • Buena protección de error. • Uso pequeño de la memoria.	• No routable. • El funcionamiento a través de WANs es pobre. • Requiere cada ordenador de red tener un nombre único.

2.4.2. Interfaces y Servicios

2.4.2.1. Interfaces

Linux es un Sistema Operativo de redes, y como tal para disfrutar totalmente de el (y aprender mucho con su uso) es necesario disponer de una red donde comprobar sus características multiusuario y de red. Un ejemplo práctico de configuración de una red con 2 ordenadores, un Linux y un Windows 9x (desde el cual podremos acceder al primero gracias a las capacidades servidoras de Linux). (La **Configuración de la Red Linux – Windows** se muestra en la **(Figura 2.3.)**).

Interface

Es un circuito electrónico que facilita la conexión entre dos aparatos y permite que intercambien información.

Tipo de Interfaces

Tenemos dos tipos de interfaces entre las cuales mencionamos:

Interface lo.- El valor 127 en el primer campo se llama *loopback* y se refiere a una interfaz que permite al host enviarse paquetes a sí mismo, y se abrevia como (interface lo).

Interface de red eth0.- La interface de red eth0 permite ligar la direccion IP correcta asignada a la tarjeta de red. En el caso del servidor Linux esta IP será 192.168.0.1.

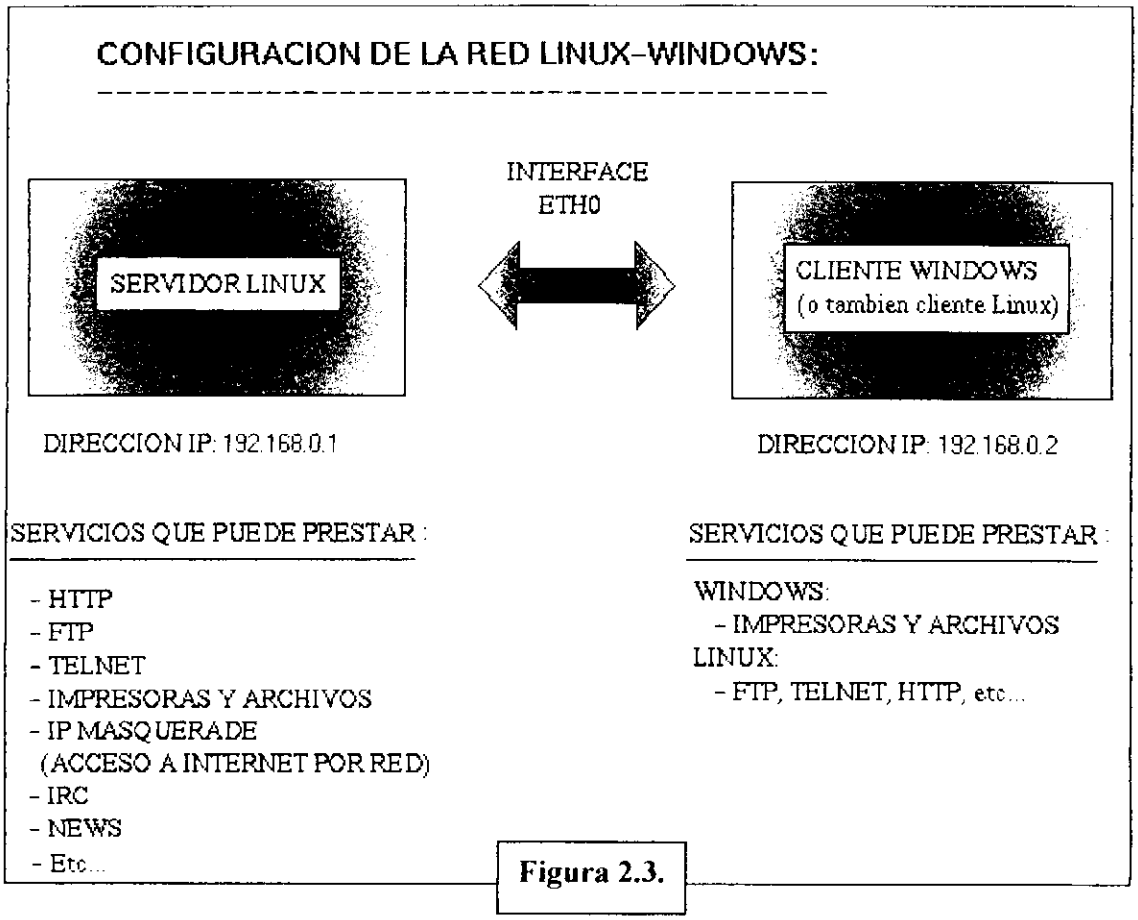


Figura 2.3.

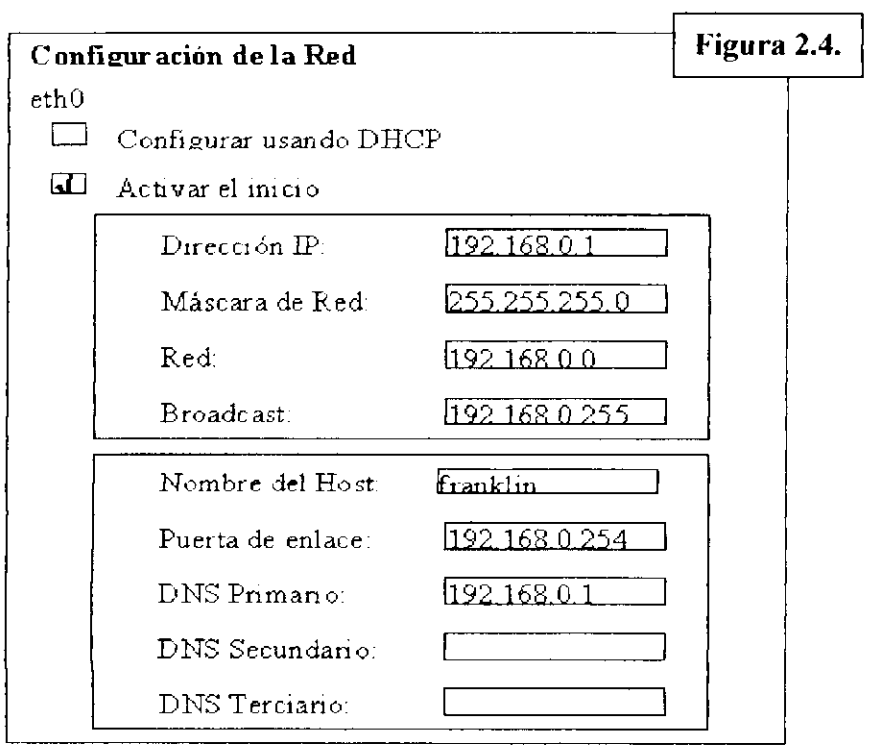
Configuración de la Red Linux

En el momento de la instalación, si desea configurar la tarjeta de red de forma automática se usa el protocolo DHCP. Si existen múltiples dispositivos Ethernet, cada dispositivo tendrá su propia configuración. Si se selecciona "Activar al inicio" la tarjeta de red se activará automáticamente durante el inicio del sistema. Se debe introducir, si es necesario, la dirección IP, la máscara de red, la red y las direcciones Broadcast. Se debe introducir, el nombre del host para el computador. En caso contrario tendrá el nombre predeterminado "localhost". En ultimo lugar, introduzca las direcciones de puerta de enlace, DNS Primario, DNS Secundario, DNS Terciario.

En nuestro ejemplo tendremos 2 ordenadores conectados por 2 tarjetas de red 10/100M PCI Fast Ethernet Adapter. Los parámetros de la red serán los siguientes:

Máquina 1: En la **(Figura 2.4.)** que se muestra a continuación.

Figura 2.4.



Máquina 2: En la **(Figura 2.5.)** que se muestra a continuación.

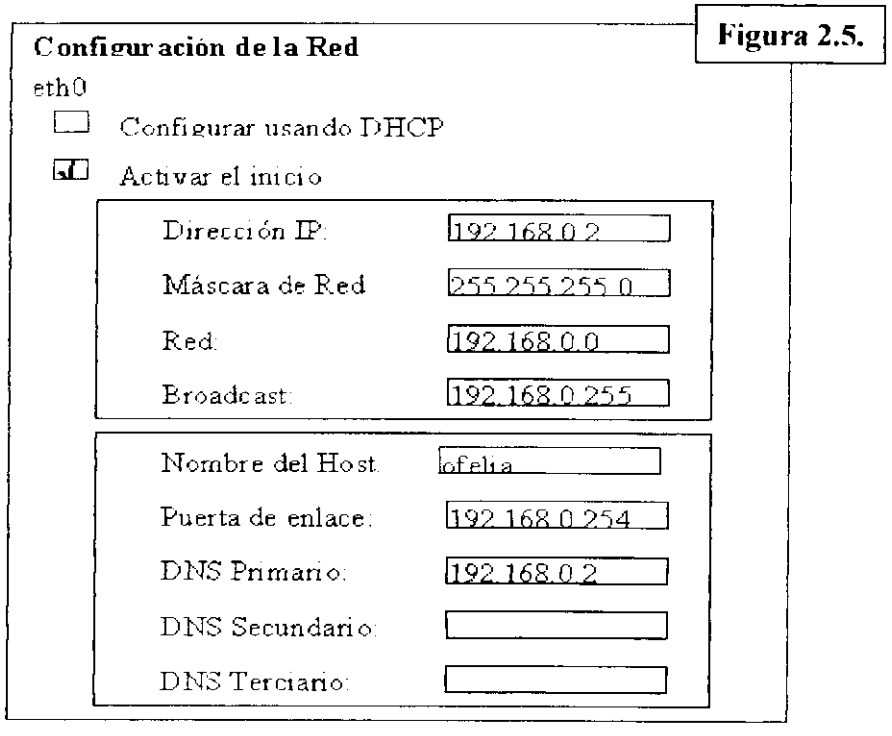


Figura 2.5.

Configuración del Hostname del Servidor Linux

Para hacer accesibles las máquinas por su nombre (no sólo por su dirección IP), es necesario modificar algunos ficheros de configuración de Linux, entre ellos el fichero /etc/hosts, una base de datos de direcciones IP y nombres asociados, que además permite incluir alias o nombres cortos para las máquinas.

Un ejemplo de /etc/hosts para nuestros propósitos (incluyendo ya la dirección IP de la otra máquina) sería el siguiente:

Ejemplo de /etc/hosts (líneas # son comentarios)

Las separaciones entre cadenas son tabuladores.

# Dirección IP	Alias	Nombre y dominio
127.0.0.1	localhost	localhost.localdomain localhost
192.168.0.1	franklin	localhost.localdomain localhost
192.168.0.2	ofelia	cliente ofelia

Otra manera de configurar el hostname sería la siguiente:

Al inicio del **KDE → Sistema → Network Configuration**

Las siguientes pantallas **(Figura 2.6.)**, **(Figura 2.7.)**, **(Figura 2.8.)**, **(Figura 2.9.)** muestran la configuración del hostname del Servidor Linux y los clientes que pertenecen a la red, la ayuda que proporciona este ambiente gráfico es porque modifica automáticamente el fichero que mencionamos anteriormente /etc/hosts:

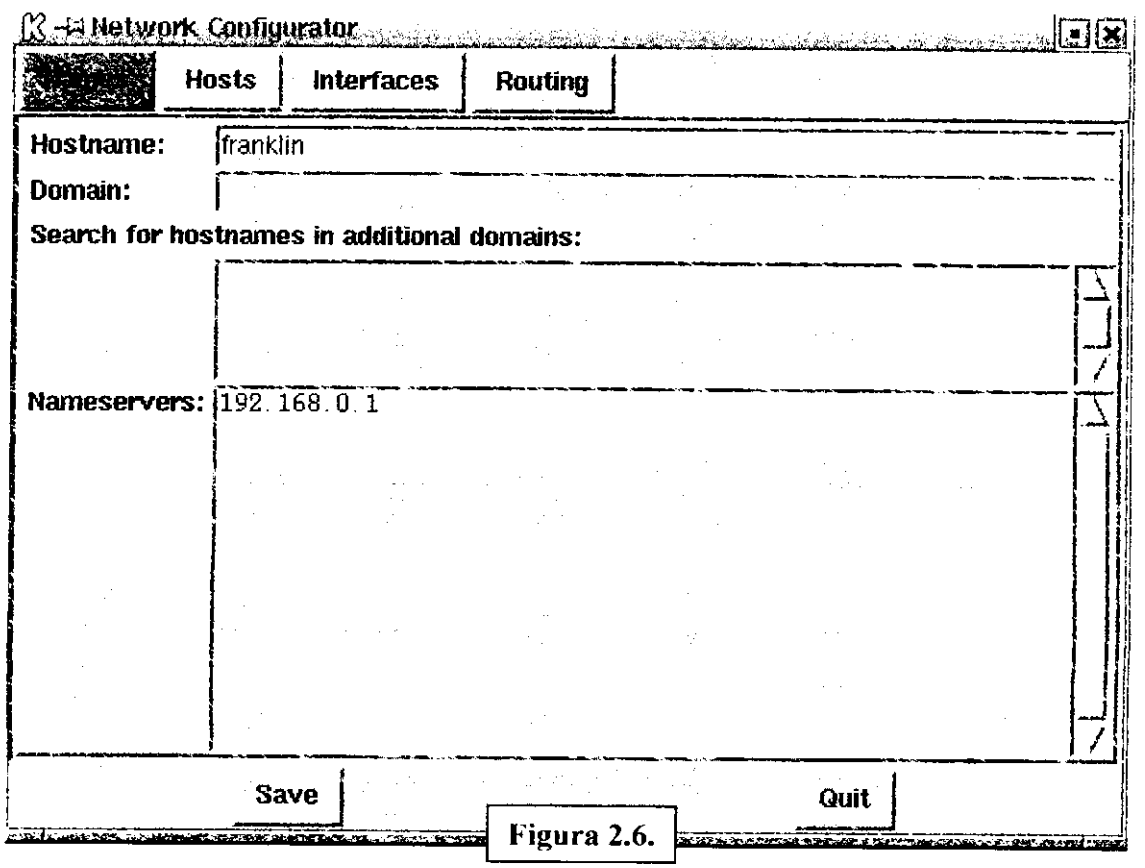


Figura 2.6.

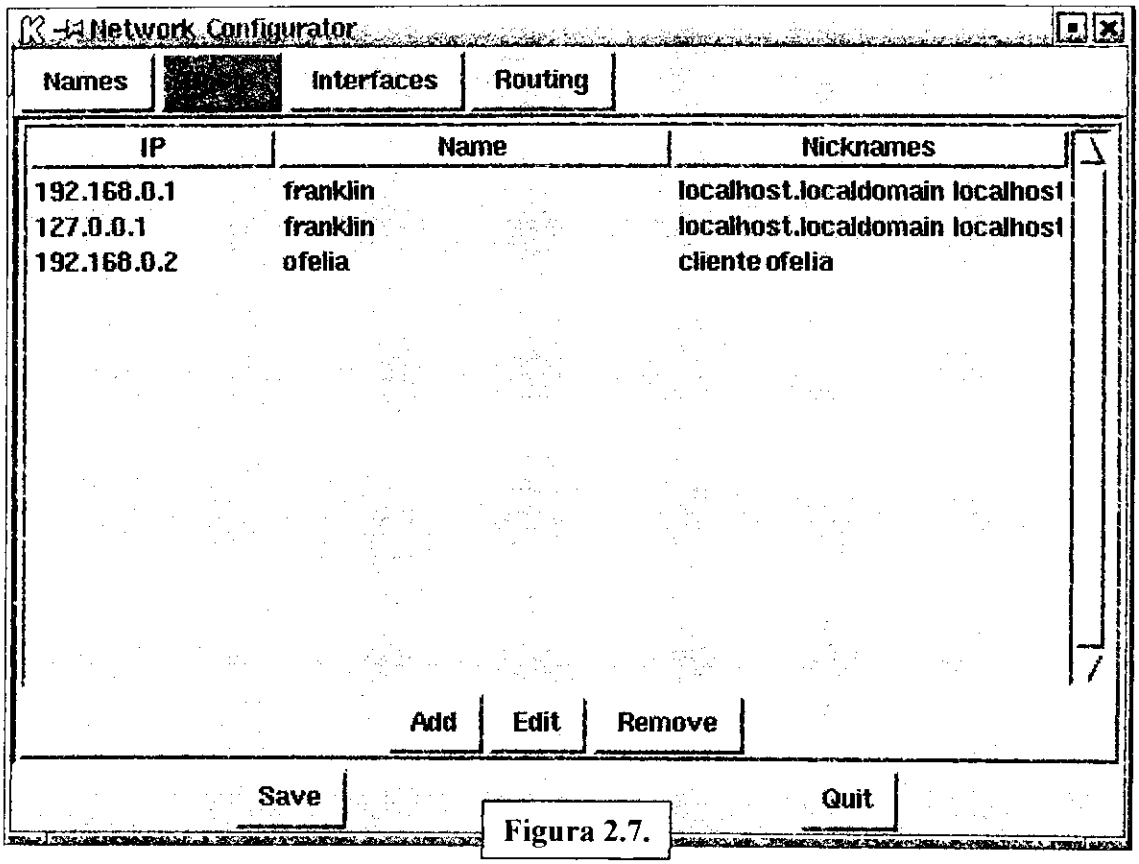


Figura 2.7.

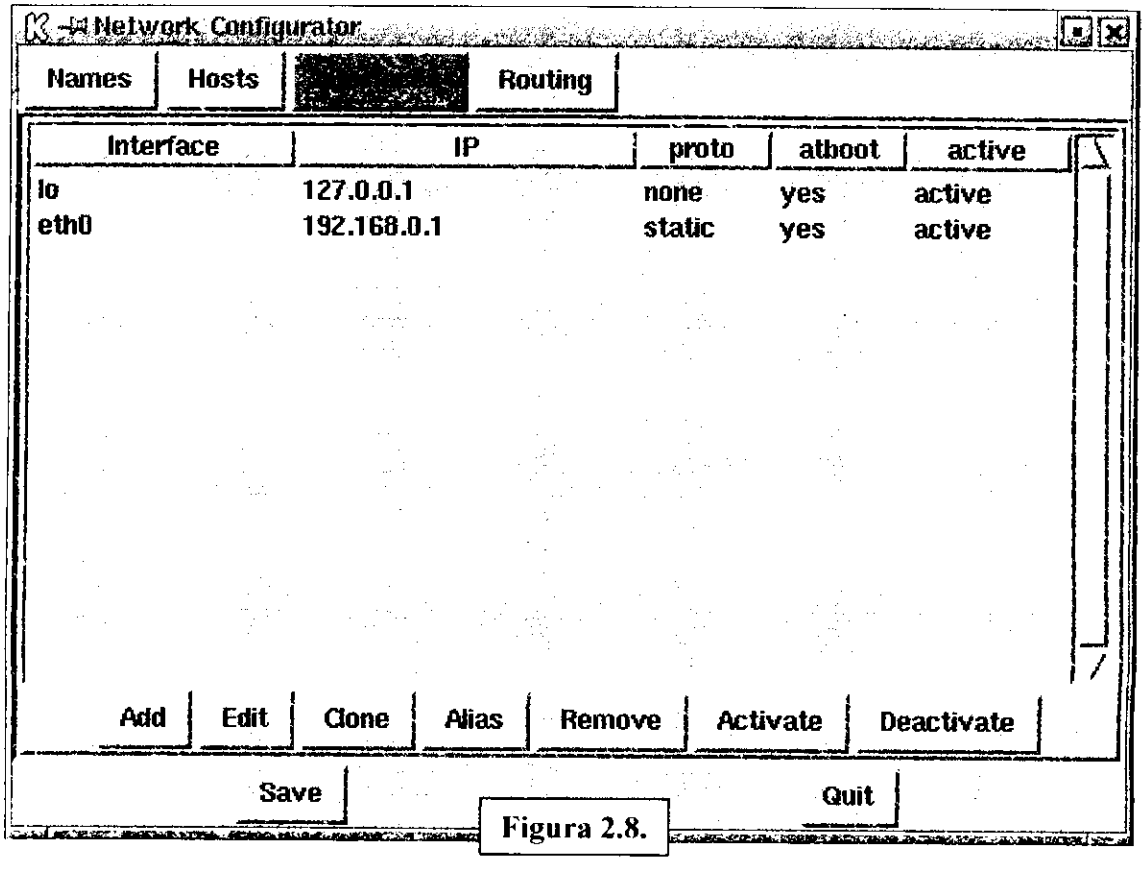


Figura 2.8.

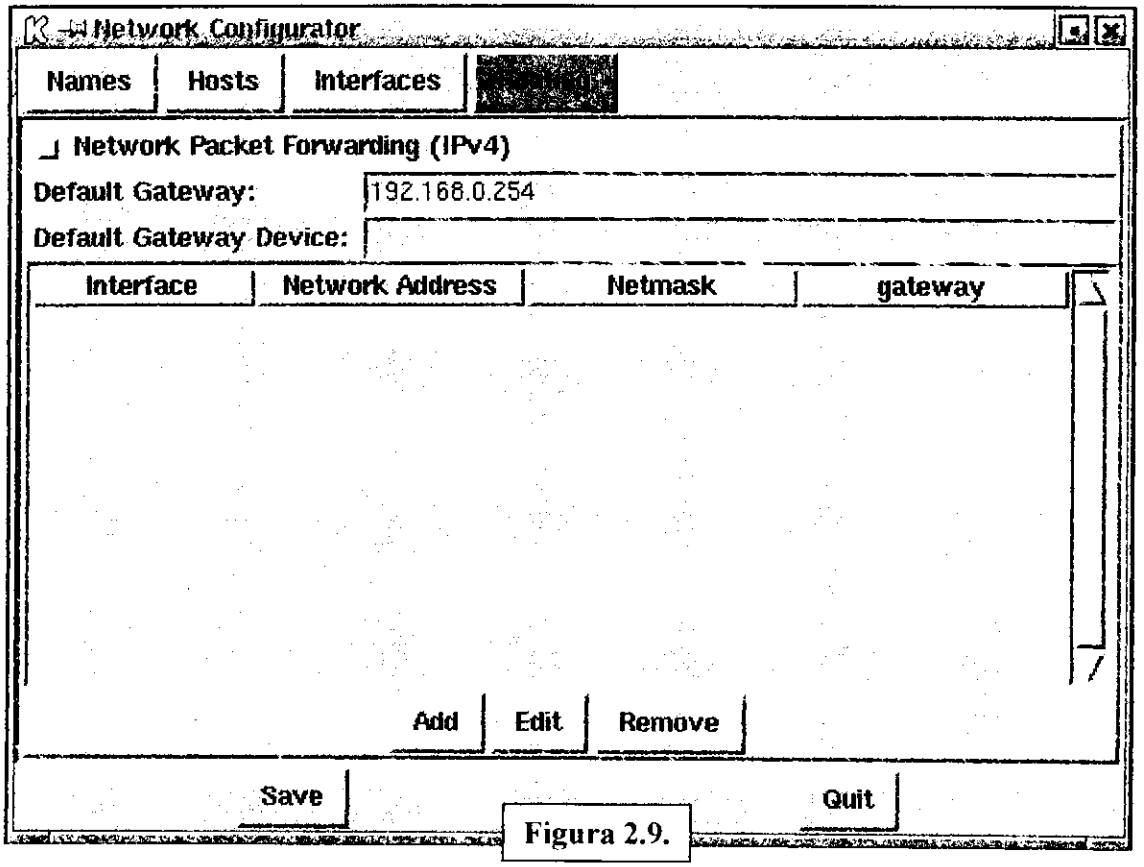


Figura 2.9.

Tras estas modificaciones (y tras una recomendable inicialización a menos que se restablezcan todos los servicios y nombres desde la línea de comandos, algo perfectamente posible en Linux) ya es posible ejecutar cualquiera de los siguientes comandos con el mismo resultado (de modo que es recomendable realizar una prueba si el servidor de nombres está funcionando correctamente):

ping franklin

ping 192.168.0.1

Si también hemos configurado el cliente Windows también deberíamos poder realizar un comando ping hacia el otro IP o hostname. Por otra parte, si en lugar de (o además de) un cliente Windows quisiéramos configurar otro cliente Linux, la configuración se realizaría con los mismos parámetros que la máquina servidora Linux.

Configuración del Cliente Windows

La configuración del cliente Windows suele ser la más problemática de ambas: si éste no detecta su tarjeta de red (aunque sea Plug&Play es muy habitual) y no se dispone de los discos de controladores del fabricante se tendrá que sacar la tarjeta de red del slot, arrancar, apagar y volver a conectar la tarjeta (para que Windows detecte el cambio e instale el nuevo hardware). Tras esto deberemos comprobar que la dirección del puerto E/S y la IRQ asignadas con Linux coinciden con las de la tarjeta (que se suelen cambiar mediante una utilidad incluida en los diskettes de controladores de la misma). Para esto miramos en el Panel de Control, Red, Propiedades, y examinamos los valores asignados, cambiándolos si no son correctos.

Tras esto debemos asignar a este nodo de la red la dirección IP que decidimos en un principio: la 192.168.0.2; esto se realiza desde el Panel de control, Red, Configuración - Propiedades TCP/IP (de la tarjeta de red, no las genéricas, como puede apreciarse en la **(Figura 2.10.)**).

Especificamos los siguientes datos:

Dirección IP - Especificar dirección IP - 192.168.0.2

Máscara de subred - 255.255.255.0

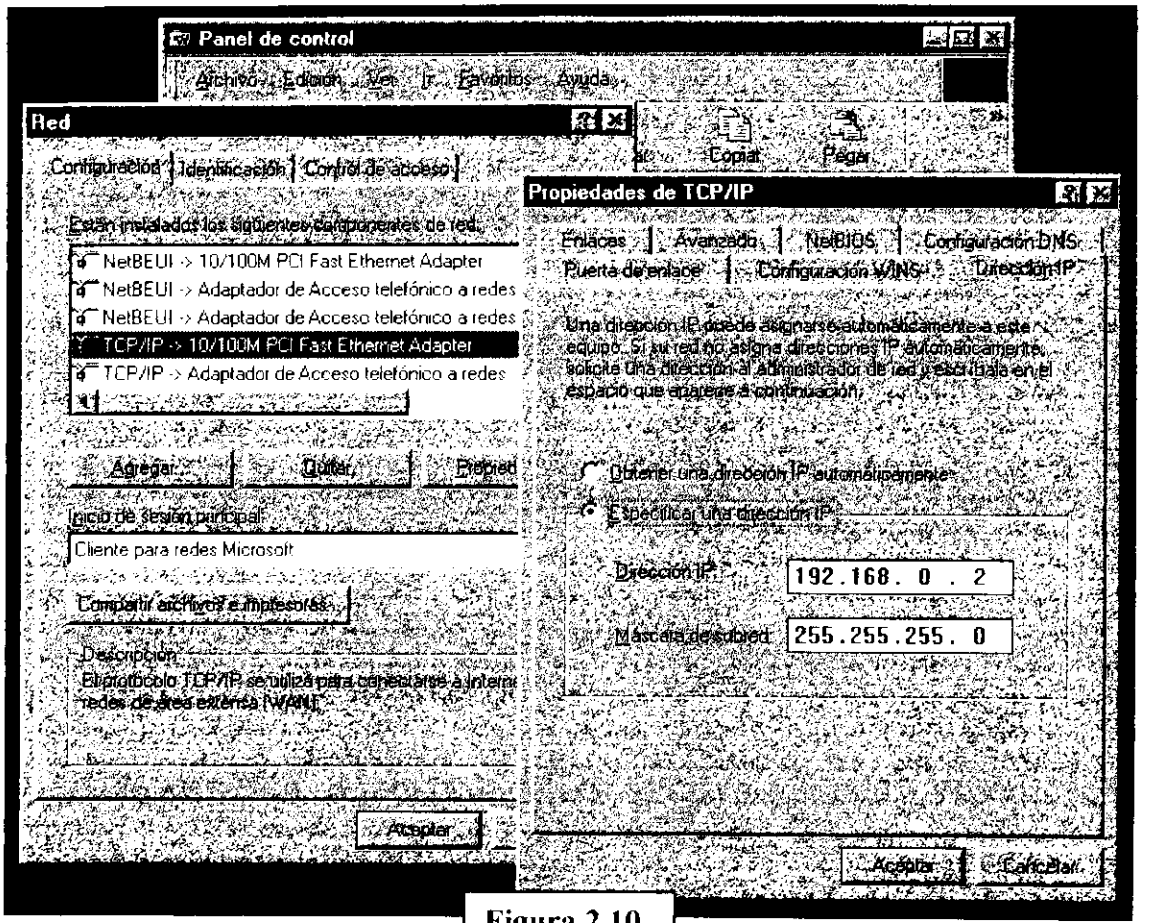


Figura 2.10.

Posteriormente debemos asignar la Identificación respectiva a la máquina; esto se realiza desde el Panel de control, Red, Identificación como puede apreciarse en la **(Figura 2.11.).**

Identificación - Nombre del PC - ofelia

Grupo de trabajo - WORKGROUP

Descripción del PC - Es indiferente

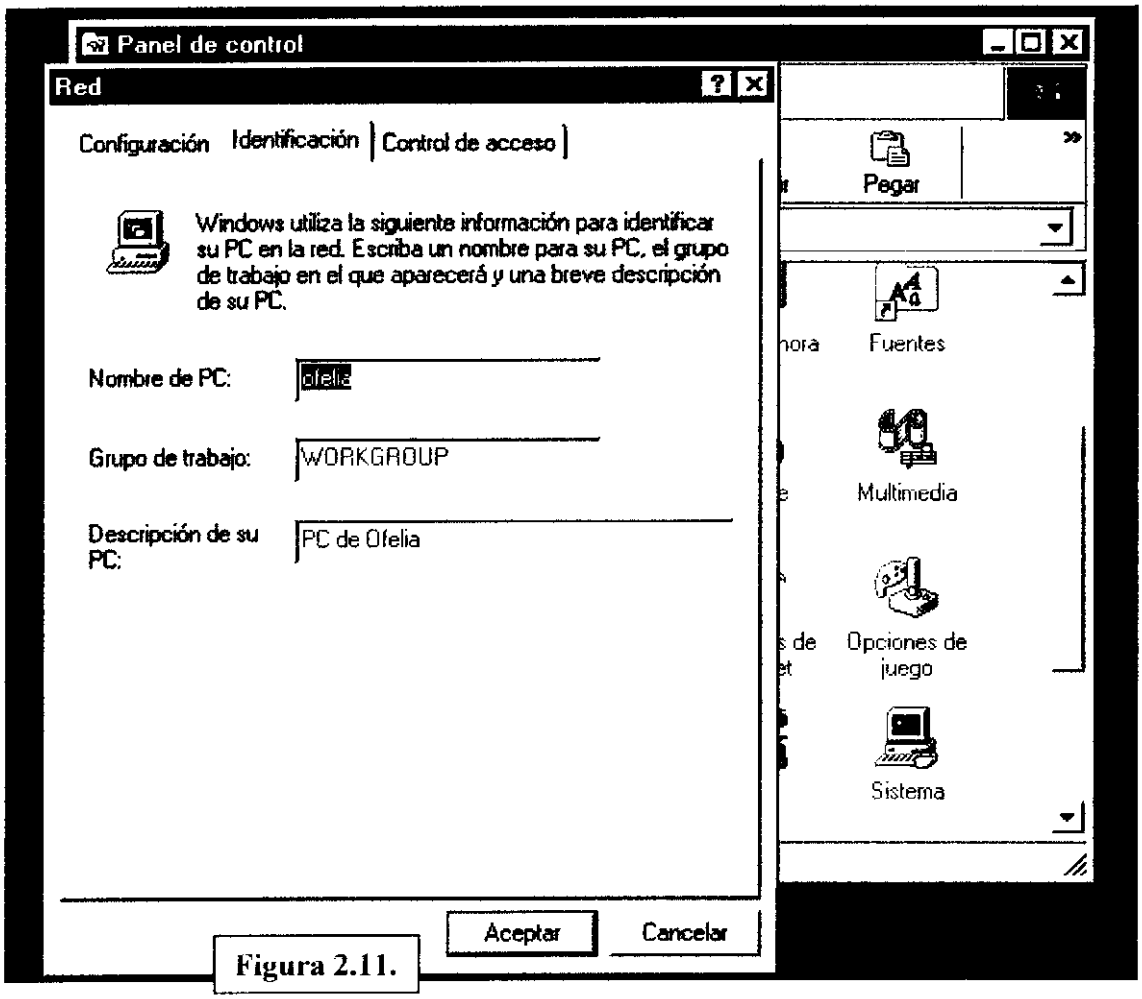


Figura 2.11.

Después debemos crear la base de datos de nombres y direcciones IP, situada en el:

Fichero c:\windows\hosts (líneas # son comentarios)

Los espacios entre campos son tabuladores.

Direccion IP Nombre Alias

127.0.0.1 localhost

192.168.0.1 franklin

192.168.0.2 ofelia

2.4.2.2. Servicios

Un servicio permite que una computadora o equipo comparta archivos, impresoras y demás recursos con otros equipos.

Telnet

El método más modesto de utilizar una computadora es Telnet, que consiste en conectarse a la misma por intermedio de la red (en vez de en forma local) pero a través del protocolo TCP/IP. De esta manera en vez de utilizar una terminal tonta, ya se puede usar una PC común y operar cualquier computadora que ofrezca este servicio.

También se puede acceder vía Telnet a otros servicios como Gopher, servidores Web en modo texto, y bibliotecas de servidores WAIS, aún cuando no se tengan los clientes de estos servicios.

Usando Telnet en los distintos Sistemas Operativos.- Si se trabaja con un sistema Linux o Windows. Telnet se reduce a poner *telnet nombre host* donde el nombre del host se puede expresar como su dirección IP o su nombre de dominio.

FTP (File Transfer Protocol)

Este servicio permite la transferencia de archivo al y desde el servidor de FTP, se diseñó para permitir el intercambio de datos, archivos entre computadoras host y cliente.

La estructura de FTP es Cliente-Servidor, el servidor posee una estructura de directorios o carpetas en donde se encuentran alojados, los archivos de texto, gráficos, etc y el

cliente accede mediante un utilitario de FTP o línea de comando para extraer archivos a su PC o enviarlos al servidor.

Cuando se ingresa a un servidor FTP se puede hacer como usuario con permisos definidos o como usuario invitado, siempre y cuando el administrador del sistema habilite el mismo, luego puede recorrer las distintas carpetas hasta encontrar el archivo buscado, una vez encontrado este se transfiere a nuestra computadora.

Windows 98 tiene incluido un cliente FPT de línea de comandos que se ejecuta desde una ventana DOS, también existen programas shareware o freeware con entornos gráficos mucho más amigables.

Ping

El autor de la versión inicial del programa PING que actualmente utilizamos fue Mike Muss.

Ping fue nombrado debido al sonido del sistema de barrido de un sonar. Inclusive existe una historia en la cual se dice que un administrador de sistemas escribió un script que repetidamente hacia un Ping a una computadora en la red y obtenía una respuesta audible de un "Ping" por cada éxito. El administrador de sistemas entonces era capaz de ir metódicamente alrededor de su red revisando conectores BNC hasta encontrar el conector fallido que habia estado haciendo Ping en su red — cuando el sonido se detenía, había encontrado su repuesta.

El Ping es la más sencilla de todas las aplicaciones TCP/IP. Envía uno o más datagramas a un host de destino determinado solicitando una respuesta y mide el tiempo que tarda en retornarla; es decir este programa nos indica el tiempo exacto que tardan los paquetes de datos en ir y volver a través de la red desde nuestro PC a un determinado servidor remoto.

La palabra *Ping*, que puede usarse como nombre o como verbo(en inglés), procede de la operación de sonar empleada para localizar un objeto submarino. También es un acrónimo de *Packet InterNet Groper*.

Tradicionalmente, si se podía hacer un Ping a un host, otras aplicaciones como Telnet o FTP podían comunicarse con ese host. Con el advenimiento de las medidas de seguridad en Internet, particularmente los cortafuegos (firewalls), que controlan el acceso a redes a través del protocolo de aplicación y/o el número de puerto, esto ha dejado de ser estrictamente cierto. Aún así, el primer test para comprobar si es posible alcanzar un host sigue siendo intentar hacerle un ping.

La sintaxis usada en diferentes implementaciones varía de unas plataformas a otras. La mostrada aquí corresponde a la implementación en OS/2:

ping [-Opción] host [Tamaño [Paquetes]]

Donde:

Opción : Activa diversas opciones del ping.

Host : El destino: un nombre simbólico o bien una dirección IP.

Tamaño : El tamaño del paquete.

Paquetes: El número de paquetes a enviar.

Ping usa los mensajes Eco y Respuesta al Eco("Echo", "Echo Reply") de ICMP. Ya que se requiere ICMP en cada implementación de TCP/IP, a los hosts no les hace falta un servidor separado para responder a los Pings.

El Ping es útil para verificar instalaciones TCP/IP. Consideremos las cuatro siguientes formas del comando; cada una requiere operar con una parte adicional de la instalación TCP/IP.

ping loopback: Verifica la operatividad del software de la base de TCP/IP.

ping my-IP-address: Verifica si el correspondiente dispositivo de la red física puede ser direccionado.

ping a-remote-IP-address: Verifica si es posible acceder a la red.

ping a-remote-host-name: Verifica la operatividad del servidor de nombres(o del "flat namespace resolver", dependiendo de la instalación).

Opciones del Comando Ping.- Existen muchas opciones útiles para el comando Ping.

Estas son agrupadas en la siguiente tabla: **(Tabla 2.5. Opciones del Comando Ping)**.

(Tabla 2.5. Opciones del Comando Ping)

Opción	Descripción
-a	Resuelve direcciones a nombres de host.
-c conteo	Dejar de enviar y recibir paquetes después de conteo paquetes.
-d	Activa el SO_DEBUG en el socket utilizado.
-f	Enviar los paquetes lo más rápido posible (flood o inundación). No fragmentar el paquete.
-i TTL	Establecer un intervalo de tiempo segundos entre paquetes. Tiempo de vida.
-j lista de hosts	Ruta origen variable en la lista de host.
-k lista de hosts	Ruta origen estricta en la lista de host.
-l tamaño	Envía paquetes precargar tan rápido como sea posible, y regresa a modo normal. Tamaño del búfer de envíos.
-n cantidad	No buscar los nombres de la computadora (hostnames), sólo dar direcciones IP (numéricas). Cantidad de solicitudes de eco a enviar.
-p patron	Especificar hasta 16 bytes de "datos de acolchonamiento" a ser enviados con el paquete.
-q	Solo desplegar el sumario (silencioso).
-r cantidad	No utilizar tablas de ruteo para enviar el paquete, sólo arrojarlo por la interface local. Registrar la ruta para esta cantidad de saltos.
-R	Activar la opción de Grabar Ruta.
-s cantidad	Activa el número de bytes de datos a tamaño. Registrar horarios para esta cantidad de saltos.

-t	Solicita eco al host hasta ser interrumpido.
-T soloes	Envía un ping con la opción de una estampa de tiempo.
-T esydir	Obtiene estampas de tiempo y direcciones.
-T espresepec [host1 [host2 [host3 [host4]]]]	Obtiene estampas de tiempo y direcciones de brincos pre-especificados.
-v TOS	Tipo de servicio.
-w tiempo	Tiempo de espera agotado de respuesta en milisegundos.
.	Para ver estadísticas y continuar: presione Ctrl-Inter.
.	Para interrumpir: presione Ctrl-C.

Correo Electrónico

Permite enviar mensajes casi de forma instantánea a usuarios de todo el mundo, se lo conoce también como e-mail.

El correo electrónico trabaja de forma parecida a como trabaja el correo tradicional, tenemos el mensaje en si y una dirección de la persona destinataria, la diferencia radica en los medios. El correo electrónico utiliza el internet como medio de transporte, por esto el correo electrónico es mucho más rápido de llegar a su destinatario.

Para poder trabajar con el correo electrónico debemos utilizar algunos programas disponibles en el mercado. El más común es Microsoft Outlook Express, debido a que viene ya incluido en el sistema operativo Windows que a su vez son los más utilizados por el usuario final.

Protocolo del Correo Electrónico.- El objetivo dle SMTP es transferir correo electrónico segura y eficazmente.

SMTP es independiente del subsistema de transmisión y requiere sólo un fiable y ordenado canal de datos. Apéndices A, B, C, y D describen el uso de SMTP con varios servicios de transporte.

Un rasgo importante de SMTP es su capacidad para retransmitir correo a través de entornos del servicio del transporte. Un servicio de transporte provee e interprocesa un entorno de comunicación (IPCE). Un IPCE cubriría una red, varias redes, o un subconjunto de una red. Es importante notar que el sistemas del transporte (o IPCEs) no es punto-a-punto con redes. Un proceso puede comunicar directamente con otro proceso por cualquier IPCE conocido. El correo electrónico es una aplicación o uso de comunicación entre procesos. El mail se puede comunicar entre procesos en IPCEs diferentes por transmisión de un proceso que se conectó con dos o más IPCs. Más específicamente, el mail se puede transmitir entre hosts en diferentes sistemas de transporte por un host en ambos sistemas del transporte.

2.5. MODELOS DE REFERENCIA

Una de las necesidades más necesarias de un sistema de comunicaciones es el establecimiento de estándares, sin ellos sólo podrían comunicarse entre si equipos del mismo fabricante y que usaran la misma tecnología.

La conexión entre equipos electrónicos se ha ido estandarizando paulatinamente siendo las redes telefónicas las pioneras en este campo. Por ejemplo la histórica CCITT definió los estándares de telefonía: PSTN, PSDN e ISDN.

