



UNIVERSIDAD NACIONAL AUTÓNOMA DE MÉXICO

FACULTAD DE INGENIERÍA

SISTEMA DE CONSULTA DE PRODUCTOS Y SERVICIOS VÍA INTERNET PARA UNA AGENCIA DE PUBLICIDAD UTILIZANDO HERRAMIENTAS ORACLE.

T E S I S
QUE PARA OBTENER EL TITULO DE:
INGENIERO EN COMPUTACIÓN

P R E S E N T A N ²⁸⁶⁸⁸⁶

JUAN MARTÍN CARRERA RODRÍGUEZ
JAIME ANTONIO MORÁN PERALES
HÉCTOR JAVIER REYES ROSALES
CECILIA SÁNCHEZ GONZÁLEZ
NOEMÍ SEGURA LÓPEZ



ASESOR: M.I. JUAN CARLOS ROA BEIZA

CIUDAD UNIVERSITARIA

1999

286886



Universidad Nacional
Autónoma de México



UNAM – Dirección General de Bibliotecas
Tesis Digitales
Restricciones de uso

DERECHOS RESERVADOS ©
PROHIBIDA SU REPRODUCCIÓN TOTAL O PARCIAL

Todo el material contenido en esta tesis esta protegido por la Ley Federal del Derecho de Autor (LFDA) de los Estados Unidos Mexicanos (México).

El uso de imágenes, fragmentos de videos, y demás material que sea objeto de protección de los derechos de autor, será exclusivamente para fines educativos e informativos y deberá citar la fuente donde la obtuvo mencionando el autor o autores. Cualquier uso distinto como el lucro, reproducción, edición o modificación, será perseguido y sancionado por el respectivo titular de los Derechos de Autor.

AGRADECIMIENTOS

A Dios, por permitirnos cumplir con esta meta.

A la Universidad Nacional Autónoma de México, por ser nuestra Alma Mater.

Al M.I. Juan Carlos Roa Beiza, por habernos orientado en la realización de este trabajo.

A los profesores: Dra. Ana María Vázquez Vargas, Ing. Laura Sandoval Montaña, Ing. Juan Manuel Martínez Villalobos e Ing. Manuel Manríquez Miranda; gracias por sus observaciones y consejos.

DEDICATORIAS

En un momento como este cuando la meta soñada se ha cumplido, es importante detener el paso y dar gracias a aquellos que lo han hecho posible. Por tal motivo, no puedo dejar de dar gracias a Dios por hacer posible este momento.

A mis padres Rafael y María la confianza y el amor que en mi siempre han depositado. Ustedes han representado a lo largo de mi vida 2 pilares fuertes de apoyo, símbolos de responsabilidad y de trabajo arduo. Jamás podré agradecer los cuidados, las enseñanzas y lo mucho que han hecho por mí. Los amo.

A mis hermanas Maru y Paty que nunca han perdido la confianza en lo que hago y siempre me han apoyado incondicionalmente. Y muy especialmente a mi hermano Guillermo a quien Dios llamo muy temprano a su presencia y de quien no dudo hubiera sido un hombre de bien.

A ti Humberto que siempre me has apoyado, que has estado pendiente de mis triunfos y fracasos, de mis aciertos y errores. Como agradecer tus palabras de apoyo en los momentos difíciles, tu crítica constructiva en los momentos precisos y el amor que me has brindado a lo largo de estos años. No encuentro las palabras para decirte lo importante que eres en mi vida y lo mucho que agradezco a Dios por tenerte a mi lado.

A mis amigos que siempre han estado a mi lado, por los momentos felices que no olvidaré y por su sincera amistad.

Cecilia Sánchez González.

A Mis Padres Rosa y Rafael:

Por los sacrificios y esfuerzos realizados para llegar a este fin, como hija traté de dar lo mejor, hoy como profesionista será aún más, para no defraudarlos y darles satisfacciones, porque esta profesión no la hice sola la hicimos juntos, cada día y noches en vela de estudio, y principalmente por ser los mejores amigos.

A mis Hermanos Rosa María y Rafael:

Por todo el apoyo y paciencia que me dieron a lo largo de esta carrera, gracias a sus regaños y consejos, por fin lo logré, los quiero mucho y espero que se sientan orgullosos de mí como yo de ustedes.

A Dios:

Gracias Señor por todo lo que me brindaste, con esa gran fe hacia ti, he terminado, en momentos difíciles siempre estuviste a mi lado.

A mis Maestros:

Gracias a todos ellos he logrado llegar a este fin, por sus enseñanzas y consejos, llevando la frente muy alta orgullosa de haber estudiado en una de las mejores universidades, hoy como profesionista y mañana como universitaria siempre.

A mis Amigos:

Por la amistad y ayuda desinteresada que me dieron principalmente a Juan por todo el apoyo, amor y cariño que me ha brindado, y darme ánimo para seguir adelante siempre.

Noemí Segura López.

A mis padres que me enseñaron
el camino de la superación y humildad
y que ahora se ven reflejados
en la culminación de esta
etapa académica.

A la memoria de mis abuelos
por sus sabios consejos.

A mi familia y habitantes
de esa tierra generosa de
San Jerónimo Ocotitlán, Puebla.

Héctor Javier Reyes Rosales.

A Dios:

Por haberme dado las fuerzas para llegar hasta aquí y por darme la oportunidad de vivir tan intensamente.

A mis Padres:

Por todo el apoyo que me han dado en todo momento, por darme la oportunidad de recibir una educación, por haberme guiado por un camino que me ha dado grandes satisfacciones y por todo su amor.

A mi Esposa:

Ma. Gloria por todos sus consejos, apoyo y ánimo; a su gran amor y confianza que son la luz que guía mi vida.

A mis Hermanos:

Por todo el apoyo, motivación y cariño que me han dado, para terminar mi carrera; gracias y les deseo lo mejor de la vida a todos.

A mis Amigos:

Que de alguna forma me han motivado y apoyado para llegar hasta aquí; ya que algunos de ellos los recordare siempre.

Gracias a todos.

Juan Martín Carrera Rodríguez.

A mi Padre: Rogelio Morán Oaxaca (1935-1998 †)

por ser para mí, el mejor ejemplo a seguir.

A mi Madre: Marina Perales Córdova

por el apoyo que me ha brindado a lo
largo de mi vida.

A mis Hermanos: José Luis, Rosa Aurelia,
Rocío del Pilar, Rogelio Martín, Jesús Eduardo, Marina
Elmira y Adriana María, por haber compartido conmigo
y junto con mis padres los momentos más
significativos en parte importante de mi vida.

A Judith: Con todo mi amor, por ser como eres.

Jaime Antonio Morán Perales.

ÍNDICE

CAPÍTULO 1. MARCO TEÓRICO.

1.1 Conceptos Básicos de Publicidad.	1
1.1.1 Introducción a la Publicidad.	1
1.1.2 Las Agencias de Publicidad.	8
1.1.3 La Campaña Publicitaria.	13
1.1.4 Los Medios Publicitarios.	16
1.1.5 El Mensaje Publicitario.	24
1.2 Conceptos Básicos de Internet.	31
1.2.1 Cómo se Inició Internet.	31
1.2.2 Cómo Funciona Internet.	34
1.2.3 Servicio Gráfico de Internet.	49
1.2.4 Herramientas no Pertenecientes a la Red en Internet.	52
1.2.5 El Crecimiento de Internet.	55
1.2.6 Servicios de Conexión Directa.	56
1.2.7 Seguridad.	60
1.3 La Publicidad en Internet.	68
1.3.1 Captar al Público.	68
1.3.2 Marketing en la WWW.	70
1.3.3 Navegación.	73
1.3.4 Medir el Éxito.	75
1.3.5 Establecer unos Objetivos Realistas.	76
1.3.6 Venta de Productos.	79

1.4 Sistemas Operativos.	81
1.4.1 UNIX.	82
1.4.2 Windows NT.	96
1.5 Comunicaciones.	103
1.5.1 Redes de Computadoras.	103
1.5.2 Modelos de Interconexión entre Redes.	114
1.5.3 Medios de Transmisión.	122
1.5.4 Conmutación y Transmisión.	128
1.5.5 Serie X del CCITT.	133
1.5.6 Modelo OSI.	133
1.5.7 Arquitecturas.	135
1.5.8 Protocolos.	139
1.5.9 Suite de Protocolos TCP/IP.	140
1.5.10 Redes Orientadas y no Orientadas a Conexión.	157
1.6 Bases de Datos Relacionales.	162
1.6.1 Cartucho Context.	162
1.6.2 Oracle Enterprise Manager.	171
1.6.3 SQL Plus.	175
1.7 Arquitectura NCA.	183
1.7.1 Ambiente Económico.	183
1.7.2 Evolución del Concepto Network Computing.	185
1.7.3 Ventajas de NCA.	188
1.7.4 Formas de Conexión.	189
1.7.5 Estructura de NCA.	190

1.8 Web Application Server.	198
1.8.1 Arquitectura Conceptual del Web Application Server.	200
1.8.2 Arquitectura Física del Web Application Server.	201
1.8.3 Seleccionando un Web Server.	209
1.8.4. Hardware Requerido.	213
1.8.5 Comunicaciones Requeridas.	216
1.8.6 Seguridad Web.	217
1.9 Metodología de Ingeniería de Software (Ventajas y Desventajas).	218

CAPÍTULO 2. PLANTEAMIENTO DEL PROBLEMA Y PROPUESTA DE SOLUCIÓN.

2.1 Situación Actual.	250
2.2 El Usuario y sus Requerimientos.	254
2.2.1 Requerimientos Generales.	254
2.2.2 Requerimientos Particulares.	258
2.2.3 Requerimientos Adicionales.	263
2.3 Recopilación y Análisis de Información.	264
2.3.1 Formatos.	264
2.3.2 Clasificación de la Información.	270
2.3.3 Tamaños y Precios.	278
2.3.4 Estilos de Publicación.	279
2.3.5 Concentrado de Anuncios Para Internet.	281

2.4 Planteamiento del Problema.	284
2.5 Descomposición Funcional.	286
2.5.1 Módulos Correspondientes al Back End.	287
2.5.2 Módulos Correspondientes al Front End.	288
2.5.3 Arquitectura de Cómputo.	291
2.6 Opciones de Solución.	292
2.6.1 Sistemas Operativos.	292
2.6.2 Arquitecturas de Cómputo.	297
2.6.3 Lenguajes de Programación para la Interfaz del Usuario.	300
2.7 Elección de la Solución Óptima.	305

CAPÍTULO 3. DESARROLLO E IMPLEMENTACIÓN DEL PROBLEMA.

3.1 Características y Configuración de Oracle 8.	309
3.1.1 Instalación y Configuración de Oracle 8.	325
3.1.2 Configuración del Cartucho de Context.	335
3.1.3 Enterprise Manager.	335
3.2 Configuración del Web Application Server.	342
3.2.1 Requerimientos del Sistema.	342
3.2.2 Configuración Estándar del Web Application Server.	343
3.2.3 Inicializando el Web Application Server.	345
3.2.4 Configuración Para una Instalación Multinodo.	347

3.3 Diseño y Construcción del Back End.	349
3.3.1 Diagrama de Contexto.	349
3.3.2 Diagrama de Flujo de Datos.	350
3.3.3 Diagrama Entidad-Relación.	365
3.3.4 Normalización.	378
3.4 Diseño y Programación del Front End utilizando PLSQL.	388
3.5 Integración y Depuración del Sistema.	407
3.6 Pruebas de Rendimiento, Rapidez y Operatividad de la Publicación en Internet.	412
MANUAL TÉCNICO.	424
MANUAL DE USUARIO.	441
CONCLUSIONES.	450
BIBLIOGRAFÍA.	453
ANEXO 1.	458
ANEXO 2.	459
ANEXO 3.	474

CAPÍTULO 1.

MARCO TEÓRICO.

1.1 CONCEPTOS BÁSICOS DE PUBLICIDAD.

1.1.1 Introducción a la Publicidad.

Definición y Origen.

La publicidad consiste esencialmente en dar a conocer un producto o servicio, estimulando el interés en probarlo hasta conseguir su consumo o empleo. Para que esto suceda, el producto o servicio debe responder a una necesidad real.

Las raíces de la publicidad se encuentran en todas las civilizaciones que alcanzaron una economía de cambio y de comercio. Las formas de expresión usadas en el pasado difieren de las actuales, en la misma medida en que la vida de hoy difiere de la época precedente: el cine, la televisión, los periódicos no existían y la publicidad se manifestaba a través de las marcas, los rótulos y carteles, los cuales aún se utilizan como medios publicitarios.

La historia de la publicidad está ligada íntimamente con la historia de la economía, ya que su existencia está condicionada con el tipo de economía en vigor.

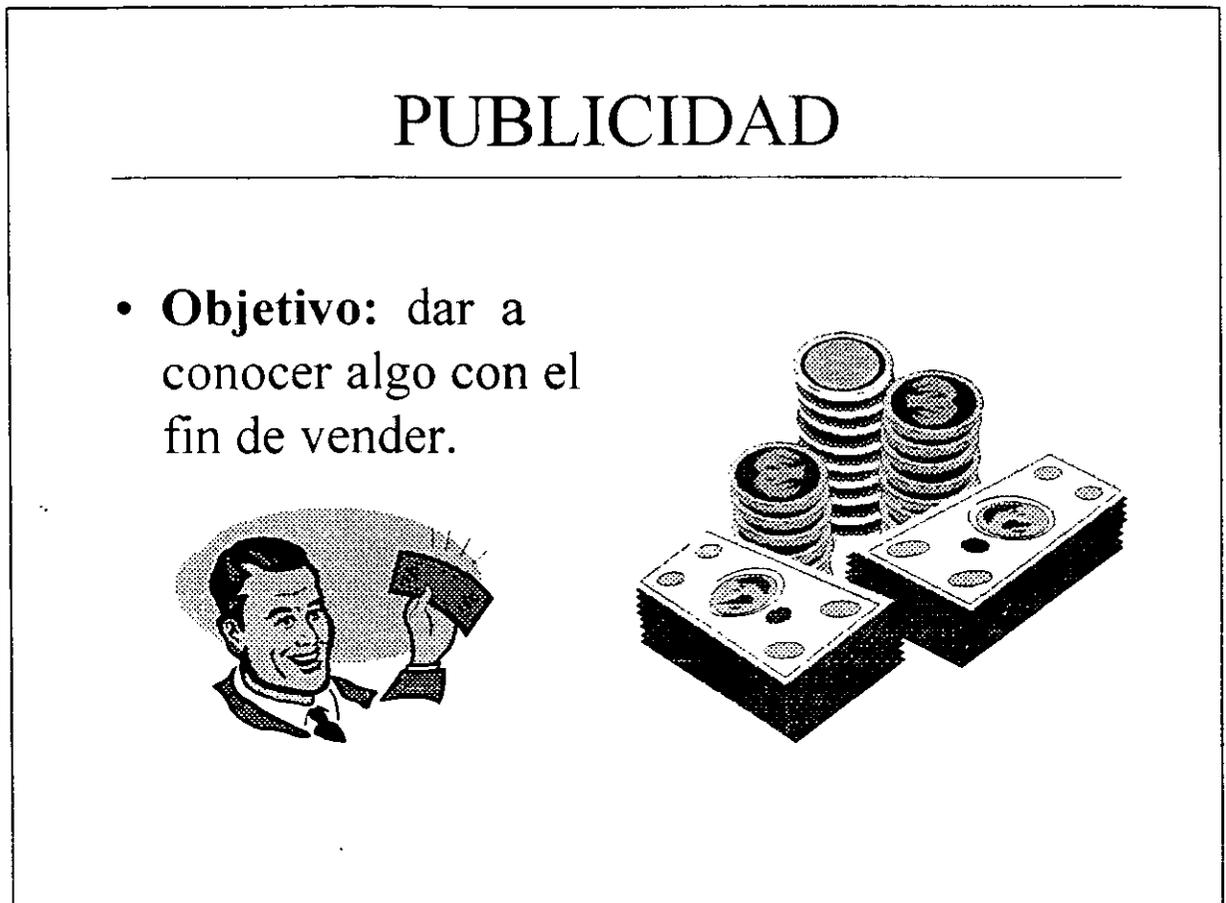
Confrontación de Definiciones: Publicidad, Propaganda, Relaciones Públicas, Relaciones Humanas.

El término publicidad difiere del término propaganda principalmente por su distinta finalidad:

Publicidad quiere decir dar a conocer algo con el fin de vender, siendo su carácter esencialmente comercial. Su finalidad principal es la de crear una circulación de dinero. En la figura 1.1.1.1 se ejemplifica este término.

Propaganda significa dar a conocer algo con un fin ideológico (social, político, religioso,

etc.). Pretende provocar un cambio en cualquier ordenamiento preexistente. En la figura 1.1.1.2 se hace referencia a este término.



- **Objetivo:** dar a conocer algo con el fin de vender.

Figura 1.1.1.1 Concepto de Publicidad.

Existen otros campos relacionados con la publicidad y la propaganda; estos son denominados relaciones públicas y relaciones humanas.

Las relaciones públicas no son una manifestación de idealismo ni de utilitarismo, simplemente son un puente entre la sociología y la economía; pero relacionándolas con la publicidad, se puede decir que sirven para introducir al consumidor, en la vida y en los problemas de la empresa productora, y en consecuencia al productor, en el problema económico y social del consumidor.

PROPAGANDA

Objetivo: dar a conocer algo con un fin ideológico:

- Social
- Político
- **Religioso**
- etc.

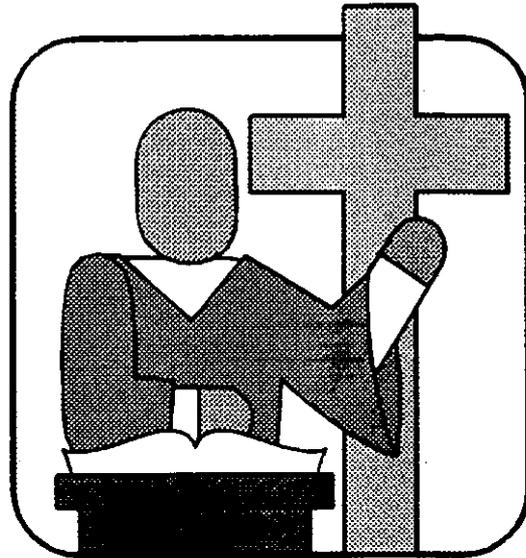


Figura 1.1.1.2 Concepto de Propaganda.

Las relaciones públicas se establecen entre un grupo y el mundo exterior. Cuando estas relaciones se establecen dentro del grupo mismo, se les denomina ahora como relaciones humanas, las cuales están encaminadas a mejorar las relaciones de convivencia entre la empresa y el trabajador. Si las condiciones de trabajo son satisfactorias para el trabajador, también su rendimiento será satisfactorio para la empresa. En la figura 1.1.1.3 se define la diferencia entre estos términos.

Funciones de la Publicidad en el Ámbito de la Economía.

La publicidad es uno de los elementos básicos para una política económica de competencia y forma parte importante para conseguir un rápido y positivo desarrollo de

las ventas de un producto; los otros aspectos fundamentales son: la calidad, el precio y la distribución.

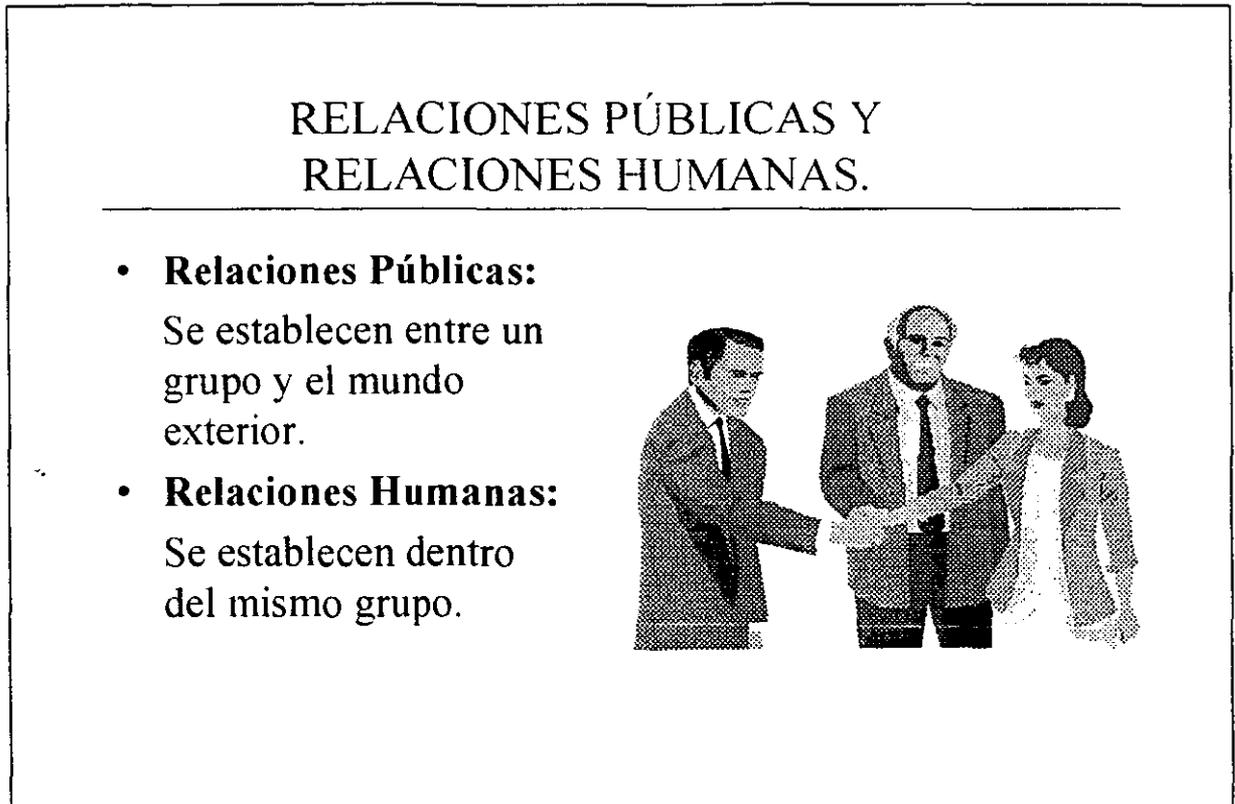


Figura 1.1.1.3 Relaciones Públicas y Relaciones Humanas.

La publicidad debe ser considerada como una partida normal del costo, como una materia prima; sin la cual las ventas del producto habrían tenido un desarrollo más lento, por lo que constituye también una inversión (Ver figura 1.1.1.4).

Tipos de Publicidad.

Como se mencionó antes, el fin de la publicidad es el de dar a conocer algo con una finalidad de venta. Pero los tipos de publicidad son diversos y pueden clasificarse a grandes rasgos como sigue:

PUBLICIDAD Y ECONOMÍA

- La **Publicidad** es un elemento básico para una política económica de competencia y se considera como una inversión.
- **Otros aspectos fundamentales:**
 - **Calidad, precio y distribución.**

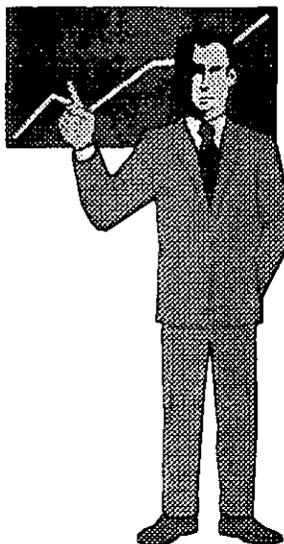


Figura 1.1.1.4 La Publicidad como una Inversión.

- a) En relación con la empresa o ente que realiza o comisiona la publicidad puede ser: Privada, colectiva, asociada y de utilidad pública.
- Privada: es la publicidad hecha por cuenta de un productor individual para sus productos, sea aquel un individuo, una empresa o un organismo estatal o paraestatal. Esta clase de publicidad es la más frecuente y normal.
 - Colectiva: cuando la campaña publicitaria es sostenida por un grupo de empresas productoras de los mismos artículos.
 - Asociada: es similar a la colectiva; la diferencia fundamental consiste en que no se hace publicidad de los mismos productos; sino de varios productos similares o distintos pero complementarios. Es decir, de cada producto hace publicidad una sola marca. En la Figura 1.1.1.5 se ejemplifica este tipo de publicidad.

- De utilidad pública o de servicio: son las campañas publicitarias encaminadas a conseguir un fin social y proviene generalmente de un ente público.

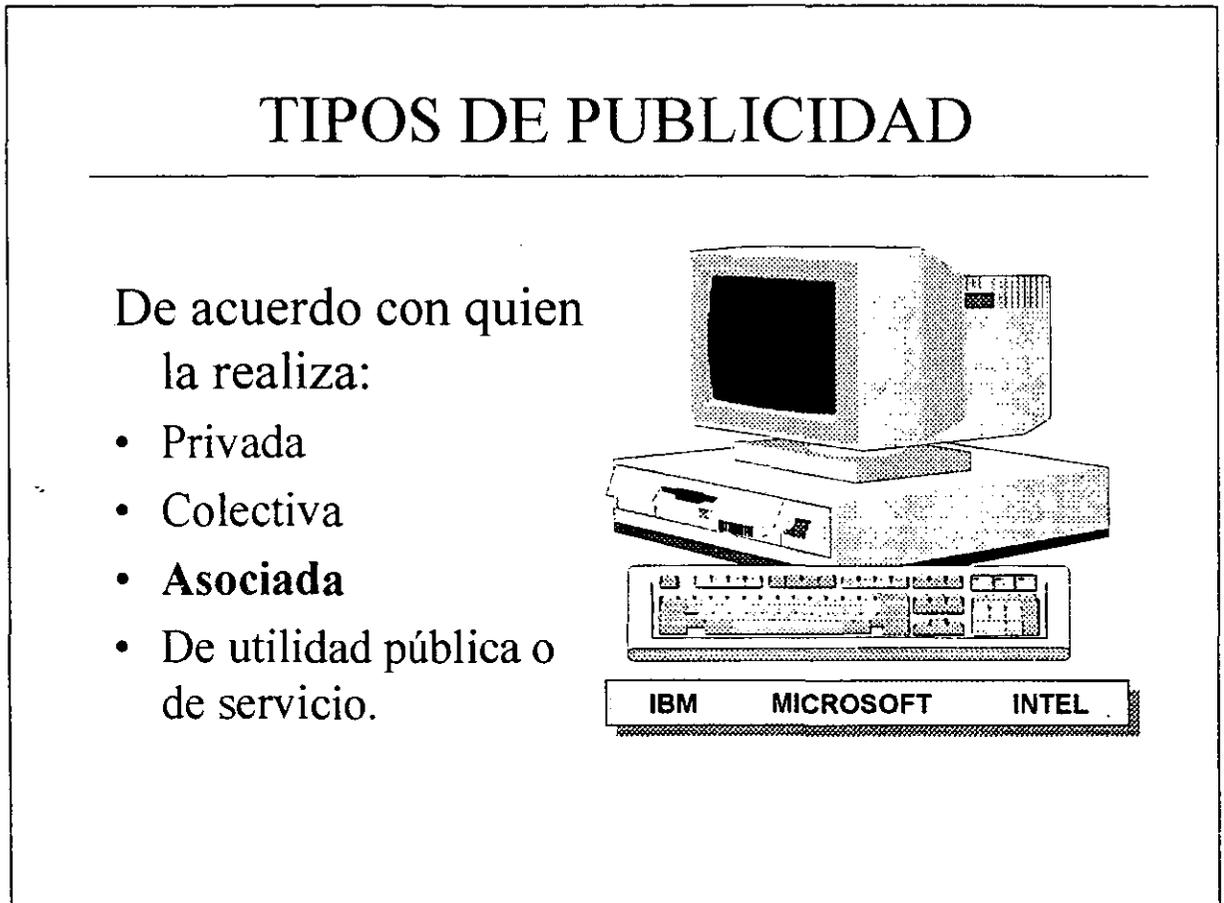


Figura 1.1.1.5 Tipos de Publicidad de Acuerdo con Quien la Realiza.

- b) De acuerdo con los objetivos inmediatos (el objetivo final es siempre la venta) puede ser: de venta, de recuerdo o de prestigio (ver figura 1.1.1.6).
- De venta. Es el tipo más común, pero también el más difícil, ya que debe unir a la comunicación de las informaciones más interesantes para el consumidor, un fuerte y válido motivo por el cual se introduzca a adquirir el producto.
 - De recuerdo. No tiene como objetivo dar a conocer condiciones preferenciales tales como ofertas, bonos de descuento; ni refiere hechos nuevos, sino que se limita a repetir el slogan, presentando la marca y las características ya notificadas al público

a través de anteriores campañas de venta. El fin es mantener las posiciones adquiridas más que intentar mejorarlas.

- De prestigio. También se le llama institucional y su finalidad inmediata no es la venta, sino la ampliación del conocimiento, por parte del público, de la empresa en su ambiente económico.

TIPOS DE PUBLICIDAD

De acuerdo con los objetivos inmediatos:

- De venta
- De recuerdo
- De prestigio



Figura 1.1.1.6 Tipos de Publicidad de Acuerdo con los Objetivos Inmediatos.

c) Atendiendo a la forma es: Directa o indirecta (Ver figura 1.1.1.7).

- La publicidad directa consiste en que el mensaje publicitario es enviado individual y directamente al consumidor potencial a través de: cartas, catálogos, folletos, muestras, etc.

- La publicidad indirecta no se dirige personal ni individualmente, por ello toda la publicidad normal de prensa, radio, televisión, cine y carteles es indirecta.

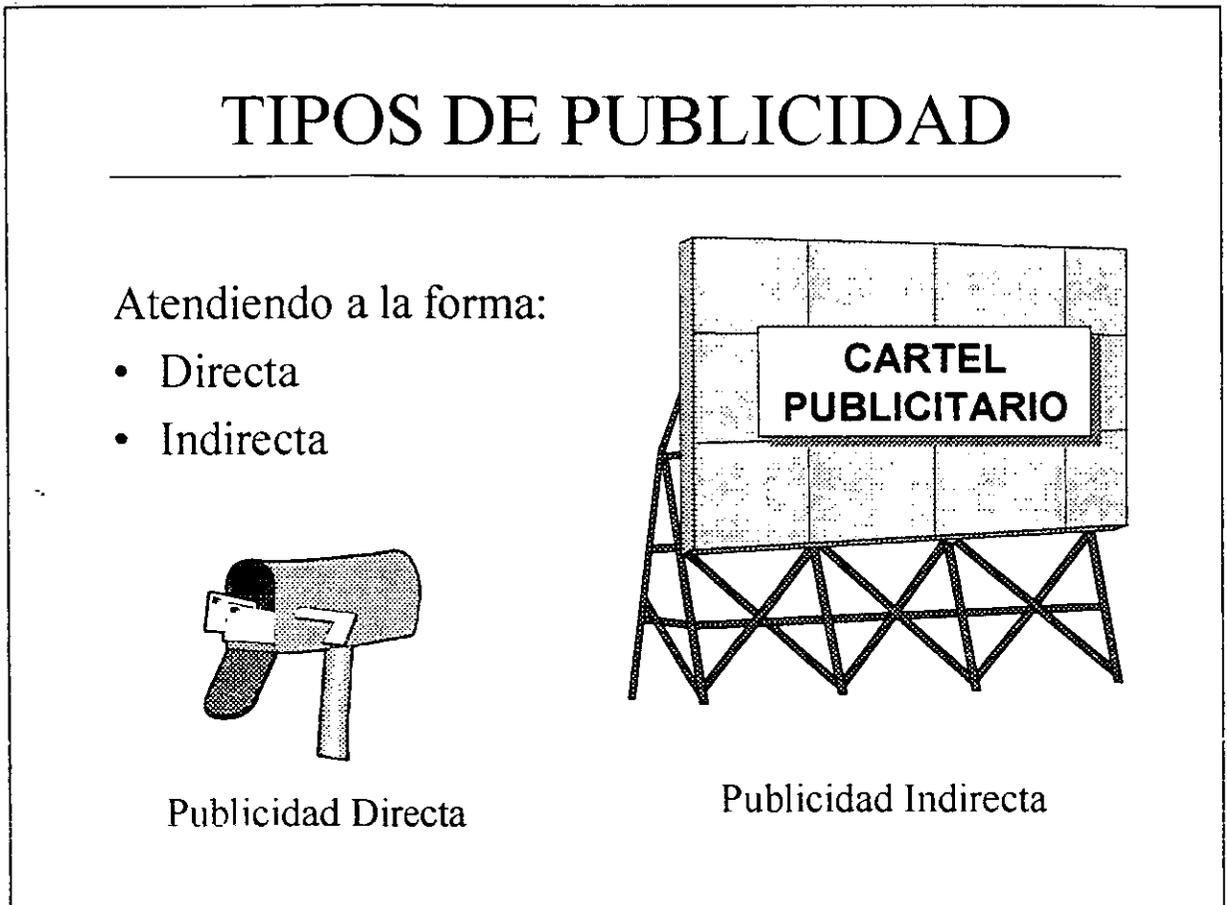


Figura 1.1.1.7 Publicidad Directa e Indirecta.

