



CAPITULO 1

ESTUDIO GENERAL DEL LINUX

1.1 Generalidades del Sistema Operativo LINUX

1.1.1. Introducción a Linux

Linux es una implementación totalmente independiente del núcleo del conocido sistema operativo UNÍX, que ha revolucionado el mercado de los Sistemas Operativos. Se ha convertido en el sistema operativo para los negocios, educación, y uso personal. Linux ya no es solo para expertos de UNIX que se sientan durante horas frente a las sencillas terminales.

Linux es un clónico del sistema operativo UNIX que corre en computadores x86, Pentium de INTEL y Athlon de AMD. Soporta un amplio rango de software, desde un panel de control a X Windows del compilador GNU C/C++ al TCP/IP. Es una implementación de



UNIX versátil, distribuida gratuitamente bajo los términos de la Licencia GNU (ver anexo C).

El núcleo de Linux a igual que la mayoría del software que lo complementa está amparado bajo los términos de la fundación GNU, esta licencia le permite al usuario final usar, compartir, copiar y modificar con toda libertad el software que se acoge a esta licencia.

A pesar de ello se puede explotar para su uso comercial, esto es, se puede utilizar para el desarrollo de aplicaciones con carácter lucrativo, al igual que extraer beneficios por su distribución.

Linux fue creado a comienzos de los noventa por Linus Torvals con la ayuda de numerosos programadores de todo el mundo. Linux está a disposición de cualquier usuario que puede descargarse gratuitamente desde Internet y su código fuente está disponible a los programadores. En teoría, el término "Linux" se refiere sólo al núcleo del sistema operativo, pero ha llegado a describir su totalidad y el conjunto de aplicaciones. Linux pretende que el cumplimiento de POSIX (una serie de normas que definen UNIX) siga siendo compatible con otros sistemas operativos tipo UNIX.

Linux es conocido por su estabilidad y comprende todas las funciones que podrían esperarse de un sistema operativo, como memoria virtual, una auténtica capacidad multitarea, redes TCP/IP, bibliotecas compartidas y capacidades multiusuario que permiten a cientos de personas utilizar un solo sistema simultáneamente. Linux, inicialmente creado para usuarios particulares, puede ahora ejecutarse en gran variedad de plataformas, como Macintosh, PowerPC, SUN Microsystems o compatibles con IBM. En la actualidad, este sistema operativo cuenta con más de 10 millones de usuarios en todo el mundo y sigue



creciendo a medida que los programadores y los entusiastas en general intercambian ideas y aportan código para continuar desarrollando este software freeware conocido como Linux.

1.1.2 Los Comienzos

En la década de los ochenta, Andrew Tanenbaum, un profesor holandés de informática especializado en el diseño de sistemas operativos, creó uno, llamado MINIX para ayudar a sus alumnos a aprender las complejidades de UNIX en los sistemas que utilizaban en casa. En 1991, un estudiante finlandés de 21 años, Linus Torvalds, comenzó a explorar formas de incrementar la funcionalidad de MINIX. Su objetivo era crear un MINIX mejorado y gratuito para su PC x86.

1.1.3 El núcleo de Linux

Los experimentos de Torvalds a comienzos de los noventa dieron lugar al núcleo de Linux, código operativo y elemento fundamental del sistema. Puso el código de origen a disposición del público a través de la Licencia Pública General de GNU (GPL) (ver ANEXO C) para que los programadores de todo el mundo pudieran estudiarlo, modificarlo y añadir mejoras a sus desarrollos iniciales. Esta colaboración mundial ha constituido la razón fundamental del rápido progreso de Linux y ha dado lugar al momento a la versión 2.4.x.

El núcleo de Linux es software de código abierto (open source): la disponibilidad del código fuente permite a los programadores localizar los errores y proporcionar parches al núcleo. En consecuencia, el núcleo de Linux se ha hecho extraordinariamente robusto desde su desarrollo inicial y con frecuencia se considera más sólido que muchos de los sistemas operativos comerciales. Además, el núcleo es modular, por lo que el código sigue siendo pequeño y rápido sin repercutir en su funcionalidad ni malgastar recursos de memoria.



1.1.4 Las ventajas de Linux

Entre algunas de las ventajas del sistema operativo Linux se encuentran la estabilidad, la capacidad multitarea, la memoria virtual, protocolos TCP/IP eficaces, bibliotecas compartidas y capacidades multiusuario. Gracias a esto, varios usuarios pueden utilizar un solo sistema al mismo tiempo a través de una red y un sitio Web. Linux se ejecuta en modo totalmente protegido y admite arquitecturas de 32 y 64 bits. Estas son algunas de las múltiples ventajas que ofrece Linux:

Versatilidad: Los sistemas Linux pueden constituir una alternativa para las empresas con necesidades informáticas específicas. Como las distribuciones de Linux contienen el código fuente de todas las partes del sistema, los programadores pueden modificarlo para satisfacer necesidades concretas.

Fácil configuración: Linux dispone de una implementación gratuita de X Windows que cumple la normativa X/Open. Por esta razón, la mayoría de las aplicaciones basadas en X se pueden ejecutar en Linux sin necesidad de grandes modificaciones. La mayoría de las distribuciones Linux se suministra completamente preconfigurada e incorpora utilidades e instaladores de configuración gráfica.

Capacidades avanzadas de red: La compatibilidad de redes de Linux suele considerarse superior a la de la mayoría de los sistemas operativos comerciales. Linux permite la conexión a Internet y a otras redes que utilicen los protocolos TCP/IP o IPX vía Ethernet, Ethernet rápida, ATM, módem, protocolo x25, RDSI o token ring. Linux se ejecuta



en hardware x86 o Pentium de Intel, lo que permite a las empresas crear un archivo de impresión o un servidor Web por el solo costo de hardware para PC. Dado que el costo de la licencia de muchos sistemas operativos comerciales representa casi la mitad del costo del hardware de PC, un servidor Linux puede resultar ideal para reducir los costos de hardware y software. Como servidor Web, Linux resulta a menudo mejor y más fiable que muchos sistemas operativos comerciales. En la actualidad, cientos de proveedores de servicios de Internet (ISP), laboratorios informáticos universitarios y algunas grandes empresas utilizan servidores Linux.

Integración en redes de área local (LAN): Linux se integra fácilmente en redes de área local, independientemente del sistema que se ejecute. Además, Linux ofrece toda una serie de posibilidades de compatibilidad e integración de Macintosh, DOS, Windows 3.1x, Windows 9x, Windows NT/2000, Novell y OS/2 mediante el uso de protocolos nativos de los respectivos sistemas operativos con menos requisitos de memoria que la mayoría de los sistemas operativos comerciales.

Velocidad: Como Linux es eficaz en la gestión de memoria, potencia de CPU y espacio en disco, el sistema operativo tiene una gran reputación por ser extraordinariamente rápido.

Interfaz gráfica intuitiva: Linux dispone de más de una docena de interfaces gráficas configurables, que se ejecutan sobre XFree86, una implementación gratuita del sistema X Windows. Actualmente, Linux dispone de dos administradores de ventanas: K Desktop Environment (KDE) y GNU Network Object Model Environment (GNOME). Estos administradores pueden adoptar muy diversos aspectos al tiempo que ofrecen la misma funcionalidad de "señalar y hacer clic" de otros entornos.



Compatibilidad y coexistencia: Linux siempre ha luchado por ser un entorno abierto y compatible, por lo que puede coexistir fácilmente con otros sistemas operativos dentro del mismo sistema. Esto permite a los nuevos usuarios de Linux ponerlo a prueba sin necesidad de abandonar su viejo sistema operativo ni comprar otro sistema.

1.1.5 Beneficios que ofrece LINUX

Una de las más importantes es, sin embargo, que Linux es una excelente elección para trabajar con UNIX a nivel personal. Linux permitirá desarrollar y probar el software UNIX en una PC, incluyendo aplicaciones de bases de datos y X Windows. Con Linux, se podrá correr un sistema UNIX propio y adaptarlo a cualquier necesidad. La instalación y uso de Linux es también una excelente manera de aprender UNIX si no se tiene acceso a otras máquinas UNIX.

Linux no es solo para los usuarios personales de UNIX. Es robusto y suficientemente completo para manejar grandes tareas, así como necesidades de computo distribuidas. Muchos negocios especialmente los pequeños se están cambiando a Linux en lugar de otros entornos de estación de trabajo basados en UNIX. Las universidades encuentran a Linux perfecto para dar cursos de diseño de sistemas operativos. Grandes vendedores de software comercial se están dando cuenta de las oportunidades que puede brindar un sistema operativo gratuito.

1.1.6 Características Técnicas del sistema

Linux implementa la mayor parte de las características que se encuentran en otras implementaciones de UNIX, mas algunas otras que no son habituales. En esta sección nos daremos una vuelta por todo ello.



Linux es un sistema operativo completo con multitarea y multiusuario (como cualquier otra versión de UNIX). Esto significa que pueden trabajar varios usuarios simultáneamente en él, y que cada uno de ellos puede tener varios programas en ejecución.

El sistema Linux es compatible con ciertos estándares de UNIX a nivel de código fuente, incluyendo el IEEE POSIX, System V y BSD (*Licencia de la Universidad de California en Berkeley por la cual el tenedor de los derechos de autor sigue siendo el propietario registrado pero donde no se imponen restricciones en la distribución al usuario del código.*). Fue desarrollado buscando la portabilidad de las fuentes: encontrará que casi todo el software gratuito desarrollado para UNIX se compila en Linux sin problemas. Y todo lo que se hace para Linux (código del núcleo, drivers, librerías y programas de usuario) es de libre distribución.

En Linux también se implementa el control de trabajos POSIX (que se usa en los shells csh y bash), las pseudo-terminales (dispositivos pty), y teclados nacionales mediante manejadores de teclado cargable dinámicamente. Además, soporta consolas virtuales, lo que permite tener mas de una sesión abierta en la consola de texto y conmutar entre ellas fácilmente.

Linux soporta diversos sistemas de archivos para guardar los datos. Algunos de ellos, como el **ext2fs** (Sistema de Ficheros Extendido 2) o el reciente **ext3**, han sido desarrollados específicamente para Linux. Otros sistemas de archivos, como el Minix-1 o el de Xenix también están soportados. Se podrán acceder desde Linux a los disquetes y particiones en discos duros formateados con MS-DOS, y Windows (FAT y FAT32). 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 archivos). Y también se han portado los clientes de TCP/IP, como FTP, telnet.

El núcleo de Linux ha sido desarrollado para utilizar las características del modo protegido de los microprocesadores x86 o Pentium de INTEL y Athlon de AMD. En concreto, hace uso de la gestión de memoria avanzada del modo protegido y otras características avanzadas. Cualquiera que conozca la programación de cualquier x86 en el modo protegido sabrá que este modo fue diseñado para su uso en UNIX. Linux hace uso de esta funcionalidad precisamente.

El núcleo soporta ejecutables con paginación por demanda. Esto significa que solo los segmentos del programa que se necesitan se cargan en memoria desde el disco. Las páginas de los ejecutables son compartidas mediante la técnica copy-on-write (extrae primero una copia de los datos menos usados almacenados en el Disco Duro), contribuyendo todo ello a reducir la cantidad de memoria requerida para las aplicaciones.

Con el fin de incrementar la memoria disponible, Linux implementa la paginación con el disco: puede tener un espacio de intercambio o "swap" mayor de 256MB en el disco duro. Cuando el sistema necesita más memoria, grabará 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 esta.



La memoria dedicada a los programas y a la cache de disco esta 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 único fichero, como sucede en SunOS (Sistema Operativo de Sun Microsystems). Así, los ejecutables serán mas 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.

Para facilitar la depuración de los programas, el núcleo de Linux puede generar volcados de la imagen de memoria de los programas (archivos **core**: Fichero texto generado por Linux, contiene errores producidos en la compilación de un programa). Entre esto y la posibilidad de compilar ejecutables con soporte de depuración, el programador podrá averiguar la causa de los fallos de su programa.

1.2 El Sistema X Windows, Interfaz Gráfica para Linux

El sistema X Windows es una interfaz gráfica estándar para máquinas UNIX. Es un potente entorno que soporta muchas aplicaciones. Usando X Windows, el usuario puede tener múltiples terminales a la vez en ventanas sobre la pantalla, cada una conteniendo una sesión diferente.

Han sido escritas muchas aplicaciones específicamente para X Windows, como juegos, utilidades gráficas, herramientas de programación y documentación, y muchas más. Con Linux y X Windows, una PC es una auténtica estación de trabajo. Junto con una red TCP/IP puede incluso



visualizar aplicaciones que se están ejecutando en otras máquinas en su pantalla local, tal y como es posible con otros sistemas corriendo X Windows.

El sistema X Windows fue desarrollado originalmente en el MIT y es de libre distribución. A pesar de esto muchas empresas han distribuido sus mejoras particulares al diseño original de X Windows. La versión de X Windows disponible para Linux es conocida como XFree86, una adaptación de X11R6 de libre distribución para sistemas UNIX basados en 80386, como es Linux. XFree86 soporta una gran variedad de tarjetas de video, incluyendo VGA, Super VGA y gran cantidad de tarjetas aceleradoras de video. Esta es una distribución completa de X Windows conteniendo el servidor de X Windows, muchas aplicaciones y utilidades, librerías de programación y documentación.

El aspecto y comportamiento de la interfaz de las X Windows esta controlado por el controlador de ventanas (KDE o GNOME). Este amigable programa es el encargado de la situación de las ventanas, redimensionado, iconización, movimiento de las ventanas, aspecto de los marcos de las ventanas, etc.

La única advertencia con las X Windows es los requerimientos de máquina y memoria. Para usarlo adecuadamente serán necesarios 64 MB o más. También es recomendable usar un procesador rápido, pero tener mayor cantidad de memoria física es mucho más importante.

Además, para conseguir realmente buenas prestaciones de video, es muy recomendable usar una tarjeta aceleradora. Con suficiente hardware, encontrara que ejecutar X en Linux es tan rápido o más rápido que ejecutar X en otras estaciones de trabajo UNIX.

A diferencia de otros sistemas operativos como Windows 9x, Macintosh, OS/2 el aspecto y funcionamiento depende mucho del gestor de ventanas elegidos, podemos elegir entre: twm, fvwm95 y los últimos KDE y GNOME



1.2.1 GNOME

Es un entorno de ventanas que permite al usuario configurar fácilmente su computador. Incluye un panel, un escritorio, un conjunto de utilidades y un conjunto de convenciones que permite a las aplicaciones cooperar y ser consistentes. GNOME es en su totalidad código abierto, con el código fuente disponible libremente y desarrollado por cientos de programadores en todo el mundo.

GNOME es un acrónimo de 'GNU Network Object Model Environment', entorno de trabajo en red orientado a objetos, por lo que GNOME forma parte del más amplio proyecto GNU. El proyecto GNU nació en 1984 con el objetivo de desarrollar un sistema operativo tipo Unix completamente libre.

1.2.2 KDE

KDE es el acrónimo de "K Desktop Environment" o "Entorno de Escritorio K". El KDE es un proyecto iniciado por [Matthias Ettrich](#) en 1996. El objetivo del proyecto KDE es compaginar el poder de sistemas operativos basados en Unix con la comodidad esperada actualmente de una interfaz moderna de usuario.

KDE no es un gestor de ventanas. Aunque KDE incluye un gestor de ventanas muy sofisticado (KWM), KDE es mucho más que eso: es un entorno de escritorio integrado (en Julio de 1998, de una cifra de líneas de código de KDE que superaba ampliamente el medio millón, sólo unas 7.000 estaban dedicadas al manejo de ventanas), incluyendo un gestor de archivos, gestor de ventanas, sistema de ayuda, sistema de configuración e innumerables herramientas y utensilios, además de un número de aplicaciones siempre en



aumento, incluyendo pero no limitándose a clientes de correo y noticias, programas de dibujo, visores postscript , etc.

En el ANEXO A podemos encontrar la pantalla principal de estos dos ambientes gráficos.

1.3 Requerimientos de Hardware

Actualmente Linux soporta sistemas con una CPU Intel x86, o Pentium con todas las variantes del tipo de CPU, como el Pentium I, Pentium II, Pentium III y Pentium IV. Los "clónicos" no Intel, como AMD también funcionan con Linux.

La mainboard debe ser de arquitectura PCI en cuanto a bus se refiere. Estos términos definen como interactúa el sistema con los periféricos y otros componentes por medio del bus principal. La mayoría de los sistemas vendidos hoy son de bus PCI.

Los sistemas que usan arquitectura de bus local (para accesos más rápidos en video y disco) también están soportados. Es recomendable que tenga una arquitectura de bus local estándar.

1.3.1 Requisitos de memoria

Linux, comparado con otros sistemas operativos avanzados, necesita muy poca memoria para funcionar. Debería contar con un mínimo de 2 MB de RAM; sin embargo, es altamente recomendable tener 4 MB para su funcionamiento mínimo y 64MB para Interfaz Gráfica. Cuanta más memoria tenga más rápido se ejecutará su sistema.

Linux soporta el rango completo de direcciones de 32-bits, es decir, utilizara toda la memoria RAM de forma automática, y funcionaría sin problemas con tan



solo 4 megabytes de RAM, incluyendo aplicaciones exigentes como X-Windows, Emacs, y otros. Sin embargo, disponer de mas memoria es casi tan importante como tener un procesador más rápido. 8 MB son mas que suficiente para uso personal; 32 MB o más pueden ser necesarios si espera una fuerte carga de usuarios en el sistema.

La mayoría de los usuarios de Linux, reservan una parte del disco duro para espacio de intercambio ("swapping") que se usa como RAM virtual. Incluso si dispone de bastante memoria RAM física en su máquina, puede que quiera utilizar un área de "swap". El área de "swap" no puede reemplazar a una memoria física RAM real, pero puede permitir a su sistema ejecutar aplicaciones más grandes guardando en disco duro aquellas partes de código que están inactivas.

1.3.2 Requisitos de espacio en disco

Efectivamente, para instalar Linux, necesitara tener algo de espacio libre en su disco duro. Linux soporta múltiples discos duros en la misma maquina; puede disponer de espacio para Linux en múltiples unidades si es necesario.

La cantidad de espacio en disco duro que necesita, depende en gran medida de sus necesidades y de la cantidad de software que va a instalar. Linux es relativamente pequeño con relación a las implementaciones de UNIX; Se podría correr un sistema completo con 100-200 MB de espacio en disco. Sin embargo, si quiere disponer de espacio para expansiones, y para paquetes más grandes como X Windows, necesitará mas espacio. Si planea permitir a múltiples usuarios utilizar la máquina, tendrá que dejar espacio para los archivos.



También, a menos que tenga una gran cantidad de memoria RAM física (64 MB o más), necesitará crear espacio de intercambio ("swap"), para ser usado como RAM virtual. Cada distribución de Linux normalmente viene con alguno que otro texto que debería ayudarle a estimar la cantidad precisa de espacio a reservar en función del software que planea instalar. Puede ejecutar un sistema mínimo con menos de 200 MB; un sistema completo con toda las opciones en 800 MB o menos; y un sistema grande con sitio para muchos usuarios y espacio para futuras expansiones en un rango de 1-1.50 GB y a nivel de servidor de 1.5-3 GB. Estas cifras son meramente de orientación; se deberá decidir, según sus necesidades y objetivos, los requerimientos específicos de almacenamiento para el sistema.

1.3.3 Requisitos de monitor y adaptador de video

Linux soporta todas las tarjetas de video estándar VGA y Super VGA así como monitores para la interfaz por defecto basado en texto. En general, si la combinación que tiene monitor y tarjeta de video funcionan bajo otro sistema operativo como MS-DOS, debería funcionar perfectamente con Linux.

Los entornos gráficos como el Sistema X Windows tienen requerimientos propios de hardware para la tarjeta de video.

1.3.4 Mouse y otros dispositivos apuntadores

Normalmente solo se usará un mouse bajo un entorno gráfico como el Sistema X Windows. Linux soporta todos los mouse serie estándar, incluyendo PS/2 de 2 o tres botones.

1.3.5 Almacenamiento en CD-ROM



Casi todas las unidades de CD-ROM usan la interfaz ATAPI. Hay un número de unidades de CD-ROM que se ha comprobado funcionan bajo Linux ya que soporta el sistema de archivos estándar para CD-ROMs ISO-9660.

1.3.6 Impresoras

Linux soporta todas las impresoras paralelo. Si puede acceder a su impresora por el puerto paralelo desde MS-DOS, u otro sistema operativo, debería poder acceder a él desde Linux también. El software de impresión de Linux consiste en el software estándar de UNIX lp y lpr.

1.4 Versiones comerciales de Linux

En el mercado existen compañías y organizaciones que han creado sus propias versiones del sistema operativo. En la actualidad ya están disponibles más de 20 tipos, creados por compañías u organizaciones no lucrativas e incluso individuos, pero quienes los hacen, están obligados a revelar el código fuente de su versión para que otros, si lo desean, puedan modificarlo. La distribución de Linux esta a cargo de varias empresas las cuales han empaquetado el Sistema Operativo, herramientas para configuración de red, servidor Web, ambiente gráfico, utilitarios, juegos, procesador de palabras, etc.

Prácticamente se paga por el CD, los manuales y el tiempo que significó para la empresa distribuidora “bajar” toda la información de Internet. Algunas de estas distribuidoras del sistema son: Red Hat, Debían y Suce. La diferencia entre una y otra está en el número de programas que se incluyen así como la instalación, pero el kernel y comandos son los mismos.

1.4.1 Red Hat 7.2



Aunque Red Hat ha desarrollado una de las distribuciones más utilizadas de Linux, esta empresa también ha creado muchos programas comerciales. Red Hat ha desarrollado por ejemplo un paquete de gestión de Linux, denominado RPM, bajo la Licencia Pública General (GPL) para su utilización otras distribuciones.

Junto a las versiones GPL de Linux y el RPM, Red Hat también cuenta con un entorno de trabajo para aplicaciones llamado Appixware, que incluye un procesador de textos, un programa de hoja de cálculo, un programa de presentación de gráficos, una herramienta de correo y varias herramientas de desarrollo. Red Hat también dispone de una versión comercial de Motif para desarrollar y ejecutar X en Linux.

1.4.2 Open Linux 2.2



Es un Sistema Operativo UNÍX a bajo costo, basado en el Kernel de Linux 2.0. Este sistema incluye una interfaz grafica de usuario capaz de administrar el sistema y algunos recursos de red, como la interacción entre cliente y servidor en Internet y en la mayoría de sistemas de trabajo en Red.

Open Linux Base es una pasarela no exclusiva que incorpora todos los protocolos y servicios de cliente, servidor y enrutador. Igualmente, incluye un servidor X comercial de metrolink.

1.4.3 COREL Linux



Corel LINUX es un sistema operativo potente y fácil de manejar, especialmente diseñado para su PC de escritorio. Entre las mejoras de Corel con respecto a Debian GNU/Linux y KDE destacan un entorno gráfico y productivo gracias al cual no se



perderá el tiempo aprendiendo de nuevo a hacer su trabajo. El sistema tiene incorporados un sencillo programa gráfico de instalación, un completo administrador de archivos, opciones centralizadas de configuración y actualización del sistema, un programa de correo electrónico y un explorador Web.

1.4.4 Otras versiones.

Existen otras versiones las cuales están disponibles en el ANEXO B.

1.5 Análisis y selección de la versión comercial de Linux a utilizar en el proyecto.

Linux ofrece una gran variedad de versiones comerciales, muchos de las cuales se distribuyen en forma gratuita (de acuerdo a la licencia GNU. Ver detalles en el ANEXO C) a través del Internet o con un mínimo costo por otros medios (ver sección 1.2). Todas estas versiones poseen casi las mismas características que ofrece el Kernel de Linux, Así que de estos se ha elegido el Sistema Operativo Red Hat Linux Versión 7.2, por las siguientes razones, siempre considerando como factor importante que para la realización de nuestro proyecto era necesario el trabajo con usuarios típicamente relacionados con el Sistema Operativo Windows 9X:

- ☞ Red Hat Linux destaca por su fácil instalación e interfaz intuitiva similar a Windows, ofreciéndose además al usuario la posibilidad de que Linux asuma el control de todo el disco duro, o que sencillamente coexista en éste con Windows, en particiones distintas, ofreciendo una alternativa para poder trabajar tanto en Linux como en Windows en una sola máquina.
- ☞ Excelente documentación y soporte gratuito durante 30 días.



- ☞ Red Hat Linux identifica automáticamente los periféricos conectados a la computadora, instalando los controladores necesarios.
- ☞ Compatible con el resto del software Linux.
- ☞ Integración total en los entornos de red de Windows y UNIX / Linux.
- ☞ Libertad de seleccionar la interfaz gráfica ya sea GNOME o KDE al gusto del usuario.
- ☞ Centro de control de fácil manejo para la configuración y personalización del sistema.
- ☞ Incluye aplicaciones y utilidades prácticas.
- ☞ Completo sistema de ayuda
- ☞ Sencillo diseño gráfico de arrastrar y soltar
- ☞ Fácil exploración de las unidades locales y de red
- ☞ Sencilla actualización del sistema con linuxconf y otras utilidades
- ☞ Los requisitos de Hardware del sistema son mínimos para la instalación en una máquina disponible en el municipio:
 - ☞ Procesador Pentium o compatible
 - ☞ 64 MB de RAM (se recomiendan 128 MB)
 - ☞ 1 GB de espacio en el disco duro
 - ☞ Unidad de CD-ROM, mouse y tarjeta PCI VGA de 2 MB o superior

A continuación presentamos algunas comparaciones de las características básicas necesarias para nuestro trabajo de Red Hat con respecto a otras distribuciones:



CARACTERÍSTICAS	RED HAT	DEBIAN	COREL
Entorno KDE	SI	SI	SI
Entorno GNOME	SI	NO	NO
Multilinguaje	SI	Inglés y Español	Inglés y Español
Maneja paquetes RPM	SI	NO	NO
Fácil Instalación	SI	NO	SI
Documentación en español	SI	SI	SI
PostgreSQL	SI	NO	NO



CAPITULO 11

SISTEMAS DE GESTION DE BASES DE DATOS (SGBDD) FREEWARE PARA LINUX

2.1 Software FREEWARE

2.1.1 Introducción

Uno de los fenómenos más atractivos que se nos presentan en el mundo Internet es el de la existencia de diversas organizaciones que, sin ánimo de lucro, desarrollan y distribuyen software de todo tipo en forma gratuita. Desde Sistemas Operativos, Compiladores y herramientas de programación; hasta procesadores de texto, hojas de cálculo y aplicaciones de usuario final.

El término software gratuito, no garantiza que no se tenga que pagar absolutamente nada para adquirirlo. El término “gratis” en materia de software hace alusión que es un



producto de código abierto y que esta disponible públicamente, los usuarios pueden personalizar al software en cualquier forma que ellos deseen, siempre y cuando den a conocer sus cambios como productos de código abierto, siempre basándose a la regulación de la licencia GNU (Anexo C).

Es software de excelente calidad, superando en ocasiones a sus homólogos comerciales. Gran parte del software FREEWARE que se encuentra disponible en Internet es de alta viabilidad, por lo que, muchas empresas lo incluyen como parte de su tecnología informática. Los mismos que puede reducir los gastos, ofrecer seguridad, soporte técnico, y otros.

2.1.2. Conceptos Básicos

Software Freeware : Son programas gratuitos, donados y con su código fuente incluido, para ser usados por cualquier persona. Supervisado Bajo licencia GNU.

Software Shareware: Son programas que permiten al usuario probarlos antes de comprarlos. Si se prueba un programa en su versión de evaluación (shareware), le llega a gustar y va a continuar utilizándolo más allá de 30 días, se debería registrar como usuario de ese programa abonando su precio al autor. A cambio, el autor le enviará la licencia de uso continuado así como la última versión completa del programa.

Diferencias : En términos técnicos, el **shareware**, al contrario que el **freeware**, pide al usuario pagar por el programa tras un período de prueba. Los precios varían entre cinco y treinta dólares. El pago queda a la honradez del usuario, aunque hay programas que se bloquean automáticamente después de un cierto número de veces y piden al usuario "por favor, regístrese y pague, que lo necesito".



La mayoría de los programas habitualmente disponibles, son de dominio público y no están protegidos bajo propiedad intelectual o "copyright". Sin embargo, aunque el **freeware** no cueste nada, generalmente su autor mantiene la propiedad intelectual.

El movimiento "freeware", que nació como reacción a los programas comerciales, es casi tan antiguo como la propia red Internet. Pero la tradición de compartir programas a través de Internet continúa hoy en día. Hay programas gratuitos disponibles en todas las categorías y para todas las plataformas, desde graficadores, programas educativos, de contabilidad, correo electrónico, juegos y hasta novedosos programas de navegación por Internet.

Los creadores de muchos de estos programas, son estudiantes, programadores aficionados y programadores profesionales. Trabajan con la fe de que sus servicios deben estar disponibles para todos, libres de pago y principalmente con su código fuente.

2.1.3 Ventajas de su empleo

- ☞ De excelente calidad.
- ☞ Permite reducir los gastos.
- ☞ Ofrece seguridad y tranquilidad a los usuarios porque brinda soporte técnico.
- ☞ Manuales en línea autorizados.
- ☞ Códigos fuentes disponibles.
- ☞ Ahorro de divisas al no necesitar comprar licencias.
- ☞ Ayuda a través de foros de discusiones.
- ☞ Aplicación de conocimientos técnicos para personalizar Sistemas Operativos y Aplicativos.



2.1.4 Investigación Técnica del Software Freeware disponible en Internet .

La mayor fuente de software es Internet, incluso, muchos programas están disponibles únicamente en él. Si un programa freeware no está disponible en Internet lo más seguro es que no esté disponible de otra forma.

La mayor cantidad de software freeware corresponde a programas para sistemas operativos UNIX, siendo el sistema [Linux](#) el principal exponente de esta clase de software, pero a diferencia de otros sistemas, Linux normalmente viene acompañado de una gran cantidad de aplicaciones de todo tipo con su documentación, inclusive Sistemas de Gestión de Bases de Datos. Para adquirir versiones actualizadas de Linux es necesario bajar de Internet, normalmente, podemos decir que, contrario al software propietario, se puede sencillamente recurrir a algún amigo que ya tenga Linux u otro software para obtener una copia. El software freeware para sistemas operativos Microsoft, que es bastante escaso, se obtiene casi que exclusivamente de Internet.

Para que un municipio o cualquier usuario potencial pueda obtener algún beneficio del software freeware es necesario llevar a cabo la mayor parte o todas las actividades que se muestran en la siguiente tabla¹:

Actividades necesarias para lograr algún beneficio del software freeware.

#	Actividad	Metodología	Resultado
1	Conocimiento de la existencia del software	Conocimiento previo o exploración	documento
2	Localización del software	Exploración	documento
3	Obtención del software y	Labor de un técnico	software

¹ REFERENCIA: <http://orbita.starmedia.com/>



	documentación relacionada	informático	
4	Conocimiento del funcionamiento del software	Auto-aprendizaje - labor de un técnico informático	conocimiento
5	Documentación del software (necesaria en algunos casos, deseable en otros)	Labor de un técnico informático	instructivos o manuales
6	Asistencia a la instalación y uso para el usuario final	Labor de un técnico informático	servicio
7	Adaptaciones y mejoras	Labor de un técnico o profesional informático	servicio y software

A continuación vamos a explicar estos pasos en la búsqueda, evaluación, selección y empleo de los SGBDD Freeware. Las primeras 5 actividades listadas en la tabla anterior se emplea para obtener los SGBDD Gratuitos desde Internet, la actividad 6 equivale a instalar el software. La actividad 7 normalmente corresponde a lo más importante, que es la necesidad de tener un motor actualizado y libre de errores aplicando los respectivos parches. Ya que este trabajo busca determinar que Sistema de Gestión de Bases de Datos Freeware disponible en Internet que pueden ser de utilidad para los municipios, se incluyen la selección y la evaluación de los SGBDD. Las diferentes actividades realizadas se detallan en seguida.

Cabe Indicar que estos pasos también puede ser aplicado a cualquier otro tipo de software distinto a los SGBDD.

Conocimiento de la existencia de los SGBDD Freeware

Se puede decir que el conocimiento de la existencia de los SGBDD Freeware, es uno de los puntos de mayor importancia ya que existen cantidades de software freeware pero muchas veces no se conoce de su existencia. La primera inquietud que surge fue: ¿existen Sistemas de Gestión de Bases de Datos Gratuitos?.



Localización del SGBDD Freeware

Es necesario establecer el sitio o los sitios (las direcciones) donde se encuentra disponible el software, sus manuales y la documentación relacionada que pueda existir.

Tanto el conocimiento de la existencia de los SGBDD Freeware como su localización, son actividades que se realizan principalmente utilizando metodología exploratoria. La exploración de Internet es posible gracias al uso los denominados motores de búsqueda, como Altavista, Yahoo, Geogle, Lycos, etc.

Selección del SGBDD Freeware

La selección del software es necesaria porque el software va dirigido a un grupo específico, como software educativo, juegos, aplicaciones, de contabilidad, etc; en este caso seleccionaremos la categoría de los motores de Bases de Datos.

Para la selección de los SGBDD Freeware, cuyo uso se propone, se hace una revisión del contexto global y necesidades particulares, así como también una revisión teórica respecto a tipos, características y prestaciones.

Obtención de los SGBDD Freeware

Los SGBDD Freeware no solo se puede obtener de Internet, aunque en muchos casos es la única alternativa. Para este trabajo se obtuvo el software principalmente de Internet pero también de CD-ROMs, ya sea adquiridos o copiados (el software freeware lo permite).



Conocimiento del funcionamiento de los SGBDD Freeware

Prácticamente la única alternativa para conocer el funcionamiento de los SGBDD Freeware es mediante el auto-aprendizaje. En otros casos el auto-aprendizaje se limitó a conocer la forma básica de instalación y operación del SGBDD Freeware.

Evaluación de los SGBDD Freeware

Por evaluación se entiende la obtención de información para toma de decisiones. La evaluación del software es ineludible, antes de su posterior instalación y uso en el municipio. La evaluación se basó considerando aspectos tales como Prestaciones, Facilidad de Instalación y otros.

Documentación de los SGBDD Freeware

Si bien los programas incluyen manuales y documentación relacionada, en varias situaciones es necesario o deseable la elaboración de documentación adicional. Muchas veces el SGBDD Freeware únicamente está disponible en idioma inglés.

2.2. Sistemas de Gestión de Bases de Datos

FREEWARE

Una vez aplicado el procedimiento anterior, consideramos que cuatro motores de BDD gratuitos que hemos seleccionado (PostgreSQL, MySQL, InterBase, Mimer) son los que ofrecen mayor cantidad de soporte técnico, documentación y facilidad de instalación (poseen paquetes RPM para RedHat 7), por lo que a continuación los estudiaremos con más detalle cada uno de ellos.



2.2.1. POSTGRESQL

PostgreSQL



2.2.1.1. Generalidades

El sistema de gestión de bases de datos objeto-relacional ahora conocido como PostgreSQL (y anteriormente llamado Postgres95) se deriva del paquete Postgres escrito en Berkeley. Con más de una década de desarrollo a sus espaldas, PostgreSQL es una de las bases de datos de fuente abierta muy avanzada, ofreciendo control de concurrencia multiversión, soportando casi todas las construcciones SQL (incluyendo subselects, transacciones, y tipos y funciones definidas por el usuario), y teniendo un amplio rango de enlaces con lenguajes disponible (incluyendo C, C++, Java, perl, tcl y python).

La implementación del DBMS Postgres empezó en 1.986. Postgres ha sobrellevado varias actualizaciones mayores desde entonces. La primera "versión de demostración" estuvo operativa en 1.987, y fue mostrada en la Conferencia ADM-SIGMOD de 1.988. Se liberó la Versión 1 a unos pocos usuarios externos en Junio de 1.989, la Versión 2 en Junio de 1.990 con el nuevo sistema de reglas. La Versión 3 apareció en 1.991 y añadió soporte para múltiples gestores de almacenamiento, un ejecutor de query avanzado, y se rescribió el sistema de reglas de reescritura. En su mayor parte, las versiones hasta Postgres95 (ver más adelante) se enfocaron en la portabilidad y en el rendimiento.

Postgres95

En 1994, [Andrew Yu](#) y [Jolly Chen](#) añadieron un intérprete del lenguaje SQL a Postgres. Postgres95 fue liberado a continuación en la Web para encontrar su propia vía en el mundo como un descendiente de fuente abierta del código original de Postgres de Berkeley.