Otros organismos internacionales que generan normas relativas a las telecomunicaciones son: ITU-TSS (antes CCITT), ANSI, IEEE e ISO.

2.5.1. Modelo de Referencia OSI (Open System Interconnection – Interconexión de Sistemas Abiertos)

La ISO (International Standards Organization) ha generado una gran variedad de estándares, siendo uno de ellos la norma ISO-7494 que define el modelo OSI, este modelo nos ayudará a comprender mejor el funcionamiento de las redes de ordenadores.

El modelo OSI surgió frente a la necesidad imperante de interconectar sistemas de procedencia diversa en los que cada fabricante empleaba sus propios protocolos e intercambiaba señales.

Este modelo fue desarrollado por la necesidad de interconectar sistemas de distintos fabricantes, por lo que fue hecho con base en necesidades generales de todos los sistemas, de tal forma que los fabricantes pudieran apegarse a estas funciones.

2.5.1.1. Capas o Niveles del Modelo de Referencia OSI

Proporciona una arquitectura de 7 capas o niveles, basadas en las siguientes premisas:

- Cada nivel realiza tareas únicas y específicas y debe ser creado cuando se necesite un grado diferente de abstracción.
- Todo nivel debe servirse de los servicios del nivel anterior, a la vez que los debe de prestar al superior.
- Los servicios de un nivel determinado son independientes de su implantación práctica.

Capa Física (Physical Layer – capa 1).- Se encarga de pasar bits al medio físico y de suministrar servicios a la siguiente capa. Para ello debe conocer las características mecánicas, eléctricas, funcionales y de procedimiento de las líneas.

Capa de Enlace de Datos (Data Link Layer – capa 2).- Esta capa debe de encargarse de que los datos se envíen con seguridad a su destino y libres de errores. Cuando la conexión no es punto a punto, esta capa no puede asegurar su cometido y es la capa superior quien lo debe hacer.

Capa de Red (Network Layer – capa 3).- Esta capa se encarga de enlazar con la red y encaminar los datos hacia sus lugares o direcciones de destino. Para esto, se produce un diálogo con la red para establecer prioridades y encaminamientos. Esta y las dos capas inferiores son las encargadas de todo el proceso externo al propio sistema y que están tanto en terminales como en enlaces o repetidores.

Capa de Transporte (Transport Layer – capa 4).- Esta capa se encarga de que los datos enviados y recibidos lleguen en orden, sin duplicar y sin errores. Puede ser

servicio de transporte orientado a conexión (conmutación de circuitos o circuitos virtuales) o no orientado a conexión (datagramas).

Capa de Sesión (Session Layer – capa 5).- Se encarga de proporcionar diálogo entre aplicaciones finales para el uso eficiente de las comunicaciones. Puede agrupar datos de diversas aplicaciones para enviarlos juntos o incluso detener la comunicación y restablecer el envío tras realizar algún tipo de actividad.

Capa de Presentación (Presentation Layer – capa 6).- Esta capa se encarga de definir los formatos de los datos y si es necesario, procesarlos para su envío. Este proceso puede ser el de compresión o el de paso a algún sistema de codificación. En resumen, se encarga de la sintaxis.

Capa de Aplicación (Application Layer – capa 7).- Esta capa acoge a todas las aplicaciones que requieren la red. Permite que varias aplicaciones compartan la red.

2.5.1.2. Funcionamiento del Modelo de Referencia OSI

El sistema de comunicaciones del modelo OSI estructura el proceso en varias capas que interaccionan entre sí. Una capa proporciona servicios a la capa superior siguiente y toma los servicios que le presta la siguiente capa inferior.

De esta manera, el problema se divide en subproblemas más pequeños y por tanto más manejables.

Para comunicarse dos sistemas, ambos tienen el mismo modelo de capas. La capa más alta del sistema emisor se comunica con la capa más alta del sistema receptor, pero esta comunicación se realiza vía capas inferiores de cada sistema. La única comunicación directa entre capas de ambos sistemas es en la capa inferior (capa física).

Los datos parten del emisor y cada capa le adjunta datos de control hasta que llegan a la capa física. En esta capa son pasados a la red y recibidos por la capa física del receptor.

Luego irán siendo captados los datos de control de cada capa y pasados a una capa superior. Al final, los datos llegan limpios a la capa superior.

2.5.2. Modelo de Referencia TCP/IP

Toda arquitectura de protocolos se descompone en una serie de niveles, usando como referencia el modelo OSI. Esto se hace para poder dividir el problema global en subproblemas de más fácil solución.

2.5.2.1. Capas o Niveles del Modelo de Referencia TCP/IP

A diferencia de OSI, formado por una torre de siete niveles, TCP/IP se descompone en cinco niveles, cuatro niveles software y un nivel hardware. A continuación se describe los niveles software, los cuales tienen cierto paralelismo con el modelo OSI.

Nivel de Aplicación.- Constituye el nivel más alto de la torre TCP/IP. A diferencia del modelo OSI, se trata de un nivel simple en el que se encuentran las aplicaciones que

acceden a servicios disponibles a través de Internet. Estos servicios están sustentados por una serie de protocolos que los proporcionan. Por ejemplo, tenemos el protocolo FTP (File Transfer Protocol), que proporciona los servicios necesarios para la transferencia de ficheros entre dos ordenadores.

Otro servicio, sin el cual no se concibe Internet, es el de correo electrónico, sustentado por el protocolo SMTP (Simple Mail Transfer Protocol).

Nivel de Transporte.- Este nivel proporciona una comunicación extremo a extremo entre programas de aplicación. La máquina remota recibe exactamente lo mismo que le envió la máquina origen. En este nivel el emisor divide la información que recibe del nivel de aplicación en paquetes, le añade los datos necesarios para el control de flujo y control de errores y se los pasa al nivel de red junto con la dirección de destino.

En el receptor este nivel se encarga de ordenar y unir las tramas para generar de nuevo la información original.

Para implementar el nivel de transporte se utilizan dos protocolos:

- **UDP.-** Proporciona un nivel de transporte no fiable de datagramas, ya que apenas añade información al paquete que envía al nivel inferior, sólo la necesaria para la comunicación extremo a extremo. Lo utilizan aplicaciones como NFS y RPC, pero sobre todo se emplea en tareas de control.
- **TCP (Transport Control Protocolo).-** Es el protocolo que proporciona un transporte fiable de flujo de bits entre aplicaciones. Esta pensado para poder

enviar grandes cantidades de información de forma fiable, liberando al programador de aplicaciones de la dificultad de gestionar la fiabilidad de la conexión (retransmisiones, perdidas de paquete, orden en que llegan los paquetes, duplicados de paquetes...) que gestiona el propio protocolo. Pero la complejidad de la gestión de la fiabilidad tiene un coste en eficiencia, ya que para llevar a cabo las gestiones anteriores se tiene que añadir bastante información a los paquetes a enviar. Debido a que los paquetes a enviar tienen un tamaño máximo, como más información añada el protocolo para su gestión, menos información que proviene de la aplicación podrá contener ese paquete.

Por eso, cuando es más importante la velocidad que la fiabilidad, se utiliza UDP, en cambio TCP asegura la recepción en destino de la información a transmitir.

Nivel de Red.- También recibe el nombre de nivel Internet. Coloca la información que le pasa el nivel de transporte en datagramas IP, le añade cabeceras necesarias para su nivel y lo envía al nivel inferior. Es en este nivel donde se emplea el algoritmo de encaminamiento, al recibir un datagrama del nivel inferior decide, en función de su dirección, si debe procesarlo y pasarlo al nivel superior, o bien encaminarlo hacia otra máquina. Para implementar este nivel se utilizan los siguientes protocolos:

- **IP (Internet Protocol).-** Es un protocolo no orientado a la conexión, con mensajes de un tamaño máximo. Cada datagrama se gestiona de forma independiente, por lo que dos datagramas pueden utilizar diferentes caminos para llegar al mismo destino, provocando que lleguen en diferente orden o bien duplicados. Es un protocolo no fiable, eso quiere decir que no corrige los

anteriores problemas, ni tampoco informa de ellos. Este protocolo recibe información del nivel superior y le añade la información necesaria para su gestión (direcciones IP).

- **ICMP (Internet Control Message Protocol).-** Proporciona un mecanismo de comunicación de información de control y de errores entre máquinas intermedias por las que viajarán los paquetes de datos. Estos datagramas los suelen emplear las máquinas (gateways, host...) para informarse de condiciones especiales en la red, como la existencia de una congestión, la existencia de errores y las posibles peticiones de cambios de ruta. Los mensajes de ICMP están encapsulados en datagramas IP.
- **IGMP (Internet Group Management Protocol).-** Este protocolo está íntimamente ligado a IP. Se emplea en máquinas que emplean IP multicast. El IP multicast es una variante de IP que permite emplear datagramas con múltiples destinatarios.

También en este nivel tenemos una serie de protocolos que se encargan de la resolución de direcciones:

- **ARP (Address Resolution Protocol).-** Cuando una máquina desea ponerse en contacto con otra conoce su dirección IP, entonces necesita un mecanismo dinámico que permite conocer su dirección física. Entonces envía una petición ARP por broadcast (o sea a todas las máquinas). El protocolo establece que sólo contestará a la petición, si esta lleva su dirección IP. Por lo tanto sólo contestará la máquina que corresponde a la dirección IP buscada, con un mensaje que incluya la dirección física. El software de comunicaciones debe mantener una caché con

los pares IP-dirección física. De este modo la siguiente vez que hay que hacer una transmisión a esa dirección IP, ya conoceremos la dirección física.

- **RARP (Reverse Address Resolution Protocol).-** A veces el problema es al revés o sea, una máquina sólo conoce su dirección física y desea conocer su dirección lógica. Esto ocurre, por ejemplo, cuando se accede a Internet con una dirección diferente, en el caso de PC que acceden por modem a Internet y se le asigna una dirección diferente de las que tiene el proveedor sin utilizar. Para solucionar esto se envía por broadcast una petición RARP con su dirección física, para que un servidor pueda darle su correspondiente IP.
- **BOOTP (Bootstrap Protocol).-** El protocolo RARP resuelve el problema de la resolución inversa de direcciones, pero para que pueda ser más eficiente, enviando más información que meramente la dirección IP, se ha creado el protocolo BOOTP. Este además de la dirección IP del solicitante, proporciona información adicional, facilitando la movilidad y el mantenimiento de las máquinas.

Nivel de Enlace.- Este nivel se limita a recibir datagramas del nivel superior (nivel de red) y transmitirlo al hardware de la red. Pueden usarse diversos protocolos.

La interconexión de diferentes redes genera una red virtual en la que las máquinas se identifican mediante una dirección de red lógica. Sin embargo a la hora de transmitir información por un medio físico se envía y se recibe información de direcciones físicas.

Un diseño eficiente implica que una dirección lógica sea independiente de una dirección física, por lo tanto es necesario un mecanismo que relacione las direcciones lógicas con las direcciones físicas. De esta forma podremos cambiar nuestra dirección lógica IP

conservando el mismo hardware, del mismo modo podremos cambiar una tarjeta de red, la cual contiene una dirección física, sin tener que cambiar nuestra dirección lógica IP.

2.5.2.2. Funcionamiento del Modelo de Referencia TCP/IP

Una red TCP/IP transfiere datos mediante el ensamblaje de bloques de datos en paquetes, cada paquete comienza con una cabecera que contiene información de control; tal como la dirección del destino, seguido de los datos. Cuando se envía un archivo por la red TCP/IP, su contenido se envía utilizando una serie de paquetes diferentes. El Internet Protocol (IP), un protocolo de la capa de red, permite a las aplicaciones ejecutarse transparentemente sobre redes interconectadas. Cuando se utiliza IP, no es necesario conocer que hardware se utiliza, por tanto esta corre en una red de área local.

El Transmission Control Protocol (TCP); un protocolo de la capa de transporte, asegura que los datos sean entregados, que lo que se recibe, sea lo que se pretendía enviar y que los paquetes que sean recibidos en el orden en que fueron enviados. TCP terminará una conexión si ocurre un error que haga la transmisión fiable imposible.

2.5.3. Estructura Cliente/Servidor

2.5.3.1. Sockets

- Para que dos PCs hablen entre sí, hay que establecer un canal de comunicación, esto es lo que se llama un "SOCKET".
- Asignación dinámica de direcciones de puertos.

- Para establecer dicho canal (o dicho SOCKET), es necesario una dirección y puerto origen y una dirección y puerto destino. Es decir, esta combinación de 2 direcciones y dos puertos es lo que se llama un socket.

Los sockets constituyen la interfaz de comunicaciones en red. Van a permitir la comunicación entre procesos que se están ejecutando en máquinas remotas, pero integradas en una red. Los sockets corresponden al nivel 4 (transporte) del modelo OSI.

Para establecer una conexión mediante sockets es necesario tener en cuenta dos características de los mismos: la *familia* de sockets que se está utilizando, y el *tipo de conexión*.

La **familia** de sockets indica la familia o tipo de direcciones que se utilizarán para especificar las diferentes máquinas. Las más comunes son la familia de protocolos internos de Unix, que se utiliza para comunicar procesos que se ejecutan en la misma máquina, y la familia de protocolos *Internet*, que se utiliza para la transmisión de protocolos como el TCP y el UDP. Existen también otras familias menos frecuentes.

El **tipo de conexión** puede ser de circuito virtual (orientado a conexión) o de tipo datagrama (no orientado a conexión). En el primero se crea una conexión permanente entre los dos ordenadores a conectar, de forma que una vez realizada la conexión todos los mensajes recorrerán el circuito establecido. En el segundo no se crea una conexión permanente. Los mensajes pueden seguir rutas distintas y no se garantiza que lleguen en el mismo orden en que fueron enviados.

2.5.3.2. Modelo Cliente-Servidor

Cuando el administrador necesita interactuar con un dispositivo de hardware específico, el software de administración se apega al modelo cliente-servidor convencional: un programa de aplicación de la computadora del administrador actúa como cliente y un programa de aplicación en el dispositivo de red actúa como servidor. El cliente de la máquina del administrador se vale de protocolos de transportación comunes (por ejemplo, TCP o UDP) para establecer una comunicación con el servidor. Luego ambos intercambian solicitudes y respuestas de acuerdo con el protocolo de administración.

Para evitar confusiones entre los programas de aplicación llamados por los usuarios y las aplicaciones reservadas a los administradores de red, los sistemas de administración de red evitan los términos cliente y servidor. En su lugar, la aplicación cliente del administrador se llama administrador y la que se ejecuta en un dispositivo de red se llama agente.

El modelo *cliente-servidor* es el más utilizado para construir aplicaciones de red. El objetivo de cliente/servidor es ofrecer una alternativa de diversidad de plataformas de proceso, aplicaciones y configuraciones que van a implementar los usuarios.

El proceso cliente/servidor no es en sí mismo un producto, sino más bien un estilo y un método de diseño y construcción de aplicaciones de proceso.

Una arquitectura cliente/servidor implica cuatro elementos básicos:

- Plataformas de proceso programables.
- Separación entre función/proceso de aplicación.
- Comunicación entre procesos.
- Enfoque "solicitante/proveedor de servicios".

Las aplicaciones en la arquitectura cliente/servidor están funcionalmente separadas en distintos procesos y utilizan comunicación solicitante/proveedor de servicios. Los clientes pueden ser cualquier tipo de sistemas inteligentes, desde PCs a sistemas propietarios, y lo mismo pueden ser los servidores.

Cliente.- Es una entidad programable que maneja parte de una aplicación que no es compartida por otros clientes y que debe solicitar servicio e interactuar con una parte de la aplicación que reside en una función "servidor programable". La relación del cliente con el servidor es necesaria para ejecutar esa aplicación en su totalidad.

La función servidor es compartida por clientes y a ellos le ofrece servicios. Las aplicaciones cliente/servidor pueden tener diferentes controles: centrado en el cliente o centrado en el host .

Para el caso del control del cliente, éste tiene el control absoluto de la ejecución de la aplicación y los recursos compartidos son controlados por el servidor.

Para el caso del control centrado en el host, éste conoce todas las opciones de que disponen todos los usuarios en todo momento, las actividades de visualización, ejecución de programas y gestión de recursos.

Servidor.- Es un proceso que está continuamente ejecutando en uno de los ordenadores de la red. El servidor realizará una determinada tarea o *servicio* cuando se lo solicite un proceso **cliente**. El proceso cliente se puede ejecutar en cualquiera de los ordenadores de la red, incluido en el que está el servidor.

Existen dos tipos de servidores: **interactivos, concurrentes**.

Un servidor **interactivo** se encuentra a la espera de peticiones por parte de los clientes, y cuando llega una de estas peticiones, la atiende el mismo. Esto es útil en el caso de que el servicio no sea demasiado complicado ya que, en caso contrario, el servidor podría recibir más peticiones mientras está sirviendo la primera, lo cual originaría retardos.

Para evitar este problema están los servidores **concurrentes** que, cuando reciben una petición de servicio, crean un nuevo proceso que se encargará de atenderla. De esta manera el servidor puede estar recogiendo las peticiones de los clientes con mayor velocidad.

La elección entre uno y otro tipo depende por tanto de la complejidad del servicio que proporciona el servidor y del número de peticiones que se espera que vayan a producirse.

La evolución de las arquitecturas cliente/servidor es el resultado de cambios que han tenido lugar entre los requerimientos de los clientes, en tecnología y en la competencia.

La estructura de los procesos cliente y servidor utilizando los servicios del sistema 4.3BSD para una comunicación **orientada a conexión** se muestra en la **(Figura 2.12.)**.

Cuando se utiliza una comunicación mediante **datagramas** la estructura de ambos procesos ha de ser diferente, ya que no es necesario establecer la conexión y se muestra en la **(Figura 2.13.)**.

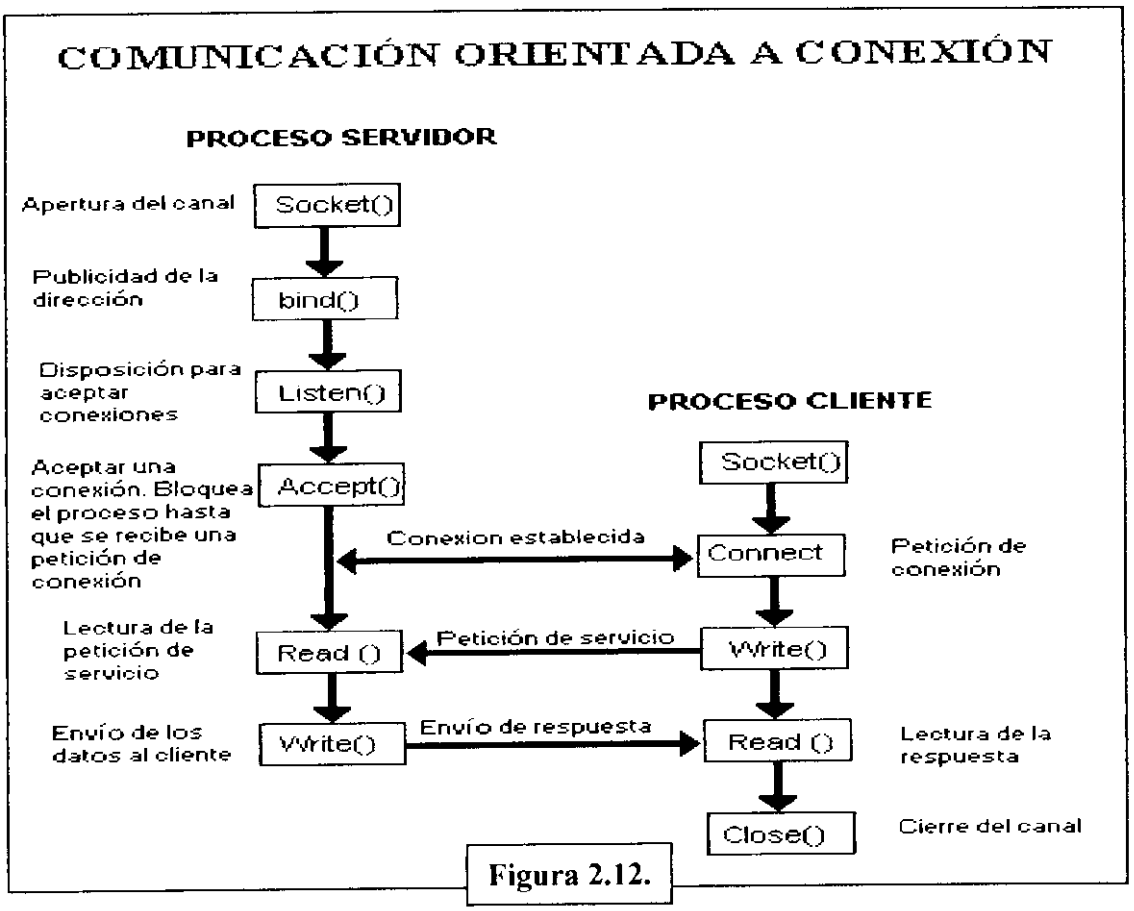


Figura 2.12.

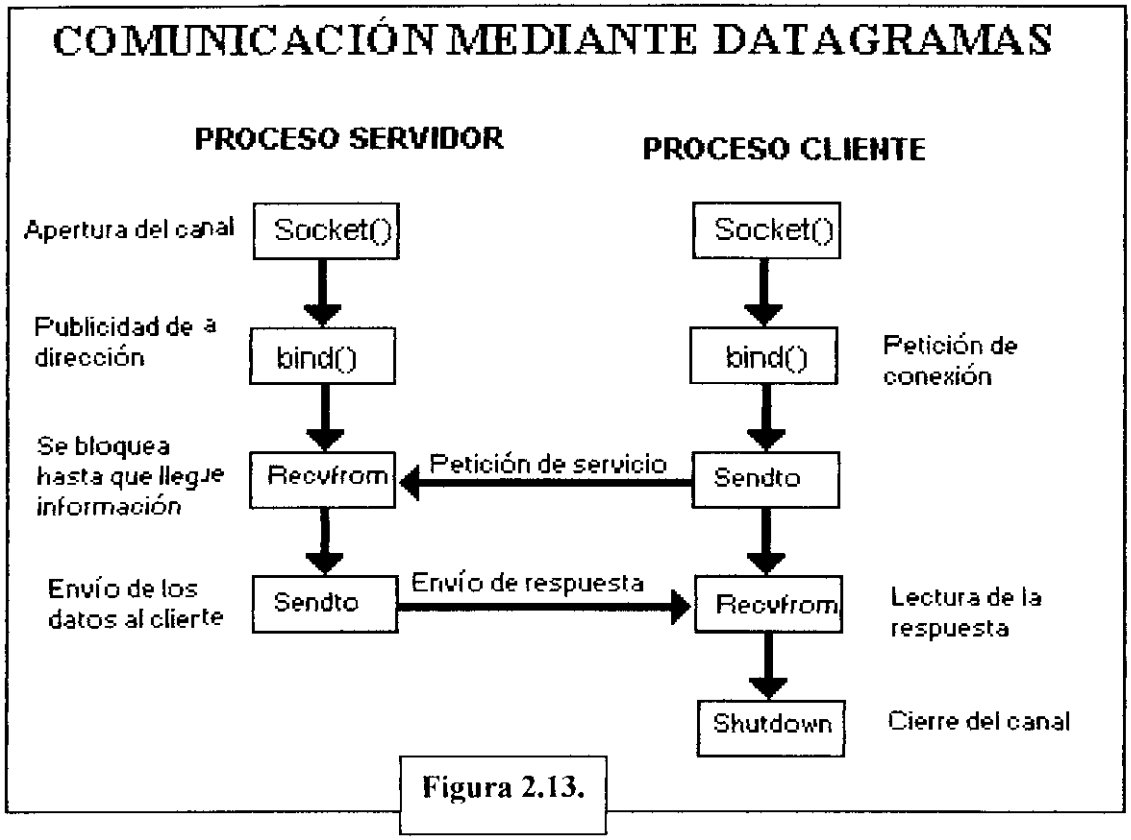


Figura 2.13.

Internet funciona con la estrategia **"Cliente/Servidor"**, lo que significa que en la Red hay ordenadores **Servidores** que dan una información concreta en el momento que se solicite y por otro lado están los ordenadores que piden dicha información, los llamados **Clientes**.

2.6. RED DE LA ESCUELA DE INGENIERÍA DE SISTEMAS DE LA PUCESA

2.6.1. Información de la Red de la Escuela de Ingeniería de Sistemas de la PUCESA

2.6.1.1. Tipo de red

Los tipos de redes que en la actualidad mantiene la Escuela de Ingeniería de Sistemas de la Pontificia Universidad Católica del Ecuador Sede Ambato es la Red de Área Local (Local Area Network, LAN), además también trabaja con una Red Intranet y a su vez dicha Intranet con Internet.

La Red de Área Local (Local Area Network, LAN) de la Escuela de Ingeniería de Sistemas de la PUCESA ha sido creada para responder a las necesidades de tratamiento de información a pequeñas distancias.

Sus principales características son:

- Utilizar una red de transmisión privada para el entorno que se pretenda cubrir.
- A ellas pueden conectarse un gran número de dispositivos que compartirán recursos comunes (discos, impresoras, etc.).
- Pueden llegar a distancias de unos pocos kilómetros; una sala, una planta de un edificio, todo el edificio, etc.
- Velocidad entre uno y cien Mbps.
- Permiten la conexión a otros medios a través de pasarelas (Gateways).

2.6.1.2. Topología de la Red

Topología de red es la forma en que se distribuyen los cables de la red para conectarse con el servidor y con cada una de las estaciones de trabajo.

La topología de una red es similar a un plano de la red dibujado en un papel, ya que se pueden tender cables a cada estación de trabajo y servidor de la red.

La topología determina donde pueden colocarse las estaciones de trabajo, la facilidad con que se tenderá el cable y el corte de todo el sistema de cableado.

La flexibilidad de una red en cuanto a sus necesidades futuras se refiere, depende en gran parte de la topología establecida.

Por ésta razón la topología escogida en la Red LAN de la Escuela de Ingeniería de Sistemas de la PUCESA es la topología estrella. En una topología de estrella, cada computadora está conectada a un concentrador (o un hub) ubicado centralmente.

2.6.1.3. Componentes del Hardware de la Red de la Escuela de Ingeniería de Sistemas de la PUCESA

Los cinco componentes básicos del hardware de la Red de la Escuela de Ingeniería de Sistemas de la PUCESA son:

- Los Servidores.
- Las Estaciones de Trabajo.

- Tarjetas Interfaces de la Red (Network Interface Cards, NIC).
- Los Cables (incluyendo los conectores y otros componentes especiales).
- Concentradores (Hubs).

Los Servidores

Existen servidores dedicados que se encargan de administrar los recursos de la red y el flujo de información en la Escuela de Ingeniería de Sistemas de la PUCESA, son computadoras de alto rendimiento en cuanto a velocidad y procesamiento y gran capacidad en disco duro.

Las Estaciones de Trabajo

Existen 2 laboratorios donde están ubicadas las estaciones de trabajo, además en otros departamentos también se ubican otras estaciones, dichas estaciones de trabajo están conectadas al servidor a través de las tarjetas interfaces de la red, los cables y los concentradores (Hubs). Las estaciones son de marcas comunes IBM, COMPAQ y clones con discos duros, con torres de floppy y CD-ROM.

Tarjetas Interfaces de la Red (Network Interface Cards, NIC)

Estas tarjetas son las que permiten la interfaz entre los servidores de la Red de la Escuela de Ingeniería de Sistemas de la PUCESA y las estaciones ubicadas en el Laboratorio IBM, Laboratorio COMPAQ y demás Departamentos, a través de los circuitos internos, el protocolo de comunicación (el protocolo es la forma en que transitan los datos a través de la red) y los comandos requeridos para el trabajo en el ambiente de red.

Las placas para 10 BaseT, que es la red más difundida hoy con el cableado estructurado, soporta 10 o 100 Mbit/seg. y es exactamente igual a las placas de salida coaxial pero poseen un conector RJ45. Muchos fabricantes proveen placas compatibles con coaxial y RJ45 al mismo tiempo. Existen diversos tipos en función de su interface con la PC (bus ISA, PCI, etc.).

Los Cables

Una vez que tenemos los servidores, las estaciones de trabajo y las placas de red, requerimos interconectar todo el conjunto. El tipo de cable utilizado depende de muchos factores, que se mencionarán a continuación:

- Cada tipo de cable o método tiene sus ventajas y desventajas. Algunos son propensos a interferencias, mientras otros no pueden usarse por razones de seguridad.
- La velocidad y longitud del tendido son otros factores a tener en cuenta el tipo de cable a utilizar.

La Escuela de Ingeniería de Sistemas de la PUCESA utiliza el sistema de cableado estructurado o abierto, esto se define por oposición a los problemas del cableado no estructurado, no estándar o cerrado, o propietario de un determinado fabricante.

El cableado estructurado UTP (Unshielded Twisted Pair – par trenzado) en categoría 5 es el tipo de cableado más solicitado hoy en día y utilizado en dicha red.

Cable UTP.- Consiste en dos hilos de cobre trenzado, aislados de forma independiente y trenzados entre sí. El par está cubierto por una capa aislante externa.

Entre sus principales ventajas tenemos:

- Es una tecnología bien estudiada.
- No requiere una habilidad especial para instalación.
- La instalación es rápida y fácil.
- La emisión de señales al exterior es mínima.
- Ofrece alguna inmunidad frente a interferencias, modulación cruzada y corrosión.

Concentradores (Hubs)

El concentrador es un dispositivo de hardware con varios puertos, y se puede conectar un conector de cable de red en uno de ellos. En la Red de la Escuela de Ingeniería de Sistemas de la PUCESA existen Hubs de diferentes marcas, los mismos que poseen puertos de 16 y 24 salidas respectivamente.

La marca del Hub de 24 puertos y la ubicación de dichos Hubs se muestra en la **(Tabla 2.6. HUB 24 PUERTOS).**

(Tabla 2.6. HUB 24 PUERTOS)
MARCA: • Centre COM 3624 TRS IEEE 802-3/ETHERNET 10 BASE-T 24 PORT HUB/REPEATER
HUB en OFICINA: • LABORATORIO COMPAQ • DIRECCIÓN DE SISTEMAS (Ing. Telmo Viteri) • LABORATORISTA (Ing. Santiago Acurio)

La marca del Hub de 16 puertos y la ubicación de dicho Hub se muestra en la **(Tabla 2.7. HUB 16 PUERTOS).**

(Tabla 2.7. HUB 16 PUERTOS)
MARCA: Accton ChetaHub Classic-2041
HUB en OFICINA: • LABORATORIO IBM

2.6.1.4. Componentes del Cableado Estructurado

La Red de la Escuela de Ingeniería de Sistemas de la PUCESA posee los siguientes componentes y sus marcas:

- **Ruteadores (Routers).-** La marca del Router es Pico Router Prt de 9.6 K – 2 Mbps.
- **Switch.-** La marca del Switch Central de 24 puertos y la ubicación de dicho Switch se muestra en la **(Tabla 2.8. SWITCH CENTRAL 24 PUERTOS).**
- **Patch Panel.-** Que se encuentra en el Rack.

(Tabla 2.8. SWITCH CENTRAL 24 PUERTOS)
MARCA: • Marca Centre COM 3726 XL 10 BASE-T ETHERNET SWITCH WITH FAST ETHERNET 100 BASE ACCESS
SWITCH CENTRAL en OFICINA: • Ing. Diego Santacruz

2.6.2. Gráficos de la Red de la Escuela de Ingeniería de Sistemas de la PUCESA

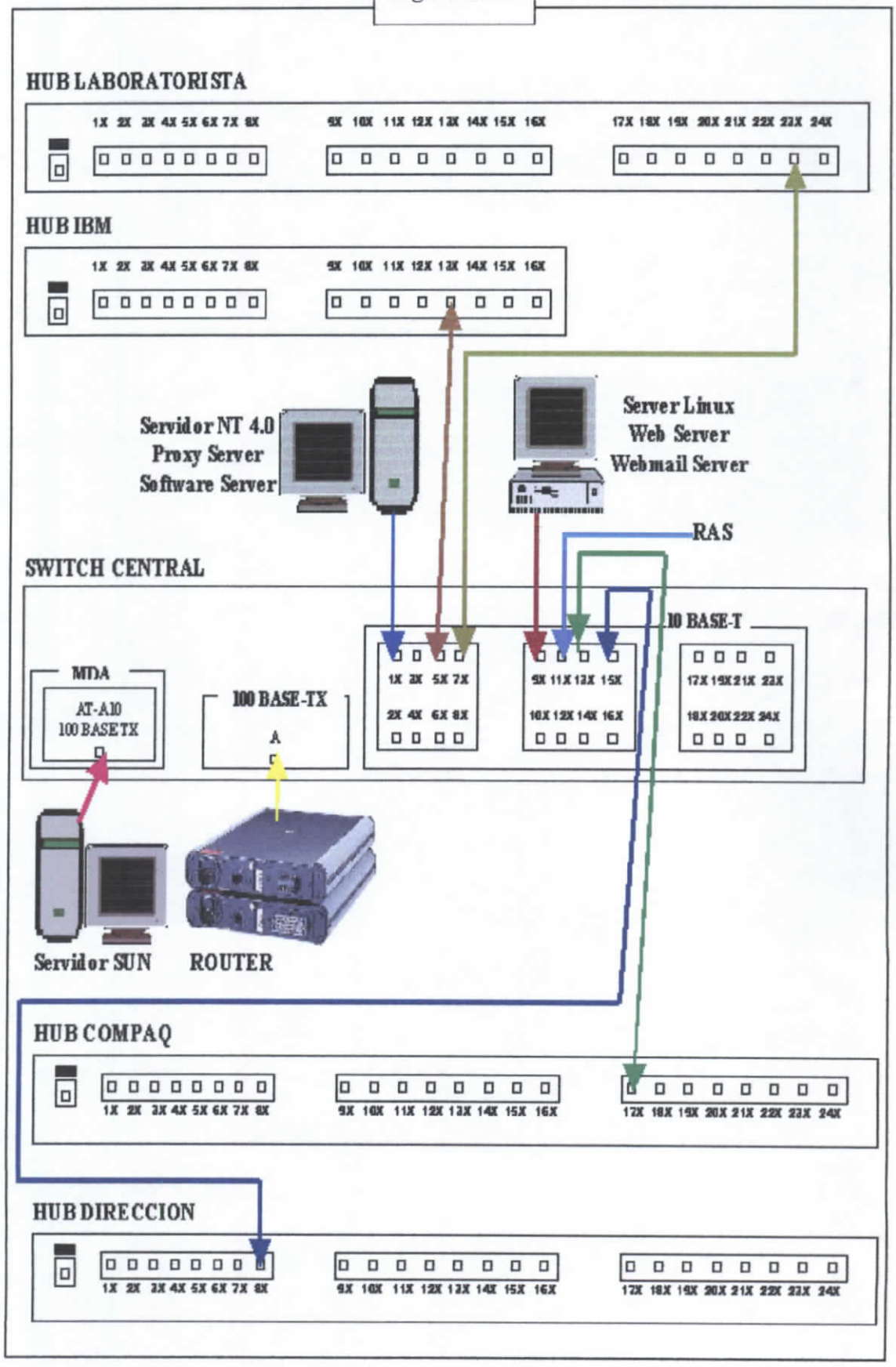
Para comprender de mejor manera de cómo está constituida la Red de la Escuela de Ingeniería de Sistemas de la PUCESA trataremos de explicar en forma gráfica.

Comenzando dichos gráficos en la **(Figura 2.14. SWITCH CENTRAL)** y para una explicación más detallada se especifica en las tablas los colores que identifican de que puerto del Switch se conecta a los diferentes Servidores, algunos dispositivos especiales y a que puerto de los Hubs en la **(Tabla 2.9. SWITCH CENTRAL)**.

(Tabla 2.9. SWITCH CENTRAL)

PUERTO SWITCH	COLOR	PUERTO HUB	HUB	EQUIPOS
AT-A10 100 BASE TX				Servidor SUN
A				ROUTER
1X				Servidor NT 4.0
5X		13X	IBM	
7X		23X	LABORATORISTA	
9X				Servidor Linux Web Server
11X				RAS
13X		17X	COMPAQ	
15X		8X	DIRECCIÓN	

Figura 2.14.

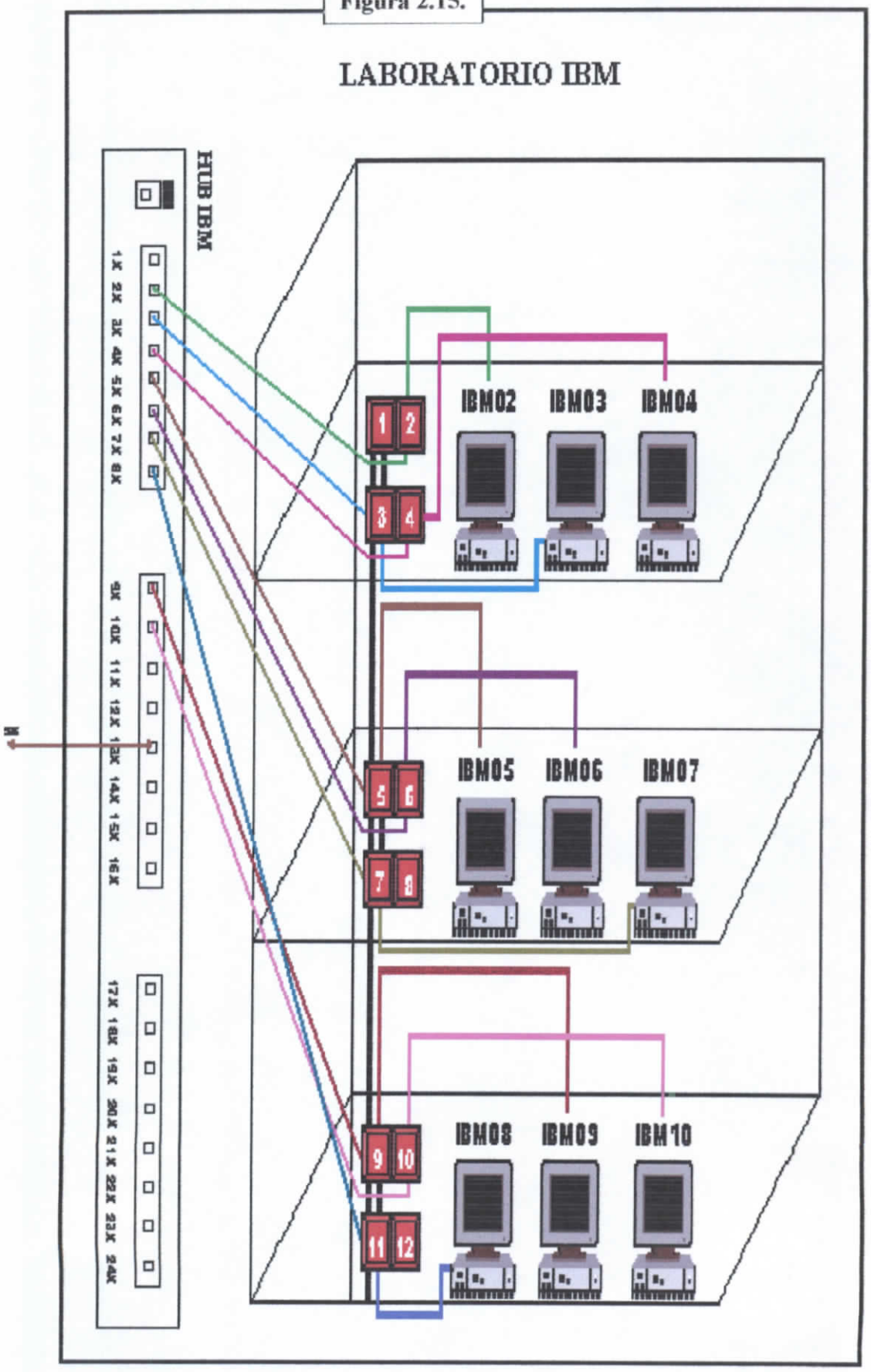

HUB LABORATORISTA
1X 2X 3X 4X 5X 6X 7X 8X
9X 10X 11X 12X 13X 14X 15X 16X
17X 18X 19X 20X 21X 22X 23X 24X
HUB IBM
1X 2X 3X 4X 5X 6X 7X 8X
9X 10X 11X 12X 13X 14X 15X 16X
Servidor NT 4.0
Proxy Server
Software Server
Server Linux
Web Server
Webmail Server
RAS
SWITCH CENTRAL
10 BASE-T
MDA
AT-A10
100 BASETX
100 BASE-TX
A
1X 3X 5X 7X
2X 4X 6X 8X
9X 11X 13X 15X
10X 12X 14X 16X
17X 19X 21X 23X
18X 20X 22X 24X
Servidor SUN
ROUTER
HUB COMPAQ
1X 2X 3X 4X 5X 6X 7X 8X
9X 10X 11X 12X 13X 14X 15X 16X
17X 18X 19X 20X 21X 22X 23X 24X
HUB DIRECCION
1X 2X 3X 4X 5X 6X 7X 8X
9X 10X 11X 12X 13X 14X 15X 16X
17X 18X 19X 20X 21X 22X 23X 24X

En la **(Figura 2.15. HUB IBM)** se muestra la distribución de dicho Hub y en la **(Tabla 2.10. HUB IBM)** se detallan los colores identificando de que puerto del Hub se conecta a los puntos y dichos puntos a las estaciones; además la flecha que no se conecta a ningún punto ni estación es el puerto que se conecta directamente al Switch Central (puerto 5X del Switch con el puerto 13X del Hub IBM).