1.1.2 Las Agencias de Publicidad.

Los Interesados en la Publicidad.

Las primeras organizaciones publicitarias tienen su origen en los diarios. Debido al aumento en la complejidad de la elaboración de los mensajes publicitarios y a las exigencias de los clientes, fue necesario fijar reglas más precisas sobre los deberes y las funciones de las agencias de publicidad.

Los interesados en la publicidad pueden clasificarse como sigue:

1. Los usuarios de la publicidad o anunciantes.
 2. Los concesionarios o propietarios de los medios publicitarios.
 3. Las agencias y estudios de publicidad.
-
1. Se denomina usuario de la publicidad al anunciante, el cual sostiene los gastos de la campaña publicitaria hecha a favor de uno o más productos fabricados o distribuidos por él.
 2. Concesionario o propietario de medios publicitarios es el individuo o sociedad que tiene la exclusiva para insertar publicidad en un medio o grupo de medios. Se considera incorrecto que éstos participen en la formulación de campañas publicitarias, ya que, siendo ellos parte interesada, podrían desvirtuar la selección de los medios.
 3. Las agencias de publicidad son organizaciones comerciales autónomas, compuestas por personal creador, comercial y administrativo, especializado, que estudia, desarrolla y pone en práctica, campañas publicitarias por cuenta de terceros, sean estos fabricantes o distribuidores.

En la figura 1.1.2.1 se muestra la clasificación anterior.

Objetivos de una Agencia Publicitaria.

Una agencia de publicidad debe:

1. Estudiar el proyecto del cliente en relación con sus cualidades intrínsecas y las de los productos de la competencia que puedan existir en el mercado.
2. Analizar el mercado actual y potencial en el que debe introducirse el producto.
3. Conocer los problemas de distribución y venta del producto.
4. Conocer los medios publicitarios existentes y estar capacitado para seleccionar cuál o cuáles deben usarse para dar a conocer el producto a los mayoristas,

revendedores y consumidores potenciales.

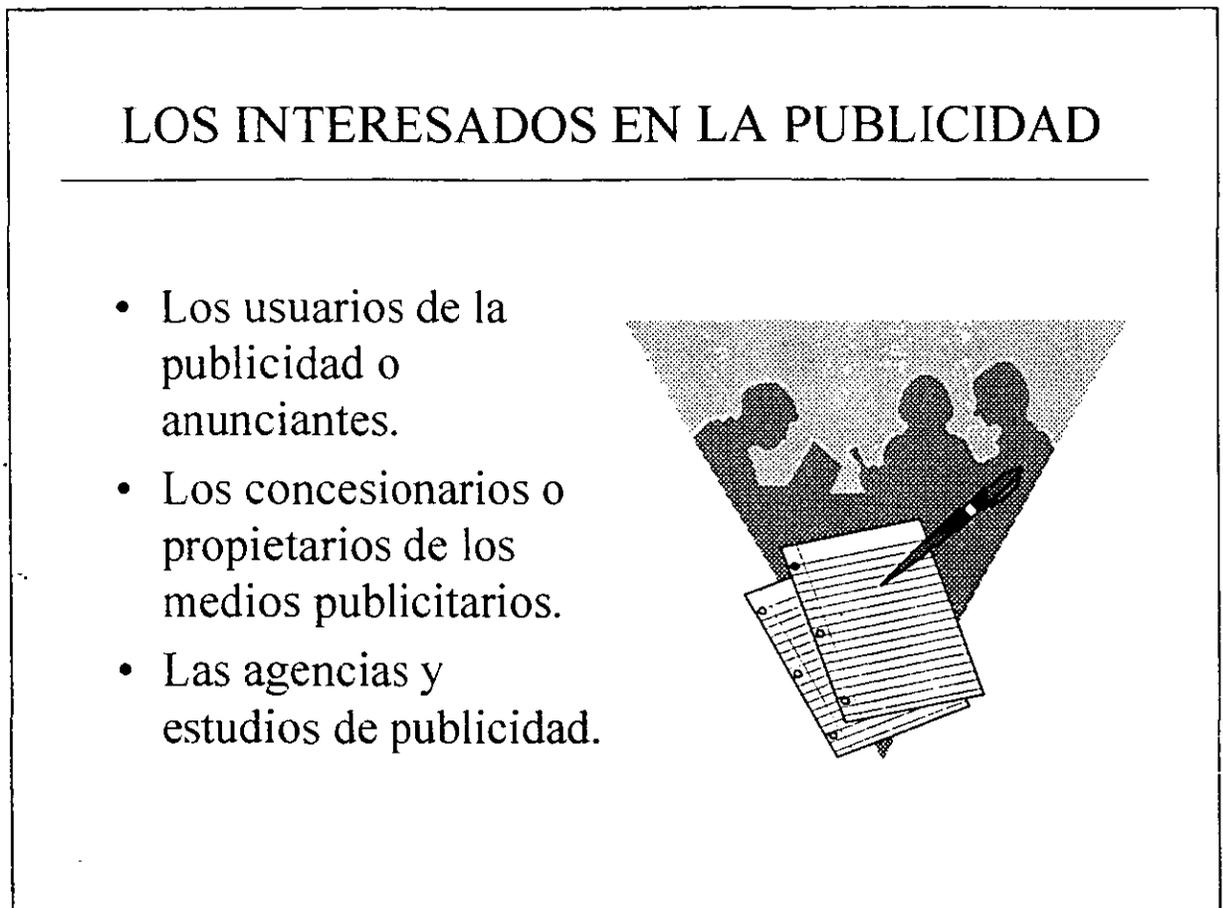


Figura 1.1.2.1 Los Interesados en la Publicidad.

5. Formular y definir el plan de publicidad y presentarlo al cliente para su discusión o aprobación.
6. Realizar la campaña, es decir, comprar el espacio en los medios y hacer llegar a los mismos el material necesario para la impresión, transmisión, etc.
7. Cooperar con los clientes para que la campaña publicitaria esté en armonía con su política de ventas.

A estos siete puntos, muchas agencias añaden otros servicios tales como son: el estudio del envase del producto, establecimiento del control de ventas, organización de seminarios de puesta al día de los vendedores, entre otros.

Estructura Organizativa de una Agencia de Publicidad.

Los sectores más importantes de una agencia de publicidad son:

1. Dirección.
2. Servicio de contacto con los clientes.
3. Servicio de **Marketing** (Mercadotecnia) e investigación de mercados.
4. Servicio de medios e investigación sobre los mismos.
5. Redacción.
6. Departamento Artístico.
7. Oficina de Producción y Tráfico.
8. Administración.

1. La Dirección se encarga de organizar el trabajo de los diversos sectores particulares de la agencia.
2. El servicio de ejecutivos de cuentas está compuesto por empleados no especializados en sectores particulares de la empresa, sino con un conocimiento adecuado de los diversos departamentos de la agencia. Su función es la de mantener las relaciones entre la agencia y el cliente.
3. La investigación de mercados, forma parte del servicio de marketing y proporciona a este último los datos en los que apoyar la política de ventas del producto y la estrategia sobre la que basar la campaña publicitaria. La oficina de marketing está compuesta por especialistas con una probada experiencia en investigación de mercados y conocer los problemas empresariales referentes a la producción y distribución de los artículos, el mercado de los consumidores y su posibilidad de desarrollo.
4. El servicio de medios comprende sectores como: la investigación de los medios, la planificación, la ejecución de la campaña y el control de la misma.
 - a. En la investigación de los medios se establecen las características de cada uno de los medios en base al tipo de público que alcanza, cualitativa y cuantitativamente.

- b. En la planificación se establece, qué medio es más apto, dados el tipo de producto considerado y su forma de distribución.
- c. En la ejecución de la campaña se establecen los contactos con los concesionarios o propietarios de los medios a contratar (el contrato se estipula a nombre del usuario de la publicidad).
- d. En el control se verifica que el desarrollo de la campaña esté de acuerdo con lo previsto en la planificación.
5. La redacción es la sección donde se redactan los mensajes publicitarios. El redactor debe expresarse con un lenguaje claro y adaptado al público que se dirige.
6. El departamento artístico comprende varios tipos de especialistas, que se encargan de reflejar gráficamente el mensaje redactado.

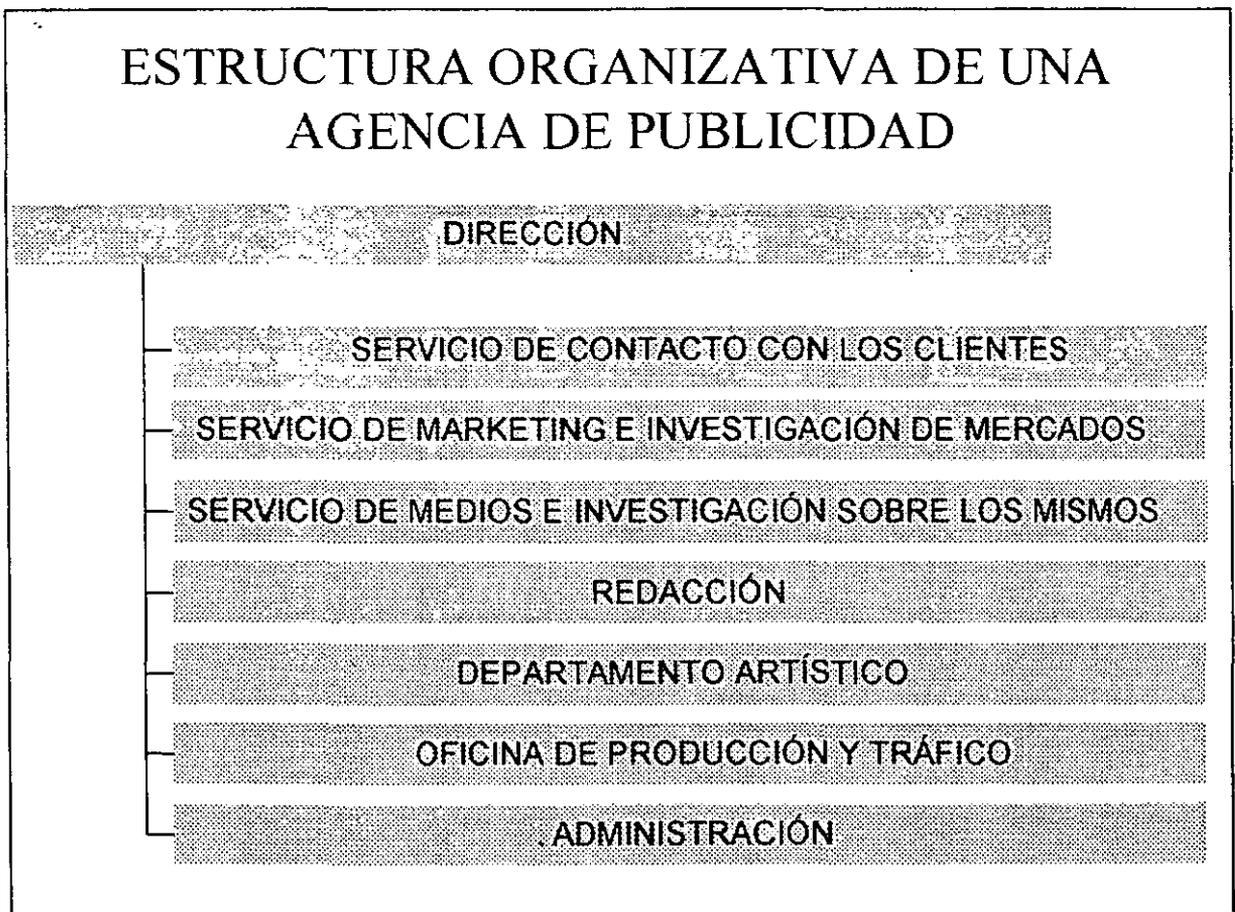


Figura 1.1.2.2 Organigrama de una Agencia de Publicidad.

7. El departamento de producción es el encargado de la realización del material en composición, fotografías, clichés, impresos, es decir, de todo lo que sirve en una campaña publicitaria. Coordina y controla los trabajos de los departamentos de redacción y artístico y también se encarga de enviar el material terminado a los editores propietarios de los medios publicitarios.
8. Por último, la oficina de administración, es la encargada de llevar la contabilidad interna y de los clientes, así como todos los aspectos administrativos (balances, presupuestos, gastos, etc.). En la figura 1.1.2.2 se muestra el organigrama de una Agencia de Publicidad.

1.1.3 La Campaña Publicitaria.

En Busca de Información.

Los datos sobre el mercado, sobre el producto, sobre los consumidores y sobre la competencia, indispensables al publicitario para planear sobre base "científica" el plan y la campaña, deberán ser proporcionados total o parcialmente por el servicio comercial de la empresa.

Comencemos nuestro análisis con los datos sobre el mercado. Todos los productos o servicios tienen su propio mercado potencial. El mercado por tanto puede ser definido como el total de los clientes potenciales en un área geográfica determinada en relación con un producto o servicio.

Una vez que se ha determinado cual es nuestro mercado, el publicitario deberá conocer las etapas que habrá de recorrer el producto para llegar desde el almacén del productor hasta la residencia del consumidor final. El conocimiento de los canales de distribución a través de los cuales el producto pasa desde la empresa hasta el cliente final, sirve para determinar si es también necesaria una acción publicitaria dirigida al revendedor del producto (minorista).

El estudio del mercado no puede prescindir del estudio de la competencia. Salvo el caso que la empresa opere en una situación de monopolio. Son empresas competidoras de la nuestra todas aquellas que fabrican productos o venden servicios idénticos o sustitutivos de los nuestros.

Identificados los competidores, es preciso proceder a la recopilación de información relativa a los mismos, tanto de los datos comerciales como los estrictamente publicitarios. El publicitario no se limita a estudiar la competencia únicamente sobre los datos y las informaciones recogidas. También examina con atención sus productos, estudia su composición, grado de rendimiento, el valor intrínseco y psicológico; interroga al consumidor para saber en qué encuentra "superior" o "inferior" al producto de la competencia con respecto al propio. De esta manera conoce con precisión sus puntos débiles y poder así enfrentar a la competencia atacando sus puntos vulnerables y con esto pretender que nuestro producto aparezca como mejor a todos los demás o al menos diferente.

Una vez conocido el mercado y la competencia; el publicitario necesita saber sobre las exigencias productivas de la empresa.

La empresa productora debe obrar con mucha cautela, no solo para afrontar una posible quiebra del producto, sino también para hacer frente a un éxito excesivo, lo cual si no se está preparado también representaría un problema. Para todas las empresas existe un óptimo de producción, y por tanto de venta, que no conviene traspasar.

Por último citaremos el estudio que el publicitario debe realizar sobre el producto en relación con el consumidor. En la economía de consumo, un producto por sí mismo, cuenta muy poco, por no decir nada. Adquiere un valor únicamente en función de las exigencias objetivas y psicológicas que logra satisfacer; es decir, si no se logra crear una necesidad de consumo del producto, por más bueno que sea este no se vende.

En la figura 1.1.3.1 se muestran los aspectos más importantes de una campaña publicitaria.

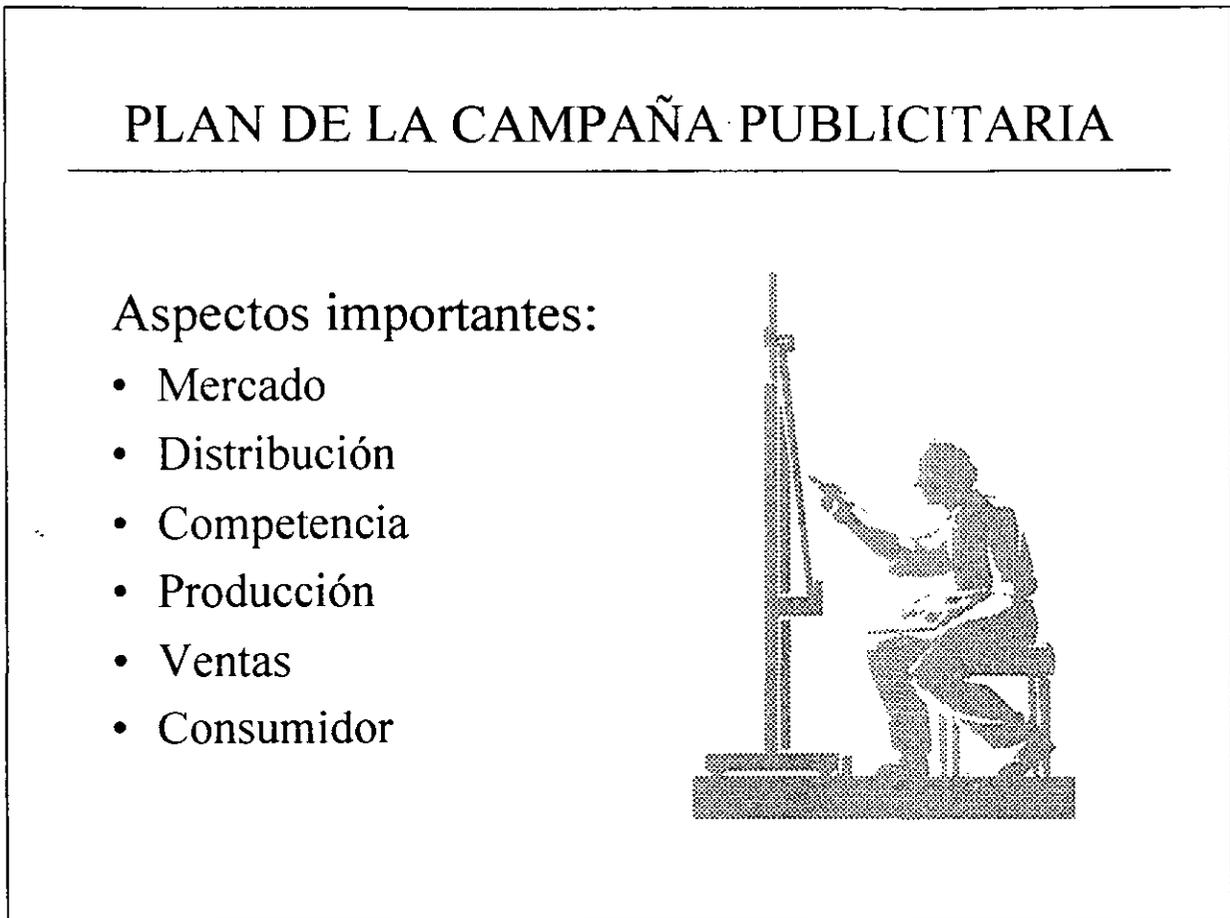


Figura 1.1.3.1 Plan de la Campaña Publicitaria.

Definición del Presupuesto y Objetivos Publicitarios.

El presupuesto publicitario es la asignación que la empresa decide poner a disposición de la publicidad para una determinada campaña o para todas las iniciativas publicitarias de un determinado período de tiempo.

Los sistemas más usados para establecer el volumen de la asignación publicitaria son:

1. Con un porcentaje sobre las ventas facturadas el año precedente o el año en curso.

2. En relación con los beneficios del año precedente o los previstos para el año en curso.
3. En relación con la finalidad de la campaña publicitaria que se debe hacer.
4. En relación con los gastos de la competencia.

El procedimiento más correcto para determinar el volumen del presupuesto publicitario, se desarrolla a través de las siguientes fases:

1. Determinación de los objetivos más específicos posibles, y que como tales permitan la posibilidad de un control final de su grado efectivo de consecución.
2. Determinación de las iniciativas publicitarias a emprender para alcanzar tales objetivos específicos y, por tanto, de la cifra mínima indispensable para asegurar las probabilidades óptimas de conseguirlo.
3. Reducción de los objetivos específicos a otros de importancia más modesta, y por tanto de menos costo, cuando la empresa no disponga de lo necesario para conseguir unos objetivos tan ambiciosos como los previstos al principio.
4. Control final de las acciones publicitarias y de su eficacia desde el punto de vista de la obtención efectiva de los objetivos preestablecidos.

1.1.4 Los Medios Publicitarios.

Los medios publicitarios a disposición del usuario son múltiples y sólo nos referiremos a los denominados "clásicos"; es decir: a la prensa, televisión, cine, radio y carteles. En la figura 1.1.4.1 se ilustran estos medios.

La Prensa.

La expresión "medio impreso " indica, desde el punto de vista publicitario, dos medios con características distintas: las revistas y diarios. Veamos cuáles son las diferencias estructurales que llevan a dos medios de la misma naturaleza a diferenciarse tanto:

1. El tipo de papel. Los diarios están impresos en papel más ordinario que las revistas, compuesto de una pasta más pobre en cola y por tanto más absorbente. La reproducción de las fotografías requiere una retícula grande.



Figura 1.1.4.1 Medios "Clásicos" de Publicidad.

2. El método de impresión. Los diarios se imprimen en rotativas tipográficas y el trabajo debe ser desarrollado en poquísimas horas, en tanto que las revistas están impresas casi siempre en fotgrabado y en plazos de tiempo de algunos días.
3. Las posibilidades de reproducción. Las revistas ofrecen una buena posibilidad de reproducción de las fotografías tanto en blanco y negro como en colores. El color en los últimos años ha llegado también a los diarios, pero las dificultades técnicas lo hacen muy costoso y con resultados no totalmente satisfactorios.

4. Las zonas de distribución. El diario normalmente se difunde con amplitud en una región: unos pocos constituyen la excepción y son distribuidos nacionalmente, aunque solamente en las grandes ciudades. La revista es de distribución nacional.
5. El tiempo de lectura. El diario usualmente "vive" un día, e incluso media jornada, mientras que la revista "vive" de una semana a un mes.

En la figura 1.1.4.2 se muestran las diferencias entre los diarios y las revistas.

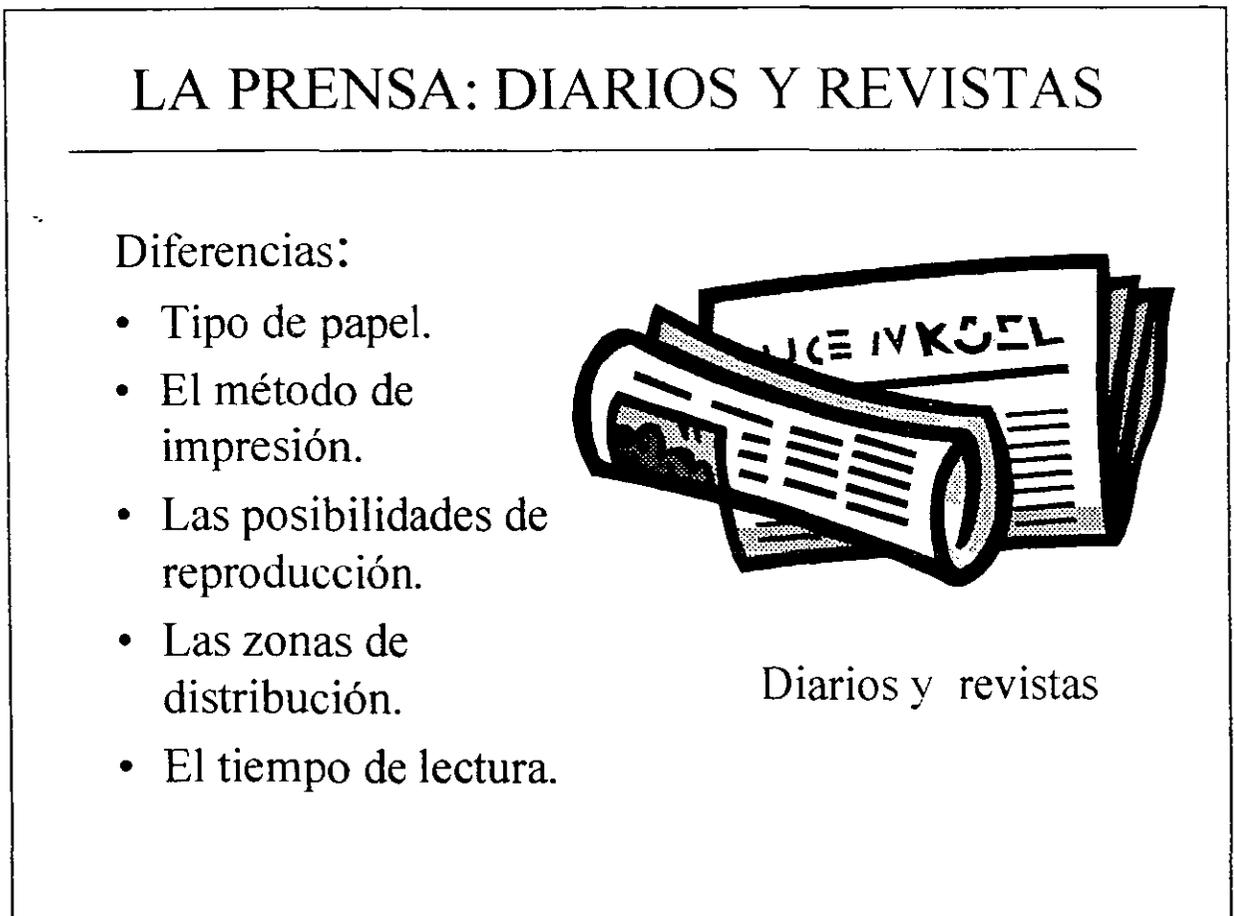


Figura 1.1.4.2 Diferencias entre Diarios y Revistas.

Tanto para los diarios como para las revistas existen grupos editoriales, es decir, varios diarios editados por un solo editor. El espacio publicitario se vende al anunciante bien directamente por el editor o mediante concesionarios. Los concesionarios son sociedades o individuos particulares que adquieren la concesión del espacio publicitario

de un periódico, garantizando al editor un mínimo anual de consumo, y revendiendo el espacio al precio establecido por ellos de acuerdo con los propietarios. Las medidas base de venta para la publicidad impresa son el milímetro columna y la página o fracción. El milímetro es generalmente usado en los diarios y semanarios, la página en las publicaciones mensuales y en las de otra periodicidad.

En los diarios, el precio se refiere a la reproducción en blanco y negro; el aumento por los colores viene determinado normalmente como un porcentaje sobre los costos por milímetro o por página. Las revistas tienen distinto precio de tarifa según se trate de anuncios en blanco y negro o en varios colores.

La posición del anuncio también influye en el precio; con un aumento que varía de un 10% a un 30% se pueden pedir posiciones fijas o especiales.

Los Diarios.

No existen actualmente diarios de distribución nacional, la cobertura es regional o provincial. Algunos de ellos tienen idealmente un valor nacional, pero en realidad su distribución fuera de la respectiva región y de algunas ciudades grandes, es bastante limitada.

Las Revistas.

Las revistas que interesan en el campo publicitario se dividen, de acuerdo precisamente con su periodicidad, en semanales, quincenales y mensuales, y por su contenido en informativas y de actualidad, familiares, deportivas, femeninas y de historietas.

Todas las revistas son de distribución nacional y no pueden ser usadas localmente sino en casos especiales. Los ejemplares vendidos alcanzan niveles de penetración y distribución notables, y además es posible a través de la elección de las revistas obtener una selección de público por sexos y clases socio-económicas.

Los semanarios representan el mejor medio impreso nacional desde el punto de vista de la distribución y los costos. El costo unitario del anuncio es generalmente más elevado en los semanarios, pero puesto que el número de lectores es mayor, el costo por lector es menor.

Se podría establecer esta distinción fundamental entre diario y semanario: el diario anuncia la existencia o la renovación de un producto, el semanario imprime el producto o la marca en la memoria del lector y favorece, por tanto, su demanda.

La Televisión.

Los espacios publicitarios se subastan a distintos concesionarios, por un periodo de un año, y son estos concesionarios los que contratan con las agencias de publicidad o con los anunciantes.

El espacio mínimo admitido como anuncio es de 15 segundos, existiendo espacios proporcionalmente superiores y admitiéndose también las inserciones de 20 segundos.

La televisión es un medio de masa, pero de gran prestigio al mismo tiempo. Sin embargo, hay que hacer notar que no todos los productos se adaptan a este medio, que requiere una distribución nacional y un mensaje adaptado a los diversos grupos socio-económicos.

La Radio.

Existe una multitud de emisoras, de las cuales sólo algunas tienen una difusión auténticamente nacional. La mayoría se escuchan únicamente en su provincia o en las provincias limítrofes.

Hay varias formas de publicidad radiofónica:

- Anuncios por palabras: incluidos en grupos dentro de la guía comercial; son leídos por los locutores de la emisora y carecen de música. Estos anuncios se miden según el número de palabras de que constan.
- Cintas: espacios mínimos de quince segundos, grabados con las voces que se desean y con posibilidad de incluir música, efectos especiales, etc. Se contratan por el número de segundos.
- Programas: con la duración mínima de cinco minutos, dentro de los cuales se admite un 10 % de publicidad, repartido en dos espacios colocados al principio y al final. El resto del espacio está dedicado al programa propiamente dicho.

En la figura 1.1.4.3 se ilustran los medios electrónicos de publicidad (radio y TV.)



Figura 1.1.4.3 Radio y Televisión.

El Cine.

La inserción de los anuncios se efectúa en los descansos de la película comercial o antes del comienzo de ésta, por lo que sirve frecuentemente de entretenimiento para los espectadores que permanecen dentro de la sala (ver figura 1.1.4.4). Las formas que puede adoptar el anuncio cinematográfico son las siguientes:

- Filmlet o corto: de quince segundos de duración.
- Película normal: con una duración mínima de treinta segundos, se proyectan antes de los filmlets.
- Documental: película de varios minutos de duración, con un contenido generalmente informativo (explicación del proceso de trabajo de una fábrica, exposición de sus instalaciones, etc.).
- Diapositiva: imagen fija, con una duración de cinco segundos; antiguamente era la única forma de publicidad cinematográfica; en la actualidad ha desaparecido casi por completo, sustituida por las películas y filmlets.

Los anuncios en este medio se contratan a través de concesionarios de publicidad de las salas existentes y se realizan por semanas o meses completos.

Los cinematógrafos están divididos en distintas categorías, a cada una de las cuales corresponde un precio distinto, siguiéndose criterios como: calidad y lujo de las instalaciones, aforo, número de sesiones proyectadas por semana, categoría del cine, tipo de público que acude normalmente, población en que radica, zona en que está situado dentro de la población, etc.

Los Carteles.

Con el término carteles se indica solamente una parte de los anuncios públicos los cuales comprenden: los rótulos, los carteles de carreteras, los anuncios en los medios de transporte y cualesquiera otros medios visuales que sean totalmente visibles en los

PUBLICIDAD EN CINE

- Filmlet o corto.
- Pelicula normal.
- Documental.
- Diapositiva.



Figura 1.1.4.4 Publicidad en el Cine.

lugares públicos. Por carteles se entienden los anuncios dibujados o fotografiados en papel, cartón o material similar, y fijados en los lugares públicos. El costo de difusión de los carteles está subordinado al Ayuntamiento en que se fijen. Cada Ayuntamiento fija un canon por dicha difusión, cuya cuantía depende de su libre arbitrio.

El cartel es uno de los medios publicitarios más antiguo y demás probada utilidad; el único inconveniente que presenta es la necesidad de un mensaje breve de comprensión inmediata. El cartel es el medio capaz de alcanzar al mayor número de personas: potencialmente, con una fijación nacional de carteles se puede alcanzar a todos los habitantes del país. Por el contrario, es posible fijar un solo cartel en una sola plaza.

En la figura 1.1.4.5 se muestran los tipos de carteles.

LOS CARTELES

Tipos de carteles:

- Rótulos.
- Carteles de carreteras.
- Anuncios en los medios de transporte
- Cualquier medio visual en los lugares públicos.

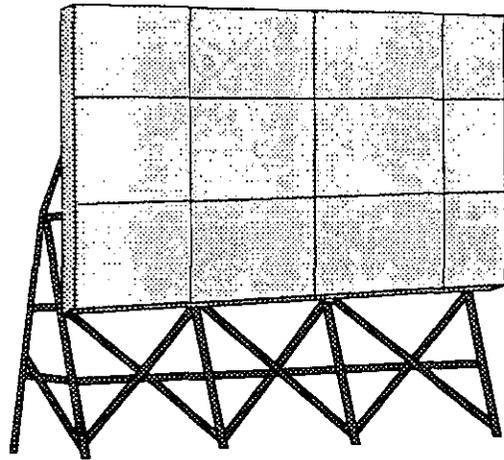


Figura 1.1.4.5 Tipos de Carteles.

1.1.5 El Mensaje Publicitario.

El mensaje publicitario interesa desde dos puntos de vista diferentes:

- a) Bajo el punto de vista redaccional.
- b) Bajo el punto de vista gráfico.

Redacción del Mensaje Publicitario.