El código de Postgres95 era completamente ANSI C y se ajustó su tamaño en un 25%. Muchos cambios internos incrementaron el rendimiento y las posibilidades de mantenimiento. Postgres95 v1.0.x corría entre un 30 y un 50% más rápido en el Benchmark de Wisconsin que Postgres v4.2. Aparte de los errores corregidos, estos eran las mayores mejoras:

PostgreSQL

En 1.996, se veía ya claramente que el nombre Postgres95 no soportaba el paso del tiempo. Se eligió un nuevo nombre, PostgreSQL, para reflejar la relación entre el Postgres original y las versiones más recientes con las capacidades SQL. Al mismo tiempo, se fijó la numeración de la versión para empezar en 6.0, volviendo a la secuencia de numeraciones que empezó originalmente con el Proyecto Postgres.

El énfasis durante el desarrollo de Postgres95 era identificar y conocer problemas existentes en el código del servidor. Con PostgreSQL, el énfasis se pone en incrementar las capacidades y las características, trabajo que continúa en todas las áreas. PostgreSQL es un producto open source. Como tal, depende de la comunidad de usuarios el realizar un soporte continuado².

2.2.1.2. Requerimientos de Software y Hardware.

Sistema Operativo	Versión SSOO	Procesador
AIX	4.3.2	RS6000
FreeBSD	4.0	x86
LINUX	2.0.x	Alpha
LINUX	2.2.x	X86
LINUX	2.0.x	MIPS
LINUX	2.2.5	SPARC

² Manual de Usuario de PostgreSQL



NetBSD	1.4U	X86
SCO Open	Server 5	X86
WINDOWS	Win32	X86

2.2.1.3. Características Generales

☞ **Versión(Septiembre del 2001):** 7.1.3

☞ **Empresa:** University of California at BERKELEY

☞ **Licencia:** Gratuita y Comercial

☞ **Home Site:**

<http://www.postgresql.org/>

<http://www.illustra.com/> (PostgreSQL es comercializado)

☞ **Disponibilidad del Código Fuente:** Sí

☞ **Paquetes Binarios:**

- Debian Package: Sí, estable (requiere libc5)
- RedHat RPM Package: Si, <ftp://ftp.redhat.com/pub/contrib/>
- **Otros Enlaces:**

<http://logical.thought.net/postgres95/>

<http://www.itm.tu-clausthal.de/mirrors/postgres95/>

<ftp://ftp.postgresql.org/pub> (FTP Originario)

<http://metalab.unc.edu/LDP/HOWTO/PostgreSQL-HOWTO.html>

<http://www.postgresql.org/supp-mlists.shtml>



2.2.1.4 Características Técnicas

a) Arquitectura

En el mundo de las bases de datos, Postgres utiliza un simple modelo cliente / servidor de "proceso por usuario". Una sesión de Postgres consiste en los siguientes procesos Unix (programas) cooperando:

- ✓ Un proceso demonio supervisor (postmaster),
- ✓ La aplicación de interfase del usuario (frontend en inglés) (por ejemplo, el programa psql),
- ✓ Los uno o más procesos servidores de acceso a la base de datos (backend en inglés) (el proceso postgres mismo).

Un único postmaster maneja una colección dada de bases de datos en un único host. Tal colección se denomina una instalación o un site. Las aplicaciones de frontend que quieren acceder a una base de datos dada en una instalación realizan llamadas a la librería. La librería envía el requerimiento del usuario a través de la red al postmaster (Cómo se establece una conexión(a)), quien en su turno arranca un nuevo proceso servidor de backend (Cómo se establece una conexión(b))

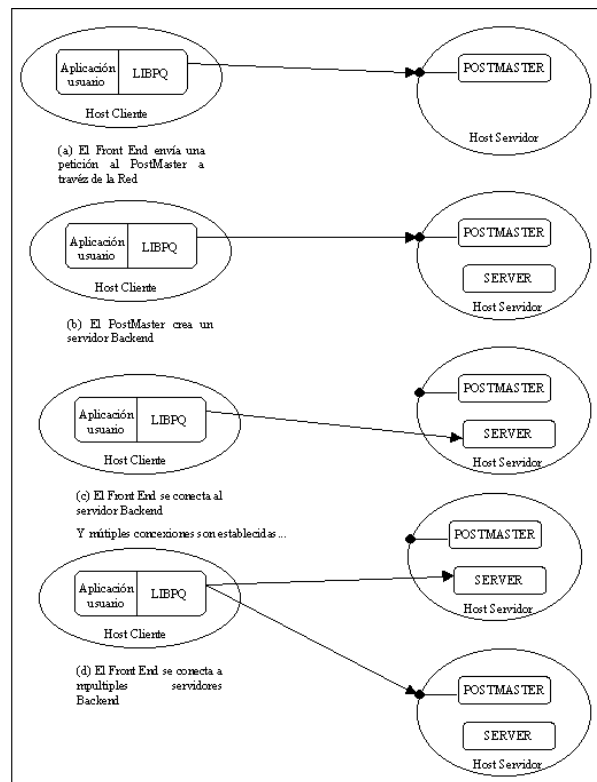


Fig. 2.1. Cómo se establece una conexión de una Aplicación FrontEnd con el servidor PostgreSQL.

Manual de Usuario de PostgreSQL

y se conecta el proceso cliente al nuevo servidor (Cómo se establece una conexión(c)). Desde aquí en adelante, el proceso cliente y el servidor se comunican entre ellos sin intervención del postmaster. En consecuencia, el proceso postmaster está siempre corriendo, esperando llamadas, mientras que los procesos cliente y servidor vienen y van. La librería libpq permite a un único proceso cliente tener múltiples conexiones con procesos servidores. Sin embargo, la aplicación cliente sigue siendo un proceso mono-enlace. Las conexiones con multi-enlace cliente / servidor no están actualmente soportadas en libpq. Una implicación de esta arquitectura es que el postmaster y los servidores siempre corren en la misma máquina (el servidor de base de datos), mientras que el cliente puede correr en cualquier sitio. Se debe tomar en cuenta esto, ya que los archivos que pueden estar accesibles en una máquina cliente, pueden no



estarlo (o estarlo sólo con un nombre de fichero diferente) en la máquina servidor, Además también se debe dar cuenta que postmaster y los servidores postgres corren bajo el user-id del "superusuario" de Postgres. Nótese que el superusuario de Postgres no tiene porqué ser un usuario especial (es decir, un usuario llamado "postgres"), aunque en muchos sistemas esté instalado así. Más aún, el superusuario de Postgres definitivamente no debe de ser el superusuario de Unix, "root". En cualquier caso, todos los archivos relacionados con una base de datos deben encontrarse bajo este superusuario de Postgres.

b) Funciones

PostgreSQL ofrece una potencia adicional sustancial al incorporar los siguientes cuatro conceptos adicionales básicos en una vía en la que los usuarios pueden extender fácilmente el sistema:

Clases
Herencia
Tipos
Funciones

Otras características aportan potencia y flexibilidad adicional:

Restricciones (Constraints)

Disparadores (triggers)

Reglas

Integridad transaccional



Estas características colocan a PostgreSQL en la categoría de las Bases de Datos identificadas como *objeto-relacionales*. Nótese que éstas son diferentes de las referidas como *orientadas a objetos*, que en general no son bien aprovechables para soportar lenguajes de Bases de Datos relacionales tradicionales. De hecho, algunas Bases de Datos comerciales han incorporado recientemente características en las que PostgreSQL fue pionera.

Las potenciaciones mayores en PostgreSQL incluyen:

- ✓ El bloqueo a nivel de tabla tiene un control de concurrencia multiversión, que permite a los lectores continuar leyendo datos consistentes durante la actividad de escritura y capacita copias con `pg_dump` mientras la base de datos permanece utilizable para las consultas.
- ✓ Tiene importantes características de servidor, incluyendo subselects, valores de defecto, restricciones (constraints) y disparadores (triggers).
- ✓ Además tienen características adicionales de lenguaje que cumplen SQL92, incluyendo claves primarias, identificadores citados, obligación de tipos de cadenas de literales, unión de tipos, y entrada de enteros binarios y hexadecimales.
- ✓ Los tipos previamente construidos han sido incrementados, incluyendo nuevos tipos `date` / `time` de amplio rango, y soporte adicional de tipo geométrico.
- ✓ La velocidad del rendimiento del código del servidor ha sido incrementado aproximadamente en un 20-40%, y el tiempo de arranque del servidor se ha reducido en un 80% desde que se liberó la Versión 6.0.

c) Seguridad

La seguridad de la base de datos esta implementada en varios niveles:



Protección de los archivos de la base de datos. Todos los archivos almacenados en la base de datos están protegidos contra escritura por cualquier cuenta que no sea la del superusuario de Postgres.

Las conexiones de los clientes al servidor de la base de datos están permitidas, por defecto, únicamente mediante sockets Unix locales y no mediante sockets TCP/IP. Ha de inicializarse el demonio con la opción `-i` para permitir la conexión de clientes no locales.

Las conexiones de los clientes se pueden restringir por dirección IP y/o por nombre de usuario mediante el fichero `pg_hba.conf` situado en `PG_DATA`. Las conexiones de los clientes pueden ser autenticadas mediante otros paquetes externos.

A cada usuario de Postgres se le asigna un nombre de usuario y (opcionalmente) una contraseña. Por defecto, los usuarios no tienen permiso de escritura a bases de datos que no hayan creado.

Los usuarios pueden ser incluidos en *grupos*, y el acceso a las tablas puede restringirse en base a esos grupos.

d) Lenguaje de Consulta

PostgreSQL, se basa en un modelo de datos relacional, una BDD es una *colección de tablas*. Una tabla consiste en filas y columnas, en la que cada fila representa un registro, y cada columna representa un atributo del registro contenido en la tabla. PostgreSQL es pionera en muchas características objeto-relacional que están disponibles ahora en algunas SGBDD comerciales de la actualidad. Tiene soporte del lenguaje estándar SQL92 y el actual SQL3, con todas sus extensiones.



e) Tipos de Datos Soportados

Datos Numéricos

Tipos numéricos	Almacenaje	Descripción	Rango
float4	4 bytes	Precisión Variable	6 posiciones decimales
float8	8 bytes	Precisión Variable	15 posiciones decimales
int2	2 bytes	Precisión Fija	-32768 a +32767
int4	4 bytes	Preferida usualmente para precisión fija	-2147483648 a +2147483647
int8	8 bytes	Precisión fija de rango extenso	+/- > 18 posiciones decimales
Serial	4 bytes	Secuencia	0 a +2147483647

El tipo de SERIAL es un caso especial construido por Postgres de otros componentes existentes. Es usado típicamente para crear un identificador único para valores de tablas secuenciales. El siguiente ejemplo creará una tabla tablename con un campo colname, el mismo que tomará valores secuenciales automáticos (como el tipo de dato auto numérico de ACCESS):

```
CREATE TABLE tablename (colname SERIAL);
```

Tipo de Datos Carácter

SQL92 define dos tipos carácter básico: char and varchar. Postgres también soporta el tipo text.

Tipo Carácter	Almacenaje	Recomendación	Descripción
Char	1 byte	SQL92-compatible	Carácter simple
char(n)	(4+n) bytes	SQL92-compatible	Longitud fija
Text	(4+x) bytes	Mayormente utilizado	Longitud variable
Varchar(n)	(4+n) bytes	SQL92-compatible	Longitud variable con límite



Tipos de Datos Date / time

Tipo de datos Date / time	Almacenaje	Recomendación	Descripción
Date	4 bytes	SQL92 type	Rango amplio
Datetime	8 bytes	Mayormente usado	Rango amplio con precisión
Time	4 bytes	SQL92 type	Rango amplio
Timestamp	4 bytes	SQL92 type	Rango limitado

Otros Tipos de Datos

Tipo de datos	Almacenaje	Rango
Money	4 bytes	-21474836.48 to +21474836.47
Boolean	1 byte	't' o 'f'

f) Estudio de conectividad con LINUX

Instalación

PostgreSQL se distribuye en tres tipos de paquetes: binaria, RPM y DEB. Para la distribución binaria es necesario volver a compilar todo, realizando ciertas modificaciones en el archivo Makefile (archivo de configuración, que indica en donde se encuentran localizados los archivos cabeceras de librerías que se necesitará para la instalación y ejecución del programa) por lo que no es aconsejable, DEB es un paquete desarrollado por la distribución Linux de Debian, pero RedHat no dispone del programa que permite trabajar con este, el paquete RPM es propio de RedHat Linux, siendo esta forma la mejor forma de instalar de PostgreSQL. Antes de iniciar la instalación se debe crear una cuenta de usuario a nivel del Sistema Operativo denominado postgres, luego ejecutar el programa GnoRPM (en el ambiente GNOME de RedHat Linux) o KpackageRPM (en el ambiente K Desktop), instalar los archivos: *postgresql-server-7.1.3-3.i386.rpm* para el servidor y *postgresql-7.1.3-3.i386.rpm* para el cliente. Existe también archivos para el desarrollo de aplicaciones cliente con lenguajes Tcl/Tk, Perl o interfaces JDBC, ODBC propias de PostgreSQL.

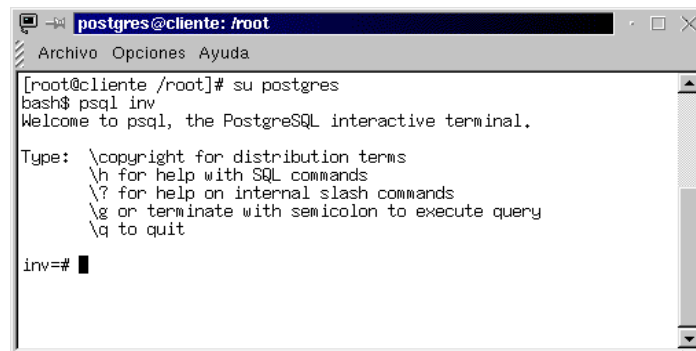


Servidor de Base de Datos

El servidor PostgreSQL es el programa **postgres** (ver sección 2.2.1.4) que se ejecuta en backend, este servicio se inicializa al inicio de la ejecución de RedHat Linux por lo que se debe activarlo con ayuda del programa *ntsysv* o *linuxconf*.

Monitor SQL

El monitor cliente SQL es el programa interactivo **psql**; para ejecutarlo es necesario hacerlo como el usuario postgres, no como root. Una vez hecho esto digitar en el shell **psql** mas el nombre de la Base de Datos, por lo que presentará lo siguiente:



```
postgres@cliente: /root
Archivo Opciones Ayuda
[root@cliente /root]# su postgres
bash$ psql inv
Welcome to psql, the PostgreSQL interactive terminal.

Type: \copyright for distribution terms
      \h for help with SQL commands
      \? for help on internal slash commands
      \g or terminate with semicolon to execute query
      \q to quit

inv=#
```

Fig. 2.2. Programa Monitor SQL de PostgreSQL

Este prompt indica que el monitor está listo y puede escribir sus consultas SQL dentro de un espacio de trabajo mantenido por el monitor. El programa **psql** responde a los códigos de escape que empiezan por el caracter "\". Por ejemplo, puede obtener la ayuda acerca de la sintaxis de varias órdenes SQL Postgres escribiendo:

```
mibdd=> \h
```

Una vez que haya terminado de introducir consultas, puede pasar el contenido del espacio de trabajo al servidor Postgres escribiendo:

```
mibdd=> \g
```

Esto le dice al servidor que procese la consulta. Si termina su consulta con un punto y coma, la "\g" no es necesaria. psql procesará automáticamente las consultas terminadas



con punto y coma. Para leer consultas desde un archivo, digamos miarchivo.txt, en lugar de introducirlas interactivamente, escriba:

```
mibdd=> \i nombreDelFichero
```

Para salir de psql y regresar a Unix escriba:

```
mibdd=> \q
```

y psql terminará y volverá a la línea de órdenes.

Desarrollo de aplicaciones cliente

PostgreSQL ofrece librerías API para los lenguajes Tcl/Tk, C/C++, Perl, Python, además de lenguajes procedurales propios de PostgreSQL como son: PL/PgSQL y PL/Tcl. Más importante aún es el programa Pgaccess, que es una interfaz gráfica para el desarrollo de consultas, diseño de reportes, desarrollo de frontend al propio estilo de Visual Basic y MS Access, añadiendo y quitando objetos, relacionando eventos a estos, etc.

Para el desarrollo de aplicaciones cliente bajo Windows 9x y NT ofrece interfaces ODBC y JDBC que permiten la conexión con el servidor postgres que corre en una máquina Linux, con esto poder desarrollar el frontend en Visual Basic, Access, Delphi o en otros lenguajes que soporten ODBC. Además existe un programa cliente para Windows con el cual puede conectarse con el servidor PostgreSQL, limitado solamente a ejecutar consultas y realizar comandos de administración de la base de datos.

2.2.2. INTERBASE



2.2.2.1 Generalidades

InterBase es un recomendable Administrador de Bases de Datos Relacional SQL un producto de InterBase Software Corporation. InterBase combina una sofisticada



tecnología de Base de datos con una incomparable facilidad de uso. En una configuración cliente / servidor, InterBase para Linux es compatible con versiones comerciales de Interbase en tres plataformas.

2.2.2.2 Requerimientos de Software y Hardware.

El servidor de InterBase se puede ejecutarse en una gran variedad de plataformas, incluyendo Microsoft Window NT 4.0 y Windows 9X, como también varias marcas de Sistemas Operativos UNIX.

Requerimientos para Sistemas Windows

- Sistema Operativo: Windows NT 4.0 o Windows 9x
- Memoria: 16 MB mínimo; 64 MB recomendable para un servidor
- Tipo del Procesador: 486DX2 66MHz; Pentium 100MHz o superior recomendado para un servidor multi-cliente
- Compiladores: Microsoft Visual C++ 4.2 y Borland C++ 5.0

Requerimientos para Sistemas UNIX

- Memoria: 32 MB mínimo; 64 MB recomendable para un servidor
- Tipo del Procesador: RISC
- Compiladores: C++

Requerimientos para Sistemas SOLARIS

- Sistema Operativo: Solaris 2.5.x o 2.6.x
- Memoria: 32 MB mínimo; 64 MB recomendable para un servidor
- Tipo del Procesador: SPARC o UltraSPARC
- Compiladores: SPARCWorks SC 4.2



Requerimientos para Sistemas LINUX

- Sistema Operativo: Kernel 2.0.x
- Memoria: 32 MB mínimo; 64 MB recomendable para un servidor
- Tipo del Procesador: Intel Pentium y Pentium II y III con un mínimo de 60 MHZ en velocidad de reloj.
- Compiladores: GNU gcc 2.7.2.1

2.2.2.3 Características Generales

- 🔒 **Versión(Septiembre del 2000):** 6.0
- 🔒 **Empresa:** Interbase Software Corporation
- 🔒 **Licencia:** Gratuita
- 🔒 **Home Site:**
<http://www.interbase.com/>
- 🔒 **Disponibilidad del Código Fuente:** Sí
- 🔒 **Paquetes Binarios:**
 - Debian Package: No
 - RedHat RPM Package: Sí (para la versión 4.0)
- 🔒 **Otros Enlaces:**
<http://www.interbase.com/open/downloads>
<ftp://ftp2.interbase.com/pub/products/InterClient>
http://www.interbase.com/products/IB_overview.html
- 🔒 **Listas de Correos / Grupos USENET News:**
<http://www.interbase.com/open/community/groups.html>



2.2.2.4 Características Técnicas

a) Arquitectura

El diseño de la arquitectura de InterBase está basado en procesos. Para la conexión de varios clientes, el Servidor inicializa un proceso separado y es ejecutado en el motor de Base de Datos, cada proceso Servidor tiene un caché dedicado. Los procesos del Servidor compiten por acceder a la Base de Datos, además existe un subsistema Administrador de Seguridad (Lock Manager), para decidir y sincronizar el acceso de entre los procesos.

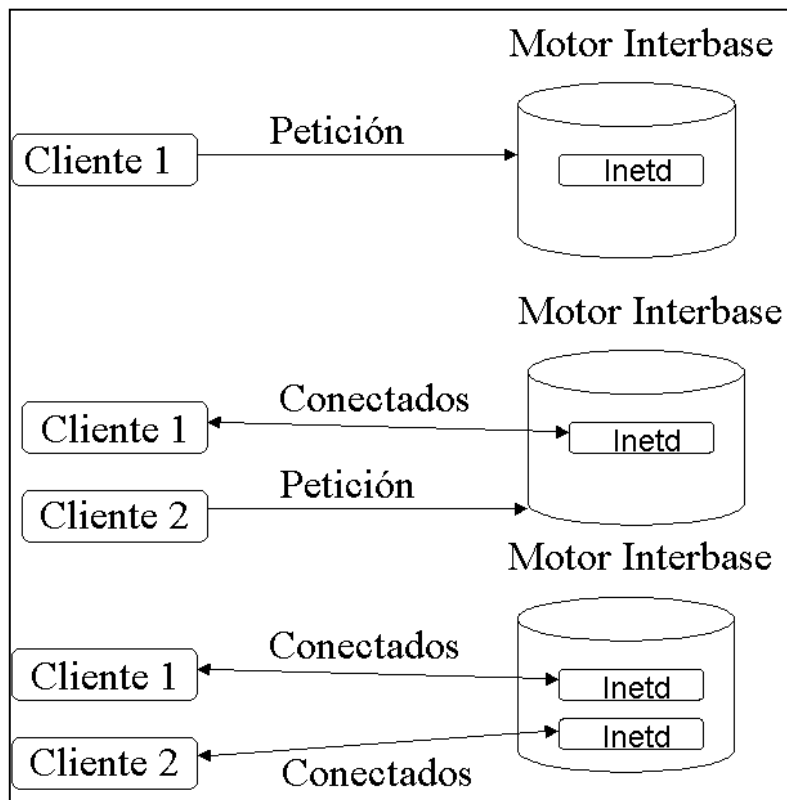


Fig. 2.3. Arquitectura Interbase

Función del Servidor

El Servidor ejecuta una demanda para múltiples procesos. Cuando un cliente intenta conectarse a la Base de Datos InterBase, asigna una instancia (estructura de



memoria llamada área global del sistema) y se ejecuta el proceso (programa) `gds_inet_server`, permaneciendo activa mientras dure la conexión del cliente.

El iniciador de `gds_inet_server` es `inetd`, servicio UNIX. Este contiene un archivo de configuración, `/etc/inetd.conf`, estos servicios están asociados con el ejecutable que recibe la conexión. Cuando `inetd` recibe una demanda de conexión por un servicio dado, este busca el programa apropiado en `/etc/inetd.conf`, lo ejecuta, y transfiere a través de la conexión de la red el programa de servicio respectivo.

Cuando el cliente se desconecta, `gds_inet_server` cierra la conexión de la base de datos y algún otro archivo existente. Cuando no hay clientes conectados a alguna base de datos, la ejecución de `gds_inet_server` no debe ser invocado.

Administrador de Seguridad (Lock Management)

Lock Management está dado por otro proceso, el `gds_lock_mgr`. Este programa es inicializado cuando un segundo cliente se conecta a una determinada base de datos. La tarea del Lock Manager es servir (metafóricamente) como un policía de tránsito. Este concede las bases de datos disponibles a los clientes. Además rechaza las peticiones de las bases de datos de los clientes que están siendo usadas por otros. El `gds_lock_mgr` permanece activa aún después de la desconexión del último cliente. De esta manera, en la siguiente ocasión en que se conecte un cliente, puede evitar un pequeño costo operativo de inicialización del proceso Lock Manager.

Uso de señales Posix

El proceso `gds_lock_mgr` comunica con cada proceso de cliente a través del uso de una área de memoria compartida, y un mecanismo de señalización usando las señales



POSIX, SIGUSR1 y SIGUSR2. Estas señales son tomadas mediante rutinas de manipulación de señales que se encuentran en el archivo *libgdslib.a*, y es por esta razón que las aplicaciones de los usuarios no pueden manipular estas señales o realizar alguna modificación a las señales ocultas. Las aplicaciones que necesiten las señales POSIX deben compilar con una librería alterna de InterBase, *libgds.a*. Pero este envía a través de Lock Manager, un pequeño proceso llamado **gds_pipe**. Todas las aplicaciones de los clientes que compilaron con *libgds.a* automáticamente ejecutan este pequeño proceso.

Uso de Recursos

Cada instancia del proceso **gds_inet_server** mantiene una caché de espacio de memoria para cada base de datos, lo que probablemente resulte alguna duplicación de datos ocultos a través del sistema. El servidor usa pequeños conjuntos de recursos cuando las conexiones concurrentes son bajas.

Monitorización de las conexiones de las Bases de Datos

La información de las Bases de Datos siempre exige conexiones activas como si fuera una única conexión, sin importar cuántos clientes estén conectados al servidor. La razón para esto es que la conexión de muchos clientes tienen su propio proceso **gds_inet_server** en el servidor, y cada instancia del programa solo conoce su propia conexión.

b) Funciones

InterBase facilita a los *metadatos activos* (ver sección 2.2.2.4) acceder al servidor de la base de datos para centralizar las tareas de:

- Seguridad
- Integridad de los datos



- Reducción del consumo de recursos de la red.

Las características que ofrece InterBase para los metadatos activos son:

- Triggers
- Procedimientos de almacenamiento
- Procedimientos SELECT
- Integridad Referencial en cascada (usando PRIMARY KEY y FOREIGN KEY)
- Privilegios SQL
- Funciones definidos por el usuario (UDFs)
- Eventos

c) Seguridades

InterBase trabaja con una mezcla de clientes locales y remotos ejecutando como diferentes usuarios, Los procesos del servidor `gds_inet_server` y `gds_lock_mgr` deben correr como root con una identificación UID (Número de Identificación de usuario en una Red) real. Cada proceso debe correr con una UID signado por el root. El proceso Lock Manager debe tener privilegios de superusuario para enviar señales a los procesos.

En algunos ambientes, la presencia de procesos ejecutables con identificación UID activados, preocupan por la inseguridad que esto causa. No obstante, no cambia la configuración del tiempo de activación del servidor InterBase.

Puesto que las aplicaciones pueden ejecutarse con alguna UID, los archivos de las Bases de Datos deben tener permiso de escritura para todas las UIDs que acceden la base de datos. Para simplificar el mantenimiento, los archivos de las bases de Datos son creados con permisos de escritura para todos. Tenga cuidado, puede restringir los



permisos de los archivos, a fin de que los archivos de Bases de Datos estén seguros de un daño accidental o deliberado. Asegúrese de comprender bien los permisos de archivos antes de intentar esto, porque todos los clientes locales y remotos necesitan tener acceso de escritura a las Bases de Datos, aún si ellos pretenden solamente leer los datos.

d) Lenguaje de Consulta

Una Base de datos InterBase es creada usando las sentencias SQL, la cual puede ser dividida en dos categorías: sentencias Data Definition Language (DDL) y sentencias Data Manipulation Language (DML).

La estructura fundamental de la Base de datos (tablas, vistas, e indexaciones) son creados por las sentencias DDL. Generalmente, los objetos definidos con las sentencias DDL son conocidos con el nombre de metadato. La definición de datos es el proceso de creación, modificación y eliminación de metadatos. Contrariamente, las sentencias DML son usadas para el llenado de la Base de datos de información, y manipular los datos almacenados en la estructura previamente definido con las sentencias DDL.

Las sentencias DDL para crear un metadato, inicia con la palabra reservada CREATE, para modificar se utiliza ALTER, y la sentencia de eliminación inicia con DROP. Algunos ejemplos tareas de definición básicas son:

Creación de Base de datos (CREATE DATABASE).

Creación de tablas (CREATE TABLE).

Modificación de tablas (ALTER TABLE).

Eliminación de tablas (DROP TABLE).



En InterBase, un metadato es almacenado en un sistema de tablas, los mismos que son agrupados automáticamente cuando una base de datos es creada. Estas tablas guardan información acerca de la estructura de la base de datos. Todos los sistemas de tablas inician con “RDB\$”. Por ejemplo la tabla de sistema RDB\$RELATIONS, con información de cada tabla que forma parte de la base de datos y RDB\$FIELDS, con información de los dominios (Dominios son usados cuando varias tablas contienen definiciones idénticas dentro de una base de datos) de la base.

Creación de dominios

Cuando se crea una tabla, se puede usar la definición de una columna global, llamado **dominio**, antes de definir una columna debe hacerse referencia a un dominio, para hacer esto primero se debe crear la definición de dominio usando CREATE DOMAIN, que actúa como una plantilla para la definición de columnas. La sintaxis es:

```
CREATE DOMAIN domain [AS] <datatype>
[DEFAULT { literal | NULL | USER}]
[NOT NULL] [CHECK ( <dom_search_condition>)]
[COLLATE collation];
```

e) Tipos de Datos Soportados

Los tipos de datos a usar dependen de la plataforma donde se está trabajando. InterBase no tiene tipos de datos nativos. Sin embargo su implementación no es dificultosa. InterBase brinda el uso de dominios. Y el primer paso es definir un dominio booleano. El segundo paso es definir el tipo del dominio. Estas son dos formas, cada uno con sus propias ventajas. De preferencia, se usa el tipo Smallint (16 bits), con valor por defecto de cero, con comprobación para asegurar que tome los valores de cero y uno, por ejemplo:



CREATE DOMAIN D_BOOLEAN AS SMALLINT DEFAULT 0 CHECK (VALUE BETWEEN 0 AND 1);

Una vez definido este dominio, se puede usar en forma permanente como si fuera un tipo booleano sin preocupación. Esta forma es similar desde el punto de vista de los datos booleano de Delphi y Pascal.

InterBase soporta los siguientes tipos de datos:

INTEGER Y SMALLINT

FLOAT Y DOUBLE PRECISION

NUMERIC Y DECIMAL

DATE

CARÁCTER Y VARYING CARÁCTER

BLOB

InterBase provee el tipo de dato **Binary Large Object (BLOB)**, es utilizado para almacenar datos y objetos de tamaño indeterminado.

Tipo de Dato	Descripción	Rango
CHAR(n)	Longitud variable, palabra opcional CHARACTER	1 a 32767 bytes
BLOB	Binary large object; almacena datos largos como gráficos, texto, voz digitalizada.	Ninguno, segmento de longitud limitado a 64K
INTEGER	Entero numérico de 32 bits	-2,147,483,648 a 2,147,483,647
SMALLINT	Entero numérico de 16 bits	Precisión: -32768 a 32767
Double precisión	Flotante de 64 bits. 15 dígitos	1.7×10^{-308} to 1.7×10^{308} y s: $0 < s < p$
FLOAT	Simple precision. 32 bits	3.4×10^{-38} a 3.4×10^{38}
DATE	También incluye información del tiempo.	1 Ene 100 a.c. a 29 Feb 32768 d.c.



f) Estudio de conectividad con LINUX

Instalación

InterBase se distribuye en paquetes: binario y RPM, como en el caso anterior es conveniente utilizar el paquete RPM *interbase-6.0-1.i386.rpm*, el mismo que genera los archivos necesarios para la ejecución del servidor InterBase. No necesita de un usuario especial.

Servidor de Base de Datos

El servidor de Interbase es el programa *ibserver* que se ejecuta en backend, este servicio no se inicializa al inicio de la ejecución de RedHat Linux por lo que se debe activarlo, ejecutando el programa localizado en */usr/local/interbase/bin* o activarlo con ayuda del programa *ntsysv* o *linuxconf*.

Monitor SQL

El monitor en InterBase es el programa *isql* el mismo que ofrece una comunicación interactiva del usuario con el servidor InterBase, una vez digitado en el shell de Linux aparecerá lo siguiente:

```
root@cliente: /usr/local/interbase/bin
Archivo Opciones Ayuda
[root@cliente bin]# pwd
/usr/local/interbase/bin
[root@cliente bin]# ./isql
Use CONNECT or CREATE DATABASE to specify a database
SQL>
```

Fig. 2.4. Programa Monitor SQL de InterBase

Esto indica que el monitor está listo para recibir y ejecutar las sentencias SQL para trabajar con la base de datos del usuario. Para salir del programa se digita *exit*; o *ctrl.+d*.



Desarrollo de aplicaciones cliente

InterBase enlaza aplicaciones de bases de datos usando las aplicaciones API (Applications Programming Interface) que son archivos (DLL en Windows, .H en Linux) que contienen librerías de funciones para conectar bases de datos con lenguajes de programación como host ya sea C/C++, Perl, Tcl, etc. En el subdirectorio include se encuentra la cabecera de librería ***ibase.h*** el mismo que proporciona todas las funciones requeridas para la conexión del servidor InterBase con el lenguaje de programación C/C++.

Como se puede observar InterBase sólo ofrece APIs para C/C++ por lo que reduce la capacidad de enlazarse con otros lenguajes propios de Linux que ofrecen mayor facilidad para desarrollar aplicaciones frontend como Tcl/Tk, Perl y otros.

InterBase fue desarrollado también para correr como servidor y cliente bajo Windows 9x, ofreciendo así una mejor prestación de servicios con lo que respecta en el desarrollo de aplicaciones cliente utilizando los lenguajes nativos para Windows como Visual Basic, Visual C++, etc. La diferencia está en que para esta plataforma no es gratuita y tiene un costo por su empleo.

2.2.3. MIMER RDBMS



2.2.3.1 Generalidades

MIMER es un Sistema de Gestión de Bases de Datos (SGBDD) desarrollado por Sysdeco MIMER. Es un motor de Bases de Datos de alto desempeño. MIMER ofrece portabilidad, incluyendo soporte de multiprocesamiento. Es ideal para corporaciones abiertas donde la interoperabilidad es importante.



Las características de ejecución de MIMER: fácil uso, estabilidad y autoconfiguración, hace de este una alternativa para el desarrollo de productos de Bases de Datos para Internet/Intranet/Extranet donde se requiere lo máximo en SGBDD. Está conformado 100% del SQL estándar e implementaciones nativas de interfaz ODBC de Microsoft, una base de datos puede ser accesado por un gran número de diferentes herramientas de desarrollo.

2.2.3.2 Requerimientos de Software y Hardware.

MIMER RDBMS es soportado por la mayoría de Sistemas Operativos UNIX (Linux, FreeBSD), Windows y plataformas VMS.

La lista siguiente muestra el hardware y software soportado:

- Intel / Windows NT
- Intel / Windows 95/98
- Intel / SCO
- IBM RS6000 / AIX
- Hewlett-Packard / HP-UX
- SUN / Solaris
- Data General / DG-UX
- Alpha AXP / Digital UNIX
- Alpha AXP / Windows NT
- Alpha AXP / OpenVMS
- Macintosh (solo como cliente)

2.2.3.3 Características Generales

🔗 **Versión(Septiembre del 2001):** 8.2

🔗 **Empresa:** Sysdeco Mimer AB en Uppsala, Sweden.



- ☞ **Licencia:** Gratuita y Comercial

- ☞ **Home Site:**
<http://www.mimer.com/>

- ☞ **Disponibilidad del Código Fuente:** No

- ☞ **Paquetes Binarios:**
 - Debian Package: No

 - RedHat RPM Package: No

- ☞ **Otros Enlaces:**
Ninguno

- ☞ **Listas de Correos / Grupos USENET News:**
info@mimer.se
Web master webmaster@mimer.se

2.2.3.4 Características Técnicas

a) Arquitectura

MIMER SGBDD está basado en una arquitectura cliente / servidor. El servidor de Base de datos se ejecuta en un proceso individual multi-enlazado con múltiples **enlaces Request y Background**. En algunas plataformas, se usan uno o más **enlaces de comunicación**. La arquitectura de MIMER posee multienlace real, con demandas dinámicamente asignadas a diferentes accesos Request. Trabaja muy bien el acceso escalar sobre múltiples CPUs, MIMER es particularmente muy conveniente para ambientes con Multiprocesadores Simétricos. Por el uso de accesos sin el Servidor de Bases de Datos, esto lo logra de manera óptima y eficiente con el entorno intercambiable en el Servidor de Base de Datos.



El uso de un proceso Servidor de Base de Datos separado garantiza que el error de un programa en un proceso de aplicación no pueda dañar el área de dato compartido controlado por el Servidor de Base de Datos.