(Tabla 2.10. HUB IBM)

PUERTO SWITCH	COLOR	PUERTO HUB	COLOR	PUNTOS	COLOR	EQUIPOS
		2X		**2**		IBM02
		3X		**3**		IBM03
		4X		**4**		IBM04
		5X		**5**		IBM05
		6X		**6**		IBM06
		7X		**7**		IBM07
		8X		**11**		IBM08
		9X		**9**		IBM09
		10X		**10**		IBM10
5X		13X				

Figura 2.15.


LABORATORIO IBM
HUB IBM
IBM02
IBM03
IBM04
IBM05
IBM06
IBM07
IBM08
IBM09
IBM10
1
2
3
4
5
6
7
8
9
10
11
12

En la **(Figura 2.16. HUB LABORATORISTA)** se muestra la distribución de dicho Hub y en la **(Tabla 2.11. HUB LABORATORISTA)** se detalla los colores identificando de que puerto del Hub se conecta a la estación y la flecha que no se conecta a ninguna estación es el puerto que se conecta directamente al Switch Central (puerto 7X del Switch con el puerto 23X del Hub Laboratorista).

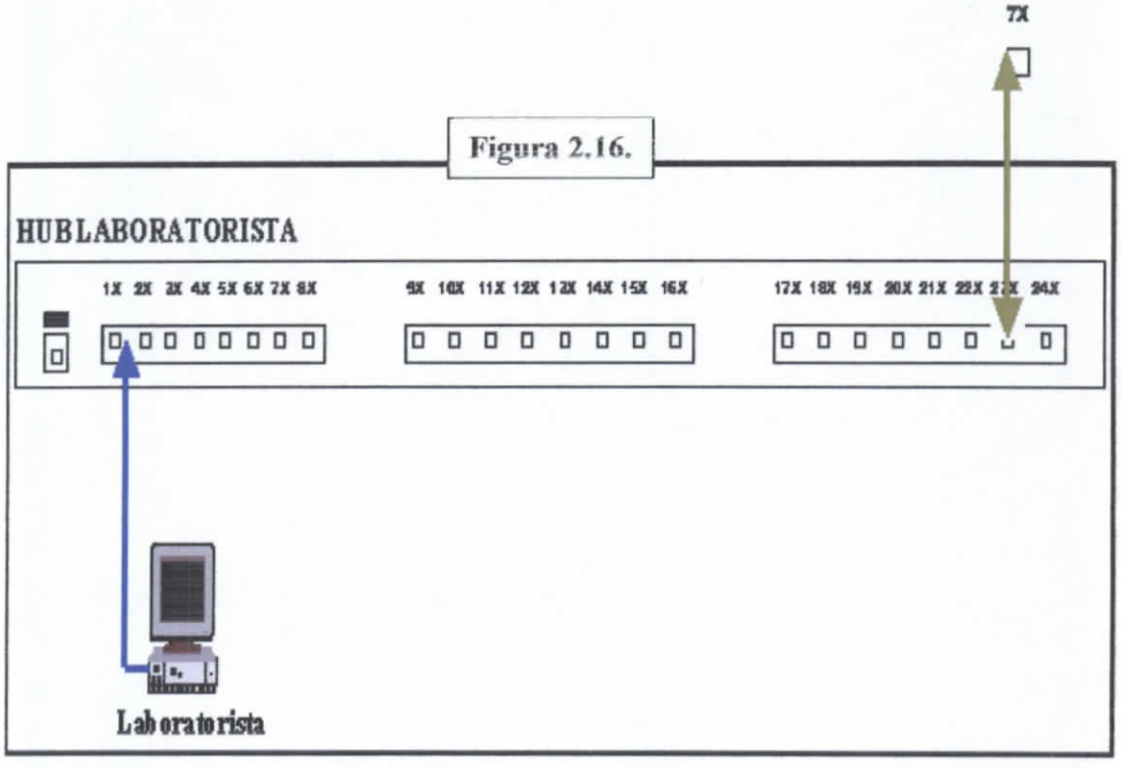


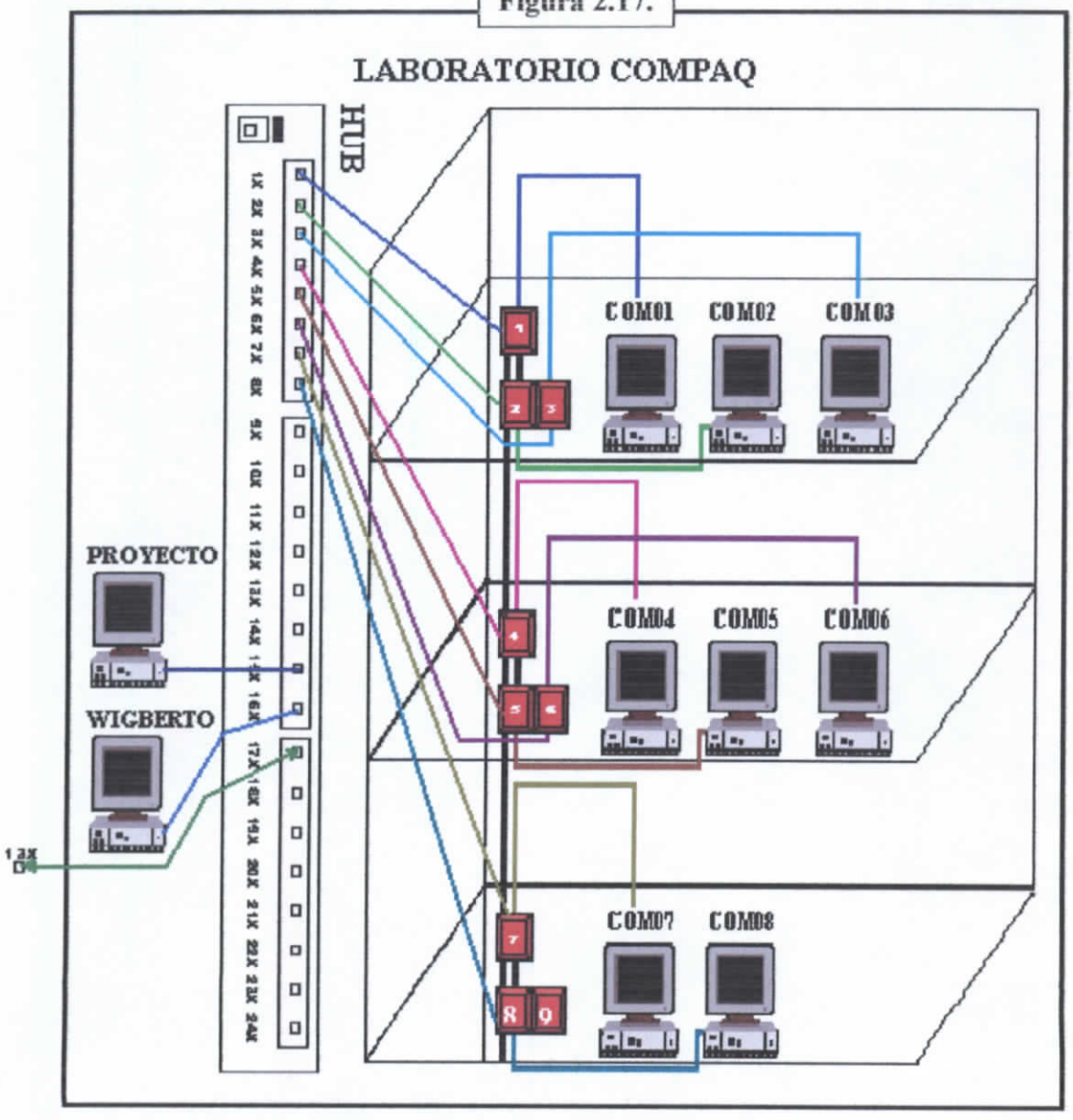
Figura 2.16.

(Tabla 2.11. HUB LABORATORISTA)

PUERTO SWITCH	COLORES	PUERTO HUB	COLORES	EQUIPOS
		1X	———	Laboratorista
7X	———	23X		

En la **(Figura 2.17. HUB COMPAQ)** se muestra la distribución de dicho Hub y en la **(Tabla 2.12. HUB COMPAQ)** se detallan los colores identificando de que puerto del Hub se conecta a los puntos y dichos puntos a las estaciones, y algunas estaciones se conectan directamente al Hub; además la flecha que no se conecta a ningún punto ni estación es el puerto que se conecta directamente al Switch Central (puerto 13X del Switch con el puerto 17X del Hub Compaq).

Figura 2.17.



(Tabla 2.12. HUB COMPAQ)						
PUERTO SWITCH	**COLOR**	**PUERTO HUB**	**COLOR**	**PUNTOS**	**COLOR**	**EQUIPOS**
		1X		**1**		COM01
		2X		**2**		COM02
		3X		**3**		COM03
		4X		**4**		COM04
		5X		**5**		COM05
		6X		**6**		COM06
		7X		**7**		COM07
		8X		**8**		COM08
		15X				PROYECTO
		16X				WIGBERTO
13X		17X				
		19X				LORENA

En la **(Figura 2.18. HUB DIRECCIÓN)** se muestra la distribución de dicho Hub y en la **(Tabla 2.13. HUB DIRECCIÓN** se detallan los colores identificando de que puerto del Hub se conecta a las estaciones y la flecha que no se conecta a ninguna estación es el puerto que se conecta directamente al Switch Central (puerto 15X del Switch con el puerto 8X del Hub Dirección).

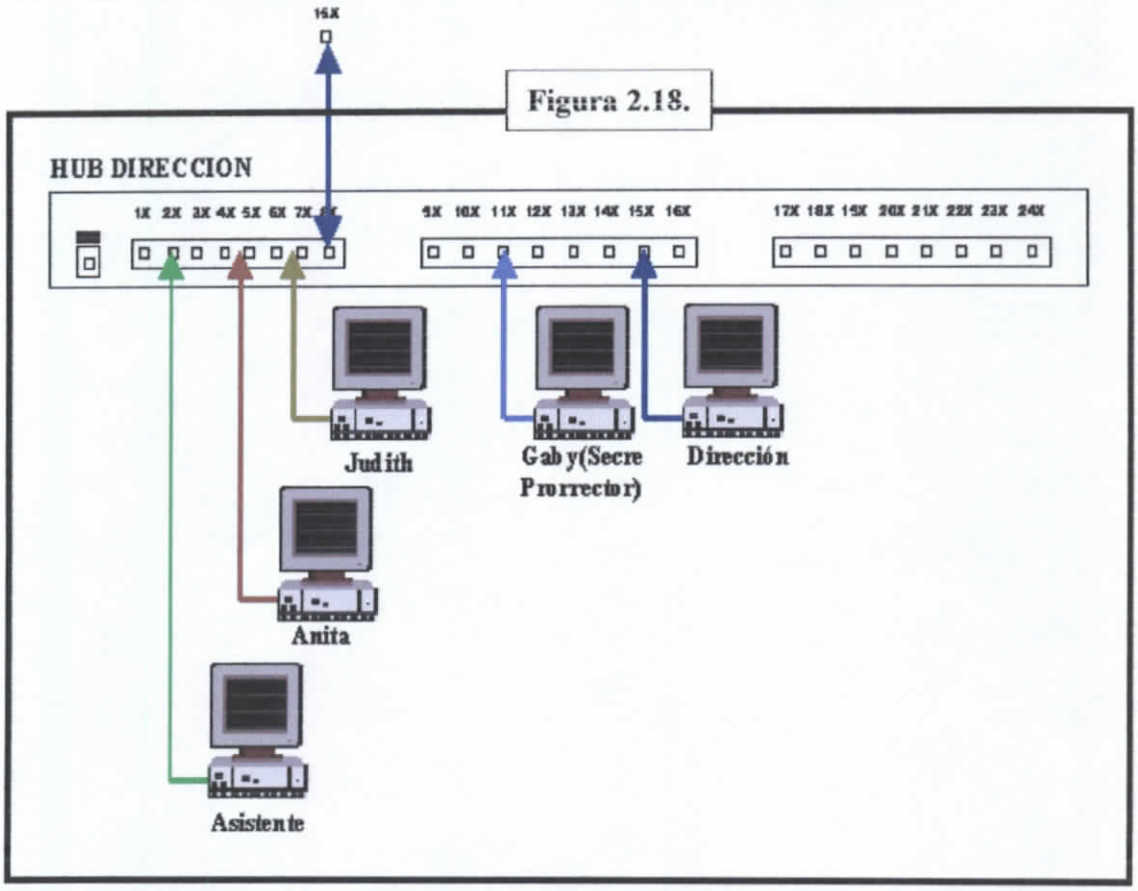


Figura 2.18.

(Tabla 2.13. HUB DIRECCIÓN)

PUERTO SWITCH	COLORES	PUERTO HUB	COLORES	EQUIPOS
		2X		Asistente
		5X		Anita
		7X		Judith
15X		8X		
		11X		Gaby(Secre Prorrector)
		15X		Dirección

CAPÍTULO III

AGENTES DE CAPTURA DE INFORMACIÓN

3.1. INTRODUCCIÓN

El propósito del presente capítulo es describir una explicación detallada de los agentes de captura de información. En este capítulo se encontrará conceptos sobre ¿Qué son programas agentes?, así como la definición de Agentes y Corrientes que dominan la Investigación sobre Agentes, ¿Cómo se captura información? y Seguridades en programas agentes.

3.2. ¿QUÉ SON PROGRAMAS AGENTES?

3.2.1. Agentes

Son aquellos programas que sirven para automatizar distintos tipos de tareas en una computadora o en internet.

La capacidad de registrar las actividades del usuario de un computador ("observarlo") y extraer de ellas los patrones respectivos ha llevado a concebir muchas aplicaciones, no sólo en el campo educacional.

Por ahora, cada agente se especializa en un determinado tipo de razonamiento, pero los investigadores desean poder integrar agentes que combinen diferentes métodos y dispongan de una especie de sentido común.

3.2.2. Corrientes que dominan la Investigación sobre Agentes

Dos corrientes dominan actualmente la investigación sobre agentes:

La de Interfaces Agentes

Software que ayuda activamente al usuario para operar interfaces interactivas.

La de Agentes Autónomos

Software que actúa sin la intervención del usuario y opera concurrentemente, o mientras el usuario realiza otras acciones.

La disertación, "Sistema de Control de Estaciones de Trabajo en la Red de la Escuela de Ingeniería de Sistemas de la PUCESA", consiste en un programa agente autónomo que es capaz de satisfacer su propia agenda sin la intervención del usuario, u otros agentes gracias a la capacidad de percibir y actuar en su medio ambiente.

Este programa opera automáticamente, cuando inicie la sesión de modo que cualquier usuario está bajo la supervisión de este programa agente, que envía información a una base de datos MYSQL.

Esta información comprende:

- Dirección IP.
- Nombre de la PC.
- Usuario.
- Fecha y Hora.
- Procesos.
- Ventanas.

Dirección IP.- En una red TCP/IP los ordenadores se identifican mediante un número que se denomina **dirección IP**. Esta dirección ha de estar dentro del rango de direcciones asignadas al organismo o empresa a la que pertenece, éstos rangos son concedidos por un organismo central de Internet, el **NIC** (Network Information Center).

Nombre de la PC.- Identifica el nombre del equipo ante otros usuarios de la red.

Usuario.- Cualquier individuo que interactúa con la computadora al nivel de aplicación.

Fecha y Hora.- La fecha y hora que se obtiene del sistema.

Procesos.- Cuando un programa está cargado en memoria para ser ejecutado se lo denomina "proceso", y si su computadora tiene una sola CPU, entonces solo un proceso puede estar ejecutándose en un determinado momento.

Varios procesos se ejecutarán al mismo tiempo gracias a las capacidades multi-tarea del SO.

¿Cómo funciona ésta multitarea? Bueno, supongamos que tenemos cinco procesos (A, B, C, D y E). El SO (Sistema Operativo) le asigna un período muy corto de tiempo a un proceso (por ej. el A) y luego cuando se acaba ese tiempo, el proceso es detenido temporalmente y el SO le da un período muy corto de tiempo a otro proceso (por ej. el B), y así sucesivamente con el resto de los procesos (por ej. el C, D y E), hasta que vuelve a ser el turno entonces del primer proceso (el A, en nuestro ejemplo), y así el ciclo se repite una y otra vez, de modo tal que en definitiva los procesos se ejecutan en un brevísimo lapso de tiempo, se detienen (para permitir que los otros se ejecuten en un brevísimo lapso de tiempo también), se vuelven a ejecutar otro poco, se detienen, y así sucesivamente hasta que terminan, dando la sensación de una ejecución continua sin interrupciones, del mismo modo en que una película no es más que una serie de fotogramas estáticos que cambian tan rápidamente que percibimos la ilusión de un movimiento continuo. De esta forma, parece como si varios procesos estuvieran corriendo al mismo tiempo.

Las capacidades multitarea de Windows no se limitan a procesos, sino que dentro de un mismo proceso también es posible tener partes del mismo ejecutándose "al mismo tiempo", es decir, en distintos "hilos de ejecución" (threads). Normalmente usamos uno solo (el hilo principal de la VCL (Visual Library Component)), pero podemos crear y ejecutar otros hilos dentro una aplicación. En este caso, se dice que la aplicación es multi-hilos.

Ventanas.- Son una área de visión separada en una pantalla de presentación, provista por software. Los sistemas operativos pueden proveer ventanas múltiples en la pantalla,

permitiendo al usuario mantener varios programas de aplicación activos y visibles al mismo tiempo.

Los programas de aplicación individuales también pueden proveer múltiples ventanas, proporcionando la capacidad de ver más de un documento, hoja de cálculo o archivo de datos.

3.3. ¿CÓMO CAPTURAR INFORMACIÓN?

El proceso por el cual el programa agente colecciona información se denomina "captura". Por defecto recoge:

La dirección IP, nombre de la PC, usuario, fecha y hora, procesos, ventanas, en un "buffer de captura independientes", que es un área de almacenamiento reservada en memoria que van a ser grabados en una base de datos MYSQL.

El presente proyecto está designado a funcionar en las plataformas de desarrollo Windows y Linux. Los dos SO son totalmente diferentes por sus características.

En **Windows**

Para acceder a los datos desde la aplicación se utiliza las llamadas al WINDOWS API (Interfaz de programa de aplicación). Lenguaje y formato utilizados por un programa para comunicarse con otro programa.

Para conocer el nombre, IP que identifica la máquina que ejecuta nuestro programa en una red, podemos apelar a la función API de Windows gethostbyName que nos devuelve el nombre, IP NetBIOS de la computadora local. Esta función encapsula la llamada a esta API para devolver el nombre de la máquina como una cadena.

La siguiente función encapsula la llamada a esta API para devolver el nombre de usuario como una cadena. Si necesitamos conocer el nombre con el que el usuario del sistema a ingresado para iniciar su sesión en Windows, podemos usar la función API de Windows GetUserName.

La API **FindWindow** busca ventanas concretas que pueden ser una aplicación o no y lo mismo pasa con **EnumWindows**, que busca todas las ventanas TopLevel, pero hay aplicaciones que pueden tener más de una y ciertamente las hay a montones que no tienen ninguna ventana asociada. Esta aplicación usa un simple código que utiliza la API **EnumWindows** para conocer las ventanas abiertas en Windows, aunque éstas estén ocultas.

En **Linux**

Para acceder a los datos desde la aplicación se utiliza CLX METHODS (Component Library for Cross Platform), llamadas al SYSTEM LIBRARIES (Sistema de librerías), para comunicarse con Linux.

El componente **Sockets** para obtener la dirección IP, nombre.

La llamada a la librería **SysUtils** para obtener el usuario activo.

La llamada a la librería **Libc** para listar los procesos.

La llamada a librería **Xlib** para listar las ventanas.

3.4. SEGURIDADES EN PROGRAMAS AGENTES

Como los administradores de todos los sistemas. Los de la PUCESA poseerán una herramienta para controlar que "todo vaya bien", si los procesos son los normales o si hay movimientos sospechosos, por ejemplo que un usuario esté recurriendo a vías de acceso para las cuales no está autorizado. Todos los movimientos del sistema son registrados en tablas, que los operadores revisan diariamente.

La información de nuestro objeto de estudio se salvaguarda con las siguientes seguridades:

En **Windows**

- Oculta la aplicación de la lista de tareas (Diálogo CTRL+ALT+SUPR).
- Oculta la aplicación de la barra de tareas.

En **Windows** y **Linux**

- El programa agente instalado en la PC no puede ser desactivado a menos que sea la persona encargada de hacerlo con su cuenta y password asignados.

CAPÍTULO IV

BITÁCORAS O ARCHIVOS DE AUDITORÍA (AUDIT LOGS)

4.1. INTRODUCCIÓN

El propósito del presente capítulo es realizar una explicación detallada de las bitácoras o archivos de auditoría (Audit Logs). En este capítulo se encontrará conceptos sobre la administración de bitácoras, así como la definición de bitácoras, propósitos, tipos y formato en el que se presentan los Audit Logs.

4.2. ADMINISTRACIÓN DE BITÁCORAS

4.2.1. Bitácoras

Las bitácoras están escritas en orden cronológico sobre cualquier tipo de tema, puede ir desde unas cuantas palabras a extensos párrafos. Pueden ser escritas por una o varias personas. En un principio las bitácoras servían para realizar vínculos a otros sitios junto con un comentario. Al pasar del tiempo el concepto se ha extendido a opiniones, emitir noticias e inclusive personas contando su vida diaria.

La seguridad y administración de un sistema operativo tiene en las bitácoras un gran aliado, ya que en ellas se registran los eventos que ocurren, es decir eventos que para el administrador pasarían inadvertidos sin el respaldo de las bitácoras.

Para una buena administración de bitácoras es necesario conocer 3 cosas:

- Conocer el propósito de cada bitácora.
- Conocer el formato en el que se presenta la información.
- Conocer los ataques propios de cada servicio.

4.3. PROPÓSITOS, TIPOS Y FORMATO EN EL QUE SE PRESENTAN LOS AUDIT LOGS.

4.3.1. Propósitos

A continuación se enumeran los propósitos de las bitácoras entre los cuales tenemos:

- Se debe considerar, el almacenamiento histórico de la información que se va acumulando en la vida de la base de datos.
- Debemos mantener siempre un tamaño razonable para la base de datos.
- Desarrollar una adecuada estrategia para almacenar información y eliminarla de la base de datos, nos asegura que estemos siempre dentro de la capacidad de almacenamiento físico que nuestro sistema dispone.
- Dado que el objetivo de la bitácora es registrar toda la información relacionada con el trabajo para ser utilizada en la elaboración de los reportes estadísticos, debe buscarse ante todo la claridad. En este sentido, no debe escatimarse en la repetición de frases o palabras, cuidando siempre la brevedad y precisión.

Un administrador de red al usar los auditlogs puede descubrir:

- ¿Qué máquinas ingresaron?
- ¿Con qué nombre de usuario se ingresaron?
- ¿En qué tiempo y hora se accedieron?
- ¿Qué archivos fueron utilizados?
- ¿Qué ventanas fueron utilizadas?

4.3.2. Tipos

Los temas pueden variar tanto como las actividades normales de cada una de las personas que usan bitácoras, por lo que surgen varios tipos de bitácoras:

- Bitácoras personales.
- Tecnológicas.
- Noticias.
- Medios.
- Poesía.
- Cuentos, etc.

La bitácora que escogimos para nuestro objeto de estudio es una bitácora personal, la misma que contendrá información de campos creados por el programador.

4.3.3. Formato

Como ejemplo de formato escogido para las bitácoras en ésta disertación tenemos: Un formato de la Tabla de movimientos(mvitacora) de la Base de Datos agentes.

Ip	Usuario	Fecha	Hora	Proceso	Ventana

En la **(Figura 4.1.)** y en la **(Figura 4.2.)** se muestra el encabezado y la información de los campos de la Tabla de movimientos(mvitacora), es decir su formato.

agentes / mvitacora: 8 Record(s)

mip	musuario	mfecha	mhora
192.168.0.1	vicente	2003-05-02	04:16:12
192.168.0.1	vicente	2003-05-02	04:17:12
192.168.0.1	vicente	2003-05-02	04:18:12
192.168.0.1	vicente	2003-05-02	04:21:35
192.168.0.1	vicente	2003-05-02	04:22:35
192.168.0.1	vicente	2003-05-02	04:23:35
192.168.0.1	vicente	2003-05-02	04:30:19
192.168.0.1	vicente	2003-05-02	04:43:01

Figura 4.1.

agentes / mvitacora: 8 Record(s)

mproceso	mventana
C:\WINDOWS\EXPLORER.EXE|C:\ARCHIVOS DE PROGRA	Barra de acceso directo de Office|Starting Babylon|-|5man
C:\WINDOWS\SYSTEM\MSGSRV32.EXE|C:\ARCHIVOS DE	Barra de acceso directo de Office|Starting Babylon|-|5man
C:\WINDOWS\SYSTEM\MSGSRV32.EXE|C:\ARCHIVOS DE	Barra de acceso directo de Office|Starting Babylon|-|5man
Procesos	kwin|knotify|ksmserver|kdesktop|Ejecutar comando|kil com
Procesos	kwin|knotify|ksmserver|kdesktop|Ejecutar comando|kil com
Procesos	kwin|knotify|ksmserver|kdesktop|Ejecutar comando|kil com
C:\WINDOWS\EXPLORER.EXE|C:\ARCHIVOS DE PROGRA	Open Project (Ctrl+F11)||Barra de acceso directo de|-|omar
Procesos	kwin|knotify|ksmserver|kdesktop|Ejecutar comando|kil com

Figura 4.2.

CAPÍTULO V

SOFTWARE DE DESARROLLO

5.1. INTRODUCCIÓN

- El propósito del presente capítulo es brindar una explicación detallada sobre el Software de Desarrollo, es decir información de los Sistemas Operativos, Lenguajes de Programación y Base de Datos. En este capítulo se encontrará sobre el Sistema Operativo Linux, así como las características del sistema, Windows, así como el administrador de Programas, el administrador de archivos y Windows 98, Lenguaje de Programación Delphi, así como sus características y conceptos fundamentales, Kylix, así como el Compilador y la RTL, Base de Datos Relacional MYSQL.

5.2. SISTEMAS OPERATIVOS

5.2.1. Linux

Linux es el Unix gratuito escrito por Linus Torvalds con la asistencia técnica de un grupo de hackers a través del Internet cumpliendo todas las características de los Sistemas Operativos UNIX actuales, y con todos los modernos avances en los Sistemas Operativos, con capacidad de multitarea real, memoria virtual, librerías compartidas, consumición escasa de recursos, manejo estupendo de la memoria, trabajo con TCP/IP.

Linux se ejecuta principalmente en máquinas basadas en PC's usando las facilidades del hardware de la familia del procesador 80386 (Segmentos TSS) que implementan ésta característica (y tienen capacidad multitarea). Portado también a otras arquitecturas que no son la familia Intel.

Linux puede convertir cualquier PC 386 o 486 en una estación de trabajo. Le pondrá todo el poder de UNIX en la punta de sus dedos. En los negocios ya se instala Linux en redes enteras, usando el sistema operativo para manejar registros financieros y de hospitales, un entorno de usuario distribuido, telecomunicaciones, etc. Universidades de todo el mundo usan Linux para dar cursos de programación y diseño de sistemas operativos. Y, por supuesto, entusiastas de los ordenadores de todo el mundo están usando Linux en casa, para programar, entretenerse, y conocerlo a fondo.

De todos los sistemas operativos "simil-UNIX", Linux es el de mayor crecimiento en el mercado de las PC Intel. Este sistema operativo (SO) de redes, multiproceso, multiusuario, escalable, estable, gratuito, de fuentes abiertos y estudiado en universidades de todo el mundo ha alcanzado un buen posicionamiento en el mercado de los servidores de Internet, y también es usado en sistemas de negocios y misiones críticas. Esto explica por qué Linux es ampliamente utilizado.

Sin embargo, aún está muy lejos de convertirse en el SO número uno del mundo, especialmente porque no le va tan bien en el ámbito de las PC de escritorio. Que en definitiva pueden traducirse en una cuestión de costos. Linux es gratis, pero la licencia no es el único costo a ser considerado. Podemos mencionar soporte técnico y capacitación por ejemplo, pero seguramente es el costo de las aplicaciones (ya sean

compradas o hechas a medida) el que evita que las empresas e individuos se pasen a Linux dado que no solo significaría la pérdida de todo el tiempo y dinero invertido en aplicaciones, sino que muchísimo más tiempo y dinero debería invertirse para tener aplicaciones hechas a medida de calidad similar, más el hecho que no habrá sustitutos adecuados para la mayoría de las aplicaciones comerciales (así como tampoco freeware/shareware/adware/postcardware/beerware/"etc-ware").

5.2.1.1. Características del Sistema

- El núcleo es capaz de emular por su cuenta las instrucciones del coprocesador 387, con lo que en cualquier 386 con coprocesador o sin él se podrán ejecutar aplicaciones que lo requieran.
- Linux soporta diversos sistemas de ficheros para guardar los datos. Algunos de ellos, como el ext2fs, han sido desarrollados específicamente para Linux.
- Otros sistemas de ficheros, como el Minix-1 o el de Xenix también están soportados. Y con el de MS-DOS se podrán acceder desde Linux a los disquetes y particiones en discos duros formateados con MS-DOS. Además, también soporta el ISO-9660, que es el estándar seguido en el formato de los CD-ROMs.
- Linux implementa todo lo necesario para trabajar en red con TCP/IP. Desde manejadores para las tarjetas de red más populares hasta SLIP/PPP, que permiten acceder a una red TCP/IP por el puerto serie. También se implementan PLIP (para comunicarse por el puerto de la impresora) y NFS (para acceso remoto a ficheros); y también se han portado los clientes de TCP/IP, como FTP, telnet, NNTP y SMTP.

- El núcleo de Linux ha sido desarrollado para utilizar las características del modo protegido de los microprocesadores 80386 y 80486. En concreto, hace uso de la gestión de memoria avanzada del modo protegido y otras características avanzadas. Cualquiera que conozca la programación del 386 en el modo protegido sabrá que éste modo fue diseñado para su uso en UNIX (o tal vez Multics). Linux hace uso de esta funcionalidad precisamente.
- Las páginas de los ejecutables son compartidas mediante la técnica copy-on-write, contribuyendo todo ello a reducir la cantidad de memoria requerida para las aplicaciones. El núcleo soporta ejecutables con paginación por demanda. Esto significa que sólo los segmentos del programa que se necesitan se cargan en memoria desde el disco.
- Con el fin de incrementar la memoria disponible, Linux implementa la paginación con el disco:
    - Puede tener hasta 256 megabytes de espacio de intercambio o "swap" en el disco duro. Cuando el sistema necesita más memoria, expulsará páginas inactivas al disco, permitiendo la ejecución de programas más grandes o aumentando el número de usuarios que puede atender a la vez. Sin embargo, el espacio de intercambio no puede suplir totalmente a la memoria RAM, ya que el primero es mucho más lento que ésta.
    - La memoria dedicada a los programas y a la cache de disco está unificada. Por ello, si en cierto momento hay mucha memoria libre, el tamaño de la cache de disco aumentará acelerando así los accesos.
- Los ejecutables hacen uso de las librerías de enlace dinámico. Esto significa que los ejecutables comparten el código común de las librerías en un único fichero, como sucede en SunOS. Así, los ejecutables serán más cortos a la hora de

guardarlos en el disco, incluyendo aquellos que hagan uso de muchas funciones de librería. También pueden enlazarse estáticamente cuando se deseen ejecutables que no requieran la presencia de las librerías dinámicas en el sistema. El enlace dinámico se hace en tiempo de ejecución, con lo que el programador puede cambiar las librerías sin necesidad de recompilación de los ejecutables.

5.2.2. Windows

Es un sistema operativo que permite coordinar todas las tareas que realiza en su computadora. Windows cuenta con una Interfaz Gráfica de Usuario denominada IGU, cuando se habla de una interfaz gráfica de usuario nos referimos a que todas las opciones se manejan por medio de gráficas o dibujos, y es por eso que se hace más cómodo su uso.

Un Sistema Operativo se encarga de adaptar el equipo para que el usuario pueda comunicarse con la máquina ya que sin el sería imposible trabajar con una computadora porque no le entenderíamos su lenguaje ni ella el nuestro. Todas las computadoras deben de tener un programa llamado Sistema Operativo para poder trabajar con ella y Windows, un Sistema Operativo que está instalado en 9 de cada 10 máquinas que hay.

Windows cuenta con una gran trayectoria no es un Sistema Operativo común y corriente y a continuación se menciona algo de esto:

La aparición de Windows 3.0 supuso en realidad un cambio espectacular, tanto en el aspecto de la pantalla de la computadora e interrelacionar con ella (mediante iconos), como en el rendimiento del propio sistema. Gracias a la gestión de la memoria que realiza Windows, es posible extenderla hasta cuatro veces, si se dispone de espacio en el disco duro.

El propio nombre de Windows (ventanas en inglés) hace referencia directa a la filosofía que sirve de soporte a este sistema. Las ventanas que aparecen en la pantalla tienen una serie de propiedades y contienen los elementos de diálogo con el usuario.

Mediante la ayuda del mouse (ratón) es posible modificar su posición y su tamaño, para ajustarlos a nuestras necesidades. Las ventanas, rematadas generalmente por un título, disponen, dependiendo de su tamaño y de su contenido, de una barra de desplazamiento que permite acceder a todo el contenido de la ventana, aunque éste no sea visible a través de ella en un momento dado.

Además de las ventanas e iconos, Windows dispone de diversos elementos característicos, tales como los administradores de programas, de archivos, de impresión, el portapapeles, el panel de control, etc.

5.2.2.1. Administrador de Programas

El administrador de programas de Windows es, en realidad, un programa que se carga en memoria en primer lugar y permanece residente en ella; su principal misión es organizar la elección y posterior ejecución de otras aplicaciones. Gracias al

administrador de programas, se puede iniciar con facilidad cada una de las aplicaciones contenidas en Windows y organizarlas en grupos que sigan una cierta lógica, que facilite su utilización y su localización. Gracias a este administrador, iniciar un programa es tan simple como hacer un doble click sobre el icono que lo representa.

5.2.2.2. Administrador de Archivos

El administrador de archivos facilita la gestión de los archivos contenidos en el sistema, la gestión de los directorios y el manejo de las unidades que contienen, registran o leen la información digital. Las operaciones que permite el administrador de archivos posibilita borrar, copiar, formatear, etc., discos y disquetes, y reorganizarlos en directorios y subdirectorios.

A partir de Windows 95 en adelante, Windows un sistema operativo propio independiente por completo del MS-DOS; está basado en la misma filosofía que el Windows inicial, pero dotados de mejoras muy notables, entre las que debe destacarse:

- El ser un sistema operativo de 32 bits.
- La intensiva utilización del botón derecho del mouse (conocido como botón contextual), que permite utilizar las funciones más habituales de Windows (como puede ser cortar, copiar, pegar, borrar, cambiar el nombre o conocer las propiedades de un determinado archivo) y, además, las propiedades y características de cada programa concreto.
- Además de una remodelación de la interfaz de usuario, Windows permite y facilita el acceso a las redes mundiales de intercambio de información a través de las autopistas de información.

- Windows es multitarea, es decir, permite trabajar simultáneamente con diversas aplicaciones.

5.2.2.3. Windows 98

Nueva versión del sistema operativo Windows. Podría decirse que es una compilación de características. Muchas de estas características ya se encontraban en Internet Explorer 4.0 (al ser instalado con la actualización de escritorio) y en Windows 95 OSR-2. Permite soporte para la FAT32 (al igual que Win95 OSR-2) y el Active Desktop (de IE 4). Soporte para USB y DVD.

Características de Windows 98:

Más fácil de utilizar.- Desplazarse por el equipo resulta más sencillo que nunca, gracias a las opciones de *escritorio* como hacer clic una sola vez para abrir archivos y la inclusión de botones para examinar en todas las ventanas. Puede utilizar múltiples monitores con un único equipo, lo que incrementa considerablemente el tamaño del espacio de trabajo. Instalar nuevo hardware es muy fácil, ya que Windows 98 es compatible con el estándar *Universal Serial Bus (USB, Bus serie universal)*, lo que le permite conectar hardware nuevo y utilizarlo inmediatamente sin necesidad de reiniciar el equipo. Con Windows 98 podrá utilizar también cámaras digitales y otros dispositivos de imágenes digitales.

Más confiable.- Puede tener acceso al sitio Web *Support Online* para obtener respuestas a preguntas frecuentes y mantener actualizada la copia de Windows. Las herramientas de Windows 98 le ayudan a realizar periódicamente pruebas del disco duro y a comprobar los archivos de sistema, e incluso a reparar automáticamente algunos problemas. Los solucionadores de problemas también le ayudan a resolver problemas relacionados con el equipo.

Más rápido.- Los programas y las ventanas se abren más rápido que nunca. Con el Asistente para mantenimiento puede mejorar fácilmente la velocidad y eficacia del equipo. La característica de administración de energía permite que los equipos más recientes entren en modo de hibernación y despierten instantáneamente, en lugar de tener que cerrar y reiniciar el equipo. También puede utilizar el *sistema de archivos* FAT32 para almacenar archivos de forma más eficaz y ahorrar espacio en el disco duro.

Auténtica integración en Web.- Utilizar *World Wide Web* es más fácil que nunca. El Asistente para la conexión a Internet hace que la conexión sea muy sencilla. Mediante Microsoft FrontPage Express puede crear sus propias páginas Web. Si utiliza *Active Desktop* Web de estilo, puede ver páginas Web en cualquier ventana, puede incluso utilizar su página Web favorita como papel tapiz del escritorio. Mediante la característica de canales puede suscribirse a sitios Web, incluyendo aquellos de los principales *proveedores de contenidos* como Disney y Time-Warner. En Microsoft Outlook Express, puede enviar correo electrónico y exponer mensajes a grupos de noticias de Internet. Mediante Microsoft NetMeeting, puede colaborar en documentos y celebrar conferencias en Internet. Si utiliza Microsoft NetShow, puede reproducir en vivo medios de "transferencia", también en Internet.

Más entretenimiento.- Windows 98 es compatible con *DVD* y con audio digital, por lo que puede reproducir audio y películas digitales de alta calidad en su equipo.

Puede cambiar considerablemente la apariencia y el sonido del escritorio si le aplica un tema. Se incluyen varios temas en Windows 98. Cada tema de escritorio incluye un *papel tapiz* único, protectores de pantalla, iconos 3D, sonidos, fuentes, combinaciones de colores y punteros de *mouse*.

5.3. LENGUAJE DE PROGRAMACIÓN

Toda la historia de los lenguajes de programación se ha desarrollado en base a una sola idea: hacer que la tarea de realizar programas para ordenadores sea cada vez lo más simple, flexible y portable posible.

La evolución en los lenguajes, ha venido impulsada por motores bien distintos:

- Los avances tecnológicos.
- Los avances conceptuales (de planteamiento).
- Los avances en cuanto a enfoque de la programación.

Un **lenguaje de programación**, es un conjunto de instrucciones entendibles directamente o traducibles al lenguaje del ordenador con el que trabajemos; combinando estas instrucciones realizamos programas

El lenguaje Pascal ha sido reciclado por Borland, pasando a llamarse **Delphi** en Windows y **Kylix** en Linux. Tiene una implementación de Programación Orientada a

Objetos bastante buena, pero al no permitir sobrecarga de funciones pierde una de las características de POO: *Polimorfismo.*

5.3.1. Delphi

La disponibilidad de Delphi 6.0 y Kylix abre una puerta a la creación de proyectos que tengan los sistemas Windows y Linux como objetivo.

Kylix y Delphi 6.0 hacen posible el desarrollo de interfaces de usuario, componentes, librerías y aplicaciones de servidor para Linux y Windows respectivamente. En Kylix todas sus posibilidades se encuentran en la librería CLX, mientras que en Delphi 6.0 se cuenta, además, con la clásica VCL (*Visual Component Library*), actualizada una vez más para facilitar el uso de los componentes más avanzados de Windows.

De manera combinada, Kylix y Delphi 6.0 ofrecen una de las mejores opciones para el desarrollo cruzado en Windows y Linux. Para ello, ambos productos comparten la mencionada librería de objetos y componentes: CLX.

La programación con Delphi 6.0 y Kylix está dirigido a los usuarios que ya tienen un conocimiento medio de Delphi y quieren aprovechar otras características más avanzadas, como por ejemplo, los mecanismos de acceso a datos, las aplicaciones de servidor, los servicios Web, el desarrollo de componentes, etc.

Se cubren aspectos que van desde el uso de Delphi y Kylix conjuntamente, pasando por la publicación de datos (tanto la impresión física como la publicación digital), hasta

conocer los fundamentos necesarios para desarrollar componentes VCL, CLX, COM y CORBA.

5.3.1.1. Características de Delphi

- Delphi genera ficheros ejecutables compilados. Lo que repercute en su velocidad.
- Delphi puede usar bibliotecas de enlace dinámico (fichero tipo dll) y los puede crear.
- Delphi permite crear sus propios controles (llamados VCL).
- Para crear un control para Delphi sólo se tiene que saber Delphi y tener el Delphi.
- Delphi tiene todos los controles. Si se necesita algún controlador especial se puede obtenerlo de Internet, hay cientos de ellos y muchos freeware.
- En Delphi se puede enviar mensajes a los controles (botones, etiquetas, etc), para reaccionar a ciertas situaciones.