Desde el punto de vista redaccional, el mensaje publicitario se expresa mediante un texto, subdividido de diversas formas, pero en el cual es posible distinguir tres elementos fundamentales:

1. El título, impreso tipográficamente con caracteres más grandes en relación con las otras partes del texto.
2. El subtítulo, que algunas veces completa lo declarado en el título, proporciona una primera respuesta sintética e invita, implícita o explícitamente, al lector a documentarse de una manera más exhaustiva en las líneas siguientes.
3. El cuerpo central, o texto propiamente dicho, que desarrolla plenamente el razonamiento con las argumentaciones relativas.

Título y Subtítulo.

La misión del título consiste en atraer la atención del público y en suscitar en él el interés. Si el título no atrae al lector, difícilmente leerá el texto; y un texto no leído no puede tener éxito en su misión de venta. Algunas veces, a falta de un título eficaz, la ilustración desarrolla sus funciones. Pero, sin duda, lo más conveniente es combinar un buen título con una buena ilustración.

Por su objetivo inmediato, los títulos pueden clasificarse en:

- a) Título noticia, cuya misión es resaltar los puntos de mayor interés desarrollados en el texto. El título constituye noticia por sí solo para atraer la atención y el interés del texto.
- b) Títulos afirmativos, son aquellos que contienen afirmaciones cuyo contenido es de interés personal para el lector. Las tesis que sostienen deben ser dignas de relieve y consideración. Los títulos afirmativos sirven para superar los estados de duda e incertidumbre del cliente potencial.
- c) Títulos-sugerencia es un título afirmativo en el que la sugerencia va seguida de una promesa de buenos resultados cuando sea aceptado por el lector. También el título sugerencia apela al interés personal del lector; el consejo dado tiende a ayudar al cliente potencial a resolver un problema específico o a evitar una indecisión.

- d) **Títulos selectivos.** El título realiza una primera selección de carácter general entre los lectores. Cualquier título de anuncio clasificado en otro tipo puede ser también selectivo.
- e) **Título-curiosidad.** Despierta el interés de la gente apelando a la curiosidad. Se espera que el título impulse al lector a buscar en el texto del anuncio la satisfacción al estímulo curioso voluntariamente sometido. Este tipo de título permite al redactor de los textos publicitarios una amplia libertad en el empleo de su imaginación.
- f) **Título-etiqueta.** En algunos casos, la denominación de la empresa puede servir por sí sola de título; esto sucede cuando el producto y la marca son tan conocidos y apreciados, que basta con poner en evidencia su nombre para suscitar el interés del lector.

"Slogan" Publicitario.

La palabra slogan deriva de una expresión gaélica - el gaélico es la lengua de las poblaciones de estirpe céltica - que traducida literalmente significa "grito de guerra" . Para el publicitario; el slogan es esencialmente una sentencia o frase destinada a suscitar en la mente del lector una actitud positiva en relación con el producto, el servicio o la empresa. El slogan publicitario también se ha definido como "la expresión breve y sugestiva de una idea comercial", y debe poseer las siguientes cualidades:

- Positivo y breve.
- Comprensible para todos y fácil de retener en la memoria.
- Dinámico y preciso, es decir, deben utilizarse, en lo posible, términos apropiados.
- Debe ser "vocativo", o sea, debe dirigirse a cada uno de modo explícito.
- No se debe usar el tiempo condicional de los verbos, puesto que suena como dubitativo y no favorece las ventas.

Sin embargo, estas reglas no deben convertirse en dogmas y conviene someterlas a revisión cada vez que se plantea la posibilidad de su uso.

Por otra parte, junto con los adverbios y adjetivos demostrativos; son ampliamente usados en los slogans los términos cargados de significados: trabajar, festejar, economizar, o bien los adjetivos calificativos. Las palabras de afecto (amor, mujer, corazón, deseo), aunque muy explotadas, son siempre eficaces gracias a su contenido emotivo.

El caso de los subtítulos, utiliza paso a paso las mismas consideraciones desarrolladas para el título, por lo que no será necesaria repetir su explicación.

El Texto de un Anuncio.

Desde el punto de vista de la "sustancia", podemos clasificar a los textos publicitarios en tres grandes grupos:

1. En un primer grupo, encontramos los textos que utilizan argumentos racionales, es decir, que desarrollan un tipo de argumento lógico, compacto, totalmente asentado sobre las cosas, los hechos, el rigor de la demostración matemática o casi matemática.
2. En el segundo grupo aparecen los textos que se apoyan sobre un mensaje emotivo, tendiente a descartar la lógica y la demostración; jugando, por el contrario, con el sentimiento y los aspectos inconscientes o subconscientes de la personalidad.
3. En un tercer grupo, están los textos mixtos, que combinan en diversa medida ambos aspectos, alternando con el rigor demostrativo, el halago del sentimiento y el empleo de la sugestión.

El mensaje racional, como su misma denominación lo indica, se dirige a la razón. Intenta convencer al lector de que adquiera la marca X en lugar de las marcas competidoras K, Y o Z, y proporciona con esta finalidad una serie de elementos, es decir, relaciona una serie de datos, que tratan de probar la superioridad de la marca promocionada sobre todas las demás.

Los textos basados sobre mensajes emotivos adquieren especial relevancia. Sobre todo en el sector de los bienes de consumo en gran escala, de adquisición por impulso, que descansa sobre la base de motivaciones inconscientes más que de argumentaciones racionales, abarca a una vasta área del conjunto de adquisiciones de los comercios tradicionales y especialmente de los auto servicios y supermercados.

Finalmente, los textos mixtos, que recurren conjuntamente en un mismo anuncio a argumentaciones de orden racional y a apelaciones de orden emotivo, son los más numerosos. Una llamada emotiva suscita con facilidad la atención y el interés del lector, algunas veces sin que el lector se dé cuenta de ello. Pero el publicitario no puede limitarse a suscitar el interés del público; debe convencer al cliente potencial de que el producto promocionado le conviene más que cualquier otro producto de la competencia. El proceso de persuasión del cliente puede implicar el paso de un texto emotivo a otro racional.

Si un texto publicitario acierta a mantener despierta en el lector la carga emotiva a medida que se desarrollan las argumentaciones persuasivas, el objetivo perseguido se alcanzará totalmente. La capacidad creativa de un redactor publicitario se mide también por este parámetro.

Dibujos e Ilustraciones de un Anuncio Publicitario.

Desde un punto de vista gráfico, los elementos principales de un anuncio publicitario son:

1. El título y eventualmente el subtítulo.
2. El texto.
3. La ilustración o ilustraciones.
4. El material auxiliar.
5. El logotipo.

6. El espacio en blanco.

El dibujante publicitario tiene amplias posibilidades de elección en la colocación del título y subtítulo del anuncio.

El hábito normal del lector, lo lleva a comenzar las páginas impresas por arriba, y a fijarse en los elementos de arriba abajo. Por ello, los títulos se disponen la mayoría de las veces en la parte superior de la página, o más bien, inmediatamente debajo de la ilustración o al lado de ésta.

El texto interesa obviamente al dibujante, no desde el punto de vista literario, sino como "bloque" o "cuerpo" destinado a ocupar cierta superficie del anuncio en una posición determinada.

Se dice en general que un dibujo, una foto, una imagen en suma, valen más que mil palabras. Lo que es cierto en líneas generales, es verdad también en publicidad, naturalmente; en algunos casos la ilustración es autosuficiente para lograr los objetivos que se pretenden obtener. En otros casos, sin embargo, la ilustración por sí sola no basta, y es necesario por ello recurrir a la palabra, sea como título o subtítulo, sea como verdadero y propio texto. Un anuncio puede tener una o varias ilustraciones dependiendo de las exigencias intrínsecas del mensaje.

La expresión "material auxiliar" es un término con el que se indican varios elementos del **layout** (composición), como los espacios periféricos, o márgenes del anuncio, las líneas continuas o "filetes" y las áreas delimitadas o "cajas": todos los elementos, en resumen, que el dibujante y el tipógrafo han de tener a mano para utilizarlos en momentos variados y repetitivos.

El logotipo es la marca de fábrica del usuario de la publicidad. El fin fundamental del logotipo es favorecer la identificación inmediata del usuario de la publicidad. El logotipo puede estar constituido por un símbolo o por cualquier elemento que identifique la

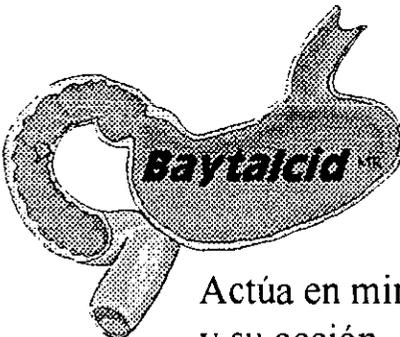
marca; además para que adquiera más realce se rodea del suficiente espacio en blanco.

La habilidad en el empleo de los espacios en blanco. es una de las más importantes para un artista publicitario; de hecho, los espacios en blanco junto con otros elementos adecuadamente seleccionados, puede rodear de un atmósfera de prestigio al anuncio completo. Pero un espacio en blanco mal distribuido, puede subdividir el anuncio en una serie de secciones antiestéticas y antifuncionales que distraigan la atención del lector y la desvíen, en lugar de concentrarla en uno o más puntos prefijados. La figura 1.1.5 muestra las características de un mensaje publicitario.

EL MENSAJE PUBLICITARIO

- **Redaccional.**
 - Título
 - Subtítulo
 - Cuerpo Central o Texto
- **Gráfico.**
 - Título
 - Texto
 - Ilustración
 - Material Auxiliar
 - Logotipo
 - Espacio en Blanco

En el mundo de la
Acidez Gástrica



Actúa en minutos
y su acción
dura por horas.

El antiácido de Bayer
y si es Bayer...



Figura 1.1.5 Características de un Mensaje Publicitario.

1.2 CONCEPTOS BÁSICOS DE INTERNET.

1.2.1 Como se Inicio Internet.

Después de la Segunda Guerra Mundial comenzó un período denominado Guerra Fría, en la cual los países se dividieron en dos bloques en constante pugna por militarizarse y conquistar espacios en el planeta y fuera de él. Por un lado los Estados Unidos y sus aliados en América y Europa, y por otro lado la Unión de Repúblicas Socialistas Soviética, URSS, y sus aliados comunistas. En 1958 se suceden dos eventos importantes en esta carrera armamentista: Rusia lanza su primer satélite artificial, el Sputnik, y el Departamento de la Defensa de los Estados Unidos crea la **Advanced Research Projects Agency**, (ARPA), con la idea de hacer a ese país líder en tecnología militar.

En 1962 Paul Baran de Rand Corporation presenta un informe titulado "**On Distributed Communications Networks**" donde se cuestiona la manera de crear un sistema de comunicaciones que permitiera a la milicia de los Estados Unidos sobreponerse a cualquier ataque (incluso nuclear), que pudiera dañar alguna parte de él. Este sistema estaría basado en una extensa red con muchas ramificaciones (universidades, instituciones de investigación y desarrollo y la milicia, por supuesto, en los países aliados), comunicado mediante paquetes conmutados de información. Estas y otras inquietudes inspiraron a la Defensa Norteamericana para crear la red militar APARNET, la cual quedó instalada con cuatro nodos iniciales (el primero en la Universidad del Sur de California en los Ángeles) operando bajo el protocolo **Network Control Protocol**, NCP, entre septiembre y diciembre de 1969.

En el modelo APARNET, la comunicación siempre ocurre entre una computadora fuente y una destino. La red asume por sí misma que es factible; cualquier parte de la red puede desaparecer en cualquier momento. La red fue diseñada para requerir un mínimo de información de las computadoras que forman parte de ella. Para enviar un

mensaje en la red, una computadora sólo tiene que poner la información en un sobre, llamado paquete de protocolo Internet (IP: Internet Protocol) y le asigna el domicilio destino en forma correcta. Las computadoras que se comunican -no la red- tienen la responsabilidad de asegurar que la comunicación se lleve a cabo. La filosofía era que cada computadora en la red se pudiese comunicar, como un elemento individual, con cualquier otra computadora.

Con estas especificaciones, Estados Unidos fue capaz de desarrollar una red que funcionara (la antecesora de la red Internet) y los usuarios académicos e investigadores que tenían acceso a ella rápidamente se volvieron adictos. La demanda por la red muy pronto se esparció.

Para abril de 1971 se hallaban conectadas 23 computadoras denominadas **host** (anfitriones) en 15 nodos ubicados en las principales Universidades, Institutos de Investigación, Institutos Tecnológicos, el CDS, los laboratorios Lincoln y el Centro de Investigaciones Ames de la NASA de los Estados Unidos.

En 1973 se desarrolla la primer conexión internacional de Apanet con el nodo de la Universidad de Londres en Inglaterra. En 1974 Vint Cery y Bob Kahn publican las especificaciones de un nuevo protocolo más abierto y estándar en su "**A Protocol for Packet Network Intercommunication**". Este evento marca el inicio del que se utiliza en la actualidad: **Transmision Control Protocol (TCP)**. Por ese entonces los laboratorios Bell de AT&T desarrollaban el protocolo Unix-to-Unix Copy, UUCP, que se distribuyó con el sistema operativo UNIX, que es el que más se utiliza en las computadoras de las grandes universidades, entidades de gobierno y centros de investigación.

El crecimiento de la red fue lento al principio, de manera tal que para 1984, quince años después de su nacimiento, había 1,024 *anfitriones* conectados; sin embargo, a partir de la segunda mitad de la década de los ochenta se nota un explosivo

crecimiento que la lleva a terminar el año 1989 con aproximadamente 160,000 y en menos de tres años, en 1992, alcanza la cifra de 1,000,000 de computadoras y más de 8,000 redes interconectadas a la gran red.

La tabla 1.2.1 presenta este interesante fenómeno de crecimiento de las computadoras interconectadas a Internet.

FECHA	ANFITRIONES	REDES	DOMINIOS
DICIEMBRE 1969	4		
ABRIL 1971	23		
JUNIO 1974	62		
MARZO 1977	111		
AGOSTO 1981	213		
MAYO 1982	235		
AGOSTO 1983	562		
OCTUBRE 1984	1024		
OCTUBRE 1985	1961		
FEBRERO 1986	2308		
NOVIEMBRE 1986	5089		
DICIEMBRE 1987	28174		
JULIO 1988	33000	204	
OCTUBRE 1988	56000	291	
ENERO 1989	80000	346	
JULIO 1989	130000	603	3900
OCTUBRE 1989	159000	809	
OCTUBRE 1990	313000	2063	9300
ENERO 1991	376000	2338	
JULIO	535000	3086	16000
OCTUBRE 1991	617000	3556	18000
ENERO 1992	727000	4526	

ABRIL 1992	890000	5291	20000
JULIO 1992	992000	6031	16300
OCTUBRE 1992	1136000	7354	18100
ENERO 1993	1313000	9118	21000
ABRIL 1993	1486000	11252	22000
JULIO 1993	1776000	14121	26000
OCTUBRE 1993	2056000	17979	28000
ENERO 1994	2217000	23494	30000
JULIO 1994	3212000	36153	46000
OCTUBRE 1994	3864000	41250	56000
ENERO 1995	4852000	46318	71000
ABRIL 1995	5706114	56842	
JULIO 1995	6710582	69767	120000
OCTUBRE 1995	7891869	85646	
ENERO 1996	9281102	105155	240000
ABRIL 1996	10914886	129,130	
JULIO 1996	12836272	158598	488000
OCTUBRE 1996	15095885	194822	
ENERO 1997	17753266		
ABRIL 1997	20878436		
JULIO 1997	24553739		
OCTUBRE 1997	28876019		
ENERO 1998	33959165		
JULIO 1998	46967385		

Tabla 1.2.1 Crecimiento de Conectividad

1.2.2 Cómo Funciona Internet.

El sistema de redes moderno está construido sobre el concepto de "niveles o capas de

servicio". Usted empieza tratando de mover bits de un lugar a otro, perdiendo algunos en el trayecto. Este nivel se compone de cables y hardware, y no necesariamente de cables muy confiables. Después, agrega una capa de software básico que permite aislar los problemas del hardware. Incorpora otra capa de software para dar al software básico algunas características deseadas. Continúa agregando funcionalidad e inteligencia a la red, capa por capa, hasta que obtiene algo amigable y útil.

Redes de Conmutación de Paquetes.

Cuando uno se trata de imaginar que es Internet y cómo opera, es normal pensar en un sistema telefónico. Después de todo, ambos son electrónicos, ambos permiten abrir una conexión y transferir información, e Internet esta compuesta principalmente por líneas telefónicas permanentemente dedicadas a este uso. Desafortunadamente, esto crea una idea errónea y provoca mucha confusión sobre la forma en la que opera Internet. La red telefónica es una red de conmutación de circuitos. Cuando usted habla por teléfono, se separa una parte de la red para dedicarla a atender su llamada. Aun cuando no esté utilizando su parte de la red (por ejemplo cuando la línea está en espera), ésta es inaccesible por otras personas; por lo que provoca una subutilización de un recurso muy costoso: la red.

Un mejor modelo para comparar Internet es el Servicio Postal. El Servicio Postal es una red de conmutación de paquetes. Usted no cuenta con una parte de la red dedicada a sus actividades. Lo que se quiere enviar se mezcla con los mensajes de otras personas, se ponen en un conducto, se transfiere a otra Oficina Postal y se clasifica todo nuevamente. Aunque las tecnologías son completamente diferente, el Servicio Postal es sorprendentemente similar.

El Protocolo Internet, IP.

Un cable puede llevar información de un lugar a otro. Sin embargo ya se sabe que Internet puede hacer que la información llegue a distintos lugares distribuidos en todo el mundo. ¿Cómo sucede esto? Las diferentes partes de Internet están conectadas por un conjunto de computadoras llamadas *enrutadores*, que interconectan las redes. Estas redes pueden ser Ethernets, token ring o en ocasiones líneas telefónicas, como se muestra en la figura 1.2.2.1.

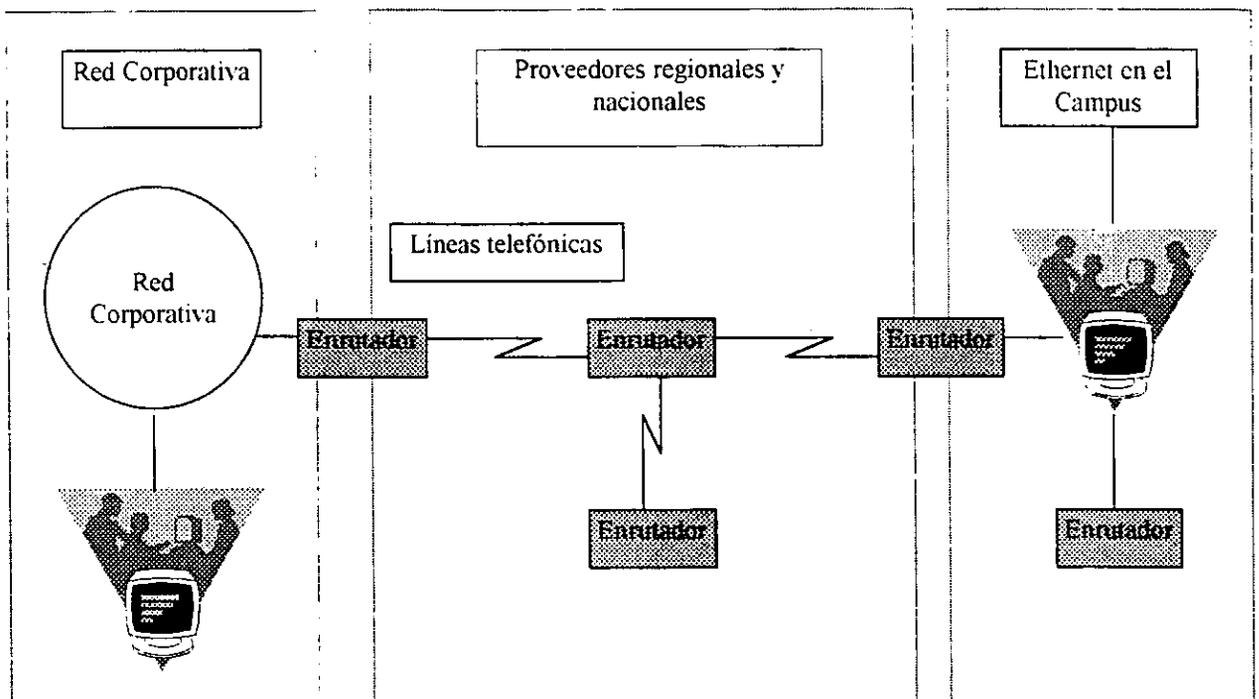


Figura 1.2.2.1 Hardware de Internet.

Las líneas telefónicas y las redes Ethernets son equivalentes a los camiones y aviones del Servicio Postal. Son el medio a través del cual el correo va de un lugar a otro. Los enrutadores son sucursales postales; estos equipos deciden cómo dirigir la información ("paquetes"), de la misma forma que una oficina postal decide cómo distribuir los sobres por correo. No toda subestación o todo enrutador cuenta con una conexión a cada uno de los otros enrutadores de la red. Si se envía un sobre de correo desde Dixville Notch, New Hampshire, con destino a Boonville, California, la Oficina Postal no reserva un avión de New Hampshire a California para llevarlo, sino que se

envía el sobre a la sucursal de correo y ésta a su vez lo envía a otra, y así sucesivamente hasta alcanzar su destino final. Esto significa que cada subestación solo necesita conocer las conexiones con las que cuenta y cuál es el mejor "siguiente paso" para acercar el paquete a su destino. Internet trabaja de manera similar: un enrutador se fija en el destino de la información y decide a donde enviarla. El enrutador elige cual es el enlace más apropiado para enviar la información.

¿Cómo sabe la red a donde se dirige la información? Si se quiere enviar una carta, no basta con poner el papel escrito en el buzón y esperar a que sea entregado. Es necesario poner el papel con la información en un sobre, escribir el domicilio del destinatario y pegar los timbres postales.

De la misma manera que la Oficina Postal tiene reglas que definen la operación de su red, también existen reglas que definen la operación de Internet. Las reglas son llamadas protocolos. El protocolo Internet (IP) se hace cargo de establecer domicilios o se asegura de que los enrutadores sepan qué hacer con la información que les llega. Continuando con la analogía de la Oficina Postal, el Protocolo Internet trabaja justo como un sobre postal. Ver figura 1.2.2.2.

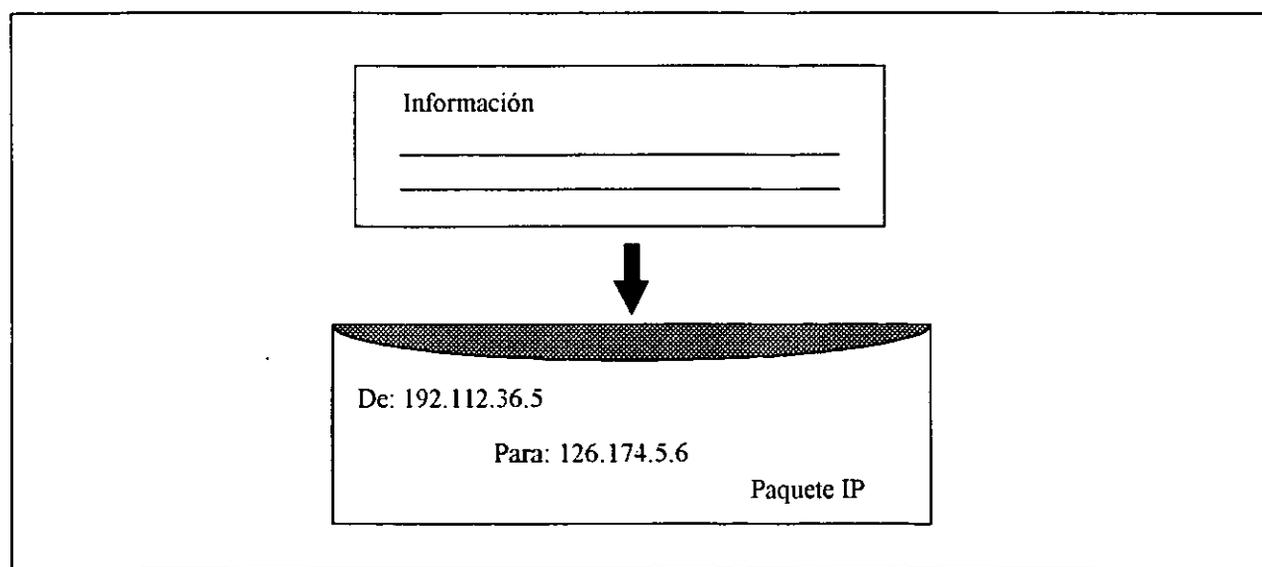


Figura 1.2.2.2 Sobres IP.

El protocolo IP identifica a cada ordenador que se encuentre conectado a la red mediante su correspondiente dirección. Esta dirección es un número de 32 bits que debe ser único para cada *host*, y normalmente suele representarse como cuatro cifras de 8 bits separadas por puntos.

La dirección de Internet (IP Address) se utiliza para identificar tanto al ordenador en concreto como la red a la que pertenece, de manera que sea posible distinguir a los ordenadores que se encuentran conectados a una misma red. Con este propósito, y teniendo en cuenta que en Internet se encuentran conectadas redes de tamaños muy diversos, se establecieron tres clases diferentes de direcciones, las cuales se representan mediante tres rangos de valores:

- **Clase A:** Son las que en su primer byte tienen un valor comprendido entre 1 y 126, incluyendo ambos valores. Estas direcciones utilizan únicamente este primer byte para identificar la red, quedando los otros tres bytes disponibles para cada uno de los *hosts* que pertenezcan a esta misma red. Esto significa que podrán existir más de dieciséis millones de ordenadores en cada una de las redes de esta clase. Este tipo de direcciones es usado por redes muy extensas, pero hay que tener en cuenta que sólo puede haber 126 redes de este tamaño. ARPAnet es una de ellas, existiendo además algunas grandes redes comerciales, aunque son pocas las organizaciones que obtienen una dirección de "clase A". Lo normal para las grandes organizaciones es que utilicen una o varias redes de "clase B".
- **Clase B:** Estas direcciones utilizan en su primer byte un valor comprendido entre 128 y 191, incluyendo ambos. En este caso el identificador de la red se obtiene de los dos primeros bytes de la dirección, teniendo que ser un valor entre 128.1 y 191.254 (no es posible utilizar los valores 0 y 255 por tener un significado especial). Los dos últimos bytes de la dirección constituyen el identificador del *host* permitiendo, por consiguiente, un número máximo de 64516 ordenadores en la misma red. Este tipo de direcciones tendría que ser suficiente para la gran

mayoría de las organizaciones grandes. En caso de que el número de ordenadores que se necesita conectar fuese mayor, sería posible obtener más de una dirección de "clase B", evitando de esta forma el uso de una de "clase A".

- **Clase C:** En este caso el valor del primer byte tendrá que estar comprendido entre 192 y 223, incluyendo ambos valores. Este tercer tipo de direcciones utiliza los tres primeros bytes para el número de la red, con un rango desde 192.1.1 hasta 223.254.254. De esta manera queda libre un byte para el *host*, lo que permite que se conecten un máximo de 254 ordenadores en cada red. Estas direcciones permiten un menor número de *host* que las anteriores, aunque son las más numerosas pudiendo existir un gran número de redes de este tipo (más de dos millones). La tabla 1.2.2 resume estas clasificaciones.

CLASE	PRIMER BYTE	IDENTIF. DE RED	IDENTIF. DE HOST	NÚMERO DE REDES	NÚMERO DE HOST
A	1 .. 126	1 byte	3 byte	126	16.387.064
B	128 .. 191	2 byte	2 byte	16.256	64.516
C	192 .. 223	3 byte	1 byte	2.064.512	254

Tabla 1.2.2 Direcciones IP de Internet.

En la clasificación de direcciones anterior se puede notar que ciertos números no se usan. Algunos de ellos se encuentran reservados para un posible uso futuro, como es el caso de las direcciones cuyo primer byte sea superior a 223 (clases D y E, que aún no están definidas), mientras que el valor 127 en el primer byte se utiliza en algunos sistemas para propósitos especiales. También es importante notar que los valores 0 y 255 en cualquier byte de la dirección no pueden usarse normalmente por tener otros propósitos específicos.

El número 0 está reservado para las máquinas que no conocen su dirección, pudiendo utilizarse tanto en la identificación de red para máquinas que aún no conocen el número de red a la que se encuentran conectadas, en la identificación de *host* para máquinas que aún no conocen su número de *host* dentro de la red, o en ambos casos. El número 255 tiene también un significado especial, puesto que se reserva para el *broadcast*.

El *broadcast* es necesario cuando se pretende hacer que un mensaje sea visible para todos los sistemas conectados a la misma red. Esto puede ser útil si se necesita enviar el mismo datagrama a un número determinado de sistemas, resultando más eficiente que enviar la misma información solicitada de manera individual a cada uno. Otra situación para el uso de *broadcast* es cuando se quiere convertir el nombre por dominio de un ordenador a su correspondiente número IP y no se conoce la dirección del servidor de nombres de dominio más cercano.

Lo usual es que cuando se quiere hacer uso del *broadcast* se utilice una dirección compuesta por el identificador normal de la red y por el número 255 (todo unos en binario) en cada byte que identifique al *host*. Sin embargo, por conveniencia también se permite el uso del número 255.255.255.255 con la misma finalidad, de forma que resulte más simple referirse a todos los sistemas de la red.

El *broadcast* es una característica que se encuentra implementada de formas diferentes dependiendo del medio utilizado, y por lo tanto, no siempre se encuentra disponible. En ARPAnet y en las líneas punto a punto no es posible enviar *broadcast*, pero sí que es posible hacerlo en las redes *Ethernet*, donde se supone que todos los ordenadores prestarán atención a este tipo de mensajes.

En el caso de algunas organizaciones extensas puede surgir la necesidad de dividir la red en otras redes más pequeñas (*subnets*). Como ejemplo podemos suponer una red de clase B que, naturalmente, tiene asignado como identificador de red un número de

dos bytes. En este caso sería posible utilizar el tercer byte para indicar en qué red *Ethernet* se encuentra un *host* en concreto.

Esta división no tendrá ningún significado para cualquier otro ordenador que esté conectado a una red perteneciente a otra organización, puesto que el tercer byte no será comprobado ni tratado de forma especial. Sin embargo, en el interior de esta red existirá una división y será necesario disponer de un software de red especialmente diseñado para ello. De esta forma queda oculta la organización interior de la red, siendo mucho más cómodo el acceso que si se tratara de varias direcciones de clase C independientes.

Regresando a nuestra analogía del funcionamiento de Internet con una Oficina Postal, es importante mencionar que cuando se envía un mensaje, una parte de la información de la dirección va al principio del mensaje; estos datos dan a la red información suficiente para hacer la entrega del paquete.

Como ya mencionamos, una dirección IP consta de cuatro números, cada uno menor a 256. Cuando dichos números se escriben, se separan por puntos, como se muestra a continuación:

192.112.36.5

128.174.5.6

En efecto, el domicilio está compuesto por varias partes. Como Internet es una red de redes, los primeros números del domicilio indican a los enrutadores cuál es la red a la que usted pertenece. Los últimos números indican qué computadora personal o equipo anfitrión de la red debe recibir el paquete. Bajo este esquema, cada computadora en Internet tiene un domicilio único. Considere el domicilio "Río Pánuco 170, Cuauhtémoc, D. F.". La parte "Cuauhtémoc, D. F." es como la parte del domicilio correspondiente a la red, el cual permite al sobre llegar a la Oficina Postal correcta, que es la que tiene la información correcta de las calles en una área determinada. La parte "Río Pánuco 170"

es como el domicilio del equipo anfitrión; este identifica a un buzón particular en el área de servicio de la Oficina Postal. La oficina concluye su trabajo cuando entrega el correo a la oficina local correcta y ésta lo pone en el buzón correcto. De la misma forma, Internet concluye su trabajo cuando los enrutadores llevan la información a la red local correcta y ésta entrega dicha información a la computadora personal o equipo anfitrión correctos, localizados en dicha red.

Por muchas razones prácticas (sobre todo por limitaciones de hardware), la información enviada a través de las redes IP se divide en pedazos de tamaño distinto, llamados paquetes. La cantidad de información en un paquete normalmente se encuentra entre 1 y aproximadamente 1500 caracteres de largo. Esto previene que cualquier usuario monopolice la red, permitiendo que todos tengan un acceso equitativo. También significa que cuando la red se sobrecarga, su comportamiento sólo desmerece un poco para todos los usuarios: la red no se inutiliza cuando algunos usuarios la monopolizan.

Una de las propiedades más importantes de Internet es que, a un nivel básico el protocolo IP es todo lo que se necesita para participar en la red. No será muy amigable, pero si usted es lo suficientemente capaz, puede serle útil. Siempre y cuando la información se ponga en un sobre IP, la red tiene toda la información necesaria para llevar el paquete hasta su destino. No obstante, habrá que resolver varios problemas:

- La mayoría de las transferencias de información es mayor que 1500 caracteres. Usted estaría decepcionado si la Oficina Postal sólo entregara tarjetas postales y rechazara cualquier cosa más grande.
- En ocasiones se presentan errores: ocasionalmente la Oficina Postal puede perder una carta; algunas veces la redes pierden paquetes o pueden dañarse durante la transmisión. A diferencia de lo que sucede con la Oficina Postal, Internet puede resolver estos problemas exitosamente.