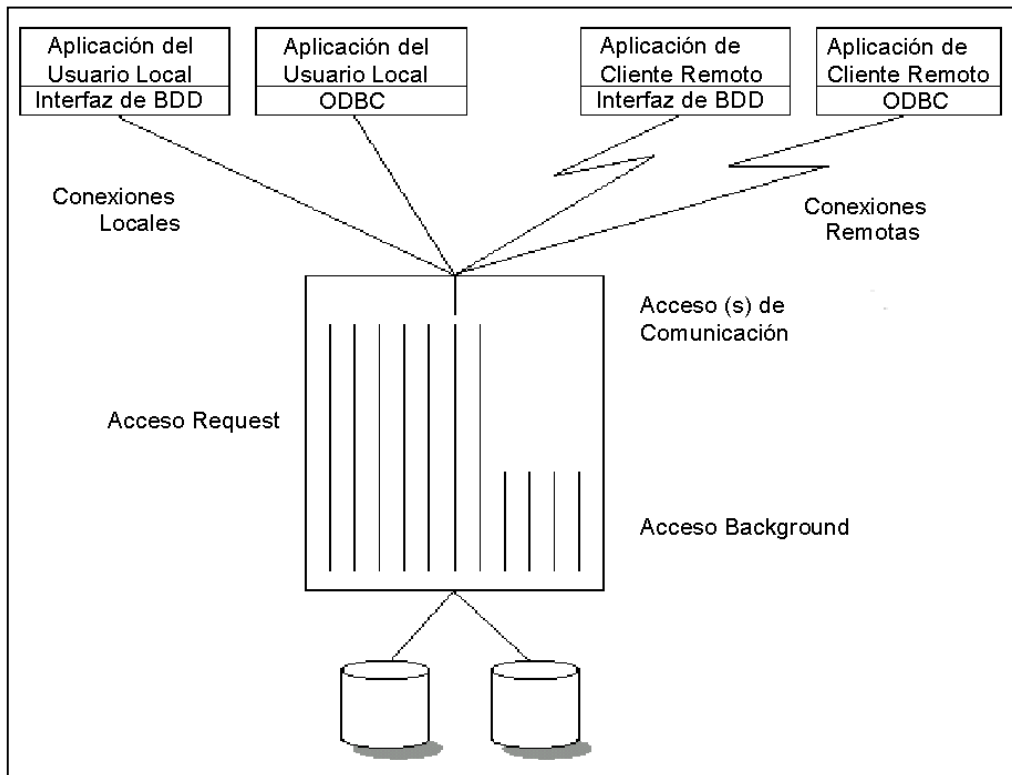


Fig. 2.5. Arquitectura del Servidor de Bases de Datos MIMER. Manual de Usuario MIMER

El o los accesos de comunicación en algunos de las plataformas MIMER manejan partes de la comunicación entre las aplicaciones y el Servidor de Base de Datos. Esto es una implementación definida si los accesos de comunicación son usados o no en una plataforma específica como y también si uno o más accesos de comunicación son usados. En algunas plataformas son utilizados otros mecanismos de manejo de comunicación entre las aplicaciones y el Servidor de Bases de Datos.

Tanto las aplicaciones locales como las remotas son manejadas directamente por el Servidor de Base de Datos. Este método es aplicable en ambientes Cliente Servidor,



donde MIMER se ejecuta en un ambiente distribuido con el Cliente y el Servidor en diferentes máquinas, todos los Clientes Remotos están conectados directamente al Servidor de Base de Datos. Por ende esto evita gastos adicionales de servicios de red para procesos existentes inicializados.

El **acceso Request** ejecuta las operaciones SQL demandadas por las aplicaciones, cuando el Servidor de Base de Datos requiere ejecutar una operación SQL, este asigna espacio a uno de estos accesos Request que ejecutara la tarea. Cuando la operación SQL es completada el resultado es devuelto a la aplicación y el acceso Request que ejecuto la operación retorna a un estado de espera hasta recibir otra demanda del Servidor. Puesto que las operaciones SQL son evaluadas enteramente sin la intervención del Servidor de Base de Datos, la comunicación entre proceso es reducido al mínimo.

Cuando una consulta SQL o un procedimiento de almacenaje es ejecutado por un acceso Request, la versión compilada de la consulta o del procedimiento es almacenada por la intervención del Servidor de Base de Datos. En esta forma la misma versión compilada de la consulta o el procedimiento puede ser usado nuevamente por otras aplicaciones. Esto marca la ejecución improvisada, desde una consulta SQL o un procedimiento de almacenamiento solo necesita ser compilado una vez por el Servidor de Base de Datos.

El **acceso Background** ejecuta cada servicio de Bases de Datos como una transacción completa aplicando la técnica Shadowing. Este servicio es ejecutado sincrónicamente en el fondo del proceso de la aplicación, de manera que el proceso de la aplicación no tiene que esperar para la terminación física de una transacción o una actualización, pero puede continuar tan pronto como la transacción haya sido preparada y asignada al disco.



Las operaciones de Entrada y salida son ejecutadas directamente en paralelo por el Servidor y accesos espías usando sincronizaciones I/O.

b) Capacidades

Sysdeco Mimer está comprometido a adaptarse a las bases de datos relacionales estándares definido por las organizaciones como: ANSI, ISO Y X/Open. Con el driver nativo MIMER para interfaces ODBC de Microsoft, un gran número de herramientas para la Web y para Windows pueden ser usados conjuntamente con MIMER (EJ. Tango, Delphi, Visual Basic, Access, Centura y muchos más). Mimer también se adapta fácilmente con muchos productos middleware basados en CORBA y OTM. La interfaz ODBC en MIMER está implementada al mismo nivel de la interfaz SQL, proveyendo un excelente desarrollo cuando el ODBC es usado con MIMER.

MIMER 8.2 es también el primer RDBMS que posee soporte ISO estándar con procedimientos SQL/PSM, así se puede incrustar una interfaz para C, COBOL y FORTRAN, de esta manera es posible desarrollar aplicaciones 3GL (tercera generación) portátiles.

c) Seguridades

- 🔒 Seguridad avanzada. Facilita un control de acceso restringido, usando una diferenciación de aplicaciones.
- 🔒 Monitoreo de privilegios de acceso al administrador de base de datos.
- 🔒 Brinda un conjunto de controles de integridad para asegurar la consistencia lógica de los datos.



- 🗄 Operaciones de respaldo y recuperación, garantizando la consistencia y actualización de la información que forma parte la base de datos, siempre podrán ser restaurados después de una falla en el disco.
- 🗄 Utilidad de seguimiento que provee información sobre el desempeño de las transacciones en el sistema.

El sistema de seguridad fine-grained que provee MIMER permite a los datos ser protegidos bajo un único elemento (fila / columna) para cada usuario que está accediendo al sistema. Una única característica de MIMER es el “role concept”, donde el acceso correcto por parte de un usuario puede estar bajo la protección de un password. El role-concept concede un sistema de seguridad que distingue a cada usuario para el acceso a la base de datos controlado por la aplicación. El role-concept provee MIMER a través del programa ***PROGRAM ident***.

La integridad de una base de datos MIMER, son asegurados a través de:

- 🗄 Cuatro diferentes tipos de autorizaciones Idents
- 🗄 Sistema de privilegios, restringiendo el uso de las funciones del administrador de la base de datos.
- 🗄 Views
- 🗄 Dominios
- 🗄 Integridad referencial (foreign keys)
- 🗄 USER idents – idents de usuario, generalmente asociados con autorizaciones individuales específicas, para uso del sistema.
- 🗄 OS_USER idents – Un tipo de User ident, con el mismo nombre de usuario que da el sistema operativo. Si un usuario es definido en el sistema operativo es también definido por MIMER como un OS_USER, de esta manera el usuario no necesitará proveer un nombre y password adicional en MIMER.



- GROUP Idents – son identificaciones de grupos de idents. Sus privilegios son también aplicados a todos los que pertenecen al grupo.

d) Lenguaje de Consulta

MIMER ofrece un SQL basado a los estándares establecidos, estas son:

- ISO SQL1 (o también llamado SQL89)
- ISO SQL2 (o SQL92)
- ISO SQL Persistent Stored Modules (SQL/PSM)
- XPG3 SQL
- X/Open SQL 1992 (definido como X/Open-92)
- X/Open SQL 1995

MIMER adiciona, la Application Programming Interface ODBC de Microsoft, ha sido ampliamente aceptado como un estándar de facto.

MIMER combina mucho de las ventajas de los diferentes estándares:

- ISO SQL1
- ISO SQL2
- ISO SQL/PSM
- XPG3 SQL
- X/Open SQL 1992
- ODBC nivel 2 API, Extended SQL

Así forma su lenguaje SQL denominado MIMER/SQL.



e) Tipos de Datos Soportados

Los tipos de datos explícitos son en base a las sentencias SQL para la creación de una tabla por el usuario. Los tipos de datos permitidos por MIMER son los siguientes:

Tipo de Dato	Descripción	Rango
CARÁCTER(n)	Cadena de caracteres de longitud fija	Precisión de n: $1 < n < 15000$
CHARACTER	Cadena de caracteres de longitud variable	
VARYING(n)	Longitud limitada	Precisión de n: $1 < n < 15000$
INTEGER(p)	Entero numérico	Precisión de p: $1 < p < 45$
SMALLINT	Entero numérico	Precisión: -32768 a 32767
INTEGER	Entero numérico	Precisión: -2,147,483,648 a 2,147,483,647
DECIMAL(p,s)	Numérico con decimales	Precisión de p: $1 < p < 45$ y s: $0 < s < p$
FLOAT(p)	Numérico redondeado	$1 < p < 45$ Precisión de la Mantisa p: Cero o valor absoluto 10^{-999} to 10^{+999}
DATETIME	Representa la descripción absoluta, descrita abajo dependiendo del subtipo *	

- DATETIME Puede ser uno de los siguientes subtipos: DATE (YYYY-MM-DD), TIME (HH:MM:SS[.]) o TIMESTAMP (YYYY-MM-DD HH:MM:SS[.sF]).

f) Estudio de conectividad con LINUX

Instalación

A diferencia de las anteriores MIMER posee un propio programa de instalación basado en modo texto y de fácil manejo, solo debe responderse a ciertas preguntas como donde instalar, en que directorios se van a colocar los archivos de las bases de datos, etc.



Servidor de Base de Datos

El servidor de MIMER se denomina MIMERDB el mismo que se ejecuta como un servicio más de Linux, es necesario que se inicie junto con Linux.

Monitor SQL

MIMER provee el programa monitor Batch SQL: **bsql**, este verifica el usuario que está ingresando al servidor mediante un password que se establece al crear la base de datos mediante el comando **dbinstall**, de esta manera, Mimer protege la integridad de los datos de personas particulares.

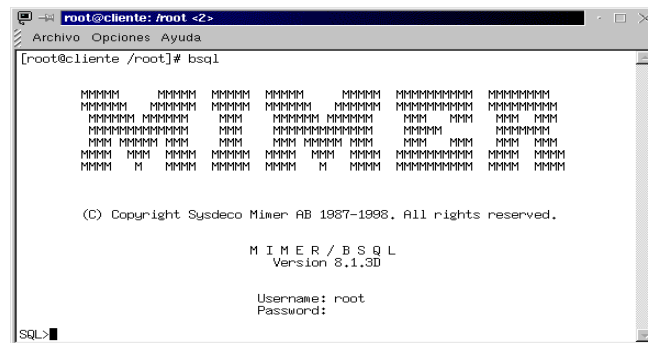


Fig. 2.6. Programa Monitor SQL de MIMER

Desarrollo de aplicaciones cliente

A través de ODBC, MIMER soporta muchas herramientas disponibles en las plataformas que lo soporte como Windows, Unis, Linux, etc.

MIMER además utiliza una forma diferente a todos los SGBDD estudiados para enlazar un lenguaje de programación con el motor de Base de Datos. Se puede enlazar con lenguajes C, COBOL o FORTRAN que se encuentren disponibles en la máquina, incrustando las sentencias SQL en el código fuente del programa de la aplicación, el mismo que es enlazado y compilado con el preprocesador de MIMER.



Preprocesamiento

Una aplicación que use las sentencias SQL incrustadas debe primero pasar por el preprocesador antes de ir al compilador del lenguaje (gcc para lenguaje C), puesto que por sí mismo no reconoce el identificador EXEC SQL (instrucción necesaria para ejecutar sentencias SQL). Existe un Preprocesador para cada uno de los lenguajes especificados anteriormente.

El preprocesador es esencialmente un traductor, el cual analiza gramaticalmente las sentencias SQL incrustadas y los convierte al código fuente apropiado para hacer las llamadas de las sentencias SQL al motor MIMER. Las sentencias SQL se mantienen en el código del programa original como comentarios. La salida del preprocesador es un código entendible para el usuario, manteniendo en gran parte la estructura del programa original.

El preprocesador es llamado por el siguiente comando:

```
$ esql lenguaje [ opciones] archivoentrada [ archivosalida]
```

```
lenguaje -c
```

```
-cob[ol]
```

```
-for[tran]
```

```
opciones -l[ineas] (genera #lineas, solo en lenguaje C)
```

Todas las sentencias SQL deben iniciar con el identificador EXEC SQL, por ejemplo:

```
EXEC SQL DELETE FROM INVENTARIO
```



2.2.4. MySQL



2.2.4.1. Generalidades

MySQL es un gestor de bases de datos SQL (Structured Query Language). Es una implementación Cliente-Servidor que consta de un servidor y diferentes clientes (programas / librerías).

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. Esta base de datos es considerada (en su propia documentación así lo reseña) como la más rápida y robusta tanto para volúmenes de datos grandes como pequeños (siempre, claro está, comparada con las de su categoría), aunque como veremos más adelante está rapidez es a costa de no implementar ciertos aspectos del SQL.

La versión libre fue escrita por Michael Widenis, la versión comercial es distribuida por TCX Datakonsulter AB.

2.2.4.2. Requerimientos de Software y Hardware.

Sistemas Operativos Soportados

MySQL está disponible (Fuentes y binario) para: Linux 2.0+, SCO, Solaris 2.5, 2.6, SUNOS 4.x, BSDI 2.x, 3.0, SGI IRIX 6.x, AIX 4.x, DEC UNIX 4.x, HP-UX 10.20, Windows 95, entre los sistemas operativos más populares.

Hardware

Sun Solaris 2.5 and 2.6, SUNOS, Linux 2.# (Intel and sparc), FreeBSD 2.#, FreeBSD 3.0, IBM AIX, SCO, Dec-Unix.



2.2.4.3. Características Generales

- ☞ **Versión(Septiembre del 2001):** 3.23.41
- ☞ **Empresa:** TCX Datakonsulter AB
- ☞ **Licencia:** GPL (Licencia Pública), gratuita
- ☞ **Home Site:** <http://www.mysql.com/>
<http://www.tcx.se/>
- ☞ **Disponibilidad del Código Fuente:** Sí
- ☞ **Paquetes Binarios:**
 - Debian Package: Sí, <http://www.debian.org/distrib/packages>
 - RedHat RPM Package: Sí, En su página web
- ☞ **Otros Enlaces:**

No posee
- ☞ **Listas de Correos / Grupos USENET News:**
<http://www.tcx.se/FAQ/mysql-faq.html#SEC3>

2.2.4.4 Características Técnicas

a) Arquitectura

MySQL es una implementación cliente / servidor que consiste en un demonio servidor "mysqld" que se ejecuta en segundo plano (background) en espera de las demandas del cliente, estas demandas se procesan entonces por el motor de BDD MySQL con la tecnología de multienlace (multithread), Multithread quiere decir que MySQL puede ocuparse de muchas tareas y demandas al mismo tiempo las conexiones de los clientes



conectadas al servidor, no necesita terminar los procesos o consultas de los clientes para atender a otros que requieran del servidor, esta respuesta rápida es cumplida a través del uso de un programa multithreaded que actúa como si muchas instancias del mismo programa están corriendo concurrentemente. Cuando un usuario se conecta al servidor MySQL, un nuevo proceso de la aplicación genera un enlace(thread) y el servidor accede a trabajar con este, en cualquier momento dado, un thread extra está corriendo siempre para manejar las demandas de otros enlaces de clientes. Si una conexión está requiriendo el acceso a una tabla ocupada ya por otra, el thread administrador, obligará a dicha conexión a esperar hasta que el otro proceso se haya completado.

Puesto que los threads individuales se ejecutan independiente uno del otro, pueden usarse máquinas multiprocesador para distribuir el trabajo de los threads hacia varios CPUs.

La base sobre la que está construida MySQL es un conjunto de rutinas que han sido utilizadas en un entorno de producción altamente exigente durante muchos años. Aunque MySQL sigue en desarrollo, proporciona en estos momentos un conjunto de funciones robusto y altamente útil.

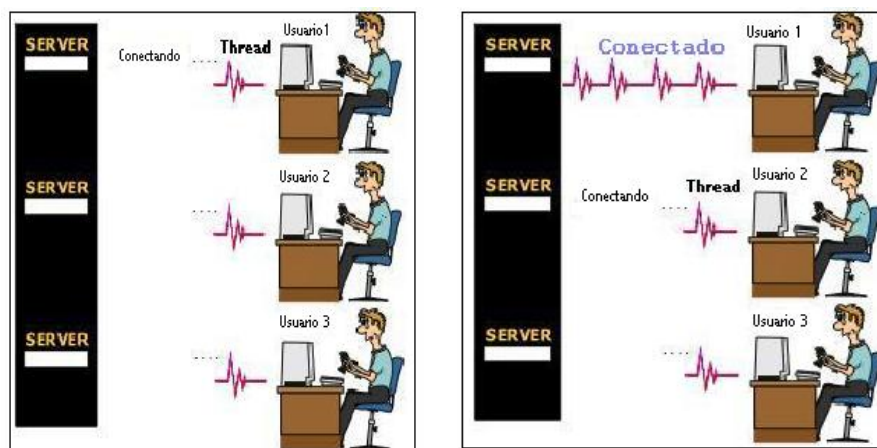


Fig. 2.7. Conexión de Clientes con el servidor MySQL (Mysql.com)



b) Capacidades

- 👉 Escrito en C y C++, testado con GCC 2.7.2.1. Usa GNU autoconf para portabilidad.
- 👉 Clientes C, C++, JAVA, Perl, TCL.
- 👉 Multiproceso, es decir puede usar varias CPU si éstas están disponibles.
- 👉 Puede trabajar en distintas plataformas y S.O. 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, carácter, fechas, enumerados, etc.
- 👉 ODBC para Windows 95 (con fuentes), se puede utilizar Access u otras herramientas de desarrollo para conectarse con el servidor.



c) Seguridades

El sistema de seguridad de MySQL garantiza que cada usuario pueda hacer las cosas que le están permitidas (nada más ni nada menos). El sistema decide los diferentes privilegios dependiendo de "QUÉ USUARIO" conectado a "QUÉ BASE DE DATOS" desde "QUÉ HOST". El sistema de privilegios está basado, en el contenido de 3 tablas, "USER", "HOST" y "DB" de la base de datos "mysql".

Las columnas de estas 3 tablas son:

Base de Datos: mysql

Tables
Db
Host
User

Tabla: db

Field	Type	Null	Key	Default	Extra
Host	char(60)		PRI		
Db	char(32)		PRI		
User	char(16)		PRI		
Select_priv	char(1)			N	
Insert_priv	char(1)			N	
Update_priv	char(1)			N	
Delete_priv	char(1)			N	
Create_priv	char(1)			N	
Drop_priv	char(1)			N	



Tabla: host

Field	Type	Null	Key	Default	Extra
Host	char(60)		PRI		
Db	char(32)		PRI		
Select_priv	char(1)			N	
Insert_priv	char(1)			N	
Update_priv	char(1)			N	
Delete_priv	char(1)			N	
Create_priv	char(1)			N	
Drop_priv	char(1)			N	

Tabla: user

Field	Type	Null	Key	Default	Extra
Host	char(60)		PRI		
User	char(16)		PRI		
Password	char(16)				
Select_priv	char(1)			N	
Insert_priv	char(1)			N	
Update_priv	char(1)			N	
Delete_priv	char(1)			N	
Create_priv	char(1)			N	
Drop_priv	char(1)			N	
Reload_priv	char(1)			N	
Shutdown_priv	char(1)			N	
Process_priv	char(1)			N	



Se puede decidir si se autoriza o no a seleccionar (SELECT), insertar (INSERT), actualizar (UPDATE), y borrar (DELETE) filas de una tabla.

Se puede permitir, o no, crear o borrar (CREATE, DROP) tablas o bases de datos. Se puede permitir, o no, usar comandos de administración del gestor de bases de datos como "shutdown", "reload", "process", etc.

Se puede siempre comprobar privilegios con el script "mysqlaccess. Una columna "USER" vacía significa cualquier nombre de usuario.

d) Lenguaje de Consulta

Este servidor de bases de datos está considerada como la más rápida entre las de su misma categoría tanto para volúmenes de datos grandes como pequeños, pero como también indicamos al principio es a base de no implementar ciertas funcionalidades SQL, éstos aspectos, es por un lado los Triggers y por otro lado la Lógica Transaccional (como commit y rollback).

Los triggers no es nada más que una porción de código almacenado que se "dispara" o se ejecuta cuando se realiza una operación (actualización, borrado, etc.) con la base de datos. Naturalmente comprobar la propia existencia del disparador y ejecutarlo si existe consumo de recursos y tiempo y es como su propio manual indica, la única razón por la que los triggers no están soportados.

Un aspecto muy importante en cualquier base de datos relacional es la consistencia de las diferentes tablas que la componen, para conseguir esto de una forma más o menos fácil es utilizando la "Lógica Transaccional", será el propio gestor de base de datos el que



proporcione mecanismos de bloqueo de archivos y consolidación o retroceso en las operaciones con las tablas. Nos podemos ayudar con los comandos LOCK tables /UNLOCK tables que permiten bloquear tablas impidiendo que otros usuarios puedan acceder a ellas pero sin la posibilidad de deshacer las operaciones realizadas con los datos.

e) Tipos de datos

Datos numéricos

Tipo de Dato	Almacenamiento
TINYINT	1 byte
SMALLINT	2 bytes
MEDIUMINT	3 bytes
INT	4 bytes
INTEGER	4 bytes
BIGINT	8 bytes
FLOAT(4)	4 bytes
FLOAT(8)	8 bytes
FLOAT	3 bytes
DOUBLE	8 bytes
DOUBLE PRECISION	8 bytes
REAL	8 bytes
DECIMAL(M,D)	M bytes (D+2, if M < D)
NUMERIC(M,D)	M bytes (D+2, if M < D)

Datos Date y Time

Tipo de Dato	Almacenamiento
DATE	3 bytes
DATETIME	8 bytes
TIMESTAMP	4 bytes
TIME	3 bytes
YEAR	1 byte

Datos String

Tipo de Dato	Almacenamiento
CHAR(M)	M bytes, 1 ≤ M ≤ 255
VARCHAR(M)	L+1 bytes, where L ≤ M and 1 ≤ M ≤ 255
TINYBLOB, TINYTEXT	L+1 bytes, where L < 2 ⁸
BLOB, TEXT	L+2 bytes, where L < 2 ¹⁶
MEDIUMBLOB, MEDIUMTEXT	L+3 bytes, where L < 2 ²⁴
LOBLOB, LONGTEXT	L+4 bytes, where L < 2 ³²
ENUM('value1','value2',...)	1 or 2 bytes
SET('value1','value2',...)	1, 2, 3, 4 or 8 bytes



Estudio de conectividad con LINUX

Instalación

MySQL junto con PostgreSQL tal vez son los que mejor prestaciones ofrecen en lo referente a instalación y aplicaciones APIs, es así que MySQL también se distribuye en los tres tipos de paquetes conocidos: binario, RPM y DEB. Por estar trabajando con RedHat Linux se eligió la distribución RPM, cuyos archivos necesarios para la instalación son: **MySQL-3.23.41-1.i386.rpm** para el servidor y **MySQL-client-3.23.41-1.i386.rpm** para el cliente. La instalación genera y activa el servicio **mysqld** que se viene a constituirse el servidor de la base de datos. Si se desea se puede desactivar mediante el programa **ntsysv** o **linuxconf**. No requiere de un usuario especial.

Servidor de Base de Datos

El servidor es el programa **mysqld** mismo que permite administrar las bases de datos creados por el usuario.

Monitor SQL

El monitor es el programa **mysql** mismo que se ejecuta tan solo con escribirlo en el shell de Linux, mismo que presenta lo siguiente:

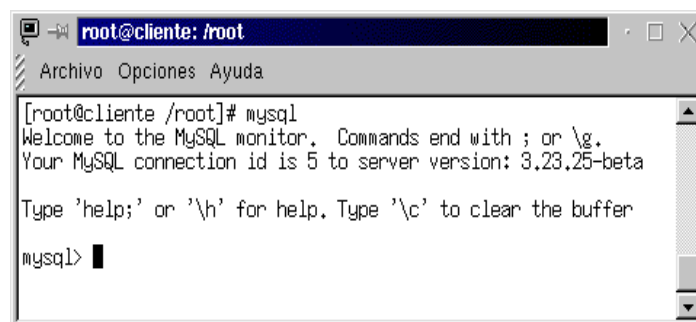


Fig.2.8. Monitor de MySQL



Para conectarse desde un workstation digitar lo siguiente:

```
shell> mysql -h host -u user -p
Enter password: *****
```

donde host es el nombre de la máquina servidor y user el usuario.

A más de esto MySQL proporciona programas clientes con interfaces gráficas que facilitan las consultas y la conexión con el servidor, pero no son interactivos como el monitor mysql.

Desarrollo de aplicaciones cliente

MySQL proporciona librerías API para los lenguajes: C/C++, Tcl/Tk, Perl, Python, Delphi, mismos que permiten realizar front/ends de aplicaciones cliente con MySQL.

Además posee un controlador ODBC denominado MyODBC para la conexión con lenguajes bajo Microsoft. Para Windows existen programas clientes que permiten conectarse con el servidor de base de datos en una máquina Linux como por ejemplo MySQLAdmin de Artronic.

MySQL también está desarrollado para la plataforma Windows 9x y NT tanto de servidor como de cliente, pero a igual manera que PostgreSQL, su uso tiene un coste. Una versión Shareware está disponible en la red.



CAPITULO III

EVALUACIÓN DE LOS SGBDD FREEWARE

3.1. Introducción

Juzgar el rendimiento de un SGBDD cuando se están tomando decisiones de compra o uso es algo crítico a fin de retardar la obsolescencia y proteger la inversión y así obtener el máximo beneficio de un sistema.

Parte de esto consiste en asegurarse que el SGBDD que se elija, permita hacer lo que se desea durante toda la vida del producto. Esto significa que no solo va a almacenar los datos sino que disponga de recursos necesarios para tener la capacidad de resolver problemas complejos de actualización, seguridad, escalabilidad (El administrador de sistemas no necesita preocuparse del número de usuarios para el que fue desarrollado el sistema) y portabilidad que exige la tecnología de la información actual.



Es importante evaluar a una serie de aspectos relativos al rendimiento cuando se han de tomar decisiones de compra y uso. El decidir que SGBDD es el adecuado, tiende a ser cada vez más difícil dado que los nuevos productos que están apareciendo en el mercado son numerosos y cada uno ofreciendo mejores prestaciones. Una manera de diferenciar estos productos consiste en saber cómo su rendimiento se ve afectado con los trabajos en los que se está interesado en obtener un mejor provecho, a fin de ganar tiempo de respuesta en las actividades que diariamente realiza. Conocer este aspecto, permitirá realizar una elección mejor informada. Sin embargo, medir el rendimiento no es siempre algo simple. Para tomar la decisión correcta, se necesita comprender qué es lo que se está midiendo.

3.2. El Benchmark

Para realizar una buena selección entre alternativas, existen pruebas comparativas (Benchmark) que son programas software que miden el rendimiento de una computadora, o software. Estas pruebas no solamente pueden ayudar a comprender el rendimiento de los diferentes sistemas a comparar, sino que también pueden ayudar a evaluar el rendimiento de un sistema determinado bajo diferentes configuraciones hardware y software. Cada prueba evalúa un diferente tipo de trabajo. Algunas pruebas evalúan lo rápido que una computadora puede generar un documento. Algunas pruebas evalúan lo rápido que se dibuja un gráfico en la pantalla. Otras evalúan cómo de rápidas se realizan determinadas operaciones matemáticas. Algunas pruebas pueden evaluar todas esas tareas al mismo tiempo.

Hay dos tipos de pruebas: **APLICACION y SINTETICAS.**

- Pruebas de aplicación evalúan el rendimiento usando aplicaciones reales en la forma en que se utilizan.
- Las pruebas sintéticas usan programas software creados específicamente para evaluar el rendimiento. Ambos tipos tienen ventajas y desventajas.



Existe otro tipo de pruebas denominadas **ARTIFICIALES**³ que son test que permitan cuantificar ciertos parámetros particulares del sistema.

Hay muchas pruebas de rendimiento diferentes disponibles hoy día, pero no todas las pruebas han sido creadas de la misma forma. Crear una prueba requiere una considerable experiencia e inversión. La utilidad y exactitud de cualquier prueba depende fundamentalmente de la calidad de su implementación y las suposiciones que se hacen, por lo que puede darse que los resultados pueden variar de una prueba a otra.

Usualmente, cuando se va a evaluar, se presentan una serie de técnicas de manera aislada, y en algunos casos, se presentan técnicas como si fueran excluyentes, sin serlo. Afortunadamente, en la práctica se impone la evaluación como una totalidad, y los evaluadores se ven obligados a desarrollar técnicas, cuantitativas y cualitativas para cumplir, lo más integralmente posible, con su labor. Los instrumentos básicos para la evaluación, están relacionados e integrados.

En el conocimiento de las cosas, como en el conocimiento analítico de los sistemas, se transita de lo fácil a lo difícil, de lo inmediato a lo mediano, *"de lo que se tiene a mano a lo que hay que conseguir"*, de lo superficial a lo profundo, de lo simple a lo complejo. De ahí que no es aconsejable, por ello, discriminar algunas técnicas que se han desarrollado en nuestro medio. De ahí que nuestra metodología se basa a partir de un Análisis de Conceptos y Características (categorizado como pruebas ARTIFICIALES), como los indicadores de orden técnico, respaldo de la empresa, costo del equipo, mediciones de carga (son índices cuantitativos y cualitativos que soporta un sistema con respecto al tiempo. Como por ejemplo: Tiempo de respuesta, tiempo de

³ PADILLA Byron; TERAN Gustavo, RUIZ Gerson: Evaluación de Herramientas Cliente / Servidor para Sistemas Financieros Contables en Instituciones públicas. Tesis de Grado de La Universidad Técnica del Norte, 1999, Pág.: 109



retorno, tiempo de paginación, etc), etc. De acuerdo a los cuales se puede llegar a determinar la mejor alternativa que ofrece mayores beneficios.

Sobre la evaluación de SGBDD Freeware, se formularan ciertos parámetros o criterios estratégicos de acuerdo a los requerimientos dentro del IMH, que luego de ser evaluadas permitirán establecer las conclusiones obtenidas del análisis.

3.3. Determinación de factores de evaluación

A efectos de seleccionar el SGBDD será necesario, en forma previa, dimensionar la evaluación, es decir, establecer los parámetros o factores sobre los cuales se aplicarán las pruebas Benchmark, para cada alternativa, según los requerimientos del método a emplear. Hemos determinado los siguientes criterios:

- A. Prestaciones del Productor
- B. Prestaciones del Motor
- C. Soporte Técnico
- D. Interfaz de Usuario
- E. Desarrollo, Control y Administración de BDD
- F. Compatibilidad SQL
- G. Soporte De Programación
- H. Capacidad de Respuesta

3.3.1 Prestaciones del Productor

Resulta imprescindible conocer lo que ofrece la empresa productora del motor de BDD, debido a que sin este apoyo no se podría obtener algún tipo de ayuda ante cualquier



situación de errores que se puedan producir durante la instalación, configuración, uso, etc. de la herramienta.

3.3.2 Prestaciones del Motor

Un SGBDD debe ofrecer ciertas facilidades en lo referente a características técnicas, para de esta manera llegar a conocerle desde el fondo mismo del motor, no se puede empezar a trabajar si saber con que mismo se cuenta.

3.3.3 Soporte Técnico

Tomando en cuenta que un SGBDD Freeware para Linux es gratuito y aún más si es distribuido a través de Internet, estamos exentos de un soporte técnico presencial, pero podemos recibirlo en base a libros o documentos completos y fáciles de entender (esta categoría tiene cierta relación con la primera).

3.3.4 Interfaz de Usuario

Es tal vez la categoría que más escasamente proveen los SGBDD Freeware, especialmente lo que tiene que ver con el desarrollo de la aplicación, no existe, como en Windows, lenguajes visuales que faciliten la construcción de la interfaz gráfica de usuario, pero sí, ambientes GUI que proveen una administración completa de las BDD (creación, eliminación, seguridades, control de las BDD y usuarios).

3.3.5 Desarrollo, Control y Administración de BDD

La facilidad y utilidades que ofrezcan los SGBDD Freeware para la administración, control y desarrollo de BDD se constituye en factor muy importante para la determinación de la mejor opción entre las diferentes alternativas, ya que la información es la parte más importante y valiosa que se debe proteger.



3.3.6 Compatibilidad SQL

SQL se ha convertido en el lenguaje de query relacional más popular. En la historia del SLQ (ver ANEXO D) a pasado varias etapas, de ahí que la compatibilidad del lenguaje de consulta que proveen los SGBDD es muy importante, debido a que muchos usuarios pueden necesitar trabajar con ciertos comandos del SQL Estándar y si estos no lo proveen, puede causar cierto rechazo hacia ellos.

3.3.7 Soporte de programación

Los SGBDD proporcionan las herramientas API para la conexión y ejecución de comandos SQL desde un lenguaje de programación específico para el desarrollo de aplicaciones, exclusivamente para el ambiente Linux, así como también drivers ODBC para la conexión con clientes Windows y front-ends desarrollados para la comunicación con el servidor de Bases de Datos Linux.

3.3.8 Capacidad de respuesta

Son índices con los cuales se determina el desempeño del SGBDD. Uno de estos índices es el **Tiempo de Retorno** que es el tiempo que se demora el SGBDD en ejecutar una transacción o sentencia SQL, desde que el cliente digita un ENTER hasta que el servidor presente los resultados. Otro es el **Tiempo de Respuesta** que es el número de transacciones que ejecuta el servidor por unidad de tiempo. Hemos considerado también ciertas características técnicas de limitaciones que puede tener un SGBDD en lo que respecta a número límite de registros, columnas, bases de datos, etc.

3.4 Generación y Cuantificación del Benchmark

El método nos facilita una serie de formularios que contienen test de parámetros que permitirán evaluar los diferentes motores de Bases de Datos Freeware para Linux, recopilados en el ANEXO E de acuerdo a los grupos especificados con anterioridad.



3.4.1 Cuantificación de los criterios de evaluación

Todos los parámetros de evaluación a excepción de la CATEGORÍA H tiene un valor de **UN PUNTO**. La categoría H se evaluará de acuerdo a un rango de valores preestablecidos de acuerdo a la medición obtenida de los cuatro SGBDD que se están evaluando. El rango estará dividido en tres, el mismo que equivale a: 10 para el mejor valor, 5 para el valor medio y 3 para el valor más bajo.