5.3.1.2. Conceptos Fundamentales

Objetos.- Como su propio nombre indica, el Object Pascal (usado por Delphi) es un lenguaje totalmente orientado a objetos. Prácticamente todos los elementos del entorno Delphi son objetos, que unas veces se encuentran definidos de antemano (los forms, los componentes de la paleta, objetos no visuales, etc), y otras se definen en nuestra propia aplicación.

Todo el desarrollo de aplicaciones en Delphi está íntimamente ligado con la definición y uso de objetos, por lo que es fundamental conocer la mecánica que Object Pascal utiliza

para describir un objeto, sus características y su funcionamiento, sobre todo a la hora de que el programador cree sus propios componentes. En muchos casos no es necesario conocer el funcionamiento de la programación orientada a objetos para programar con Delphi, puesto que en la mayoría de los casos existen controles ya creados sin necesidad de tener que programarlos. La mayoría de los objetos forman parte de la VCL de Delphi y otros dos componentes han sido escritos por terceros (RChart y TformulaParser).

La propia programación visual de Delphi, que escribe parte del código automáticamente, hace posible utilizar objetos de forma práctica sin tener por qué comprender al cien por ciento su funcionamiento.

Componentes.- Un componente es cualquiera de los elementos que podemos insertar en una ficha, tanto si su función es visual como si no lo es (por supuesto un componente es también un objeto). El programador manipula una serie de propiedades, métodos y eventos que caracterizan al componente, a través de los cuales se "maneja" el componente en la forma deseada. Por supuesto el usuario puede crear sus propios componentes y usarlos en distintas aplicaciones, de forma que la reusabilidad del código es máxima. A los componentes que cuentan con una parte visual, como puede ser un botón, se les denomina controles.

Propiedades.- Los componentes, y de hecho todos los objetos de Delphi son de uso general, por lo que a la hora de usarlos de alguna forma deberemos adecuarlos a nuestras necesidades. Para ello nos serviremos de las propiedades de cada objeto, mediante las cuales podremos establecer el título de una ventana, el tipo de letra de una etiqueta de texto o el color en el que aparecen los distintos controles.

Se puede pensar en las propiedades como si fuesen variables pertenecientes a un objeto (y no es así exactamente), de tal forma que para acceder a ellas generalmente habrá que indicar no solo el nombre de la propiedad, sino también a qué objeto pertenece.

La modificación o consulta del valor de una propiedad puede diferir según intentemos acceder a ella mientras estamos diseñando un form, en tiempo de diseño, o bien mediante el código del programa, en tiempo de ejecución. Ciertas propiedades están sólo accesibles en tiempo de ejecución, por lo que mientras estamos diseñando la ficha no aparecerán. Además hay algunas propiedades que son de sólo lectura, por lo que su valor puede ser consultado, pero no modificado, y otras que son sólo de escritura.

Aunque a primera vista si usamos un componente prefabricado, las propiedades pueden parecer simples variables, en la mayoría de las ocasiones una propiedad no es una variable, y la modificación o consulta de su valor puede conllevar que internamente el componente ejecute un cierto código. Si por ejemplo tuviésemos un componente para comunicaciones serie que se encargase de enviar mensajes a un ordenador remoto, podríamos tal vez asignar una cadena de caracteres a una hipotética propiedad "Envía".

De esta forma, con una simple asignación (Envía:='Mensaje a mandar') se pondría en marcha todo el mecanismo implícito en el componente para enviar el mensaje al ordenador remoto. Esta propiedad podría ser un ejemplo de propiedad de solo escritura, ya que solo interesa mandar el mensaje al ordenador remoto, sin conservar información acerca del mensaje, por lo que sería inapropiada su lectura.

Eventos.- La programación en el entorno Windows se caracteriza por estar dirigida por eventos, de tal forma que un programa no tiene por qué ejecutarse necesariamente de forma secuencial, sino que ciertas porciones de código se ejecutarán cuando ocurra un cierto evento.

Los eventos son señales que el entorno recibe desde distintos elementos, como puedan ser el mouse (ratón), el teclado o un temporizador. Estos eventos son redirigidos a las aplicaciones, que en caso de aceptarlos deberán responder adecuadamente de ellos.

Ciertos eventos pueden ser gestionados por el propio Windows, otros quedarán a cargo del propio lenguaje que estemos usando, y un tercer grupo serán los que lleguen hasta nuestro programa. En Delphi prácticamente todo el código que escribimos irá asociado a algún evento. Si retomamos el ejemplo del componente para comunicaciones serie, podría interesarnos que se ejecutara un evento cada vez que se recibiese un caracter por el puerto serie, de forma que podríamos escribir el código necesario para guardar el caracter en un archivo cada vez que se produjese el evento. Normalmente los eventos a los que reaccionarán los componentes serán las pulsaciones del teclado o el mouse (ratón), activaciones de los componentes, etc.

Métodos.- Los componentes Delphi además de disponer de propiedades y poder responder a ciertos eventos, habitualmente también disponen de métodos. Un método es un procedimiento o función que nos permite realizar una determinada acción en el componente, pudiendo necesitar o no el paso de algún parámetro.

Al igual que ocurre en las propiedades, a la hora de usar un cierto método normalmente tendremos que indicar primero el objeto o componente al que pertenece, de tal forma que la acción del método recaiga sobre él, y no sobre cualquier otro. El hecho de que cada objeto sea propietario de una serie de propiedades y métodos, variables y código, que no son accesibles a ningún otro objeto externo, recibe el nombre de encapsulación, aunque también es posible definirlos de tal modo que sean accesibles a otros objetos.

5.3.2. Kylix

Borland Software sostiene que será una de las primeras herramientas comerciales de Desarrollo Rápido de Aplicaciones (Rapid Application Development - RAD) para la plataforma Linux. El proyecto Kylix abarca las herramientas de desarrollo Delphi, C++ Builder y JBuilder de Borland para los lenguajes de programación Object Pascal, C, C++ y Java, así que no es sólo "Delphi para Linux" como muchos piensan, aunque el plan es que Delphi para Linux sea el primer producto producido por el proyecto Kylix.

Será un entorno de desarrollo visual de aplicaciones basado en componentes. Incluirá una versión para Linux de la Biblioteca de Componentes Visuales (Visual Component Library o simplemente VCL) que actualmente se usa en Delphi y C++ Builder.

¿Por qué las aplicaciones para Linux hechas a medida requerirían más tiempo y más dinero para desarrollarse? ¿Por qué no hay tantas aplicaciones comerciales?, es porque no existe Delphi para Linux!!! Si han programado con Delphi, no hay necesidad de explicar cuán fácil es desarrollar y mantener aplicaciones de alto desempeño (desde programas de sistema hasta complejas aplicaciones de bases de datos, desde

aplicaciones cliente hasta aplicaciones de servidor, desde aplicaciones de consola hasta interfaces gráficas). Y hay más. Es orientado a objetos y viene con una jerarquía de componentes que se pueden usar en las aplicaciones tal como vienen, o que fácilmente se pueden heredar para derivar nuestros propios componentes. Además se tiene una importante comunidad de desarrolladores, el sólido soporte de Borland, foros, grupos de usuarios, boletines, etc.

Muchos usuarios de Linux están acostumbrados a las aplicaciones gratuitas, incluso a las de fuentes abiertos cuando es posible, pero en el caso de una herramienta RAD como ésta, algunos esperan que será un verdadero "best-seller". Hay muchos programadores esperando que aparezca una herramienta como ésta antes de considerar Linux como una alternativa real, y se espera que muchos programadores en Delphi y C++ Builder y empresas de desarrollo de software comprarán Kylix para entrar en el mercado de Linux.

5.3.2.1. El Compilador y la RTL

Los programadores en Delphi que quieran comenzar a preparar sus aplicaciones para migrarlas a Kylix deben contemplar lo siguiente:

Herramientas de línea de órdenes

- DCC (Delphi Command-line Compiler) es un ejecutable nativo de Linux y produce ejecutables x86 nativos de Linux.
- Nuevo ensamblador incorporado, escrito en código portable.
- Tasm, make y brcc no existirán en Linux.

- Opciones de la línea de órdenes: usarán '-' en vez de '/'.
- Para especificar muchos directorios habrá que usar muchas opciones. Los puntos y coma (';') no son separadores de caminos de directorios en Linux. Para especificar muchos directorios habrá que usar muchas opciones.
- Habrá algunas restricciones en cuanto a los recursos que se incorporan a los ejecutables que no existen en Windows.
- La introspección de recursos no estará soportada.
- Las oportunidades de las herramientas de edición de recursos estarán limitadas o serán más difíciles.
- Posición Independiente del Código (PIC). Las bibliotecas de objetos compartidas (el equivalente de las DLL) requiere que todo el código sea reubicable en memoria sin modificación. Esto afectará su código escrito en ensamblador.

Sintaxis del lenguaje

- Existirán en Kylix los tipos GUID e interface de Delphi.
- Las directivas condicionales podrán evaluar expresiones y podrán acceder a constantes declaradas.
- El símbolo condicional WIN32 no estará definido en Kylix.
- No se soportarán variables en direcciones absolutas.
- La convención de llamadas Stdcall se mapeará a cdecl. Esto sólo afectará al código escrito en ensamblador.
- La convención de llamadas Safecall se mapeará a cdecl, perdiendo toda su semántica de tiempo de ejecución. Generará una advertencia durante la compilación.

Biblioteca de Tiempo de Ejecución (Run-time library - RTL)

- Nueva implementación portable del tipo Variant. Solo los tipos de datos variables listados como compatibles con automatización OLE del lado Windows han sido implementados del lado Linux. El tipo VT_DECIMAL de 12 bytes de Windows no será soportado.
- Las WideStrings tendrán contador de referencias (igual que las AnsiStrings).
- Las unidades como ComObj, ComServ, Activex, Windows, etc. no existirán en Kylix.
- No habrá excepciones en llamadas Safecall.
- RaiseLastWin32Error, OleCheck y Win32Check no existirán.
- ExpandUNCFilename no existirán dado que Linux no soporta UNC (serverdirectory).
- Los nombres de archivos son sensibles a mayúsculas/minúsculas: las aplicaciones deberán estar conscientes de esto.
- WideChar es (aún) un caracter Unicode 2 byte. El tipo widechar de Linux (wchar_t), es en realidad de 4 bytes por caracter.
- Las AnsiStrings se codificarán como UTF-8, una codificación multibyte de 4 bytes por caracter, según el estándar UCS.
- Las secuencias multibyte de caracteres pueden tener más de 2 bytes de longitud.
- Eficiencia en los recursos de cadenas. Los recursos de cadenas se codificarán en UTF-8 (1 byte por caracter, generalmente) en el archivo ejecutable.
- Las horas de archivos (enteros de 32 bit) estarán en formato Linux.
- DiskFree, DiskSize, y ExtractFileDrive serán modificadas o desaparecerán debido a que no hay unidades de disco en Linux.

- Cuestiones de separación de caminos. Se agregará una nueva constante, PathSeparator, que contendrá el caracter apropiado de separación para la plataforma.

5.4. BASE DE DATOS

5.4.1. Base de Datos Relacional

Las ***bases de datos*** representan los datos en tablas. Las ***tablas*** son colecciones de filas que consisten de columnas, (registros y campos en otra terminología).

5.4.1.1. Sistema de Administración de Base de Datos Relacionales

Las principales funciones de un sistema de administración de bases de datos relacional son:

- Representar los datos como un conjunto de tablas.
- Facilitar el diseño, la visión y la modificación de los datos.
- Usar los valores de los datos para establecer relaciones, brindando máxima flexibilidad.
- Proveer operaciones relacionales como selección, proyección y unión.
- Proveer el acceso más rápido posible a los datos deseados.
- Proteger los datos de usuarios no autorizados con un sofisticado control de seguridad.
- Asegurar la consistencia de los datos con control de transacciones atómicas.

- Asegurar los datos contra fallas de hardware y software con herramientas de recuperación basadas en bitácoras.
- Integrar con lenguajes de programación, usando sentencias de manipulación de datos incluidas directamente en programas fuentes.
- Proveer fácil acceso a bases de datos remotas.
- Proveer un entorno interactivo para escribir y probar consultas a la base de datos.
- Trasladar las definiciones de datos en los programas y los ubica en la base de datos.
- Proveer funciones para buscar, ordenar y realizar análisis estadísticos.
- Proveer herramientas de diseño y administración.
- Permitir definición de datos en forma dinámica: crear tablas, almacenar datos, consultar, deshacer y reintentar.
- Permitir cambios dinámicos a las definiciones de la base de datos; agregar, cambiar o eliminar campos, tablas e índices.

5.4.2. MYSQL

MYSQL es un Sistema de Gestión de Base de Datos. Una ***Base de Datos*** es una colección estructurada de datos. Puede ser, desde una simple lista de artículos a las inmensas cantidades de información en una red corporativa.

MYSQL es un gestor de bases de datos SQL (Structured Query Lenguage). Es una implementación Cliente-Servidor que consta de un servidor y diferentes clientes (programas/librerías). Podemos agregar, acceder, y procesar datos grabados en una base

de datos. Actualmente el gestor de base de datos juega un rol central en la informática, como única utilidad, o como parte de otra aplicación.

Es un Sistema de Gestión de Base de Datos Relacional. El modelo relacional se caracteriza a muy grandes rasgos por disponer que toda la información debe estar contenida en tablas, y las relaciones entre datos deben ser representadas explícitamente en esos mismos datos. Esto añade velocidad y flexibilidad.

MYSQL es un software de código abierto esto quiere decir que es accesible para cualquiera, para usarlo o modificarlo. Podemos descargar MYSQL desde Internet y usarlo sin pagar nada, de esta manera cualquiera puede inclinarse a estudiar el código fuente y cambiarlo para adecuarlo a sus necesidades. MYSQL usa el GPL de la GNU (Licencia Pública General) para definir que podemos y no podemos hacer con el software en diferentes situaciones. Entre otras cuestiones esta licencia aclara que no cuesta dinero a menos que lo incluyamos en un software comercial y tenemos el código fuente.

MYSQL es muy rápido, seguro y fácil de usar. MYSQL también ha desarrollado un conjunto de características muy prácticas, en estrecha cooperación con otros usuarios.

MYSQL fue desarrollado para manejar grandes bases de datos mucho más rápido que las soluciones existentes y ha sido usado exitosamente en ambientes de producción con altas demandas, por varios años. Aunque está bajo un desarrollo constante, MYSQL siempre ofrece conjunto de funciones muy poderoso y eficiente. La conectividad,

velocidad y seguridad hace de MYSQL una suite poderosa para accesar a bases de datos en Internet.

La versión libre fue escrita por Michael Windenis, la versión comercial es distribuida por TCX Datakonsulter AB. Es de fuente abierta ya que puede ser usado y modificado por cualquiera. Alguien puede bajar MYSQL desde Internet y usar éste sin tener que pagar nada. Alguien puede estudiar el código fuente y cambiarlo de acuerdo a sus necesidades. MYSQL usa el GPL de la GNU, para definir lo que se puede hacer con el software en diferentes situaciones.

5.4.2.1. Historia

IBM empezó a comercializar en 1.981 el SQL y desde entonces este producto ha tenido un papel importante en el desarrollo de las bases de datos relacionales. IBM propuso y fue aceptada, una versión de SQL al Instituto de Estándares Nacional Americano(ANSI) y desde entonces es utilizado de forma generalizada en las bases de datos relacionales. En 1.983 nació DB2 la más popular(por lo menos en los grandes ordenadores) de las bases de datos de este tipo hasta estos mismos momentos.

En el mundo GNU, una de las bases de datos que se reseña en cualquier referencia de aplicaciones de éste tipo bajo LINUX, es MYSQL aunque no está incluida en ninguna distribución ya que no tiene licencia GNU como tal, para comercializarla a ella o a cualquier software que la utilice o se sirva de ésta habrá que adquirir una licencia.

Alrededor de la década del 90, Michael Windenis (monty@analytikerna.se) comenzó a usar mSQL para conectar tablas usando sus propias rutinas de bajo nivel (ISAM). Sin embargo, después de algunos testeos llegó a la conclusión que mSQL no era lo suficientemente rápido ni flexible para sus necesidades. De todo esto surgió en una nueva interfaz SQL (claro que con código más portable) con algunas apariencias similares en la API de C y en los nombres y funciones de muchos de sus programas.

Esto habia sido hecho para lograr con relativa facilidad portar aplicaciones y utilidades de MiniSQL a MYSQL.

El origen del nombre MYSQL no esta perfectamente claro. Algunos lo atribuyen al hecho de que un gran numero de nuestras librerías y herramientas le asignamos el prefijo "My" por costumbre. Sin embargo la hija de 'Monty' es además llamada My. Así que cual de las dos razones da el nombre a MYSQL es aún un misterio.

Diversas aplicaciones ya soportan MYSQL. La forma oficial de pronunciar MYSQL es My Ess Que Ell. Los valores centrales de MYSQL son :

- La mejor y más usada base de datos en el mundo.
- Disponible y accesible para todos.
- Fácil de usar.
- Se está perfeccionando continuamente mientras permanece rápida y segura.
- Divertida para usar y perfeccionar.
- Libre de molestias.

5.4.2.2. Características Técnicas de MYSQL

MYSQL es un sistema Cliente/Servidor que consta de un servidor SQL multi-hilo que soporta diferentes backends, variados programas cliente y de librerías, administrador de herramientas y un programa de interface.

5.4.2.3. Características principales de MYSQL

- El principal objetivo de MYSQL es velocidad y robustez.
- Escrito en C y C++, testado con GCC 2.7.2.1. Usa GNU autoconf para potabilidad.
- Clientes C, C++, JAVA, Perl, TCL.
- Multiproceso, es decir puede usar varias CPU si éstas están disponibles.
- Puede trabajar en distintas plataformas y SO distintos.
- Sistema de contraseñas y privilegios muy flexible y segura.
- Todas las palabras de paso viajan encriptadas en la red.
- Registros de longitud fija y variable.
- 16 índices por tabla, cada índice puede estar compuesto de 1 a 15 columnas o partes de ellas con una longitud máxima de 127 bytes.
- Todas las columnas pueden tener valores por defecto.
- Utilidad (Isamchk) para chequear, optimizar y reparar tablas.
- Todos los datos están grabados en formato ISO8859_1.
- Los clientes usan TCP o UNIX Socket para conectarse al servidor.
- El servidor soporta mensajes de error en distintas lenguas.
- Todos los comandos tienen -help o -? Para las ayudas.

- Diversos tipos de columnas como enteros de 1, 2, 3, 4, y 8 bytes, coma flotante, doble precisión, caracter, fechas, enumerados, etc.
- ODBC para Windows 95 (con fuentes), se puede utilizar ACCESS para conectar con el servidor.

5.4.2.4. Sistemas Operativos soportados por MYSQL

- AIX 4.x con consulta nativa.
- BSDI 2.x con el paquete MIT-pthreads incluido.
- BSDI 3.0, 3.1 y 4.x con consulta nativa.
- DEC Unix 4.x con consulta nativa.
- FreeBSD 2.x con el paquete MIT-pthreads incluido.
- FreeBSD 3.x y 4.x con consulta nativa.
- HP-UX 10.20 con el paquete MIT-pthreads incluido.
- HP-UX 11.x con consulta nativa.
- Linux 2.0+ con LinuxThreads 0.7.1+ o glibc 2.0.7+ .
- Mac OS X Server.
- NetBSD 1.3/1.4 Intel y NetBSD 1.3 Alpha (carece de la marca GNU).
- OpenBSD > 2.5 con consulta nativa. OpenBSD < 2.5 con el paquete MIT-pthreads incluido.
- OS/2 Warp 3, FixPack 29 y OS/2 Warp 4, FixPack 4.
- SGI Irix 6.x con consulta nativa.
- Solaris 2.5 y con consulta nativa sobre SPARC y x86.
- SunOS 4.x con el paquete MIT-pthreads incluido.
- SCO OpenServer con un puerto reciente al paquete de FSU Pthreads.

- SCO UnixWare 7.0.1.
- Tru64 Unix.
- Win95, Win98, NT, y Win2000.

5.4.2.5. Conectándose y desconectándose al Servidor MYSQL

Para conectarse al servidor, generalmente se necesita proveer a **MYSQL** un nombre de usuario, y un password. Si el servidor corre en una máquina distinta a la que se está utilizando se debe especificar el nombre del host (máquina).

Cuando se conocen todos estos parámetros la conexión se realiza de la siguiente manera:

shell> mysql -h host -u user -p

Enter password:*********

Si la conexión se realizó con éxito, *mysql* despliega el siguiente mensaje:

Welcome to the MySQL monitor. Commands end with; or \g.

Your MySQL connection id is 459 to server version: 3.23.41

Type 'help;' or '\h' for help. Type '\c' to clear the buffer.

mysql>

El prompt indica que *mysql* está listo para recibir los comandos que ingrese el usuario.

Algunas instalaciones de **MYSQL** admiten usuarios *anonymous* (sin nombre) cuando el servidor corre en el host local. En este caso, se debe invocar al servidor *mysql* sin ninguna opción:

shell>mysql

Una vez que se ha realizado la conexión con éxito, para desconectarse al servidor en cualquiera de los dos casos anteriores se debe tipear:

mysql>quit

o

mysql>\q

Al salir aparece un mensaje **Bye**

5.4.2.6. Visualización de las Bases de Datos existentes en el Servidor MYSQL

Antes de crear una base de datos, se debe conocer que base de datos existen actualmente en el servidor, para ello se utiliza el comando SHOW, de la siguiente manera:

mysql> SHOW DATABASES;

Database
mysql
test
tmp

Esta lista probablemente no es igual en todas las máquinas, pero las bases de datos **mysql** y **test** están siempre entre ellas. La base de datos **mysql** se requiere porque en ella se describe la información de los privilegios de acceso a los usuarios. La base de datos **test** proporciona el espacio de trabajo para los usuarios .

5.4.2.7. Selección de una Base de Datos

Para seleccionar o accesar a una base de datos determinada se utiliza el comando USE.

```
mysql> USE nombre_database;
Database changed
```

Una vez, que se ha realizado la conexión con éxito se puede comenzar a trabajar con la base de datos, pero siempre y cuando se tengan los permisos adecuados. Si no se tienen los permisos el administrador debe darle los permisos al usuario para poder trabajar, esto se realiza con la ejecución del siguiente comando:

```
mysql> GRANT ALL ON nombre_database.* TO nombre_usuario;
```

5.4.2.8. Creación de una Base de Datos

Para crear una base de datos se debe tener permiso para poder crear base de datos en el servidor MYSQL , si se tiene el permiso entonces la sentencia a seguir es:

```
mysql> CREATE DATABASE nombre_database;
```

Bajo Unix, los nombres de las bases de datos y de las tablas son casos sensibles, esto quiere decir que se hace diferencia entre minúsculas y mayúsculas, así es que para referirse a una base de datos determinada hay que llamarla tal como se le nombró cuando fue creada.

5.4.2.9. Creación de Tablas

Para crear las tablas que va a contener la base de datos, se realiza de la siguiente forma:

mysql> CREATE TABLE nombre_tabla(campo_1 tipo(tamaño), campo_2 tipo(tamaño),...,campo_3 tipo(tamaño));

El *campo* indica el nombre de la columna y *tipo(tamaño)* especifica el tipo de dato y el espacio que se va a conservar para cada dato almacenado en esa columna. Ejemplo: codigo int(5), nombre char(25), fecha date, etc.. Cuando se trata de fechas no se especifica el tamaño, puesto que ya está determinado.

Ejemplo: Se va a crear una tabla llamada **(estacion)**, de la siguiente forma:

mysql> CREATE TABLE estacion(ip varchar(15) NOT NULL ,nombre varchar(30), PRIMARY KEY (ip));

Para verificar que la tabla ha sido creada de la forma indicada, se usa el comando DESCRIBE.

mysql> DESCRIBE estacion;

Field	Type	Null	Key	Default	Extra
ip	varchar(15)		Pri	0	
nombre	varchar(30)	Yes		0	

2 rows in set (0.11 sec)

Esto es muy útil cuando se olvida el nombre o tipo de una columna. El *Field* indica el nombre de la columna, *Type* es el tipo de dato que acepta esa columna, *Null* indica si la columna puede contener valores NULL, *Key* indica la clave por la cual la columna se va a indexar y *Default* especifica el valor por defecto que tiene la columna.

Para visualizar las tablas que tiene una base de datos se usa el mismo comando **SHOW** utilizado para ver las bases de datos, pero con la diferencia de que en vez de **database** se coloca **tables**, es decir:

mysql> SHOW TABLES;

5.4.2.10. Ingreso de Datos a las Tablas

Para ingresar información a una tabla se puede hacer básicamente de dos maneras. La primera se utiliza cuando se tiene mucha información a ingresar de una sola vez, entonces es conveniente almacenar esta información en un archivo de texto, es decir, *.txt*. Una vez que se tiene este archivo, se procede de la siguiente forma:

mysql> LOAD DATA LOCAL INFILE "nombre archivo.txt" INTO TABLE nombre tabla;

Para el caso que se desee ingresar un solo registro, entonces la sentencia a seguir es:

mysql> INSERT INTO nombre_tabla VALUES ("valor_1","valor_2",....);

Los datos a ingresar se separan por comas y van entre comillas. Estos datos indican los valores que va a tomar cada una de las columnas, según el orden en que fueron creadas.

En el caso que se quiera ingresar un valor NULL no es necesario las comillas, sólo se coloca NULL.

5.4.2.11. Recuperación de la Información de una Tabla

Para recuperar la información que está contenida en una tabla, la sentencia general a seguir es:

mysql> SELECT qué_es_lo_que_se_desea_ver FROM nombre_tabla WHERE condiciones_a_satisfacer;

Para los casos en que, se requiera:

- Ver o seleccionar toda la información de una tabla:

  mysql> SELECT * FROM nombre_tabla;

- Seleccionar filas en particular:

  mysql> SELECT * FROM nombre_tabla WHERE nombre_columna="lo que se desee buscar"

- Seleccionar columnas en particular:

  mysql> SELECT nombre_columna_1, nombre_columna_n FROM nombre_tabla;

CAPÍTULO VI

SEGURIDAD DE LA INFORMACIÓN

6.1. INTRODUCCIÓN

El propósito del presente capítulo es realizar una explicación detallada sobre la Seguridad de la Información y las Medidas de Seguridad.

En la Seguridad de la Información se encontrará el concepto de seguridad, en la Seguridad de Datos se encontrará su objetivo, la definición de datos, acceso y cambios, identificación, reconstrucción, movimiento y destrucción.

En las Medidas de Seguridad se encontrará tres tipos de seguridades: seguridad legal, seguridad física y seguridad lógica.

6.2. SEGURIDAD DE LA INFORMACIÓN

Hay que proteger la información para proporcionar seguridad a la misma.

6.2.1. Seguridad

Es la facultad de estar a cubierto de algún riesgo o amenaza. Así, la seguridad de la información resulta muy difícil de establecer como tal, puesto que implicaría descubrir

todos los riesgos y amenazas a que puede verse sometida. Lo que se manifiesta en los sistemas informáticos no es la seguridad, sino más bien la inseguridad o vulnerabilidad.

No se puede hablar de un sistema informático que sea seguro, sino más bien que no se conocen tipos de ataque que puedan vulnerarlo. Por lo general, se admite que los sistemas de información crecen y se desarrollan a una velocidad bastante superior a la que lo hacen sus dispositivos de seguridad. El desafío que todo sistema informático ha de abordar es el conseguir unas funcionalidades, que vienen exigidas por unos condicionantes de empresa. Este desafío tiene contenido y problemas en sí mismo de suficiente entidad como que las eventualidades ligadas a la seguridad pasen irremediablemente a un segundo plano.

Probablemente la única y verdadera razón que explique este comportamiento, más o menos generalizado, del mundo informático es que existen presiones de más calado y con más apremio que las que normalmente se encuadran en el concepto de seguridad, y que se podría definir como fiabilidad de los sistemas de información. Esta fiabilidad viene determinada por factores como:

- El cumplimiento de plazos.
- Conseguir las funcionalidades realmente necesitadas.
- Disponer del sistema siempre que se necesite, sin que quede "colgado".
- Un funcionamiento sin errores ni fallos.

Estos problemas son los que pueden comprometer la viabilidad y continuidad de los proyectos. Con la prioridad que se da a estos problemas es poco probable que la confidencialidad, integridad y continuidad, términos bajo los que se suele detallar el

concepto de seguridad, realmente llegar a obtener la atención que necesitan. A pesar de la visión general un tanto escéptica, es necesario precisar que, en los últimos tiempos, parece apreciarse un avance en el desarrollo de los aspectos y conceptos de seguridad informática, especialmente en lo que se refiere a concienciación colectiva y, por consiguiente, en acciones concretas de diversos tipos.

Probablemente, una de las razones de esta mayor concienciación es la universalización de los sistemas de información. No se trata ya de que todo el mundo los utiliza (empresas, universidades, investigadores, particulares, comercios, etc.), sino de que los utilizan interconectados. La conectividad puede resultar, pues, un motor de concienciación y avance en seguridad informática.

Los sistemas informáticos pueden diseñarse de acuerdo a criterios de economía, de eficiencia, etc., porque para dichos criterios se conocen parámetros que, maximizando unos y minimizando otros, se puede tender hacia diseños óptimos. Cuando se trata del criterio de seguridad el problema se vuelve más complejo y difícilmente resoluble. Los criterios y puntos de vista que utiliza el diseñador para obtener seguridad no son los mismos bajo los cuales se moverá el atacante.

En el convencimiento de que establecer medidas de seguridad en un sistema de tratamiento de información es un problema extraordinariamente complejo, lo más importante es que los administradores y gerentes de los sistemas tomen conciencia de que son medidas que llevan un costo aparejado en su implantación y que obligan a subordinar algunas ventajas de los sistemas.

6.2.2. Seguridad de Datos

Objetivo

- Asegurar la seguridad de los datos "sensibles" sin merma de su acceso por los demás usuarios autorizados.

Definición

- **Datos.-** La información que la organización usa de acuerdo con su propósito. En cualquier soporte o en forma legible. Se entiende por Seguridad de la Información, por supuesto a la confidencialidad, disponibilidad e integridad de la misma.

Acceso y Cambios

- Han de existir procedimientos que definan qué datos son "sensibles".
- Han de existir controles que restrinjan el acceso a datos "sensibles" sólo por personas autorizadas.
- Cualquier acceso no autorizado ha de ser puesto de manifiesto e investigado.
- Los cambios han de ser autorizados y sometidos a una pista de auditoría.

Identificación

- Todos los archivos y salidas han de ser pronta y exactamente definidos.

Reconstrucción

- Archivo, manejo y proceso, han de ser tales que en caso de perdida o corrupción, se puedan reconstruir los archivos en el tiempo preciso.

Movimiento

- Ha de registrarse todo movimiento de modo que se garantice la seguridad en integridad de los datos.

Destrucción

- De modo seguro.

6.3. MEDIDAS DE SEGURIDAD

Las medidas de seguridad a establecer en un sistema de tratamiento de la información se muestran en la **(Figura 6.1.)** Son de cuatro tipos:

- **De Seguridad Lógica.**
- **De Carácter Físico.**
- **De Carácter Administrativo.**
- **Y por último, Legales.**

MEDIDAS DE SEGURIDAD

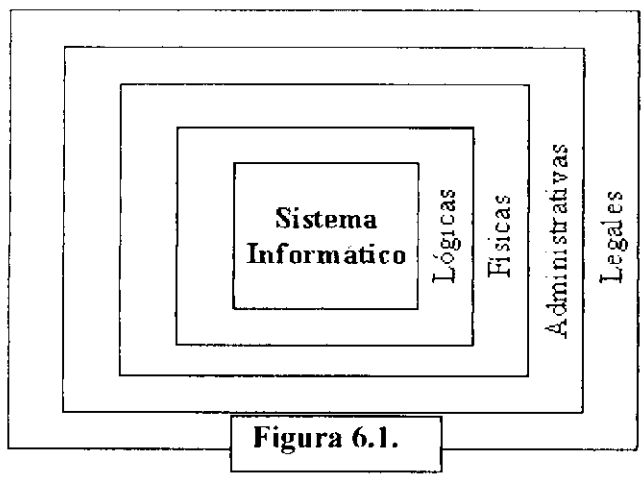


Figura 6.1.

Los administradores de los sistemas informáticos deben conocer la legalidad vigente para decidir contra qué tipo de acciones deben preservar al sistema informático y, en el caso de producirse un ataque, poder perseguir al malhechor. Para ello, deben establecer medidas de carácter administrativo que permitan clarificar las responsabilidades ante determinados ataques. Las medidas de carácter físico y lógico son las que están directamente encaminadas a evitar los ataques.

6.3.1. Seguridad Legal

La necesidad de proteger la información parece evidente. Sin embargo, es tan difícil de delimitar lo que hay que proteger que se convierte en un problema extraordinariamente arduo de resolver. Si se analiza por ejemplo el caso de los bancos se comprueba que es mucho más sencillo. Un banco tiene muy delimitados los bienes (joyas, valores, dinero, etc.) que debe proteger, y para ello, usa medios muy conocidos como cajas fuertes, sistemas de video, guardias, etc., para poder contrarrestar las distintas amenazas conocidas. En el caso de la información es mucho más complicado el análisis. Puede parecer que determinados datos carecen, en general, de valor intrínseco y sin embargo, pueden tenerlo, y mucho, para determinadas personas. La información es algo que unas veces permite ver claro a quién le puede interesar y qué amenazas se ciernen sobre ella y otras veces aparenta ser algo que a nadie le interesa. De esto surge la gran polémica existente en todos los países con respecto a las leyes que regulan el derecho a la intimidad.

Así pues, no se debe olvidar que **informática jurídica** es la utilización de la informática como una simple herramienta al servicio de los juristas en sus diferentes trabajos.

El **derecho informático,** a diferencia de la informática jurídica, es aquella parte del derecho que regula el mundo informático. Fruto del mismo son la protección jurídica del software o leyes de protección intelectual que castigan el delito informático y también las leyes que regulan la protección de datos de carácter personal, así como otras que irán apareciendo en el futuro para regular la contratación informática, la transferencia electrónica de datos, el documento electrónico, etc.

6.3.2. Seguridad Física

La seguridad física surge como respuesta a los problemas físicos de la seguridad informática y se refieren a cualquier peligro que pueda ser causado por agentes externos al sistema del tipo de fuego, inundaciones, sobrecargas eléctricas, etc.

La seguridad física de un sistema informático supone conocer perfectamente la configuración de dicho sistema. El que un sistema disponga solamente de una computadora central o que sea un sistema distribuido implica una estrategia de seguridad completamente diferente.

El lugar donde vaya a situarse la computadora y la seguridad propia del recinto del centro de proceso de datos implican que se estén primando una serie de medidas de seguridad sobre otras. Por ejemplo, el instalar la computadora en un recinto subterráneo puede facilitar el control del acceso por parte de personal ajeno a la organización, pero provoca que aumente el peligro de inundaciones del sistema debido a fugas de agua o a que los sistemas contraincendios del edificio, caso de que no se tenga esto en cuenta, entren en funcionamiento.

Cada vez son más las medidas de seguridad que se tienen en cuenta en relación a la situación del sistema informático y ya no suelen ser habituales las organizaciones que colocan su computadora central en una sala con paredes de vidrio transparente para que los visitantes observen su "tecnología punta". La discreción prima cada vez más en los ambientes informáticos.

Los principales problemas de carácter físico a los que se van a enfrentar los encargados de los sistemas informáticos son cuatro:

- El fuego.
- El agua.
- El robo.
- El sabotaje físico.

6.3.3. Seguridad Lógica

Surge como respuesta a los problemas lógicos que conllevan una serie de aspectos más amplios, por su complejidad e indefinición. Estos pueden ir desde la recuperación del sistema tras un problema hasta proteger el acceso al sistema frente a usuarios no autorizados.

Como ya se ha dicho, la necesidad de protección del sistema informático es independiente del coste que éste haya tenido, pero aumenta en forma proporcional al valor que éste represente para su poseedor.

Habría que incidir también en algo que viene produciéndose en el mercado informático desde hace ya un tiempo; el cambio de tendencia que se ha producido en el costo de un sistema informático. Este cambio comenzó a producirse a finales de los años ochenta y principios de la década de los noventa, cuando el precio del software paso a ser, proporcionalmente, más caro que el del hardware que lo soportaba. Debido al aumento del valor del software existente en ellos, los sistemas informáticos han debido ir introduciendo una serie de medios para proteger el creciente valor que contenían.

Otro factor fundamental en la creciente necesidad de mejora de la seguridad en los sistemas informatices es la:

- Cada vez mayor cantidad de procesos que se realizan en tiempo real dentro de diversas actividades, como las bancarias, financieras, etc. Estos procesos necesitan tener una serie de controles que permitan asegurar la fiabilidad de la información implicada en ellos.

Por todo ello, las mejoras en seguridad, como todas las mejoras, provocan un aumento de valor de todos los elementos del sistema informático implicados, además de una mayor confianza de los usuarios en un entorno en el que cada vez están más implicados.

En general, los sistemas informáticos deben incorporar mecanismos que permitan ofertar unos servicios de seguridad a los usuarios. Estos servicios proporcionan determinadas propiedades o características de seguridad a la información manejada por temas informáticos.

6.3.3.1. Características de la Seguridad Lógica de la Información

Se encuentran tres criterios o propiedades fundamentales al establecer la seguridad de la información. Estas son:

- **Confidencialidad(secreto).**
- **Integridad.**
- **Accesibilidad.**

Los mecanismos que pueden proporcionar alguna de estas tres características no suelen hacerlo de forma aislada, sino que muchas veces proporcionan más de una de ellas al mismo tiempo. Existen características de seguridad que un análisis profundo permitiría asegurar que están implícitas en las anteriores, pero que el gran auge de las comunicaciones y de los sistemas de transmisión de datos entre computadoras, así como su masiva introducción en el mundo de los negocios actual, hace que deban mencionarse por separado.

Esto se apoya en la clasificación que hacen las normas ISO y CCITT respecto de los servicios de seguridad en redes de datos. En concreto, las características de seguridad que se están referenciando son:

- **Autenticidad.**
- **Imposibilidad de Rechazo.**

Confidencialidad(secreto).- La información debe estar disponible tan solo para los usuarios autorizados a manejar la misma, es decir, a todos aquellos (personas, entidades, programas, etc.), que tengan derecho legal a usarla. Esta propiedad es la más

directamente relacionada con los procedimientos criptográficos. En el secreto debe incluirse no sólo el de los datos, sino también el del flujo de información. Existen circunstancias en que lo importante no es salvaguardar la información en sí misma, sino salvaguardar que origen y destino, se envían información. Es decir, la información debe quedar protegida contra el análisis de tráfico.

Integridad.- Esta propiedad permite asegurar que no se ha falseado la información. Por ejemplo, que los datos recibidos y/o recuperados son exactamente los que fueron enviados y/o estaban almacenados que un programa corre con los parámetros que le son propios, etc.

Accesibilidad.- Esta propiedad asegura quién puede acceder a la información y en qué momento se puede hacer. No debe entenderse sólo en sentido restrictivo, puesto que los mecanismos que la proporcionen deben permitir la disponibilidad de la información a aquellos sujetos que tengan derecho a la misma. En resumen, se deben proporcionar mecanismos para impedir el acceso a las entidades no autorizadas y mecanismos para que las que lo están puedan disponer de la misma.

Autenticidad.- Esta propiedad permite asegurar el origen y destino de la información.

Es muy importante a la hora de establecer la firma electrónica, dar validez al correo electrónico, etc.

Imposibilidad de Rechazo.- Esta propiedad permite asegurar que cualquier entidad que envía o recibe información no puede alegar ante terceros que no la envío o la recibió. Es importante a la hora de establecer el correo electrónico certificado.

Estas cinco características no deben entenderse aisladas unas de otras, sino más bien interrelacionadas. Ellas permiten hablar de aspectos de seguridad en los sistemas y proporcionan una gran ayuda a la hora de implantar servicios de seguridad en un sistema informático.

Dependiendo del uso de un sistema informático, la implantación de servicios que proporcionen alguna de estas características de seguridad puede ser vana y en otros casos imprescindibles. El amplio empleo de los cajeros automáticos es debido al establecimiento de un servicio que permite autentificar al usuario, en este caso resulta imprescindible; sin embargo sería innecesario en una sala de computadoras en donde se realice un control de acceso en la puerta de la sala.

A pesar de todo, tal y como se ha indicado anteriormente, el establecimiento de servicios de seguridad para salvaguardar la información no puede asegurar la invulnerabilidad de la misma. Con ello, lo que se quiere indicar es que la seguridad no es un valor absoluto; sólo se puede hablar de mayor o menor seguridad.

CAPÍTULO VII

METODOLOGÍA DE DESARROLLO DEL SOFTWARE

7.1. INTRODUCCIÓN

A medida que se acercaba los años 80, el "paradigma orientado a los objetos" para la ingeniería del software comenzaba a madurar como un enfoque de desarrollo de software. Se comenzó a crear diseños de aplicaciones de todo tipo usando la forma de pensar orientada a los objetos y se implementa (se codifica) programas usando lenguajes y técnicas orientados a los objetos.

El propósito del presente capítulo es dar una explicación detallada sobre la metodología de desarrollo del software como es el Análisis, Diseño y Programación Orientada a Objetos.

En el AOO (Análisis Orientado a Objetos) se encontrará los conceptos y principios orientados a los objetos en donde estará la identificación de Objetos, especificación de atributos, definición de las operaciones y comunicación entre objetos, además se encontrará el modelo conceptual en donde estarán los diagramas de clase, diagramas de interacción diagramas de secuencia.

El Diseño se dividirá en dos partes Diseño de la Base de Datos en donde estará el Diseño Lógico y el Diseño Físico, y el DOO (Diseño Orientado a Objetos) en donde

estarán los diagramas de interacción – diagramas de colaboración y definir como se hace cada operación del sistema.