- Los paquetes pueden llegar en desorden. Si se envían dos cartas al mismo lugar en días consecutivos no existe la garantía de que viajarán por la misma ruta o llegarán en el mismo orden. Lo mismo sucede con Internet.

Para evitar esto, la siguiente capa de la red nos permitirá enviar grandes cantidades de información y corregir todas las alteraciones que puedan ser causadas por la red.

Protocolo de Control de Transmisión TCP.

El TCP es el protocolo que se menciona frecuentemente junto con el IP y que se utiliza para resolver los problemas mencionados. ¿Qué pasaría si usted deseara enviar un libro a alguien y el Servicio Postal sólo aceptara cartas? ¿Cómo podría solucionarse este problema? Una de las soluciones sería arrancar todas las hojas del libro, ponerlas en un sobre cada una y depositarlas en un buzón. La persona que reciba las cartas tendrá que asegurarse de recibirlas todas y volverlas a empastar en el orden original. Esto es lo que hace el protocolo TCP, es decir, toma la información que desea enviar y la divide en segmentos. Además, enumera cada segmento para que el receptor pueda verificar la información y ponerla en el orden adecuado.

Para que el protocolo pueda enviar esta secuencia de números a través de la red, cuenta con su propio sobre que le permite "escribir" en él la información requerida para su reordenamiento. Ver figura 1.2.2.3

Un segmento de la información a transmitir se coloca en el sobre del protocolo TCP. Este sobre es puesto, a su vez, dentro del sobre del protocolo IP y posteriormente es transmitido a la red. Una vez que se pone algo en un sobre IP, la red lo puede transmitir.

Del lado del destinatario, una parte del software del TCP reúne los sobres, extrae la información de ellos y la pone en el orden adecuado. Si algún sobre se pierde en la

transmisión, el receptor solicita su retransmisión al emisor. Una vez que el protocolo TCP tiene toda la información en el orden adecuado, la pasa a la aplicación del programa que esté utilizando sus servicios.

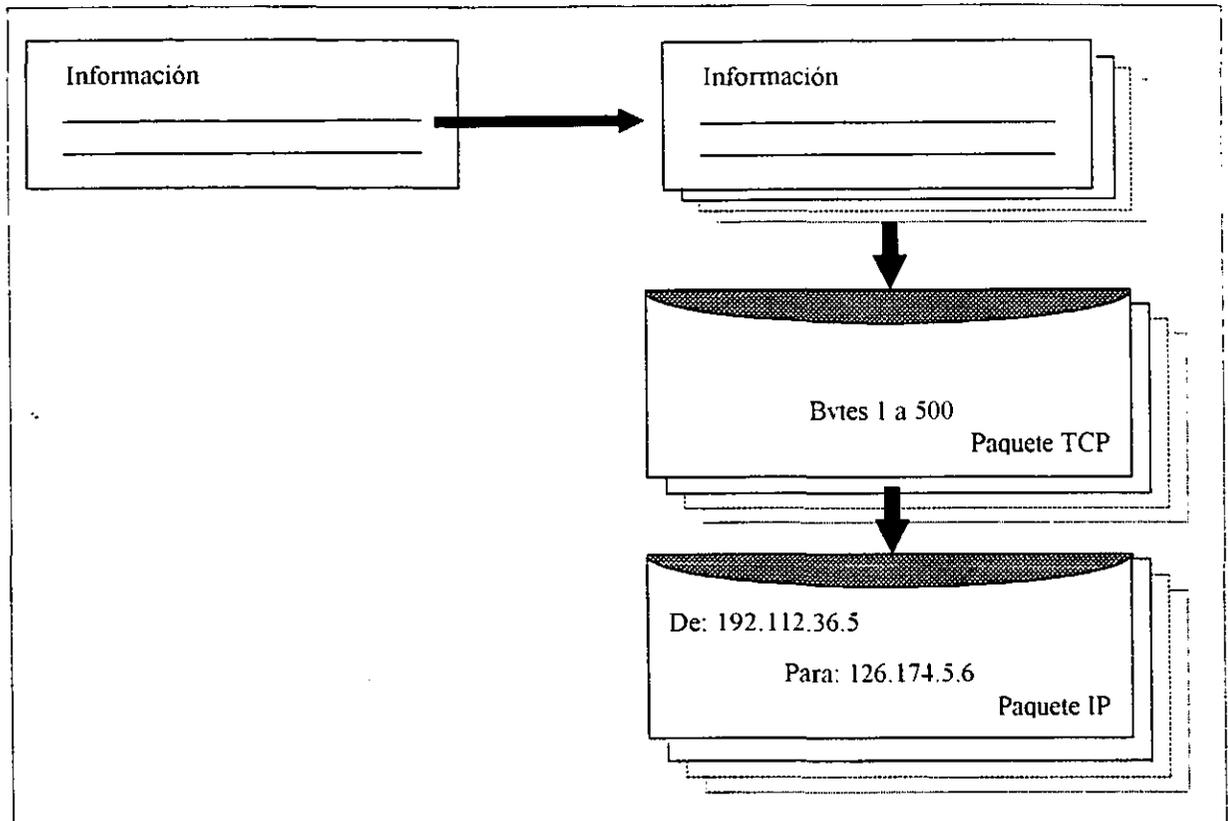


Figura 1.2.2.3 Encapsulamiento del Paquete TCP.

La descripción anterior del funcionamiento del protocolo TCP es ligeramente utópica. En la realidad, los paquetes no sólo se pierden, además de esto pueden ser modificados por el mal funcionamiento durante la transmisión a través de las líneas telefónicas. El TCP también resuelve este tipo de problemas. Así como coloca la información en un sobre, el protocolo calcula algo llamado *número de verificación*. El número de verificación es un número que permite que el receptor TCP detecte errores en el paquete transmitido. Cuando un paquete llega a su destino, el receptor calcula el número de verificación y lo compara con el enviado por el transmisor. Si no coinciden,

significa que ocurrió un error en la transmisión. El receptor deshecha el paquete y solicita la retransmisión.

Sistema de Nomenclatura de Dominios.

Al principio, la gente acepta que las combinaciones de números como domicilios estaban bien para que las máquinas se comunicarán entre sí, pero las personas prefieren utilizar nombres. Es difícil hablar usando sólo números y es aún más difícil recordarlas. Por esto, a las computadoras de Internet se les asignaron nombres para la conveniencia de los usuarios. Todas las aplicaciones de Internet permiten el uso de nombres en lugar de números para definir los domicilios de las computadoras.

Es un hecho que los nombres tienen problemas implícitos. Por una sola razón, usted tiene que asegurarse de que nunca dos computadoras de Internet se llamen igual. Es necesario tener una forma de convertir los nombres a combinaciones numéricas. Después de todo, los nombres están bien para toda la gente, pero las computadoras prefieren números. Es posible dar un nombre de computadora a un programa, pero es necesario que el programa tenga una forma de convertir el nombre en una combinación de números.

En un principio, cuando Internet era un pequeño lugar, el manejo de los nombres era sencillo. El NIC, Centro de Información de la Red, estableció un registro. Una persona enviaba una forma, electrónicamente, y el NIC incorporaba la información en una lista de nombres y domicilios. Este archivo, llamado hosts, era distribuido en forma periódica a todos los nodos de la red. Los nombres eran simples palabras, no se podían repetir. Si se usaba un nombre, la computadora buscaba en la lista y lo convertía a un domicilio numérico. Funcionaba bien.

Cuando Internet creció y se multiplicó, lo mismo sucedió con la tabla de nombres. Se requería mucho tiempo para que un nombre quedara registrado y se volvió más difícil

encontrar nombres que no hubiesen sido usados. También se requería de mucho tiempo de red para poder distribuir el archivo de la lista a cada máquina de la red. Era obvio que un sistema distribuido en línea era requerido para satisfacer la rapidez con la que cambiaba la información del archivo de nombres. A ese sistema se le denomina Sistema de Nomenclatura de Dominios o DNS.

Estructura del Sistema de Dominios.

El Sistema de Nomenclatura de Dominios es un método para administrar nombres distribuyendo en diferentes grupos la responsabilidad de subconjuntos de nombres. A cada nivel de este sistema se le llama *dominio*. Los dominios se separan por puntos:

ux.cso.uiuc.edu

nic.ddn.mil

yoyodyne.com

Puede haber cualquier cantidad de dominios en un nombre, pero en muy raras ocasiones se verán nombres con más de cinco. Al leer un nombre de izquierda a derecha, cada dominio será más vasto que el dominio que tenga a la izquierda.

En el nombre (ux.cso.uiuc.edu), ux es el nombre del equipo anfitrión, una computadora con un domicilio IP. Ver figura 1.2.2.4. El nombre para esa computadora se asigna y mantiene por el grupo cso, que es el departamento donde ésta se localiza. El departamento cso es parte de la **University of Illinois at Urbana Champaign** (uiuc). uiuc, a su vez, es parte de un grupo nacional de instituciones educativas (edu).

De esta forma, el dominio edu está compuesto por todas las computadoras de las instituciones educativas, el dominio uiuc.edu contiene a todas las computadoras de la University of Illinois y así sucesivamente.

Cada grupo puede crear o cambiar todo lo que esté dentro de él. Si uiuc decide crear otro grupo que se llame ncsa, lo puede hacer sin solicitar ningún permiso. Sólo tiene que agregar el nuevo nombre a su parte de la base de datos mundial y tarde o temprano todo aquel que lo necesite descubrirá el nuevo nombre (ncsa.uiuc.edu). Si cada grupo a partir de edu respeta las reglas y se asegura de que los nombres que asigne sean únicos, ningún nombre en Internet se repetirá.

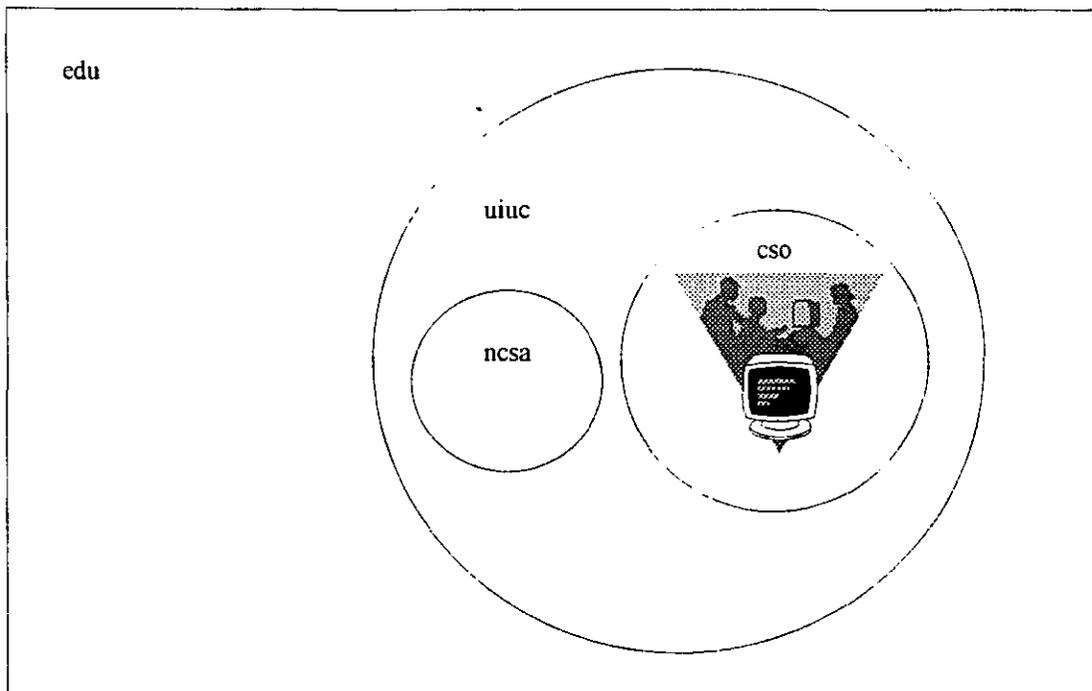


Figura 1.2.2.4 Autoridad Sobre Dominios.

La institución encargada de asignar y controlar el direccionamiento de las grandes redes de Internet es el *Centro de Información de la Red, NIC*, a través de la *Autoridad para la asignación de Números de Internet, IANA*. Estas a su vez bautizan a los miembros o nodos de su propia red. La responsabilidad y el cuidado diario al asignar y actualizar direcciones de dominio recae en el grupo del *Padrón Internet, IR*.

A partir del crecimiento desmesurado de la super-red, InterNIC tuvo que designar a diversas instituciones en todo el mundo para que colaborasen con ella en la ardua

tarea de asignar nombres y dominios a los nodos que día con día se integran a Internet. Por ejemplo, en México el organismo que se encarga de estas actividades se conoce como *NIC-México*, y está ubicado en el *Instituto Tecnológico y de Estudios Superiores de Monterrey, ITESM*, en la ciudad de Monterrey.

Búsquedas en el Sistema de Nomenclatura de Dominios.

Ahora que se sabe cómo se relacionan los dominios entre sí y cómo se crean los nombres, usted se preguntará cómo se usa este maravilloso sistema. Se emplea automáticamente, siempre que se utilice un nombre de domicilio en una computadora que esté configurada para utilizar el Sistema de Nomenclatura de Dominios.

Cuando se usa un nombre como `ux.cso.uiuc.edu`, la computadora necesita convertir el nombre del domicilio en un domicilio numérico. Para poder hacer esto, empieza a hacer peticiones de ayuda a los servidores DNS, empezando por el extremo derecho y recorriendo el domicilio hacia la izquierda. Primero, se pregunta por el domicilio al servidor DNS local. Hasta este punto, hay tres posibilidades:

- El servidor local conoce el domicilio debido a que éste se encuentra en la parte local de la base de datos mundial contenida en el servidor. Por ejemplo, si usted está usando una computadora en el departamento de Ciencias Computacionales de la Universidad de Illinois, es probable que el servidor local tenga información acerca de las computadoras del departamento.
- El servidor local conoce el dominio solicitado porque alguien más lo solicitó recientemente. Cuando se solicita un dominio, el servidor DNS lo guarda por un tiempo, sólo por si alguien necesita el dominio más tarde; esto hace al sistema mucho más eficiente.
- El servidor local no conoce el dominio, pero sabe cómo buscarlo.

¿Cómo busca el servidor local? El software del servidor local sabe cómo comunicarse con el servidor *raíz*. Éste es el servidor que conoce los domicilios de los servidores de nombres que tienen a su cargo los dominios de jerarquía superior (los dominios de la derecha, como por ejemplo, edu). Un servidor de nombres pregunta al servidor raíz el domicilio de la computadora responsable de la zona edu. Con información, se comunica con ese servidor y le pide el domicilio del servidor uiuc. Después, contacta a esa computadora y le solicita el domicilio del servidor cso. Finalmente, se comunica también con esa máquina y obtiene el domicilio de ux, que es la máquina con la que originalmente se quería comunicar.

1.2.3 Servicio Gráfico de Internet.

El servicio gráfico de Internet conocido como **World Wide Web** o W3, algo así como *la gran telaraña mundial de comunicaciones* es realmente una herramienta nueva comparada con la red de redes. Antes del Web, los investigadores de las universidades utilizaban las aplicaciones de Internet como el Correo electrónico en sus comunicaciones, **Telnet** para acceder a computadoras remotas, creaban directorios para el almacenamiento de archivos que se pudieran compartir y buscar mediante **Gopher**, **Archie** y **Veronica**, o **Finger** para localizar y conocer datos sobre un usuario conectado a la red; todo esto a través de la clásica pantalla negra de un terminal UNIX o mediante una PC que emulaba la terminal UNIX con un programa de comunicaciones en modo texto, usando el protocolo TCP/IP.

El problema se presentaba cuando los trabajos o investigaciones contenían archivos de gráficos, texto o vídeo y un colega deseaba consultarlos o trasladarlos hasta su computadora. Los desarrollos de las instituciones dedicadas a la computación como la Fundación Nacional para la Ciencia, el Centro Nacional de Aplicaciones de Supercómputo, la Agencia de Proyectos de Desarrollo Avanzado y otras, se enfocaron en la solución del problema que presentaba el transporte de datos complejos a través de la red.

El World Wide Web es un sistema distribuidor de información basado en el hipertexto e hipermedia. Fue desarrollado en 1990 por un grupo de investigadores bajo la dirección de Tim Berners-Lee en el Laboratorio Europeo de Física en Partículas, CERN, ubicado en Suiza. Ellos definieron los conceptos **HyperText Transport Protocol**, **HTTP** (Protocolo de transporte de HiperTexto), **HyperText Markup Language**, **HTML** (Lenguaje para el diseño y creación de las páginas de Web) y **Uniform Resource Locator**, **URL** (Localizador Uniforme de Recursos), que son la trilogía Base para construir, localizar y acceder a las páginas de Web en cualquier nodo o red conectados a Internet.

En octubre de 1990 se define el nombre del nuevo servicio de información compartida, quedando como se conoce ahora, World Wide Web. Pronto comienzan los primeros *hojeadores/editor* que permiten crear ligas de hipertexto en línea. La primer demostración del WWW se realiza en 1991 en San Antonio, Texas, en los Estados Unidos. En 1992 surge uno de los primeros *hojeadores* (conocidos como Browsers) del Web en el CERN: Viola, aunque todavía en modo texto, que fue la característica de los navegadores del CERN.

En el Centro Nacional de Aplicaciones de Supercómputo, NCSA, de la Universidad de Illinois, Marc Andreessen, junto con el grupo de desarrollo de la universidad, trabajó a principios de 1993 en un proyecto cuyo propósito era leer las páginas del Web que estaban en formato HTML, pero no en modo texto, sino en forma gráfica, utilizando las capacidades de hipertexto e hipermedia. El producto de estas investigaciones fue el navegador del Web más conocido hasta ahora: Mosaic, simiente de los nuevos *hojeadores* como Netscape, Explorer y muchos más.

En 1993 ya existían 200 servidores del World Wide Web utilizando el protocolo HTTP. En enero de 1994 la compañía Spry anuncia el paquete de navegación en el Web denominado **Internet in a box**, que proporcionaría a cualquier usuario de computadoras caseras, un paquete de herramientas para *surfear en la red*. En marzo

del mismo año, Marc Andreessen y varios colegas forman la compañía Mosaic Communications Corp, que en poco tiempo después se convertiría en la multimillonaria actual, Netscape.

Sistema de Localización de Sitios Web.

Al utilizar el protocolo http, hace uso del servicio World Wide Web de Internet. La localización de los sitios o páginas se realiza mediante un esquema denominado Localizador Uniforme de Recursos, URL, que comienza siempre con las siglas del protocolo que se va a emplear para la transferencia de información seguidas por dos puntos (:) y dos diagonales (//), en este caso: http://, que significa Protocolo de Transferencia de Hipertexto. Si se accesa a un servidor de Gopher, el URL indicará al inicio Gopher://, y así para cada protocolo de Internet.

La segunda parte de un URL es la dirección del servidor del recurso. Por ejemplo, la dirección o URL del servidor del World Wide de Microsoft es http://www.microsoft.com, todo esto, utilizando letras minúsculas. Por último, el o los directorios del disco duro donde se encuentran los documentos o páginas se separan de la dirección con otra diagonal (/) cada uno. Al final, el nombre del archivo al que se desea acceder, escrito tal y como se ha diseñado; es decir, mayúsculas, minúsculas o ambas.

Hipertexto e Hipermedia. El concepto de hipertexto no es algo nuevo o que haya nacido junto con el paradigma del World Wide Web. Desde 1945, el Dr. Vannevar Bush, entonces Director de la Oficina de Investigación y Desarrollo Científico de los Estados Unidos, propuso un sistema que rebasaba las limitantes de la lectura del texto normal; es decir, la posibilidad de *lectura no lineal*, basada en la manera en la que pensamos, que difiere de la forma lineal de leer que acostumbramos.

Es hasta 1965 cuando Ted Nelson, acuña los términos de *hipertexto* para describir el nuevo modo de *lectura no lineal* que asocia ligas o vínculos con textos alternos que

pueden estar en otro lugar del mismo documento o en otros documentos independientes, e *hipermedia* para el mismo concepto, sólo que los vínculos permiten ligar imagen, sonido, vídeo y otros medios al documento original. Por lo tanto, una definición para *hipertexto*, puede ser: el texto contenido en un documento provisto de *ligas* o *enlaces* que permiten leerlo en forma no lineal.

En 1990 Tim Berners-Lee del Laboratorio Europeo de Física de Partículas (CERN) en Ginebra, Suiza, junto con un grupo de investigadores, crea el primer sistema *hipermedia* de distribución global de información en Internet, que se denominó como World Wide Web.

1.2.4 Herramientas no Pertenecientes a la Red en Internet.

La World Wide Web ha captado la tensión del público y de los profesionistas del Marketing gracias a la difusión alcanzada a través de los medios de comunicación. Aunque esto es capaz de entusiasmar a cualquiera, las viejas herramientas de Internet demostraron ser muy valiosas para el marketing. Se les tiene que comprender y utilizar en conjunción con el espacio ocupado por una página de Web para que esta tenga éxito. Desde una perspectiva de marketing, hay dos clasificaciones de los mecanismos de Internet: herramientas de comunicación y máquinas vendedoras de información. Las herramientas de comunicación se utilizan para transmitir mensajes específicos a personas concretas. Construyendo un medio para la discusión electrónica. Las tres herramientas principales son el correo electrónico, los grupos de noticias y las listas.

Correo Electrónico.

El correo electrónico permite al usuario de una computadora teclear un mensaje y enviarlo a otra computadora. El mensaje llega en cuestión de segundos, el receptor puede editarlo y enviarlo de regreso. Un mensaje de correo electrónico se puede

enviar a múltiples individuos. Se puede almacenar electrónicamente o imprimirse. Exige el arte de las comunicaciones escritas, que acelera a la velocidad de la luz.

Grupo de Noticias.

En muchas tiendas locales de productos frescos se suelen ver grandes tableros anunciadores cerca de la puerta, confeccionados con planchas de corcho. Los tableros suelen estar divididos por tiras verticales con encabezamientos como "Canguros", "Coches", "Muebles", etcétera. Se trata de la versión comunal de la sección de anuncios clasificados de un periódico. Los clientes locales colocan sus mensajes en el tablero, para que los vea todo el mundo.

En Internet se añaden unos pocos matices más a este método de comunicación acreditado por el tiempo. En primer lugar, el tablero ahora tiene doce Kilómetros de longitud y está dividido en mil secciones por cada Kilómetro. En segundo lugar, la gente puede leerlos desde la comodidad de sus propias computadoras. Finalmente, la cultura del tablero de anuncios se hace interactiva.

En lugar de tratarse de simples anuncios, los mensajes incluyen preguntas, buscan consejos y plantean peticiones de asistencia. En lugar de limitarse a copiar una dirección o arrancar un trozo de papel donde hay anotado un número de teléfono, la gente contesta a los mensajes que ve cuando está colocando su propio mensaje.

Debido a las numerosas y diferentes categorías de grupos de noticias, podemos estar seguros de encontrar varios grupos de interés personal. Y, lo más importante, podemos estar seguros de encontrar grupos de interés profesional. Las listas sobre carreteras o excursiones de bicicleta tendrán interés para los fabricantes de artículos deportivos. Un grupo de noticias sobre un tema relacionado con la salud tendrá interés para empresas fabricantes de compuestos vitamínicos.

Listas.

Se trata de la combinación del correo electrónico y de los grupos de noticias. Cuando un mensaje se coloca en el tablero electrónico de noticias, tiene que ser "visitado" por el lector. Cuando un mensaje se coloca en una lista, todos los que se suscriben a la lista reciben una copia en sus buzones privados. Como lector de un grupo de noticias, puede usted echar un vistazo de vez en cuando. Como suscriptor de una lista, tiene que repasar la lista para encontrar alguna comunicación personal enviada a usted.

Los grupos de noticias y las listas tienen por así decirlo dos sabores: moderados e inmoderados. Un moderador puede asumir un papel dictatorial, determinando qué mensajes son valiosos y colocando sólo esos mensajes. Un moderador puede asumir, sin embargo, una postura más cómoda, dirigiendo la discusión si se aleja del tema o se ve superada por una guerra encendida. Estos individuos proporcionan un inestimable ojo editorial.

Todos tenemos capacidad para recibir informaciones directas de los servicios de radio, pero no las recibimos. Existe, simplemente, demasiada información, y no toda ella es valiosa. Algunos desean poner más énfasis en los deportes y comprar revistas deportivas. Otros desean una visión más liberal, o más conservadora. Se suscriben a periódicos y revistas y escuchan los programas de discusión que son más de su gusto.

Lo mismo podemos decir de los grupos de noticias y las listas. Sin un moderador, estos grupos de discusión pueden progresar bastante bien, puesto que son autogobernados. Por otro lado, también se pueden deteriorar para transformarse en una tertulia sin ningún valor, en la que intervienen gran cantidad de personas que disfrutan mucho expresando su opinión. En consecuencia, algunas listas son muy buenas y otras progresan y retroceden, entre el entusiasmo de la ilustración y la angustia del desagradable sonido.

El correo electrónico, los grupos de noticias y las listas se complementan con una cultura de uso conocida como "etiqueta de red".

Máquinas Vendedoras de Información.

Gopher, ftp (protocolo de transferencia de archivo) y las notas de correo, se encuentran en la misma categoría. Todas estas herramientas realizan la misma función básica: cuando alguien de la red plantea una petición de un archivo, estas herramientas se lo buscan y se lo ofrecen. Son como los programas de apoyo de fax de Internet. Pueden estar abiertos a todos, o protegidos por una contraseña. Poseen tanta o tan poca información como se quiera. Constituyen el trabajo de base sobre el que se formó la World Wide Web.

1.2.5 El Crecimiento de Internet.

Antes de dedicar algún tiempo al marketing en la red, desde luego, antes de gastar dinero alguno, el profesionalista del marketing desea saber si esta cuestión de Internet continuará manteniéndose al año que viene.

La respuesta es un "sí" rotundo. Es mejor estar conectado que no estarlo. El valor de Internet es claro y actual. Los departamentos técnicos responsables de mantener las conexiones de Internet desde el principio, se han encontrado con un alto valor administrativo en las yemas de sus dedos. Planificar reuniones, distribuir memorándums y procurar que la política sobre información se encuentre en conexión directa, no fueron más que algunas de las formas que descubrieron estos innovadores para agilizar y facilitar sus negocios. Cuando otros se dieron cuenta de en qué andaban metidos, también quisieron participar de los beneficios. Internet empezó entonces a desarrollarse a mayor velocidad.

Este nuevo medio de comunicación está cambiando la forma de hacer negocios, del mismo modo que lo hicieron antes el teléfono y el fax. Pasar del papel de carbón a la fotocopiadora impulsó tremendamente nuestra capacidad para diseminar información con mayor rapidez. Pasar de la máquina de escribir al procesador de textos eliminó muchos trabajos de secretaria. Tendrán que transcurrir varios años antes de que conozcamos los efectos de pasar a utilizar a Internet. Quizá para entonces tengamos acceso a Internet como tenemos ahora de teléfono y de televisión. Quizá todos los instrumentos destinados a pasar páginas por la pantalla tendrán teclados. Quizá necesite tener un acceso a Internet para presentar su declaración de renta. Sean cuales fueren los cambios a largo plazo, el valor inmediato para los negocios es algo evidente: una mejora extraordinaria de las comunicaciones.

1.2.6 Servicios de Conexión Directa.

Tome el protocolo Internet, el correo electrónico, el tablero-boletín de los grupos de discusión, el intercambio libre de información y embotéllelo. Establezca una versión privada y cobre el ingreso a la misma. Añádale algo de software avanzado para que sea utilizable por parte de quienes sólo conocen sus computadoras de vista. Ofrezca servicios adicionales basados en unos horarios, y cobre tranquilamente diez dólares al mes de la tarjeta de crédito de todos los participantes. En la actualidad hay diez millones de personas en compuServer, America Online, Prodigy y Genie.

Otro ámbito con el que esta clase de servicios directos ganan dinero es la publicidad. El espacio de logotipo se vende caracterizado como modelo de la revista. Cuando más importante es el logotipo y el público, tanto mayores serán los honorarios. Además, le cobran dinero por situarlo en un fórum electrónico y público. Le cobran dinero por situarlo en su tienda electrónica. Y también le cobran un porcentaje de todo aquello que venda a través de su tienda, lo que los convierte en una forma de ganar dinero permanentemente.

Entonces, a principios de la década de 1990 ocurrió algo bastante interesante. Los subcontratistas activos en la estructura Internet de la Fundación Nacional de Ciencia se dieron cuenta de que se podía ganar dinero ofreciendo esos servicios a las grandes empresas. La NFS comprendió que las transacciones comerciales a cambio de unos honorarios superarían la pura utilidad de la investigación. Si los contratistas deseaban entrar en el negocio, la NFS podía dejar de gastar cada año millones de dólares en infraestructura de telecomunicaciones. Como consecuencia de ello, relajaron sus reglas de la realización de negocios en Internet.

En los meses que siguieron floreció el tráfico comercial, Internet apareció en la portada de Time, Business World y Newsweek, y nació así toda una nueva industria de proveedores de acceso. Internet permitía a todo el mundo ser su propio editor. Los foros públicos estaban abiertos para quienes los pidieran. Las páginas Web estaban dispuestas para atraer la atención por sí mismas y no dependían del "tráfico de a pie" de los servicios directos.

Los Servicios Directos como Proveedores de Acceso.

En 1995, los servicios directos completarán su transición desde redes privadas a proveedores de acceso a Internet. Seguirán ofreciendo extras basados en el cobro de honorarios. Seguirán cobrando a sus clientes comerciales el espacio que ocupen en sus tiendas electrónicas. Pero abrirán las puertas de sus redes privadas al resto de Internet.

Empezaron por permitir a los usuarios que se relacionaran mediante correo electrónico a través de Internet, y terminarán por permitir el acceso gráfico a la World Wide Web. No tendrán control sobre el contenido de las páginas Web, y algunos de ellos entrarán en competencia directa con sus servicios a cambio de honorarios. No obstante, y a la luz del crecimiento de Internet hacia su adolescencia, a la luz del Eworld de Apple, del

networkMCI y de The Microsoft Network, ese es el único movimiento que pueden efectuar los servicios directos para seguir manteniendo sus índices de crecimiento.

Al margen del aspecto que tenga el mecanismo de acceso del usuario, no cabe la menor duda de que Internet es muy útil para los negocios, que dependen de la comunicación, algo en lo que Internet es muy bueno. Como consecuencia de ello, está cambiando nuestra forma de hacer los negocios, tal como hicieron en su día el teléfono y el fax. Mejora la investigación, acelera el servicio al cliente, ofrece una ventaja competitiva y reduce el tiempo para acceder al mercado.

Servicios Directos como Emisores de Publicidad.

A medida que los grandes servicios directos obtengan el control del mercado de acceso, continuarán luchando por apoderarse del mercado de la publicidad. Los lectores de Newsweek y Time leen las mismas noticias difundidas por todo el mundo. Difieren el estilo y el giro editorial, pero los temas abordados son relativamente similares. El precio que se paga por los anuncios se basa en el índice de lectores. ¿Cuántas personas verán probablemente su anuncio? ¿Cuántas impresiones causará? Los servicios directos venden espacio de logotipo basado en el tráfico electrónico de paso. Compiten entre sí basándose en el número de posibles impresiones causadas en sus suscriptores. Es un clásico modelo editorial.

Si lo que le interesa es el reconocimiento de su marca, esta forma de publicidad puede tener sentido. Cuanto mayor sea el número de clientes de America Online que puedan conectar cada día, tanto mayor será el valor que tendrá usted.

No obstante, cuando el proveedor de acceso permite a sus clientes acceder a la World Wide Web, la imagen cambia ligeramente. La revista Time firmó un acuerdo exclusivo con America Online. Llegaron al acuerdo de no situar una presencia electrónica en ningún otro servicio directo. Más tarde, Time creó su propia página de Web. Más tarde

aún, Prodigy anunció acceso a la red para todos sus clientes. Ahora, los usuarios de Prodigy pueden acceder a la página Web de Time. No obstante, todavía se mantiene el acuerdo.

Los servicios directos pierden su capacidad para controlar el contenido cuando se convierten en proveedores de acceso. Imagine una pista de patinaje que cobrara por horas por el uso de los patines y por horas el uso de la pista. De pronto, la presión del mercado obliga a los directores de la pista a permitir que sus clientes patinen en la acera situada al lado de la pista. Con ello han perdido los ingresos derivados de patinar en la pista.