3.5 Aplicación de los test Benchmark

La aplicación del test de evaluación a todas las alternativas de Bases de Datos: MySQL, PostgreSQL (PSQL), InterBase (INTB) y MIMER, arrojaron los siguientes resultados:

Simbología:

- Si soporta o provee*
- No soporta o provee*



A. PRESTACIONES DEL PRODUCTOR				
PARAMETROS	MySQL	PSQL	INTB	MIMER
1. Tiene el productor fuerte imagen en Internet?	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
2. ¿Garantiza el productor mejoras a futuro ?	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
3. ¿Cuenta el productor con:				
➤ Foros	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>
➤ Grupo de noticias	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
➤ Afiliaciones	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
➤ Soporte y asesoramiento vía e-mail	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
4. Provee un site con preguntas frecuentes (FAQ)	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>
5. El site está renovándose y ofreciendo mayor soporte?	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
6. El soporte vía e-mail es inmediata?	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>
7. Ofrece información técnica completa del producto?	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
8. Ofrece una lista de posibles bugs que se pueden producir?	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
9. Facilita parches frente a bugs encontrados?	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
TOTAL	12	5	8	7



B. PRESTACIONES DEL MOTOR

B.1 INSTALACION Y MANTENIMIENTO

PARAMETROS	MySQL	PSQL	INTB	MIMER
1. Qué tipo de instalación provee?				
➤ Binaria	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>
➤ RPM o DEB	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>
➤ Asistente Propio	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>
2. ¿Qué incluye la distribución?				
➤ Herramientas Servidor y Cliente	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
➤ Documentación	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
➤ Herramientas de desarrollo	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
➤ Código Fuente	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>
➤ ODBC	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
3. Soporta como mínimo la instalación en una PC Pentium con 16 MB RAM?	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
4. ¿Facilita la migración de los datos durante la actualización del sistema?	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>
5. Que funciones básicas incorpora				
➤ Ambiente GUI (nivel administrador)	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
➤ Escalabilidad	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
➤ Seguridad	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
➤ Multiplataforma	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
➤ Interfase de usuario (monitor)	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
➤ Conectividad	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
➤ Lenguajes de programación	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>

B.2 CARACTERISTICAS GENERALES

PARAMETROS	MySQL	PSQL	INTB	MIMER
5. ¿Qué sistemas operativos soporta con licencia GPL?				
➤ Unix / Linux	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
➤ WINx	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
➤ SUN OS	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>
6. ¿Puede correr el SGBDD en el actual ambiente de la institución?	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
7. Qué protocolos de red soporta?				
➤ NetBEUI y IPX/SPX	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>
➤ TCP/IP	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
➤ DNS	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
➤ DHCP	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
➤ PIPES de Windows NT	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>
8. Usa caché para las Bases de Datos	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
9. Soporta Sistemas Multiprocesamiento Simétrico SMP?				



	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>
10. Puede ejecutar múltiples servidores en una misma máquina?	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
11. Puede trabajar como servidor Web?	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>
12. Tiene soporte multi-thread?	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
13. Soporta Integridad Referencial?	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
TOTAL	28	25	21	19



C. SOPORTE TECNICO				
C.1 DOCUMENTACION				
PARAMETROS	MySQL	PSQL	INTB	MIMER
1. ¿Cómo se distribuye la documentación proporcionada?				
➤ CD-ROM	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
➤ En línea	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
➤ Libros	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
➤ Vía Internet	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
2. ¿Están disponible versiones en español?	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
3. ¿Es fácil de entender y usar ?	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
4. ¿Qué tipo de documentación se provee ?				
➤ Referencia de lenguaje	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
➤ Administración	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
➤ Guías del programador	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
➤ Guía API's	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
➤ Tutorial	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
5. El contenido de la documentación es completa?	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
6. ¿Proporciona ejemplos ?	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
7. Qué formatos proporciona la documentación?				
➤ DOC de MS Word	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
➤ PDF	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
➤ HTML	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
➤ Texto genérico	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
C.2 COSTOS EXTRAS				
PARAMETROS	MySQL	PSQL	INTB	MIMER
8. ¿Existe un cargo por concepto de asesoramiento?	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>
9. ¿Tiene un costo para soporte técnico adicional (vía e-mail)?	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>
TOTALES	14	14	8	10



D. INTERFAZ DE USUARIO				
PARAMETROS	MySQL	PSQL	INTB	MIMER
1. Cómo Interactúa con el administrador?				
➤ Monitor basado en texto	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
➤ Ambiente GUI	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
2. El ambiente GUI se distribuye por:				
➤ Binaria	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
➤ RPM o DEB	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
➤ Otra forma	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
3. Que tipo de ayuda en línea posee?				
➤ En línea con ambiente GUI	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
➤ En línea basado a comandos	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
4. Qué operaciones de administrador permite el GUI?				
➤ Creación de usuarios	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
➤ Eliminación de usuarios	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
➤ Matar procesos	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
➤ Creación de BDD	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
➤ Eliminación de BDD	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
➤ Limpiar BDD	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
➤ Editar y ejecutar Querys	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
➤ Apagar el servidor	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
➤ Conexión con servidor	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
➤ Status del servidor	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
5. Permite creación de Informes?	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
6. Permite creación de Reportes?	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
7. Posee herramienta GUI cliente para Win 9x?	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
TOTALES	16	16	1	1



E. DESARROLLO, CONTROL Y ADMINISTRACIÓN DE LAS BASES DE DATOS

PARAMETROS	MySQL	PSQL	INTB	MIMER
1. Que tipo de recuperación maneja?				
➤ Roll-Back	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
➤ Commit Work	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
2. El administrador de usuarios permite:				
➤ Creación	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
➤ Eliminación	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
➤ Modificación	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
➤ Seguridad de acceso	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
3. Permite el API la administración de la BDD (creación, eliminación, seguridad de las BDD)?	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>
4. Qué niveles de seguridad soporta?				
➤ Validación de usuarios	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
➤ Privilegios (nivel de bases de datos)	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
5. Posee herramientas o comandos que dan flexibilidad y eficiencia la configuración de seguridades de BDD voluminosas con gran cantidad de usuarios?	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
6. Los usuarios del motor de BDD es diferente al del SSOO?	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
7. Dispone de scripts para:				
➤ Chequeo, optimización, recuperación de tablas.	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
➤ Chequeo de privilegios para un host.	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
➤ Reporte de problemas.	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
➤ Administración de la Base de Datos y usuarios.	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
➤ Importación de datos desde archivos texto.	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
8. Ofrece opciones de Backup?	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
TOTALES	15	14	13	12



F. COMPATIBILIDAD SQL				
PARAMETROS	MySQL	PSQL	INTB	MIMER
1. Permite la migración de los datos a otros formatos?	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
2. Que compatibilidad SQL provee?				
➤ Nivel 1 (SQL/86)	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
➤ Nivel 2 (SQL/89)	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
➤ Nivel 3 (SQL/92 o SQL/2)	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
➤ Nivel 4 (SQL/3)	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
3. ¿Cómo se crea la base de Datos ?				
➤ Scripts SQL	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
➤ Comandos	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
➤ Asistentes	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
4. ¿Facilita la construcción de triggers ?	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
5. ¿Permite el tipo de dato BLOB (multimedia) ?	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
6. Permite el uso de subqueries?	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
7. Permite subqueries recursivas?	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
8. Permite el uso de joins?	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
9. Permite el uso de group by?	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
10. Permite el uso de Order by?	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
11. Permite views?	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
12. Soporta el monitor la creación y almacenamiento de procedimientos de consultas personalizados?	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
TOTALES	12	17	13	14



G. SOPORTE DE PROGRAMACION				
PARAMETROS	MySQL	PSQL	INTB	MIMER
1. ¿Qué lenguaje soporta ?				
➤ Alto nivel (C/C++, Pascal)	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
➤ Traductores (Tcl, Perl, Python)	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
2. ¿El acceso a los datos es mediante:				
➤ APIs	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
➤ Pre-Compilador	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
➤ ODBC	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
3. Permite la programación visual bajo Linux?	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
4. ¿Permite programación orientada a objetos ?	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
5. ¿El tiempo de aprendizaje del lenguaje es corto?	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
7. ¿Es fácil personalizar las aplicaciones?	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
8. Posee asistentes para el desarrollo de informes y reportes?	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
9. Facilita ejemplos de conexión?	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
10. Que funciones permite los drivers ODBC:				
➤ Importar las BDD desde el servidor	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
➤ Exportar una BDD desde Windows al Servidor	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
➤ Vincular una BDD del servidor al gestor local	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
TOTALES	12	13	7	7



H. CAPACIDAD DE RESPUESTA							
PARÁMETROS				MySQL	PSQL	INTB	MIMER
1. Tiempo de retorno? (comando select)							
Motor	< 2s	2s	> 2 s				
MySQL	<input checked="" type="checkbox"/>			10			
PSQL		<input checked="" type="checkbox"/>			5		
INTB		<input checked="" type="checkbox"/>				5	
MIMER		<input checked="" type="checkbox"/>					5
2. Tiempo de respuesta (8000 registros, comando insert)							
Motor	< 20s	40s	> 40 s				
MySQL	<input checked="" type="checkbox"/>			10			
PSQL			<input checked="" type="checkbox"/>		3		
INTB			<input checked="" type="checkbox"/>			3	
MIMER			<input checked="" type="checkbox"/>				3
3. Nivel de dificultad de la configuración Cliente / servidor.							
Motor	Fácil	Difícil	Muy Difícil				
MySQL	<input checked="" type="checkbox"/>			10			
PSQL	<input checked="" type="checkbox"/>				10		
INTB		<input checked="" type="checkbox"/>				5	
MIMER			<input checked="" type="checkbox"/>				3
4. Tiempo aproximado para la instalación del motor y configuración del cliente Windows?.							
Motor	< 1 día	1 día	> 1 día				
MySQL	<input checked="" type="checkbox"/>			10			
PSQL		<input checked="" type="checkbox"/>			5		
INTB			<input checked="" type="checkbox"/>			3	
MIMER			<input checked="" type="checkbox"/>				3
5. Número máximo de columnas por Tabla.							
Motor	< 3398	3398	> 3398				
MySQL		<input checked="" type="checkbox"/>		5			
PSQL	<input checked="" type="checkbox"/>				3		
INTB			<input checked="" type="checkbox"/>			10	
MIMER	<input checked="" type="checkbox"/>						3
6. Número máximo de registros por Tabla.							
Motor	< 65534	65534	> 65534				
MySQL		<input checked="" type="checkbox"/>		10			
PSQL	<input checked="" type="checkbox"/>				3		
INTB	NI						
MIMER	<input checked="" type="checkbox"/>						3
7. Cuál es el tamaño máximo de la Base de Datos?							
Motor	< 4 GB	4 GB	> 4 GB				
MySQL		<input checked="" type="checkbox"/>		5			



PSQL	<input checked="" type="checkbox"/>				3			
INTB		<input checked="" type="checkbox"/>				5		
MIMER		<input checked="" type="checkbox"/>					5	
8. Máximo tamaño de char								
Motor	< 255	255	> 255					
MySQL		<input checked="" type="checkbox"/>			5			
PSQL			<input checked="" type="checkbox"/>			3		
INTB		<input checked="" type="checkbox"/>					5	
MIMER			<input checked="" type="checkbox"/>				3	
9. Máximo tamaño del tipo text o blob								
Motor	< 1048543	1048543	> 1048543					
MySQL		<input checked="" type="checkbox"/>			5			
PSQL	<input checked="" type="checkbox"/>					3		
INTB			NI					
MIMER			<input checked="" type="checkbox"/>				10	
10. Máximo tamaño de varchar								
Motor	< 255	255	> 255					
MySQL		<input checked="" type="checkbox"/>			5			
PSQL			<input checked="" type="checkbox"/>			10		
INTB			NI					
MIMER			<input checked="" type="checkbox"/>				10	
11. Número de conexiones simultáneas								
Motor	< 30	30	> 30					
MySQL		<input checked="" type="checkbox"/>			5			
PSQL			<input checked="" type="checkbox"/>			10		
INTB	<input checked="" type="checkbox"/>						3	
MIMER			<input checked="" type="checkbox"/>				10	
TOTALES					80	53	39	58

- **NI:** No existe información

3.6 Análisis de Resultados

Una vez aplicado el test y calificado cada uno de estos, a los SGBDD Freeware para Linux se obtuvo los siguientes puntajes:



CRITERIOS	MySQL	PSQL	INTB	MIMER
A. Prestaciones del Productor	12	5	8	7
B. Prestaciones del Motor	28	25	21	19
C. Soporte Técnico	14	14	8	10
D. Interfaz de Usuario	16	16	1	1
E. Desarrollo, Control y Administración de BDD	15	14	13	12
F. Compatibilidad SQL	12	17	13	14
G. Soporte De Programación	12	13	7	7
H. Capacidad de Respuesta	80	55	41	60
TOTAL	189	157	110	128

El gráfico siguiente nos muestra la comparación de los distintos puntajes obtenidos:

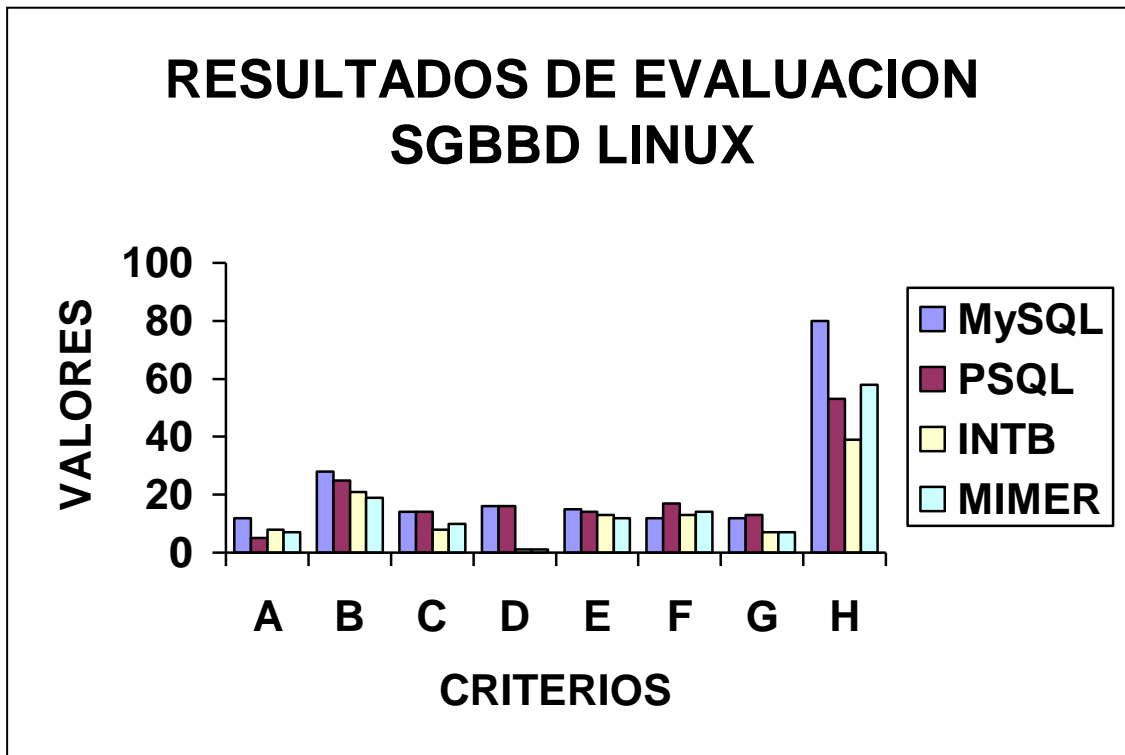


Fig. 3.1. Gráfico de barras que compara los puntajes individuales, obtenidos de acuerdo a los parámetros de evaluación

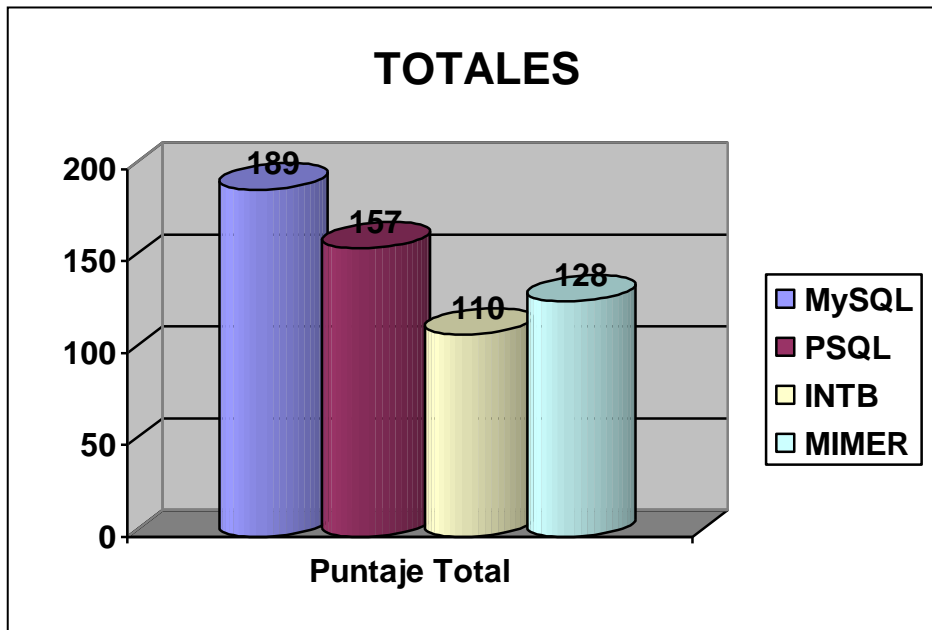


Fig. 3.2. Gráfico de barras que compara los puntajes totales, obtenidos de acuerdo a los parámetros de evaluación

Con estos datos podemos definir lo siguiente:

- ✓ Encontramos que existe dos polos bien marcados de SGBDD semejantes, el primero compuesto por MySQL y PostgreSQL, que básicamente proveen las mismas capacidades y prestaciones por parte de los motores; el segundo por InterBase y MIMER, cuyo fallo podríamos determinar que existe en la falta de herramientas con ambientes GUI, ya sea tanto para el servidor como para el cliente.
- ✓ Todos los motores proporcionan una excelente compatibilidad con el lenguaje SQL, que hoy por hoy es un estándar en materia de Bases de Datos.
- ✓ La más clara falencia de los SGBDD Freeware para Linux comparadas con sus similares para plataforma Windows, es la no-existencia de herramientas visuales para el desarrollo de aplicaciones cliente, peor aún de herramientas CASE para la construcción de Bases de Datos, sin embargo, consideramos que esto no puede ser un gran motivo para desechar completamente estos motores que tienen una gran acogida



a nivel mundial, ya sea como servidores de datos para empresas de todo tipo, como también de servidores WEB.

- ✓ Los SGBDD Linux tienen muchas características presentes en grandes SGBDDs comerciales, como transacciones, subselecciones, triggers, vistas y bloqueos sofisticados. Pero también otras que no son proporcionadas como integridad referencial de claves externas o enlaces externos, en algunos casos claves foráneas, etc., pero lo más importante de esto es que continuamente se está mejorando en las versiones subsiguientes (aparece una nueva edición o release cada tres meses de promedio), reparando fallas (bugs), adicionando nuevas características, etc.

3.7 Selección de la mejor alternativa para la aplicación

Ahora trataremos de determinar cuál de los SGBDD Freeware para Linux seleccionaremos para el desarrollo de nuestra aplicación de Inventarios y Bodegas del Ilustre Municipio de Huaca.

Primeramente, podemos observar de los resultados obtenidos con anterioridad, que los SGBDD que ofrecen mayores prestaciones para poder trabajar eficientemente en una entidad municipal son: PostgreSQL y MySQL; para seleccionar una de las dos veremos las siguientes comparaciones:

- ✓ PostgreSQL se ejecuta en dos modos. El modo normal *fsync* pasa cada transacción completada en el disco, garantizando que si el Sistema Operativo se bloquea o se produce una pérdida de energía en los siguientes segundos, todos sus datos están almacenados y sin daños en el disco (documento FAQ de PostgreSQL). Podemos darnos cuenta que en



este modo, es más lento que muchas bases de datos. En el modo *no-fsync*, es generalmente más rápido, aunque en este modo un fallo del Sistema Operativo podría corromper los datos. Según PostgreSQL se está trabajando en una solución intermedia que provoque menos sobrecargas en el rendimiento. El modo es seleccionable por el administrador de la base de datos. En comparación con MySQL, es más rápido en inserciones / actualizaciones debido a que no emplea estos tipos de ejecución.

- ✓ PostgreSQL tiene características, como subselects, disparadores, vistas y bloqueos que MySQL no posee; lo contrario es que MySQL posee Integridad Referencial de claves externas o enlaces externos, soporte multi-Thread que PostgreSQL no proporciona.
- ✓ PostgreSQL y MySQL se esfuerzan en generar código estable y bien probado que tenga un mínimo de errores. Cada release o edición tiene al menos un mes de prueba como versión beta, y las diferentes ediciones muestran que pueden suministrar versiones sólidas y estables que están listas para uso productivo.
- ✓ Ninguno de los dos, implementan completamente el ANSI SQL, pero puede considerarse que MySQL provea mayormente tipos y funciones de este Estándar.
- ✓ Las dos APIs son transparentes durante la conexión. Cualquier problema que ocurre es debido a la documentación inadecuada, mas no al API mismo. Ofrecen APIs para C/C++, TCL, PHYTON, PERL, ODBC, JDBC y soporte PHP.
- ✓ MySQL tiene una mayor capacidad de almacenamiento por base de datos (ver test, categoría H), a diferencia de PostgreSQL que tiene un límite de 8K por registro, reduciendo de esta manera su rendimiento.



3.7.1 Conclusión de la Evaluación

Al analizar los resultados del test benchmark, los gráficos, y las comparaciones expuestas anteriormente, concluimos que cualquiera de los dos motores: PostgreSQL y MySQL cumplen con los requerimientos básicos para implementarse como un servidor de Bases de Datos Municipal. Aunque nuestra inclinación será por MySQL debido a las siguientes razones:

- ✓ MySQL está mayormente posesionado y establecido en Internet como servidor de datos para la WEB así como también en empresas que han acogido este servidor de BDD.
- ✓ Facilita también un gran número de direcciones WEB en las que se puede conseguir ayuda en línea, soporte vía e-mail, foros, reporte de bugs, aportes de clientes, herramientas, APIs, etc., además de un completo site (www.mysql.com) con todo lo necesario.
- ✓ Ejecuta las transacciones con mayor rapidez y permite el almacenamiento de un mayor volumen de datos, que una entidad municipal como el Ilustre Municipio de Huaca no puede estar libre de estas exigencias.
- ✓ Los drivers ODBC permiten la conexión y desarrollo de aplicaciones con Visual Basic, mismo con el cual se realizará el Sistema de Inventarios y Bodegas del Ilustre Municipio de Huaca.
- ✓ MySQL ofrece opciones en su documentación, sobre la forma como responder ante las exigencias por la falta de Triggers y SubSelects (Capítulo 5 del Manual de Referencia),



que nos parece buenos consejos en el momento de diseñar y manejar las bases de datos.

- ✓ Además MySQL se ha incluido en la última versión de Red Hat Linux 7.1, demostrando con esto que MySQL posee estabilidad, eficiencia y principalmente fácil de uso.

- ✓ Finalmente, MySQL provee un servidor de Base de Datos para la plataforma Windows 9x y NT, mismo que nos facilitará el trabajo y el futuro empleo del Sistema en el Ilustre Municipio de Huaca debido a la no existencia de una Red en la que pueda instalarse MySQL en una máquina Linux.



CAPITULO IV

ESTUDIO DE LA OPERABILIDAD DE CLIENTE WINDOWS CON SGBDD SOBRE LINUX

4.1 Introducción.

Los usuarios de MS-Windows se encuentran con frecuencia con el problema del acceso remoto a bases de datos. Efectivamente MS-Dos y Windows nunca fueron pensados como sistemas de red. Incluso, cuando no hubo mas remedio, las soluciones no dejaron de ser sino un parche. Del mismo modo las bases de datos eran locales, con soluciones propietarias.

Por otro lado, los grandes sistemas llevan trabajando desde hace años con servidores de bases de datos. Accesos concurrentes, transacciones, bloqueos, *rollback*, etc., son procesos



largamente conocidos y estudiados. El acceso de la informática personal a las redes locales trajo los mismos problemas a los que los grandes mainframes se habían enfrentado hacia años. Solo había un problema: El sistema operativo. En efecto: todas las aplicaciones desarrolladas partían de la base de un sistema operativo monotarea, monousuario, sin servicios de red, y casi sin ningún tipo de control de acceso a los recursos. Incluso en la actualidad, Windows 95 tiene problemas con el acceso a recursos entre aplicaciones.

En los grandes sistemas, y en las redes UNIX/Linux, desde el primer momento los clientes son aplicaciones independientes del servidor, hablan su lenguaje y se entienden con él. En sistemas basados en MS-Dos y Windows se hace necesario el proveer una interfaz entre la aplicación y los servidores. Del mismo modo, en los sistemas UNIX/Linux todo el mundo habla el mismo lenguaje de bases de datos -SQL-, mientras que los PC's están plagados de soluciones propietarias, en función de la aplicación y del fabricante. Se hace pues necesaria una arquitectura en capas para realizar las siguientes tareas:

- Una capa para que la aplicación sea capaz de hablar con el Sistema Operativo a través de un API común.
- Una capa en el Sistema Operativo que se encarga de discernir el tipo de acceso al sistema de archivos, a las bases de datos de otro proveedor, a un servidor remoto y de invocar al driver (controlador) adecuado.
- Un driver que se encarga del acceso a los datos, y de su conversión al API común del sistema operativo
- Si el acceso se efectúa vía red local hacia un servidor remoto, todavía hace falta una capa nueva en el lado del servidor: la que convierte las peticiones del cliente al API de programación que este provee.



Para conseguir esta funcionalidad Microsoft definió un API denominado **Open DataBase Conectivity (ODBC)**. Toda aplicación de bases de datos que se precie para el mundo Windows debe ser capaz de implementar y manejar el API de acceso a la base de datos.

Microsoft ha hecho público el API de programación por lo que en teoría, cualquiera puede escribir un driver ODBC, las aplicaciones de bases de datos que hablen ODBC puedan comunicarse con un servidor, bien sea local, remoto, fichero, aplicación o incluso otro programa de bases de datos que se esté ejecutando en la misma máquina.

Puesto que ODBC es un estándar "de facto", el cliente de la base de datos no tiene que ser un sistema de Microsoft. De hecho, existe un proyecto de colaboración en la comunidad Internet para proveer a los sistemas Linux de un API de acceso ODBC unificado para sus bases de datos denominado FreeODBC (Ver sección 4.3.3.1). Cuál sería el fin de esto, pues SQL tiene ya un API definido, y todos los SGBDD UNIX/Linux saben hablar en SQL, pero se entiende fácilmente si pensamos que si el API de acceso a la base de datos es el mismo en Windows y en UNIX/Linux, facilitará el trabajo de conectividad de BDD Windows hacia Linux.

4.2. SGBDD Cliente / servidor

El resultado del desarrollo del procesamiento distribuido es un modelo Cliente / servidor. Ahora Linux puede utilizarse como el cliente, el servidor o ambos. El cliente (un usuario de PC) solicita un servicio (como imprimir) que un servidor le proporciona (un procesador conectado a la LAN). Este enfoque común de la estructura de los sistemas informáticos se traduce en una separación de las funciones que anteriormente forman un todo.

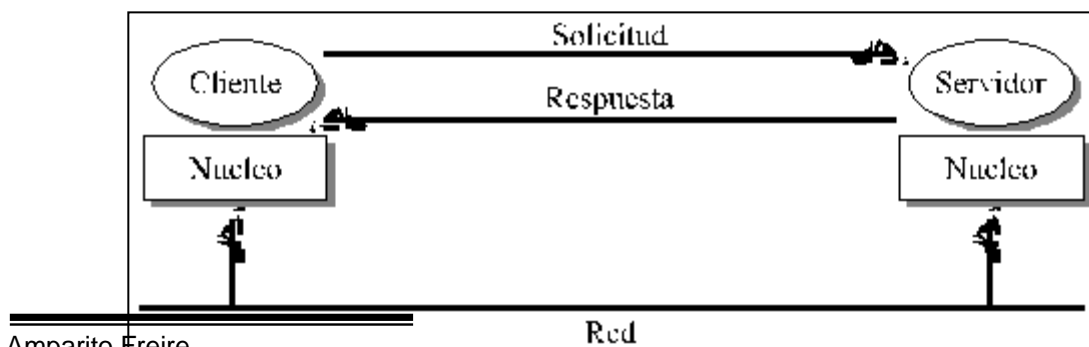




Fig. 4.1. Modelo Cliente / servidor

De igual manera los sistemas de gestión de Base de datos con arquitectura cliente / Servidor permite la comunicación entre un proceso cliente (por ejemplo una consulta de datos) en una máquina remota con el Servidor de Bases de Datos ubicada en la máquina Servidora.

4.2.1 Descripción del Cliente.

Un cliente es una aplicación típica de usuario escrita en C, C++, Delphi, Tcl, Perl, Python o Java, que accedan a los datos de un determinado SGBDD. En otras palabras un cliente es un proceso de alguna aplicación que usa librerías cliente, en forma directa, estableciendo un canal de comunicación con el Servidor. La conexión puede ser local si la aplicación es ejecutada en la misma máquina que se encuentra el Servidor o remoto si la aplicación es usada en una máquina cliente de una red, conectada al Servidor de Base de Datos.

Las librerías son funciones que permiten desarrollar aplicaciones cliente usando conexiones al Servidor, para de esta manera realizar diferentes operaciones sobre una Base de Datos. Una de las más comunes es ODBC de Microsoft.

4.2.2 Descripción del Servidor

El Servidor es un proceso de software ejecutado en el host de una red que almacena las bases de Datos. El proceso Servidor es el único proceso de un host que puede manipular la información de las Bases de Datos. Los clientes envían al proceso Servidor peticiones o acciones de diferente tipo para la base de Datos, como por ejemplo:

- Búsqueda de una base de datos de acuerdo a un criterio
- Comparar, ordenar y organizar los datos



- Modificar valores de los datos
- Insertar nuevos datos a la BDD
- Remover datos de la BDD
- Crear nuevas bases de datos o estructura de datos
- Ejecutar código de procedimientos en el Servidor
- Enviar mensajes a otros clientes conectados al mismo tiempo

4.3 Soportes ODBC para SGBDD sobre LINUX

“ODBC no es sino un API de conectividad entre aplicaciones de bases de datos cliente y servidor.

Dicho API esta organizado en varias capas: de aplicación, de sistema, y de acceso.”

4.3.1. Funcionamiento de ODBC

Como hemos explicado en la introducción, ODBC es un API de interfaz entre clientes de bases de datos y servidores de bases de datos. La figura siguiente ilustra este esquema:

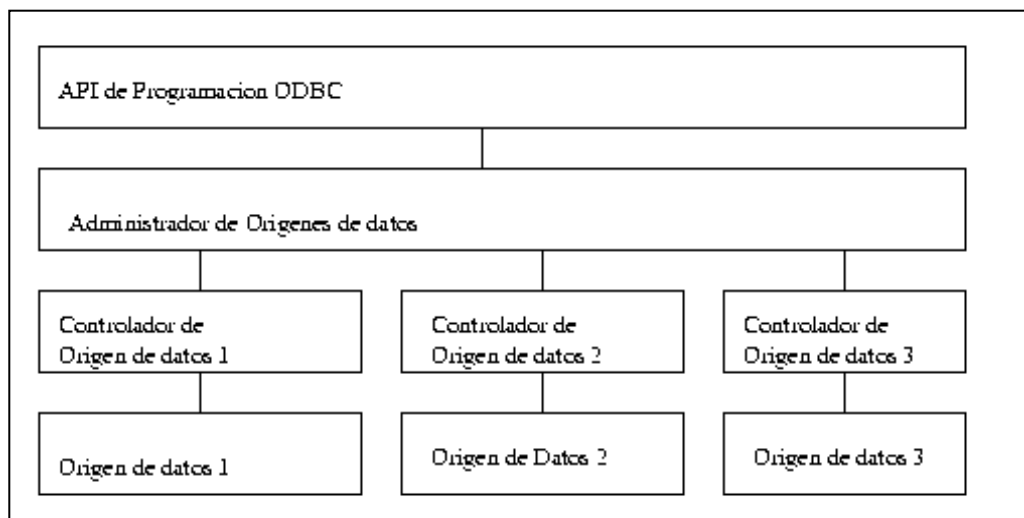


Fig. 4.2. Estructura en capas del API ODBC



La primera capa constituye la librería del API que utilizan las diversas aplicaciones que "hablan" ODBC. Microsoft proporciona para sus sistemas el fichero *ODBC32.DLL*, que contienen el API y la interfaz con el sistema operativo, permitiendo a los desarrolladores de controladores ODBC incluir dicha librería en sus distribuciones. Para sistemas UNIX/Linux, el proyecto FreeODBC, ha desarrollado su propia librería GPL *libodbc.so.x.x* que es totalmente compatible con las especificaciones descritas por Microsoft.

El administrador de orígenes de datos es el responsable del "rutado" de peticiones de ODBC desde la librería hasta los controladores. Para ello se tienen tres tipos de orígenes de datos: de usuario, de archivo y de sistema. Esta nomenclatura es motivo frecuente de confusión: cuando desde Windows se abre desde el panel de control el menú de "controladores ODBC" se encuentra con esta clasificación, y cuando abre cada una de las ventanas se encuentra con los mismos contenidos. Vamos a explicarlo:

- Los orígenes de datos de usuario, realmente se refieren a las operaciones que realiza el usuario con su base de datos desde la aplicación nativa para la que han sido desarrollados, y sin realizar ningún tipo de compartición con otros usuarios. Es decir, como cuando se trabaja con MS-Access, y no se comparte la base de datos.
- Cuando se comparte la base de datos mediante un servidor de archivos compartiendo físicamente los datos almacenados en un fichero determinado hablamos de orígenes de datos de archivo. Este método permite, por ejemplo a un usuario de Oracle manejar una base de datos de MS-Access, o bien que varios usuarios puedan compartir una misma base de datos.
- Cuando no se comparte un fichero, sino que se trabaja con la base de datos a través de un sistema cliente-servidor, hablamos de un origen de datos de sistema.
- Las aplicaciones más comunes, ofrecen drivers para orígenes de datos de usuario y de archivo, para permitir a los usuarios el poder trabajar, importar y exportar datos entre diversas aplicaciones de gestión. Las aplicaciones de servidores de bases de datos,



por contra, proporcionan drivers para orígenes de datos de sistema: **PostODBC** (Servidor postgresSQL), **MyODBC** (Servidor MySQL), **InterSOLV** (Servidor InterBase), **MIMER/NET** (Servidor MIMER) y cualquier controlador iODBC pertenecen a esta última categoría.

Cada origen de datos tiene asociado un controlador, que actúa de "pasarela" entre el API y el acceso físico a los datos.

4.3.2. El API de ODBC en LINUX

El API ODBC consiste en una interfaz que implementa un método de pasar peticiones en lenguaje SQL a través de una serie de funciones. Con ODBC podemos:

- Crear punteros de conexión, especificando origen de datos y controlador.
- Efectuar conexiones con la base de datos (abrir un socket).
- Crear punteros de peticiones SQL (básicamente, obtener estructuras de datos que nos permitan insertar nuestras peticiones).
- Ejecutar peticiones sobre la base de datos.
- Recoger resultados de nuestras peticiones.
- Liberar y cerrar todos los recursos previamente asignados.

En la práctica, ni todas las aplicaciones, ni todos los controladores de orígenes de datos son capaces de gestionar todas las funcionalidades previstas por el API. Por ello se establecen los denominados "niveles de conformidad SQL" en la aplicación así como "niveles de conformidad del controlador", que permiten al administrador de orígenes de datos saber qué puede hacer tanto con el driver como con la aplicación.



4.3.2.1 El proyecto FreeODBC

El proyecto **FreeODBC** ha desarrollado una librería, denominada iODBC, que cumple con las especificaciones del API ODBC 2.0 de Microsoft, y que integra las funciones de API y de administrador de orígenes de datos. Cada servidor de bases de datos provee un driver que hace las funciones de controlador de orígenes de datos y de origen de datos de sistema específico de cada servidor de bases de datos.

Existe un fichero `$(HOME)/iodbc.ini`, que indica a la librería `libodbc.so.x.x`, los controladores de que dispone cada sistema, y cómo se accede a ellos. Todo el interfaz esta implementado mediante librerías dinámicas. El resultado de todo esto, es que el programador se encuentra con un API virtualmente idéntico al que se encontraría si estuviera trabajando en una maquina MS-Windows.

Buscando por las paginas web, podremos encontrar drivers de iODBC para casi todas las bases de datos disponibles en Linux. De hecho, Los desarrolladores de iODBC han decidido incluir en sus nuevas versiones todos los drivers de aquellas bases de datos que libremente los provean.

4.4 Proceso de conexión Cliente Windows con Servidor SGBDD MySQL LINUX

4.4.1. Instalación del soporte ODBC para MySQL



Antes que nada, es preciso que el Protocolo TCP-IP se encuentre instalado en el sistema Windows, y que la conectividad entre Windows y Linux funcione sin problemas. En la distribución de MySQL: www.mysql.com bajar el archivo myodbc-2.50.38-win95.zip que contiene los archivos de instalación correspondientes a la versión 2.50.38 así como también de archivos de ayuda, adecuados para las versiones de MySQL superiores a la 3.23.x.(septiembre del 2001). Para instalar se debe descomprimir y ejecutar el programa "setup".