En la POO (Programación Orientada a Objetos) se encontrará una explicación de donde se puede encontrar el Código Fuente.

7.2. ANÁLISIS ORIENTADO A OBJETOS

El Análisis Orientado a los Objetos es comprender el dominio del problema y de esta forma reconocer las responsabilidades del sistema. En esta etapa lo más importante es comprender la profundidad y dominio del problema para poder determinar completamente todos los detalles de una determinada situación. Es importante comprender y modelar el dominio del problema especialmente para sistemas grandes y complejos.

7.2.1. Conceptos y Principios Orientados a Objetos

Vivimos en un mundo de objetos. Estos objetos existen en la naturaleza, en entidades hechas por el hombre, en los negocios y en los productos que usamos. Ellos pueden ser clasificados, descritos, organizados, combinados, manipulados y creados. Por esto no es sorprendente que se proponga una visión orientada a objetos para la creación de software de computadora, una abstracción que modela el mundo de forma tal que nos ayuda a entenderlo.

Los conceptos y principios básicos que forman el fundamento para la comprensión de tecnología de objeto.

7.2.1.1. Identificación de Clases y Objetos

En realidad, el **AOO** lo que intenta es definir *clases* de las que se pueden *instanciar* objetos. Por tanto, cuando identificamos posibles objetos, también estamos identificando posibles clases. Podemos empezar a identificar objetos examinando la descripción del problema llevando a cabo un "análisis gramatical" de la narrativa de procesamiento del sistema a construir. Determinamos los objetos subrayando cada nombre o cláusula nominal y añadiéndolo en una tabla. Si un objeto es necesario para implementar una solución, entonces es parte *del espacio de la solución*; en caso contrario, cuando el objeto sólo es necesario para describir la solución, entonces es parte del *espacio del problema.* Pero, ¿qué debemos observar una vez que ya hemos aislado todos los nombres?

Las clases y objetos se manifiestan en una de las formas que se representan en la **(Figura 7.1.).** Las clases y objetos pueden ser:

- ***Entidades externas*** (p. ej.: otros sistemas, dispositivos, gente) que producen o consumen información a ser utilizada en el sistema basado en computadora.
- ***Cosas*** (p. ej.: informes, visualizaciones, cartas, señales) que son parte del dominio de información del problema.
- ***Ocurrencias o Sucesos*** (p. ej.: una transferencia de una propiedad o la terminación de una serie de movimientos de un robot) que ocurren en el contexto de operación del sistema.

- ***Papeles*** que juegan las personas que interactuan con el sistema (p.ej.: gestor, ingeniero, vendedor).
- ***Unidades Organizativas*** (p. ej.: división, grupo, equipo) que son relevantes para la aplicación.
- ***Lugares*** (p. ej.: sala de facturación o muelle de descarga) que establecen el contexto del problema y del funcionamiento general del sistema.
- ***Estructuras*** (p. ej.: sensores, vehículos de cuatro ruedas o computadoras) que definen clases de objetos o, en casos extremos, clases de objetos relacionadas.

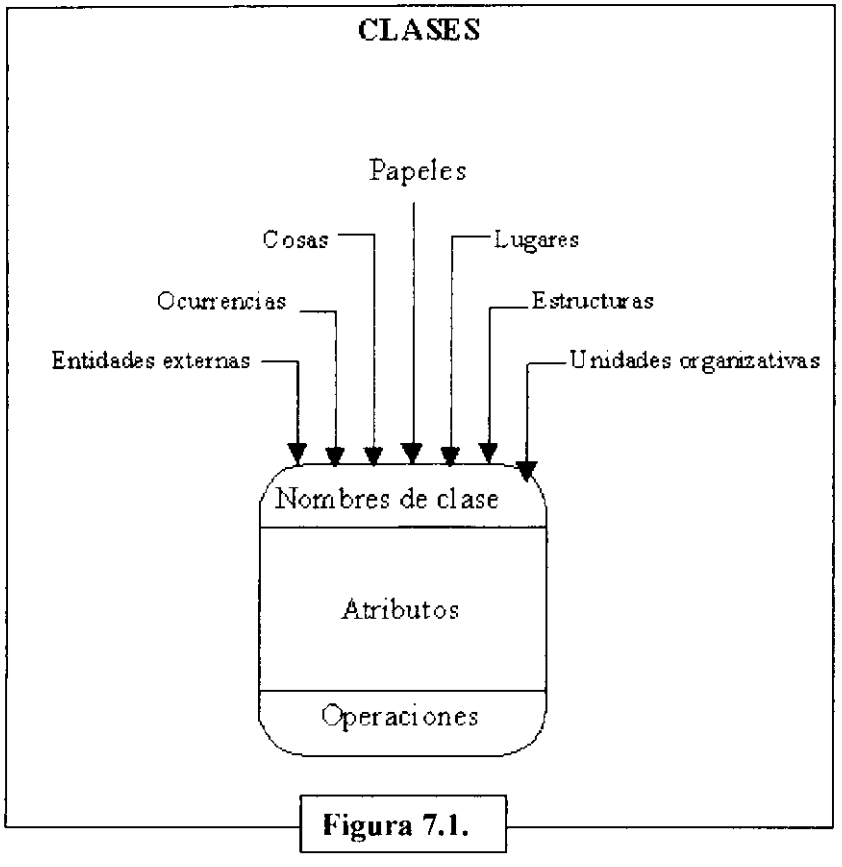


Figura 7.1.

Para ilustrar cómo se pueden identificar los objetos ya en las primeras etapas del análisis, partiendo de la descripción del problema que consta en el **Capitulo I** en el punto **1.2.1.** vamos a volver a recordar lo que especificó en la delimitación que consta

en el **Capitulo I** en el punto **1.2.2.** A continuación, reproducimos la narrativa de procesamiento:

"El administrador encargado de la Red de la Escuela de Ingeniería de Sistemas de la PUCESA debe iniciar primero el Sistema Servidor de Agentes el cual provee fecha, hora, intervalo de tiempo, usuario y contraseña de desactivación.

El Sistema de Control de Estaciones de Trabajo en la Red de la Escuela de Ingeniería de Sistemas de la PUCESA que es un programa agente, se ejecuta automáticamente al iniciar la PC; en donde:

- Captura la dirección IP, y el nombre de la máquina que se obtienen de las estaciones de trabajo se almacena en la (Tabla estacion que es la tabla principal).
- La dirección IP capturada, el nombre de usuario, el sistema operativo, las actividades o procesos y las ventanas que se obtienen de las estaciones de trabajo, la fecha y la hora que se obtienen del servidor se almacena en bitácoras, es decir en la (Tabla mvitacora que es la tabla de movimientos).
- El nombre, el sistema operativo y el número de veces que se ha utilizado una aplicación se almacena en la (Tabla aplicacion que también es una tabla de movimientos)."

Extrayendo los nombres podemos proponer varios objetos potenciales, que se muestran en la **(Tabla 7.1.)**:

(Tabla 7.1. Identificación de Clases y Objetos)

Objeto/Clase potencial	Clasificación General
Usuario(sinónimo: Tabla padmin se utiliza para desactivar con un usuario y un password el programa agente que se ha instalado)	entidad externa
Intervalo(sinónimo: Tabla pintervalo se utiliza para dar un intervalo de tiempo de envío de información del Sistema Agente a la Base de Datos agentes)	entidad externa
Estación(sinónimo: Tabla estacion, que detalla las estaciones de trabajo activas)	entidad externa
Bitácoras(sinónimo: Tabla mvitacora que es la tabla de movimientos de las estaciones de trabajo activas)	entidad externa
Aplicación(sinónimo: Tabla aplicacion se utiliza para reportes estadísticos en base al número de veces que se han utilizado dichas aplicaciones en las estaciones de trabajo activas)	entidad externa

La lista anterior continuaría hasta que hayan sido considerados todos los nombres contenidos en la narrativa de procesamiento. Se puede observar cómo hemos identificado cada entrada de la lista como objeto potencial. Antes de tomar una definición final, hemos de considerar cada uno de ellos más detenidamente.

Existen seis características selectivas que se debe usar para considerar la inclusión o no de cada objeto potencial en el modelo de análisis:

1. **Información retenida.-** El objeto potencial será útil durante el análisis sólo si la información sobre el mismo ha de ser recordada para que el sistema pueda funcionar.
2. **Servicios necesarios.-** El objeto potencial debe tener un conjunto de operaciones identificables que puedan cambiar de alguna forma el valor de sus atributos.
3. **Múltiples atributos.-** Durante el análisis de requisitos, se debe centrar la atención en la información "principal"; un objeto con un sólo atributo puede resultar útil durante el diseño, pero probablemente se represente mejor como atributo de otro objeto durante la fase de análisis.
4. **Atributos comunes.-** Se pueden definir conjuntos de atributos para los objetos potenciales y aplicar así esos atributos a todas las ocurrencias del objeto.
5. **Operaciones comunes.-** Se pueden definir conjuntos de operaciones para los objetos potenciales y aplicar así esas operaciones a todas las ocurrencias del objeto.
6. **Requisitos esenciales.-** En el modelo de requisitos casi siempre se definirán como objetos las entidades externas que aparecen en el espacio del problema y que producen o consumen información esencial para el funcionamiento de cualquier solución que se desarrolle para el sistema.

Para que un objeto sea considerado como legítimo para ser incluido en el modelo de requisitos, el objeto potencial debe satisfacer todas (o casi todas) las características anteriores. La decisión de incluir o no un objeto potencial en el modelo de análisis puede resultar algo subjetiva, haciendo que una posterior evaluación pueda descartar o

reinstanciar un determinado objeto. Sin embargo, el primer paso del AOO debe consistir en la definición de los objetos, debiendo llevar a cabo decisiones (incluso aunque sean subjetivas). Teniendo esto en cuenta, aplicamos las características de selección a la lista de objetos potenciales del software agente en la **(Tabla 7.2.)** que se muestra a continuación:

(Tabla 7.2. Características Selectivas de la Identificación de Clases)

Objeto/Clase potencial	**Clasificación General**
Usuario(sinónimo: Tabla padmin)	aceptado: todas son aplicables
Intervalo(sinónimo: Tabla pintervalo)	rechazado: falla 3, aunque 6 sea aplicable
Estación(sinónimo: Tabla estacion)	aceptado: todas son aplicables
Bitácoras(sinónimo: Tabla mvitacora)	aceptado: todas son aplicables
Aplicación(sinónimo: Tabla aplicacion)	aceptado: todas son aplicables

7.2.1.2. Especificación de Atributos

Los atributos.- describen un objeto que haya sido seleccionado para ser incluido en el modelo de análisis. Esencialmente, son los atributos lo que definen un objeto – son los que aclaran el significado del objeto en el contexto del espacio del problema.

Para desarrollar un conjunto significativo de atributos para un objeto, se debe estudiar de nuevo la narrativa de procesamiento (o descripción del alcance) para el problema

concreto y seleccionar aquello de "pertenezca" de forma razonable al objeto. Además, se debe responder a la siguiente pregunta para cada objeto: ¿qué elementos de datos (elementales y/o compuestos) definen completamente ese objeto en el contexto del problema actual?

La lista de atributos razonable para la clase Usuario se muestra en la **(Figura 7.2.)**.

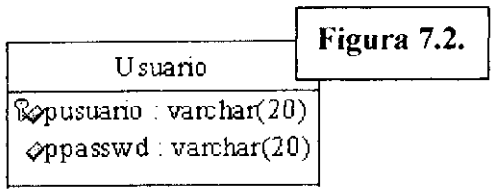


Figura 7.2.

La lista de atributos razonable para la clase Intervalo se muestra en la **(Figura 7.3.)**.

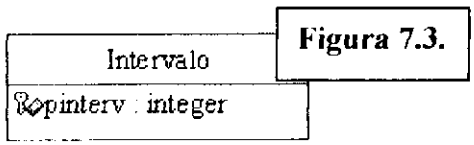


Figura 7.3.

La lista de atributos razonable para la clase Estación se muestra en la **(Figura 7.4.)**.

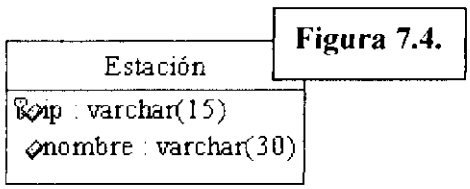


Figura 7.4.

La lista de atributos razonable para la clase Bitácora se muestra en la **(Figura 7.5.)**.

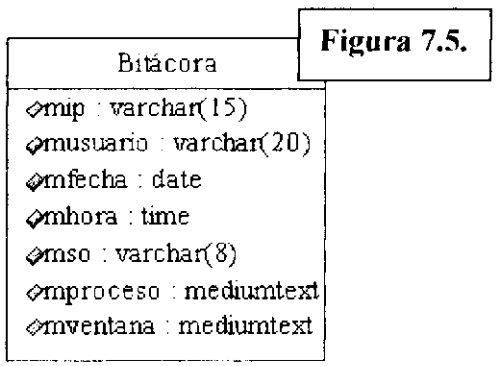


Figura 7.5.

La lista de atributos razonable para la clase Aplicación se muestra en la **(Figura 7.6.)**.

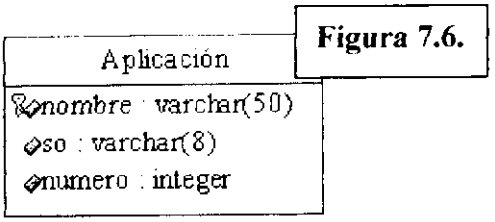


Figura 7.6.

7.2.1.3. Definición de las Operaciones

Las operaciones definen el comportamiento de un objeto y cambian, de alguna manera, los atributos de dicho objeto. Más concretamente, una operación cambia valores de uno o más atributos contenidos en el objeto. Por lo tanto, una operación debe tener <<conocimiento>> de la naturaleza de los atributos de los objetos y deben ser implementadas de manera tal que permita manipular las estructuras de datos que han sido derivadas de dichos atributos.

Aunque existen muchos tipos diferentes de operaciones, estas pueden clasificarse en tres categorías:

1. Operaciones que manipulan, de alguna manera, datos (p.ej.: añadiendo, eliminando, reformateando, seleccionando.
2. Operaciones que realizan algún cálculo.
3. Operaciones que monitorizan un objeto frente a la ocurrencia de un suceso de control.

Por ejemplo, de la narrativa de proceso, presentada anteriormente en este capítulo, nos sirve para definir las operaciones de nuestras clases y objetos identificados.

La lista de operaciones razonable para la clase Usuario se muestra en la **(Figura 7.7.)**.

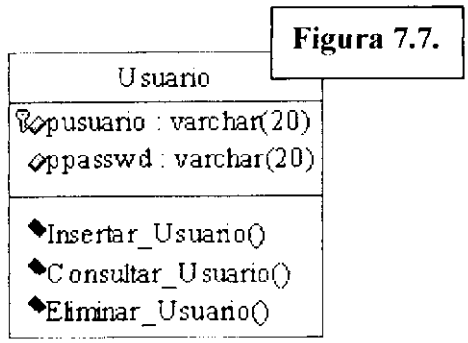


Figura 7.7.

La lista de operaciones razonable para la clase Intervalo se muestra en la **(Figura 7.8.).**

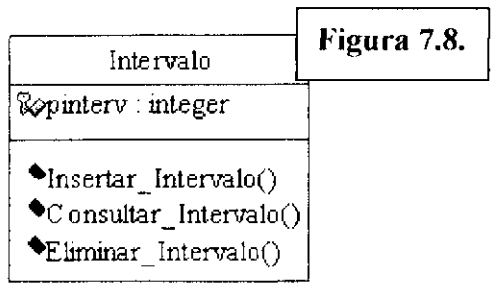


Figura 7.8.

La lista de operaciones razonable para la clase Estación se muestra en la **(Figura 7.9.).**

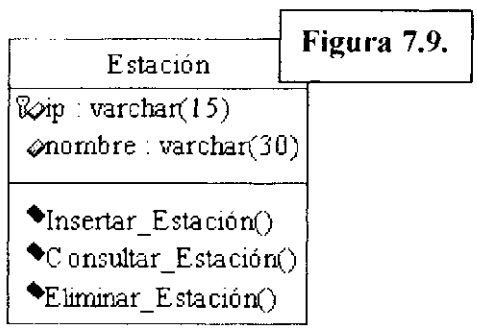


Figura 7.9.

La lista de operaciones razonable para la clase Bitácora se muestra en la **(Figura 7.10.).**

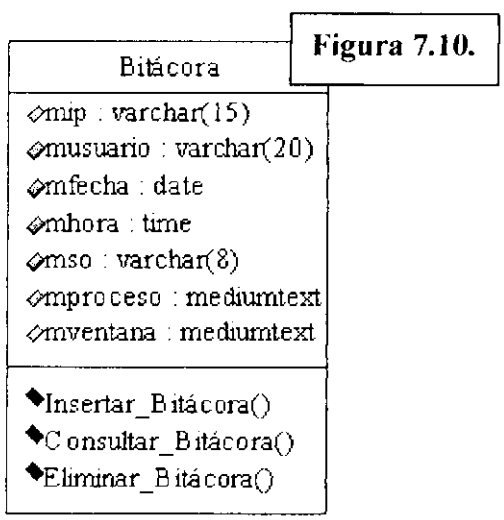


Figura 7.10.

La lista de operaciones razonable para la clase Aplicación se muestra en la **(Figura 7.11.)**.

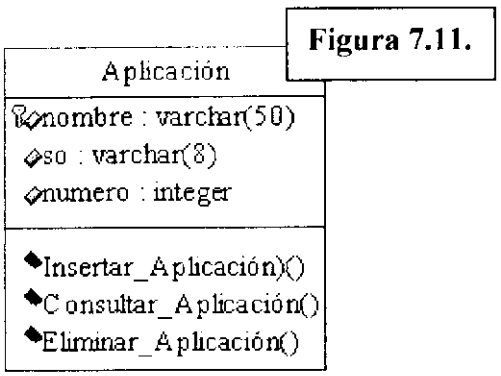


Figura 7.11.

7.2.1.4. Comunicación entre Objetos

Los mensajes son el medio a través del cual los objetos interactúan. Usando la terminología introducida en la sección precedente, un mensaje estimula la ocurrencia de cierto comportamiento en el objeto receptor. El comportamiento se realiza cuando se ejecuta una operación.

La interacción entre objetos se ilustra esquemáticamente en la **(Figura 7.12.)**.

Una operación dentro de un *objeto emisor* genera un mensaje de la forma:

mensaje: [destino, operación, argumentos]

donde *destino* define el objeto *receptor* el cual es estimulado por el mensaje, *operación* se refiere al método que recibe el mensaje y *parámetros* proporcionan información requerida para el éxito de la operación.

Como un ejemplo de paso de mensajes dentro de un sistema OO, considere los objetos que se muestran en la **(Figura 7.13).** Los cuatro objetos, **A, B, C** y **D,** se comunican

entre sí mediante el paso de mensajes. Por ejemplo, si el objeto **B** requiere el proceso asociado con la operación **op10** del objeto **D**, entonces el primero le envía a **D** un mensaje que contendría:

mensaje: [**D**, **op10**,<datos>]

Como parte de la ejecución de **op10,** el objeto **D** puede enviar un mensaje al objeto **C** de la forma:

mensaje: [**C**, **op08**,<datos>]

C encuentra **op08**, la ejecuta, y entonces envía un valor de retorno apropiado a **D**.

La operación **op10** completa su ejecución y envía un valor de retorno a **B**.

Paso de mensajes

Mantiene comunicados un sistema orientado a objetos. Los mensajes proporcionan una visión interna del comportamiento de objetos individuales, y del sistema OO como un todo.

Figura 7.12.

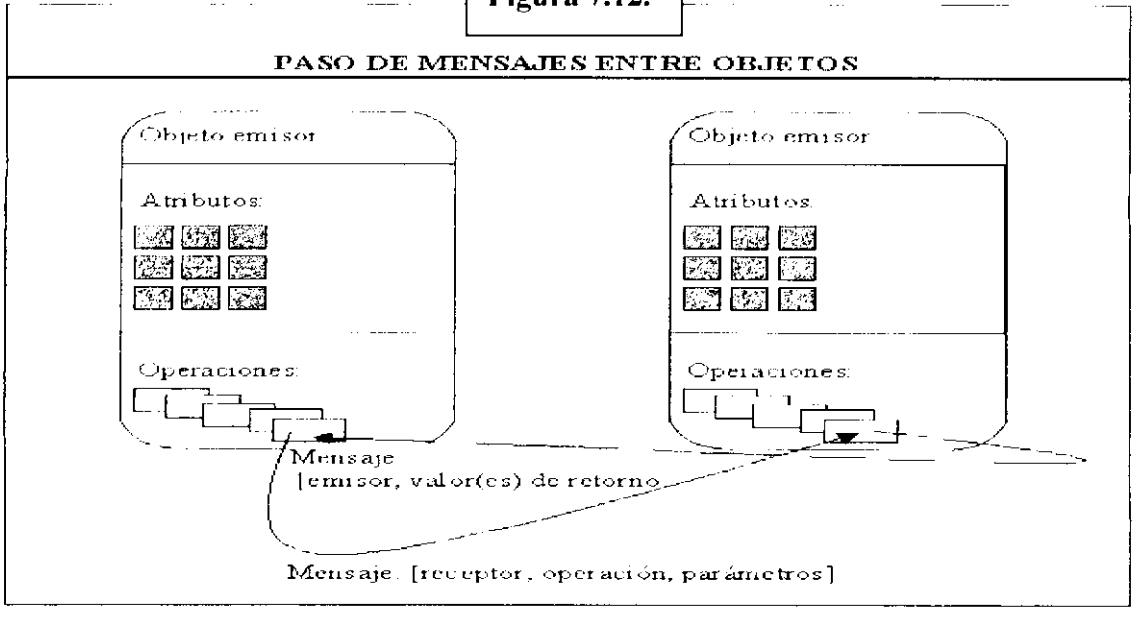
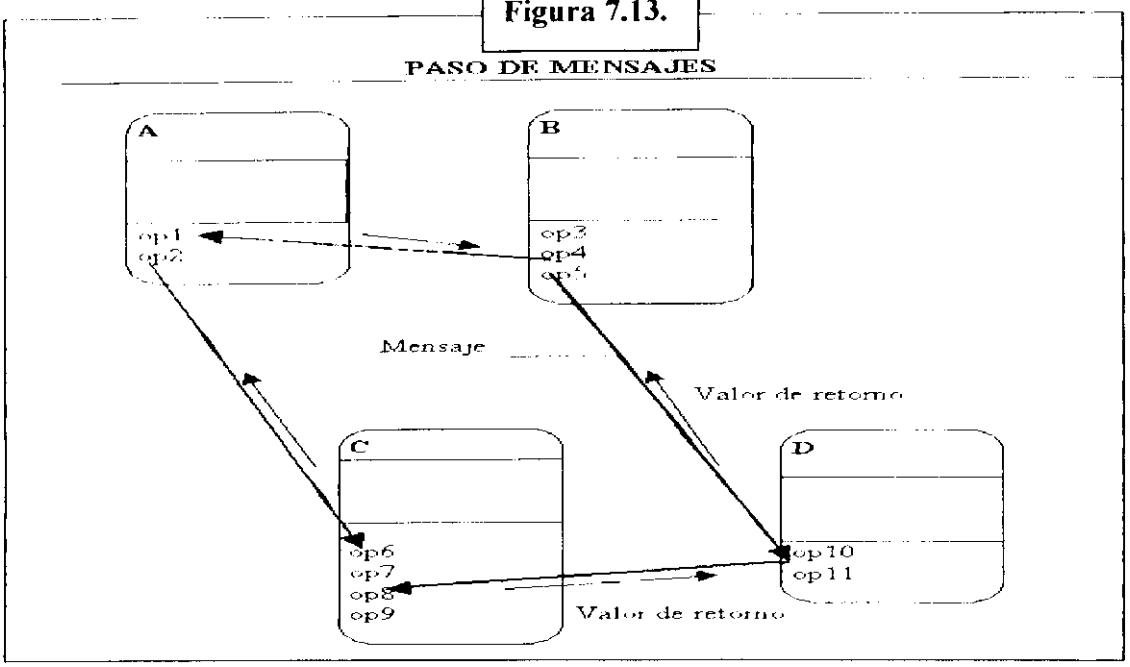


Figura 7.13.



7.2.2. Modelo Conceptual

El proceso de AOO no comienza con una preocupación por los objetos. Más bien comienza con una comprensión de la manera en la que se usará el sistema: por las personas, si el sistema de interacción con el hombre; por otras máquinas, si el sistema está envuelto en un control de procesos; o por otros programas; si el sistema coordina y controla otras aplicaciones. Una vez que se ha definido el escenario, comienza el modelado del software.

Un modelo conceptual explica los conceptos significativos en un dominio; es lo más importante a crear durante el AOO.

Las secciones que siguen definen una serie de técnicas que pueden usarse para recopilar requisitos básicos del usuario y después definen un modelo de análisis para un sistema orientado a objetos.

7.2.2.1. Diagramas de Clase

La técnica del diagrama de clase se ha vuelto medular en los métodos orientados a objetos.

El *diagrama de clase,* además de ser de uso extendido, también está sujeto a la más amplia gama de conceptos de modelado. El *diagrama de clase* describe los tipos de objetos que hay en el sistema y las diversas clases de relaciones estáticas que existen entre ellos.

Los diagramas de clase también muestran los atributos y operaciones de una clase y las restricciones a que se ven sujetos, según la forma en que se conecten los objetos.

A continuación en la **(Figura 7.14)** se muestra el diagrama de clases que hemos desarrollado para nuestro objeto de estudio.

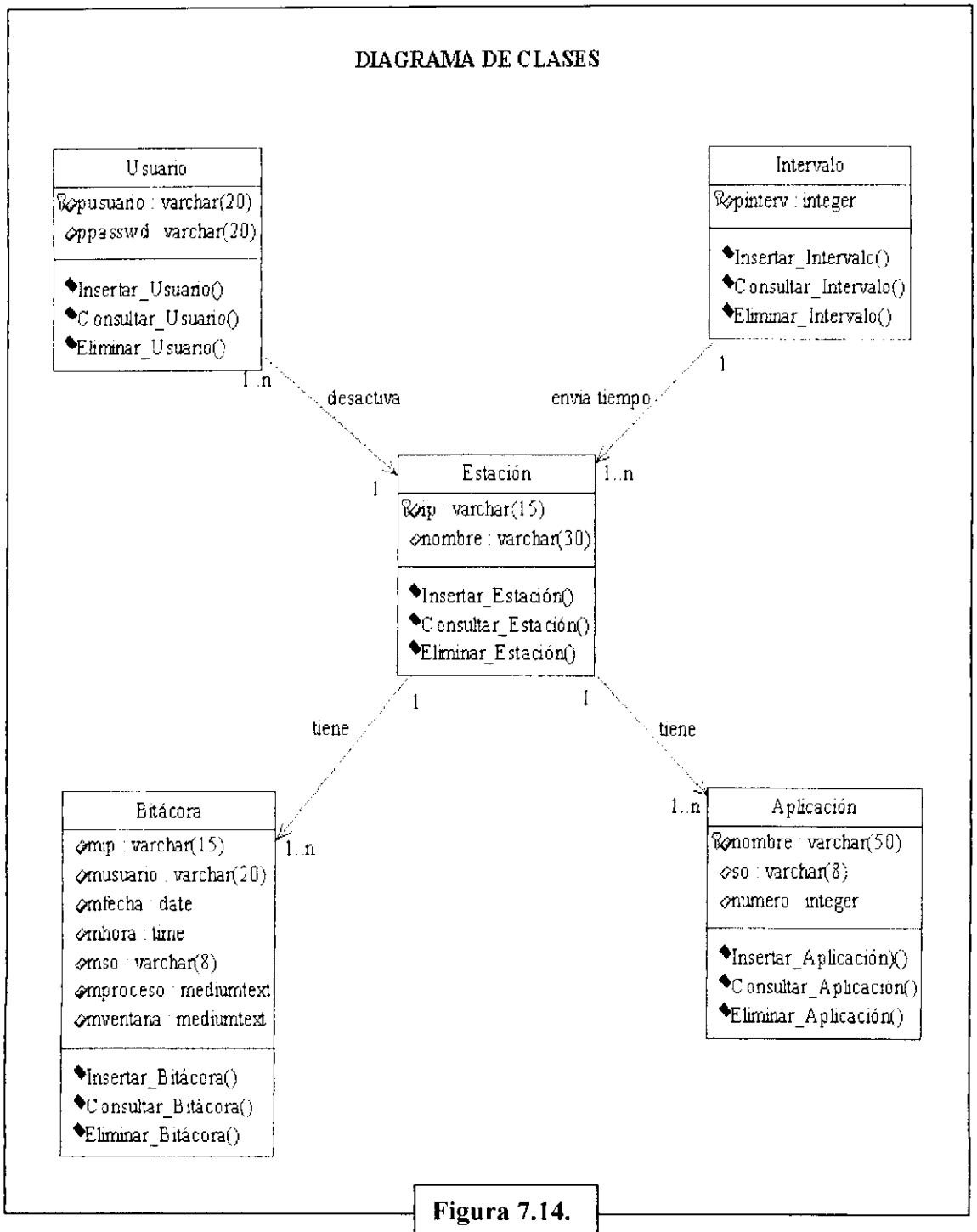

DIAGRAMA DE CLASES
Usuario
pusuario : varchar(20)
ppasswd : varchar(20)
Insertar_Usuario()
Consultar_Usuario()
Eliminar_Usuario()
Intervalo
pinterv : integer
Insertar_Intervalo()
Consultar_Intervalo()
Eliminar_Intervalo()
1..n
desactiva
envia tiempo
1
1..n
Estación
ip : varchar(15)
nombre : varchar(30)
Insertar_Estación()
Consultar_Estación()
Eliminar_Estación()
1
1
tiene
tiene
Bitácora
mip : varchar(15)
musuario : varchar(20)
mfecha : date
mhora : time
mso : varchar(8)
mproceso : mediumtext
mventana : mediumtext
Insertar_Bitácora()
Consultar_Bitácora()
Eliminar_Bitácora()
1..n
1..n
Aplicación
nombre : varchar(50)
so : varchar(8)
numero : integer
Insertar_Aplicación()
Consultar_Aplicación()
Eliminar_Aplicación()

Figura 7.14.

7.2.2.2. Diagramas de Interacción

Un diagrama de interacción explica gráficamente las interacciones existentes entre las instancias de un modelo conceptual.

El UML define dos tipos de diagramas de interacción:

- Diagramas de Secuencia(se tratará aquí en el Análisis Orientado a Objetos).
- Diagramas de Colaboración (se tratará con más detenimiento en el Diseño Orientado a Objetos).

Diagramas de Secuencia

En un diagrama de secuencia, un objeto se muestra como caja en la parte superior de una línea vertical punteada.

Esta línea vertical se llama línea de vida del objeto. La línea de vida representa la vida del objeto durante la interacción.

Cada mensaje representa mediante una flecha entre las líneas de vida de dos objetos. El orden en el que se dan estos mensajes transcurre de arriba hacia abajo. Cada mensaje es etiquetado por lo menos con el nombre del mensaje; pueden incluirse también los argumentos y alguna información de control, y se puede mostrar la audelegación, que es un mensaje que un objeto se envía a si mismo, regresando la flecha de mensaje de vuelta a la misma línea de vida. Los diagramas de secuencia le ayudan a ver la secuencia.

El mensaje se envía sólo si la condición es verdadera. Este diagrama incluye un regreso, el cual indica el regreso de un mensaje, no un nuevo mensaje.

Los gráficos de los diagramas de secuencia que hemos escogido para nuestro objeto de estudio se muestran en la **(Figura 7.15.)**, **(Figura 7.16.)**, **(Figura 7.17.)**, **(Figura 7.18.)**, **(Figura 7.19.)**, **(Figura 7.20.)** y **(Figura 7.21.)**.

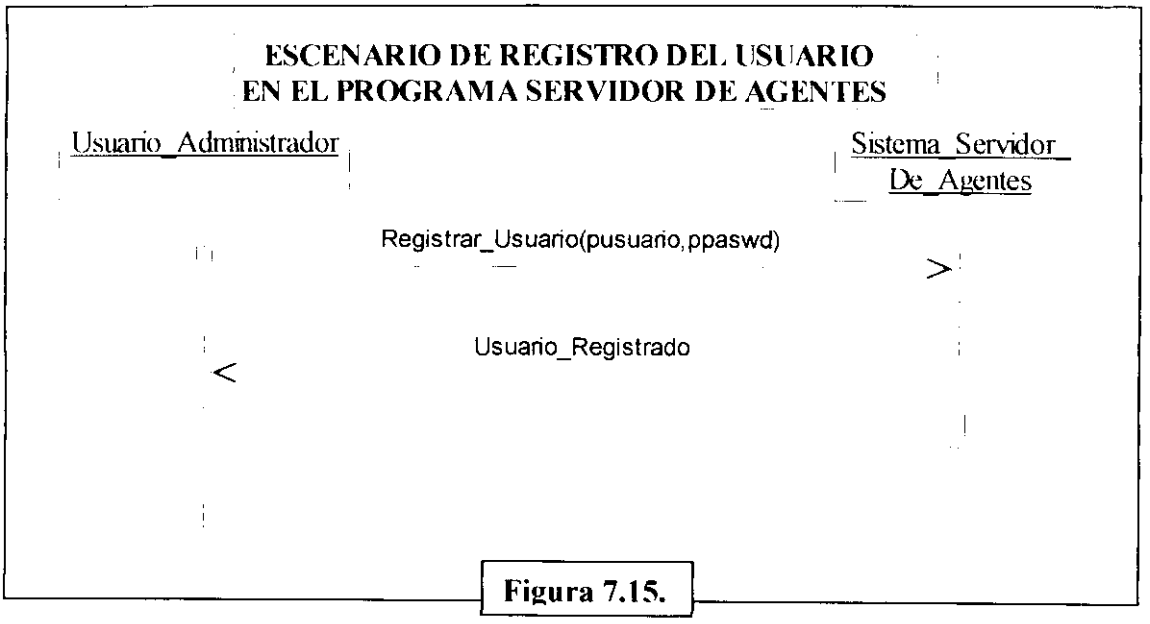


Figura 7.15.

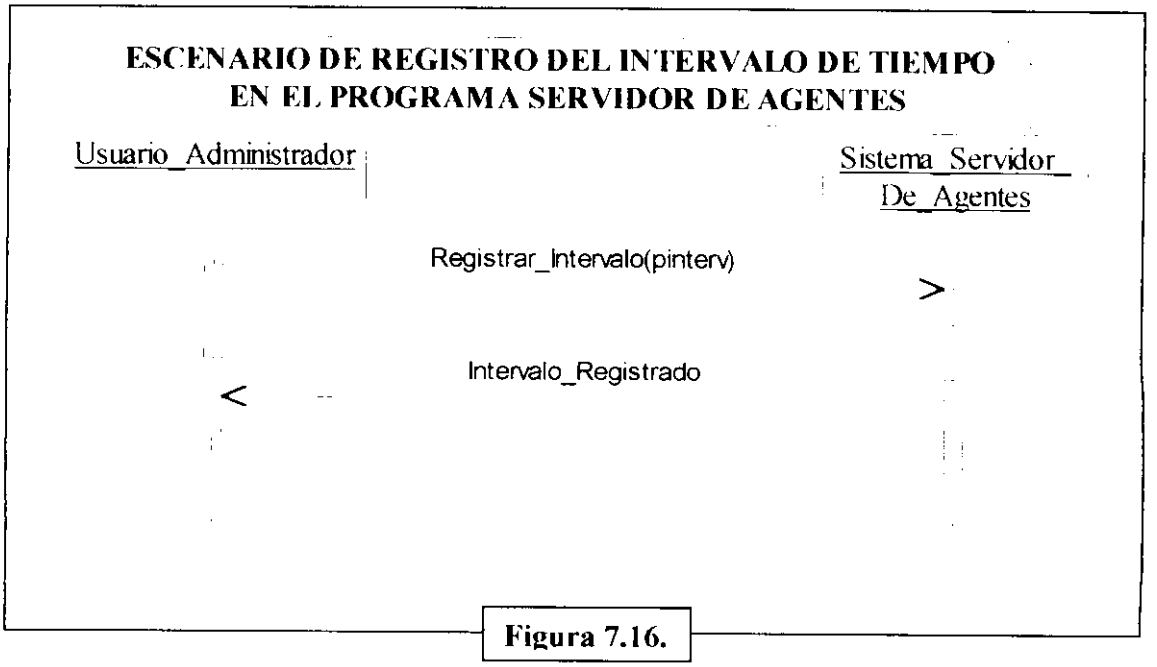


Figura 7.16.

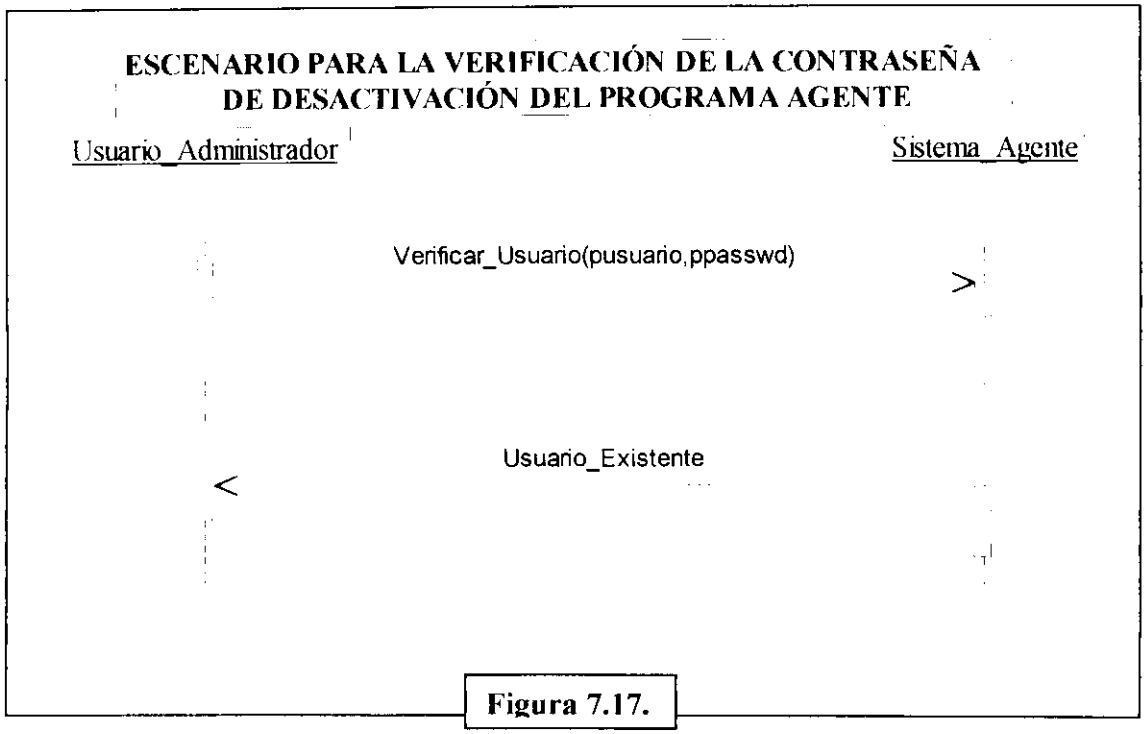


Figura 7.17.

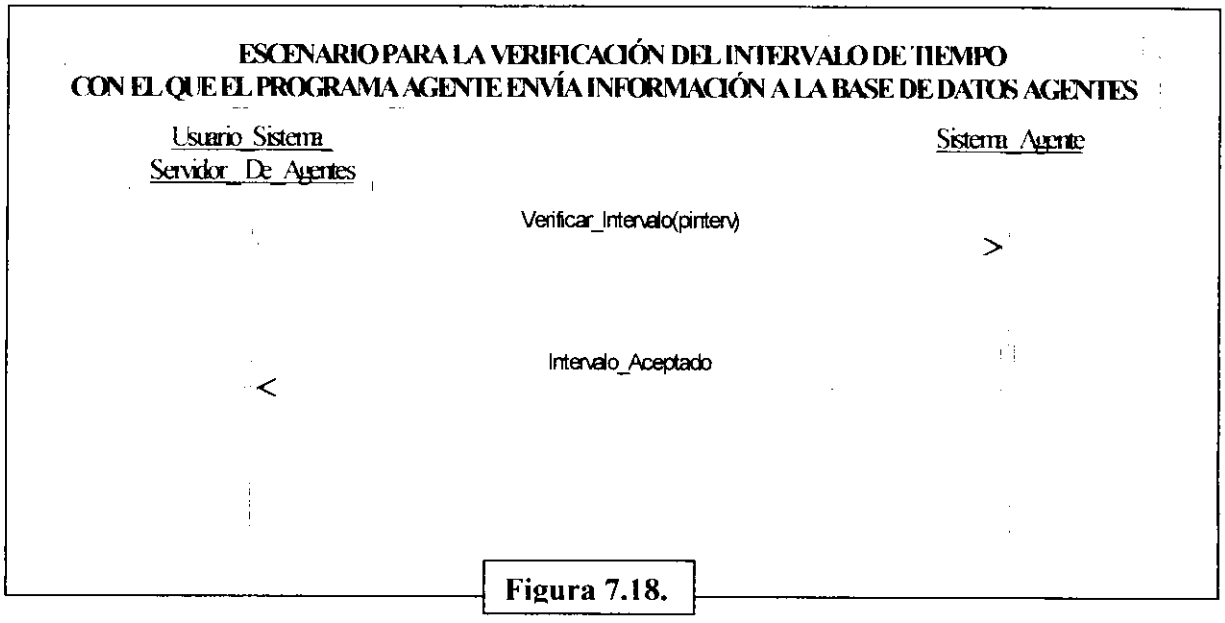


Figura 7.18.