Entre sus alternativas se incluye la de colocar el grande y ancho mundo fuera del alcance de sus clientes. Sería la receta más segura para morir en manos de la competencia. Los patinadores no utilizarán su pista, y favorecería el uso de aquella otra pista de patinaje que permitiera el acceso al resto del mundo. Su segunda alternativa consistiría en ofrecer los mejores patines, en grandes cantidades y en todos los tamaños. Como opción final podrían conseguir que el patinaje en pista cubierta fuera tan deseable como para conservar al mismo tiempo la lealtad de sus clientes y los ingresos por el alquiler de la pista.

Ninguno de los servicios de acceso público directo se ha posicionado de modo que pueda ofrecer el único contenido que usted llegue a necesitar. Ya han entrado en competencia con proveedores de acceso a Internet a niveles regional y nacional para el acceso de marcación directa a Internet. CompuServer ya ha reducido sus honorarios de acceso a la mitad en los últimos doce meses, como reconocimiento de que los proveedores de acceso a Internet no necesitan ofrecer servicios de valor añadido para atraer a clientes.

Lo que les falta a estos servicios es lograr que sus ofertas de valor añadido sean lo más atractivas posible. America Online hizo que su pista cubierta fuera de lo más

deseable al llegar al acuerdo con la revista Time. Ahora puede usted visitar Time en Internet sin la ayuda de AOL. La carrera entre las compañías será muy reñida. El éxito se medirá sobre una base mensual. El ganador en cualquier momento dado será aquel que ofrezca los servicios más interesantes, informativos, divertidos y estimulantes... que todavía no se encuentren disponibles en el Internet general. El tiempo lo dirá y el tiempo traerá consigo The Microsoft Network.

1.2.7 Seguridad.

Aunque este tema parece estar exclusivamente en manos del departamento de sistemas informáticos, es importante saber dónde están las minas que pueden encontrarse en su camino. La primera de ellas, y la que inspira un mayor temor, es la seguridad de los sistemas de la propia empresa.

Protección de la Red Interna.

De vez en cuando, los periódicos informan a cerca de un cliente de guante blanco que actúa por computadora, o de cualquier otro que molesta las bases de datos de una empresa y crea un verdadero caos en la red. La verdad es que en toda reunión suficientemente grande siempre encontrará a gente mala, con muy malas intenciones, empeñados en causar daños maliciosos.

También es cierto que la proporción entre riesgo y beneficio es muy pequeña. Decenas de personas mueren cada año en las autopistas, a pesar de lo cual los beneficiarios superarán tanto el riesgo, que seguimos conduciendo por ellas. Los aviones se caen del cielo, los edificios públicos se utilizan como objetivos terroristas, y hogares de la gente son asaltados y robados con aterradora frecuencia. Sin embargo, seguimos volando en aviones, nos congregamos en lugares públicos y vivimos en nuestros cómodos hogares con una protección mínima.

¿Estamos locos? No. Simplemente, sabemos cuándo ha habido algo que se ha desproporcionado. ¿Significa eso que debemos ignorar las advertencias? No. Solo significa que tenemos que reconocerlas lo más pronto posible y afrontarlas según nuestra mejor capacidad. Es prudente protegernos, pero no lo es vivir en una cueva. ¿Qué decirle entonces al presidente ejecutivo que desea proteger la red interna de la empresa? Se le pueden ofrecer alternativas.

El blindaje. Los habilidosos departamentos de gestión de la información han estado utilizando blindajes o muros antiincendios desde hace muchos años. Un blindaje es una combinación de hardware y software que sólo permite el paso de entrada y salida de la clase de tráfico correcto por la red informática de una empresa.

Algunos blindajes se crean internamente. Puesto que Internet ha disfrutado de tanta atención y ha experimentado tanto crecimiento en los últimos dos años, ahora ya disponemos de blindajes comercialmente apoyados.

Quizá escuche decir al presidente ejecutivo: "Lo haríamos si pudiéramos, pero en estos momentos no disponemos de los recursos para hacer un blindaje suficiente". En este caso es posible recomendar el blindaje de aire.

El blindaje de aire. Lo mismo que el blindaje antiincendios, el blindaje de aire protege la red interna de los delincuentes externos. Pero, a diferencia del primero, un blindaje de aire protege la red interna de un modo incontrovertible, al cien por cien.

El término "blindaje de aire" se refiere a la conexión establecida entre su servidor de Web y el resto de las computadoras de su empresa. Se hallan conectadas por el aire. No hay cables, ni hilos, modems, rutas o enlaces infrarrojos. No hay más que dos latas de estaño y un rollo de cuerda. El servidor está al otro lado del blindaje y atiende al público general. Usted se conecta a él y obtiene todos los datos que han dejado sus

usuarios. Al no existir conexión alguna con el interior, no hay de que un bribón que actúe en Internet tenga acceso a las joyas de la familia.

Quizá le oiga decir al presidente ejecutivo: "Lo haríamos si pudiéramos, pero no disponemos en estos momentos de los recursos para tener un servidor externo".

El servidor externo. Se trata de una joven industria y de unos servicios nuevos que se ofrecen de una forma cotidiana. Yahoo muestra unas 650 listas de Internet **Presence Providers** (Proveedores de Presencia). Algunas de estas empresas le ofrecen ayuda para levantarse y echar a andar, y otras le ofrecen el pleno desarrollo de un servidor y el mantenimiento. El trabajo de usted consistirá en seleccionar a la mejor para realizar el trabajo, y eso no le será fácil. Algunos proveedores de presencia cobran 50 dólares por hora. Otros cobran 50,000 dólares sólo por decirle hola. La diferencia entre los dos puede no tener nada que ver con la calidad. La industria es, sencillamente, demasiado nueva para haber establecido un mercado estable para estos servicios.

Además, el comprar esos servicios será algo que dependerá exclusivamente de lo que usted quiera conseguir. ¿Cuál puede ser su mejor método para seleccionar un servicio externo? Dedíquese a navegar y a comprobar referencias. Salga al Web y encuentre las páginas de Web que más le gusten. Si fueron creadas por la misma empresa, habrá la firma de un director de Web donde podrá hacer clic para enviar mensajes. Su mensaje puede ser: "Me gusta su estilo. ¿Le gustaría conseguir un aumento del diez por ciento en su salario actual?".

Si la página de Web fue creada por un servicio externo, a menudo lo indican así como un enlace situado al final de la página. Hacer clic en su logotipo también le permitirá enviar un mensaje o le pondrá en contacto directo con su propia página de Web, donde tienen el listado de sus otros clientes. Si le gusta su estilo, llame a sus clientes y hágales preguntas, las respuestas a esas preguntas resultan ser siempre las más informativas.

Identificación.

Un tema relacionado con la seguridad es el de la identificación. Si recibe un e-mail o un pedido de compra del presidente de los Estados Unidos, ¿cómo sabe que procede realmente de la Casa Blanca? Si se siente satisfecho y suficientemente seguro de saber que procede de ahí, ¿cómo sabe que le ha llegado de Bill o de Hillary?

La gente que introduce datos en una página de Web puede ser incluso más difícil de localizar. Pueden introducir las respuestas que quieran... si no se identifican. La única solución consiste en solicitarles una contraseña. Pero, si deseamos atraer la atención de la gente, no se recomienda preguntarle quién es antes de permitirles echar un vistazo a la página. Esa capacidad para realizar un recorrido antes de revelar la identidad atiende a un par de cualidades propias de la naturaleza humana.

La memoria es el segundo rasgo humano que actuará en contra del enfoque de "regístrese antes de echar un vistazo". Los enemigos aquí son la buena y la mala memoria. La gente recuerda que usted lo incordió antes de ofrecerles algo a la venta. Si entrara a una tienda y alguien le preguntara inmediatamente su nombre, dirección y número de teléfono, se sentiría acosado y lo recordaría como una experiencia desagradable y muy probablemente jamás volvería.

Una memoria deficiente también actuará en contra del método de avanzar paso a paso antes de darse a conocer. La gente tiende a olvidar las cosas. Cuando una computadora le pide que se identifique, es necesario ofrecerle una contraseña para comprobar que se encuentra de hecho en el lugar correcto. Teniendo en cuenta las numerosas páginas Web a las que uno puede asociarse, recordar todas esas contraseñas no es, sencillamente, algo práctico.

La solución que se les ocurrió fue el pedirle a la gente que creara sus propias parejas de preguntas y respuestas. En Open Market, "cuando se compra algo por una cantidad

mayor que el límite de su cuenta (previamente seleccionado), se le planteará un desafío, ya sea relativo al perfil de su cuenta o a un desafío que usted mismo haya establecido. Se selecciona esta opción para crear un desafío corriente (como por ejemplo, saber cuál es el nombre de su perro para comprobar que es realmente usted quien hace la transacción)". Éste tipo de desafíos se refieren a cosas que usted no anotó en ninguna parte y que probablemente no olvidará nunca.

Verificación.

Supongamos que el problema anterior se ha solucionado y está usted suficientemente satisfecho y seguro de que la persona quien envía la solicitud es realmente el presidente de los Estados Unidos. Hace usted que se registre con una contraseña y el nombre de su primera novia. Supongamos que él ha hecho un pedido de 20 hamburguesas. ¿Está seguro? Si un bit desapareciera (ya fuera intencionadamente o por casualidad) mientras el mensaje se abre paso a través de cientos de computadoras hasta llegar a usted, se sentiría en una situación muy embarazosa si se presentara en la Casa Blanca con 20 grandes hamburguesas para 200 individuos. Por otro lado, podría tener que comérselas todas si resultara que Bill sólo quería dos.

En las redes privadas es posible y accesible asegurar las transmisiones. Eso es algo que se tendrá que añadir a Internet a medida que pase el tiempo. Recuerde que sólo desde hace un par de años ha empezado el comercio a utilizar Internet de una manera seria. Se necesitaría un poco más de tiempo para solucionar todos estos problemas. La mayoría de la gente confía en haber recibido el mensaje intacto y en que Bill deseara realmente 20 hamburguesas.

La confirmación de ida y vuelta. Con el tiempo tendremos firmas digitales y podremos efectuar transacciones seguras y garantizadas. Mientras tanto, el sistema de confirmación de ida y vuelta parece el enfoque más prudente. El método de ida y vuelta es muy sencillo. Los clientes hacen un pedido en su página de Web. Su sistema

les envía un e-mail detallando lo que usted cree que se le ha pedido. Después, espera a su confirmación antes de enviar la mercancía y la factura.

Llegan las firmas digitales. En el estado de Utha ya se ha aprobado una ley por la que se aceptan las firmas digitales como vinculación legal. California y Washington no le andan a la zaga. Una firma digital utiliza una encriptación clave pública. Usted registra su clave pública ante una autoridad independiente que la certifica. Las personas a las que usted envía los mensajes encriptados, pueden descifrarlos mediante una clave privada. La autoridad que ha certificado la firma es la encargada de autenticar su identidad. Dentro de poco se producirá una avalancha de servicios destinados a proporcionar autenticación, que serán utilizados sobre todo por los bancos, los empresarios y las oficinas de correos.

Aceptar Dinero a través de Internet.

¿De qué sirve todo este sistema si no podemos confiar en él lo suficiente como para efectuar verdaderas transacciones comerciales en directo? No se le puede utilizar como un medio de mover el dinero de aquí para allá, o para enviar o recibir pedidos; por el momento no es más que una especie de fax de alta calidad, que todavía no merece toda nuestra confianza. Estupendo para enviar folletos, pero no para comerciar.

Los temores al robo de la tarjeta de crédito. Todas las discusiones relativas a la seguridad se refieren a enviar o aceptar dinero a través de Internet. El temor actual a que se detecten en la red y se utilicen los números de la tarjeta de crédito es algo sano y enfurecedor. Técnicamente, es posible hacerlo, aunque también es cierto que resulta muy difícil. Analizar los miles y miles de millones de bits de información que recorren la red con la esperanza de detectar esos tres o cuatro paquetes que contienen un número válido de tarjeta de crédito, una fecha de caducidad y el nombre de un propietario de la tarjeta, no es una tarea para pusilánime.

El tema preocupa y se le ha dedicado atención, existe una necesidad acuciante de encontrar una solución. Y como esa necesidad es acuciante, está surgiendo toda una pequeña industria dedicada a proporcionar algunas respuestas.

La primera solución (virtual). Una de las primeras respuestas que se han encontrado ha sido planteada por **First Virtual Holdings Incorporated** (<http://www.fc.com>). Para ello utilizaron el modelo de ida y vuelta, conjuntado con un enfoque del "fuera de banda" para el uso de la tarjeta de crédito. Su solución también es notable porque permite realizar microtransacciones.

Las microtransacciones son compras por las que el costo de la transacción sería normalmente superior al precio del artículo vendido. Eso se aplica inmediatamente a la gente que vende información por la red Internet. Un artículo publicado en una revista puede venderse por 25 centavos. A un precio de 35 centavos por transacción, más el dos por ciento para la empresa de la tarjeta de crédito, no es esta precisamente la mejor forma de ganarse la vida.

First Virtual sitúa en primer término las preocupaciones de la tarjeta de crédito. Se registra usted con ellos como un comprador potencial. Se rellena un formulario de la página de Web. En contestación, ellos le envían un e-mail con un identificador de cuenta y un número de aplicación. El primero actúa como una contraseña, el segundo se utiliza como una forma ingeniosa de transferir información de su tarjeta de crédito.

Usted ya les ha indicado su nombre en la página de Web. Se le dirige entonces para que llame a su número 800. El sistema automatizado situado al otro extremo de la línea telefónica ("fuera de banda" con respecto a Internet) le pide el número de su tarjeta de crédito y la fecha de caducidad. Marca usted esos datos utilizando los números del teléfono. Recibe como contestación un e-mail en el que se le felicita por su nueva capacidad de comprar, y se le invita también a registrarse para vender sus propios productos.

Cuando realiza usted una compra utilizando su contraseña, ya sea por e-mail o en una página de Web, el vendedor envía una notificación de la venta a First Virtual, que le envía a su vez un e-mail a usted, diciéndole: "Este vendedor afirma que usted ha efectuado tal transacción. ¿Es correcto?". Usted tiene entonces la opción de contestar: "Sí, deseo recibir ese producto"; o "No, he cambiado de opinión"; o si cree que alguien está utilizando fraudulentamente su contraseña, puede contestar: "Es un fraude". A lo largo de toda esta transacción, su número de tarjeta de crédito no ha cruzado en ningún momento por la red de Internet. First Virtual actúa como intermediario monetario, y una vez al mes, carga a su tarjeta de crédito todas las compras que ha efectuado, y deposita los ingresos de sus ventas a su cuenta bancaria mediante una transferencia electrónica de fondos.

El inicio de la Banca. En mayo de 1995, Cardinal Bankshares, Inc., de Lexington, Kentucky, fue la primera empresa en conseguir luz verde de la Oficina de Supervisión Económica para establecer el Security First Network Bank. Los clientes dispondrán de la misma clase de acceso que tienen a través de los sistemas de banca telefónica, incluido la comprobación de saldos, el pago de recibos, y la transferencia de dinero entre cuentas bancarias. Aparentemente, la Oficina de Supervisión Económica se mostró satisfecha con las medidas de protección electrónica aplicadas. Cardinal Bankshares reconoció, sin embargo, que tiene que superar un problema de percepción. De ahí su propio nombre: **Security First** (seguridad primero).

1.3 LA PUBLICIDAD EN INTERNET.

Hay varias formas de utilizar Internet en un ambiente de negocios. Por maravillosa que sea la perspectiva de la comunicación global, y por mucho entusiasmo que produzca esta tecnología desde la perspectiva de la magia electrónica, su valor para los negocios se mide estrictamente por el impacto que produce sobre las cifras finales de pérdidas y ganancias.

Ese impacto dependerá de los objetivos de la organización que planea hacer uso de Internet y de la **World Wide Web** (Red extensa mundial). Algunas empresas se beneficiarán más utilizando esta nueva herramienta de comunicación para las ventas directas. Otras observarán una mejora inmediata en el costo del servicio al cliente. Y otras verán la Web como una oportunidad de crear y vender nuevos servicios.

Los negocios encontrarán muchas utilidades en la comunicación electrónica a través de Internet, desde reducir el costo de enviar faxes y paquetes rápidos, hasta una gran mejora de las comunicaciones internas. Esa mejora de las comunicaciones internas conducirá a una atención más rápida a las condiciones del mercado. La mejora al servicio al cliente engendrará a su vez una mayor satisfacción de éste y los directores de producto dispondrán también de una mejor información transmitida por el propio cliente.

1.3.1 Captar al Público.

La World Wide Web ha cambiado la forma en que los individuos conectados buscan ahora la información. A medida que aumenta el número de personas y organizaciones conectadas, también aumentará esa velocidad de cambio.

La WWW ha contribuido mucho a promover el uso de Internet. Su capacidad para transmitir imágenes, video y sonido ha cautivado la imaginación del mundo. Eso ha

inducido a un creciente número de organizaciones a introducir sus páginas Web en la red llenas de información, imágenes y descripciones de productos.

Como consumidor, puede usted valorar el acceso a toda la información que desea, cuando lo desee y durante el tiempo que quiera. Como especialista en **Marketing** (Mercadotecnia), dispone de un nuevo medio para vender mientras usted duerme. Una máquina vendedora de información que funciona a control remoto que ofrece productos a velocidad de la luz, mientras que sus vendedores tienen que dedicar más tiempo a cerrar tratos.

El objetivo de captar al público consiste en que la persona que se conecta sea un participante activo en la acción. Es decir, se trata de captar su atención y mantenerla.

Por ejemplo, si usted está vendiendo **software** (programas de cómputo), tiene que permitir que los clientes experimenten por sí mismos. Si se está vendiendo **hardware** (equipo de cómputo), se tiene que permitir que prueben el equipo. Si se vende información, capte en los clientes su más plena atención con una muestra del trabajo. Si se ofrecen servicios, procure que participen los clientes en una conversación electrónica acerca de sus necesidades.

Tráfico en su Página Web.

Si se edita una revista, procure mejorar su circulación. Si produce un programa de televisión, trate de aumentar su índice de audiencia. Si es propietario de unos grandes almacenes procure aumentar el tráfico de quienes pasen por ahí. En todos estos casos, el propósito consiste en conducir a más gente hacia el publicitario o el propietario de la tienda. Eso le permitirá cobrar más por el espacio publicitario, el tiempo de emisión o el espacio de escaparate.

Los grandes almacenes y los editores de las revistas electrónicas tratan de conducir más tráfico hacia sus clientes, los anunciantes. Es una tarea desalentadora, que exige el pensamiento más innovador. Regalar algunas cosas es una buena idea para ese mercado y puede causar en los usuarios interés en su página Web. O bien, puede proporcionar constantemente algo de diversión, de información interesante o útil, así como actividades que induzcan a la gente a seguir volviendo a conectarse una y otra vez.

Marketing con Valor Añadido.

La economía de la red exige que un vendedor de mercancías y servicios ofrezca algo de valor al visitante. Un estudio sobre el tema, un índice utilizable o una lista de tiendas de su zona puede ser más que suficiente. Quizá también sea un atractivo suficiente el ofrecer un grupo de discusión sobre los acontecimientos actuales que se están produciendo en su industria, o una página de indicadores a otras páginas específicamente relacionadas con el tema, existentes en la red.

En alguna parte de su empresa hay alguna información que es valiosa para la gente que trata de comprar los productos que se desean vender. Ofrezca esa información al mundo.

1.3.2 Marketing en la World Wide Web.

Introducirse en el marketing de la World Wide Web (WWW) supone apartarse de la autopista de 500 canales de cable y pasar al apacible terreno de más de cinco millones de canales. No se trata de canales emitidos por las ondas. Por estos canales no se emiten antiguos programas ya emitidos, se trata más bien de mensajes que esperan pacientemente a que seamos nosotros los que interactuemos con ellos.

La WWW es un medio que atrae, no que empuja. No se puede utilizar la red para emitir mensajes hacia las masas, con la esperanza de que alguien los capte mientras hace girar el sintonizador. La red ofrece información a aquella gente que puede estar dispuesta a salir a su encuentro y extraerla.

Es posible que algunas personas acudan sólo para echar un vistazo. Cuanto mejor aspecto ofrezca su página Web, cuanto más fácil de navegar sea, y mayor diversión e información ofrezca, tanto mayor será la probabilidad de que la gente quiera regresar, e incluso llevar consigo a otros amigos.

La WWW permite a una organización crear una biblioteca de materiales a la que pueda tener acceso cualquiera que disponga de una conexión con Internet. La capacidad multimedia de la red significa que la biblioteca se encuentra casi más restringida por los límites de la imaginación que por la tecnología. En consecuencia, el primer paso a dar en el Marketing en la red consiste en establecer unos objetivos realistas.

Servicio al Cliente.

Uno de los mejores usos del nuevo medio de comunicación es el contacto con el cliente. El departamento de servicio al cliente debería ser siempre el primero en recibir y utilizar los nuevos métodos de intercambio de información. El teléfono, el fax, y el contestador automático se iniciaron como curiosidades, encontraron luego la forma de llegar a más y más empresas, y al cabo de poco tiempo, se convirtieron en exigencia ineludible para atender y mantener a los clientes. La WWW pertenece también a esta lista.

Los departamentos de ayuda y servicio al cliente ya han descubierto el valor de la base de datos. Con el casco telefónico colocado y las manos sobre el teclado, el nuevo empleado puede ofrecer la explicación y contestación más detallada posible a la

pregunta más complicada. El cuello de botella había sido hasta ahora la transmisión de la información. La WWW ofrece un método que es más rápido y más directo que el envío de un fax.

Las Claves para el Diseño Exitoso de la Página Web.

Existe un número infinito de formas de navegar por el espacio electrónico. Los límites pertenecen sólo a la imaginación. A continuación se mencionan algunos métodos utilizados actualmente.

Tamaño de la Página.

La mayoría de los usuarios ven la red desde una plataforma PC, con pantallas más pequeñas que las que se encuentran en la mayoría de las estaciones de trabajo UNIX. Por tal razón, es mejor basarse en pantallas pequeñas.

Las herramientas de navegación deberían estar organizadas de modo que se acomodaran estas pantallas pequeñas y permitieran una travesía rápida, sin tener modos de desplazamiento. La mayor parte del software nos ha acostumbrado a ver una página a la vez, nunca se espera tener que pasar la página horizontalmente.

Usar la Barra de Título como Detentador de la Página Web.

Un enfoque habitual permite utilizar la barra de título como detentador de página web, lo que permite al usuario moverse hacia abajo y arriba a partir de la jerarquía. Se pone en práctica al tener un icono sólido en la parte superior de la página de inicio e iconos más pequeños que designan capas. El usuario puede ver la profundidad del nivel actual echando un vistazo a la barra.

Utilización de un Índice.

La facilidad de que el usuario pueda elegir libremente el entorno de navegación es proporcionado por el índice de temas que contiene la página. El hecho de poder profundizar en el índice desde el nivel superior permite a los usuarios ver el conjunto de la página al mismo tiempo. Además, tampoco hay necesidad de revisar una interminable lista alfabética o de hacer "click" una y otra vez sobre las distintas alternativas para descender hacia las profundidades de la página Web.

Utilizar un Mapa.

Si la página Web fuera un lugar, sería un gesto agradable ofrecer un mapa electrónico, que permitiría a los usuarios navegar en el lugar que desearan.

Utilizar Herramientas de Búsqueda.

Las herramientas creadas para encontrar información se han hecho muy eficientes. Desgraciadamente, existe una relación entre la facilidad de uso y el valor de la información. En algunas páginas resulta bastante sencillo teclear una palabra de búsqueda, pero el tiempo de respuesta puede ser lento.

1.3.3 Navegación.

El trabajo de un diseñador de páginas Web consiste en ayudar a la gente a encontrar la forma de moverse en esa página. Si los usuarios tienen problemas para encontrar algo de interés o se pierden dentro de la página, se sentirán frustrados y perderán todo interés. Las fuertes pistas visuales les ayudarán a encontrar el acceso a la información deseada, incluida la información que desean ver.

A la hora de ofrecer claves de navegación hay varios factores a tener en cuenta. La gente responde a la pantalla de la computadora de una forma diferente a como lo hace sobre papel. Pero lo cierto es que hemos sido formados para crear documentos como si fueran a ser impresos.

Interactividad.

Antes no se decía nada acerca de lo que se había que fabricar respecto a la opinión de la demanda de los clientes. La televisión se convirtió en una forma ideal de moldear a la opinión pública y de servir como vehículo para adquisición de artículos. Conforme fue evolucionando, el objetivo trataba de imaginar formas nuevas y mejores de conseguir que la gente deseara los productos que se estaban fabricando. Crear demanda era prácticamente equivalente a la supervivencia.

La tarea del profesional del marketing es la de buscar en todo su entorno, para encontrar un producto que la gente quisiera, pero que no pudiera encontrar todavía, en las estanterías de las tiendas. Además, el público ya no era una audiencia de masas. Empezaba a ser segmentado mediante el marketing de base de datos. A partir de estas circunstancias se contempló que la interactividad con los usuarios eran de gran importancia para los empresarios, de tal forma que apoyándose con la WWW, como medio de interactuar con los clientes, se podrían lograr estupendos resultados.

Retroinformación.

La medida final del éxito será el voto de confianza que obtenga de su clientela. Cuando hace preguntas a la gente acerca de sus productos, su empresa y su página Web se pueden recabar respuestas que nos ayuden a superar el estado actual de las ventas de nuestros productos o servicios.

Usted sabrá que ha hecho un buen trabajo cuando esa información le llegue de una forma lo bastante consistente y continua como para que la proporción de ruido causado se convierta en parte integral de su ciclo de desarrollo del producto.

1.3.4 Medir el Éxito.

Ha determinado cuidadosamente sus objetivos, ha planificado su estrategia y ha reunido a la gente adecuada. Ha convencido a la alta dirección de proporcionarle los recursos y creado un proceso para conseguir buen acceso directo y a tiempo. Los textos son animados, los gráficos asombrosos, se han comprobado los enlaces y las herramientas de navegación se encuentran en su lugar para ayudar a la gente a pasar de una página a otra.

Ha anunciado su página Web con gran difusión, ha recibido atención en las publicaciones especializadas por su singular uso de la tecnología. Solo queda por resolver una cuestión: ¿Cómo sabe que todo eso ha valido la pena?

Saber cuántas veces se conectó la gente con páginas específicas dentro de su página Web puede ser muy informativo. De ese modo podrá percibir cuál es la información realmente importante para la gente. Podrá ver cuáles son los productos más populares y qué productos se ignoran. Percibirá qué zonas se pueden ignorar o eliminar, y cuáles otras necesitan de una ampliación.

Temas Legales.

En su mayor parte, hemos dependido de cartas de crédito internacional, acuerdos comerciales y las compañías de tarjetas de crédito para aliviar los rigores de comprar y vender a través de las fronteras. Cuando la transacción es discreta, de un país a otro, las leyes de importación o exportación se pueden defender y aplicar. Pero la cuestión de quién es el propietario del ciberespacio es algo que tendrá que arreglarse en los

tribunales correspondientes. Y tendrá que transcurrir mucho tiempo todavía para que esto se produzca.

Por regla general, un equipo de marketing en buena forma puede afrontar las complicadas leyes que gobiernan la venta de mercancías entre un país y otro. Sabe que no tiene que hacer afirmaciones sobre sus productos que no pueden sostener en la práctica. La cultura internacional, sin embargo, ya es otra cuestión. Por lo tanto, se deben tener en cuenta las culturas de otros países a la hora de realizar el marketing por la Web.

1.3.5 Establecer unos Objetivos Realistas.

Cada establecimiento, ya sea de una gran empresa multinacional, de una organización anónima, o de una empresa, debe determinar qué espera conseguir al poner en práctica su presencia global y electrónica. En todo anuncio que se publica en una revista, o en cualquier proyecto, se emplea una gran cantidad de tiempo y esfuerzo. Ese nivel de esfuerzo todavía es más crítico para la red. Su mensaje puede llegar a ser leído por millones de personas, de modo que hay que llevar un cuidado especial para crear una página Web que produzca la respuesta deseada. Determinar la respuesta deseada es el primer requisito para diseñar la página Web.

Durante la Conferencia de la Red Mundial de la DCI, en febrero de 1995, cuando se le preguntó al público por qué deseaba aprender más sobre el uso de Internet y de la WWW, se encontraron las siguientes respuestas:

- Mejorar la imagen de la empresa.
- Mejorar el servicio al cliente.
- Encontrar nuevos clientes potenciales.
- Aumentar la visibilidad.
- Realizar transacciones.

- Expandir el mercado.
- Satisfacer las expectativas del cliente.
- Reducir costos.
- Adquirir velocidad o actualizarse antes de que sea demasiado tarde.

Por el momento, crearse una imagen como una empresa técnicamente actual, experta y ventajosa, es la razón más habitual que tienen los usuarios de la red para diseñar una página Web.

La Imagen de Vanguardia.

Optar entre un escaparate vacío y un escaparate retrasado es una dura elección. Si configura una página con excesiva precipitación, corre el riesgo de que su público se sienta desilusionado con sus esfuerzos apresurados. Las noticias sobre una página nueva, estimulante e intrigante en la red sólo viajan ligeramente más rápidas en Internet que las críticas a una página Web vacía de contenido.

Si crea la impresión de ser una empresa de vanguardia, también creará la imagen de una empresa que se preocupa por sus clientes, hasta el punto de estar dispuesto a explorar nuevas tecnologías y dominarlas en beneficio de sus clientes.

Relaciones Públicas.

El simple hecho de estar en Internet es como tener abierta una pequeña ventana a la oportunidad, desde donde se tiene que hacer un gran ruido y llamar la atención editorial. Si es usted el primero de su nicho que se conecta, o si su página Web ofrece algo singular, conseguirá que se hable mucho de usted en la prensa comercial. Esa clase de menciones permite que el resto del mundo sepa que su empresa es una organización de vanguardia, que cuenta con un equipo de pensadores avanzados. Y esa es una medida característica del éxito del marketing de red.

Una página Web bien situada puede cosechar una buena atención en las publicaciones especializadas de su industria por ser considerada como técnicamente avanzada. También es un lugar estupendo para comunicarse con la prensa. Los comunicados de prensa masivos, enviados por correo electrónico a editores y escritores quizá no sean el mejor enfoque a seguir.

Calificación del Cliente Potencial.

Es posible obtener un beneficio dedicándose exclusivamente a vender, sin marketing. Es posible obtener un beneficio dedicándose exclusivamente al marketing, sin tener vendedores. El coste de una hora de tiempo del vendedor difiere para cada empresa y, de hecho, para cada producto. Pero no cabe duda de que una gran parte de la función de marketing consiste en encontrar al mayor número posible de compradores calificados. El marketing procura que el vendedor se dedique a vender, en lugar de ir a buscar clientes. Para una empresa dedicada a la venta de un producto sofisticado, que exige un prolongado tiempo de contacto, el objetivo más atractivo de una página Web sería el de acortar el ciclo de venta. Un cliente potencial quizá necesite de mucha educación previa antes de apreciar el producto y de tomar una decisión de compra razonable. En tal caso, la WWW es un buen lugar para la promoción de esa clase de productos.

Su cliente potencial puede retirar electrónicamente educación, comprensión, aprecio y muchas cosas más mientras usted duerme. Las fotografías, los detalles técnicos y las respuestas a las preguntas planteadas con mayor frecuencia se encuentran disponibles en cualquier momento del día o de la noche. Cada vez que alguien comprende la respuesta electrónica a una pregunta alucinante, le habrá ahorrado a su vendedor por lo menos cinco minutos de tiempo de conversación telefónica. Como consecuencia de ello, el vendedor podrá dedicar más tiempo a atender clientes potenciales con el propósito de efectuar una venta, en lugar de emplear ese tiempo en la fase de educación e información al cliente. La página Web ha asumido el papel de

tutor del cliente potencial, de modo que el vendedor puede concentrarse en efectuar más ventas.

1.3.6 Venta de Productos.

Las compañías técnicas ya empezaron a utilizar con éxito Internet para el marketing, mucho antes de que llegara la WWW. Estos servidores "gopher" y "ftp" realizan la misma función que la red: son máquinas vendedoras de información. La información se pone a disposición de todo el mundo y la gente buscará mientras tenga tiempo. Dependiendo del producto los resultados pueden ser sorprendentes.

La página Web puede ser suficiente para cerrar un trato sin ayuda de un vendedor, dependiendo de los productos que venda. La página Web puede permitir que los clientes potenciales elijan colores, estilos, configuraciones, métodos de envío y condiciones de pago. Luego su computadora imprimirá las listas del pedido, las etiquetas de envío y preparará las facturaciones sin necesidad de ninguna intervención humana.

Interacción con el Cliente e Información Retroactiva.

Las páginas Web pueden contener formularios incluidos en largos bloques de texto para comentarios prolongados. De esta forma, son como el correo electrónico. Una página Web también tiene capacidad para contener secciones en las que los visitantes pueden dejar comentarios que puedan ser leídos por todos. De este modo, comparten atributos con los grupos de noticias.