4.4.2. Configuración del Driver ODBC en el cliente Windows

A continuación se describe la instalación y configuración de MyODBC bajo Windows98 Segunda Edición.

En primer lugar, ejecutamos del fichero setup.exe. Aparece el familiar install wizard. Seguidamente, desde el panel de control, seleccionamos la opción ODBC 32 bits y configuraremos las diversas DSN (data source names) de usuario, archivo y sistema, añadiendo el driver MyODBC que acabamos de instalar. Para ello haciendo clic en la opción "Add" nos aparece el cuadro de la figura 10. Seleccionamos MySQL.

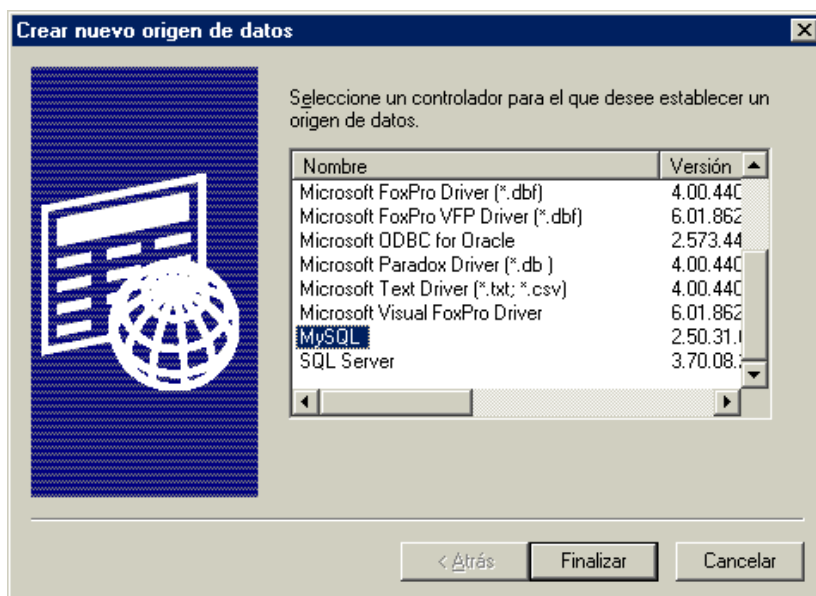




Fig. 4.3. Selección de los diversos DSN

Una vez añadidos los DSN procedemos a configurarlos. Para ello en la opción "Propiedades" nos aparece la ventana de la figura 11. En ella deberemos insertar los datos correspondientes a nuestra base de datos, nuestro servidor, el userid y el password de acceso (que recordamos son propios del servidor de bases de datos, y que no tienen por qué coincidir con cuentas de usuario)

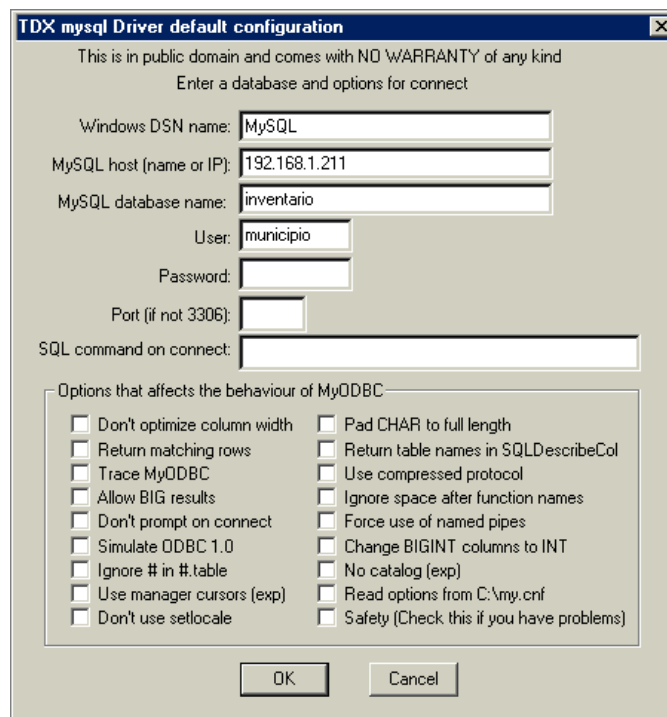


Fig.4.4. Ventana de setup del driver ODBC

Podemos seleccionar las opciones avanzadas que se encuentran en la parte inferior de la ventana que nos permitirán:

- Seleccionar opciones de acceso desde el archivo my.cnf.
- Chequeo de problemas.
- Permitir resultados grandes.
- Usar cursores.

Una vez configurado el sistema, cerrar el panel de control y reiniciar Windows.



Para que la configuración tenga éxito, el servidor de MySQL debe estar correctamente configurado y funcionando. Un telnet desde Windows al Linux al puerto 5432 (donde se instala por defecto el servidor) no debe dar lugar a "conexion refused by server".

Finalmente, ahora que se tiene el servidor y el cliente instalados, vamos a empezar a trabajar con el sistema.

4.5 Manipulación y programación del servidor MySQL sobre WINDOWS

Tomaremos como ejemplo para la conexión la base de datos de nuestra aplicación que se encuentra en el servidor de MySQL. Vamos a describir cómo se puede usar, consultar y modificar desde Microsoft Access 97/2000 corriendo como cliente dicha base de datos.

Para ello, se debe haber instalado los paquetes de acceso remoto de MS-Access. Una vez realizado, arrancamos Access y abrimos una base de datos vacía. ODBC nos permite:

- Importar una base de datos desde el servidor
- Exportar una base de datos desde Windows al servidor
- Vincular una base de datos del servidor al gestor local, de manera que podamos realizar modificaciones desde el cliente.

Primero, importaremos la base de datos. Para ello seleccionamos el menú de tablas, y desde el menú principal (o con el botón derecho) seleccionamos la opción "Importar". Aparecerá un menú que nos solicita que escojamos el tipo de datos. Escogemos "ODBC Driver" y nos aparece la ventana "Seleccionar origen de datos" (en Access 2000 seleccionamos la opción "Obtener datos externos" del menú "Archivo" y luego "importar", en el cuadro de diálogo en la



parte de “tipo de archivo” seleccionar “ODBC databases”). En el menú de "Origen de datos de sistema" escogemos MySQL (que previamente hemos configurado desde el panel de control, aunque si es necesario, Windows nos ofrece la posibilidad de seleccionar una nueva configuración desde este menú) tal y como indica la figura.

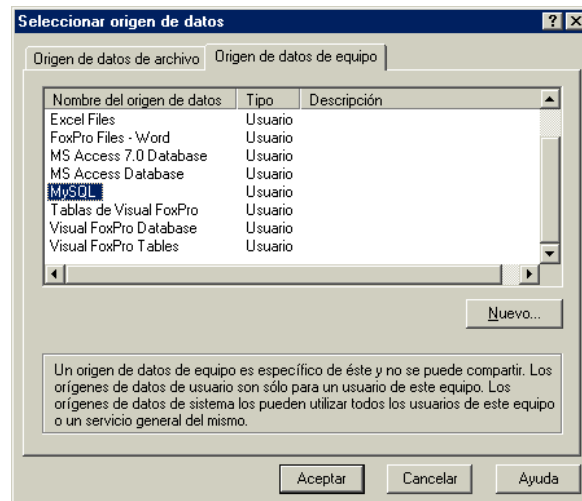


Fig. 4.5. Vinculación de datos a un DSN

Windows entonces establece la conexión y nos presenta una lista de tablas que están incluidas en la base de datos seleccionada. Escogemos la base de datos de nuestra aplicación. De tal manera que la tabla es cargada en Access y a partir de ahora se convierte en una base de datos local.

La operación que realizamos es "Importar". Esto es: coger una base de datos remota, traerla y convertirla en una base de datos local. cualquier modificación que hagamos en ella no se verá reflejada en la base de datos original del servidor. Es más, si intentamos exportar dicha tabla, el servidor indicará, que esa base de datos ya existe, y que no podemos cargarla. La única solución es exportarla con un nombre nuevo. (siempre y cuando hayamos configurado MySQL para que el usuario pueda crear y/o modificar base de datos).



Para ello, se debe realizar una vinculación, esto es, establecer un enlace entre los datos que vemos y modificamos y la base de datos. Técnicamente hablando, lo que se hace es almacenar en el servidor la secuencia de operaciones realizadas en el cliente, y cuando el usuario pulsa "Guardar" se genera una orden COMMIT que hace que todo el history se vuelque en la base de datos. En caso de que por alguna razón no se pueda realizar el COMMIT, el servidor genera el mensaje de error correspondiente, y efectúa un rollback (ver Glosario) para deshacer todos los cambios realizados desde el último "guardar".

Se realiza entonces un vínculo a la base de datos y se añade una nueva entrada. La pantalla queda tal y como se indica en la imagen de la figura siguiente. Al pulsar en "guardar", los datos son volcados al servidor. Para comprobarlo, desde un telnet al servidor, se ejecuta el interprete de sql y realizamos un select sobre la base de datos. Como se esperaba, la entrada ha sido añadida.

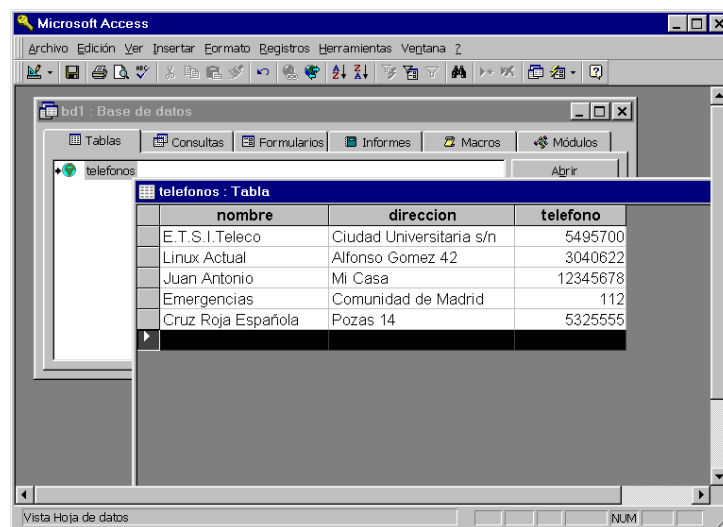


Fig. 4.6. Ejemplo de base de datos remota con MS-Access

Como se puede observar resulta un poco complicado de configurar, pero una vez realizada la configuración, el trabajo se reduce simplemente a arrancar Access y seleccionar la base de datos. Además, varios usuarios pueden trabajar a la vez con la misma base de datos, y desde



programas distintos. ODBC provee un API de programación, que en esencia es un interprete de ordenes SQL, y que permite a los programadores enganchar sus aplicaciones con el servidor de base de datos, de forma sencilla y transparente. Access no es sino otra aplicación mas.

4.6 Análisis de resultados

En una organización, los recursos de computadoras suelen estar situados en departamentos o grupos de trabajo separados, y a menudo utilizan varias topologías de red y protocolos de comunicación como también distintos sistemas operativos y aplicaciones. Es necesario entonces, integrar estos sistemas de forma que cualquier persona de la organización pueda comunicarse y compartir información y recursos, de ahí que un SGBDD debe proporcionar esta prestación para que la organización sea eficiente en la manipulación y suministro de la información, a esta prestación se la denomina *portabilidad*; Hemos visto que con MySQL resulta relativamente fácil la conexión de un cliente Windows con el servidor, las particularidades de la configuración como de la manipulación de los datos demuestran que este SGBDD facilita esta prestación.

MySQL además proporciona mucha información sobre problemas que pueda ocurrir durante la conexión, así como también de soporte a través de Internet, ya sea por e-mail, por foros, proyectos, etc. (ver ANEXO G).

De ahí que existe un motivo más para elegir a MySQL como el motor sobre el cual vamos a desarrollar nuestra aplicación, que se implementará en el I. Municipio de Huaca.



CAPITULO V

DESARROLLO DEL SISTEMA DE INFORMACIÓN MUNICIPAL BASADO EN SGBDD FREEWARE

5.1. Diagnóstico de la situación actual del Ilustre

Municipio de Huaca

5.1.1 Introducción

En el mundo actual, los acontecimientos suelen desarrollarse a una velocidad vertiginosa. Un Municipio dinámico y ágil tiene mucho más fácil su acceso al éxito, pero también las decisiones deben tomarse sin dejar de considerar todos los elementos que intervienen en cada situación. Así pues la información se convierte en un bien valiosísimo



para cualquier actividad, y los medios e comunicación son elementos vitales de un Municipio, pues ponen la información a disposición del usuario y deben hacerlo en forma rápida, y eficaz, y precisa.

La incorporación de la informática, de forma generalizada, a la gestión Municipal a supuesto una nueva forma de ver las cosas. Tanto los medianos como los grandes Municipios, han realizado un serio esfuerzo para automatizar determinadas tareas. Empujados por la necesidad de aumentar la eficacia y agilidad a la atención del público, los responsables han introducido los medios informáticos, primero en funciones rutinarias, como contabilidad, gestión de bodegas, Avalúos y Catastros, etc., para incorporarlos más adelante en un gran Sistema Integrado sobre una interconexión de Redes.

Esto no sucede con municipios pequeños, los mismos que por falta de recursos económicos no han podido realizar un plan estratégico informático coherente con la tecnología actual, este es el caso del Ilustre Municipio de Huaca, el mismo que por ser relativamente nuevo (cinco años al momento de terminar el trabajo), todos sus Sistemas administrativos son realizados manualmente, a excepción de algunos departamentos que poseen ciertos sistemas informáticos, pero que, por su plataforma de ejecución (MS-DOS), no son lo suficientemente óptimos para realizar las tareas asignadas, ni para una futura integración de Sistemas.

5.1.2. Información General del Ilustre Municipio de Huaca(IMH)

Ubicación

- 📍 **Provincia:** Carchi
- 📍 **Cantón :** San Pedro de Huaca
- 📍 **Dirección:** Calle 8 de Diciembre



Organigrama Estructural Del IMH

El Ilustre Municipio de Huaca tiene una estructura jerárquica, el mismo que a la cabeza se encuentra el Consejo Municipal, dirigido por el Sr. Alcalde Arquitecto Francisco Salazar. Formado también por departamentos, cada cual con su respectivo jefe departamental, como lo muestra el ANEXO E.

5.1.3 Recursos Informáticos

a) Hardware

COMPUTADORAS									
Departamentos	Marca	Procesador	Memoria RAM	Disco Duro	CD ROM	Multimedia	Tarjeta de Red	Fax Modem	Monitor
Alcaldía	DTK	Pentium 166 Mhz	16 MB	1.6 GB	8x	♣	<input checked="" type="checkbox"/>	♣	SVGA 14"
Secretaria	DTK	Pentium-s 120 Mhz	7 MB	1.6 GB	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	SVGA 14"
Financiero	DTK	AMD-K5 100 Mhz	15 MB	4.3 GB	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	SVGA 14"
Avaluos y C	DTK	AMD-K5 100 Mhz	7 MB	4.3 GB	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	SVGA 14"
	DTK	Pentium-s 100 Mhz	7 MB	421 MB	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	SVGA 14"
Tesoreria	DTK	AMD-K5 100 Mhz	7 MB	6.4 GB	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	SVGA 14"
Comisaria	DTK	AMD-K5 100 Mhz	7 MB	6.4 GB	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	SVGA 14"
Contabilidad	DTK	AMD-K5 100 Mhz	32 MB	4.3 GB	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	SVGA 14"
OO.PP	DTK	AMD-K5 100 Mhz	7 MB	6.4 GB	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	SVGA 14"
	DTK	Pentium-s 120 Mhz	7 MB	4.3 GB	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	SVGA 14"

IMPRESORAS			
Departamentos	Marca	Modelo	Tipo
Alcaldía	Okidata	OL-830	Laser
Secretaria	Epson	LQ-1070+	Matricial
Financiero	Epson	LQ-1070+	Matricial
Avaluos y C	Epson	LX-300	Matricial
	Citizen	GSX-190	Matricial
Tesoreria	Epson	LX-300	Matricial
Comisaria	Epson	LX-300	Matricial
Contabilidad	Epson	LQ-1070+	Matricial
OO.PP	Epson	LX-300	Matricial
	Epson	LQ-1070+	Matricial



b) Software

SOFTWARE				
Departamentos	SS00	Ver. OFFICE	APLICACIONES DOS	DONACIONES
Alcaldia	Windows 95	Office 4.0	WordPerfect 6.0, QPRO 4.0	<input checked="" type="checkbox"/>
Secretaria	Windows 95	Office 4.0	WordPerfect 6.0, QPRO 4.0	<input checked="" type="checkbox"/>
Financiero	Windows 95	Office 4.0	WordPerfect 6.0, QPRO 4.0	<input checked="" type="checkbox"/>
Avaluos y C	Windows 95	Office 95	WordPerfect 6.0, QPRO 4.0	SICPU Ver. 2.0
Tesoreria	Windows 95	Office 97	WordPerfect 6.0, QPRO 4.0	SICPU Ver. 2.0
Comisaria	Windows 95	Office 97	WordPerfect 6.0, QPRO 4.0	<input checked="" type="checkbox"/>
Contabilidad	Windows 95	Office 97	WordPerfect 6.0, QPRO 4.0	<input checked="" type="checkbox"/>
OO.PP	Windows 95	Office 95	WordPerfect 6.0, QPRO 4.0	<input checked="" type="checkbox"/>

* SICPU Sistema de Avaluos y catastros Urbano donado por Ministerio de Finanzas

Como se puede observar de los cuadros anteriores, el IMH cuenta con recursos informáticos de hardware de mediana capacidad y software relativamente caducos, en especial aquellos basados en DOS.

c) Sistemas Informáticos y Redes

En la actualidad el IMH no cuenta con una red Informática que unifique sus departamentos, como consecuencia de esto no existe Sistemas Integrados para el desempeño en la administración Municipal y peor aun para la atención al cliente, detallamos a continuación la situación informática de cada departamento.

5.1.4 Distribución y Situación de los Recursos Informáticos

Departamento Financiero

Su función es de administrar los Recursos económicos de Municipio. No existe un Sistema Financiero como tal, por lo que el procesamiento de datos es realizado en base a



macros de Microsoft Excel 4.0 reduciendo de esta manera el rendimiento eficaz de las gestiones asignadas a este departamento.

Departamento de Contabilidad

Este departamento posee un Sistema de Contabilidad sin licencia de uso, basado en DOS que hasta el momento de la realización de este informe no era aplicado aún, ahora considerando que posee una computadora DTK de 32 MB en RAM y Procesador Pentium de 120 Mhz., se puede concluir que no se esta alcanzando el máximo provecho de esta PC.

Departamento Avalúos y Catastros

Trabaja con un Sistema donado por el Ministerio de Finanzas, cuyo nombre es SICPU bajo DOS hecho en Fox Pro 2.6 , desarrollado para procesamiento de Avalúos y Catastros Urbanos. Además se trabaja en Excel para la Facturación de agua potable del Cantón.

Departamento de OOPP

Cuenta con dos computadoras, una de las cuales esta asignada a la Srta. Secretaria, cuyas funciones solamente están restringidas a la realización de oficios, solicitudes, cartas, etc., la otra PC desarrolla actividades para el calculo de costos de obras que se van a realizar por parte del municipio, el mismo que esta desarrollado mediante macros en QPRO Ver.4.0

Inventarios y Bodegas

Todos los procesos de adquisición y entrega de materiales y bienes muebles que se efectúan en este departamento, son desarrollados en forma manual arriesgándose



muchos fallas que puedan ocurrir, como pérdida de documentos, desorganización de los materiales, etc.

Comisaría

Este departamento, posee una computadora en la cual se realiza funciones de procesamiento de texto únicamente.

Tesorería

Los cobros tanto de los impuestos prediales, catastros como de las facturas del agua potable se realizan en forma manual en un 80%. Existe también una copia del Sistema de Avalúos SICPU (el mismo del departamento de avalúos y catastros) con los datos actualizados por año, alimentado mediante el envío de un disquete de departamento a departamento. Se realiza también el cobro de las planillas de agua potable mediante facturas provenientes del departamento de Avalúos. De idéntica forma sucede con el departamento de Contabilidad, cuyos informes de ingresos y egresos son entregados al final de cada día, en un disquete o escrito para que sean procesados por este.

Secretaría de la Alcaldía

De igual forma que lo anterior, las funciones de la computadora esta restringido al procesamiento de textos.

5.1.5 Conclusiones del Diagnóstico

Como podemos observar, la situación informática del municipio es precaria, debido a la falta de equipos y Sistemas Informáticos especializados y actualizados, necesarios para el buen desempeño de las actividades, como también de la atención al público. Los departamentos son islas en la organización, pudiéndose observar fácilmente que puede



existir duplicación de datos, lentitud en la realización de los procesos y tareas repetitivas que fácilmente pueden ser automatizados. El procesamiento de la información es realizado mediante programas no dedicados exclusivamente a su tarea (como por ejemplo Excel y Qpro) con sus respectivas limitaciones que puedan tener, o con programas que no permiten la conexión en red, ni el soporte de datos compartidos sobre una sola base.

El Cantón San Pedro de Huaca está viviendo una etapa de gran crecimiento poblacional y económica, por lo que, muy pronto de lo previsto el Ilustre Municipio deberá soportar una gran cantidad de problemas generados por el aumento en el volumen de los datos y la gran cantidad de transacciones que estos requerirían. De ser así, el municipio deberá mirar el futuro, e iniciar de inmediato un plan estratégico informático que comprenda: Desarrollo de Sistemas Informáticos Integrados y la Interconexión departamental, para de esta manera mejorar el servicio a la comunidad en general.

Nuestro proyecto, vendría a solucionar parte de esto, ya que nuestra propuesta consiste en desarrollar un sistema para el departamento de Bodegas, sobre la mejor opción de los Sistemas de Gestión de Base de Datos para Linux con tecnología Cliente / Servidor, que reduce significativamente el costo para su desarrollo por ser gratuitos. Sería una magnífica oportunidad para que el municipio continúe con este proyecto.

5.2. Desarrollo del Sistema de Información de Inventarios y Bodegas del SIB-IMH

La metodología del desarrollo de sistemas es el camino que siguen los analistas de sistemas al realizar su trabajo. En la figura siguiente se muestra un esquema general de la metodología para el desarrollo de sistemas a utilizar en el IMH :

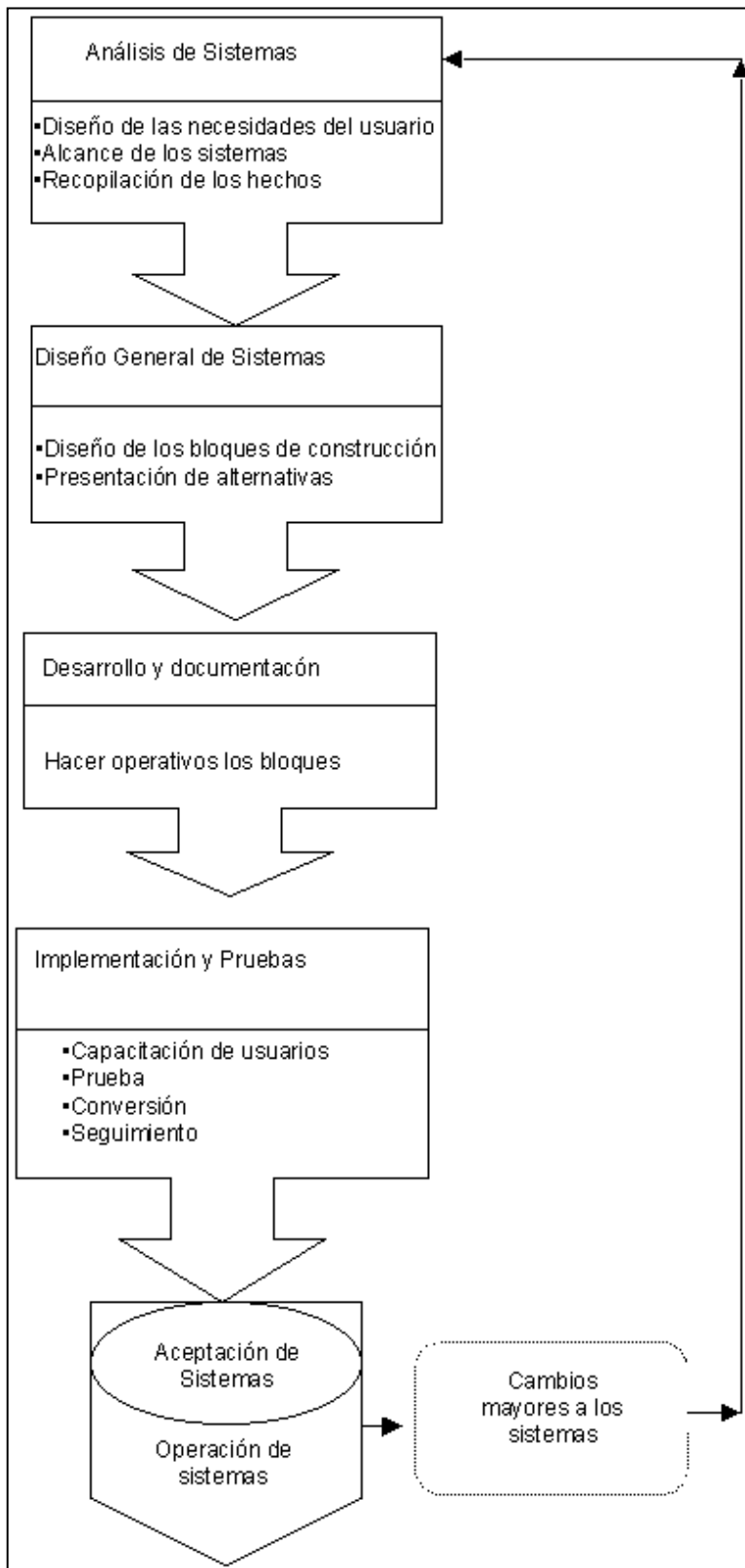


Fig.5.1. Metodología a seguir para el desarrollo del Sistema de Inventarios y Bodegas IMH







Sus fases principales son: Análisis de sistemas, el diseño general de sistemas, desarrollo y documentación de sistemas y la implementación y pruebas de sistemas. Dentro de cada fase se incluyen las principales actividades o tareas. Las primeras dos fases están dirigidas a proporcionar valores específicos para los componentes estructurales, la fase siguiente se ocupa de que los componentes estructurales sean funcionales y finalmente la fase de implementación y prueba que corresponde a la capacitación y educación de los usuarios, la prueba y conversión para hacer que el sistema sea operacional.

5.2.1. Análisis del Sistema de Inventarios y Bodegas(SIB)





Introducción

El departamento de Bodegas realiza actividades de inventarios y Kardex tanto de materiales (repuestos, suministros de oficina, accesorios eléctricos. etc) como de los bienes muebles existentes en el Municipio (computadoras, escritorios, maquinas eléctricas, etc). Procesos que los efectúa una sola persona y en forma manual, requiriendo de esta manera tiempo y recursos tanto materiales como humanos, se ha detectado que se duplica innecesariamente varios procesos y los reportes son difíciles de realizar por lo que pueden ir con algunos errores, de tal forma que se ha visto conveniente el desarrollo e implementación de un Sistema de Información que automatice las operaciones.








Razones para la realización del Sistema

-  Ahorro de Recursos materiales (tarjetas de Kardex, carpetas, etc.)
-  Tiempo de Respuesta más corto.
-  Reportes confiables, eliminación de errores.
-  Adicionar reportes que antes no estaban disponibles.



-  Simplificación de procesos para eliminar pasos innecesarios.
-  Integración de procesos.
-  Mejorar la integración entre sistemas y subsistemas.
-  Aprovechar la tecnología existente.

Problemas Encontrados

-  Existe una gran cantidad de tarjetas de kardex, ocupando mucho espacio físico.
-  Por lo expuesto anteriormente se complica la organización de las tarjetas de una manera más adecuada y eficiente.
-  Al solicitar un material, el personal encargado debe comprobar la existencia o no de lo solicitado, por lo que el tiempo de respuesta es muy alto.
-  Los reportes se reducen en tan solo al informe anual de existencias de los materiales y de los bienes muebles.
-  Muchos de los procesos son repetitivos pudiéndose producir duplicación de datos como por ejemplo al ingresar un bien mueble se añade tanto en un registro de inventarios como en Kardex.
-  La comunicación entre el Financiero y Contabilidad con respecto a los ingresos y egresos de materiales y bienes muebles se lo realiza en forma verbal y mediante documentos poco eficientes para dejar constancia de la transacción realizada.
-  Al departamento se le ha asignado una computadora, el mismo que es utilizado solamente para procesamiento de textos e impresión de informes en Excel.



Objetivos

- 👉 Reducir costos en recursos materiales.
- 👉 Reducir el tiempo de procesamiento y reporte de datos.
- 👉 Iniciar con la integración departamental con la utilización de herramientas que faciliten este objetivo como Linux y MySQL.
- 👉 Mejorar la calidad de servicio a los usuarios.

Alcance

El sistema comprenderá el procesamiento y reporte de datos de inventarios de los bienes muebles y materiales fungibles que posee el Municipio, dentro del departamento de Bodega, con la posibilidad de integrarse con los demás subsistemas que se vayan desarrollando a futuro en los diferentes departamentos.

Determinación de Factibilidades

a) Factibilidad Técnica

Para el desarrollo e implementación del Sistema necesitamos los siguientes Recursos Informáticos:




Hardware

Se necesitará la instalación de una Red LAN teniendo como Servidor una PC con las siguientes características:

- 📄 Pentium III de 550 Mhz o superior.
- 📄 Memoria RAM de 64 MB.
- 📄 Disco Duro: 10 GB.
- 📄 CD ROM de 32x o superior.







Y como mínimo una terminal inteligente conectada a esta, con las siguientes características:

-  Procesador Pentium I de 100 Mhz como mínimo.
-  Memoria RAM de 16 MB o superior.
-  Disco duro de 2 GB.




Además se requerirá una impresora Epson LX-300 para impresión de reportes.

Software

El software a nivel de Servidor se necesitará lo siguiente:

-  Sistema Operativo: RedHat Linux 7
-  Sistema de Gestión de Base de Datos MySQL Ver. 3.23.x
-  KMySQL Ver 1.1
-  XMySQLadmin Ver. 0.4.0

A nivel de cliente:

-  Sistema Operativo: Windows 9x
-  Front End: Visual Basic 6.0
-  MyODBC Ver. 2.5.X de MySQL

Personal técnico

Se requerirá de un profesional Ingeniero en Sistema Computacionales para el mantenimiento, administración y control tanto de la Red como del Sistema; con su respectivo auxiliar.

b) Factibilidad Económica

El desarrollo e implementación del Sistema generará los siguientes costos:



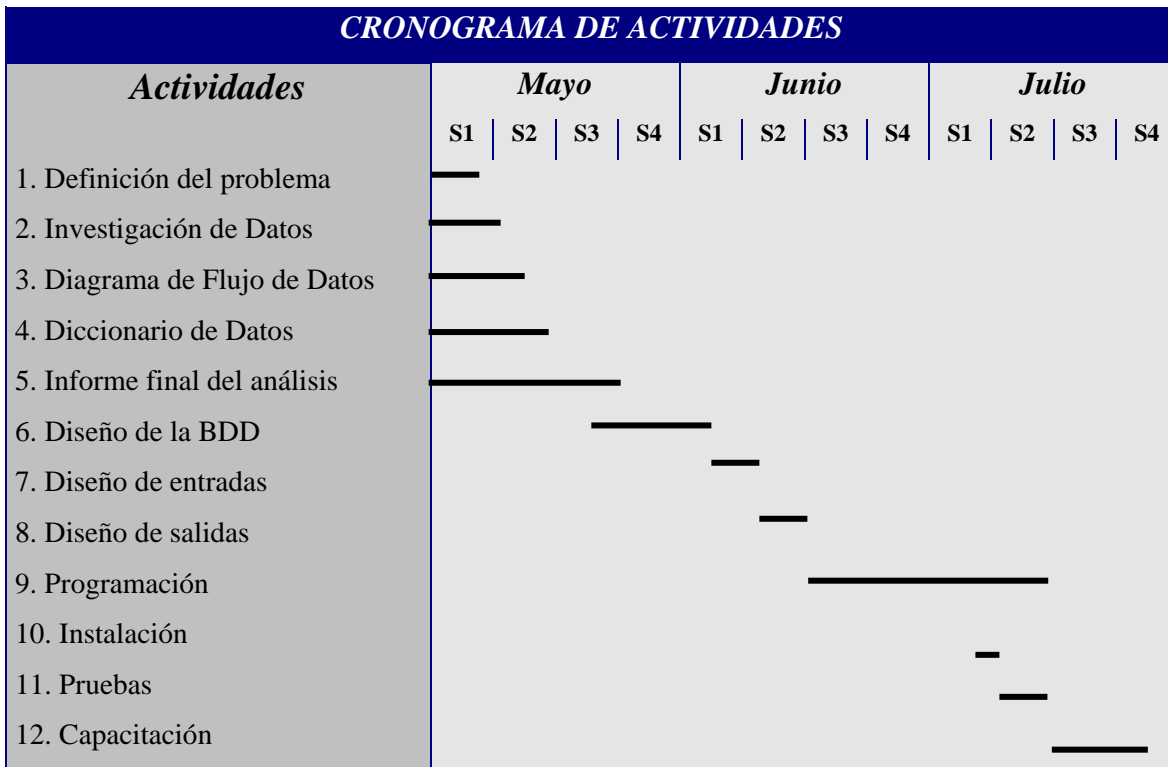
Recurso	Costo
💰 Servidor	\$ 1500
💰 Cliente(actualización)	\$ 400
💰 Personal (Ingeniero y auxiliar)	\$ 400
💰 Hardware de Red	\$ 300
TOTAL	\$2600

Cabe destacar que el Software necesario sobre el cual se va a desarrollar el Sistema es completamente gratuito existiendo un ahorro considerable de divisas, lo que no sucedería si se empleara software comercial como Windows NT que tiene un costo de \$2000 para cuatro usuarios.

Además estos costos se verán recuperados a un corto plazo debido a que se ahorra en la compra de materiales de escritorio mensuales (como tarjetas de Kardex, esferos, carpetas, etc.), y más aún si consideramos que el personal técnico seguiría en el desarrollo del proyecto de interconexión del Ilustre Municipio de Huaca, vital para el mejoramiento del desempeño de las actividades del Municipio, sin olvidarnos de la mejor atención que se brindaría a la comunidad con servicios eficientes y rápidos.



c) Factibilidad Operativa



El Sistema se realizará en un tiempo estimado de tres meses a partir de la aprobación del proyecto, de acuerdo al siguiente plan de actividades:

Documentación Adicional

Diagramas de Flujo de datos ver ANEXO F

5.2.2. Diseño del Sistema

El Diseño de Sistemas es el proceso de aplicar ciertas técnicas y principios con el propósito de definir un Sistema, con suficientes detalles como para permitir su interpretación y realización física.