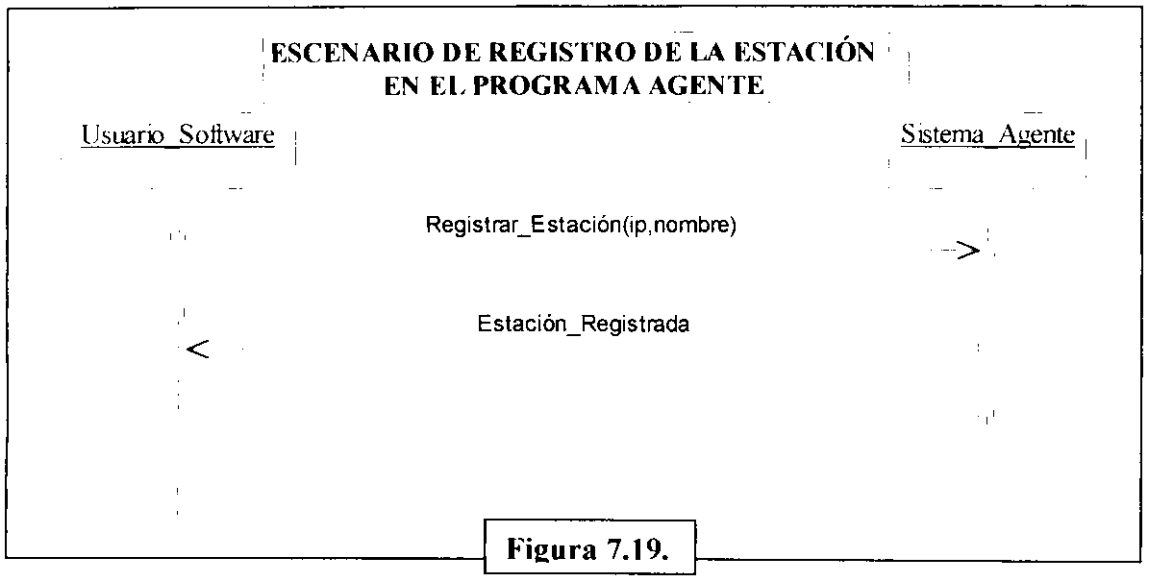


Figura 7.19.

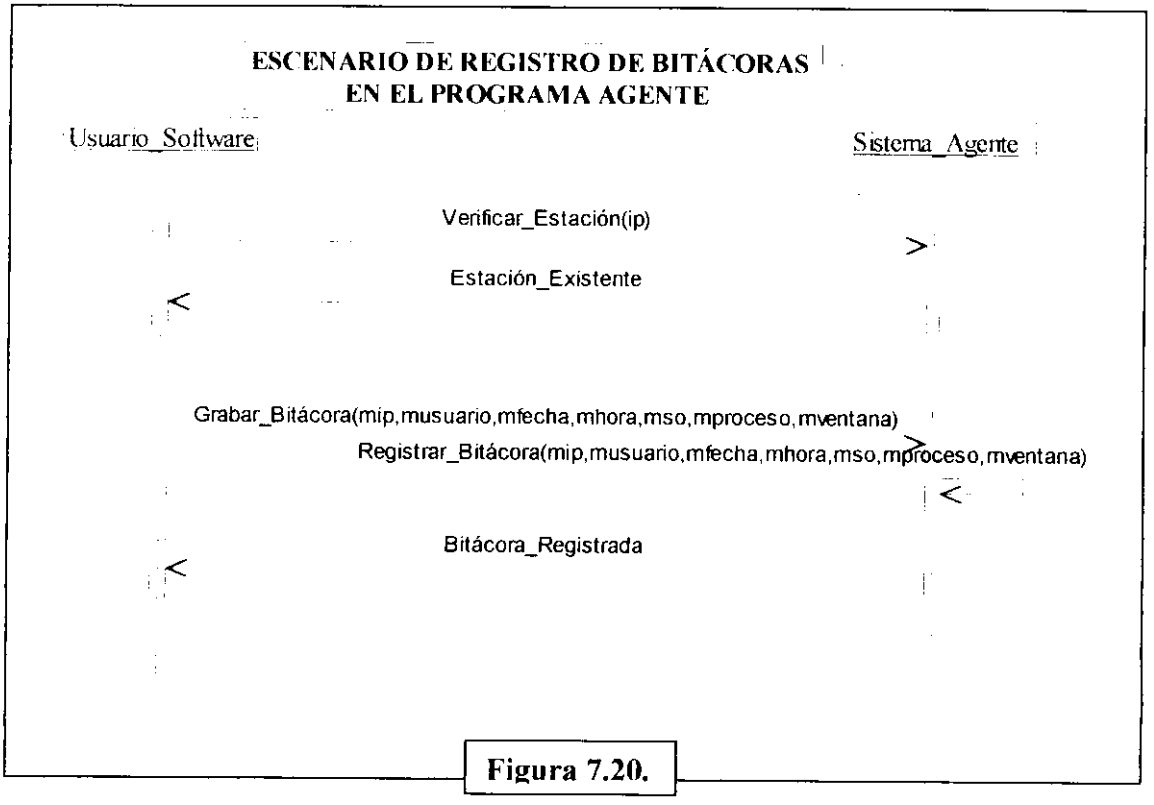


Figura 7.20.

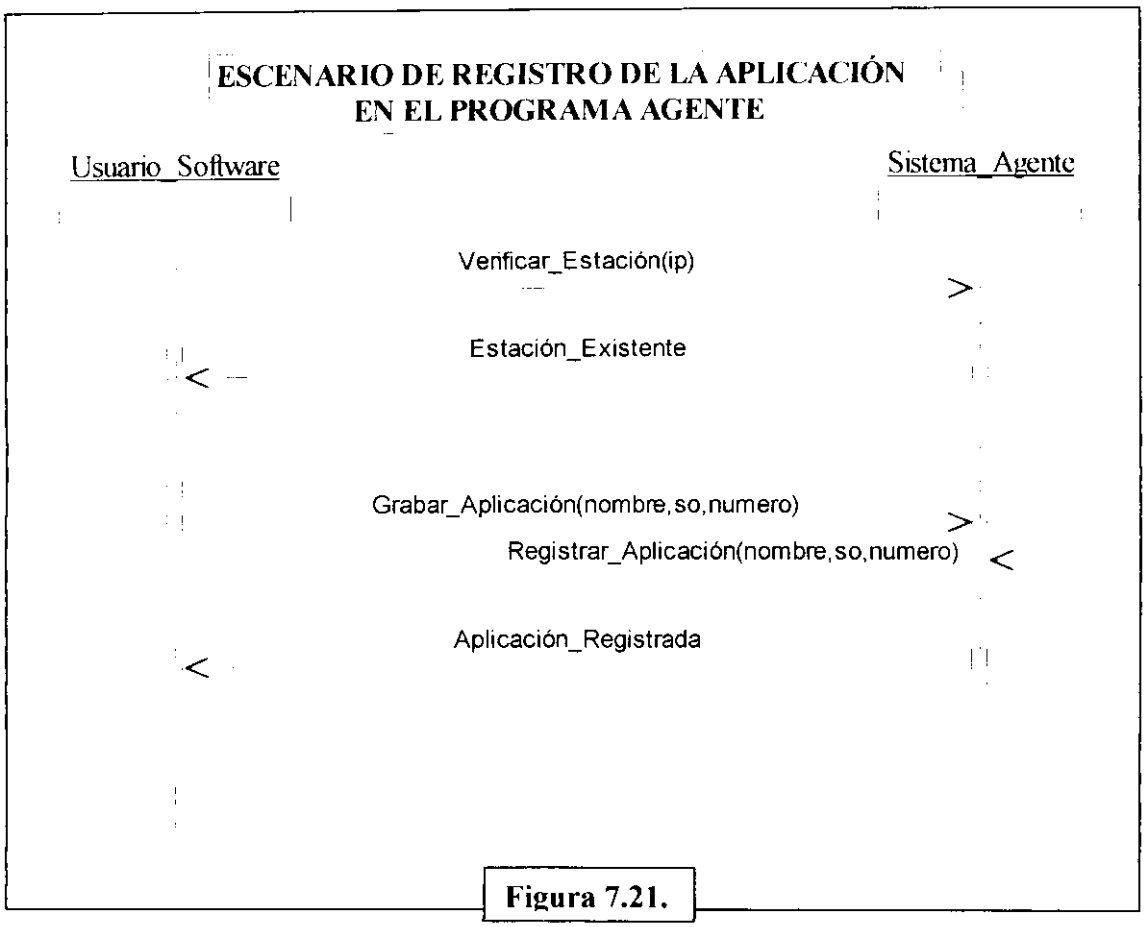


Figura 7.21.

7.3. DISEÑO

Este diseño debe preocuparse de obtener las mejores respuestas y resultados del análisis y la implementación de envío al registro de Windows y al Autorun de Linux un programa agente a todas las estaciones y se ejecutará automáticamente al iniciar el equipo.

7.3.1. Diseño de la Base de Datos

Para diseñar una base de datos, es necesario seguir determinados pasos o tener un método. Entendemos que siempre es necesario primero realizar un adecuado análisis del sistema a implementar y luego realizar un adecuado diseño.

Un modelo sencillo es el siguiente:

- Análisis de requerimientos.
- Crear el diseño de la base de datos: Diseño lógico y diseño físico.
- Implementar la aplicación que utiliza la base de datos.
- Revisión y ajuste de la aplicación y la base de datos.
- Mantenimiento de la aplicación y de la base de datos.

El resultado de cualquier método de diseño elegido para construir una base de datos relacional será un conjunto de tablas o relaciones, un conjunto de dominios de datos, vistas, código para restricciones, código para ejecutar ante determinados eventos (triggers), un conjunto de índices, definiciones de áreas de almacenamiento, y mapas de almacenamiento asociados a aquellas tablas.

7.3.1.1. Diseño Lógico

Para realizar un buen diseño lógico, se debe entender las relaciones existentes entre todos los objetos de la base de datos. En el diseño lógico se representan las entidades, se eliminan atributos redundantes y se produce un esquema con las tablas que representan los datos de forma óptima. Es importante también entender qué tablas son más accedidas, por qué usuarios y para qué. Cada acceso a la base de datos se denomina una transacción, entonces es importante analizar cuales son las transacciones típicas que se realizarán en la base de datos para tenerlas en cuenta en nuestro diseño.

Para el modelo de base de datos relacional, se aplica la normalización de las tablas para reducir la estructura a su forma más simple, remover las columnas redundantes e identificar todos los datos que son dependientes de otros. Hay cinco grados de normalización, pero los datos en tercera forma normal se dicen que están completamente normalizados. Sin embargo, el grado de normalización de los datos depende de la aplicación en particular. Si los datos no están lo suficientemente normalizados, hay muchos datos repetidos en las tablas y surgen problemas de actualización, modificación y eliminación de aquellos datos. Si existe demasiada normalización, tenemos mayor tiempo de acceso a los datos, debido a que se deben unir varias tablas para recuperar la información.

Una normalización efectiva consiste de las siguientes tareas:

- Verificar que cada columna de cada tabla muestre un hecho relevante.

- Traer cierta cantidad de redundancia dentro de la tabla para mejorar la performance eliminando la necesidad de uniones que crucen más de dos o tres tablas para las transacciones más realizadas por la/las aplicaciones.

Respecto al análisis de las transacciones, van en conjunto con los resultados del análisis de requerimientos y el análisis de datos que involucra trasladar los requerimientos en ítems de datos, sobre relaciones entre columnas de las tablas, y la respectiva normalización. Durante este proceso, se verifica que el modelo lógico de la base de datos pueda soportar las transacciones requeridas por los usuarios del sistema.

Se debe contar con la siguiente información acerca de cada transacción:

- Una secuencia de pasos que muestra que tablas y columnas son accedidas y si son actualizadas.
- Medidas de base para tiempos de respuesta y cantidad de transacciones esperadas.
- Estimación de cantidad de accesos a la base de datos.

El análisis de datos genera la siguiente información:

- Una lista de tablas, columnas e índices.
- Un gráfico de entidad/ relación.

El análisis de las transacciones debe brindar respuestas a:

- ¿Qué caminos sigue cada transacción en nuestro modelo?
- ¿Cuáles son los puntos de entrada?

- ¿Existen relaciones necesarias y no definidas?
- ¿Qué relaciones están definidas y no se utilizan?
- ¿Cuáles caminos son los más accedidos o utilizados por las transacciones?
- ¿Qué caminos son usados repetidamente sólo para obtener una sola columna?

Diferentes tipos de tablas en la base de datos crecen de diferente manera.

Algunas tablas permanecen constantes mientras que otras tablas crecen sin límite.

Las filas de la base de datos caen en dos categorías:

Filas de referencia.- son filas en las tablas que se vuelven estables en número sobre el tiempo y representan información que existe como referencia a través del tiempo. Por ejemplo la tabla de estaciones (estacion) de la Red de la Escuela de Ingeniería de Sistemas de la PUCESA.

Filas de eventos.- por ejemplo la fila de movimientos de la tabla (mvitacora) y de la tabla (aplicacion) de la Red de la Escuela de Ingeniería de Sistemas de la PUCESA. La meta principal del almacenamiento histórico es preservar la información de eventos viejos removiéndolos de la base de datos activa. Teniendo cuidado de que cuando eliminamos los eventos podemos perder su referencia al objeto, entonces debemos planear y contemplar en nuestro modelo como guardaremos esta información.

El diseño lógico escogido para nuestro objeto de estudio se muestra en la **(Figura 7.22.)**.

7.3.1.2. Diseño Físico

El diseño físico consiste en una descripción de cómo se almacenarán físicamente los datos en el servidor. Debido a que la descripción del diseño físico de la base de datos es el resultado de la integración de toda la información acerca de las tablas y columnas, relaciones entre tablas, de todas las transacciones, la descripción se crea sobre la base de un conocimiento exacto de cómo el administrador de base de datos almacena las tablas y otras estructuras como índices, sobre el disco. Para bases de datos de 100 Mb. a 1 Gb.

Con más de 15 tablas se debe realizar un adecuado diseño físico para lograr mejores rendimientos.

Como ya se explicó toda la información referente a las tablas de estaciones activas (estacion), tablas de movimientos (mvitacora) y (aplicación) se almacena en la Base de Datos MYSQL ya sea de Linux o Windows y como dicha base se instaló en el disco duro, entonces es ahí donde se almacenarán los datos.

Los principales componentes que se almacenan en disco al crear una base de datos son:

- El archivo raíz de la base de datos contiene punteros a todas las áreas de almacenamiento.
- Las tablas del sistema contienen las definiciones de la base de datos.
- Los datos e índices, en áreas de almacenamiento preestablecidas.

El diseño físico escogido para nuestro objeto de estudio se muestra en la **(Figura 7.23.)**.

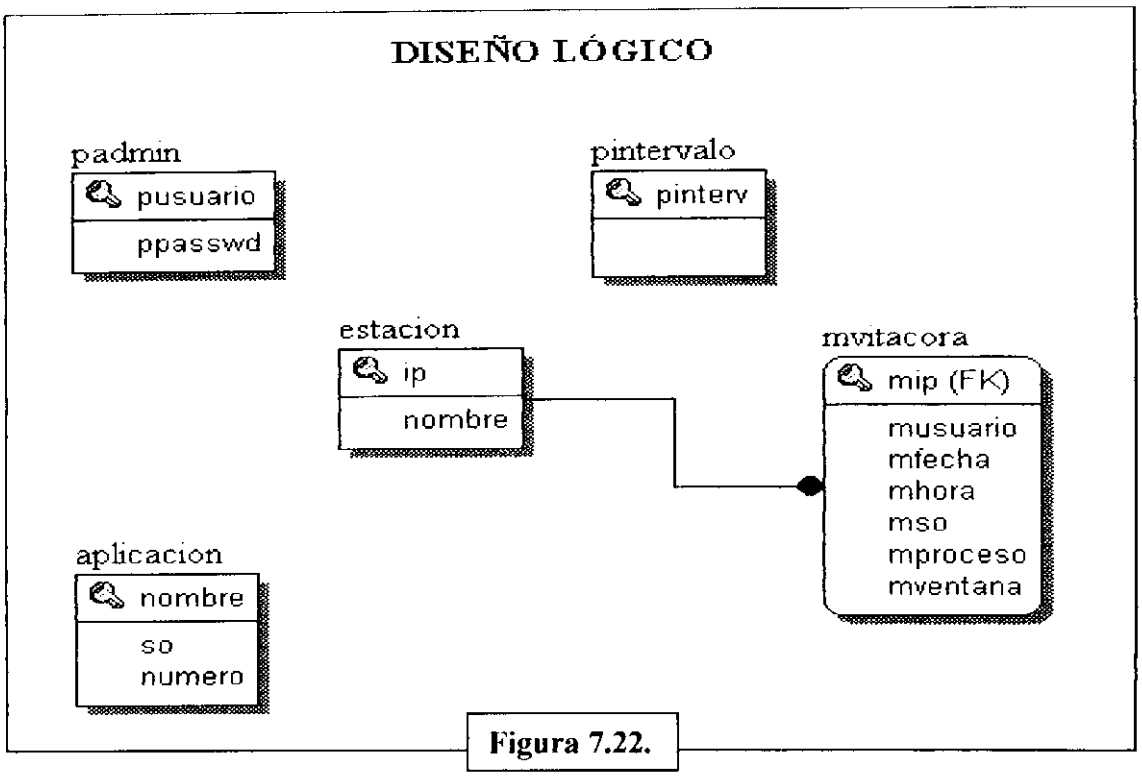


Figura 7.22.

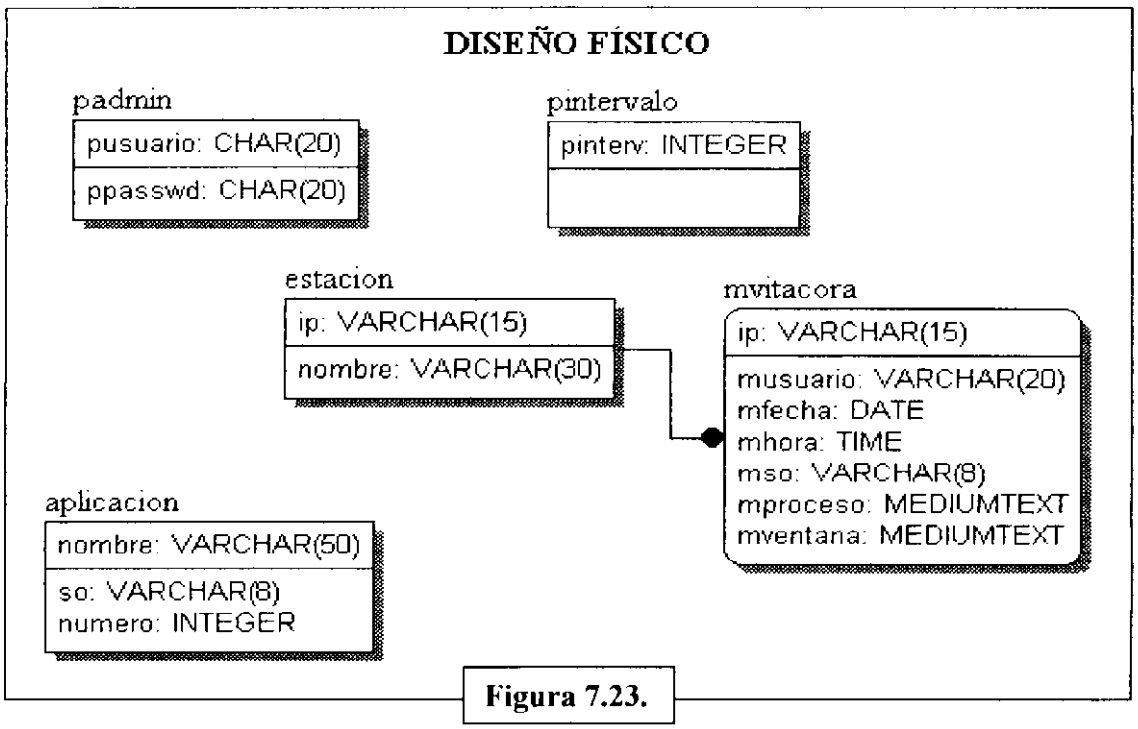


Figura 7.23.

7.3.2. Diseño Orientado a Objetos

El DOO se basa en la actividad del diseño del software, por lo que se hace indispensable la productividad para realizar mejoras en la calidad de software.

Se debe tomar muy en cuenta el diseño para que permita hacer modificaciones futuras y también constar partes fijas que no permitan el cambio, debido a que son estables.

El DOO va a tener los siguientes objetivos que son:

- Mejorar la calidad de la aplicación.
- Permitir dar un mejor mantenimiento.
- Obtener una mayor productividad.

7.3.2.1. Diagramas de Interacción

Los diagramas de interacción son modelos que describen la manera en que colaboran grupos de objetos para cierto comportamiento.

El UML define dos tipos de diagramas de interacción:

- Diagramas de Secuencia(se trató con anterioridad en el Análisis Orientado a Objetos).
- Diagramas de Colaboración (se tratará aquí en el Diseño Orientado a Objetos).

Diagramas de Colaboración

La segunda forma del diagrama de interacción es el diagrama de colaboración.

En los diagramas de colaboración, los objetos ejemplo se muestran como iconos. Las flechas indican, como en los diagramas de secuencia, los mensajes enviados. Sin embargo, en esta ocasión la secuencia se indica numerando los mensajes.

Los diagramas de colaboración que se muestran a continuación en la **(Figura 7.24.)**, **(Figura 7.25.)**, **(Figura 7.26.)**, **(Figura 7.27.)**, **(Figura 7.28.)**, **(Figura 7.29.)** y **(Figura 7.30.)** se han creado basándose en los diagramas de secuencia denominados en el AOO y describen cada operación del sistema allí especificada.

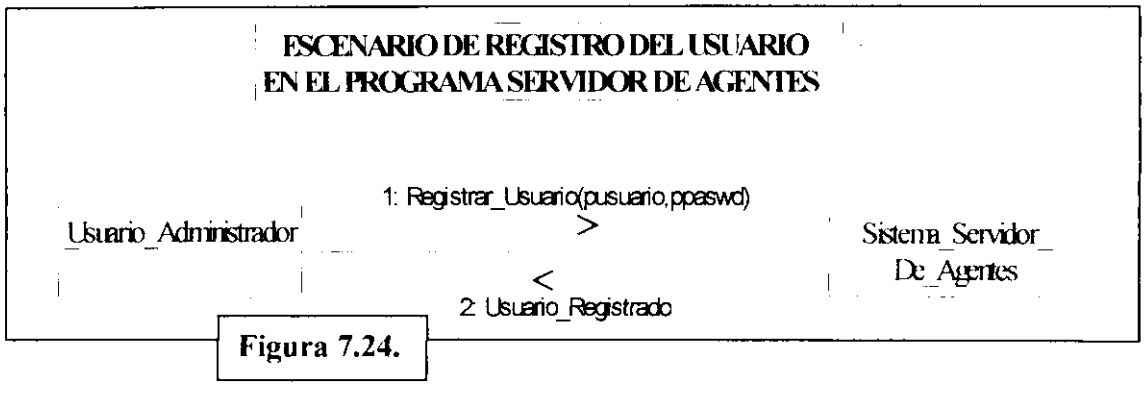


Figura 7.24.



Figura 7.25.

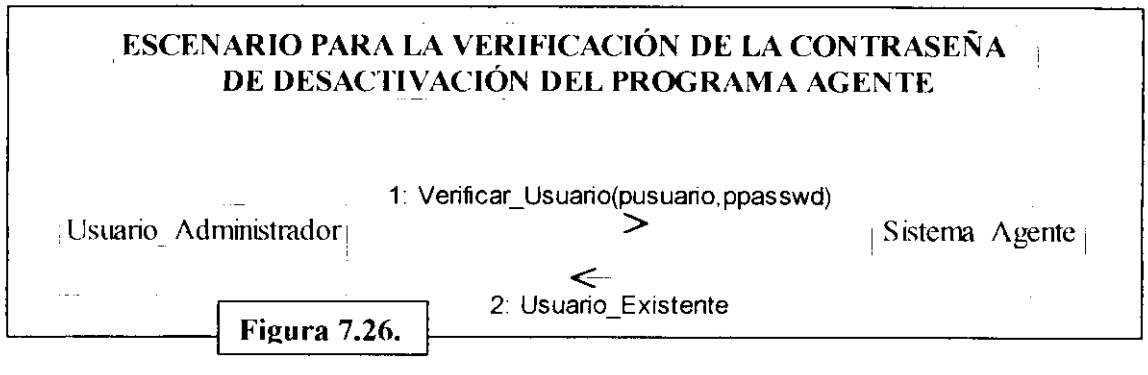


Figura 7.26.

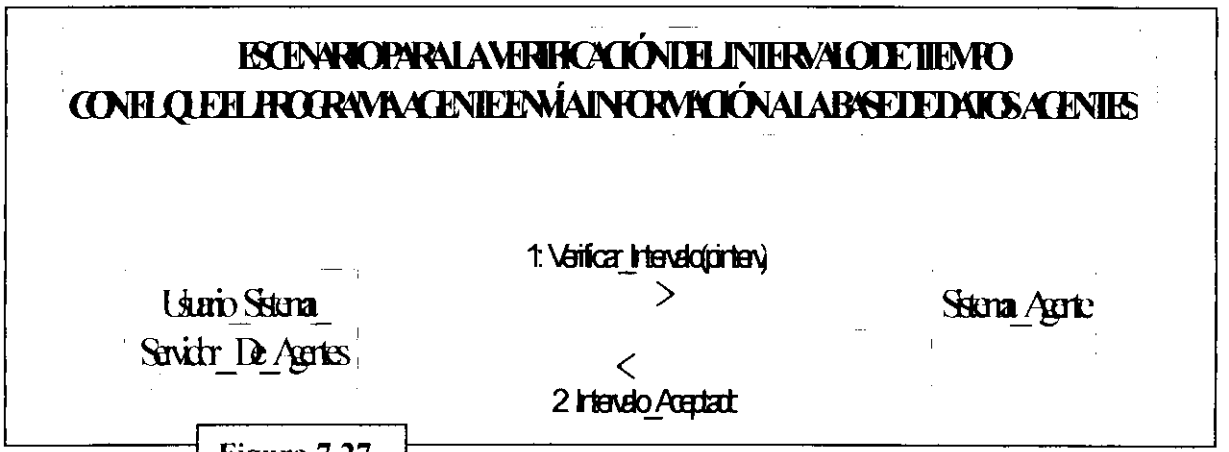


Figura 7.27.

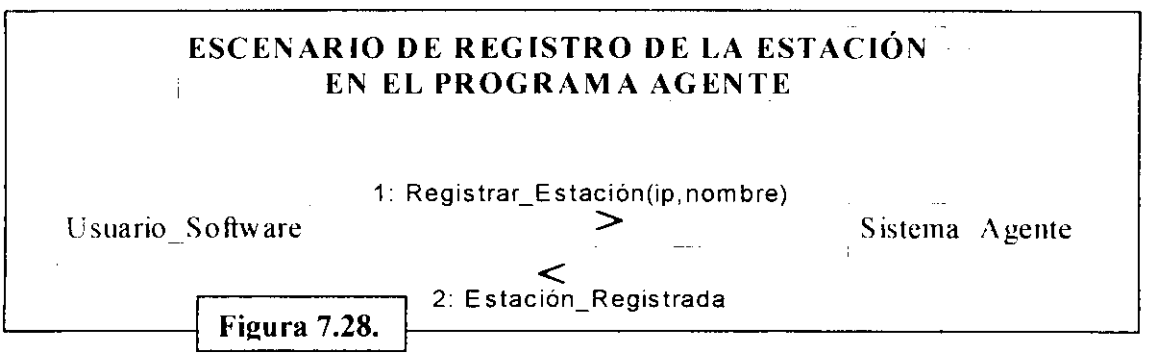


Figura 7.28.

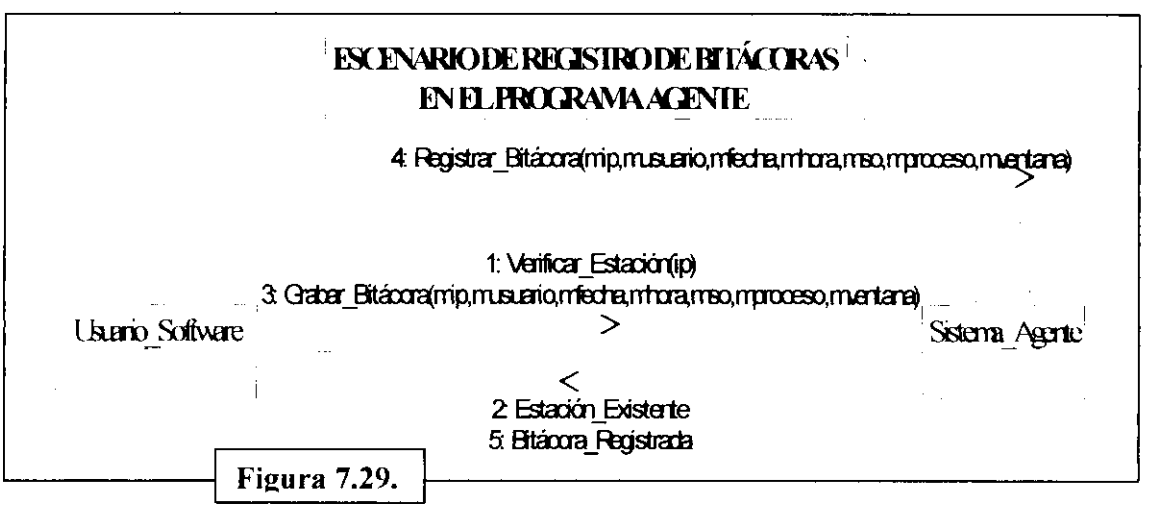


Figura 7.29.

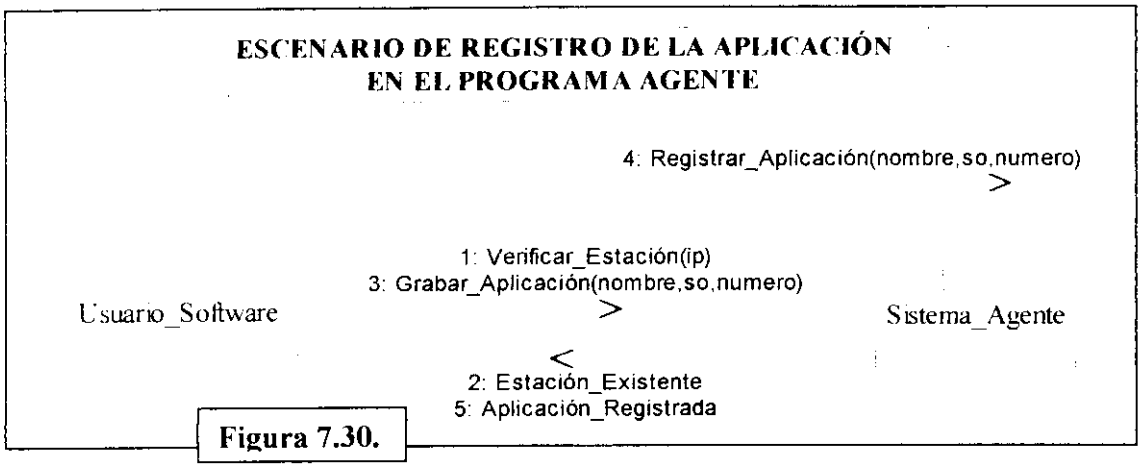


Figura 7.30.

7.3.2.2. Definir como se hace cada operación del sistema

ESCENARIO DE REGISTRO DEL USUARIO

EN EL PROGRAMA SERVIDOR DE AGENTES

Operación:	Registrar Usuario
Descripción:	Permite registrar un nuevo usuario en la Tabla de Usuarios (padmin) para poder desactivar el Programa Agente de la Bandeja de Entrada o área de notificación de Windows y en el escritorio de Linux.
Lee:	supplied pusuario:nombre de usuario supplied ppaswd:password
Modifica:	Tabla padmin new Usuario
Envía:	
Asume:	pusuario no existe
Resulta:	Se añade un registro del usuario en la Tabla de Usuarios (padmin),

se actualiza pusuario y ppaswd.

ESCENARIO DE REGISTRO DEL INTERVALO DE TIEMPO EN EL PROGRAMA SERVIDOR DE AGENTES

Operación: Registrar Intervalo

Descripción: Permite registrar un nuevo intervalo en la Tabla de Intervalos de Tiempo (pintervalo) para poder enviar un valor del intervalo de tiempo con el que se desea que el Programa Agente envíe información a la Base de Datos agentes

Lee: supplied pinterv:intervalo de tiempo

Modifica: Tabla pintervalo

new Intervalo

Envía:

Asume: pinterv no existe

Resulta: Se añade un registro del intervalo en la Tabla de Intervalos de Tiempo (pintervalo),

se actualiza pinterv.

ESCENARIO PARA LA VERIFICACIÓN DE LA CONTRASEÑA DE DESACTIVACIÓN DEL PROGRAMA AGENTE

Operación: Verificar Usuario

Descripción: Permite verificar si el usuario existe en la Tabla de Usuarios (padmin)

para poder desactivar el Programa Agente de la Bandeja de Entrada o área de notificación de Windows y en el escritorio de Linux

Lee: supplied pusuario:nombre de usuario

supplied ppaswd:password

Modifica:

Envía:

Asume: pusuario existe

Resulta: Se verifica si el registro del usuario existe en la Tabla de Usuarios (padmin),

se consulta pusuario y ppaswd.

ESCENARIO PARA LA VERIFICACIÓN DEL INTERVALO DE TIEMPO CON EL QUE EL PROGRAMA AGENTE ENVÍA INFORMACIÓN A LA BASE DE DATOS AGENTES

Operación: Verificar Intervalo

Descripción: Permite verificar si el intervalo existe en la Tabla de Intervalos de Tiempo (pintervalo) para poder enviar un valor del intervalo de tiempo con el que se desea que el Programa Agente envíe información a la Base de Datos de MYSQL denominada (agentes)

Lee: supplied pinterv:intervalo de tiempo

Modifica:

Envía:

Asume: pinterv existe

Resulta: Se verifica si el registro del intervalo existe en la Tabla de Intervalos de Tiempo (pintervalo),

se consulta pinterv.

ESCENARIO DE REGISTRO DE LA ESTACIÓN EN EL PROGRAMA AGENTE

Operación: Registrar Estación

Descripción: Permite registrar una nueva estación en la Tabla de Estaciones activas (estacion)

Lee: supplied ip:Dirección Ip capturada de la estación

supplied nombre:Nombre del Host capturada de la estación

Modifica: Tabla estacion

new Estación

Envía:

Asume: ip no existe, interviene para la búsqueda del ip de la estación activa en intervalos establecidos en la Tabla (intervalo)

Resulta: Se añade un registro de la estación en la Tabla de Estaciones activas (estacion),

se actualiza ip y nombre.

ESCENARIO DE REGISTRO DE BITÀCORAS

EN EL PROGRAMA AGENTE

Operación: Registrar Bitácora

Descripción: Permite registrar una nueva bitácora en la Tabla de Movimientos (mvitacora) que son los procesos que se capturan de una estación de trabajo activa

Lee: supplied mip: Dirección Ip capturada de la estación

supplied musuario: Nombre del Usuario capturada de la estación

supplied mfecha: Fecha capturada del Servidor

supplied mhora: Hora capturada del Servidor

supplied mso: Sistema Operativo

supplied mproceso: Procesos capturados de la estación

supplied mventana: Ventanas capturadas de la estación

Modifica: Tabla mvitacora

new Bitácora

Envía:

Asume: Se graban todos los registros de las estaciones activas con intervalos de tiempo establecidos en la Tabla de Intervalos de Tiempo (pintervalo) buscando la estación de trabajo existente es decir el ip de la Tabla (estacion) que será el mismo mip de la Tabla (mvitacora)

Resulta: Se añade un registro de bitácoras en la Tabla de Movimientos (mvitacora),

se actualiza mip, musuario, mfecha, mhora, mso, mproceso, mventana.

ESCENARIO DE REGISTRO DE APLICACIÓN

EN EL PROGRAMA AGENTE

Operación: Registrar Aplicación

Descripción: Permite registrar una nueva aplicación en la Tabla (aplicacion) que son las aplicaciones de todas las estaciones de trabajo activas, de esta tabla podemos extraer información para los reportes

Lee: supplied nombre: Nombre de la aplicación
supplied so: Sistema Operativo
supplied numero: Número de aplicaciones producidas

Modifica: Tabla aplicación
new Aplicación

Envía:

Asume: Se incrementa en uno el valor del campo numero dependiendo de la aplicación ejecutada en las estaciones de trabajo activa de acuerdo al intervalo establecido en la Tabla (pintervalo)

Resulta: Se añade un registro de aplicación en la Tabla (aplicacion),
se actualiza nombre, so, numero.

7.4. PROGRAMACIÓN ORIENTADA A OBJETOS

La OOP no es un lenguaje de programación, puede aplicarse a cualquier lenguaje, y de hecho hoy en día está disponible en mayor o menor medida en todos los lenguajes tradicionales (C se ha convertido en C++, Pascal en Delphi, VB incorpora parte de la OOP) y no aparece un lenguaje nuevo sin que incluya OOP (como es el caso de Java). Es por esto que todo lo que aquí se diga pueda ser aplicado a cualquier lenguaje OOP.

7.4.1. Código Fuente

Todo el Código Fuente empleado para nuestro objeto de estudio tanto para el "Sistema de Control de Estaciones de Trabajo en la Red de la Escuela de Ingeniería de Sistemas de la PUCESA" que es el programa agente que envía información a la Base de Datos MYSQL denominada agentes y el "Sistema Servidor de Agentes" que es el programa que provee usuario, intervalo, fecha y hora al programa agente se almacenará en el CD que se entregará en Biblioteca o en el Laboratorio de Sistemas.

CONCLUSIONES

Se detallan estas conclusiones que son de suma importancia para que los software desarrollados sean utilizados de la mejor manera y para la realización de nuevos proyectos.

- De acuerdo a la multiplataforma que enfocamos para nuestro plan de disertación se consideró como posible herramienta de desarrollo los lenguajes de programación orientados a objetos **Java o Visual Age for Java** en colectividad con **MYSQL o PostgreSQL.** Pero al momento de comenzar a desarrollar la disertación tuvimos problemas porque no conseguimos la suficiente ayuda para desarrollar nuestro sistema en estos lenguajes y optamos en escoger como posible herramienta de desarrollo los lenguajes de programación orientados a objetos **Delphi 6.0** para Windows y **Kylix 1.0** para Linux en colectividad con la Base de Datos **MYSQL;** ya que investigamos que son herramientas aptas para el desarrollo de este tipo de aplicaciones de una manera más fácil y comprensible porque nos proporciona un ambiente gráfico. Los lenguajes de programación Delphi, Kylix ofrecen una de las mejores opciones para el desarrollo cruzado en Windows y Linux.
- Los lenguajes de programación Delphi, Kylix no son de libre distribución ya que se deben comprar para desarrollar aplicaciones.
- Las herramientas utilizadas para la creación de estas aplicaciones son muy potentes en cuanto al manejo de bases de datos relacionales, las cuales brindan todas las herramientas necesarias como interactuar con las API, conexiones con

Bases de Datos, crear servicios de Internet, diseño de gráficos estadísticos, reportes, etc.

- El sistema operativo Linux que tiene licencia GNU hace posible crear un entorno confiable, multiusuario y con una infinidad de servicios para Internet.
- MYSQL es un servidor de Base de datos multiusuario, multiplataforma, con código fuente abierto ya que se puede utilizar, modificar bajo procedimientos GNU.
- Hemos determinado mediante nuestra investigación que se puede desarrollar cualquier tipo de aplicaciones desde los cálculos más complejos hasta aplicaciones científicas.
- Los objetivos planteados al inicio de la disertación de la tesis, se han cumplido obteniendo como resultado, dos productos de Software para el control de estaciones en la red.
- Un programa agente que registra las actividades de una PC, distribuido en distintas máquinas de la red.
- El objetivo de la arquitectura cliente/servidor ofrece una alternativa de diversidad de plataformas de proceso, aplicaciones y configuraciones que van a implementar los usuarios.
- El mayor problema que tuvimos fue en la configuración de la Red Linux - Windows; ya que al momento de instalar Linux no reconocía la tarjeta de red SIS 900 y tuvimos que comprar una nueva tarjeta. Por más que intentamos cambiar en todos los archivos de configuración de la red no logramos nada.
- Para que funcione el componente CLX TSQLConnection de la paleta dbExpress tuvimos que conseguir de Internet algunas librerías tanto para Delphi 6.0 el **libmySQL.dll** como para Kylix 1.0 el **libmysqlclient.so.6.0.0**. Y continuaba el

problema porque nos detectaba un error de usuario no permitido, pensábamos que faltaba dar permisos al usuario en la Base de Datos mysql en la tabla user pero nos equivocamos; el error había sido que las librerías deben ser compatibles con la versión de MYSQL.