La comunicación directa interactiva con los clientes efectivos y potenciales permite una información retroactiva más directa que nunca. Cada fase de desarrollo del producto, de su posicionamiento y promoción, pueden incluir el recurso más inteligente,

**ESTA TESIS NO SALE
DE LA BIBLIOTECA**

experimentado y experto del mundo: sus propios clientes, que se pueden convertir así, en parte de su equipo.

Comunicaciones Internas.

La tarea del marketing no está completamente dirigida hacia el exterior. Se emplea una gran cantidad de tiempo en comunicarse dentro de las paredes de una empresa. Tratar de conseguir que todos comprendan el posicionamiento del producto o la visión de la empresa, puede poner a prueba la paciencia de algunos. Mantener a los vendedores informados sobre los cambios ocurridos en la línea de producción y acerca de las actualizaciones de precios constituye a veces una tarea monumental. Del mismo modo que la red ofrece comunicaciones con los clientes efectivos y potenciales, también puede ofrecer mejores comunicaciones internas, dentro de la propia empresa.

Tráfico de la Página Web como Recompensa Propia.

La filosofía del tablón de anuncios nos indica la conveniencia de situar nuestros nombres delante de la gente con la mayor frecuencia que nos sea posible. Cuanto mayores sean las impresiones que reciban las personas acerca del producto, tanto más probablemente lo recordarán, confiarán en él y lo comprarán. Algunas páginas Web han sido diseñadas pensando en el tráfico por usuario. Los grandes almacenes de Internet y las revistas electrónicas tratan de vender espacio publicitario basado en el número de personas que acuden a visitar la página Web.

Provocar en la gente el hábito de visitar una página Web concreta plantea un desafío singular. Si su propósito al crear una página Web es el de difundir información técnica, el número de nuevos visitantes quizá no tenga tanta importancia para usted. Pero si su objetivo consiste en causar un elevado número de impresiones, puede utilizar ciertas técnicas que animarán a repetir las visitas. O simplemente, puede limitarse a comprar espacio o a patrocinar otras páginas.

1.4 SISTEMAS OPERATIVOS.

Un sistema operativo es un software que supervisa la forma en que se pueden usar los recursos de una computadora. En algunas computadoras el sistema operativo es un solo programa, y en otras es un conjunto de programas que interactúan entre sí de diversas maneras.

Conceptualmente, un sistema operativo puede ser dividido en tres componentes:

- Las utilerías
- Los Shell
- El Kernel

Utilerías.

Las utilerías de un sistema operativo son los comandos estándares y programas asociados con el sistema operativo. Por ejemplo, las utilerías del DOS son los comandos que uno espera encontrar en cada sistema de DOS: DISKCOPY, BACKUP, etc. Las utilerías son la parte más visible del sistema operativo.

Shell.

Shell significa cáscara o concha, el Shell es un sistema operativo que se encuentra entre el Kernel y el usuario, un Shell es un programa que ejecuta otros programas. En el DOS un programa de este tipo usualmente se llama intérprete de comandos.

Kernel.

La palabra Kernel, que significa núcleo, da la idea de la parte interior de una semilla o nuez y, de manera similar, el Kernel es la parte central de un sistema operativo. El

Kernel es el conjunto de software que proporciona las capacidades básicas del sistema operativo. Los servicios y la seguridad de los programas se apoyan en el kernel.

1.4.1 UNIX.

El primer sistema UNIX fue desarrollado por los laboratorios Bell de la AT&T en Murray Hill, Nueva Jersey, a finales de los años sesenta. Fue un sistema de investigación para probar nuevos conceptos de sistema operativo y proporcionar a un grupo de expertos en ciencias de la computación un ambiente de programación altamente productivo.

La mayoría de los similares al UNIX fueron derivados directamente de versiones del código del UNIX de la AT&T. Los fabricantes tuvieron licencias sobre el código de AT&T, lo convirtieron para ejecutar en sus propias máquinas y frecuentemente añadieron características adicionales para hacer más atractivo el producto. Debido a que la AT&T mantenía los derechos del nombre UNIX, otros fabricantes dieron a sus productos similares a UNIX otros nombres: XENIX, AIX, UCOS, etc., solo por mencionar unos cuantos. En la figura 1.4.1 se muestran algunos similares.

UNIX es un sistema multiusuario. Los diseñadores del UNIX anticiparon que diversas personas querrían usar el sistema, y que varias personas desearían tener archivos personales que no quisieran compartir con los demás usuarios. Puede imaginarse un sistema multiusuario como una oficina, donde los usuarios tienen sus propios escritorios separados pero también tienen acceso a algunos recursos que mantienen en común, como la fotocopidora y el enfriador del agua. En un sistema multiusuario, los recursos comunes pueden ser determinados directorios y la impresora.

Por otro lado, DOS es un sistema de un solo usuario, si se arranca un sistema DOS estándar, se tiene automáticamente acceso a cualquier archivo del sistema. (El acceso a los archivos ocultos y de sólo lectura del DOS es un poco más difícil que el de los

archivos normales, pero sólo un poco.)

La seguridad de los archivos es sólo una diferencia entre los sistemas multiusuario y los sistemas de un solo usuario. Si se quiere comenzar a usar un sistema multiusuario, se tiene que identificar usted mismo, para que el sistema sepa cuál usuario es usted. Por lo tanto, los usuarios de UNIX tienen que ejecutar un procedimiento de registro (log-in), en el cual le dice al sistema su nombre y contraseña, para probar que son quienes deseen ser. Con el DOS estándar simplemente se enciende la máquina y se comienza a trabajar, ya que no hay que identificarse, debido a que el DOS no tiene la capacidad de distinguir entre diferentes personas que pueden usar la misma máquina.

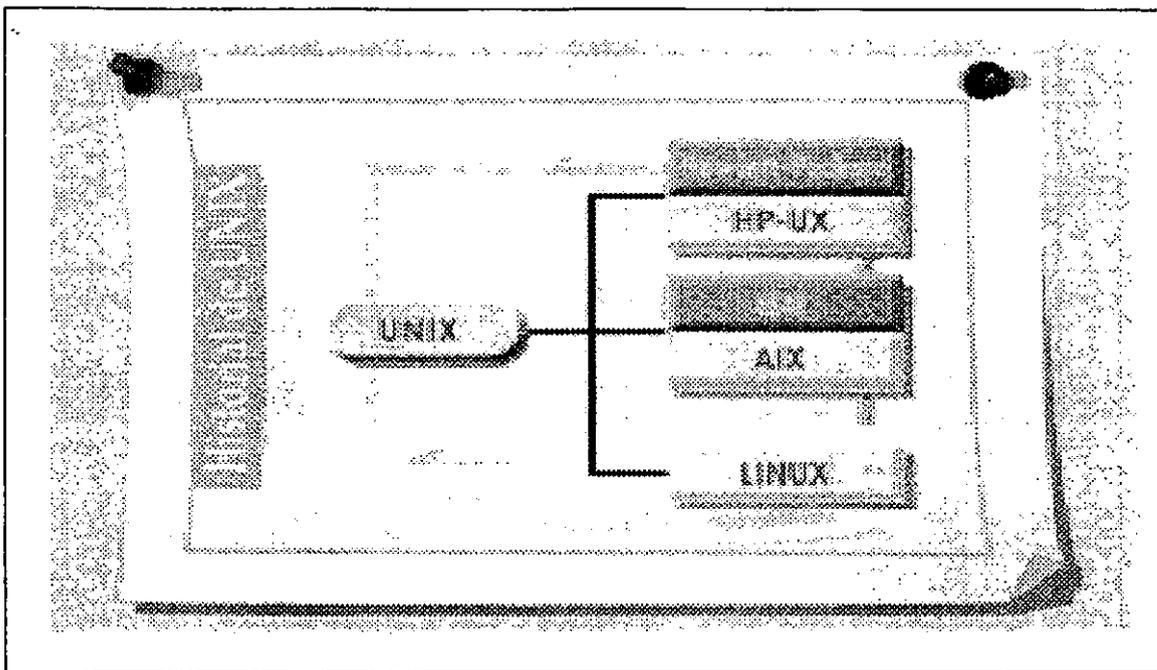


Figura 1.4.1 Clones de Unix.

Tanto el DOS como el UNIX proporcionan funciones que se pueden aprovechar para adaptar el sistema a las necesidades propias. Sin embargo, con el DOS estándar la personalización afecta a todo lo que hace con la máquina. Con el UNIX la personalización hecha por el usuario no necesita afectar a los demás. Típicamente,

cuando usted se registra en un sistema UNIX, el sistema identifica de quien se trata, y luego busca si se ha pedido alguna personalización mientras se está usando la computadora.

Esta personalización se mantiene en efecto por el tiempo en que se está trabajando. Si usted se despide de la máquina (log-out) y otro usuario se registra, el sistema limpia la personalización de usted y revisa si hay alguna personalización deseada por el nuevo usuario. Cada usuario comienza con su ambiente de trabajo completamente limpio, sin ser afectado por otros que quieran usar el sistema.

Un sistema multiusuario permite a los usuarios ajustar sus sesiones de trabajo de acuerdo con sus gustos, sin interferir con los demás que comparten la misma máquina. un sistema multiusuario le facilita designar archivos privados, a los que ningún otro puede tener acceso, así como archivos públicos disponibles para todos.

Multitarea Contra una Sola Tarea.

Dentro de un sistema UNIX, varias personas pueden usar el mismo sistema simultáneamente. Este proceso es llamado multitarea debido a que el sistema parece ejecutar varias tareas al mismo tiempo.

El DOS, por otro lado, es un sistema de una sola tarea. Puede ejecutar solamente una cosa a la vez. Unos cuantos arreglos especiales pueden hacer parecer que el DOS está ejecutando varias tareas al mismo tiempo (por ejemplo, con la cola de impresión se puede comenzar a imprimir algo y luego hacer otro trabajo mientras la impresora imprime), pero estos pocos arreglos especiales no se acercan a la versatilidad de la multitarea real.

Sin embargo, el Software Windows de Microsoft introduce una forma de multitarea real en el DOS. En esencia, el Windows y programas similares actúan como sistemas

operativos pequeños, trabajando dentro del mismo DOS. Ellos supervisan el proceso de multitareas.

Las facilidades de multitareas del UNIX son versátiles. No solamente varios usuarios pueden trabajar con el sistema simultáneamente, sino que un solo usuario puede, de hecho, ejecutar varios programas diferentes al mismo tiempo. Un usuario de UNIX a menudo puede comenzar la ejecución de un trabajo largo y trabajar en uno corto mientras espera a que termine el largo. Esta es la razón por la que se puede ser más productivo con el UNIX que con un sistema de una sola tarea, ya que se pueden hacer varias cosas a la vez.

UNIX es un sistema maduro. Ha estado en uso por 30 años y ha pasado por un periodo exhaustivo de revisión. Los programadores de UNIX han tenido mucho tiempo para eliminar los errores y las deficiencias. El UNIX también se ha beneficiado de los similares a UNIX, ya que la presión de la competencia ha forzado a las mejoras continuas.

A continuación se mencionan algunos de los clones más conocidos de UNIX:

- XENIX.
- LINUX.

XENIX.

En 1980 Microsoft obtuvo la licencia para fabricar su propia versión del UNIX y no solo adoptó su utilización en ordenadores personales sino que le añadió nuevas posibilidades, como el visual shell. Al nuevo sistema se le llamó XENIX. Desde entonces, han ido apareciendo nuevas versiones siendo creada por Santa Cruz Operation (SCO XENIX SYSTEM V) una de las más difundidas.

Características y Requisitos de XENIX.

- 1.- La característica principal es que XENIX es un sistema operativo multiusuario y multitarea.
- 2.- Posee la posibilidad de trabajar en multipantalla lo que significa que podemos trabajar en multitarea , manteniendo varias tareas interactivas. Esta característica hace que podamos utilizar otros sistemas operativos mientras estamos trabajando en XENIX.
- 3.- Está diseñado para recuperarse automáticamente, después de una interrupción. Cuando se produce una interrupción por fallo en el suministro eléctrico, al activarse de nuevo, XENIX verifica la integridad del disco corrigiendo los defectos antes de permitir la entrada al usuario.
- 4.- Aprovecha la disponibilidad de más memoria y menos disco para que sea mucho más rápido.
- 5.- Seguridad en el acceso de información. XENIX puede asegurar un registro, grupo de registros o archivos completos, para que dos usuarios no accedan al mismo tiempo en la misma información por medio de semáforos de sincronismos de las tareas.

A diferencia de otros sistemas operativos, al instalar el XENIX en un disco duro necesitará de tres de las cuatro particiones:

- La primera denominada sectores malos, la utiliza el propio sistema para anotar, por medio de un conductor de disco duro. Los sectores defectuosos del disco, evitando su utilización.
- La segunda , denominada principal , es en la que se encuentra el núcleo , o Kernel y es donde arrancamos el XENIX.
- La tercera, denominada del usuario, es donde se encuentran las utilidades del XENIX y los programas del usuario.

La memoria mínima que se requiere para trabajar con XENIX, dependerá de la

utilización que se vaya a hacer de él, si se utiliza en su versión base y un solo ordenador, la necesidad es de 512 Kb, pero si se desean utilizar todas, o casi todas las posibilidades, sus necesidades irán en aumento llegando a precisar 15 Mb, si se instala completo, en tres particiones del disco duro. Si está instalado en una red local cada terminal deberá disponer de 1Mb de memoria RAM para su perfecto funcionamiento.

Sus limitaciones también merecen una explicación, ya que lo que representa es una limitación para el usuario que trabaja interactivamente y puede representar una ventaja en los trabajos de programación.

Este es el caso del editor (ed) o de la utilidad copiar (cp) entre otras, donde el usuario no recibe ningún tipo de mensaje y si se está acostumbrado a trabajar con otros sistemas operativos, se espera un mensaje de entrada de comando o de confirmación de copia correcta de archivos, esta limitación o desventaja frente a otros sistemas, es de gran ventaja para la programación ya que estas utilidades pueden servir de entrada a otros programas, para lo cual es necesario que no existan líneas de mensaje.

Otras de las limitaciones de XENIX es que la interface shell es poco clara al usuario y un error mecanográfico puede destruir archivos, producir errores o salirse del sistema inadvertidamente.

Un sistema operativo en tiempo compartido está compuesto físicamente por un computador que coordina el uso de distintos recursos, entre ellos, varias terminales a través de las cuales diversos usuarios pueden comunicarse independientemente e interactivamente con el, ejecutando sus propios programas de forma individual.

El sistema operativo reparte el tiempo de CPU entre todos los usuarios del sistema, cambiando, en intervalos de tiempo muy cortos de ejecutar, las instrucciones de un programa de usuarios a las del siguiente, según un algoritmo preestablecido. Se

consigue con esto optimizar la utilización de la CPU y del resto de los recursos del sistema.

Componentes del Sistema Xenix.

Al igual que la mayoría de los sistemas operativos, el XENIX posee una estructura interna organizada en capas. La más interna de estas capas es el Núcleo, la parte del sistema que más cerca está del hardware del ordenador .

Las demás capas se disponen unas sobre otras de forma que las tareas que desempeñan dependerán de las funciones aportadas por las capas anteriores. La capa más externa recibe el nombre de Shell. Es la parte del sistema operativo más próxima al usuario y actúa como interface entre éste y el sistema. Un esquema a modo de ejemplo del Sistema Operativo sería el siguiente:

- Kernel o Núcleo
- Dispositivos y órdenes del sistema
- Utilidades
- Shell
 - Intérprete de órdenes
 - Lenguaje de programación.

Kernel o Núcleo del Xenix.

Es el corazón del sistema, y la parte más cerrada al usuario, reside permanentemente en memoria y realiza las funciones de más bajo nivel con respecto al hardware. Entre sus funciones principales se encuentran la planificación de tareas y la administración de la Entrada/Salida.

Forman parte de su cometido:

- Manejar los procesos:
 - Asignar recursos.
 - Planificar procesos.
 - Manejar solicitudes de servicio Shell.
- Manejar utilidades
 - Supervisando la transmisión de datos.
 - Controlador de periféricos.

Esencialmente, el Kernel es el conjunto de programas que controlan el acceso al ordenador, administran su memoria y asignan los recursos del sistema a los usuarios cuando éstos lo demandan.

Dispositivos y Órdenes del Sistema Xenix.

Por encima del Núcleo se encuentran los archivos de dispositivos y los archivos de órdenes del sistema. Los primeros son los manipuladores de dispositivo de hardware, interfaces entre el sistema operativo y el dispositivo físico. Los segundos son los archivos ejecutables correspondientes a las órdenes del sistema. Es una segunda capa que interacciona entre el Shell y el Núcleo del sistema. Las utilidades ejecutan una variedad de recursos y funciones especiales de mantenimiento del sistema.

Shell del Xenix.

Puede estudiarse al shell como dos puntos de vista: como intérprete de órdenes y como lenguaje de programación.

Intérprete de órdenes: Cuando el usuario introduce una orden, el Shell, que es un programa en continua ejecución, analiza la línea y llama a ejecución al programa o programas que realiza la función solicitada por la orden. Se ejecuta un Shell para cada usuario que se conecta al sistema. Existen cuatro Shells diferentes:

- **sh:** Bourne Shell, para el usuario de XENIX experimentado. Recibe su nombre de su creador, S. R. Bourne, y su indicativo por defecto es el símbolo dólar "\$". Este Shell está capacitado para redireccionar la entrada y salida estándar, interpretar los metacaracteres, manejar variables y filtros. Posee además su propio lenguaje de programación.
- **rsh:** Reduced Shell. Es el conjunto de la Bourne Shell que contiene las órdenes básicas. Entre las limitaciones de este shell están; la restricción del uso de la orden cd, impidiendo el cambio de directorio; no permite redireccionar la salida estándar, etc.
- **csch:** C Shell específica para programadores en lenguaje C. Tiene todas las características de la Bourne Shell, pero añade algunas habilidades específicas para el programador de C.
- **vsh:** Visual Shell. Este Shell permite el manejo del sistema a través de menús.

El administrador del sistema debe de adjudicar un tipo de Shell a cada usuario del mismo. Éste, a su vez, puede ejecutar cualquier Shell (si tiene autorización para ello) tecleando el nombre del Shell que quiere ejecutar. Cada vez que un usuario entra en un nuevo Shell se crea un nuevo proceso del que se sale pulsando las teclas Ctrl+D.

Lenguaje de programación: Combina, mediante estructuras de control, grupos de órdenes almacenados en archivos llamados Shell Scripts o procedimientos Shell. Además, posee sus propias sentencias o instrucciones.

En cuanto al funcionamiento del Shell se presenta el siguiente esquema:

1. login.
2. El Shell espera una orden:
 - El usuario introduce la orden.
 - El Shell interpreta la línea, consulta permisos y ejecuta la utilidad solicitada.
 - El usuario dialoga con la utilidad.

- El Shell Pregunta por la próxima orden.
3. Ctrl+D.
 4. Logout.

LINUX.

Linux es único en la evolución de sistemas operativos. Aunque no es un producto comercial respaldado por una gran corporación, es un sistema operativo fruto de la creación de un equipo heterogéneo de entusiastas de la computación en todo el mundo. Este equipo se valió de los recursos de internet para comunicarse y crear este sistema operativo. Podemos decir que LINUX es un clon bien habilitado del sistema operativo UNIX, pero de hecho es un sistema operativo multiusuario y multitareas que cumple con POSIX, para los procesadores 486 de intel y posteriores. POSIX es un estándar internacional de los sistemas operativos y software, en el que se detallan los estándares de interoperabilidad. Linux no requiere del MS-DOS o de Windows para funcionar; de hecho, puede sustituir a estos programas en su computadora.

Linux es el único sistema operativo actual disponible gratuitamente que proporciona capacidades de multitarea y multiprocesamiento a numerosos usuarios en plataformas de hardware compatible con las PCs de IBM, Linux lo aparta de los caprichos mercantiles de los diversos proveedores comerciales. Así, no estará comprometido a efectuar actualizaciones cada dos años y pagar por ello elevadas sumas, son muchas las aplicaciones para linux que están al alcance de todos en forma gratuita en el internet, así como está disponible en internet el propio código fuente para Linux, de esta manera usted puede ampliar el sistema operativo de acuerdo a sus necesidades, lo que no puede realizar con los sistemas operativos comerciales como Windows95, MS-DOS u OS/2.

Esta característica, también es una desventaja potencial para los usuarios de Linux, ya que como ningún vendedor comercial le da soporte, cuando se necesita ayuda no

basta con una llamada telefónica. Linux puede ser voluntarioso y quizá corra o no de manera adecuada en una amplia variedad de hardware, pero también puede dañar o eliminar archivos de información almacenados en su sistema. En la tabla 1.4.1 se muestra una lista del hardware que soporta linux.

Componente	Descripción
CPU	CPU 386/486 y compatibles
BUS	Bus local ISA, EISA Y VESA, no soporta bus de Microcanal
RAM	2 Mb. RAM mínimo; se recomienda 4 Controladores de DD Controladores de unidad de disco duro estándar AT, MFM,RLL,ESDI e IDE; así como varios controladores populares SCSI.
Espacio en disco	20 Mb. mínimo; más recomendable 80 Mb.
Monitores	Linux soporta tarjetas y sistemas de video Hércules, CGA, EGA, VGA Y SVGA.
Unidades de cinta	Cualquier unidad de cinta SCSI así como unidades Colorado Jumbo 120 y 250 con formato QIC 80.
Impresoras	Si tiene acceso a su impresora desde MS-DOS, también podrá hacerlo desde Linux
Tarjetas Ethernet	Entre las tarjetas soportadas están 3com, 3c503 y 3c503/ Novell NE1000, NE2000 y Western Digital

Tabla 1.4.1 Lista del Hardware que Soporta Linux

No obstante, Linux es relativamente estable en numerosos sistemas y, en consecuencia, le proporciona una económica oportunidad de aprender y usar uno de

los sistemas operativos más populares del mundo actual: UNIX. Linux es una alternativa a otros sistemas UNIX y puede emplearse en lugar de los sistemas que, en ocasiones, son costosos. Por tanto, si programa en un sistema UNIX durante el día, puede tener un sistema tipo UNIX en casa.

Características Generales de Linux.

Multitareas.

La palabra multitarea describe la habilidad de ejecutar, aparentemente al mismo tiempo, numerosos programas sin obstaculizar la ejecución de cada aplicación. Esto se conoce como multitareas preferentemente, porque cada programa tiene garantizada la posibilidad de correr, esto es, cada programa no se ejecuta si no hasta que el sistema operativo lo aparta para permitir que otros programas corran.

Este tipo de multitareas es exactamente lo que Linux hace. MS-DOS y windows 3.1 no soportan multitareas preferentemente, sólo llamada multitareas cooperativa, bajo la cual los programas corren hasta que ellos mismos permiten la ejecución de otro programa o no tienen otra cosa que hacer durante ese periodo.

Multiusuario.

La capacidad de Linux para asignar tiempo del microprocesador a numerosas aplicaciones simultáneas se prestó como consecuencia a servir a numerosas personas al mismo tiempo, cada una ejecutando una o más aplicaciones.

La particularidad en realidad destacada de Linux y sus características de multiusuario y multitareas, es que más de una persona puede trabajar en la misma versión de la misma aplicación de manera simultánea, desde las mismas terminales o desde terminales separadas.

Shells Programables.

El shell programable es otra de las características que hace de UNIX y en consecuencia de Linux, lo que es: el sistema operativo más flexible, aunque muchas versiones de UNIX y Linux incluyen más de un tipo de Shell, en general, todos funcionan igual. Un Shell hace el papel de intérprete entre el usuario y el Kernel (el Núcleo, corazón o cerebro, del sistema operativo Linux). La principal diferencia entre los tres Shell disponibles radica en la sintaxis de la línea de comandos. A pesar de que no es una limitación grave, encontrará dificultades si trata de usar comandos o sintaxis de Shell C en el shell Bourne o BASH.

Independencia de Dispositivos bajo Linux.

Al igual que UNIX, Linux comparte muchos de los mismos beneficios de la independencia de dispositivos. Por desgracia, una de las mejores características de Linux es al mismo tiempo una de las mayores desventajas: su independencia del mundo comercial. Linux ha sido desarrollado durante los últimos años por una variedad de programadores de todo el mundo, quienes no tienen acceso a todos los equipos creados para las PCs de IBM y las compatibles. De hecho, Linux no corre en algunas PCs de IBM, en especial las que utilizan el Bus de microcanal.

Comunicaciones y Capacidades de Red.

La superioridad de UNIX (y de Linux también sobre otros sistemas operativos es igual de evidente en sus utilerías para comunicaciones y red. Ningún otro sistema operativo incluye capacidades para red estrechamente acopladas, y tampoco tiene la flexibilidad integral de estas mismas características.

Si necesita hablar con otro usuario mediante la utilería de correo electrónico o bajar (transferir) archivos grandes de un sistema que está al otro lado del país, Linux le

ofrece los medios para hacerlo.

Aplicaciones.

Hoy en día, Linux cuenta literalmente con miles de aplicaciones disponibles, entre éstas hay programas de hojas de cálculo, de base de datos, de procesamiento de texto, para desarrollo de aplicaciones con una variedad de lenguajes para computadora y paquetes de telecomunicaciones para conectarse en línea.

Ventajas para los Profesionales de la Computación.

Linux le facilita un caudal de herramientas para el desarrollo de programas. Hay compiladores para muchos de los principales lenguajes de programación de computación actuales, como C, C++ y Smalltalk, si no le gustan éstos, Linux le ofrece herramientas para que construya sus propios lenguajes de computación. Para ello están disponibles herramientas como Flex y Bison

Incapacidad para Usar Software Común.

Otra desventaja es que es poco probable que sus aplicaciones generales para sistemas operativos como el DOS y el OS/2 funcionen bajo Linux. Por fortuna, estos sistemas pueden coexistir con Linux; por tanto, aunque no es posible usar los dos sistemas operativos al mismo tiempo, puede salir de Linux y arrancar el otro sistema para utilizar sus aplicaciones ahí.

Hay trabajos en evolución sobre emuladores que ejecuten programas del DOS y de Windows. Aunque el proyecto del DOS está más avanzado que el de Windows, ambos están en una fase inicial y no listos del todo para el momento de prueba. sin embargo , algún día no muy lejano, Linux podrá ejecutar aplicaciones del DOS y de Windows.

1.4.2 Windows NT.

Windows NT es un nuevo sistema operativo de la casa de Microsoft, a diferencia de Windows 3.11 para trabajo en grupo, no necesita MS-DOS para funcionar. Las limitaciones históricas, como el límite de 640 KBytes, ya no le afecta, además, tiene capacidad de acceso de 32 bits. Se puede desarrollar nuevo software capaz no sólo de trabajar con la alta frecuencia de reloj de los modernos procesadores, sino también con una arquitectura perfeccionada. Como consecuencia, un aumento considerable de la velocidad en el mismo hardware, por ejemplo: compilar un programa en Windows 3.1 se realiza en la mitad de tiempo.

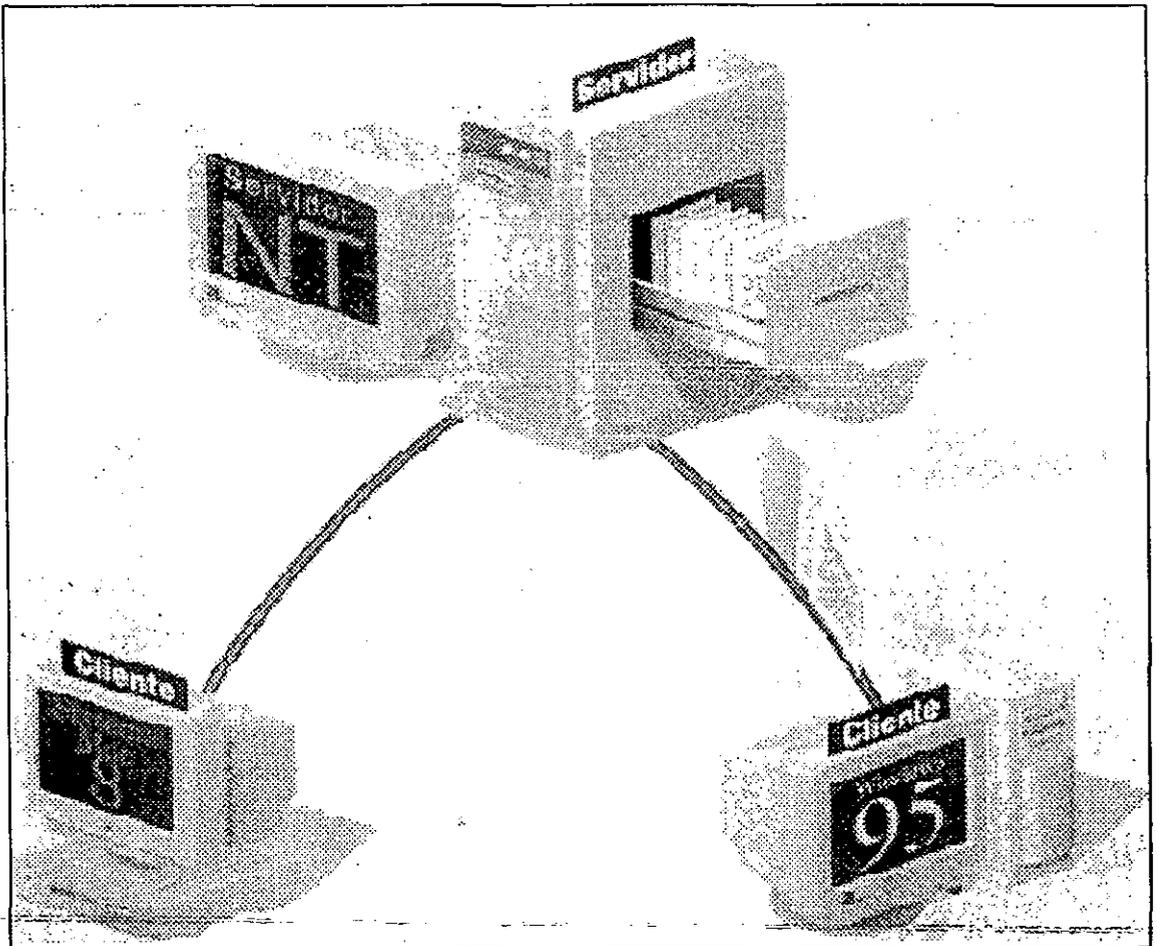


Figura 1.4.2 Windows NT Como Servidor de una Red.

Windows NT es un sistema operativo de red, completo. Dentro de las propiedades de red tienen un alcance considerablemente mayor que el de windows para trabajo en grupo. Se le ha introducido gran número de diarios de registro en los que el administrador del sistema puede comprobar las operaciones realizadas por cada usuario. En la Fig. 1.4.2 se puede apreciar un Servidor de Windows NT conectado con otros sistemas operativos

Multitarea.

Windows NT puede controlar por fin todo el hardware. Esto significa una auténtica multitarea: a cada programa se le asigna un tiempo de cálculo determinado y a su vez transcurrido ese tiempo el sistema recupera el control.

Con Windows 3.11 eso no ocurría, y el sistema recuperaba el control siempre dependiendo del capricho y la benevolencia del programa en cuestión. Si, por un error del programador, caía en un bucle sin fin o si quedaba absorbido por alguna función de llamada del sistema (como el formateo de un diskette), la multitarea quedaba paralizada, o incluso se venía abajo todo el sistema. con Windows NT ese peligro ya no existe.

Aún cuando se esté formateando un diskette, podemos trabajar con otras aplicaciones sin ningún problema y sin aplazamientos, y todas las aplicaciones quedan tan independientes unas de otras que el " colapso" de un programa no pueda afectar negativamente a los demás.

Compatibilidad.

Windows NT es de una versatilidad extraordinaria. Además de las nuevas aplicaciones Windows NT con acceso de 32 bits, funcionan igualmente todas las aplicaciones de Windows 3.1, la mayoría de programas de DOS y los programas compatibles con

POSIX y OS/2 en modo gráfico. Para ello, Windows NT lleva incorporado, por ejemplo, un intérprete de comandos de DOS.

Al igual que antes, Windows NT trabaja con el conocido sistema de datos de DOS, cuya propiedad más conocida es la limitación de ocho caracteres para los nombres de archivo.

A través de un cargador de arranque usted puede determinar inmediatamente después del arranque si quiere que se cargue Windows NT o el antiguo DOS.

Eso le permite cierta flexibilidad y ahorrar tiempo de cambio. Windows NT le ofrece además un nuevo sistema de archivos, llamado NTFS, cuyo nombre puede ser casi tan largo como se quiera y que contiene, además, una relación automática de errores y una protección para los datos.

Transportabilidad.

Además de los procesadores de intel de la serie x86, Windows NT también funciona con procesadores RISC, con lo que en el futuro bien pudiera ofrecer un entorno unitario para una amplia gama de ordenadores con diferentes capacidades.

Periféricos.