La etapa del Diseño del Sistema lo encerraremos en las siguientes fases:

1. Diseño de Datos
2. Tecnología
3. Entradas
- 4 . Salidas

5.2.2.1. Diseño de Datos

Trasforma el modelo de dominio de la información, creado durante el análisis, en las estructuras de datos necesarios para implementar el Software. Para cumplir con los objetivos de modelar los datos hemos desarrollado el siguiente diagrama bloques, con la cual identificaremos los objetos y los procesos que intervendrá en el Sistema de Inventarios y Bodegas:

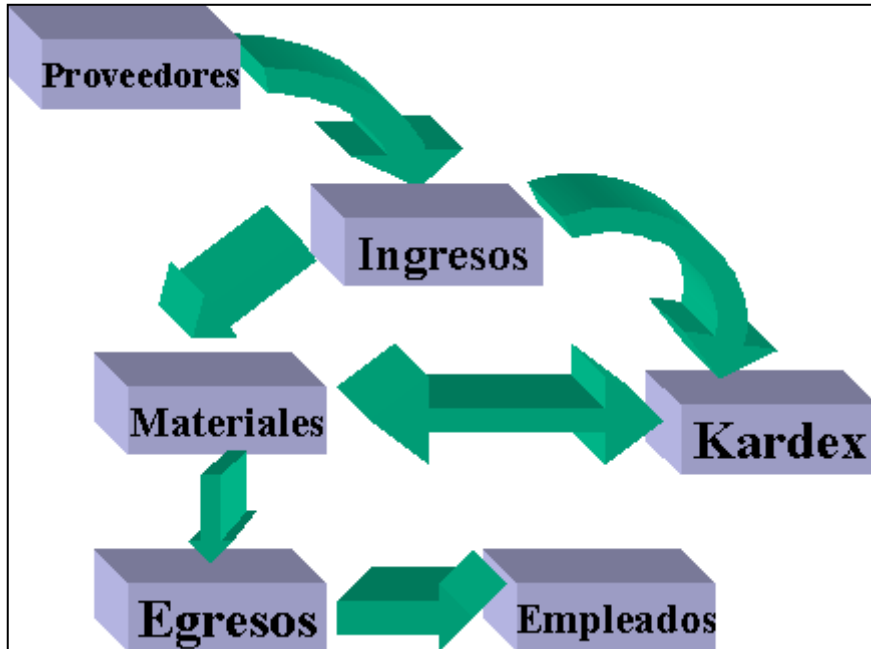


Fig. 5.2. Diagrama de Bloques del Sistema de Inventarios y Bodegas IMH



Del diagrama anterior determinamos entonces la siguiente estructura que tendrá la

Base de Datos del Sistema de Inventarios y Bodegas con el nombre de **inventariosimh**:

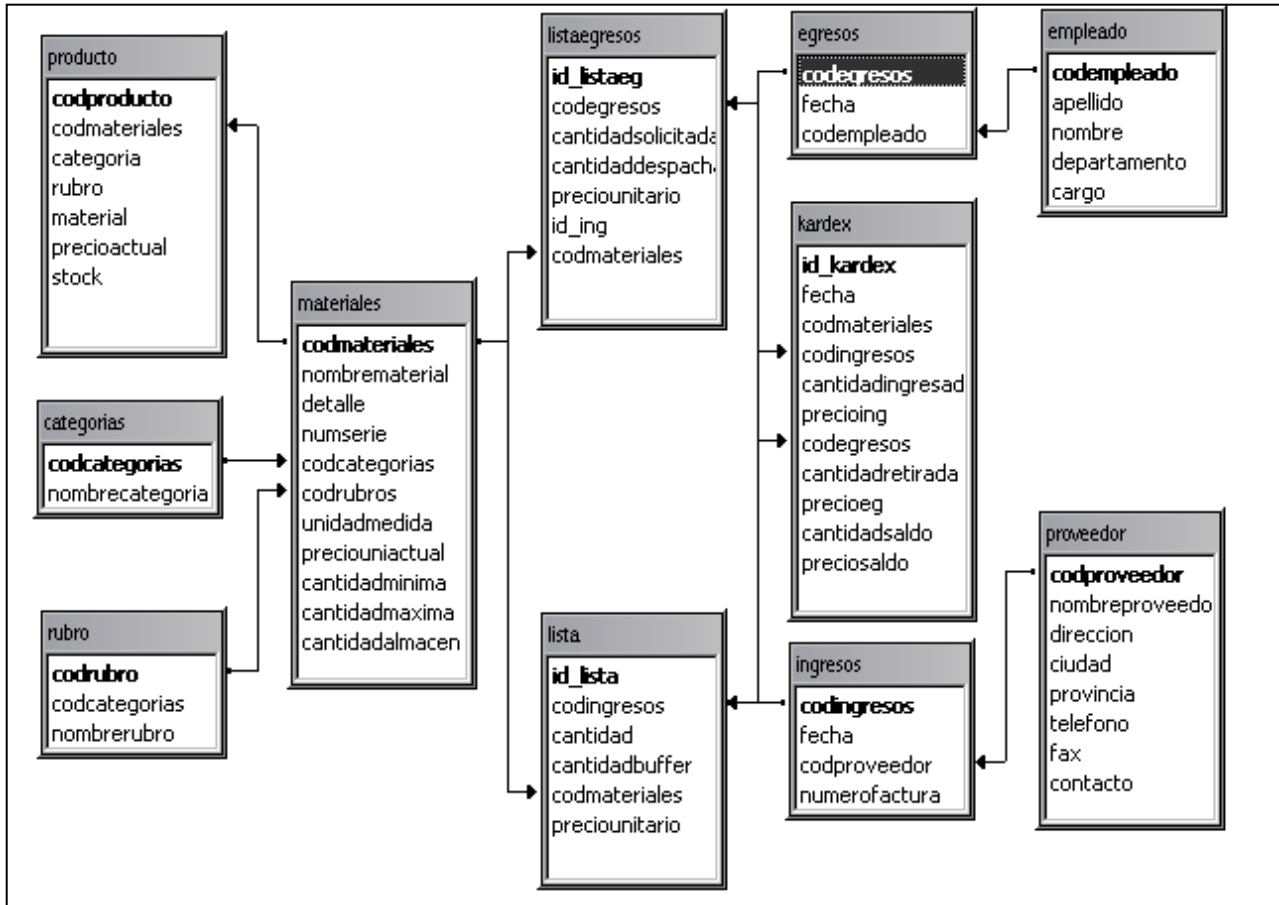


Fig. 5.3. Tablas del Sistema de Inventarios y Bodegas IMH

Una vez obtenido la estructura física de la Base de Datos procedemos a continuación a crearle en el motor MySQL, utilizando para ello el monitor de ejecución de instrucciones SQL, el mismo que se localizará en la máquina sobre el cual se está ejecutando el servidor MySQL (cabe recordar que MySQL no necesariamente debe ejecutarse en la máquina servidor de la red).

Creamos entonces primero la Base de Datos **InventarioIMH** en el monitor, con el siguiente comando:

```
mysql> create database InventarioIMH;
```



Query OK, 1 row affected (0.00 sec)

mysql/>>_

Una vez creado la Base, creamos ahora las respectivas tablas, utilizando un archivo script con el siguiente contenido:

```
/* CREACION DE TABLES */  
  
CREATE TABLE proveedor(  
codproveedor char(15) NOT NULL PRIMARY KEY,  
nombreproveedor CHAR(40) NOT NULL,  
direccion CHAR(40) default " ",  
ciudad CHAR(20) default "Tulcán",  
provincia CHAR(20) default "Carchi",  
telefono CHAR(15) default "000-000 ",  
fax CHAR(15) default "000-000 ",  
contacto CHAR(40) default "Distribuidor"  
);  
  
CREATE TABLE empleado(  
codempleado char(15) NOT NULL PRIMARY KEY,  
apellido CHAR(40) NOT NULL,  
nombre CHAR(40) default " ",  
departamento CHAR(20) default "Varios",  
cargo CHAR(20) default "Empleado"  
);
```



```
CREATE TABLE rubro (  
    codrubro char(15) NOT NULL PRIMARY KEY,  
    codcategorias char(15) NOT NULL,  
    nombrerubro CHAR(40) NOT NULL  
);
```

```
CREATE TABLE categorias (  
    codcategorias char(15) NOT NULL PRIMARY KEY,  
    nombrecategoria CHAR(40) NOT NULL  
);
```

```
CREATE TABLE materiales (  
    codmateriales char(15) NOT NULL PRIMARY KEY,  
    nombrematerial CHAR(40) NOT NULL,  
    detalle CHAR(100),  
    numserie CHAR(25) NOT NULL default "Sin Serie",  
    codcategorias char(15) NOT NULL,  
    codrubros char(15) NOT NULL,  
    unidadmedida CHAR(15) default "Unidades",  
    preciouniactual float UNSIGNED default "0.0",  
    cantidadminima int UNSIGNED default "0",  
    cantidadmaxima int UNSIGNED default "0",  
    cantidadalmacen int UNSIGNED default "0"  
);
```



```
CREATE TABLE egresos (  
    codegresos int unsigned NOT NULL PRIMARY KEY,  
    fecha DATE NOT NULL,  
    codempleado char(15) NOT NULL  
);
```

```
CREATE TABLE lista (  
    id_lista int auto_increment primary key,  
    codingresos int unsigned NOT NULL,  
    cantidad int UNSIGNED default "0",  
    cantidadbuffer int UNSIGNED default "0",  
    codmateriales char(15) NOT NULL,  
    preciounitario float UNSIGNED default "0.0"  
);
```

```
CREATE TABLE kardex(  
    id_kardex int auto_increment primary key,  
    fecha DATE NOT NULL,  
    codmateriales char(15) NOT NULL,  
    codingresos int unsigned default "0",  
    cantidadingresada int UNSIGNED default "0",  
    precioing float UNSIGNED default "0.0",  
    codegresos int unsigned default "0",  
    cantidadretirada int UNSIGNED default "0",  
    precioeg float unsigned default "0.0",  
    cantidadsaldo int UNSIGNED default "0",
```



```
preciosaldo float default "0.0"
```

```
);
```

```
CREATE TABLE producto(
```

```
codproducto int auto_increment primary key,
```

```
codmateriales char(15) NOT NULL,
```

```
categoria CHAR(40) NOT NULL,
```

```
rubro CHAR(40) NOT NULL,
```

```
material CHAR(40) NOT NULL,
```

```
precioactual float UNSIGNED default "0.0",
```

```
stock int UNSIGNED default "0"
```

```
);
```

```
CREATE TABLE ingresos (
```

```
codingresos int unsigned NOT NULL PRIMARY KEY,
```

```
fecha DATE NOT NULL,
```

```
codproveedor char(15) NOT NULL,
```

```
numerofactura char(25) default "00000"
```

```
);
```

```
CREATE TABLE listaegresos (
```

```
id_listaeg int auto_increment primary key,
```

```
codegresos int unsigned NOT NULL,
```

```
cantidadsolicitada int UNSIGNED default "0",
```

```
cantidaddespachada int UNSIGNED default "0",
```

```
preciounitario float UNSIGNED default "0.0",
```

```
id_ing int,
```



```
codmateriales char(15) NOT NULL
```

```
);
```

```
CREATE TABLE clave(
```

```
claveacceso char(15) NOT NULL,
```

```
usuario CHAR(15) NOT NULL
```

```
);
```

Ejecutamos ahora el siguiente comando:

```
> source TablasSistemaIMH.txt;
```

Con el siguiente comando podemos comprobar la creación o no de las respectivas tablas:

```
mysql> show tables;
```

```
+-----+
| Tables in inventarioimh |
+-----+
| categorias |
| clave      |
| egresos   |
| empleado  |
| ingresos  |
| kardex    |
| lista     |
| listaegresos |
| materiales |
| producto  |
| proveedor |
| rubro     |
+-----+
```



12 rows in set (0.00 sec)

mysql> _

Estamos listos ahora para la conexión con el Front-End desarrollado en Visual Basic a través de ODBC (más detalles de la conexión ODBC ver sección 4.4).

5.2.2.2. Tecnología

Esta fase consiste en determinar los componentes principales que son: Hardware, Redes y Software. Con respecto al hardware y a redes habíamos ya identificado los equipos con que dispone el municipio, así como también de su distribución. (ver sección 5.1.2.3).

Sobre el software sobre el cual se desarrolló el Sistema SIB-IMH lo detallamos a continuación:

Sistema Operativo	Servidor: RedHat Linux 7.0 Cliente: Windows 9x
Lenguaje de Programación para Front-End	Microsoft Visual Basic 6.0
Motor de Base de Datos	MySQL Ver. 3.23.x para Linux
Controladores ODBC	MyODBC Ver. 3.50.36
Generador de Reportes	Cristal Report 4.6
Generador de Instaladores	Herramienta de Empaquetado y Distribución de Visual Basic

5.2.2.3. Entradas

Las entradas son los ingresos del sistema que pueden ser recursos materiales, recursos humanos o información. Las entradas constituyen la fuerza de arranque que suministra al sistema sus necesidades operativas. Se presenta a continuación los



formularios utilizados para los ingresos de los datos al Sistema, cabe indicar que cada formulario se indica detalladamente más adelante en el manual del usuario (anexo G):

No	Nombre
1	Inicio de Sesión
2	Pantalla Principal del Sistema SIB-IMH
3	Sección Ingresos de Materiales
4	Sección Egresos de Materiales
5	Sección Información General

5.2.2.4. Salidas

En este caso salida se refiere a los resultados e informaciones generadas por el Sistema, Para la mayoría de los usuarios la salida es la única razón para el desarrollo de un Sistema y la base de evaluación de su utilidad. Dividiremos a los formularios que permiten la salida de los datos en dos aspectos:

a. Por Pantalla

No	Nombre
1	Sección Kardex
2	Sección Reportes
3	Sección Inventarios

b. Por Impresora

Los formularios que describimos a continuación provee también una opción de imprimir las respectivas órdenes de ingresos y egresos, así como también del Kardex:



No	Nombre
1	Sección Ingresos de Materiales
2	Sección Egresos de Materiales
3	Sección Kardex

5.2.3. Desarrollo y Documentación del sistema

El Sistema de Inventarios y Bodegas SIB-IMH propuesto para el desenvolvimiento eficiente de la oficina de Bodegas del Ilustre Municipio de San Pedro de Huaca se desarrolló por el lapso de seis meses, desde el mes de febrero hasta agosto del año 2001. Para el desarrollo del sistema se utilizó como Lenguaje: Visual Basic 6.0, y el control ADO para la conexión con el servidor MySQL, que permite realizar un acceso a datos externos de manera eficiente y de alto rendimiento con ODBC. pero para esto se debe definir primero los parámetros de conexión ODBC de los controladores MyODBC proporcionados por MySQL (para más detalles ver sección 4.4), los principales son:

- Pad CHAR to full length. Permite que el servidor MySQL no añada espacios en blanco a los campos de la Base de Datos de Tipo carácter, al momento de realizar las consultas y presentar en la aplicación.
- Don't optimize column width. Es un parámetro idéntico al anterior, el mismo que tiene la función de no añadir espacios pero al realizar las respectivas actualizaciones desde la aplicación.

Una vez hecho esto, se procede a realizar el código que permitirá realizar los procesos de ingresos, egresos, kardex, así como de los reportes respectivos requeridos por el usuario.



5.2.4. Instalación y Pruebas del sistema

La implantación del sistema es la última fase del desarrollo de Sistemas. Es el proceso de instalar equipos o Software nuevo, como resultado de un análisis y diseño previo de la sustitución o mejoramiento de la forma de llevar a cabo un proceso automatizado.

Al Implantar un Sistema de Información lo primero que debemos hacer es asegurarnos que el Sistema sea operacional o sea que funcione de acuerdo a los requerimientos del análisis y permitir que los usuarios puedan operarlo.

Así para la correcta instalación del Sistema de Inventarios y Bodegas SIB-IMH en una máquina cliente Windows se necesitará de los siguientes componentes:

- Controladores MyODBC 2.50.x
- El servidor MySQL en plena ejecución.
- El paquete de Instaladores del sistema, que copiará los archivos OCX (controles Active X), RPT (reportes), HLP (ayuda) y EXE (ejecutable de la aplicación), en el disco duro.
- Un programa administrador Cliente del servidor para resolver posibles errores en la parte cliente, en este caso Artronic admin. 2.1.

Cabe indicar que en la preparación de la Implantación, aunque el Sistema esté bien diseñado y desarrollado correctamente, su éxito dependerá de su implantación y ejecución por lo que es importante capacitar al usuario con respecto a su uso y mantenimiento.

Así después de esto se estableció un período de prueba para identificar puntos débiles y fuertes del Sistema implantado, mismos que se enfocaron en dos puntos estratégicos:



1. Se puso a prueba la manera en que funciona el Sistema, esto incluye su facilidad de uso, Tiempo de respuesta ante una necesidad o proceso, como se adecuan los formatos en que se presenta la Información, inventario de materiales y su nivel de Utilidad.
2. Comprobar si los controles permitieron que el usuario pueda actualizar, modificar, añadir, reducir materiales al stock de bodega sin tener problema alguno.
3. Finalmente, determinar si el proceso de Kardex es realizado como el usuario espera.

Al observar los componentes y las relaciones involucradas en el sistema desde la definición del problema hasta las fases de Implementar y probar, determinamos que el sistema se ha desarrollado cumpliendo con todos los requerimientos que se encontraron en la oficina de Bodegas del Ilustre Municipio de san Pedro de Huaca.



CAPITULO VI

CONCLUSIONES Y RECOMENDACIONES

6.1. Conclusiones

6.1.1 Conclusiones sobre la plataforma LINUX y los SGBDD Freeware

1. Linux está en un constante proceso de mejora y actualización, para soportar cualquier novedad que pueda surgir en lo que al hardware se refiere, permitiendo al usuario o a una institución que lo adopte, tener más estabilidad, rendimiento o productividad.
2. El empleo de los Sistemas de Gestión de Bases de Datos Freeware para Linux reducen considerablemente el coste en el desarrollo de aplicaciones informáticas en una empresa privada o pública como es el caso del Ilustre Municipio de San Pedro de Huaca, por estar protegido por la licencia GNU.



3. Podemos añadir además que son motores robustos, fáciles de instalar, no necesitan estar instalados en la máquina servidor de la red, ocupan poco espacio en disco así como también no necesitan muchos requerimientos de hardware para su perfecta ejecución.
4. Tanto PostgreSQL como MySQL son enormemente conocidas por los desarrolladores en Internet, quienes ofrecen ayudas, manuales, tips, foros; de tal manera que se puede decir que una empresa no está sola si se encamina a emplear estos motores como base para sus sistemas informáticos integrados.
5. Si bien es cierto, MIMER e INTERBASE no tienen muchos adeptos ni considerables características, no pueden quedarse atrás, ya que según sus promotores se esfuerzan para desarrollarlos y robustecerlos cada día más, a tal punto de que puede llegar algún día a tener tanta fiabilidad como PostgreSQL y MySQL.
6. Para la selección de uno u otro motor de Bases de Datos debe considerarse los requerimientos que la empresa considere de importancia, así, para una selección entre PostgreSQL y MySQL consideraríamos lo siguiente: si la institución demanda de mayor velocidad para el procesamiento de los datos así, como de una amplia capacidad de almacenamiento, se seleccionará MySQL, si requiere protección de los datos convincente no se dudará en seleccionar PostgreSQL, de esta manera se podrá elegir el motor que más se acerque a satisfacer las necesidades de la institución.
7. Todos los SGBDD proveen también excelentes controladores ODBC para la conexión del servidor bajo Linux con máquinas cliente Windows 9x, para el desarrollo de aplicaciones Front-Ends, que permiten conectarse con cualquier herramienta como Visual Basic, Visual C++, Fox Pro, Access, Delphi, y muchos otros; de igual manera proveen librerías API para herramientas como: C/C++, Perl, Tcl/Tk, Python en ambientes Linux/Unix,,
8. Pero nada puede estar excepto de puntos débiles, entre los más principales tenemos: falta de desarrolladores de aplicaciones gráficas para la plataforma Linux, muchas funcionalidades del SQL estándar van añadiéndose poco a poco, a la falta de estas funcionalidades el administrador deberá estar a la par con el apareamiento de los parches



- o nuevas versiones (que aparecen a un promedio de cuatro meses o menos), muchos de estos no proveen administradores de Bases de Datos gráficos (a excepción de PostgreSQL y MySQL).
9. Los SGBDD Freeware para Linux poseen gran estabilidad, seguridad, rapidez y eficiencia (como lo demuestran los tests de evaluación) tanto como sus homólogos comerciales, siendo por estas razones que muchas empresas nacionales e internacionales las han utilizado para sus Sistemas Informáticos.
 10. La cantidad de SGBDD Freeware existente en Internet es significativa y cuyo número se incrementa considerablemente.
 11. La Licencia Pública General (GPL) de la Free Software Foundation, desempeña un papel clave en la promoción y desarrollo del software libre, ya que protege al software que se distribuye bajo esta licencia, contra patentes y restricciones desfavorables para los usuarios, lo que constituye una garantía para ellos. Quienes desarrollan software propietario prohíben su copia libre, quienes desarrollan software libre prohíben que se impida su copia.

6.1.2 Conclusiones sobre el Aplicativo SIB-IMH

1. Por otro lado el sistema que se desarrolló para la oficina de bodegas del IMH, trabaja a satisfacción, teniendo como servidor de Bases de Datos MySQL, pudiendo observarse que los datos son consultados y almacenados con rapidez y principalmente con estabilidad.
2. El Sistema de inventarios y Bodegas SIB-IMH, se desarrolló de forma que pueda integrarse fácilmente hacia otros Sistemas Informáticos que en lo posterior se Implanten.
3. El Sistema resultó novedoso, intuitivo y productivo comparado con otros sistemas que son utilizados en el Municipio basados en el viejo DOS, que no aprovecha los beneficios del ambiente GUI.



4. El equipo computacional de la oficina es aprovechado al máximo, considerando que anteriormente sólo era aplicado para la realización de documentos y hojas de cálculo de informes y oficios que poco aportaba con el buen desempeño del trabajo en la oficina y el municipio en general.
5. Así también, los procesos, tareas e informes que la oficina efectúa, se realiza de manera más efectiva, rápida y segura con ahorro de recursos materiales y económicos.
6. El principal objetivo de un sistema es mejorar el servicio a los usuarios, y nuestra aplicación no esta ajeno a este, ya que fue diseñado en forma modular y realizado con herramientas de última generación.
7. Cabe indicar que el Sistema de Inventarios y Bodegas está desarrollado exclusivamente para cumplir con los reglamentos y requerimientos que el Municipio exige para cumplir con sus servicios.

6.1.3 Conclusiones sobre el I. Municipio de San Pedro de Huaca

1. Los equipos con que cuenta el Municipio, claro que con pequeños cambios y la adquisición de otros, están aptos para su integración a una red municipal, sobre el cual se ejecutaría en el futuro un Sistema Integrado que abarque los diferentes departamentos que posee la institución, y así mejorar su integración .
2. No solo el Municipio de San Pedro de Huaca, sino también muchos otros municipios o empresas pequeñas que no poseen los recursos económicos suficientes, podrían desarrollar un buen plan informático tomando en cuenta estas magníficas herramientas relativamente nuevas y “gratuitas”.
3. La posibilidad de optar por los SGBDD Freeware para sistemas informáticos municipales representa una gran oportunidad para los municipio pequeños de nuestros país, así como



también una decisión importante ante las fallas potenciales y conocidas que puede encerrar la dependencia tecnológica de las marcas propietarias como Microsoft. Lo que está en juego es la posibilidad de disponer de la tecnología al servicio de todos y no sólo al servicio de las necesidades económicas de las grandes corporaciones.

4. Otras entidades que pueden ser también favorecidos por los beneficios del software freeware, son las universidades, por la existencia de varios cientos de aplicaciones para Linux sobre los cuales se podrían realizar los estudios académicos, ya que se cuenta con su código fuente.
5. Creemos que el Municipio de San Pedro de Huaca ha acertado con aprovechar los beneficios mutuos que atrae la firma de convenios tan importantes como éste, con los cuales los estudiantes de los últimos años de nuestra facultad puedan realizar proyectos de investigación en bien de la prosperidad de los municipios y nuestro país en general.
6. El Ilustre Municipio de San Pedro de Huaca está en capacidad de albergar nuevos proyectos para su Planificación Informática para de esta manera mejorar sus servicios en forma integral y eficiente hacia sus usuarios cada día más exigentes.
7. Finalmente recomendamos a nuestra Universidad que aproveche la capacidad que poseen sus estudiantes y conjuntamente con el empleo de estas herramientas gratuitas que ofrece Internet, para el desarrollo de excelentes sistemas a ser aplicados en los distintos departamentos, y así, de esta manera, ahorrar divisas tan importantes en estos tiempos.

6.2. Recomendaciones

6.2.1 Recomendaciones sobre la plataforma LINUX y los SGBDD Freeware

1. El empleo de SGBDD para el desarrollo de Front-Ends para la manipulación de datos sugiere un conocimiento profundo sobre los diferentes parámetros que provee los



controladores ODBC, por lo que, antes de iniciar un proyecto se debería investigar aquello, para realizar una correcta conexión con el servidor.

2. En el proceso de desarrollo del sistema se encontró un falla con respecto al trabajo con el controlador ODBC de MySQL y el Front - End de Visual Basic, falla que se refiere al manejo de caracteres, pero que fácilmente se puede arreglar con los pasos indicados en la sección 5.2.3, recomendamos seguir este consejo cuando se emplee el MySQL para las aplicaciones.
3. Para una fácil instalación de los motores SGBDD, se recomienda que se utilice los paquetes RPM o DEB (si su sistema Linux pueda manejarlos), ya que estos permiten una rápida compilación y configuración del SGBDD.
4. Existen varias páginas WEB a más de las que enumeramos en los anexos, que muestran diversos artículos sobre las ventajas y desventajas del uso de los SGBDD FREEWARE, sería muy bueno que nuestros lectores se interesen por estos temas para que vayan formando su propio criterio acerca del tema que hemos puesto a consideración con nuestro trabajo.
5. Recomendamos a los profesores de nuestra escuela, el uso de los diferentes motores de Bases de Datos: Mimer, Interbase, PostgreSQL, MySQL y otros disponibles en Internet, como material didáctico para la enseñanza - aprendizaje del área de Diseño y Administración de Bases de datos, así como también del área de Programación, ya que la mayor parte de estos motores proporcionan su código fuente para su estudio y modificación respectiva.
6. De igual manera, a los estudiantes que aprovechen los beneficios no solo de los motores de Bases de Datos FREEWARE sino también de todo el software que se encuentra disponible en la Red Internet para su superación y enriquecimiento de sus conocimientos
7. Invitamos a los Municipios de nuestro país, que tomen en cuenta para sus proyectos de desarrollo informático, a que tomen en cuenta el software FREEWARE que



proporcionan diversas Fundaciones, personas y Universidades a nivel mundial, por las muchas ventajas que se han descrito a lo largo de nuestra Tesis.

6.2.2 Recomendaciones sobre el Aplicativo SIB-IMH

1. Tener un sistema independiente diferente para cada departamento del Municipio, permite la incomunicación de estos y la duplicación e incompatibilidad de datos, de ahí que se hace necesario desarrollar lo más pronto posible un **Sistema Integrado**, para que el Municipio se beneficie de las grandes capacidades que ofrece este tipos de Sistemas y de esta manera abandonar los programas basados en DOS.
2. Para esto se debería producir subsistemas con capacidad de integración, mejor aún si se continúa con el uso de el motor MySQL, que como se ha dicho, reduce su costo de producción sin requerir de licencia alguna.
3. Es recomendable complementar al Sistema de Inventarios y Bodegas SIB-IMH para que pueda realizar un control completo sobre los Inventarios de Activos Fijos y de materiales donados por empresas o fundaciones a la sociedad huaqueña, para ser distribuidos por el Municipio.
4. Consideramos también, añadir componentes de parametrización del Sistema, para poder personalizar el uso de cualquier Municipio que requiera de este Sistema con el menor cambio posible.



6.2.3 Recomendaciones sobre el I. Municipio de San

Pedro de Huaca

1. Es importante que el personal administrativo del Ilustre Municipio de Huaca tenga una constante capacitación, por lo menos en lo esencial con respecto al manejo de un Computador y más aún si ya se ha implementado un sistema informático integrado.
2. Se hace necesario continuar con una Planificación Informática Completa en el I. Municipio de Huaca a nivel de Hardware, Redes y Software, para no desaprovechar esta investigación.
3. Consideramos que el convenio establecido entre el I. Municipio de Huaca y la Universidad Técnica del Norte se mantenga para beneficio de los estudiantes así como también del municipio.
4. Sugerimos a aquellos estudiantes que deseen aportar con nuevos proyectos a esta institución pública, se acojan al convenio firmado por la Universidad y el Municipio para de esta manera, trabajar buscando siempre el beneficio mutuo de las dos partes ya que existe mucho por hacer.
5. Al realizar el estudio de la situación actual del municipio descrito en el capítulo 5 fácilmente se detecta que el Municipio necesita un departamento u oficina dedicado al desarrollo y mantenimiento de Sistemas Informáticos, tanto en Hardware y software, así como también brindar capacitación al personal que elabora en la institución.



COMPROBACION DE HIPÓTESIS

“Si se cuenta con un Sistema de Gestión de Base de Datos de gran potencia y fácil utilización que sean de libre distribución, se pueden desarrollar aplicaciones informáticas de bajo costo y alto rendimiento en organizaciones públicas y privadas de nuestro país”.

Después de realizar un exhaustivo estudio de evaluación y análisis de los SGBDD FREEWARE determinamos que su aplicación durante el proceso de desarrollo de un Sistema Integrado, reduce considerablemente su costo únicamente al pago de los analistas y desarrolladores, liberándose de las licencias costosas que por ley se debería tener.

Por otra parte, al observar los resultados de la aplicación del Benchmark, notamos que los SGBDD FREEWARE ofrecen grandes prestaciones, seguridades, compatibilidad SQL y principalmente estabilidad, semejante a lo que ofrecen sus homólogos comerciales.

Por lo tanto, cualquier institución pública o privada, en especial una entidad municipal como el Ilustre Municipio de San Pedro de Huaca, pueden poner absoluta confianza al utilizar estas herramientas disponibles en forma gratuita por Internet, demostrando de esta manera que nuestra hipótesis anteriormente descrita, ha sido satisfactoriamente comprobada.



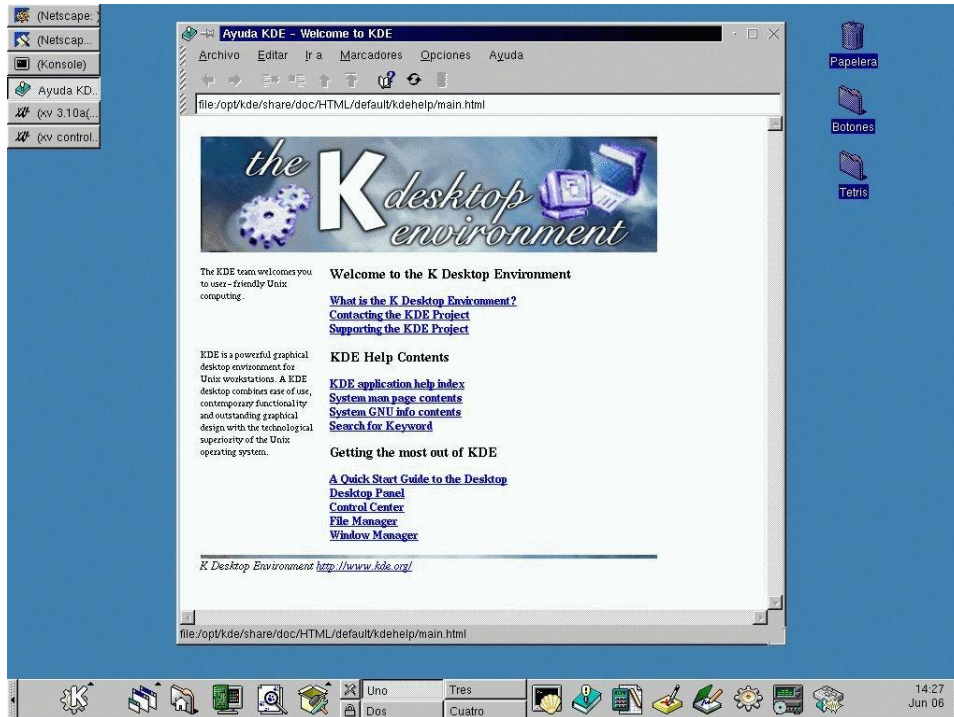
ANEXOS



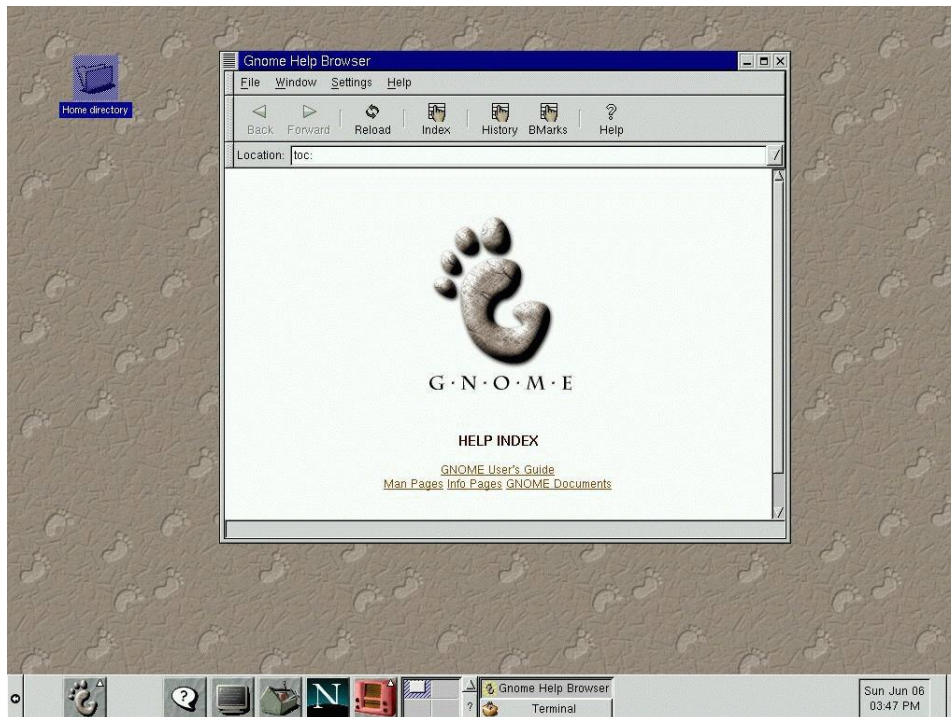
ANEXO A



ENTORNO KDE



ENTORNO GNOME





ANEXO B

VERSIONES COMERCIALES DE LINUX



Red Hat es la más popular y usada de las distribuciones, se caracteriza por su fácil instalación, es la creadora de los paquetes RPM. Su última versión es la 7.1, pero aún la 6.2 es la más usada. Su dirección en Internet es <http://www.redhat.com/>



Conectiva es una versión supuestamente en español, esta hecha en Brasil y es basada en el Red Hat, todavía no tiene muchos adeptos. La última versión es la 4.0 que incluye KDE 1.1, Gnome 1.0 y el Kernel 2.2 Se puede conseguir en Internet en el sitio <http://www.conectiva.com/>



Debian es una muy conocida y completa distribución muy estable y con más de 1500 paquetes en formato DEB. La última versión es la 2.1 Su web en Internet es <http://www.debian.org/>



SuSE es una distribución alemana (con una versión en inglés) que está ganando mercado en Argentina, adopta el sistema de paquetes RPM. La última versión es la 7.1, es muy completa, trae 5 CD El sitio de SuSE en Internet es http://www.suse.com/index_us.html



SLACKWARE es una de las distribuciones históricas, es muy completa y estable, junto con Red Hat, SuSE y Debian son las cuatro más populares. La última versión es la 4.0 recién liberada. En Internet a Slackware lo encontramos en <http://www.slackware.com/>



Caldera es una distribución comercial orientada a empresas y grandes usuarios por lo que tiene muchos paquetes adicionales. La última versión es la 2.2. Su sitio en Internet está en <http://www.calderasystems.com/products/openlinux/powerquest-bundle.html>



Linux Mandrake es una distribución basada en el Red Hat esta muy actualizada y adopta el sistema de paquetes RPM. La última versión es la 6.0 y viene con el KDE. En Internet podemos conseguirla en <http://www.linux-mandrake.com/>



Linux Pro Plus es una distribución muy profesional y con amplio soporte técnico, trae 6 CDs y Manuales. La última versión es la 5.4 Se puede conseguir en la red en <http://www.wgs.com/>



MKLinux es una distribución para la plataforma Apple Mac. Para más información visite <http://www.mklinux.apple.com/>



Stampede GNU/Linux es una distribución nueva dedicada a facilitar el uso y la instalación para los usuarios nuevos y optimizar la velocidad de trabajo. El URL de su sitio es <http://www.stampede.org/>



Turbo Linux de Pacific HiTech es una distribución con la base del sistema operativo, aplicaciones y documentación. Su versión más reciente es la 3.0.1 En Internet la conseguimos en <http://www.turbolinux.com/>



YGGDRASIL es la primer distribución, Linux le debe mucho, ya que fue la encargada de difundirlo por la red. La dirección en Internet es <http://www.yggdrasil.com/>



Eurelec es otra distribución del Linux en castellano, esta basada en el Red Hat por lo que adopta los paquetes RPM. La última versión es la 2.1 En Internet podemos ubicarla en eurelec.etsit.upm.es

ANEXO C



EL PROYECTO GNU

El nombre GNU se eligió siguiendo una tradición hacker, como acrónimo recursivo para «*GNU's Not Unix*». El proyecto [GNU](#) se inició desde 1984 teniendo como objetivo el desarrollo de un sistema operativo libre completo, similar a UNIX. Después de 15 años, luego de haber adoptado al kernel Linux, el proyecto GNU ha iniciado a dar sus frutos. No en vano, gracias al proyecto GNU y a todos quienes han contribuido al desarrollo de Linux, en el momento se estima que existen más de 10 millones de usuarios del sistema operativo [GNU/Linux](#), normalmente denominado en forma impropia como Linux. La primera versión de prueba del sistema GNU utilizó su kernel Hurd, lo que ocurrió en agosto de 1996. Sin embargo, debido a varios años de esfuerzo en el invertidos, el proyecto Hurd continúa, con la esperanza de que su arquitectura superior haga a los sistemas operativos más poderosos, como lo anuncia la [FSF](#) (Free Software Foundation cuyo patrocinador es la comunidad GNU).