- Al compilar el Sistema Servidor de Agentes en la plataforma Linux, descubrimos que a pesar de que el Kylix 1.0 es compatible con Delphi 6.0 de Windows no funcionó al 100%; ya que tuvimos que renombrar algunas librerías y componentes y en otros casos hasta borrarlos para que corra el programa. No funcionó los reportes tanto estadísticos como normales y la visualización de la ayuda del Sistema, lo demás funcionó correctamente.
- El tiempo de respuesta depende del intervalo de tiempo que el administrador decida en que la información capturada se envíe a la Base de Datos.
- Por falta de conocimiento en la plataforma Linux no pudimos incrementar las seguridades suficientes para el Sistema.
- La implementación de las aplicaciones desarrolladas ayudará a un mejor control de las estaciones de trabajo de la red.

RECOMENDACIONES

Al finalizar, con la investigación y desarrollo de nuestra disertación de grado sugerimos las siguientes recomendaciones:

- La importancia creciente de este tema que ha revolucionado como es la de los programas agentes que sirven para automatizar distintos tipos de tareas en un computador en el mundo contemporáneo y futuro. Se recomienda más énfasis en el área de Inteligencia Artificial, además se debería crear un laboratorio para investigaciones, desarrollo de proyectos científicos como en otras universidades del mundo.
- Debido a que este tema es nuevo se recomienda formar un equipo que puede estar integrado por profesores y alumnos de la universidad para investigaciones, desarrollo de este tipo de aplicaciones. Para lo cual la universidad debería financiar los costos de investigación, Hardware y Software necesarios para realizar estos trabajos.
- Estas aplicaciones van dirigidas directamente al administrador de la red ya que sólo esta persona es la directamente responsable del control de los equipos y software existente.
- El volumen de datos que se maneja en la Base de Datos es muy grande para lo cual se recomienda al administrador realizar un adecuado mantenimiento.
- Se recomienda que el administrador inicie el servidor de Base de datos MYSQL y ejecute la aplicación Servidor de Agentes antes de que se utilicen las estaciones e inicie el programa agente.

- Implementar el proyecto en los equipos de la red de la Pontificia Universidad Católica del Ecuador Sede Ambato.
- Si la universidad posee recursos económicos debería comprar un software que permita que el sistema que hemos realizado se pueda instalar en cualquier computadora, sin necesidad de tener instalado Delphi y Kylix, es decir poder hacer instalador que recoja todas las librerías que posee tanto el Delphi como el Kylix. Se recomienda conseguir el InstallShield profesional.
- Se recomienda obtener el Kylix 2.0, porque esta versión debe poseer más componentes compatibles y se podrá visualizar el Sistema Servidor de Agentes de la misma forma como funciona en Windows.

BIBLIOGRAFÍA

- **PRESSMAN Roger S.** Ingeniería del Software, 3ra Edición.
- **PRESSMAN Roger S.** Ingeniería del Software, 4ta Edición.
- **FOWLER Martín, SCOTT Kendall.** UML Gota a Gota.
- **FREEDMAN Alan.** Diccionario de computación Bilingüe, Tomo I, II, III.
- **BOOCH Grady.** Análisis y Diseño Orientado a Objetos.
- **MANNING Michelle M.** Delphi 2 Guia Oficial de Borland.

Sitios Web:

- **Redes de Computadoras.**

  http://www.ing.ula.ve/~mfrand/

- **Redes, Protocolo TCP/IP, Intranet, Internet, Sistemas Operativos**

  http://www.monografias.com

- **Redes, Intranet, Internet**

  http://www.microsoft.com

- **Las Redes de Computadoras**

  http://www.ing.ula.ve/~mfrand/

- **Modelo de Referencia TCP/IP.**

  http://www.tau.org.ar/base/lara.pue.udlap.mx/redes/indexred.htm

- **Control de Sockets.**

  http://www.arrakis.es/~alcon/scrip-soc.htm

- **Kylix Developers Guide.**

  http://www.amazon.com/exec/obidos/ASIN/0672320606/qid%3D1010056998/sr%3D8-1/ref%3Dsr%5F8%5F3%5F1/103-1346812

- **Ejemplos de Kylix.**

  http://www.developpez.com/kylix/index.htm

- **Componentes para Kylix.**

  http://www.tempest-sw.com/freeware/freeware.cgi?kylix

- **Tutoriales Kylix.**

  http://www.developpez.com/kylix/tutoriels.htm

- **Kylix Tools.**

  http://www.borland.com/kylix/resources/kylixtools.html#installation

- **Programas en Kylix.**

  http://www.delphi-jedi.org/Jedi:JEDIXLIBRARY:0

- **Foros Grupo Kylix.**

  http://www.latiumsoftware.com/en/subscription.php3

- **Trucos para Delphi.**

  http://www.q3.nu/

  http://trucomania.com

- **Utilidades, Foros, Trucos , Componentes para Delphi, Kylix.**

  http://www.clubdelphi.com

- **HelpScript.**

  http://www.tornado.be/~johnfg/

- **Mysql.**

  http://www.mysql.com/

- **Mysql-Front.**

  http://www.anse.de/mysqlfront/

- **libmySQL.dll.**

  http://www.fichtner.net/delphi/mysql.delphi.phtml

- **Agentes.**

  http://www.acm.org/crossroads/espanol/xrds5-4/dumbagents.html

- **Bitácoras.**

  http://www.bitacoras.net

ANEXO A

MANUAL DEL USUARIO

¿CÓMO UTILIZAR EL SERVIDOR DE AGENTES SOFTWARE ® ING.2002?

Una vez ejecutado el programa ServidorAgeP. Aplicación para las plataformas Linux y Windows, aparecerá la siguiente figura.

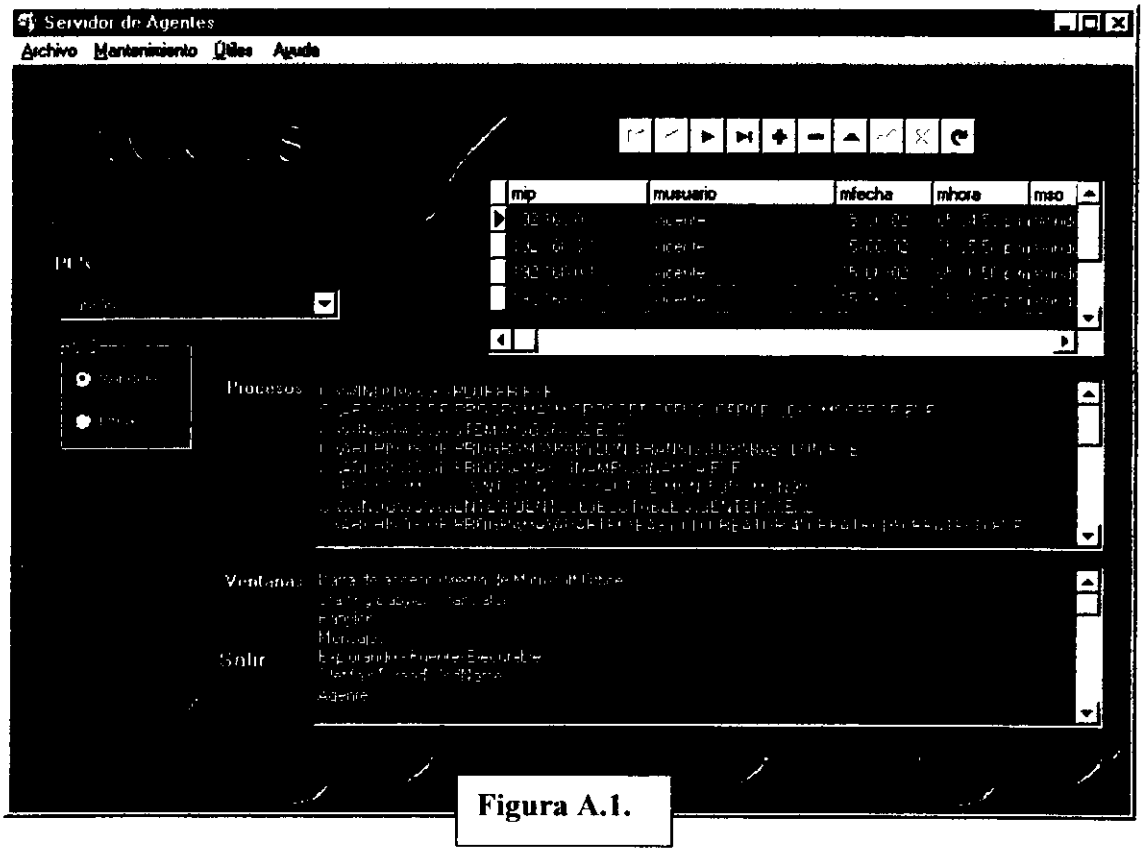


Figura A.1.

En la **(Figura A.1.)** se visualiza los datos haciendo doble click en PC's, seleccione la estación.

La opción **S.O** (Sistema operativo):

- Windows.
- Linux.

La opción **WINDOWS** permite visualizar los datos registrados de las estaciones de Windows.

La opción **LINUX** permite visualizar los datos registrados de las estaciones de Linux.

La barra de navegación para desplazarse por los registros.

El botón Refrescar para actualizar los datos.

El botón Salir para cerrar la aplicación, donde se muestra la **(Figura A.2.)**.

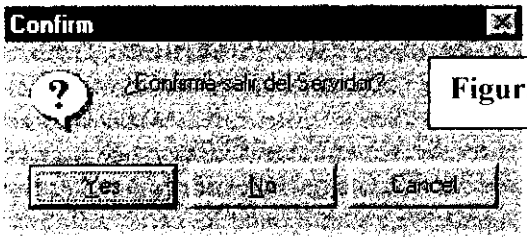


Figura A.2.

Esta pantalla contiene los siguientes menús:

- Archivo.
- Mantenimiento.
- Útiles.
- Ayuda.

El menú **ARCHIVO** que contiene:

- Salir.

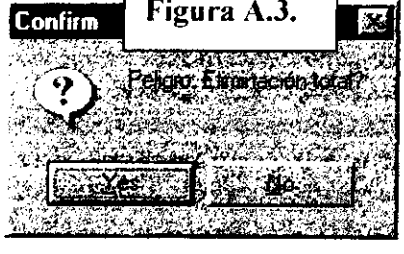
La opción **SALIR** permite cerrar la aplicación donde se visualiza la **(Figura A.2.)**.

El menú **MANTENIMIENTO** que contiene:

- Eliminación.
- Parámetros.

La opción **ELIMINACIÓN** permite borrar todos los registros de la tabla de estaciones (estacion), de las tablas de movimientos de las estaciones activas (mvitacora) y (aplicacion), menos los registros de parámetros, es decir registros de las tablas (padmin) y (pintervalo) donde se muestra la **(Figura A.3.)**.

Figura A.3.



La opción **PARÁMETROS** que contiene dos submenús:

- Usuarios.
- Intervalo.

El submenú **USUARIOS** que contiene:

- Altas Bajas.
- Consulta.

La opción **ALTAS BAJAS** permite ingresar a la pantalla que representa la **(Figura A.4.)** en la cual debemos ingresar, borrar el nombre del usuario y la clave. Toda la información acerca del nombre de usuario y contraseña solo la tiene el administrador del centro de cómputo, con los cuales tendrá acceso para cerrar los programas agentes.



Figura A.4.

El botón Salir para cerrar la pantalla.

La opción **CONSULTA** permite visualizar la pantalla que representa la **(Figura A.5.)** en la cual se muestra una consulta generalizada de los usuarios y su correspondiente contraseña.

Figura A.5.



El botón Salir para cerrar la pantalla.

El submenú **INTERVALO** que contiene:

- Altas Bajas.
- Consulta.

La opción **ALTAS BAJAS** permite visualizar la pantalla que representa la **(Figura A.6.)** en la cual debemos ingresar, borrar un valor numérico que corresponde al intervalo de tiempo en que el agente envía la información capturada a la base de datos.

Si el administrador necesita el registro de actividades de las estaciones de trabajo por minuto entonces deberá ingresar el valor de 60000.

Minutos con respectivos valores:

MINUTOS	VALORES
1	60000
2	120000
3	180000

Nota: Se deberá ingresar solamente un valor, si ingresa más de uno el agente tomará el valor más alto.

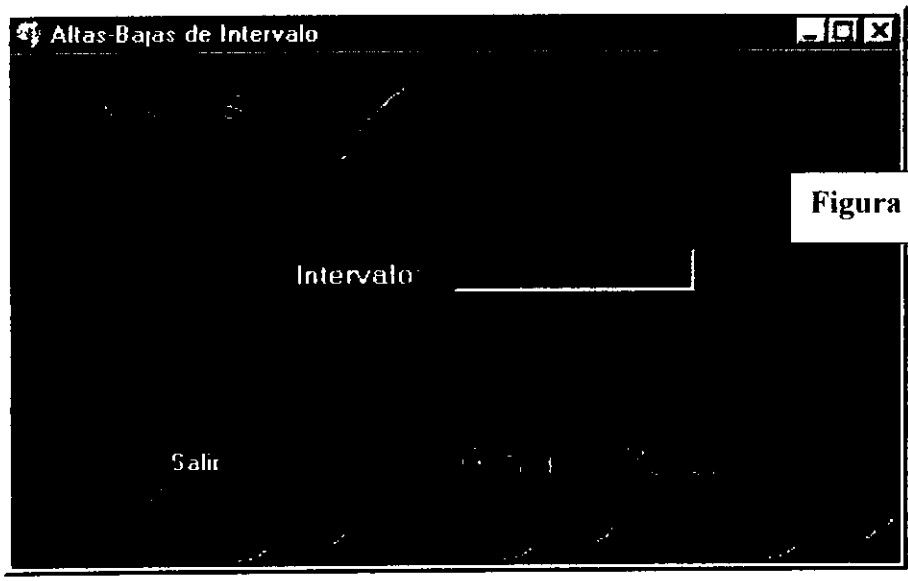


Figura A.6.

El botón Salir para cerrar la pantalla.

La opción **CONSULTA** permite visualizar la pantalla que representa **la (Figura A.7.)** en la cual se muestra una consulta generalizada del intervalo.

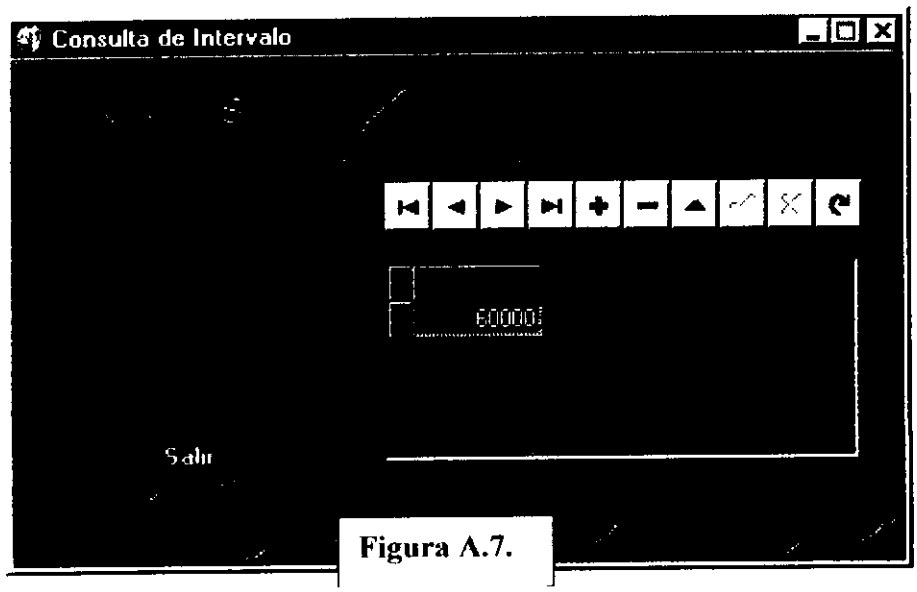


Figura A.7.

El botón Salir para cerrar la pantalla.

El menú **ÚTILES** que contiene:

- Mensajes.
- Estadísticas.
- Reportes.
- Colores.

La opción **MENSAJES** permite visualizar la pantalla que representa la **(Figura A.8.)** en la cual debemos ingresar la dirección IP o el Nombre, en el cuadro de texto digite el mensaje.

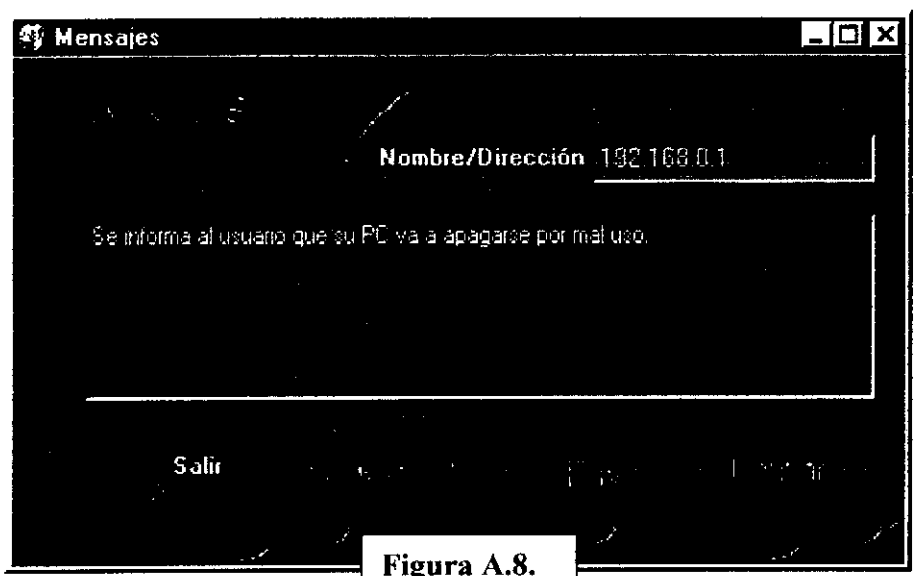


Figura A.8.

El botón Apagar PC permite apagar la estación digitada lo que se muestra en la **(Figura A.9.)**.

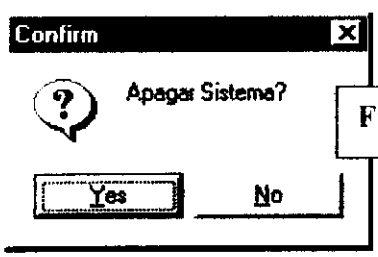


Figura A.9.

El botón Enviar permite enviar el mensaje a la estación digitada.

El botón Limpiar para poner en blanco el cuadro de texto.

El botón Salir para cerrar la pantalla.

La opción **ESTADÍSTICAS** permite visualizar la pantalla que representa la **(Figura A.10.)** con dos opciones:

- Estaciones.
- Aplicaciones.

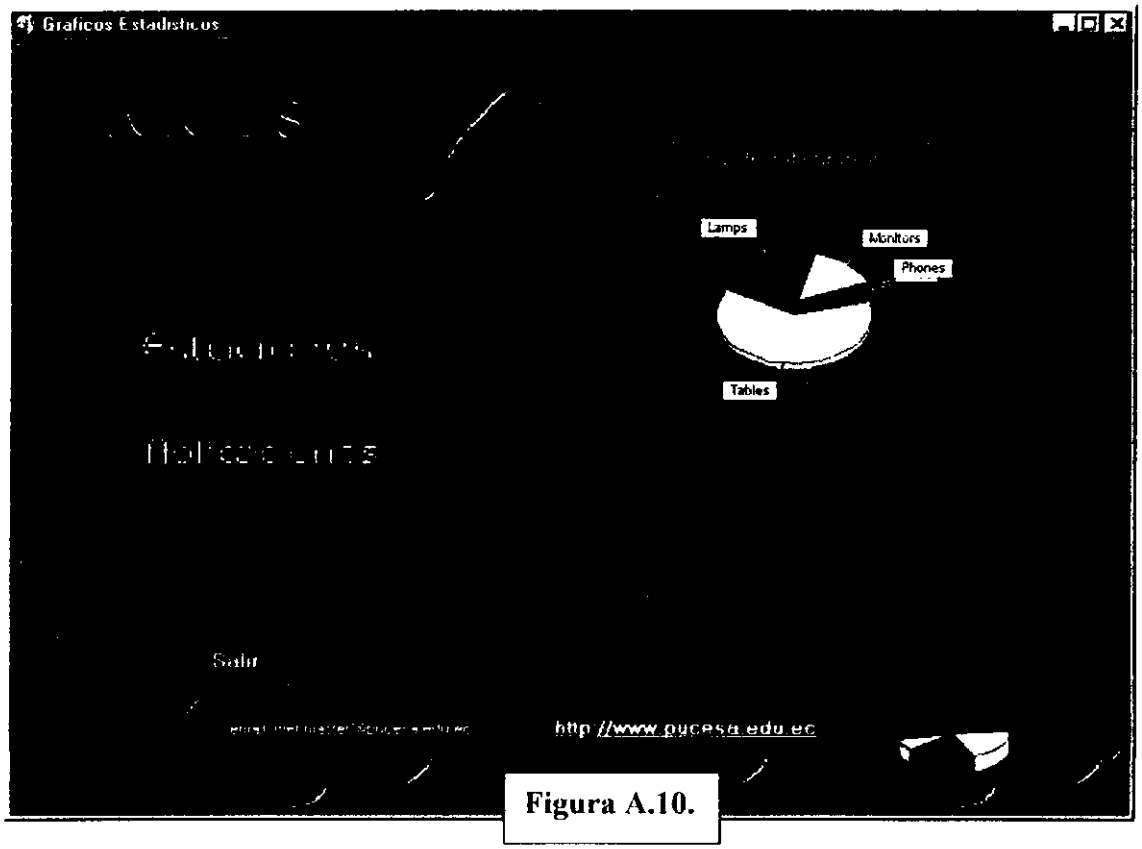


Figura A.10.

El botón **ESTACIONES** permite visualizar la pantalla que representa la **(Figura A.11.)** en la cual muestra un gráfico con un porcentaje de utilización de dichas estaciones.

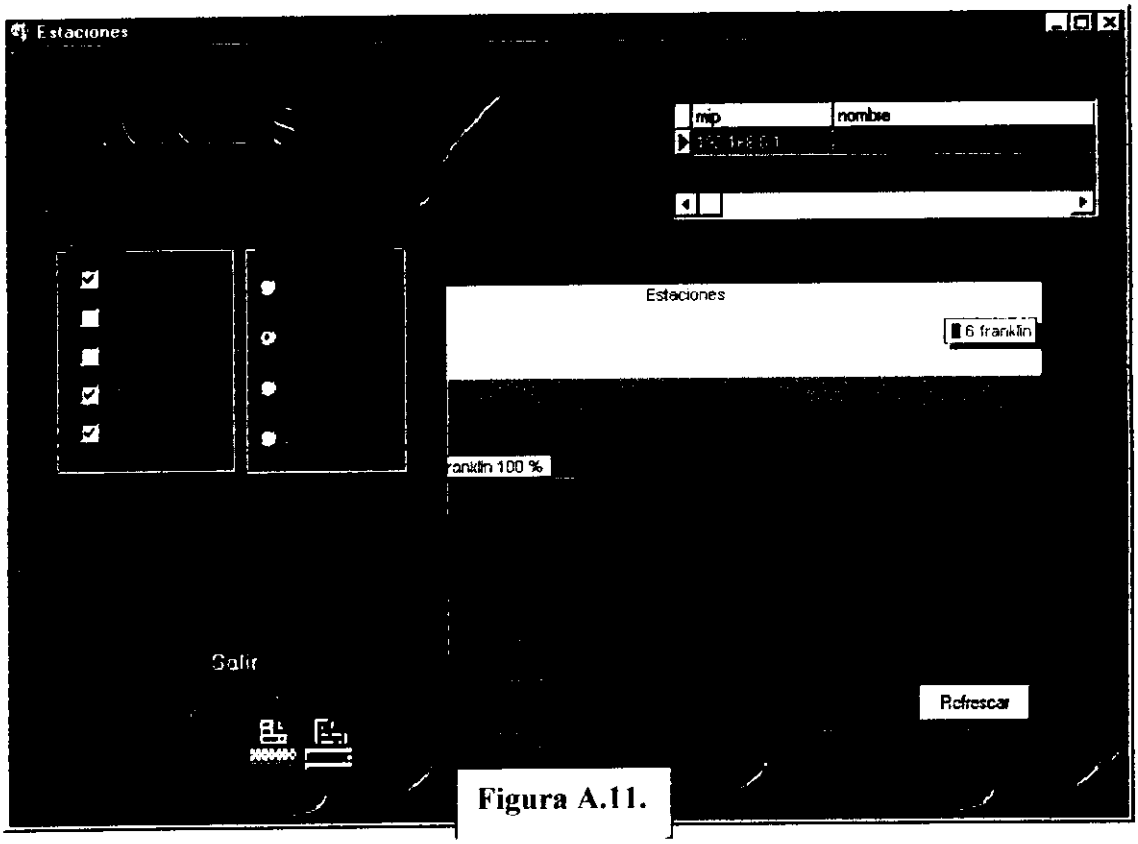


Figura A.11.

La opción **EFECTOS** permite visualizar el gráfico en diferentes facetas:

- ✓ 3D.
- ✓ Círculo.
- ✓ Use prototipo.
- ✓ Etiquetas.
- ✓ Gradiente.

La opción **POSICIÓN DEL LETRERO** permite visualizar el letrero en diferentes posiciones:

- Left (Izquierda).
- Right (Derecha).
- Top (Arriba).
- Down (Abajo).

El botón Refrescar para actualizar el gráfico.

El botón con imagen de una impresora para imprimir el gráfico.

El botón Salir para cerrar la pantalla.

El botón **APLICACIONES** permite visualizar la pantalla que representa la **(Figura A.12.)** en la cual muestra un gráfico generalizado del porcentaje de uso de aplicaciones.

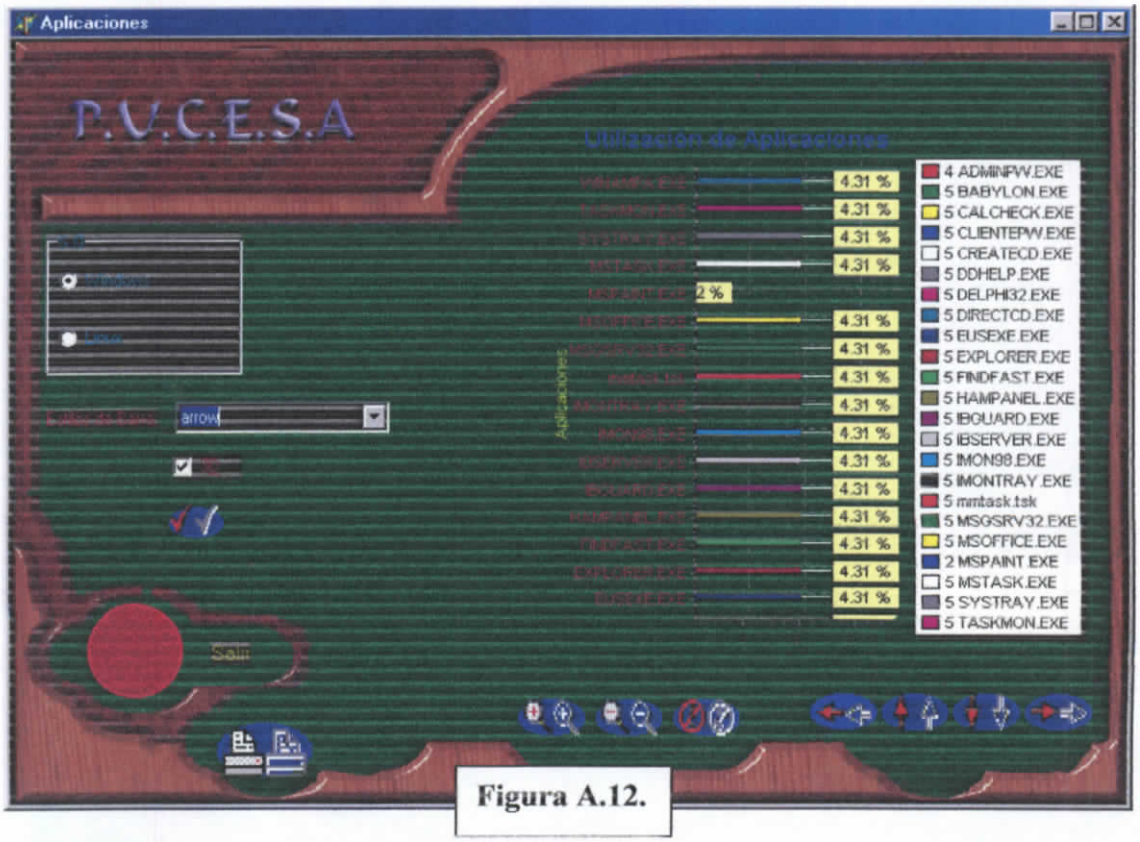


Figura A.12.

La opción **S.O** (Sistema operativo):

- Windows.
- Linux.

La opción **WINDOWS** permite visualizar el porcentaje de utilización de aplicaciones de Windows.

La opción **LINUX** permite visualizar el porcentaje de utilización de aplicaciones de Linux.

La opción **EFECTOS DE BARRA** permite seleccionar el tipo de barra para el gráfico:

- Rectangle (Rectángulo).
- Pyramid (Pirámide).
- InvPyramid (Pirámide Invertida).
- Cylinder (Cilindro).
- Ellipse (Elipse).
- Arrow (Flecha).

En la opción **3D** que permite visualizar el gráfico en tercera dimensión.

El botón con la imagen de visto con un click para mostrar el gráfico y con doble click para ocultarlo.

El botón con la imagen de una impresora para imprimir el gráfico.

- Ampliar.
- Disminuir.
- Deshacer.
- Izquierda.
- Arriba.
- Abajo.
- Derecha.

La opción **REPORTES** permite visualizar la pantalla que representa la **(Figura A.13.)** en la cual debe escoger el tipo de reporte:

- Reportes por Parámetros.
- Reportes Generales.

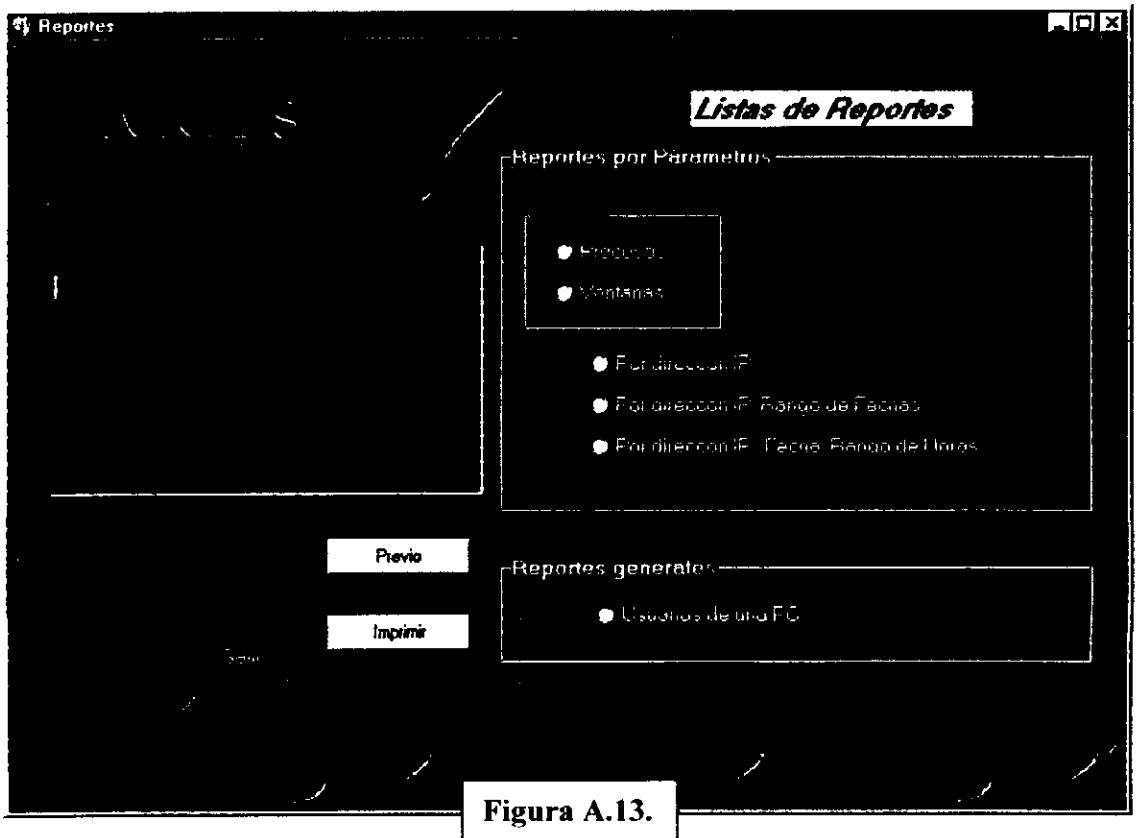


Figura A.13.

La opción **REPORTES POR PARÁMETROS** permite obtener reportes de procesos o ventanas ingresando los parámetros respectivos:

- Por Dirección IP.
- Por Dirección IP, Rango de Fachas.
- Por Dirección IP, Fecha, Rango de Horas.

La opción **POR DIRECCIÓN IP** le permite visualizar la pantalla como la que representa la (**Figura A.14.**) seleccione la dirección IP y luego OK.

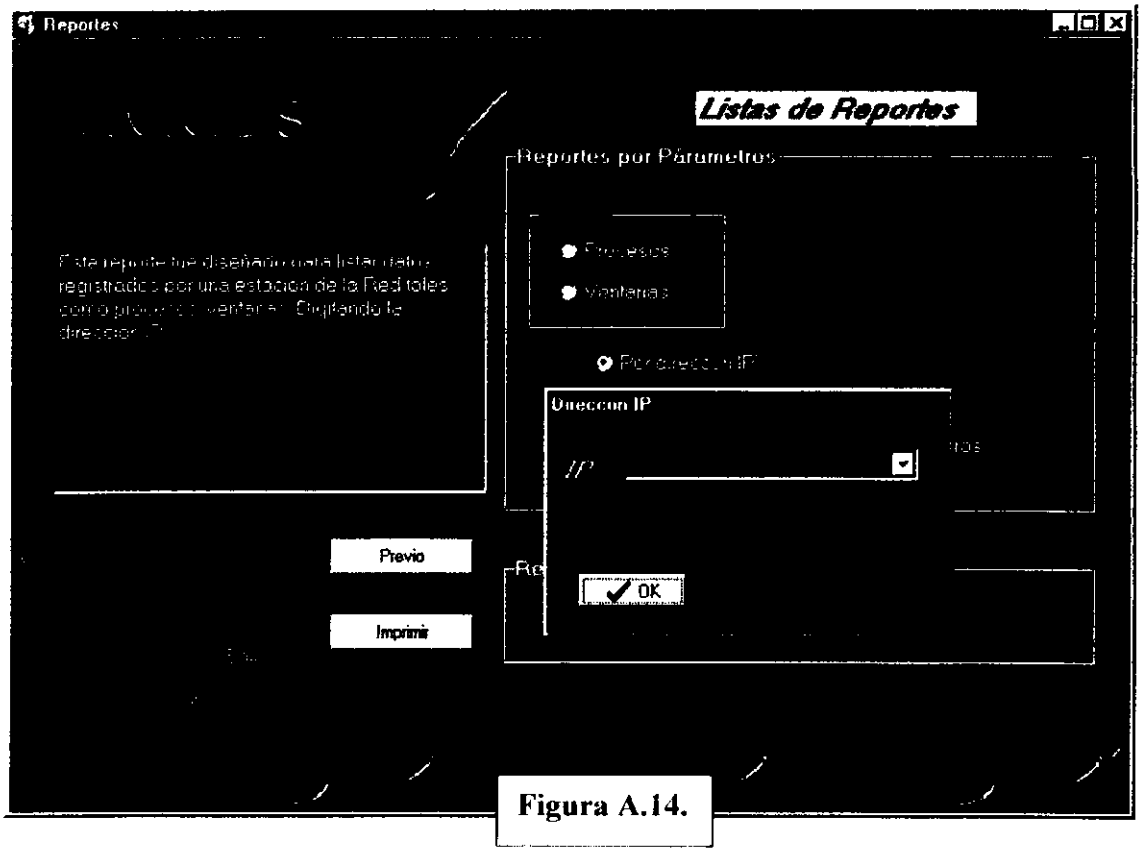


Figura A.14.

Luego haga click en el botón Previo o Imprimir de la pantalla que representa la (**Figura A.13.**). Este reporte le proporciona al administrador todos los procesos o ventanas filtrando por la dirección IP.

La opción **POR DIRECCIÓN IP, RANGO DE FECHAS** le permite visualizar la pantalla como la que representa la (**Figura A.15.**) seleccione la dirección IP e ingrese la fecha inicial y fecha final luego OK.

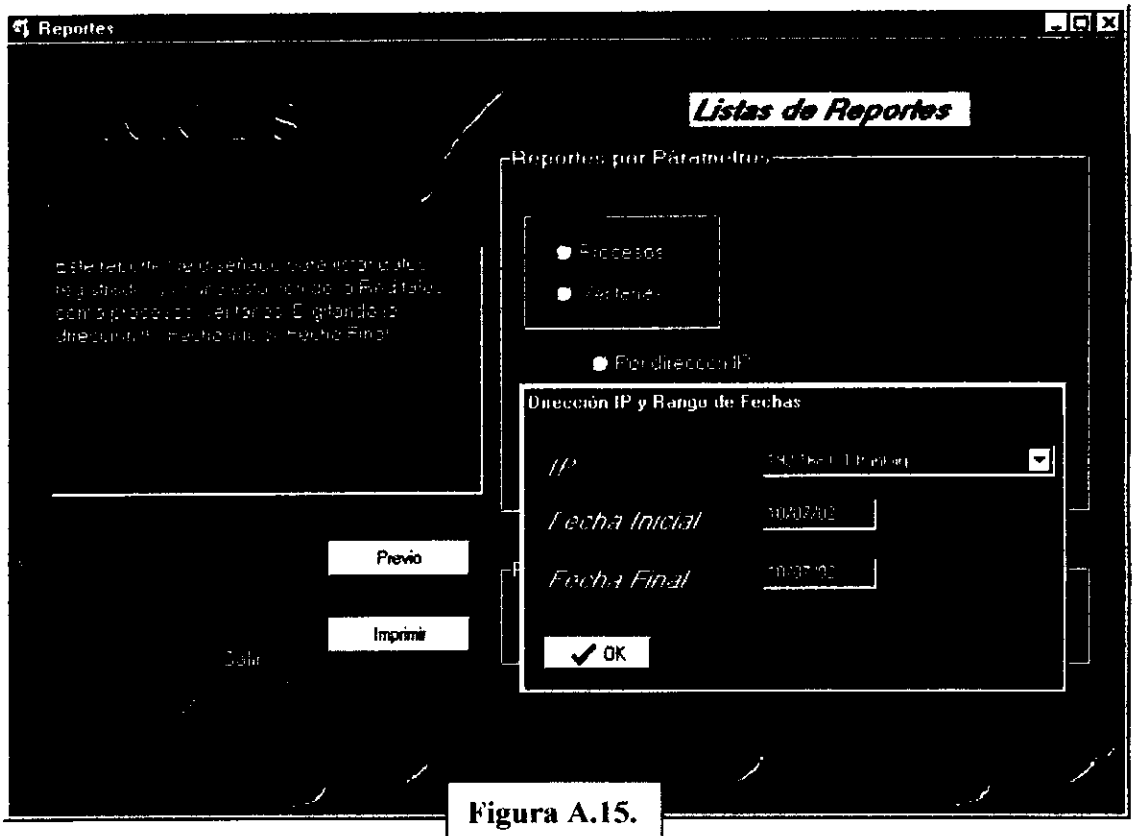
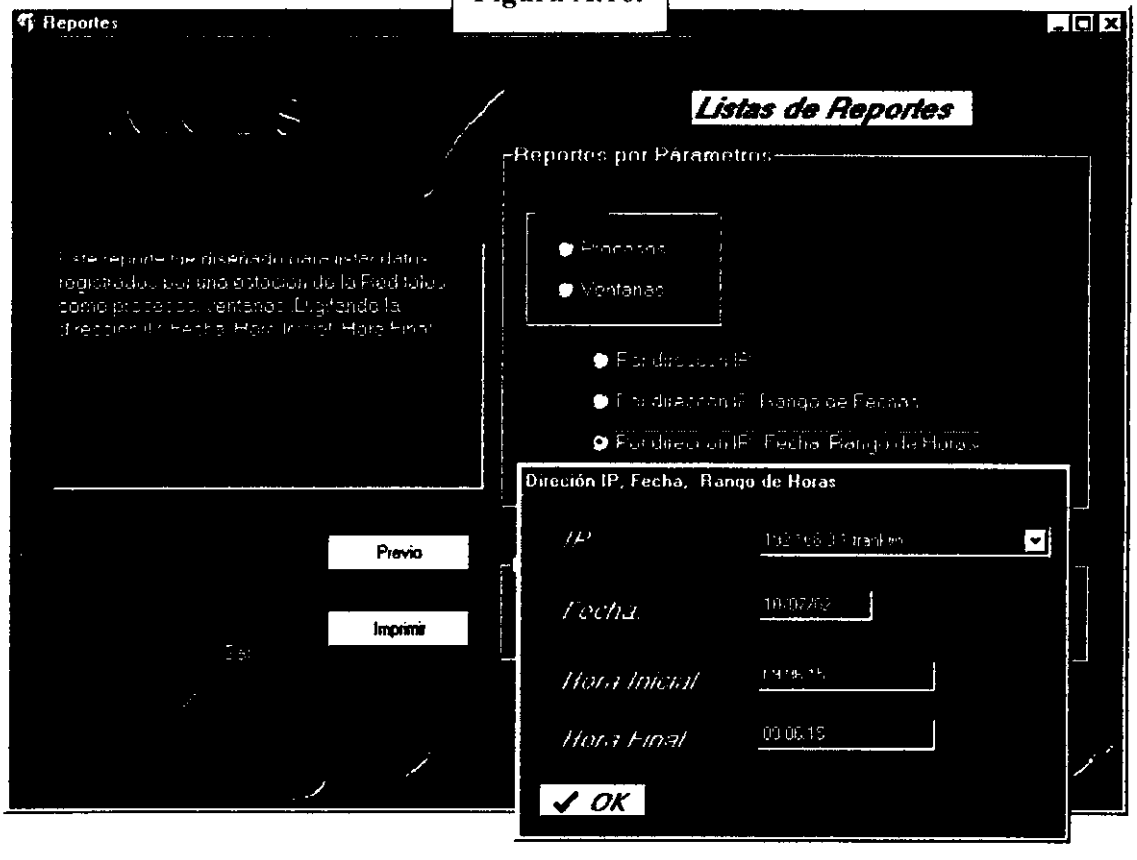


Figura A.15.