Todos los periféricos que aparecían hasta ahora con drivers en el CONFIG.SYS, ya no funcionan con Windows NT: Windows NT trabaja independientemente de DOS y no se preocupa de las entradas en el CONFIG.SYS.

Otra posibilidad se la ofrece el sistema de red que instale los periféricos en un ordenador dotado de Windows para trabajo en grupo y conéctelo con su ordenador Windows NT a través de la red.

Red.

A causa de la relativa rigidez que presenta actualmente el hardware necesario para Windows NT, este programa no está inmediatamente al alcance de todos los usuarios. Sin embargo, a través de red, Windows NT funciona sin ningún problema con Windows para ordenadores de trabajo en grupo, de modo que los ordenadores pequeños es mejor equiparlos con este último.

Muchos de los protocolos de red, incluidos los conocidos TCP / IP del mundo de UNIX, también se soportan, de modo que son posibles combinaciones heterogéneas de Windows y UNIX con otros ordenadores.

Windows NT Server 4.0 vs UNIX.

Windows NT se escoge a veces por razones presupuestales, ya que muchos profesionales de la tecnología de la información no están dispuestos a invertir en hardware más costoso donde se usa la mayoría de los sistemas operativos UNIX comerciales. Lo que realmente importa es que el costo global de la implantación de Windows NT que incluye la administración del sistema, los costos asociados con caídas del sistema, llamadas de teléfono para soporte técnico, pérdida de datos por la no confiabilidad del sistema, etc. En la tabla 1.4.3 se muestra el costo de Windows NT por número de usuarios.

Windows NT 4.0 Server	5 usuarios	\$ 809 USD
Windows NT 4.0 Server	10 usuarios	\$ 1,129 USD
Enterprise Edition	25 usuarios	\$ 3,999 USD
Enterprise Edition	50 usuarios	\$ 4,799 USD

Tabla 1.4.3 Costo de Windows NT.

Para otros profesionales más conscientes de los costos, la elección obvia sería Linux, FreeBSD, NetBSD o OpenBSD. No cuestan nada, sin embargo son tan estables y funcionalmente son equivalentes a los sistemas operativos UNIX comerciales (SunOS, Solaris, AIX, HP-UX, IRIX, Digital UNIX). En la tabla 1.4.4 hacemos una comparación en base a 4 aspectos:

	Windows NT	UNIX
Funcionalidad	<ul style="list-style-type: none"> • Es Multiusuario. • El Programa del servidor de Comunicaciones tiene un costo adicional. • Carece de cotas. 	<ul style="list-style-type: none"> • Es Multiusuario. • El Programas de Comunicaciones es gratuito. • Contempla seguridad, password , permisos y cotas.
Confiabilidad	<ul style="list-style-type: none"> • Pantalla azul de la muerte. 	<ul style="list-style-type: none"> • Pánico del Kernel.
Adm. Del Sistema	<ul style="list-style-type: none"> • Posee un Graphical User Interface . 	<ul style="list-style-type: none"> • Posee un Graphical User Interface.
Desempeño	<ul style="list-style-type: none"> • Más espacio en memoria y disco. • Archivos de sonido son importantes. 	<ul style="list-style-type: none"> • El Kernel se confecciona de acuerdo a las necesidades. • No requiere una interface gráfica como NT.

Tabla 1.4.4 Windows NT vs UNIX .

Windows NT Server 4.0 vs LINUX.

Debido a que la plataforma que NT utiliza más es INTEL, escogemos al sistema Linux para realizar una comparación en la tabla 1.4.5.

El sistema operativo Linux sería la elección para sitios con presupuesto limitado o en corporaciones enormes que demanden servidores de varios procesadores y que requieran de sistemas operativos escalables. Una compañía pequeña o mediana, que tenga que correr procesos en tareas no críticas, que esté dispuesto a contratar administradores adicionales para operar el servicio de correo Exchange o el Servidor

de Información de Internet de Microsoft, que tenga presupuesto substancial para licencias de Microsoft "por servidor" o "por usuario", entonces NT sería la elección.

Componente	Sistema Operativo Linux	Windows NT Server 4.0
Sistema Operativo.	Gratis o \$49.95 US (CD-ROM).	5-Usuarios \$809 US. 10-Usuarios \$1129 US. Enterprise Ed. 25-Usuarios \$3,999 US.
Soporte técnico gratis en línea.	Si, <u>Linux Online</u> o <u>Redhat</u>	No
Fuentes del Kernel.	Si	No
Servidor Web.	Apache Web Server	MS IIS
Servidor FTP.	Si	Si
Servidor Telnet.	Si	No
Servidor SMTP/POP3.	Si	No
DNS.	Si	Si
Sistema de archivos de red.	NFS y SMB	SMB
Servidor de Ventanas X (Correr aplicaciones remotas basadas en GUI).	Si	No

Herramientas de Administración Remotas.	Si, todas	Sólo "User Manager for Domains" y "Server Manager"
Servidor de News	Si	No
Compiladores C y C++	Si	No
Perl 5.0	Si	No
Control de Revisión	Si	No
Número de sistemas de archivo posibles	32	3
Cotas de Disco	Si	No
Número de GUI's a escoger from	4	1

Tabla 1.4.5 Comparación Windows NT vs LINUX.

El sistema Linux o FreeBSD pueden fácilmente rebasar el desempeño y funcionalidad de una solución NT, realizarlo sobre una plataforma INTEL relativamente económica y cuyo costo sería \$ 0.00, un precio que Bill Gates difícilmente puede mejorar.

1.5 COMUNICACIONES.

1.5.1 Redes de Computadoras.

Una Red permite conectar varias computadoras entre sí, con el fin de compartir sus recursos e información como documentos, bases de datos o recursos físicos. Las redes suelen clasificarse según su extensión en:

- **LAN (Red de Área Local).**

Las redes de área local se definen como una interconexión de computadoras mediante un medio de transmisión dentro de una corta distancia.

La extensión de este tipo de redes suele estar restringida a una sala o edificio, aunque también podrían utilizarse para conectar dos o más edificios próximos, estas redes se caracterizan por:

1. Distancia pequeña, hasta cinco kilómetros.
2. Velocidad de transmisión de orden de miles de bits por segundo.
3. Medio de comunicación compartido por todos los dispositivos conectados a la red.
4. Flexibilidad en la topología, es decir facilidad en la modificación y reconfiguración de la distribución física de los dispositivos de la red.

- **MAN (Red de Área Metropolitana).**

Una red de área metropolitana es esencialmente una red local muy grande que cubre una ciudad entera, suministrando el transporte de datos a grandes velocidades aproximadamente del orden de 100 Mbps (Megabits por segundo).

Típicamente la red MAN conectará redes LAN de más baja velocidad a través de una ciudad o región, solucionando las limitaciones de ancho de banda.

- **WAN (Red de Área Amplia).**

Una red de área amplia cubre un espacio muy extenso, conectando a ordenadores de una ciudad o un país completo. Para ello se utilizan las líneas de teléfono y otros medios de transmisión más sofisticados, como pueden ser las microondas, fibra óptica, cable submarino y satélites. Varias redes pueden conectarse entre sí formando una red lógica de área mayor, algunas de las aplicaciones son las siguientes: acceso a programas remotos, acceso a bases de datos remotas. y facilidades de comunicación.

1.5.1.1 Tipos de Redes.

Por la relación que hay entre sus miembros, las redes se subdividen en dos grandes grupos: las redes con servidor y redes punto a punto. En una red basada en un servidor (server-based), los recursos a compartirse centralizan en una máquina denominada "servidor" (server). Las demás máquinas, denominadas estaciones de trabajo (workstations), sólo pueden usar recursos propios o del servidor. A su vez, las redes basadas en servidor, aceptan dos subclases: con servidor dedicado o no dedicado. En el segundo, la máquina que funciona como servidor, lo hace también como estación de trabajo.

En una red punto a punto (peer-to-peer) cualquier estación puede ofrecer recursos para compartir. Las que no ofrecen recursos se llaman clientes (client) y las que lo hacen anfitrión/cliente (host/client). Las ventajas y desventajas de un tipo de red frente al otro, son los derivados de la centralización de recursos. En general, las redes importantes tienden a ser basadas en servidores dedicados, los que presentan las siguientes ventajas:

- Un servidor dedicado tiene más capacidad de trabajo que una máquina que opera además como estación.
- Ofrece más seguridad contra accesos no autorizados por su información centralizada en vez de distribuida.

- Las redes que ofrecen mayor seguridad contra pérdidas accidentales de información trabajan con servidores dedicados.
- Es más práctico para hacer actualizaciones de programas y copias de respaldo, la centralización de archivos.
- Cuando una estación de una red punto a punto ofrece recursos para compartir, le queda menos memoria libre que cuando sólo usa los de otras estaciones. La diferencia puede ser tal que no se pueda cargar el programa de aplicación que debería ejecutarse en la estación.
- No se corre el riesgo de que una estación que se conecte deshabilite el sistema.
- Las redes en las que hay terminales corriendo sistemas operativos diferentes, tienen servidores dedicados.

1.5.1.2 Estructura de una Red de Comunicaciones.

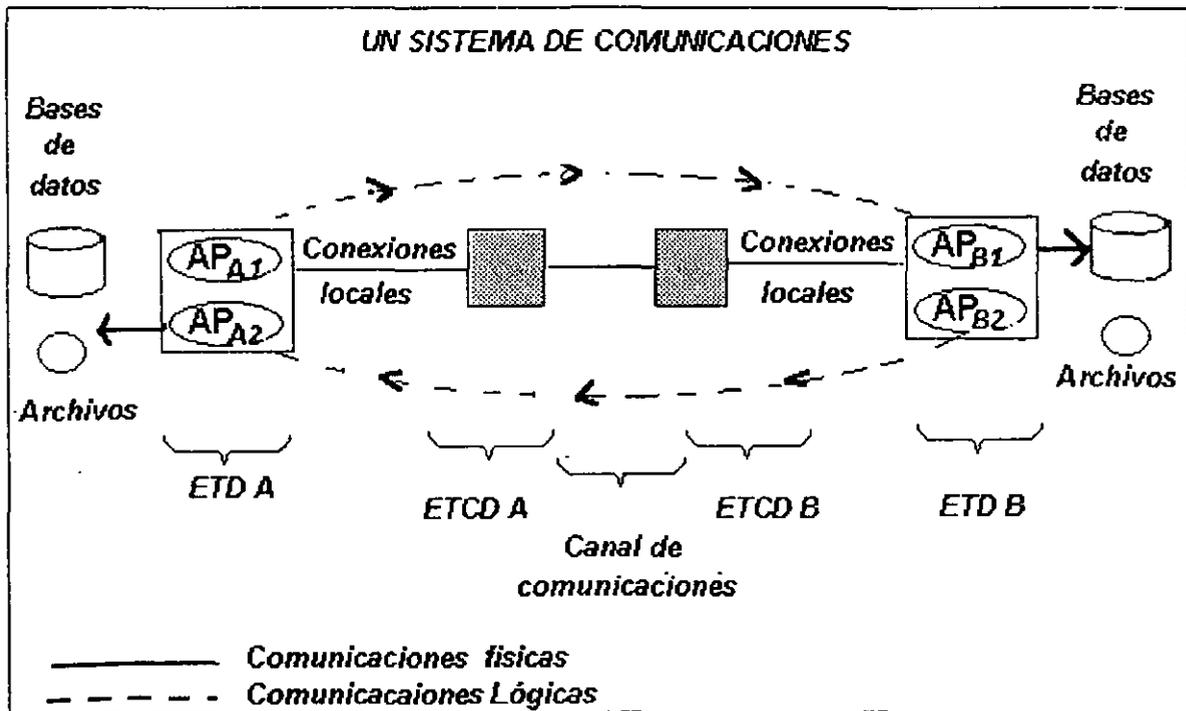


Figura 1.5.1.2 Sistema de Comunicaciones.

La finalidad de redes de comunicaciones es conectar equipos terminales de datos de forma que puedan compartir recursos, intercambiar datos, apoyarse entre sí y permitir

realizar tareas desde lugares geográficamente remotos. La figura 1.5.1.2 muestra que la red proporciona comunicaciones lógicas y físicas entre terminales y computadoras conectadas.

1.5.1.3 Componentes de una Red.

- **Servidor (Server).**

El servidor es la máquina principal de la red, la que se encarga de administrar los recursos de la red y el flujo de la información. Muchos de los servidores son dedicados, es decir, están realizando tareas específicas, por ejemplo, un servidor de impresión sólo para imprimir; un servidor de comunicaciones sólo para controlar el flujo de los datos, etc.

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

- **Estación de trabajo (Workstation).**

Es una computadora que se encuentra conectada físicamente al servidor por medio de algún tipo de cable. Muchas de las veces esta computadora ejecuta su propio sistema operativo y ya dentro, se añade al ambiente de la red.

- **Sistema Operativo de Red.**

Es el sistema (Software) que se encarga de administrar y controlar en forma general la red. Para esto tiene que ser un Sistema Operativo Multiusuario, como por ejemplo: Unix, Netware de Novell, Windows NT, etc.

- **Recursos a compartir.**

Al hablar de los recursos a compartir, estamos hablando de todos aquellos dispositivos de Hardware que tienen un alto costo y que son de alta tecnología. En estos casos los más comunes son las impresoras, en sus diferentes tipos: Láser, de color, plotters, etc.

- **Hardware de Red.**

Son aquellos dispositivos que se utilizan para interconectar a los componentes de la red, serían básicamente las tarjetas de red NIC (Network Interface Cards) y el cableado entre servidores y estaciones de trabajo, así como los cables para conectar los periféricos.

1.5.1.4 Objetivos Perseguidos al Construir una Red.

- 1 Compartir recursos con el fin de que los programas, datos y equipos estén disponibles para cualquiera de la red que así lo solicite, sin importar la localización física del recurso y del usuario.
- 2 Proporcionar una alta contabilidad y disponibilidad, al contar con fuentes alternativas de suministro, con duplicados o réplicas en dos o tres o más copias del mismo recurso.
- 3 Disminuir el gasto, ya que las computadoras pequeñas tienen una mejor relación costo/rendimiento comparada con máquinas grandes o mainframes.
- 4 Mantener niveles de seguridad razonables sin hacer más engorrosa la utilización del sistema, y adaptar de forma rápida el sistema, a las necesidades cambiantes.
- 5 Integrar sistemas de comunicación incompatibles, reducir el número de protocolos de comunicación que se utilizan en la organización.
- 6 Aumentar la capacidad de la red para manejar más usuarios y archivos de datos de gran volumen, como los de multimedia.
- 7 Permitir que los usuarios de distintas aplicaciones compartan información en diversos formatos y normas, sin que tengan por qué conocer dichas diferencias: transparencia.

1.5.1.5 Aplicaciones de Redes.

Las aplicaciones son las siguientes:

- Intercambio de datos, información y programas.
- Mantener bases de datos actualizadas instantáneamente y accesibles desde distintos puntos.
- Facilitar la transferencia de archivos entre miembros de un grupo de trabajo.
- Compartir periféricos (impresoras láser, plotters, discos ópticos, etc.)
- Bajar el costo del software comprando licencias de uso múltiple en vez de muchas individuales.
- Correo electrónico y comunicación con otras redes, con minis y mainframes (gateway), así como mantener usuarios remotos vía módem entre otras.

1.5.1.6 Estructuras.

- **Árbol o Jerárquica.**

Todas las estaciones cuelgan de un ordenador central y se conectan entre ellas a través de los hubs que haya instalados, control de tráfico: fácil, la tolerancia a fallas depende del nodo (server o nodo), su crecimiento es fácil y se mejora: habilitando funciones de control a ciertos nodos.

Esta topología puede presentar grandes problemas, como la posibilidad de aparición de cuellos de botella, que no sólo entorpecen el tráfico de información, sino que además no es confiable en el respaldo de dicha información; también existe la posibilidad de que se paralice completamente la red, si el nivel más alto tiene la capacidad de manejo sobre los demás niveles, porque si el nivel más alto llega a fallar, los demás nodos no pueden hacer nada por sí mismos.

La figura 1.5.1.6.1 muestra la topología.

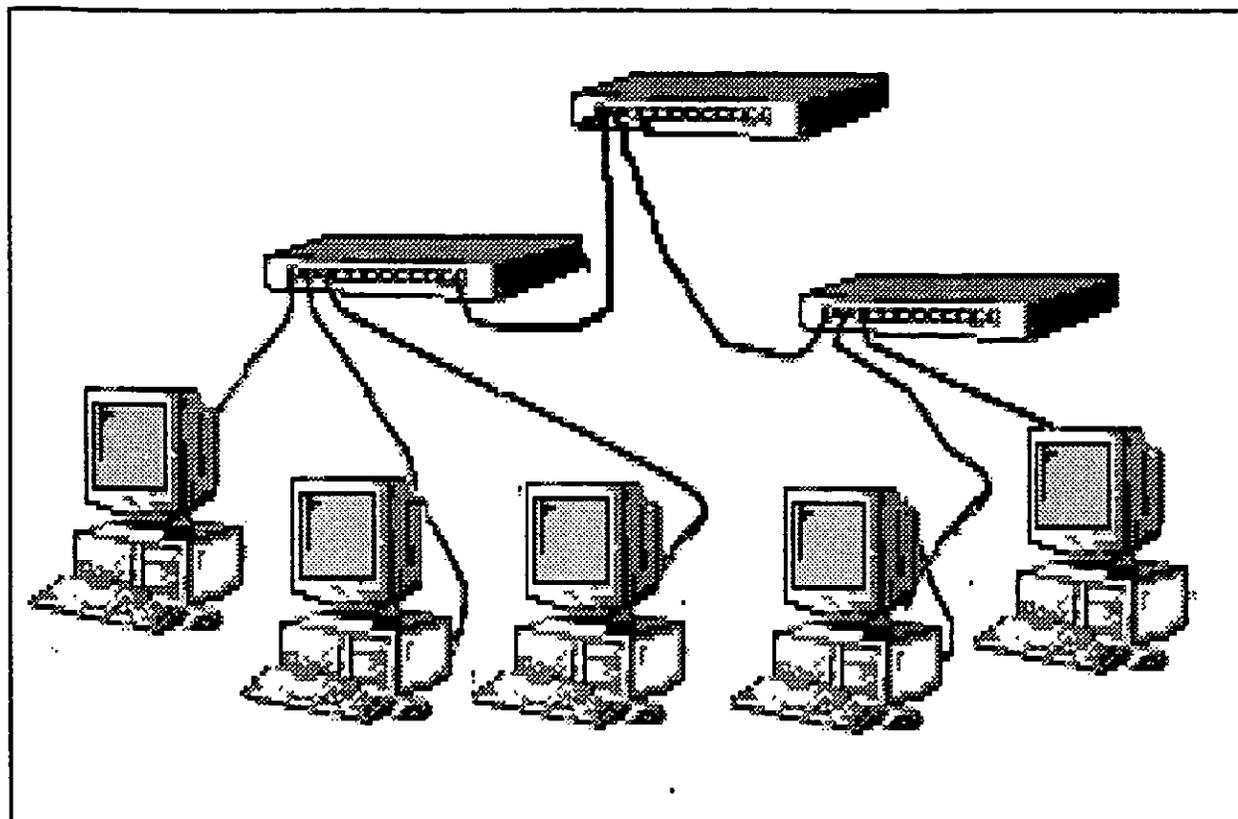


Figura 1.5.1.6.1 Topología Jerárquica o de Árbol.

- **Bus u Horizontal.**

En esta estructura el bus es el medio de transmisión y recepción, aquí se tiene un enlace por cada nodo, y éstos se conectan a un enlace que une todas las estaciones. Es importante mencionar que todas las computadoras tienen la misma jerarquía, así que si una de ellas llegara a fallar no afecta el funcionamiento del resto de la red, la figura 1.5.1.6.2 muestra la estructura.

Esta estructura es la más utilizada por redes LAN, debido a su fácil control de flujo de información, sin embargo tiene un gran inconveniente, ya que generalmente se cuenta con un solo bus y en el momento que este falle, paraliza por completo el funcionamiento de la red.

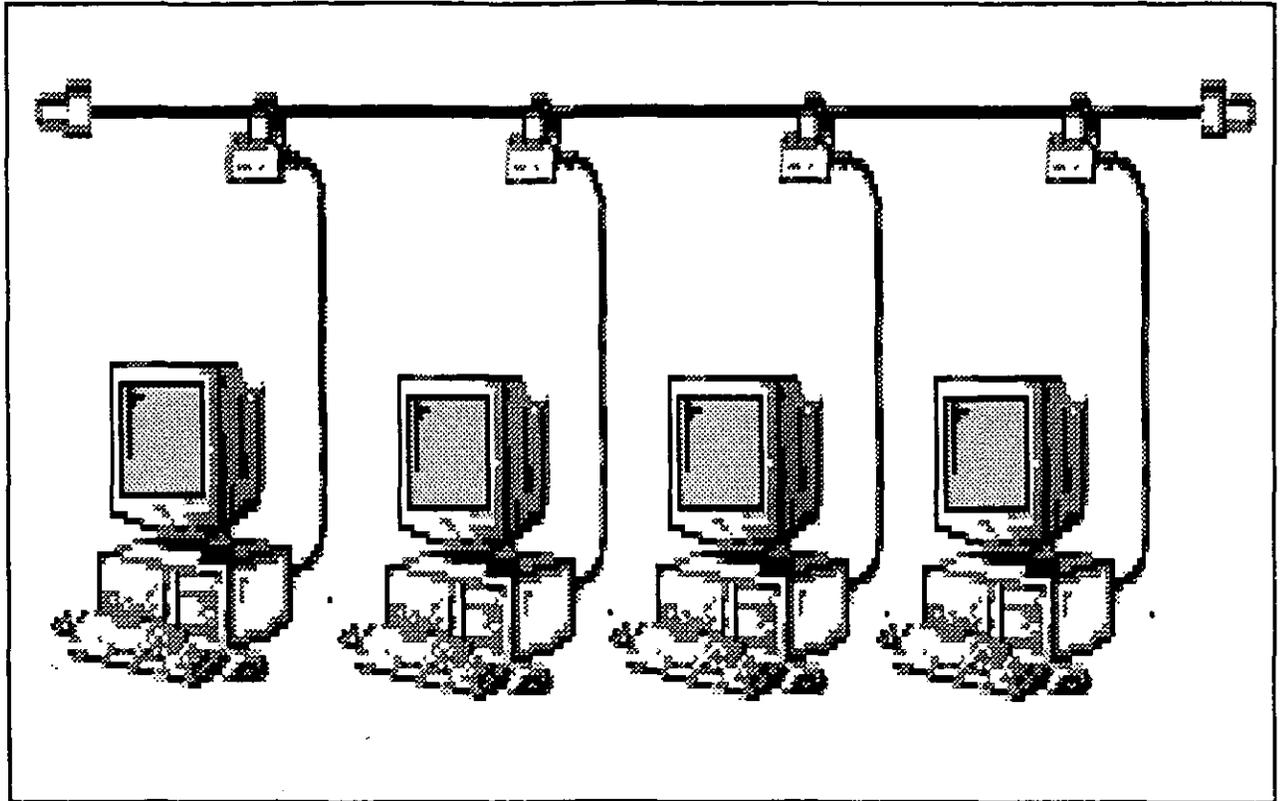


Figura 1.5.1.6.2 Topología de Bus.

- **Estrella.**

Esta configuración está siendo eliminada poco a poco, puesto que todo el sistema se centra en la estación central, ésta es responsable de encaminar el tráfico hacia los demás equipos, y controlarlos, es capaz de localizar averías en líneas, pero si la estación central falla entonces deja de funcionar la red y todo el sistema se ve afectado.

El nodo central, para n estaciones, tiene $n-1$ enlaces, mientras que las otras estaciones tan sólo tendrían uno, que es el que les vendría de la estación central.

La figura 1.5.1.6.3 muestra la configuración.

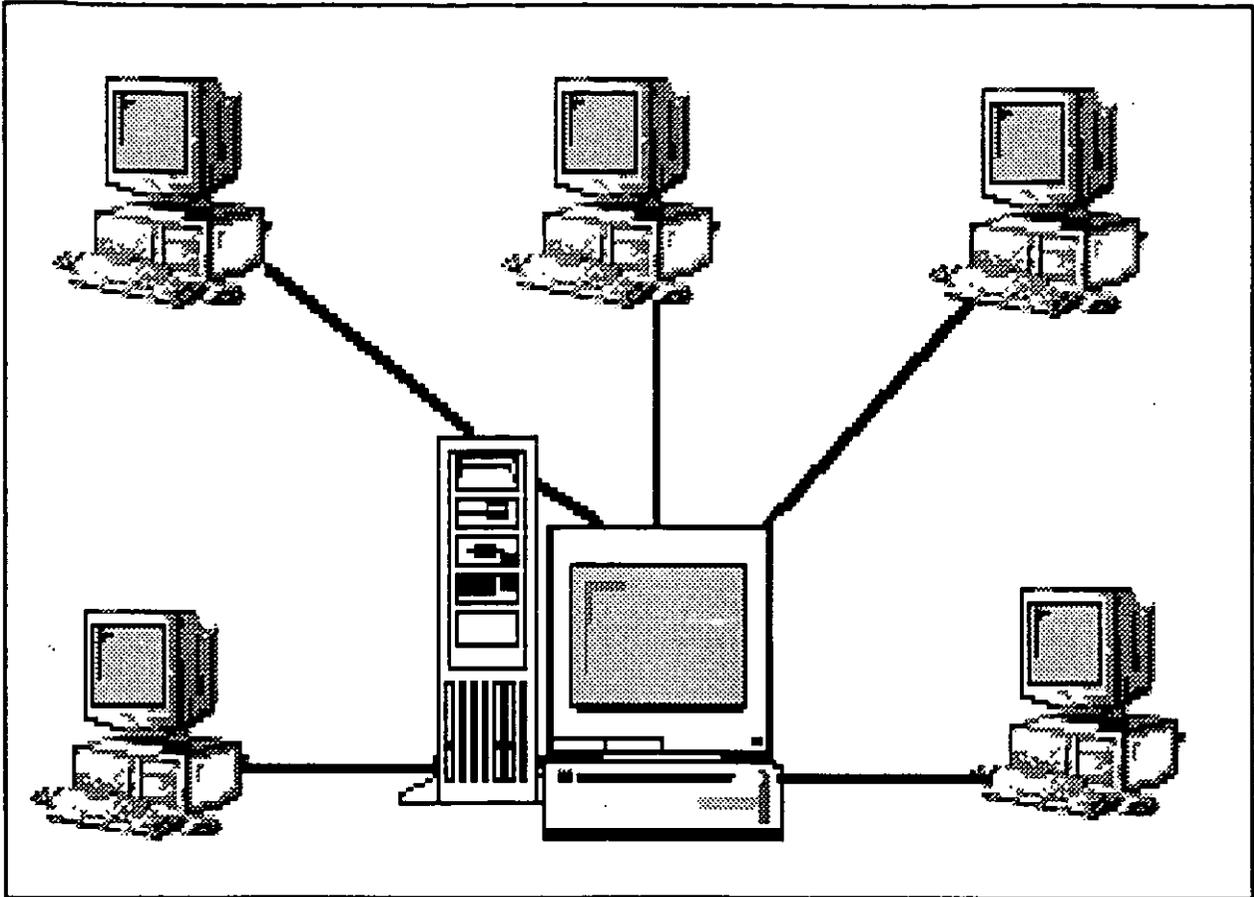


Figura 1.5.1.6.3 Topología Estrella.

- **Anillo.**

Esta configuración recibe este nombre, debido a que los nodos de la red están conectados en un ciclo, enlazando cada nodo, tanto hacia la derecha como a la izquierda. Generalmente la información fluye en una sola dirección, donde cada nodo recibe la señal y la transmite al nodo siguiente mostrándolo en la figura 1.5.1.6.4.

Tiene una ventaja sobre las redes de tipo estrella y árbol, ya que aquí es raro que se presente la saturación de información, además la lógica para ponerla en operación es sencilla, ya que lo único que debe de hacer cada equipo, es aceptar datos y enviarlos al siguiente equipo. Otra ventaja es tener todos los nodos con la misma jerarquía, si un equipo llega a quedar fuera de servicio, la red puede seguir funcionando.

Sin embargo también tiene defectos, el principal es que si llegara a fallar el canal de transmisión entre dos nodos, toda la red se paraliza.

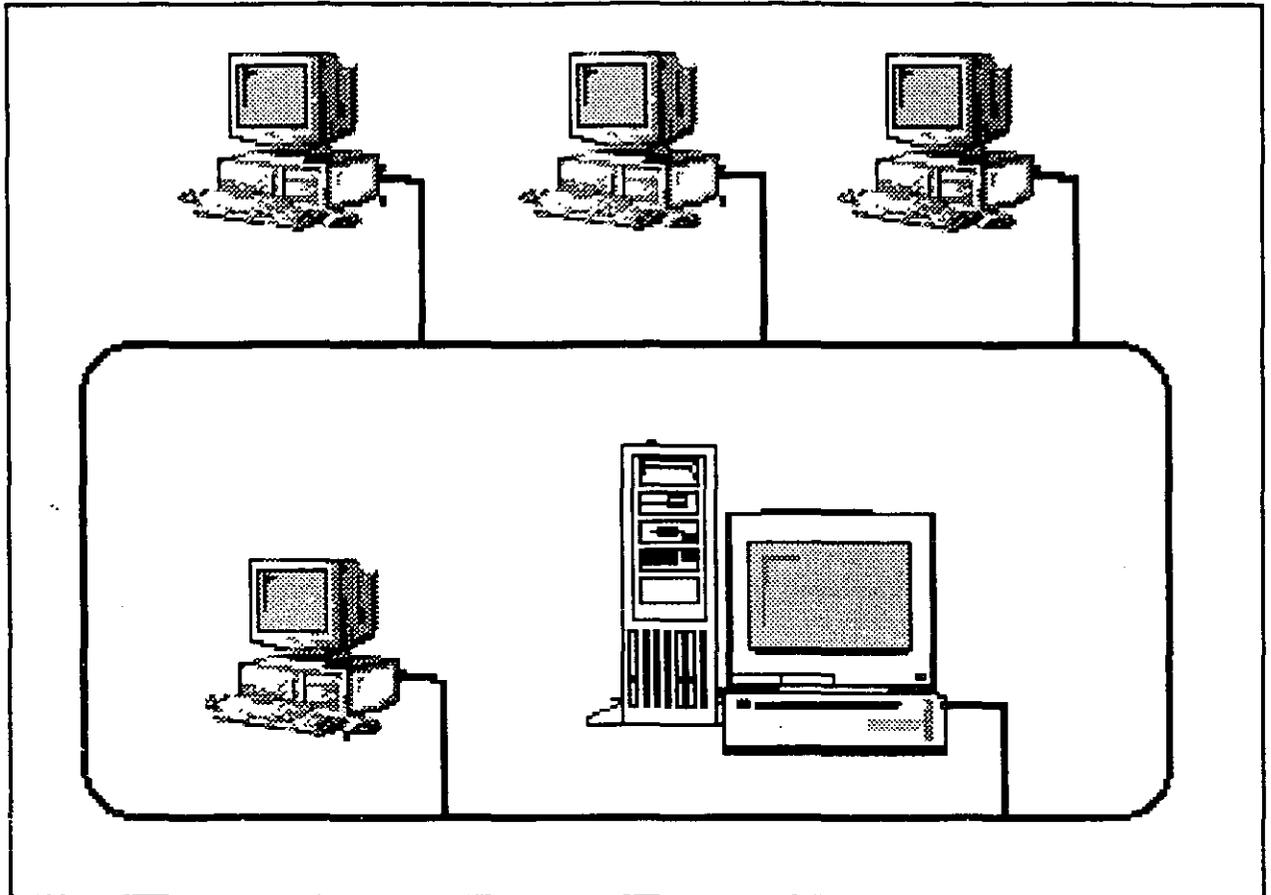


Figura 1.5.1.6.4. Topología de Anillo.

- **Malla.**

Esta estructura es atractiva debido a su inmunidad a problemas de saturación de información y averías, ya que existen varios caminos para dar información a través de los equipos, la información puede darse a través de diferentes trayectorias, en caso de que algún nodo se encuentre averiado u ocupado; además se puede enviar información simultáneamente a través de diferentes caminos. Esta red transmite la información rápidamente y de manera confiable, sin embargo es costosa, pero a su vez son las más flexibles.