El Sistema Linux es en verdad un sistema GNU, ya que el compilador de C, la biblioteca de C, casi todas las utilidades (cp, ls, mv, tar, gz, etc.) son GNU.

Normalmente las licencias de software son diseñadas por los abogados de las grandes compañías para limitar el uso de sus productos. La licencia GNU persigue un fin completamente distinto, está diseñada para que el software sea libre, y lo siga siendo por siempre. El término "software libre" (free software) tiene un problema de ambigüedad para las personas de habla inglesa, un significado indeseado: "software que se puede obtener por un precio cero o gratuito" y por otro lado un significado deseado: software que da al usuario ciertas libertades.

Todo software cubierto bajo la licencia GNU puede ser usado para cualquier fin, comercial u hogareño, y debe distribuirse con su código fuente. Se dice que es como un virus. Esto es porque una de sus cláusulas principales es que cualquier modificación de un programa cubierto



por la licencia, es también cubierta por ésta. Esto ayuda a proteger la "libertad" de los programas. Ya hubo muchos casos en que modificaciones hechas a programas "libres" tuvieron que ser liberadas también solamente por la existencia de esta licencia. Por ejemplo: El soporte de C++ del gcc fue hecho por una consorcio privado que hace a sus programas tan propietarios como pueda, pero tuvo que liberarlo al público general por estar el gcc cubierto bajo la licencia GNU. De no ser por esto, no tendríamos lenguaje C++ free-software.

Existen otras licencias que se consideran también free-software. Un tipo muy conocido es el conocido como estilo BSD (Berkeley Software Distributions) que en la práctica es otra variación gratuita del Unix. Estas licencias otorgan permiso para usar el software libremente, pero también permiten que individuos o compañías puedan tomar el programa, modificarlo, e incorporarlo a un producto propietario, sin tener la obligación de devolver a la comunidad. Los proponentes de este tipo de licencias sostienen que las empresas devolverán cosas a la comunidad sin necesidad de ser obligadas. Sostienen que una licencia tan "restrictiva" como la de GNU evita de plano que empresas usen free-software. Entonces queda abierto a la discusión cual de los dos estilos de licencia conviene. Algunos programas distribuidos con este tipo de licencias son el WinE, el XFree y Apache.

Existen otras licencias que son consideradas free-software. Los requisitos generalmente aceptados están en la página de Open Source, un sinónimo de free-software, que según la comunidad que promociona el proyecto GNU indica que no es lo mismo. El significado obvio para "software de código fuente abierto" es "usted puede mirar el código fuente". Este es un criterio más pobre que "software libre" que busca la libertad total del software.

A continuación indicamos unas páginas en español recomendadas para la lectura sobre estos temas:



-
- <http://www.gnu.org/gnu/thegnuproject.es.html> Manifiesto del precursor del Proyecto GNU Richard Stallman. Contiene una historia completa del proyecto, ambigüedad del término free software, La Licencia GPL, por qué GNU/Linux?, Free Software Foundation, etc.
 - <http://www.gnu.org/philosophy/free-software-for-freedom.es.html> Documento sobre por qué es mejor software libre con respecto a open source.
 - <http://www.oreilly.com/catalog/opensources/book/toc.html> Todo sobre Open Source
 - http://www.osmosislatina.com/diversos/open_source.htm Comparación entre GNU y Open Source



ANEXO D

EL LENGUAJE SQL

SQL se ha convertido en el lenguaje de query (consulta) relacional más popular. El nombre "SQL" es una abreviatura de *Structured Query Language* (Lenguaje de query estructurado). En 1974 Donald Chamberlain y otros definieron el lenguaje SEQUEL (*Structured English Query Language*) en IBM Research. Este lenguaje fue implementado inicialmente en un prototipo de IBM llamado SEQUEL-XRM en 1974-75. En 1976-77 se definió una revisión de SEQUEL llamada SEQUEL/2 y el nombre se cambió a SQL en consecuencia.

IBM desarrolló un nuevo prototipo llamado System R en 1977. System R implementó un amplio subconjunto de SEQUEL/2 (ahora SQL) y un número de cambios que se le hicieron a (ahora SQL) durante el proyecto. System R se instaló en un número de puestos de usuario, tanto internos a IBM como en algunos clientes seleccionados. Gracias al éxito y aceptación de System R en aquellos puestos de usuario, IBM inició el desarrollo de productos comerciales que implementaban el lenguaje SQL basado en la tecnología System R.

Durante los años siguientes, IBM y bastantes otros vendedores anunciaron productos SQL tales como SQL/DS (IBM), DB2 (IBM), ORACLE (Oracle Corp.), DG/SQL (Data General Corp.), y SYBASE (Sybase Inc.).

SQL es también un estándar oficial hoy. En 1982, la American National Standards Institute (ANSI) encargó a su Comité de Bases de Datos X3H2 el desarrollo de una propuesta de lenguaje relacional estándar. Esta propuesta fue ratificada en 1986 y consistía básicamente en el dialecto de IBM de SQL. En 1987, este estándar ANSI fue también aceptado por la Organización Internacional de Estandarización (ISO). Esta versión estándar original de SQL recibió informalmente el nombre de "SQL/86". En 1989, el estándar original fue extendido, y recibió el



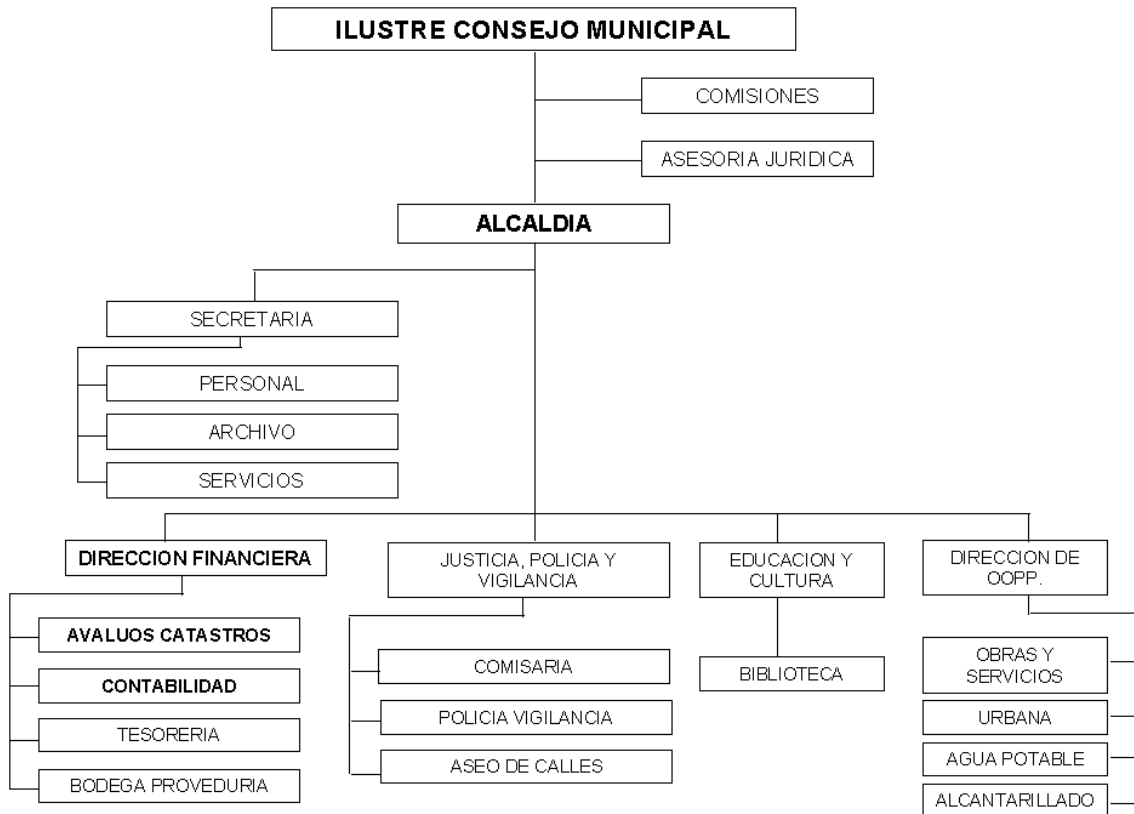
nuevo nombre, también informal, de "SQL/89". También en 1989 se desarrolló un estándar relacionado llamado *Database Language Embedded SQL* (ESQL).

Los comités ISO y ANSI han estado trabajando durante muchos años en la definición de una versión muy expandida del estándar original, llamado informalmente SQL2 o SQL/92. Esta versión se convirtió en un estándar ratificado durante 1992 - "International Standard ISO/IEC 9075:1992, Database Language SQL" -. SQL/92 es la versión a la que normalmente la gente se refiere cuando habla de " SQL estándar". Se da una descripción detallada de SQL/92 en [Date and Darwen, 1997](#). Se está desarrollando un nuevo estándar denominado informalmente como SQL3. Se plantea hacer de SQL un lenguaje de alcance completo (e Turing-complete language), es decir, todas las queries computables, (por ejemplo queries recursivas) serán posibles. Esta es una tarea muy compleja y por ello no se debe esperar la finalización del nuevo estándar antes de 1999.



ANEXO E

ORGANIGRAMA ESTRUCTURAL DEL IMH





ANEXO F

DIAGRAMAS DE FLUJO DE DATOS DEL SIB

Diagrama de Contexto de Bodegas del IMH (SIB-IMH)

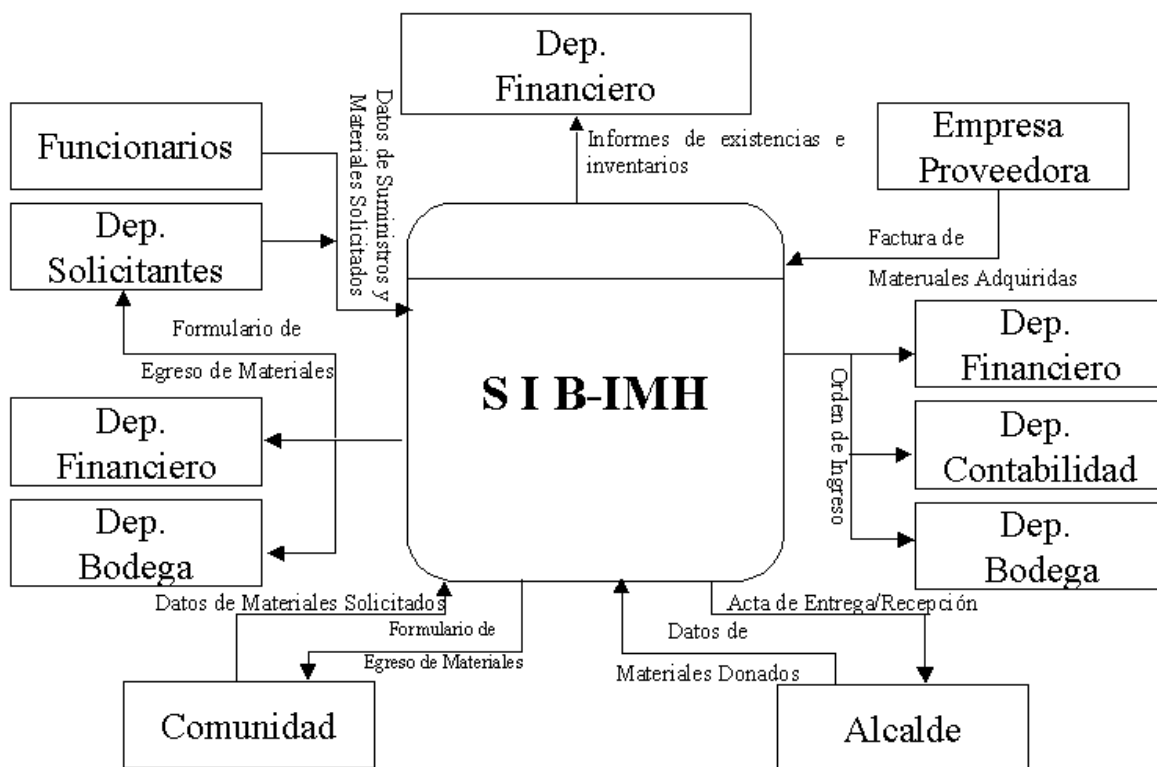




Diagrama de Flujo Nivel 0 de Bodegas del IMH (SIB-IMH)

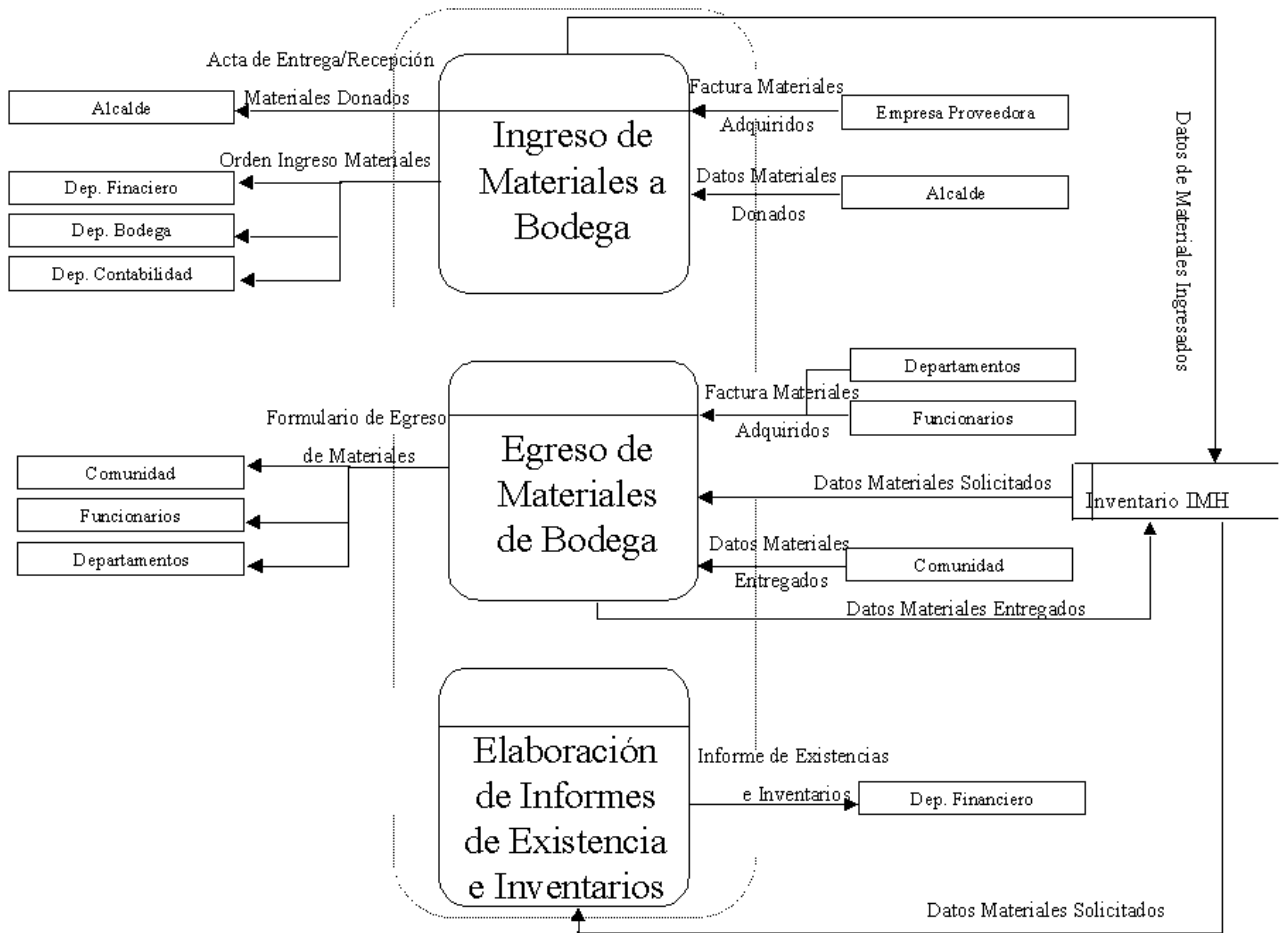




Diagrama de Flujo Nivel 1 de Bodegas del IMH (SIB-IMH)

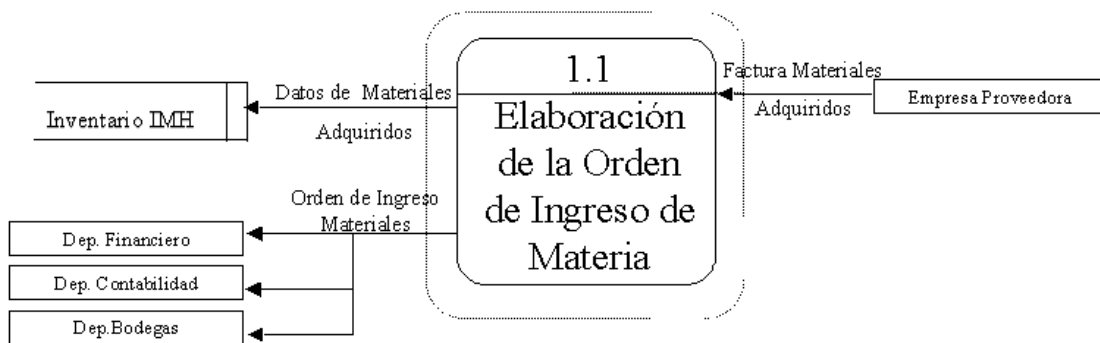




Diagrama de Flujo Nivel 2 de Bodegas del IMH (SIB-IMH)

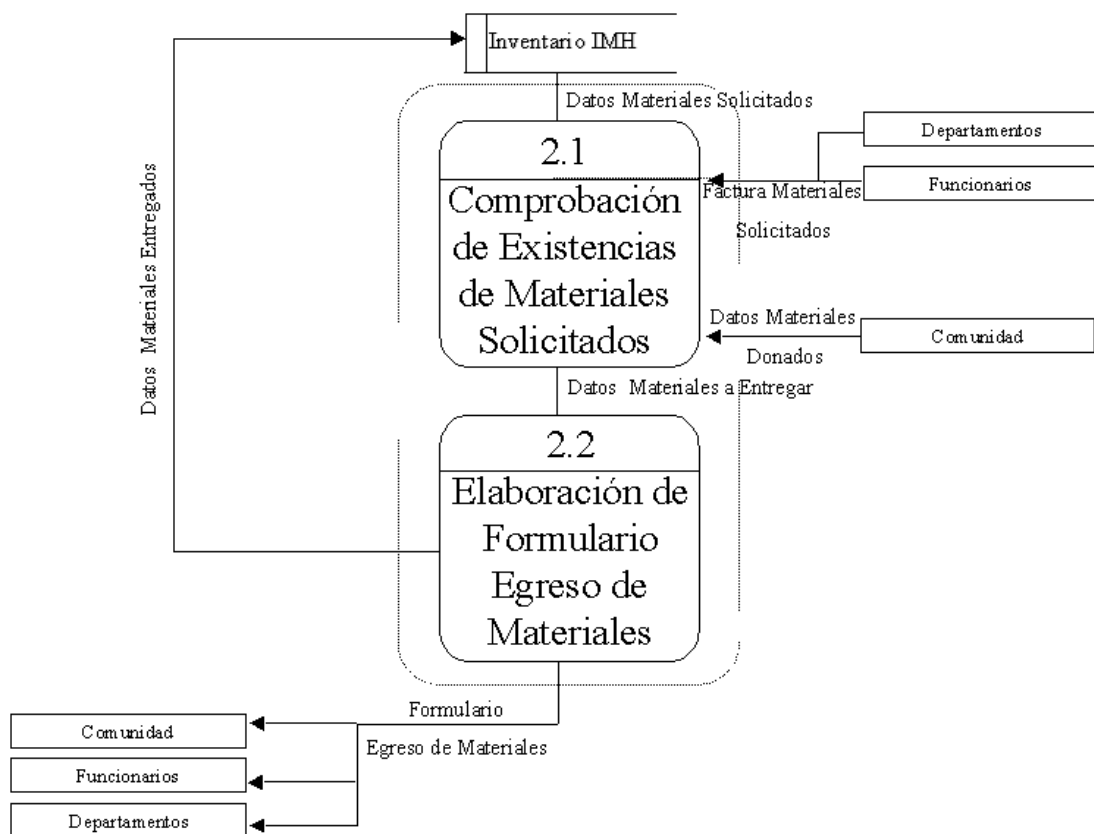




Diagrama de Flujo Nivel 2 de Bodegas del IMH (SIB-IMH)

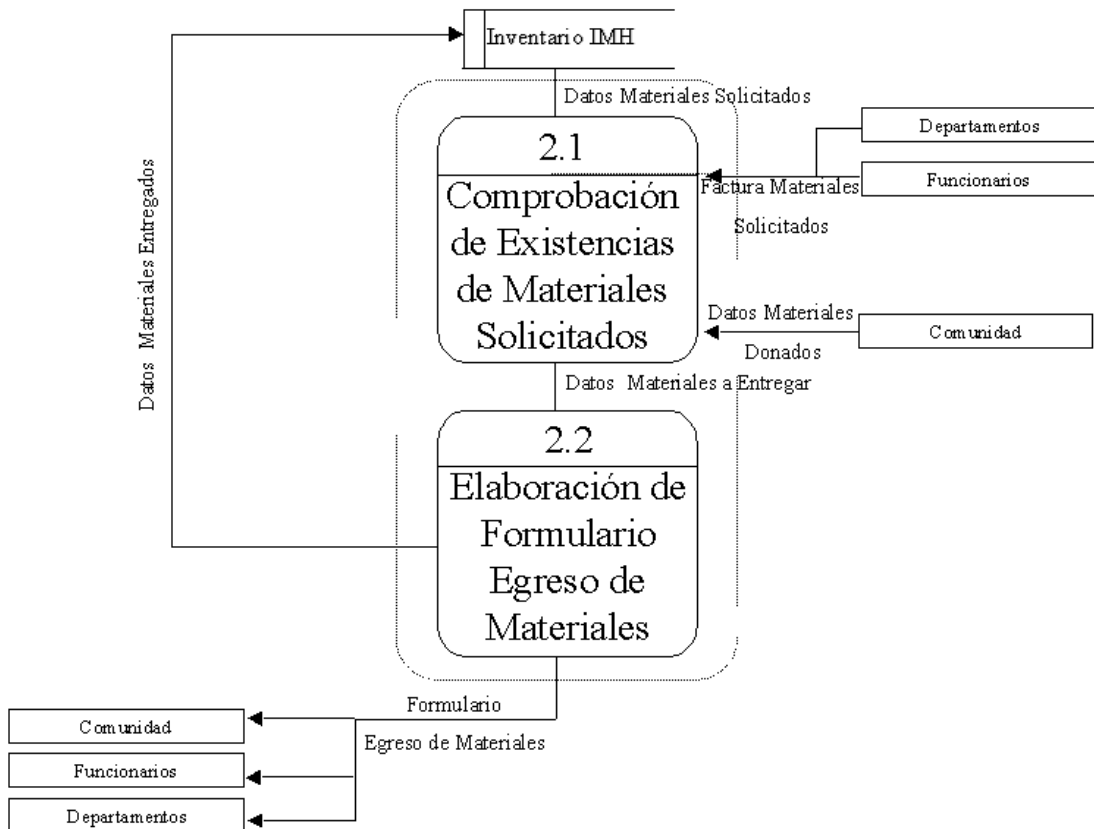
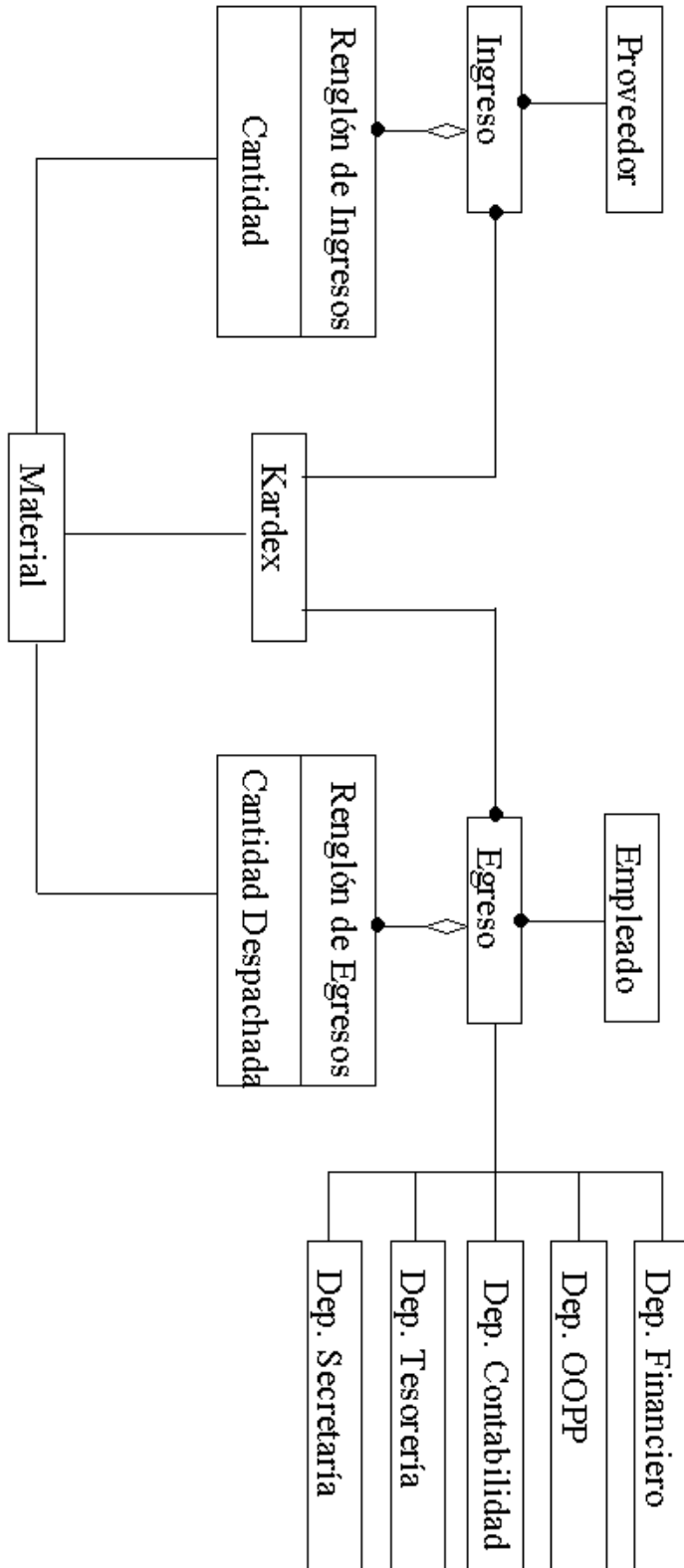




DIAGRAMA DE INTEGRACIÓN DE OBJETOS DEL SIB-IMH

SEGÚN RUMBAUGH





ANEXO G

MANUAL DEL USUARIO DEL SIB-IMH

G.1. BOTONES DE USO FRECUENTE EN LOS FORMULARIOS:

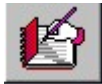
G.1.1. BOTONES DE MANIPULACIÓN DE DATOS



Crea un nuevo Registro



Elimina Registros individuales



Edita los datos de los registros para la modificación de datos



Actualiza datos por si han sido modificados por otro usuario



Imprime las órdenes de Ingreso / egreso del Material



Deshacer la operación

G.1.2. BOTONES DE NAVEGACION



Permite ir al primer registro



Ir al anterior registro



Ir al siguiente registro



Ir al último registro



Busca una orden de ingreso deseada



G.1.3. BOTONES DE MANIPULACIÓN DE DATOS DE MATERIALES

Estos botones irán apareciendo en la parte intermedia de la sección conforme se vayan requiriendo.



Permite el ingreso de un nuevo material en la Orden



Graba el material ingresado



Elimina un material



Permite al usuario realizar los procesos de actualización de stock de los materiales

G.2. Inicio de Sesión

El programa de Instalación generará los archivos respectivos para la ejecución del Sistema, optando por las siguientes formas:

a. Escritorio

Doble Click en el Icono del Sistema localizado en el escritorio de Windows con nombre “SIB-IMH”

b. Menú Inicio

1. Click en el Botón Inicio de Windows
2. Luego, hacer Click en Programas
3. Finalmente, click en el Icono del Sistema con nombre “SIB-IMH”
4. Se inicia luego la ventana de Control de Sesión en donde se ingresará el nombre del usuario y password asignado a él con un máximo de catorce caracteres.



G.3. Pantalla de Presentación

Si ha ingresado correctamente los datos requeridos anteriormente, el sistema inmediatamente visualizará la pantalla de presentación, que indica los datos generales del Sistema como: Institución, Autores y Copyright. Deberá hacer un click para continuar o digitar cualquier tecla.



G.4. Pantalla de Principal

Esta es la pantalla con la cual el usuario podrá navegar a través del sistema, el mismo que contiene diferentes secciones para el correcto control de los materiales que llegan y salen de bodega, de y hacia las personas que los manejan y control sobre las existencias de dichos materiales.



Consta de las siguientes secciones:

- Información General
- Ingresos de Materiales
- Egresos de Materiales
- Inventario
- Kardex
- Reportes

También consta de las siguientes partes: La barra de menú principal y la barra de herramientas

a. La barra de menú principal:

La misma que consta de Archivo y su submenú como muestra la figura:



La **información general** nos permite introducir datos que están asociados a cada material como Categoría, Rubro, Material, Proveedor, Empleado.

Ingreso de Materiales permite hacer el ingreso de todas las compras que se hace

Egreso de Materiales permite hacer todos los egresos de los materiales que se solicite.

Inventario permite observar los materiales existentes en bodega

Kardex permite visualizar en pantalla la tarjeta de control de inventario, sabiendo así cuantos ingreso y egresos han existido de cada material.

b. Administración.- y su submenú



Cambio de Contraseña permite renovar el actual password asignado al usuario

Nuevo Usuario Permite Cambiar el usuario anterior por un nuevo.

c. Ayuda.- y con el submenú



Contenido ejecuta el sistema de ayuda para el usuario

Acerca de.. permite visualizar los créditos de los autores así como el Copyright



d. La barra de herramientas:



En el orden que esta la barra de herramientas consta de cambio de Contraseña, Configuración de Impresora y Salir del Programa.

G.4.1 SECCIONES

G.4.1.1 Sección Información General

Esta sección permite al usuario realizar el ingreso de datos necesarios para la utilización del sistema, se encuentra dividido en cinco hojas.

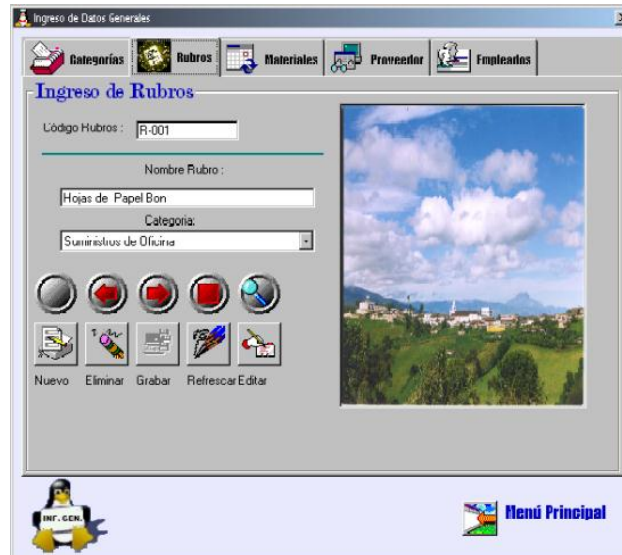
La primera permite el ingreso de Categorías a la cual va ha pertenecer cada Rubro.





Esta sección está dividida en dos partes: la parte superior que contiene el código y nombre de la Categoría la parte inferior formada de los diversos controles de navegación, manipulación de datos.

La segunda permite el ingreso de Rubros.



Esta sección está dividida en dos partes: la parte superior que contiene el código y nombre del Rubro y podrá seleccionar ya a que categoría pertenece el rubro ingresado. La parte inferior formada de los diversos controles de navegación, manipulación de datos.

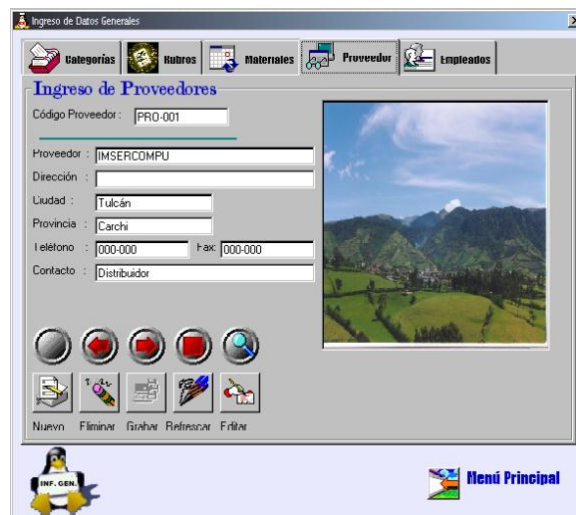
La tercera permite el ingreso de materiales.





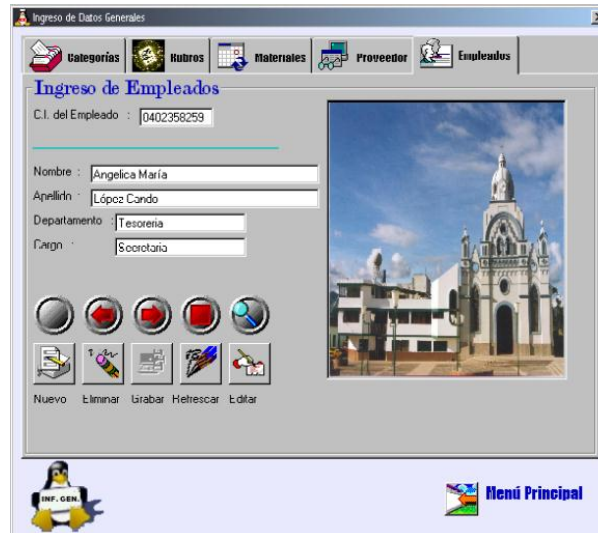
Esta sección está dividida en dos partes: la parte superior deberá ingresar el código del Material la Categoría a la cual pertenece e inmediatamente solo aparecerán los rubros pertenecientes a la Categoría señalada de los cuales el usuario debe señalar el rubro que le pertenezca al Material que desea ingresar, *En Detalle* coloque alguna observación que desee, *No Serie* si el material posee caso contrario nada, *Unidad de Medida* como el material o producto ingresado es medido para el ingreso, *Cantidad mínima* indicará cuanto podrá existir en forma mínima en Almacén, *Cantidad Máxima* indicará cuanto podrá ingresar a almacén y que no haga exceso. La parte inferior formada de los diversos controles de navegación, manipulación de datos.

La cuarta permite ingresar datos de cada proveedor que posee el Municipio.



Esta sección está dividida en dos partes: la parte superior se podrá ingresar todos los datos necesarios, como *código*, *nombre del proveedor*, *Dirección*, *Ciudad*, *Provincia*, *Teléfono*, *Fax*, nombre de la persona con quién se realiza el *Contacto*. La parte inferior formada de los diversos controles de navegación, manipulación de datos.