Haga click en el botón Previo o Imprimir de la pantalla que representa la (**Figura A.13.**). Este reporte le proporciona al administrador todos los procesos o ventanas filtrando por la dirección IP y rango de fechas.

La opción **POR DIRECCIÓN IP, FECHA, RANGO DE HORAS** le permite visualizar la pantalla como la que representa la (**Figura A.16.**) seleccione la dirección IP, ingrese la fecha, ingrese la hora inicial y hora final luego OK.

Figura A.16.



Haga click en el botón Previo o Imprimir de la pantalla que representa la (**Figura A.13.**). Este reporte le proporciona al administrador todos los procesos o ventanas filtrando por la dirección IP, fecha, Rango de Horas.

La opción **USUARIOS DE UNA PC** permite obtener un reporte de los usuarios de una estación de trabajo, no necesita ingresar parámetros.

La opción **COLORES** permite visualizar la pantalla que representa la **(Figura A.17.)** en la cual se puede realizar cambios de colores tanto del fondo como de letras.

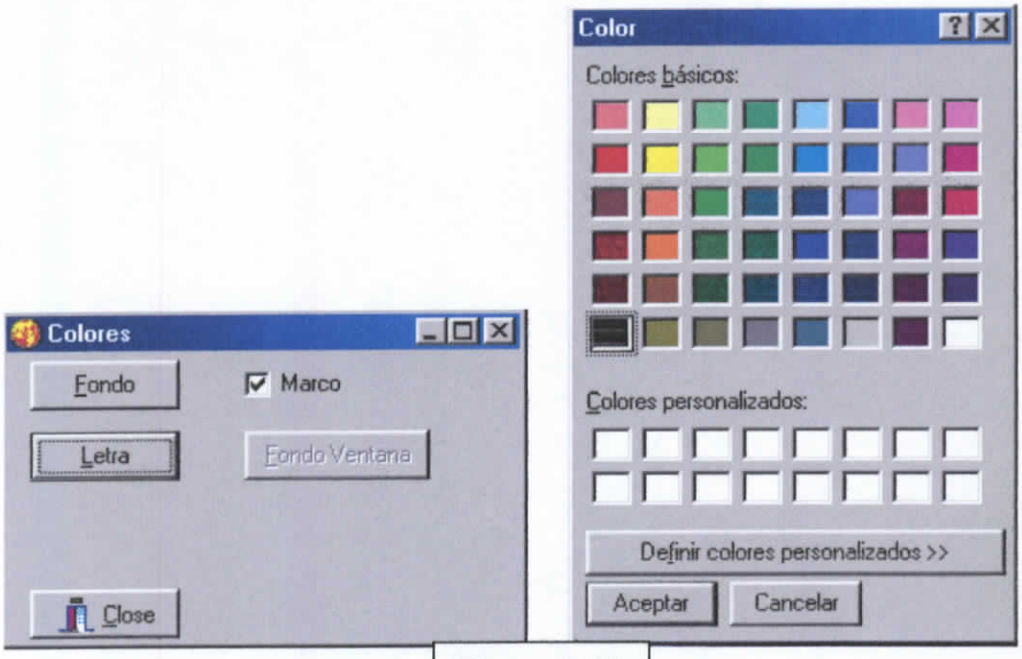


Figura A.17.

La casilla de verificación Marco permite visualizar las ventanas con o sin marco. Sin marco se activará el botón Fondo Ventana con el cual se le dará la opción de cambiar el color al fondo de todas las ventanas de la aplicación.

El menú **AYUDA** que contiene:

- Ayuda del Software.
- Acerca.

La opción **AYUDA DEL SOFTWARE** que permite visualizar la pantalla que representa la **(Figura A.18.)**.

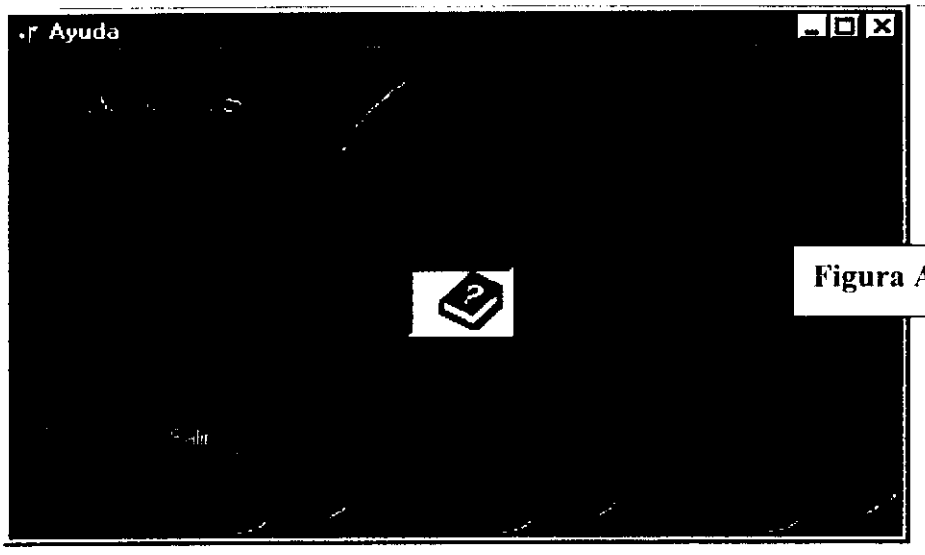


Figura A.18.

En la imagen que representa un libro hacer doble clic para mostrar la ayuda.

El botón Salir para cerrar la pantalla.

La opción **ACERCA** que permite visualizar la pantalla que representa la **(Figura A.19.)**.

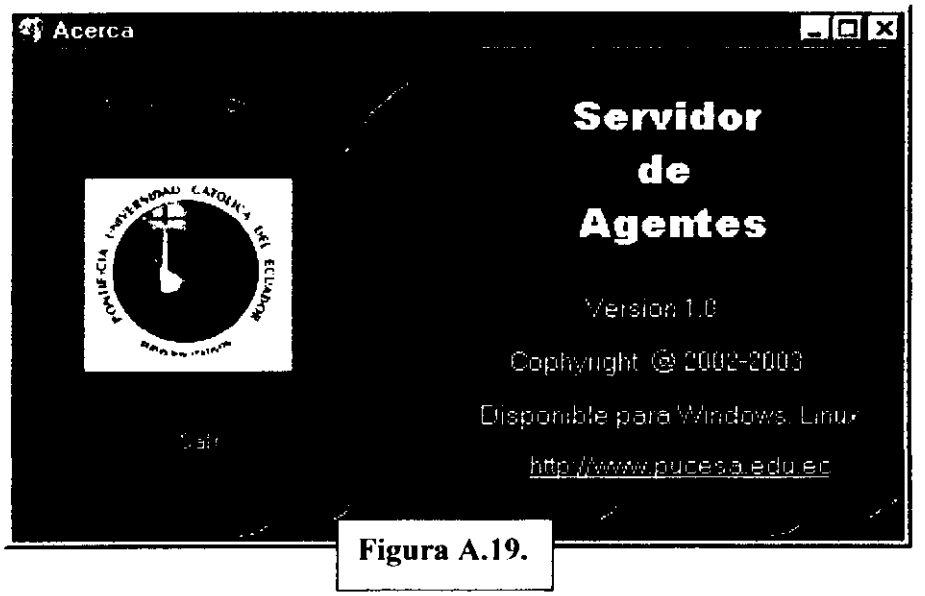


Figura A.19.

El botón Salir para cerrar la pantalla.

¿CÓMO UTILIZAR EL AGENTE SOFTWARE ® ING.2002?

Tenemos dos aplicaciones que realizan la misma función:

- AgentePW.
- AgentePL.

Estas aplicaciones se ejecutan automáticamente al iniciar la máquina para lo cual el programa Servidor de Agentes debe estar corriendo.

Plataforma Windows

Como el programa debe iniciarse automáticamente al iniciar la máquina se creo la aplicación llamada EnRegistroWP.exe la cual debe ejecutarse una vez instalado el programa agente el cual envía al registro de Windows la clave. Esta aplicación viene con el programa agente que se encuentra en el directorio C:\Windows\Agente\Herramientas, se puede apreciar en la **(Figura A.20.)**.

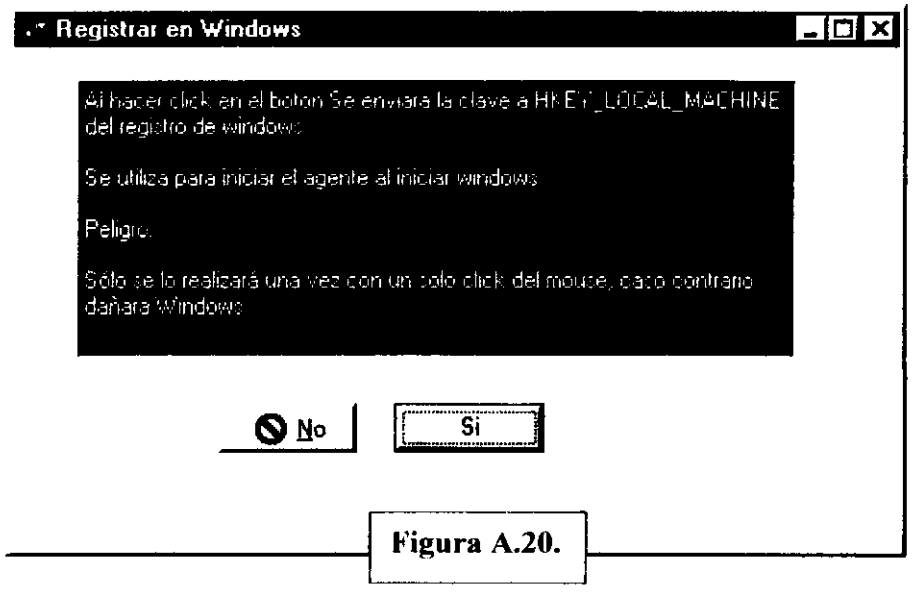


Figura A.20.

La aplicación **AgentePW.exe** corre bajo la plataforma Windows como se ilustra en la **(Figura A.21.)** el icono que representa a un ojo, se encuentra en la bandeja de entrada.



Figura A.21.

Hacer doble click en el icono indicado para desactivar el programa agente, se visualizará una pantalla coma la que ilustra la **(Figura A.22.)**.

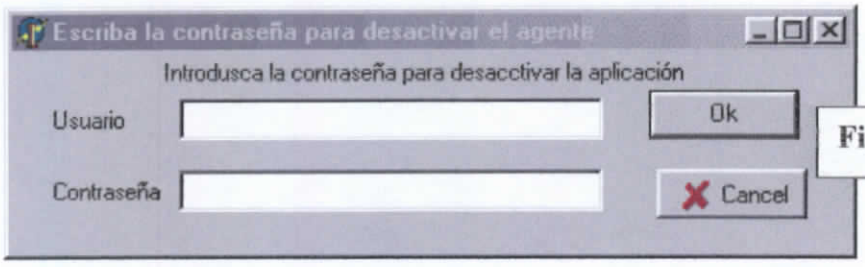


Figura A.22.

Si el Usuario y Contraseña son correctos se cierra la aplicación con un click en el botón OK.

El botón Cancel para cancelar la acción.

Plataforma Linux

Como el agente debe iniciar automáticamente al iniciar la máquina debe agregar esta línea al archivo Autorun que se encuentra en el directorio /inicio/

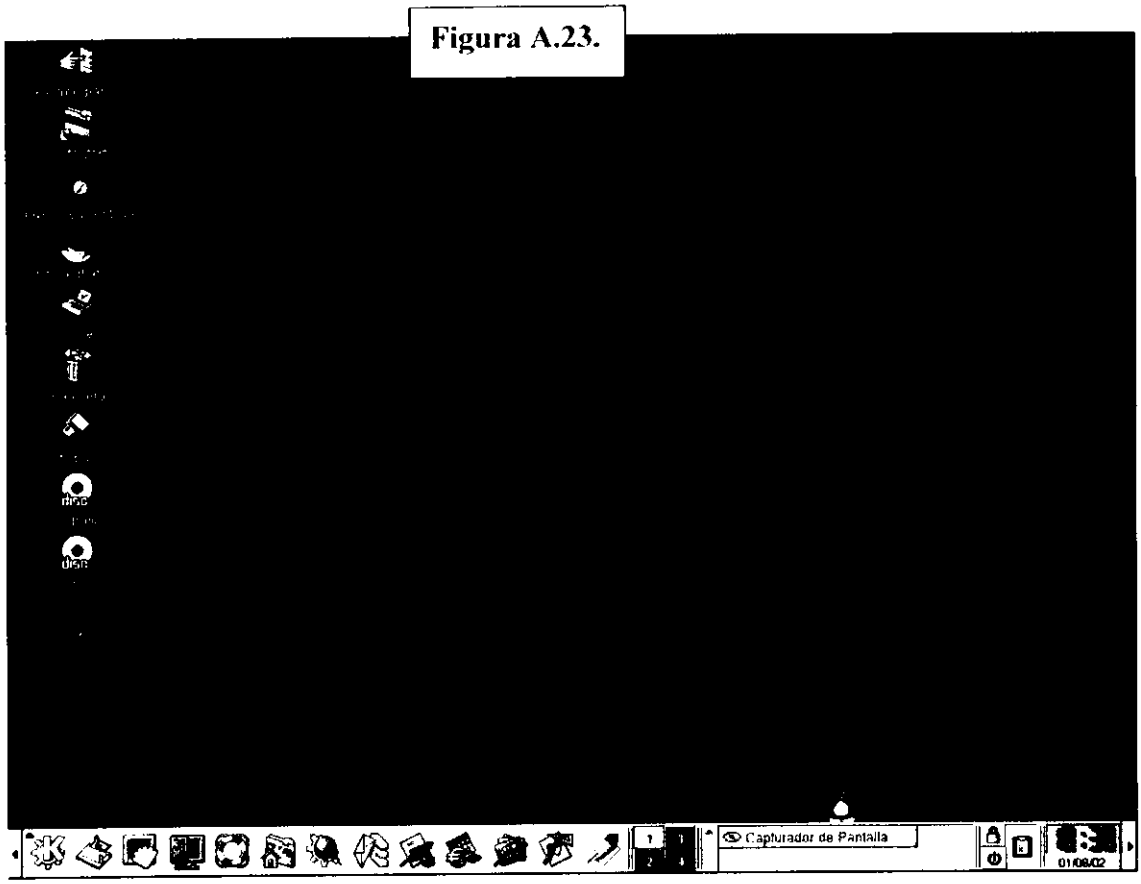
Exec /<directorio del programa agente>/ejecutable

En nuestro caso:

Exec /home/vicente/dirtrabajo/Agente/AgentePL

La aplicación **AgentePI.bin** corre bajo la plataforma Linux como se ilustra en la **(Figura A.23.)** el icono que representa a un ojo, se encuentra en el escritorio de Linux.

Figura A.23.



Hacer click en el icono indicado para desactivar el programa agente, se visualizará una pantalla coma la que ilustra la **(Figura A.22.)**.

ANEXO B

MANUAL TÉCNICO

Las aplicaciones desarrolladas en la presente disertación de grado requieren de la instalación de Lenguajes, Bases de Datos, Utilidades, Plataformas y Software para el desarrollo:

- Windows.
  - Delphi.- Lenguaje de programación.
  - MYSQL.- Servidor de Base de Datos.
  - MySQL-Front.- Interfaz gráfica para MYSQL.
- Linux .
  - Kylix.- Lenguaje de programación.
  - MYSQL.- Servidor de Base de Datos.
  - Webmin.- Interfaz gráfica para MYSQL.

WINDOWS

BORLAND DELPHI ENTERPRISE VERSIÓN 6.0

Lenguaje de programación para desarrollo de las aplicaciones bajo la plataforma Windows.

Requisitos para la instalación:

- Microsoft Windows 98, 2000, Me, NT 4.0 with Service Pack 5 o superior.
- Intel Pentium 166 MHz o superior (P2 400 MHz recomendado).
- 64 MB RAM (128 MB recomendado).
- 115 MB espacio en disco duro (Instalación mínima).
- 350 MB espacio en disco duro (Instalación completa).
- CD-ROM drive.
- VGA o superior en resolución del monitor.
- Mouse.
- Teclado.

Antes de la Instalación

Usted debería siempre proteger sus datos con respaldos antes de abrir un proyecto que fue creado con una versión anterior de este producto, para la seguridad de todos los archivos incluidos en el proyecto.

Respaldando las configuraciones de BDE (Motor de la Base de Datos de Borland)

Este producto incluye una nueva versión del BDE. Al menos que usted especifique otro directorio de instalación al de por defecto durante la instalación, Delphi sobre-escribirá su BDE existente de 32-bit e instalaciones de conexión SQL. Se recomienda que usted realice un respaldo de todos los archivos de configuración del BDE antes del proceso de instalación.

Directorio de Instalación

El directorio para la instalación por defecto es C:\Archivos de programa\Borland\Delphi6.

Advertencia:

No instale Delphi 6.0 en un directorio que contenga una versión anterior. Instale siempre una nueva versión de Delphi en un directorio nuevo. Diferentes versiones de Delphi pueden coexistir en el mismo sistema operativo, pero cada versión debe ser instalada en su propio directorio.

Si usted ya instaló este producto anteriormente

Se puede instalar en la misma máquina (sin embargo en diferente ubicación) como otra versión del producto. Si usted quiere instalarlo en el mismo directorio como una versión existente, desinstale la primera versión en cualquiera de los casos debe sacar un respaldo del archivo IDAPI.CFG (si este existe) y algún otro dato importante, incluyendo proyectos que se intenta usar con la nueva versión.

Nota: Para comenzar a trabajar con Delphi 6.0 y trabajar con el componente TSQLConnection se debe tomar en cuenta la librería VendorLib que es **libmySQL.dll** esta librería debe ser versión 3.22.34 que es la más estable para trabajar con máquinas remotas.

El archivo de conexión **dbxconnections.ini** es el archivo en el que se guarda el nombre de conexión, el driver, el nombre de la base de datos, dirección IP o el nombre de host en donde reside la base de datos, password, usuario.

Este archivo está localizado:

C:\Archivos de programa\Archivos comunes\Borland Shared\DBExpress\

La conexión que hemos creado para nuestro objeto de estudio es:

[AgeConnection]

DriverName=MYSQL

BlobSize=-1

Database=agentes

ErrorResourceFile=

HostName=192.168.0.2

LocaleCode=0000

Password=

User_Name=root

El administrador deberá cambiar el HostName de acuerdo a su dirección IP o nombre de la PC en donde se encuentre el Servidor de Base de Datos MYSQL y la aplicación Servidor de Agentes.

El archivo de conexión **TcpHost.ini** es el archivo en el que se guarda la dirección IP del host en donde reside la aplicación Servidor de agentes el cual es un servidor de fecha y hora para los programas agentes.

Este archivo está localizado en el mismo directorio donde reside el programa agente en este caso en el directorio C:\Windows\Agente\

[Server]

TCPHOST=192.168.0.2

El administrador debe cambiar esta dirección IP de acuerdo donde vaya a correr la aplicación Servidor de Agentes.

MYSQL

Servidor de Base de Datos en la plataforma Windows.

Instalación de MYSQL en Windows 95 o 98:

Por defecto, MYSQL-Windows se configura para ser instalado en C:\mysql. Si quieres instalar MYSQL en otra parte debes arrancar el servidor en otra parte. Por ejemplo, si has movido la distribución de MYSQL a D:\programs\mysql, debes empezar mysqld con: D:\programs\mysql\bin\mysqld –basedir.

Arrancar MYSQL en Windows 95 o Windows 98

MYSQL usa TCP/IP para conectar un cliente a un servidor. (Esto permitirá a cualquier máquina en tu red conectar a tu servidor MYSQL). Debido a esto, debes instalar TCP/IP en tu máquina antes de empezar MYSQL. Puedes encontrar TCP/IP en tu CD-ROM de Windows.

Hay 2 servidores MYSQL diferentes que puedes usar:

mysqld	Compilado con depuración full y verificación de asignación de memoria automático.
mysqld-pt	Perfeccionado para un procesador Pentium.

Los dos anteriores deben trabajar en cualquier procesador de Intel >= i386.

Para arrancar el servidor del mysqld, debes iniciar una ventana de MS-DOS y tipear:

C:\mysql\bin\mysqld

Para apagar el servidor de MYSQL ejecuta:

C:\mysql\bin\mysqladmin -u root shutdown

Si el mysqld no arranca, verifica si el archivo **\mysql\mysql.err** contiene cualquier razón para esto o no.

Correr MYSQL en Windows

Puedes probar si MYSQL está trabajando o no, ejecutando las siguientes órdenes:

C:\mysql\bin\mysqlshow (Visualiza información sobre base de datos que hay en el Sistema).

C:\mysql\bin\mysqlshow -u root mysql (Visualiza información sobre las tablas de la base de datos mysql).

C:\mysql\bin\mysqladmin versión status proc

C:\mysql\bin\mysql test (Se nos presentará el prompt de la shell, con lo cual podemos enviar comandos al servidor).

Si quieres usar mysqlc.exe, debes copiar **C:\mysql\lib\cygwinb19.dll** a la carpeta de C:\windows\system.

MYSQL-FRONT

Interface Gráfica para MYSQL. Este programa nos ofrece una forma de comunicarnos de manera más amigable con el servidor SQL.

MySQL-Front es una interface fácil de usar para los que trabajan en la red con Bases de Datos MSQL.

Requiere la librería **libmySQL.dll** del Servidor MYSQL como "API".

Nota: Esta librería viene con MySQL-Front, no es la misma que se utiliza para los componentes TSQLConnection para conectarse un cliente con el Servidor de Base de Datos MYSQL.

Esta librería debe residir en el mismo directorio de ejecutable **mysqlfront.exe**.

Instalación:

MySQL-Front no viene todavía con un instalador, descomprima todos los archivos en un directorio y ejecute **mysqlfront.exe**, ingrese el nombre de la máquina o la dirección IP, login, password respectivos y listo.

Con MySQL-Front se puede:

- Crear, Borrar bases de datos.
- Crear, Borrar tablas.
- Editar, Adicionar, Borrar campos.
- Editar, Insertar, Borrar registros.
- Una lista de variables.
- Ver y matar otros procesos de usuarios.
- Ejecutar largas sentencias SQL.
- Ver propiedades de las tablas tales como tipo, comentario, longitud de la clave.
- Exportar estructuras y datos de las tablas en SQL-scripts o otras Bases de datos.
- Replicar bases de datos entre dos hosts ("Exportar tablas...").
- Grabar datos para archivos CSV (Ideal para trabajar con MS-Excel) o tablas HTML.
- Copiar datos CSV, tablas HTML al portapapeles.
- Copiar tablas con nuevo nombre.
- Cambiar tipos de tablas a ISAM, MYISAM, MERGE, HEAP o tipos indefinidos.
- Adicionar usuarios.
- Copiar datos de Host/Logs/Privilegios/Tablas a un medio de almacenaje permanente.

- Escriba SQL-queries con sintaxis relevante.
- Importar datos desde ODBC-datasources.
- Hace diagnóstico de tablas (Chequear, Optimizar, Reparar, Analizar).

Requerimientos:

- **libmySQL.dll** (Librería cliente para el servidor MYSQL).
- **msvcrt.dll** (Microsoft (R) C Runtime Library).
- Protocolo TCP/IP.
- ODBC.

LINUX

KYLIX 1.0

Lenguaje de programación para desarrollo de las aplicaciones bajo la plataforma Linux.

No use un administrador de paquetes, tales como rpm, kpackage, o gnorpm, para instalar o desinstalar Kylix. Siempre use el programa instalador y desinstalador suministrado por Borland.

Requerimientos para Kylix:

Al menos todos los usuarios necesitan actualizar las librerías **glibc** antes de instalar Kylix.

Previo a la instalación debe instalar el siguiente software:

- Linux kernel versión 2.2 o superior.
- **libgtk.so** versión 1.2 o superior (requerido sólo para la instalación gráfica).
- **libjpeg** versión 6.2 (**libjpeg.so.62**) o superior.
- Una terminal server X11R6-compatible tal comoXFree86.

Kylix debe ser instalado en un archivo de sistema que soporte enlaces simbólicos tales como **ext2** de Linux. No instalar Kylix en FAT, FAT32, SMB (incluyendo Samba), o archivos de sistema Novell NFS, éstos archivos de sistema deberian trabajar pero no han sido probados.

Kylix ha sido probado con varios manejadores de Linux y escritorios incluyendo GNOME y KDE.

Para ejecutar las aplicaciones hechas en Kylix desde el entorno, usted debe incluir esta linea en el archivo **.bashcr** que está situado en el directorio /etc.

Source <directorio>/bin/kylixpath >/dev/null

Donde <directorio> es el camino completo del directorio donde está instalado Kylix, en nuestro caso de estudio será: *source opt kylix bin kylixpath opt kylix dev null.*

Desde la línea de comandos sin Kylix corriendo escriba el nombre de programa y se ejecuta.

/home/vicente/dirtrabajo/Agente/$ ./programa

Nota: Para comenzar a trabajar con Kylix 1.0 con el componente TSQLConnection se debe tomar en cuenta la librería VendorLib que es **libmysqlclient.so.6.0.0** para conectarse con la Base de Datos en este caso la versión más estable MySQL 3.22, al instalar se localiza en *usr lib* .

El archivo de conexión **dbxconnections.ini** es el archivo en el que se guarda el nombre de conexión, el driver, el nombre de la base de datos, el nombre o dirección IP de host en donde reside la base de datos, password, usuario.

La conexión que hemos creado para nuestro objeto de estudio es:

[AgeConnection]

DriverName=MYSQL

BlobSize=-1

Database=agentes

ErrorResourceFile=

HostName=192.168.0.1

LocaleCode=0000

Password=

User_Name=root

El administrador deberá cambiar el HostName de acuerdo a su dirección IP o nombre de la PC en donde se encuentre el Servidor de Base de Datos MYSQL y la aplicación Servidor de Agentes.

El archivo de conexión **TcpHost.ini** es el archivo en el que se guarda la dirección IP del host en donde reside la aplicación Servidor de agentes el cual es un servidor de fecha y hora para los programas agentes.

Este archivo está localizado en el mismo directorio donde reside el programa agente en este caso:

/home/vicente/dirtrabajo/Agente/

TCPHOST–192.168.0.1

El administrador debe cambiar ésta dirección IP de acuerdo donde vaya a correr la aplicación Servidor de Agentes.

MYSQL

Servidor de Base de Datos en la plataforma Linux.

Instalación de MYSQL en Linux:

Se recomienda instalar MYSQL en Linux usando un archivo RPM.

Los archivos RPMs de MYQSL son actualmente hechos en la versión 6.2 de RedHat, también funcionan con versiones superiores.

Los archivos RPM son:

- **MySQL-VERSION.i386.rpm**

  El servidor MYSQL es el más importante si desea conectarse desde otra máquina.

- **MySQL-client-VERSION.i386.rpm**

  El programa cliente MYSQL estándar para conectarse con el servidor.

- **MySQL-bench-VERSION.i386.rpm**

  Pruebas y conjunto de procedimientos para evaluar el rendimiento de un ordenador requerido por Perl, y msql-mysql-modules RPMs.

- **MySQL-devel-VERSION.i386.rpm**

  Librerías y archivos necesarios si se quiere compilar otros clientes MYSQL tales como módulos perl.

- **MySQL-VERSION.src.rpm**

  Este paquete contiene el código fuente de todos los paquetes. Este también puede ser usado para construir RPMs para otras arquitecturas (por ejemplo Alpha o SPARC).

Para ver todos los archivos que contiene el paquete RPM, ejecute:

shell> rpm -qpl MySQL-VERSION.i386.rpm

Para ejecutar una instalación mínima, ejecute:

shell> rpm -i MySQL-VERSION.i386.rpm MySQL-client-VERSION.i386.rpm

Para instalar el paquete cliente, ejecute:

shell> rpm -i MySQL-client-VERSION.i386.rpm

El RPM sitúa los datos en `/var/lib/mysql', también crea las entradas apropiadas en `/etc/rc.d/' para iniciar automáticamente al tiempo de iniciar el ordenador.

Después de instalar los archivos RPM, el demonio `mysqld' debería estar corriendo y disponible para usarlo.

WEBMIN

Interface Gráfica para MYSQL de Linux. Webmin es una interface basada en web para la administración del sistema Unix.

Utilizando cualquier browser puede:

- Crear cuentas de usuario para utilizar webmin.
- Instalar paquetes.
- Configurar el Apache.
- Proporciona servicios de Internet.
- Manejo, configuración de servidores de Bases de Datos.
- Configurar Samba.
- Interacción con procesos.
- Configurar el sistema.

Instalación del Webmin:

Para instalar Webmin simplemente ejecute el **setup.sh** desde la shell de Linux.

Para iniciar Webmin en la barra de direcciones de cualquier explorador digite lo siguiente:

http://<nombre o ip de la máquina>//:10000

Por ejemplo:

http://franklin:10000

El número 10000 represente el puerto de conexión del Webmin.

A continuación ingresamos el login y el password, en este caso:

Login – root

Password – crystal (el password de usuario root)

Una vez que el acceso sea permitido ingrese a Servidores donde se muestra MySQL Server. En el Servidor de Base de Datos MySQL se puede realizar:

- Crear, Borrar bases de datos.
- Crear, Borrar tablas.
- Editar, Adicionar, Borrar campos.
- Editar, Insertar, Borrar registros.
- Una lista de variables.
- Ingresar valores.

Y en opciones globales se puede dar:

- Permisos de Usuario.
- Permisos de Base de Datos.
- Permisos de Máquina, etc.

GLOSARIO

A

ANSI (American National Standards Institute) instituto de estándares nacional americano.

AOO Análisis Orientado a Objetos.

API (Application Program Interface) interfaz de programa de aplicación. Lenguaje y formato utilizados para un programa para comunicarse con otro programa.

ARP (Address Resolution Protocol) protocolo de resolución de direcciones.

ARPANET (Advanced Research Projects Agency NETwork) red avanzada de agencias para proyectos de investigación.

B

BNC un conector utilizado comúnmente para cable coaxial.

BOOTP (Bootstrap Protocol) protocolo que contiene una rutina <<bootstrap>> en el chip ROM que automáticamente se ejecuta cuando el computador se enciende o se reinicia la actividad.

Bridge puente. Un dispositivo que conecta dos redes de igual tipo.

C

CASE (Computer Aided Software Engineering o Computer Aided Systems Engineering) ingeniería de software asistida por computadora o ingeniería de sistemas asistida por computadora.

CCITT (Consultative Committee for International Telephony and Telegraphy) comité consultativo para telefonía y telegrafía internacionales. Una organización internacional de normas de comunicaciones.

CCR centro de control de red.

CLX (Component Library for Cross Platform) librería de componentes para multiplaforma.

Comma delimited delimitado por comas. Una disposición de registros en la cual los campos están separados por comas y cada dato en caracteres está habitualmente encerrado entre comillas.

Concatenate concatenar. Enlazar varias estructuras entre sí. Por ejemplo, el concatenado de archivos añde un archivo a otro.

CSMA/CD (Carrier Sense Multiple Access/Collision Detection) sensor de portadora de accesos múltiples/detección de colisiones. Un método de acceso en las comunicaciones de banda base que emplea una técnica de detección de colisiones.

CSV (Comma Separated Value) valor separado por comas. Igual a comma delimited.

D

DCC (Delphi Command-line Compiler) compilador de líneas de comandos de Delphi.

DECnet Una arquitectura de comunicaciones y serie de productos relacionados de Digital Equipment Corporation. DECnet apoya tanto a las redes de área de estilo Ethernet como también las redes de área amplia.

DLL (dynamic link library) biblioteca de enlaces dinámicos. Un conjunto de rutinas de programa que están disponibles para las aplicaciones en tiempo de ejecución.

DNS (Domain Naming System) sistema para dar nombres de dominios. Sistema de direccionamiento de correo electrónico utilizado en redes como Internet.

DOO Diseño Orientado a Objetos.

E

ECD computadoras de conmutación.

EMI (ElectroMagnetic Interference) interferencia magnética. Ondas electromagnéticas que emiten desde un dispositivo eléctrico.

ETD terminales/computadoras de usuario.

F

FAT (File Allocation Table) tabla de asignación (distribución) de archivos. La parte del sistema de archivos del DOS y OS/2 que lleva la cuenta de dónde están almacenados los datos en un disco.

FTP (File Transfer Protocol) protocolo de transferencia de archivos. Herramienta software que permite transmitir archivos a través de Internet entre una máquina local y una remota.

G

Gateway pasarela, puerta de acceso. Una computadora que conecta dos tipos diferentes de redes de comunicaciones. Realiza la conversión de protocolos de una red a otra.

GNU (Licencia Pública General).

Gopher sistema de organización jerárquica de información e Internet. Permite acceder o menú carpetas donde están incluidos todos los documentos de la red que se pueden visualizar e imprimir.

GUI (Graphical User Interface) interfaz gráfica de usuario basada en gráficos que incorpora iconos, menús, enrollables y un ratón.

H

Heap memoria libre disponible en el momento para cargar y ejecutar programas.

Host computadora central que es utilizada por más de un usuario y que contiene una gran cantidad de datos (programas, archivos, bases de datos, etc.). Soporta el trabajo de varios usuarios al mismo tiempo.

Hub's concentradores. Dispositivos que enlazan grupos de computadoras en una red de área local y permiten establecer turnos cuando las computadoras se comunican entre sí.

I

IBM (International Business Machines Corporation) corporación internacional de máquinas para negocios. La compañía informática más grande del mundo.

ICMP (Internet Control Message Protocol) protocolo de control de mensajes en Internet.

IEEE (Institute of Electrical and Electronic Engineers) instituto de ingenieros de electricidad y electrónica. Una organización de asociados que incluye ingenieros, científicos y estudiantes en electrónica y disciplinas afines.

IGMP (Internet Group Management Protocol) protocolo de grupo de manejo de Internet.

Internet (Interred). Red extensa constituida por una cantidad de redes menores.

IPC (InterProcess Communication) comunicación interprocesos. El intercambio de datos entre un programa y otro en una computadora o a través de una red.

IPX/SPX (Internet Packet Exchange/Sequenced Packet Exchange) intercambio de paquetes entre redes/intercambio secuencial de paquetes. Conjunto de protocolos de comunicaciones de Novell NetWare que se utilizan para comunicaciones entre procesos.

El protocolo **SPX** garantiza que un mensaje completo llegue intacto, y emplea el protocolo NetWare **IPX** como mecanismo de distribución.

IRQ (Interrupt ReQuest) requerimiento de interrupción. Una interrupción de hardware de un dispositivo periférico que demanda la atención del procesador.

ISAM (Indexed Sequential Access Method) método de acceso secuencial indexado.

ISDN (Integrated Services Digital Network) red digital de servicios integrados. Un estándar interacional de telecomunicaciones para la transmisión de voz, video y datos a través de una línea de comunicaciones digitales.

ISO (International Standards Organization) organización internacional de estándares.

L

LAN (Local Area Network) red de área local. Red de comunicaciones que sirve a usuarios dentro de un área geográficamente limitada.

M

MAC término común para designar la computadora Macintosh.

Mail merge inserción, mezcla por correspondencia. La impresión de un modelo de carta personalizado, que usualmente es una opción en programas de procesamiento de textos o base de datos. Está compuesta por una carta y una lista de nombres y direcciones. Por ejemplo, en esta carta: Estimado A: Gracias por ordenar B de nuestro depósito C..., A, B y C son puntos de inserción en los cuales los datos son insertados desde la lista.

MAN (Metropolitan Area Network) red de área metropolitana. Red de comunicaciones que abarca un área geográfica como una ciudad o un suburbio.

MAU (Multi-station Access Unit) unidad de acceso a múltiples estaciones. Núcleo central en una red de área local de tipo anillo de señales.

Merge insertar, combinar. Véase *mail merge* y *concatenate.*

MINIX versión de UNIX para el PC, Mac. Viene con un código fuente completo.

MYSQL es un gestor de bases de datos SQL (Structured Query Lenguaje.

N

Network adapter adaptador de redes. Tarjeta de circuito impreso que se conecta a una estación de trabajo o a un servidor y controla el intercambio de datos en una red.

NFS (Network File System) sistema de archivo en red. NFS un sistema de archivo distribuido de Sun Microsystems, que permite que múltiples usuarios compartan los datos sin tener en cuenta el tipo de procesador, sistema operativo, arquitectura de red o protocolo.

NIC (Network Interface Card) tarjeta de interfaz de redes. Igual que network adapter.

NSFNET (Red de la Fundación Nacional para la Ciencia).

O

OS/2 Sistema operativo multitarea de un solo usuario para PC 286 y superiores. Las versiones de 16 bits se han desarrollado conjuntamente por IBM y Microsoft. Las versiones de 32 bits se han desarrollado independientemente.

OSI (Open System Interconnection) interconexión de sistemas abiertos. Un modelo de referencia que fue definido por la ISO (International Standards Organization) como un estándar para la comunicaciones mundiales.

Outlet roseta de conexión.

P

PAD (Packet Assembly/Disasembly) ensamblador/desensamblador de paquetes.

PC-MOS Sistema operativo multiusuario para las PC de The Software Link, Inc. Ejecuta casi todas las aplicaciones DOS normales.

PIC (Programmable Interrupt Controller) controlador programable de interrupciones de la Posición Independiente del Código.

POO Programación Orientada a Objetos.

PSTN (Public Switched Telephone Network) red pública de telefonía conmutada. La red mundial de telefonía de voz.

PYMES (Pequeña y Mediana Empresa).

R

RAD (Rapid Application Development) desarrollo rápido de aplicaciones. Aproximación al desarrollo de sistemas que incorpora una variedad de herramientas de diseño automatizadas (CASE).

RAM (Random Access Memory) memoria de acceso aleatorio.

RARP (Reverse Address Resolution Protocol) protocolo de resolución de direcciones reversas.

Router enrutador, encaminador. Es un dispositivo usado para conectar dos subredes que pueden o no ser similares. Los enrutadores se emplean en redes complejas en las que hay múltiples vías de comunicación entre los usuarios de la red. El ruteador emplea el protocolo de internet presente en cada ruteador y en cada sistema final de la red. El ruteador opera en la capa tres del modelo OSI.

RPC (Remote Procedure Call) llamada a procedimiento remoto. Interfaz que permite a un programa llamar a otro en una ubicación remota. Una RPC normal permite que un programa de aplicación sea utilizado sin cambios en una variedad de redes.

RTL (Run-time library) biblioteca de tiempo de ejecución.

S

SMDS (Switched Multimegabit Data Services) servicios de datos conmutados multimegabit. Servicios de datos de alta velocidad comprendidos en el rango de 45 Mbits/seg. Propuesto por compañías telefónicas locales que permitirán a compañías construir MAN privados.

SMTP (Simple Mail Transfer Protocol) protocolo simple de transferencia de correspondencia. Brinda capacidad de transferencia de archivos y de correo electrónico.

Switch conmutador. Dispositivo mecánico o electrónico que comanda el flujo de señales eléctricas u ópticas.

T

TCP/IP (Transmission Control Protocol/Internet Protocol) protocolo de control de transmisiones/protocolo Internet. Conjunto de protocolos de comunicaciones desarrollado por la DARPA (Defense Advanced Research Projects Agency- agencia de proyectos de investigación avanzada de defensa) para intercomunicar sistemas diferentes. El protocolo **TCP** controla la transferencia de los datos, y el **IP** brinda el mecanismo para encaminarla.

U

UDP (User Datagram Protocol) protocolo de datagrama para usuario.

UML (Unified Modeling Language) lenguaje unificado de modelado.

UNIX un sistema operativo multiusuario y multitarea de AT&T que se ejecuta en una amplia variedad de sistemas de computación de micro y macrocomputadoras.

USB (Universal Serial Bus) bus serie universal.

UTP (Unshielded Twisted Pair) cable par trenzado.

V

VB (Visual Basic) lenguaje de programación Visual Basic.

VCL (*Visual Component Library*) librería de componentes visuales.

W

WAN (Wide Area Network) red de área ancha. Red de comunicaciones que abarca áreas geográficas amplias, como pueden ser estados y países.

WWW (World Wide Web). Servicio basado en el Sistema de hipertexto, método para presentar información a través de la conexión entre documentos. Los creadores de WWW introducen además de texto, fotos , sonido y video.

X

XENIX implementación del UNIX System V para computadoras PC 286 y superiores.