La última permite el ingreso de datos de cada uno de los empleados existentes en el Municipio.



Esta sección está dividida en dos partes: la parte superior se podrá ingresar todos los datos necesarios del empleado, *Número de Cédula*, *nombre del Empleado*, *Apellido*, *Departamento* al cual pertenece y por último el *cargo* que desempeña en dicho departamento. La parte inferior formada de los diversos controles de navegación, manipulación de datos.

G.4.1.2 Sección Ingreso de Materiales



Esta sección permite al usuario realizar el ingreso de materiales que han sido adquiridos por el municipio, organizada de forma de Orden de Ingreso con una lista de materiales para cada Orden.



Esta sección está dividida en tres partes: la parte superior que contiene los datos de la Orden de Ingresos, la parte intermedia que provee la lista de materiales anexas a la Orden y la parte inferior formada de los diversos controles de navegación, manipulación de datos.

G.4.1.3 Sección Egresos de Materiales

Esta sección permite al usuario realizar el egreso de materiales que han sido solicitados por el empleado del municipio, de forma que se encuentran organizados por Orden de Egresos con una lista de materiales para cada Orden.

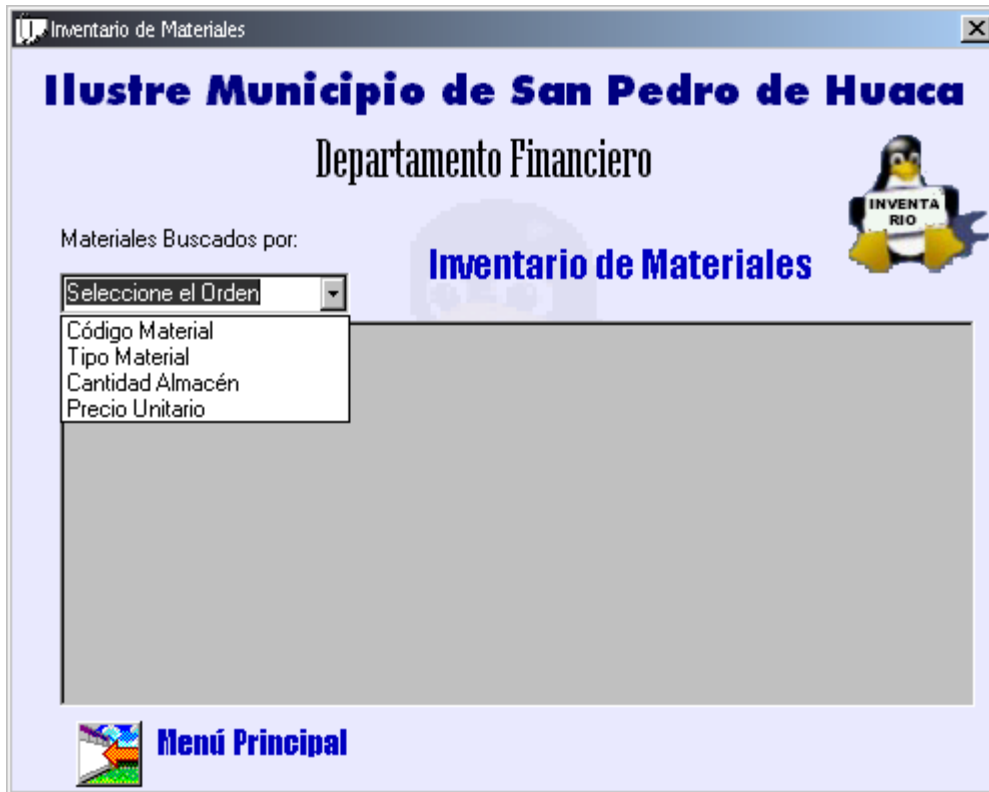


Esta sección está dividida en tres partes: la parte superior que contiene los datos de la Orden de Egresos, la parte intermedia que provee la lista de materiales anexas a la Orden y la parte inferior formada de los diversos controles de navegación, manipulación de datos.



G.4.1.4 Sección Inventario

Esta sección permite al usuario realizar una breve consulta, conocer gran parte de la información existente que tiene bodega almacenada en el sistema, posee cuatro alternativas de Búsqueda:

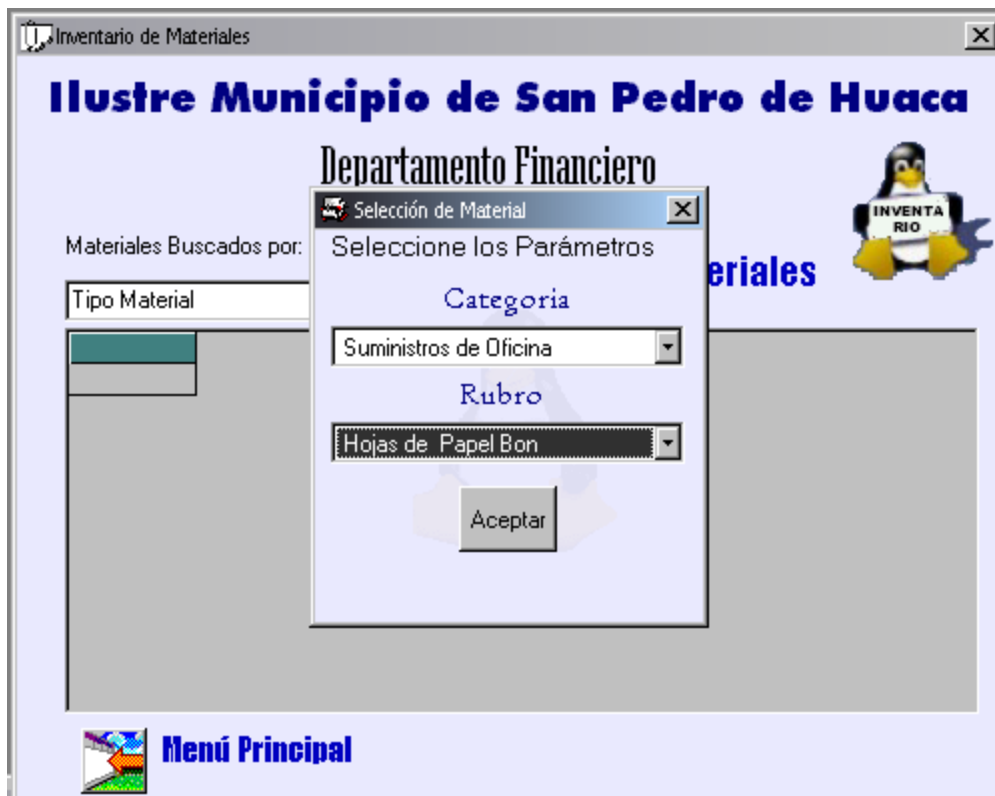


Posesiónese con el mouse en Materiales Buscados por:

Código Material: Al seleccionar esta opción se presenta un cuadro de mensaje, que pedirá al usuario que introduzca el código del material para la búsqueda, luego observará una presentación así:



Tipo Material: Esta opción permite observar un informe del material por tipo. Inmediatamente luego aparecerá una serie de selecciones tanto para categoría como de rubro del material que desea observar.



aceptamos e inmediatamente obtenemos lo solicitado al sistema con una imagen así:



Cantidad Almacén: está opción presenta otra forma de selección:



- . Mayor o igual que
- . Menor o igual que
- . Igual a

la cantidad que yo deseo observar obteniendo como resultado:





Precio Unitario: Esta opción permite hacer una revisión de los materiales que se encuentra en bodega con un precio:

- . Mayor o igual que
- . Menor o igual que
- . Igual a

lo indicado por el Usuario como se muestra en la figura:





e inmediatamente en pantalla podrá observar los resultados que se ha solicitado.



G.4.1.5 Sección Kardex

Esta sección permite observar la tarjeta de control de existencias de los materiales mostrando un recorrido por fechas de las transacciones de Ingresos y Egresos que se hayan efectuado durante el período establecido como muestra la figura.





En la parte inferior formada de los diversos controles de navegación, manipulación de datos, los mismos que le permite observar material por material, hacer una búsqueda, como también realizar la impresión.

G.4.1.6 Sección Reportes

Esta opción presenta los diferentes tipos de reportes que tiene el sistema y consta de lo siguientes opciones:



Inventarios

Presione Click y podrá elegir una ordenación por Código o por Nombre y esta se ordenará de acuerdo a la selección del usuario como se indica:

Stock <= Al Mínimo,

Stock > 0,

o todos.

aceptamos e inmediatamente un reporte en pantalla que muestra lo siguiente



Código del Material	Nombre del Material	Detalle	Número de Serie	Unidad de Medida	Precio Actual	Cantidad Mínima	Cantidad en Almacén
MAT-001	Hojas de 75 gr.	kjghdloj	Sin Serie	Unidades	12.30	0	55
MAT-002	Compag 5100	buesa	Sin Serie	Unidades	15.60	0	15
MAT-003	Pintura Blanca	CONDOR	Sin Serie	Unidades	25.60	0	1
MAT-004	Escobas de plástico	sdtd	Sin Serie	Unidades	17.80	0	14

Catálogo de Categorías

Presione Click y podrá elegir una ordenación por Código o por Nombre

la categoría.

aceptamos y mostrara en pantalla lo siguiente.



Nombre de Categoría	Código de Categoría
Materiales de Aseo	CAT-003
Materiales de Construcción	CAT-002
Suministros de Oficina	CAT-001
Total Items	3,00

Catálogo de Rubros

Presione Click y podrá elegir una ordenación por Código o por Nombre el Rubro del Material que necesita.



aceptamos y mostrara en pantalla lo siguiente.



Código del Rubro	Nombre del Rubro
R-001	Hojas de Papel Bon
R-002	Computadoras
R-003	Pinturas
R-004	Escobas
Total Items	4,00

Catálogo de Materiales Presione Click y podrá elegir una ordenación

por Código

por Nombre

Por Categorías

Por Rubros; como muestra la figura:



Según la selección que elija el Usuario presione el botón aceptar y podrá observar en pantalla el resultado como muestra la figura.



Código del Material	Nombre del Material	Detalle	Número de Serie	Unidad de Medida	Precio Actual	Cantidad en Almacén
CAT-001 Suministros de Oficina						
MAT-001	Hojas de 75 gr.	kjghdldfj	Sin Serie	Unidades	12,30	55
MAT-002	Compaq 5100	buena	Sin Serie	Unidades	15,60	15
CAT-002 Materiales de Construcción						
MAT-003	Pintura Blanca	CONDOR	Sin Serie	Unidades	25,60	1
CAT-003 Materiales de Aseo						
MAT-004	Escobas de plástico	sdésd	Sin Serie	Unidades	17,80	14

Reportes Ingresos

- Por Fechas
- Por Proveedor

Reportes Por Fechas permite realizar una consulta de todos los materiales ingresados en las fechas introducidas por el usuario similar a la figura siguiente. Así mismo **Reportes Por Proveedor** permite realizar una consulta de todos los materiales ingresados por proveedor.





Dependiendo de la opción obtendrá en pantalla el reporte solicitado como muestra la figura:

Fecha de Ingreso	Nombre del Material	Detalle	Numero de Factura	Cantidad	Precio Unitario	Ingreso No
IMSERCOMPU						
17/08/2001	Hojas de 75 gr.	kjghñbñf	123	100	12,30	1
17/08/2001	Compaq 5100	buena	123	10	15,60	1
17/08/2001	Pintura Blanca	CONDOR	123	10	25,60	1
17/08/2001	Escobas de plástico	edñs-d	123	20	17,80	1
PC Mark						
17/08/2001	Hojas de 75 gr.	kjghñbñf	5645	10	25,60	2
17/08/2001	Compaq 5100	buena	5645	10	23,20	2

Reportes Egresos

Por Fechas

Por Empleado

Reportes de Egresos Por Fechas permite realizar una consulta de todos

los materiales egresados en las fechas que se ha hecho el pedido como muestra la figura.

Reportes Varios

Ilustre Municipio de San Pedro de Huaca
Departamento Financiero

Reportes

Reporte de Egresos por Fechas

Por Favor Establezca el Rango

Formato de fecha: día/mes/año

Desde : 30/07/2001

Hasta: 01/09/2001

Aceptar Cancelar Definir Impresora

Menú Principal



introduzca la fecha desde y hasta la fecha que desee que el sistema presente todos los egresos producidos en ese rango establecido en un formato de (día, mes, año), y podrá visualizar la consulta que se está haciendo al sistema, como se indica en la figura.

Nombre del Material	Detalle	Número de Serie	Cantidad Despachada	Empleado	Egreso No
Materiales ingresados al: 18 de Agosto de 2001					
Escobas de plástico	sdfsd	Sin Serie	1	López Cando Angelica Mar	1
Pintura Blanca	CONDOR	Sin Serie	2	López Cando Angelica Mar	1
Hojas de 75 gr.	kjghvklfj	Sin Serie	10	López Cando Angelica Mar	1
Hojas de 75 gr.	kjghvklfj	Sin Serie	20	freire ampary	2
Total Parcial			33,00		
Materiales ingresados al: 20 de Agosto de 2001					
Hojas de 75 gr.	kjghvklfj	Sin Serie	15	checa marco	3
Total Parcial			15,00		
Total Materiales Entregados			48,00		

Reportes de Egresos Por Empleado permite realizar una consulta de todos los materiales que han sido pedidos por cada uno de los empleados.



introduzca la fecha inicio en un formato de (día, mes, año) y de final utilizando el mismo formato anterior, y podrá visualizar la consulta de Egresos por empleado que se está haciendo al sistema, y aceptamos.

ILUSTRE MUNICIPIO DEL CANTON SAN PEDRO DE HUACA
Departamento Financiero

Departamento: Tesorería **Nombre:** Angelica Maria Lopez Cando **Cargo:** Secretaria

Fecha:	Nombre del Material	Detalle:	Cantidad Despachada	Egreso No:
18/08/2001	Escobas de plástico	sdfsdf	1	1
18/08/2001	Pintura Blanca	CONDOR	2	1
18/08/2001	Hojas de 75 gr.	kjghdskf	10	1
Total Parcial			13,00	

Departamento: financiero **Nombre:** Mercedes Flores **Cargo:** jefe

Fecha:	Nombre del Material	Detalle:	Cantidad Despachada	Egreso No:
20/08/2001	Hojas de 75 gr.	kjghdskf	15	3
Total Parcial			15,00	

Departamento: tesoreria **Nombre:** Monica Narvaez **Cargo:** secretaria

Fecha:	Nombre del Material	Detalle:	Cantidad Despachada	Egreso No:
18/08/2001	Hojas de 75 gr.	kjghdskf	20	2
Total Parcial			20,00	

Total Materiales Despachada: 48,00

1 de 1 Salir Cerrar 5 de 5 Total:5 100%

visualizando en pantalla lo solicitado al sistema.

Cambio de Password

Permite cambiar una clave anterior por una nueva, para esto deberá introducir la clave anterior, por la nueva clave con la que quiere reemplazar la anterior, par luego confirmar la nueva clave; esta última por seguridad:



y aceptamos.

Cambio de Usuario

Permite cambiar de un usuario a un nuevo usuario



También visualizamos el usuario que queremos cambiar en Usuario Actual, e inmediatamente lo cambiamos con el nombre deseado en Nuevo Usuario y aceptamos la orden.



ANEXO H

ENLACES MySQL

Presentamos a continuación una serie de páginas WEB que pueden utilizarse para la investigación respetiva de este poderoso motor de Bases de Datos:

H.1 API's

MySQL PHP API

<http://www.php.net/>

MySQL C++ APIs

<http://www.mysql.com/Contrib/>

MySQL Python APIs

http://www.mysql.com/Contrib

MySQL TCL APIs

<http://www.binevolve.com/~tdarugar/tcl-sql/> o también http://www.mysql.com/Contrib.

H.2 Clientes

- [mysqladm.tar.gz](http://www.mysql.com/Contrib/mysqladm.tar.gz) (<http://www.mysql.com/Contrib/mysqladm.tar.gz>) Administrador de BDD Web MySQL en Perl. por Tim Sailer.
- Página principal de [mysqlgui](http://www.mysql.com/download_clients.html) (http://www.mysql.com/download_clients.html) por TCX.
- [kmysqladmin-0.4.1.tar.gz](http://www.mysql.com/Contrib/kmysqladmin-0.4.1.tar.gz) (<http://www.mysql.com/Contrib/kmysqladmin-0.4.1.tar.gz>)
- [kmysqladmin-0.4.1-1.src.rpm](http://www.mysql.com/Contrib/kmysqladmin-0.4.1-1.src.rpm) (<http://www.mysql.com/Contrib/kmysqladmin-0.4.1-1.src.rpm>)
- [kmysqladmin-0.4.1-1.i386.rpm](http://www.mysql.com/Contrib/kmysqladmin-0.4.1-i386.rpm) (<http://www.mysql.com/Contrib/kmysqladmin-0.4.1-i386.rpm>)
- [mysqlwinadmn.zip](http://www.mysql.com/Contrib/mysqlwinadmn.zip) (<http://www.mysql.com/Contrib/mysqlwinadmn.zip>) Win32 GUI administrador de BDD, por David B. Mansel, david@zhadum.org .



- netadmin.zip (<http://www.mysql.com/Contrib/netadmin.zip>) Una herramienta administrador para MySQL sobre Windows 95/98 y Windows NT 4.0.
- xmysqladmin-1.0.tar.gz (<http://www.mysql.com/Contrib/xmysqladmin-1.0.tar.gz>) Un front-end para el motor MySQL
- xmysql-1.9.tar.gz (<http://www.mysql.com/Contrib/xmysql-1.9.tar.gz>)
- Página principal de xmysql (<http://web.wt.net/~dblhack>) Un front-end para el motor MySQL. Requiere xforms 0.88 (<http://bragg.phys.uwm.edu/xforms>).
- GtkSQL (<http://www.multimania.com/bbrox/GtkSQL>) Una herramienta de consulta para MySQL and PostgreSQL.
- Cliente para Win 32 de Atronic para MySQL <http://www.mysql.com/Downloads/Win32/ArtronicWINAdmin.exe> .

H.3. Herramientas RPMs (Para RedHat 6.1)

- perl-Data-ShowTable-3.3-2.i386.rpm (<http://www.mysql.com/Contrib/perl-Data-ShowTable-3.3-2.i386.rpm>)
- perl-Msql-Mysql-modules-1.2210-2.i386.rpm (<http://www.mysql.com/Contrib/perl-Msql-Mysql-modules-1.2210-2.i386.rpm>)
- php-pg-3.0.13-1.i386.rpm (<http://www.mysql.com/Contrib/php-pg-3.0.13-1.i386.rpm>)
- php-pg-manual-3.0.13-1.i386.rpm (<http://www.mysql.com/Contrib/php-pg-manual-3.0.13-1.i386.rpm>)
- php-pg-mysql-3.0.13-1.i386.rpm (<http://www.mysql.com/Contrib/php-pg-mysql-3.0.13-1.i386.rpm>)

H.4 MyODBC

MyODBC es un controlador ODBC de 32-bits (ver. 2.50) nivel 0 para la conexión de aplicación con MySQL. Trabaja sobre Windows95, Windows98, NT y plataformas Unix.

MyODBC es de dominio público: http://www.mysql.com/download_myodbc.html .



BIBLIOGRAFIA

KENDALL & KENDALL, **“Análisis y Diseño de Sistemas”**, Practice hall, Segunda Edición, Quito-Ecuador, 1995.

KENETH L. SPENCE/KEN, **“Programación Cliente/Servidor con Microsoft Visual Basic”**, McGraw Hill, Microsoft Press.

JOHN BURCH – GAY GRUDNITSKI, **“Diseño de Sistemas de Información”**, Megabyte Editores, México, Primera Edición, 1989.

ROGER PRESSMAN, **“Ingeniería de Software”**, McGraw-Hill, Madrid-España, Cuarta Edición, 1999.

PADILLA BYRON, TERAN GUSTAVO, RUIZ GERSON, **“Evaluación de Herramientas Cliente / servidor para Sistemas Financieros Contables en Instituciones públicas”**, 1999.

KURTH JOHN, **“Fundamentos de Bases de Datos”**, McGraw-Hill, Madrid-España, Segunda Edición, 1998.

QUIN HORACIO, **“Reingeniería de Procesos con enfoque en el Benchmarking”**, España, 1997



PAGINAS WEB

MySQL

<http://www.abriasoftware.com/demo2/c1/1-7.htm> Arquitectura MySQL.

http://www.suse.de/es/produkte/susesoft/linux/Pakete_prof/mysql.html Características generales MySQL.

<http://www.analysisandsolutions.com/code/mybasic.htm> Cómo instalar MySQL en una máquina Windows.

<http://www.arrakis.es/~chessy/docs/teoria/Tutorial-MySQL.html> Manual de MySQL en español

PostgreSQL

<http://linuxfocus.nevod.ru/Castellano/May1998/article13.html> Artículo de instalación de PostgreSQL.

<http://lucas.hispalinux.es/Articulos-periodisticos/jantonio/odbc/odbc1.html> Conexión PostgreSQL con ODBC.

Mimer 8.2

<http://linxtoday.com/stories/2798.html> Criterios sobre Mimer.

<http://www.mimer.se/linux/lwc/LWCSupp.html> Conceptos de Mimer.

<http://kapitza.ras.ru/people/leva/mimer.htm> Tips sobre conexión con Mimer.

Interbase

<http://www.interbase.dthomas.co.uk/ibdownloads.htm> Herramientas de Interbase.

<http://iblinux.rios.co.jp/int/> Manuales.

Otros

<http://www.postgresql.org/comp-comparison.html> Comprobación de SGBDD comerciales y gratuitos.



<http://www.gnu.org/> Página de la Licencia GNU.

<http://www.maestrosdelweb.com> Ayuda sobre algunos aspectos de los SGBDD.

<http://www.map.es/csi/silice/Sisgesbasdat.html> Metodología de Evaluación de SGBDD.

http://www.linuxcare.com/products/prodmore.epl?PRODUCT_GROUP=Databases

Otra

evaluación de SGBDD.

<http://www.iti.upv.es/~josep/docencia/sdb/rpc/RPC.1c.html> Concepto de MultiThread.

<http://www.ucol.mx/docencia/facultades/fime/linux/sld022.htm> Principios Cliente/Servidor.

<http://www.grupocorreo.es/cibernauta/ciberpistas/freeware20.htm> Diferencias entre Freeware y Shareware.

<http://www.geocities.com/SiliconValley/8195/selec.html> Criterios de selección del Sistema Operativo correcto para una Red.

<http://www.especialistas.org.ar/plinux/distrib.html> Características de las distribuciones Linux.

http://www.linuxcare.com/products/prodreview.epl?PRODUCT_ID=366&PRODUCT_NAME=GNU

[%20SQL%20Server%200.7b](http://www.linuxcare.com/products/prodreview.epl?PRODUCT_ID=366&PRODUCT_NAME=GNU%20SQL%20Server%200.7b) Criterios de evaluación SGBDD.

<http://linuxcol.uniandes.edu.co/faq/cache/143.html> Ayudas generales para uso de Linux.

http://www.croftj.net/~barreiro/spanish/gnome-es/guia_usuario/intro.html Qué es GNOME?

<http://ttt.inf.upv.es/~jomarma1/mirada.htm> Conceptos generales de Linux.

<http://www.monografias.com> Manuales de varios temas.



GLOSARIO DE TERMINOS

ANSI (American National Standard Institute)

Instituto de Estándares Nacionales Americanos.

ANSI es la principal organización que auspicia el desarrollo de estándares tecnológicos en los Estados Unidos. ANSI trabaja con grupos de la industria y es el miembro de Estados Unidos de la Organización Internacional para la Estandarización (ISO) y la Comisión Electrotécnica Internacional (IEC). Algunos estándares computacionales que tienen tiempo en ANSI son ASCII y SCSI.

API (Application Program Interface)

Interfaz de Programas de Aplicación

Una API es un método específico prescrito por un sistema operativo de computadora o por otro programa de aplicación por medio del cual un programador que escribe un programa de aplicación puede hacer solicitudes del sistema operativo o de otra aplicación.

ARCNET (Attached Resource Computer Network)

Red de computadoras de Recursos Conectados

Arcnet es una tecnología de red de área local ampliamente instalada producida por Datapoint Corporation, el creador de la red de área local. Arcnet utiliza un esquema de token bus para la gestión de la línea, la cual es compartida entre las estaciones de trabajo y otros dispositivos conectados en la red. Arcnet puede usar cable coaxial o líneas de fibra óptica. El ancho de banda de ésta red es de 2.5 Mbps.

ASCII (American Standard Code for Information Interchange)

Código Estándar Americano para el Intercambio de Información



ASCII es el formato más común para archivos de texto en computadoras y sobre Internet. En un archivo ASCII, cada caracter alfabético, numérico o especial, es representado con un número binario de 7 bits. 128 posibles caracteres son definidos. Sistemas operativos basados en DOS y UNIX, exceptuando a Windows NT, usan ASCII para sus archivos de texto. Windows NT usa un nuevo código llamado Unicode. ASCII fue desarrollado por ANSI.

ATM. (*Asynchronous Transfer Mode*). Modo de Transferencia Asíncrono.

BENCHMARKS. Pruebas de rendimiento. Conjunto de pruebas destinadas a evaluar el *rendimiento de un sistema informático*.

BIOS. (*Basic Input/Output System*). Sistema Básico de Entrada / salida.

BLOB. *Binary Large Object*. Objeto binario grande. Entre los tipos de datos que contienen los campos BLOB están: binarios, memo, memo con formato, de imagen, de sonido y OLE.

BLOQUEO. Cuando una transacción necesita asegurarse de que el contenido de un recurso de la BD (un fichero, un registro u otro) no cambiará hasta que la transacción finalice, se bloquea. El bloqueo impide que otras transacciones lo modifiquen. Existen dos tipos principales de bloqueos: bloqueos exclusivos y bloqueos compartidos. Si una transacción realiza un bloqueo exclusivo sobre un recurso, ninguna otra podrá ejecutar ningún tipo de bloqueo contra el recurso. Si una transacción realiza un bloqueo compartido, otras transacciones podrán realizar bloqueos compartidos (pero no exclusivos) sobre el mismo recurso. Esta última técnica se utiliza cuando la transacción no va a actualizar los datos pero desea evitar que otras transacciones puedan modificarlos.

CLI. (*Command Line Interface*). Interfaz de Línea de Comandos.

CLIENTE / SERVIDOR. Arquitectura de sistemas de información en la que los procesos de una aplicación se dividen en componentes que se pueden ejecutar en máquinas diferentes. Modo de funcionamiento de una aplicación en la que se diferencian dos tipos de procesos y su soporte se asigna a plataformas diferentes.



COMMIT. Los SGBDs ofrecen sentencias especializadas para la gestión de transacciones. La nomenclatura habitual en bases de datos relacionales es: **COMMIT WORK** para finalizar una transacción y **ROLLBACK** para deshacerla.

CORBA. *Common Object Request Broker Architecture.* Fue desarrollada por OMG (*Object Management Group*) y publicada conjuntamente por IMG y X/Open. Es una especificación de mensajes basada en objetos. Establece un estándar por el que pueden emitirse mensajes en forma de peticiones a objetos (y datos) en un formato predecible. CORBA facilitará la utilización de sistemas de *software* con componentes de distintos proveedores.

CSMA/CD. (*Carrier Sense Multiple Access/Collision Detect*). Acceso Múltiple con Detección de Portadora y Detección de Colisiones.

DHCP. (*Dynamic Host Configuration Protocol*). Protocolo de Configuración Dinámica de Servidor.

DLC. (*Data Link Control*). Control de Enlace de Datos.

DNS. (*Domain Name System*). Sistema de Nombres de Dominio.

DOMINIO. En administración de bases de datos, todos los valores posibles que puede contener un campo en particular para cada registro en el archivo. En comunicaciones, todos los recursos que están bajo el control de un sólo sistema de computación.

DOS. (*Disk Operating System*). Sistema Operativo de Disco.

ENCAPSULACIÓN. Permite la conexión de varias redes informáticas entre sí para formar una sola red de nivel más alto. Cuando se utiliza encapsulación, se define un nuevo nivel de protocolo; esto proporciona una semántica uniforme para servicios tales como conmutación de paquetes, correo electrónico, etc. En programación orientada a objetos, es la asociación de datos y funciones que tiene el efecto de ocultar al solicitante de una función la forma en que ésta se ha desarrollado.

ESCALABILIDAD. Característica de un equipo que determina su capacidad de crecimiento. La escalabilidad permite aumentar el número de procesadores o cambiar a procesadores de gama superior, mejorando el rendimiento del equipo y asegurando una compatibilidad absoluta para todas las aplicaciones que se desarrollen en el futuro.



FDDI. (*Fiber Distributed-Data Interface*). Interfaz de Datos por Distribución de Fibra.

FTP. (*File Transfer Protocol*). Protocolo de Transferencia de Archivos.

FRONT-END (*herramientas*). Herramientas de planificación y definición de requisitos, herramientas de análisis y diseño y herramientas de modelización y generación de prototipos.

GNU. (*GNU not Unix*) GNU no es Unix. Acrónimo recursivo y siguiendo la tradición hacker definido al proyecto que lleva su mismo nombre cuyo precursor es el Richard Stallman.

HDLC. (*High-Level Data Link Control*). Control de Enlace de Datos de Alto Nivel.

HTML. (*Hypertext Markup Language*). Lenguaje que Señala Hipertexto.

HTTP. (*Hypertext Transfer Protocol*). Protocolo de Transferencia Hipertexto.

HOST. En una red informática, es un computador central que facilita a los usuarios finales servicios tales como capacidad de proceso y acceso a bases de datos, y que permite funciones de control de red.

EEE. (*Institute of Electrical and Electronics Engineers*). Instituto de Ingenieros Eléctricos y Electrónicos.

IP. *Internet Protocol*. Protocolo internet. Protocolo sin conexión (*connectionless*) encargado de controlar la información por la red. Permite la integración de otras subredes. Véase TCP/IP.

IPX. (*Internetwork Packet Exchange*). Intercambio de Paquetes entre Redes.

ISDN. (*Integrated Services Digital Network*). Red Digital de Servicios Integrados.

LAN. (*Local Area Network*). Red de Área Local.

MAN. (*Metropolitan Area Network*). Red de Área Metropolitana.

MIDDLEWARE. Equipo lógico intermedio que permite la comunicación entre dos módulos de *software*. Para compartir datos, los dos módulos de *software* no necesitan saber como comunicarse entre ellos, sino como comunicarse con el módulo de *middleware*. El *middleware* debe ser capaz de traducir la información de una aplicación y pasársela a otra. En términos de programación orientada a objeto, el módulo *middleware* identifica diferentes objetos y conoce qué propiedades tienen asociadas, por lo que puede responder a peticiones referentes a los mismos.



MULTI-THREAD. Cada petición es servida por una tarea diferente. Cuando hay que esperar simplemente se bloquea sin indisponer al servidor.

NDIS. (*Network Driver Interface Specification*). Especificación de Interfaz para el Controlador de Red.

NDS. (*Novell Directory Services*). Servicio de Directorios de Novell.

NetBEUI. (*NetBIOS Extended User Interface*). Interfaz Extendida de Usuario NetBIOS.

NetBIOS. (*Network Basic Input/Output System*). Sistema de Red Básico de Entrada / salida.

NFS. (*Network File System*). Sistema de Archivos de Red.

NIC. (*Network Interface Card*). Tarjeta de Interfaz de Red.

NOS. (*Network Operating System*). Sistema Operativo de Red.

NTFS. (*NT File System*). Sistema de Archivos de NT.

ODBC. *Open Data Base Connectivity*. Conectividad abierta de bases de datos. Es el pilar principal del estándar WOSA (*Windows Open System Architecture*), arquitectura de sistemas abiertos Windows.

ODI. (*Open Data-Link Interface*). Interfaz de Enlace de Datos Abierta.

OVERHEAD. Sobrecarga.

PC. (*Personal Computer*). Computadora Personal.

PHP. Es un lenguaje de programación el cual se ejecuta en los servidores web y que te permite crear contenido dinámico en páginas HTML. Dispone de múltiples herramientas que permiten acceder a bases de datos de forma sencilla, por lo que es ideal para crear aplicaciones para Internet. Es multiplataforma, funciona tanto para Unix (con Apache) como para Windows (con Microsoft Internet Information Server) de forma que el código que se haya creado para una de ellas no tiene porqué modificarse al pasar a la otra. La sintaxis que utiliza, la toma de otros lenguajes muy extendidos como C y Perl.

POP3. (*Post Office Protocol 3*). Protocolo de Servicio Postal 3.

POSIX. *Portable Operating System Interface for X*. Conjunto de especificaciones para sistemas abiertos (UNIX o no UNIX). El propósito de POSIX es proporcionar un estándar para toda la



industria de las tecnologías de la información creando una especificación de interfaz única y verificable para un sistema operativo portable.

PPP. (*Point-to-Point Protocol*). Protocolo Punto a Punto.

PPTP. (*Point-to-Point Tunneling Protocol*). Protocolo Punto a Punto por Tuneleo.

PROCEDIMIENTOS ALMACENADOS. Dispositivo para almacenar código procedural asociado con Sistemas de Gestión de Bases de Datos Relacionales (SGBDR), que hace obligatorio su uso durante cualquier operación de la base de datos.

PUERTO. *Port.* Conector de la mainboard para instalar elementos externos.

SHAREWARE. Software distribuido sobre una base de ensayo a través de BBS, servicios en línea, distribuidores de pedidos por correo y grupos de usuarios. Si se utiliza en forma regular, debe registrarse y pagarse por éste. Se requieren licencias de pago para distribución comercial.

SCSI. (*Small Computer System Interface*). Interfaz de Sistema de Computadoras Pequeñas.

SGBD. *Sistema de Gestión de Bases de Datos.* Conjunto de programas que permite crear una base de datos, manipular la información que contiene y realizar todas las tareas de administración necesarias para mantenerla operativa.

SMB. (*Server Message Block*). Bloque de Mensajes del Servidor.

SMTP. (*Simple Mail Transfer Protocol*). Protocolo de Transferencia de Correo Simple.

SNA. (*Systems Network Architecture*). Arquitectura de Sistemas de Red.

SOHO. (*Small Office Home Office*). Oficina en Casa/Oficina Pequeña.

TCP/IP. *Transmission Control Protocol/Internet Protocol.* Protocolo de Control de Transmisión / protocolo Interredes. Protocolo para el control de la transmisión orientado a la conexión (*connection-oriented*) TCP, establecido sobre el protocolo internet (IP). Su amplia extensión permite reconocerla como una norma de facto aunque no es una norma internacional. Mientras que TCP es un protocolo de transporte (nivel cuatro de OSI), el IP es un protocolo de red. Son un conjunto de normas (nivel tres de OSI) para RALs definidas en Estados Unidos para los organismos de defensa para la DARPA (*Defense Advanced Research Projects Agency*), donde



está definida la forma en que deben comunicarse los computadores, las redes entre sí y el encaminamiento del tráfico de la red.

TIEMPO DE RESPUESTA. Por regla general, tiempo transcurrido entre la acción realizada por el usuario de un sistema informático y la recepción de alguna clase de respuesta o realimentación del sistema.

TRIGGER. *Desencadenante.* Suceso que no es parte del proceso pero hace que este arranque.

TUPLA. Fila de una tabla de un SGBD relacional. Representa una ocurrencia.

UNIX Internacional. Grupo liderado por AT&T y Sun Microsystems, que patrocinan el sistema operativo UNIX System V Release 4 (SVR4) como la versión estándar universal de UNIX.

UNIX. Sistema operativo multiproceso, multiprograma y multiusuario. Software diseñado por AT&T para ingeniería de telecomunicación. Ha sido el primer sistema operativo concebido con independencia de los fabricantes. Posee una gran facilidad para adaptarse a computadores con diferentes arquitecturas, siendo ampliamente autónomo respecto del *hardware*. Está escrito en lenguaje de alto nivel **C**.

URL. (*Uniform Resource Locator*). Localizador Uniforme de Recursos.

UUCP. (*UNIX-to-UNIX Copy Protocol*). Protocolo de Copia UNIX a UNIX.

WAN. (*Wide Area Network*). Red de Área Amplia.

X/OPEN. Consorcio de fabricantes que especifican una plataforma de sistema abierto basada en el sistema operativo UNIX. Se basa en normas internacionales y normas de facto.