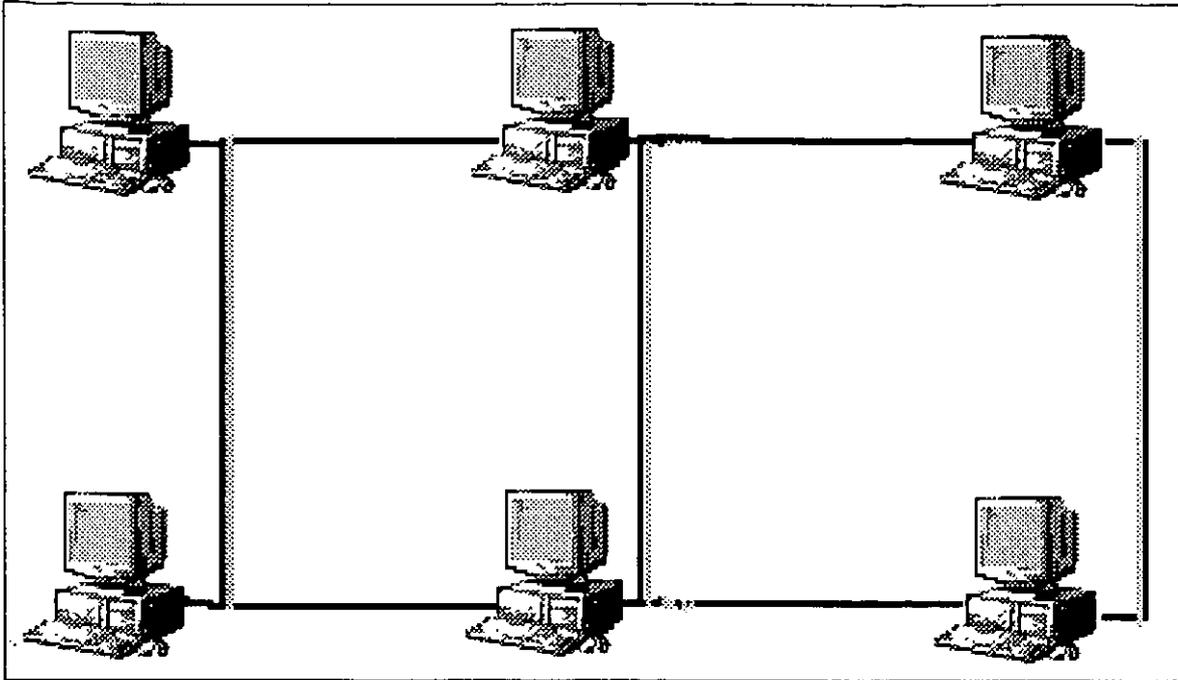


Figura 1.5.1.6.5. Topología de Malla.

1.5.1.7 Objetivos de las Topologías.

Los objetivos de las topologías de las redes son:

- Fiabilidad y costo, el manejo del número de errores en la transmisión de datos.
- La capacidad de recuperación de errores por fallos, en el canal o en los nodos.
- Tolerancia a fallas.
- Facilidad de mantenimiento.
- Facilidad de aislamiento de sectores en caso de fallos.
- Disponibilidad de un sistema de diagnóstico de fallas.
- Disminuir la distancia entre nodos.
- Disminuir el número de nodos intermedios a los dispositivos.
- Proporcionar el canal menos costoso para un servicio determinado.
- Que el usuario obtenga bajos tiempos de respuesta a sus requerimientos y rendimiento.

1.5.2 Modelos de Interconexión entre Redes.

1.5.2.1 Concentradores.

Comúnmente conocidos como hubs, estos elementos se basan en el principio de interconexión más básico, podemos entenderlos como un armario de conexiones donde se centralizan todas las conexiones de una red, o sea un dispositivo con entradas y salidas, que no hace nada más que centralizar conexiones, trabaja en las capas 1 y 2 del modelo OSI, sus funciones son:

Sondeo de terminales, conversión de códigos, de protocolos, de velocidades, comparación de datos, administración remota, detección y corrección sencilla de problemas. Suelen utilizarse para implementar topologías físicas en estrella, pero funcionando como un anillo o un bus lógico. Podemos distinguir entre dos tipos de hubs, los activos y los pasivos.

- **Activos** Pueden conectar nodos a 600 metros como distancia máxima, tienen unos 10 puertos. Pueden amplificar y repetir señales, hay algunos que realizan estadísticas.
- **Pasivos** Únicamente son cajas de conexión. Tienen unos 10 puertos, pueden conectar nodos a 30 metros de distancia como máximo.

1.5.2.2 Repetidores.

Estos operan en el nivel físico de la OSI, ya que interconectan redes iguales (del mismo tipo). Su utilización sirve para extender la distancia máxima de la LAN o para unir o interconectar distintos soportes de comunicación, aunque también puede servir para unir varios segmentos o varios anillos constituyendo una LAN física y lógica única. La figura 1.5.2.2 muestra la posición de un repetidor, hay que tener claro que un repetidor no analiza la información que le llega, hace una transmisión transparente de todas las tramas de un segmento de LAN a otro (en ambos sentidos). Regenera las señales. Lo

que sí hace es restaurar el preámbulo (parte que activa los sincronismos) dado que no varía nunca; el resto lo amplifica, también corrige la frecuencia y la amplitud. Ofrece la conversión de señales y de interfaz para acoplar distintos soportes de comunicación manteniendo a nivel lógico una sola LAN.

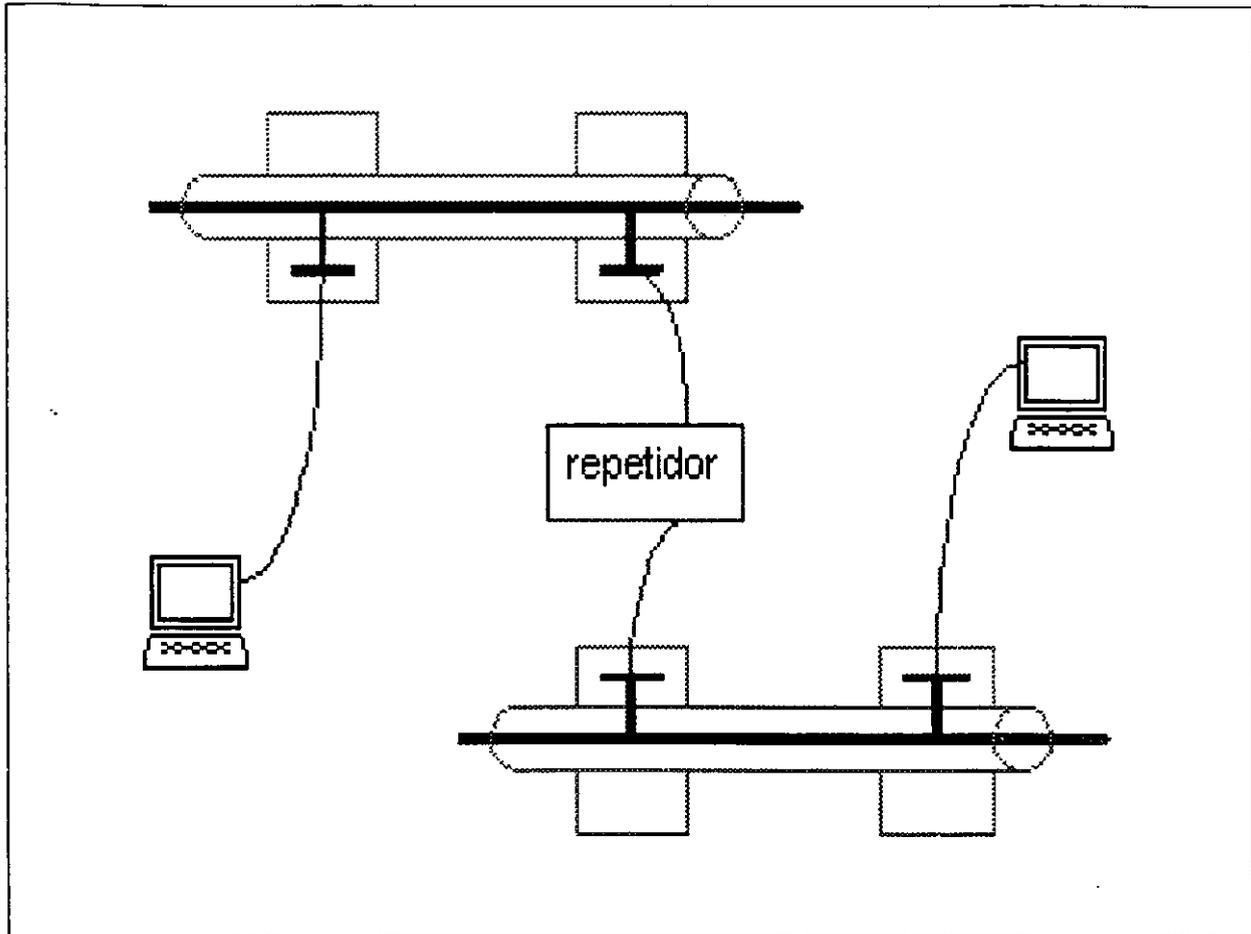


Figura 1.5.2.2 Repetidor.

1.5.2.3 Puentes.

Los puentes o "bridges" son dispositivos que operan en el nivel de enlace de datos. No analizan los campos de datos, los puentes delimitan el tráfico entre redes, a las redes que tienen acceso directo y deben preservar las características de las LANs que interconectan (retardo de transmisión, capacidad de transmisión, probabilidad de pérdida, etc.). Estos elementos filtran el tráfico en función de una tabla de direcciones.

La decisión si son transmitidos hacia delante o no se toma en función de la dirección destino que se halla en cada paquete.

Dan servicio de conexión mejor que los repetidores, ya que éstos accesan paquetes de información para leer tanto la dirección de origen como la dirección destino, si el destino esta dentro de la red para que pueda llegar dentro de su destino.

Para que los puentes puedan lograr esto, es muy importante saber las direcciones locales y las remotas, en la tabla de direcciones. Una ventaja que tienen los puentes sobre los repetidores es que para el usuario, los puentes permiten ver varias redes conectadas como una red extendida que permite el acceso a recursos nuevos, segmentan el tráfico en la red dejando pasar solo la información.

Existen varios tipos de puentes como: simple, transparente, de encapsulamiento y de encaminamiento de fuente.

- **Puente Simple.**

Utiliza una tabla de direcciones y está basada en un conocimiento previo. Las ventajas que tiene son la velocidad y la simplicidad pero le falta flexibilidad y el número de estaciones depende del tamaño de la tabla.

- **Puentes Transparentes.**

Permite la conexión entre dos redes que utilizan los mismos protocolos tanto en la capa física como en la capa de enlaces de datos, su inicialización es automática y realiza funciones de reenvío de tramas, de auto aprendizaje de las estaciones de la red, y de resolución de los posibles bucles que existan en la topología de la red.

En caso de tener dos redes conectadas por medio de un puente de este tipo, como la figura 1.5.2.3.1, su funcionamiento consiste en los siguientes pasos:

1. El puente lee las direcciones destino de todos los mensajes transmitidos por los dispositivos de la red A.
2. El puente ignora todos los mensajes dirigidos a los dispositivos ubicados en la red A.
3. El puente acepta todos los mensajes dirigidos a dispositivos ubicados en la red B y utilizando los protocolos comunes, envía estos mensajes a la red B.
4. El puente realiza las mismas funciones para todos los mensajes transmitidos por la red B.

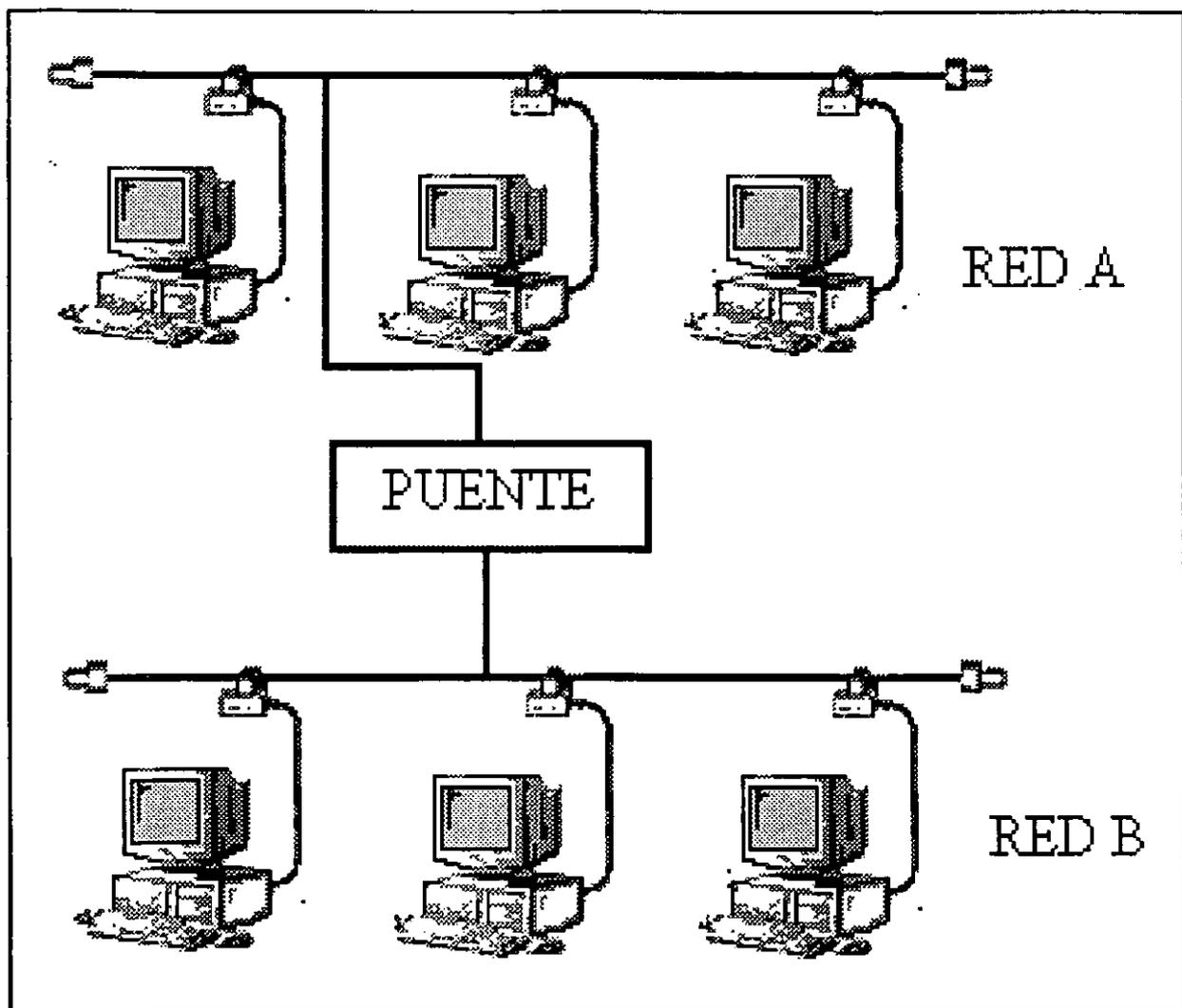


Figura 1.5.2.3.1 Puente Transparente.

Para que esto se pueda llevar a cabo se requiere que el puente conozca la ubicación de los dispositivos, esto se puede hacer por configuración manual o por función directa del equipo. Se lee la dirección de cada paquete de información que se transmite, y se compara con las existentes en la base de datos decidiendo si se ignora o se manda a otra red.

- **Puente de Encapsulamiento.**

Este puente encapsula los mensajes en un nuevo formato, viajan por un backbone, y se desencapsulan hasta llegar a su destino, la figura 1.5.2.3.2 muestra un puente de encapsulamiento y los pasos a seguir si se desea mandar un mensaje a la red A hacia la red B son los siguientes:

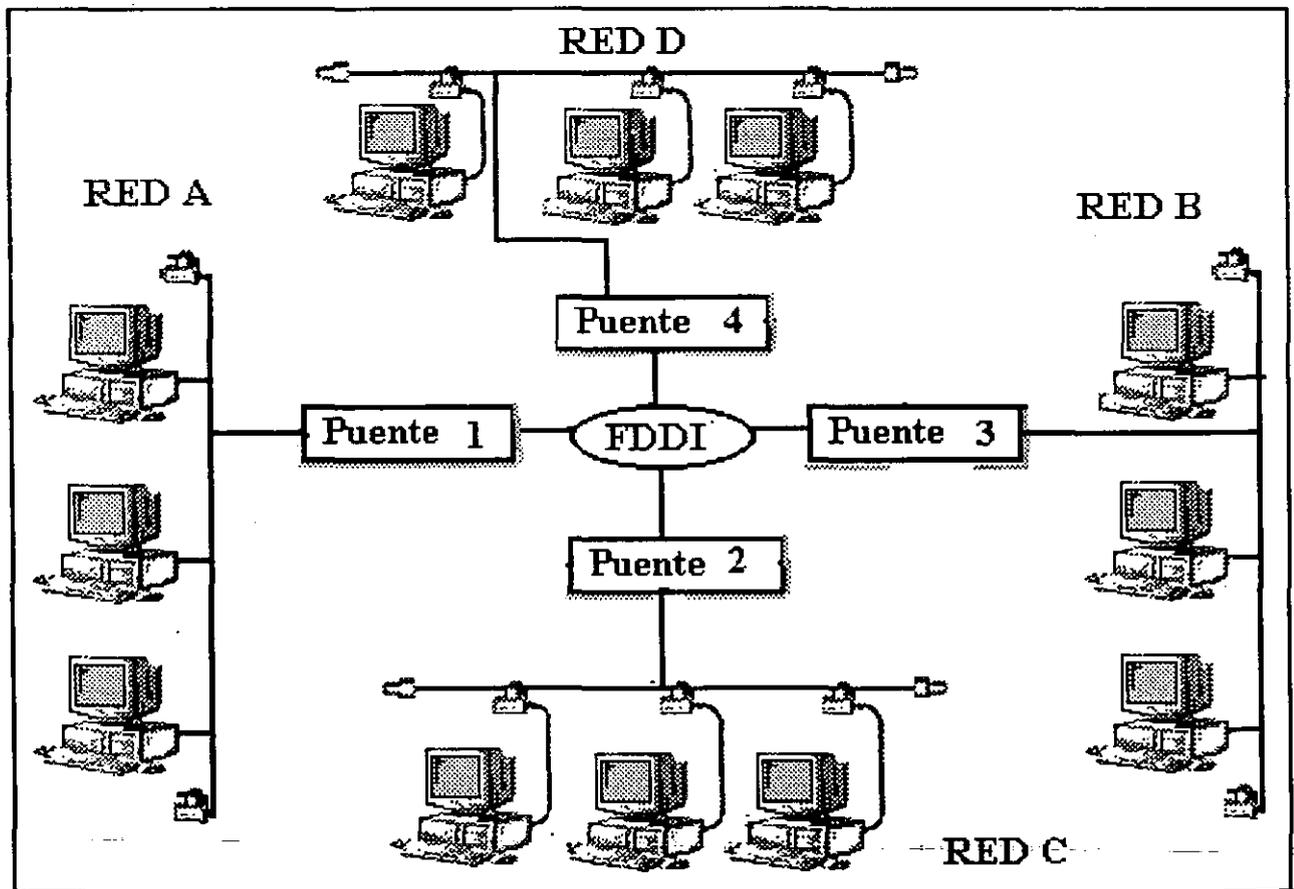


Figura 1.5.2.3.2 Puente de Encapsulamiento.

1. El puente 1, usando los protocolos de la red A lee la dirección destino de todos los mensajes transmitidos por dispositivos ubicados en la red A.
2. El puente 1 ignora todos los mensajes dirigidos a dispositivos ubicados en la red A.
3. El puente 1 acepta todos los mensajes dirigidos a otras redes, coloca el mensaje en una envoltura FDDI y lo envía a través del Backbone.
4. El puente 2 recibe el mensaje, quita la envoltura y revisa la dirección destino. Como el destino no le pertenece, ignora el mensaje.
5. El puente 3 recibe el mensaje, quita la envoltura y revisa la dirección destino, como esta dirección si le pertenece, utiliza los protocolos de Ethernet para obtener el paquete.
6. El puente 4 recibe el mensaje, quita la envoltura y revisa la dirección destino. Como el destino no le pertenece, ignora el mensaje.
7. El puente 1 elimina el mensaje encapsulado del backbone de FDDI.

- **Puente de Encaminamiento Fuente .**

Describe un método de puenteo en redes Token Ring, el cual requiere que un paquete exploratorio proporcione la información necesaria para hacer llegar un mensaje a su destino. Aquí los puentes no requieren almacenar una base de datos con direcciones, ya que se basan en la información contenida en la envoltura del mensaje por lo que deben describirse las rutas más convenientes. La estación origen, determina la ruta que seguirá la trama e incluye esta información en la misma en forma de identificadores de puentes y de LANs, de manera que un puente retransmite la trama si su identificador está en la ruta designada, sino, la desecha. En este tipo de puentes no se requieren tablas de encaminamiento, ya que este únicamente debe conocer sus identificadores. la ruta que escoge la estación origen hasta cualquier destino viene dada por unos modos de direccionamiento.

En la figura 1.5.2.3.3 se muestran cinco redes Token Ring enlazadas por tres puentes, para enviar un mensaje de la red 1 a la red 5 se tendría que hacer lo siguiente:

1. El dispositivo origen de la red 1, envía un paquete de exploración, el cual tiene un formato especial que es reconocido inmediatamente por el puente.
2. Al recibir este paquete, el puente graba en el mismo, el número de conexión por la cual llegó y su identificación en una sección de la envoltura del paquete.

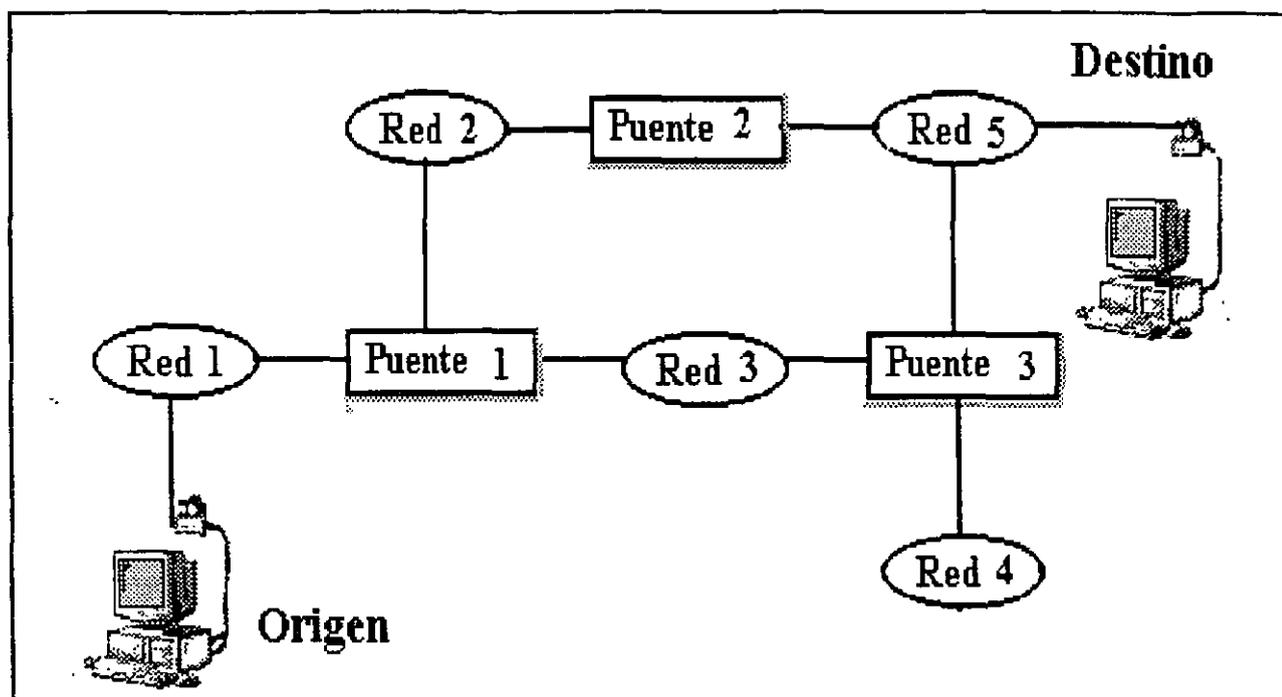


Figura 1.5.2.3.3 Puente de Encaminamiento Fuente.

3. El puente envía este paquete a través de todas las conexiones, excepto por la que llegó, originando así copias del paquete explorador .
4. El destino de la red 5, recibe varios paquetes exploradores, cada uno indicando la ruta que siguió, eligiendo el más adecuado ya sea por rapidez o por ser el más directo, después envía una respuesta a la red 1 indicándole la ruta adecuada.
5. El dispositivo de la red 1 almacena esta información para que siempre que envíe un mensaje a la red 5 utilice esa ruta, empaquetando la información en un formato especial que reconozca el puente .
6. El puente recibe esos paquetes y lee la información que contienen para saber por donde debe de enviarlo.

1.5.2.4 Ruteadores.

Los ruteadores trabajan sobre las tres capas inferiores de red (red, enlace y física), en la capa de red se incluye la dirección lógica de red, la cual es comúnmente asignada por el administrador de la red. También puede extender el tamaño de la red, las redes a conectar pueden tener diferentes protocolos en las capas física y de enlace de datos. Así como leen la dirección de los paquetes de información y deciden la ruta que deben de seguir, pueden seleccionar uno de entre varios caminos según parámetros como retardo de transmisión, congestión, etc., pero sólo examinan los paquetes dirigidos a ellos. Estos dispositivos, dependen del protocolo usado, en el nivel de red se controla el tiempo de vida de un paquete. El tiempo requerido para que un paquete vaya de un punto a otro de la Internet, hará que el tamaño máximo de ésta sea mayor o menor.

1.5.2.5 Multiplexor.

Los multiplexores son dispositivos que permiten optimizar los canales de comunicación de la red. Combinan las señales digitales de múltiples dispositivos en una única señal que se transmite a través de un solo canal. Debe de existir otro multiplexor en el otro extremo del enlace de transmisión, de forma que las señales mezcladas en un extremo, puedan separarse en el contrario.

1.5.2.6 Gateways.

Las redes con distintos tipos de hardware y distintos protocolos, por ejemplo SNA y OSI, se comunican entre sí por medio de un gateway. A diferencia de un router o un bridge, el gateway traduce el protocolo de una red y lo convierte en el protocolo utilizado por otra red. Para lograrlo realizan la conversión completa de una arquitectura a otra, sin modificar los datos transmitidos, de modo que los protocolos utilizados en la red fuente, pueden ser entendidos en la red destino. Actúan en niveles superiores, a partir de la capa 4 de OSI. Cuando se habla de gateways en el ámbito de LAN, se está

haciendo referencia a los routers y cuando no son LANs, actúan como traductoras de información

1.5.3 Medios de Transmisión.

La transmisión de datos en las redes, puede ser por dos medios:

- 1.- Terrestres: Son limitados y transmiten la señal por un conductor físico.
- 2.- Aéreos: Son "ilimitados" en cierta forma. Transmiten y reciben las señales electromagnéticas por microondas o rayo láser.

Los terrestres son los siguientes:

- **Par Trenzado.**

Es el que comúnmente se utiliza para los cables de teléfonos, consta de 2 filamentos de cobre, cubiertos cada uno por plástico aislante y entrelazados el uno con el otro. Existen dos tipos de cable par trenzado: el blindado, que se utiliza en conexiones de redes y estaciones de trabajo y el no blindado, que se utiliza en las líneas telefónicas y protege muy poco o casi nada de las interferencias. La tabla 1.5.3.1 a) muestra la estandarización de los colores, generalmente se utiliza la categoría 5, y la figura 1.5.3.1 b) muestra la conexión de cable de par trenzado

Nº PAR	Color Conductor Nº 1	Color Conductor Nº 2
1	Blanco	Azul
2	Blanco	Anaranjado
3	Blanco	Verde
4	Blanco	Marrón
5	Blanco	Gris Oscuro
6	Rojo	Azul

Tabla 1.5.1.3.1 a) Estandarización de colores.

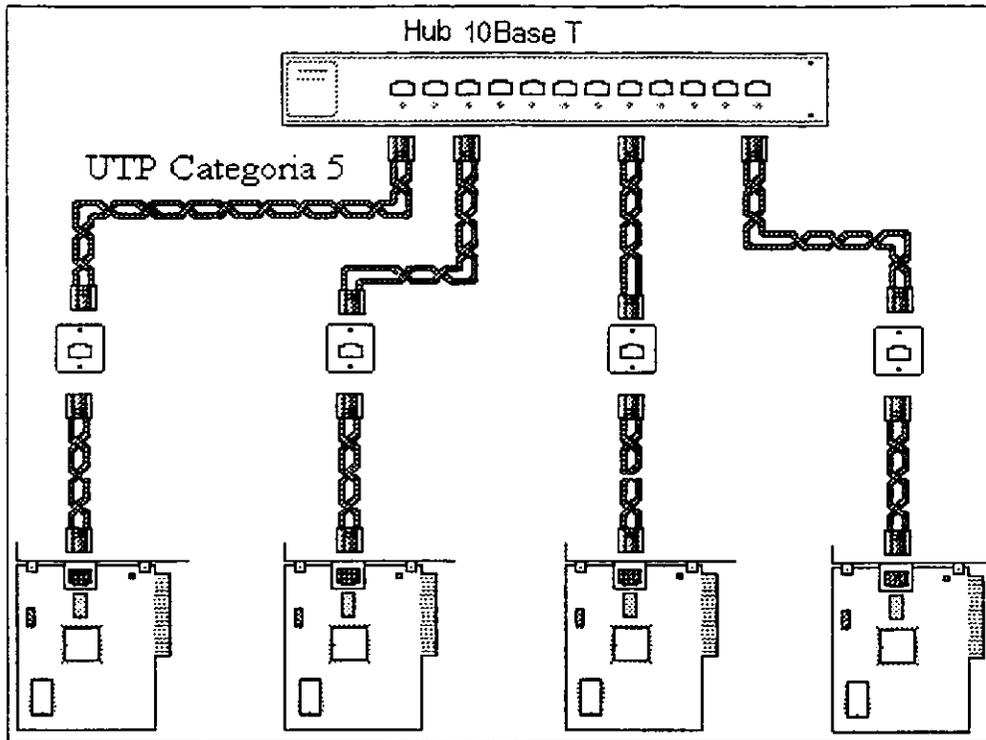


Figura 1.5.3.1 b) Par Trenzado.

- **Cable Coaxial.**

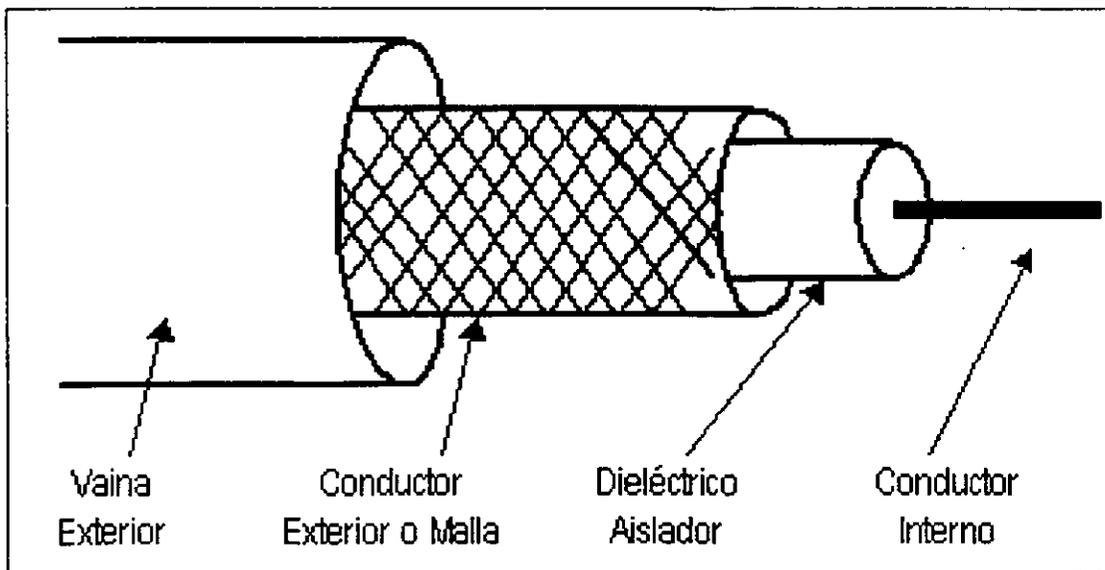


Figura 1.5.3.2 Cable Coaxial.

Este tipo de cable es popular en las redes, por la poca susceptibilidad de interferencia y por el ancho de banda, los datos son transmitidos por dentro del cable en un ambiente completamente cerrado, bajo una cubierta exterior. Se puede observar en la figura 1.5.3.2, la estructura de un cable, formado por conductores, dieléctricos y un recubrimiento o vaina exterior.

- **Fibra Óptica.**

Es un filamento de vidrio sumamente delgado diseñado para la transmisión de la luz. Las fibras ópticas poseen enormes capacidades de transmisión, del orden de miles de bits por segundo. Además de que los impulsos luminosos no son afectados por interferencias causadas por la radiación aleatoria del ambiente. Actualmente la fibra óptica reemplaza en grandes cantidades a los cables comunes de cobre.

La figura 1.5.3.3 muestra como está formado un tramo de fibra óptica.

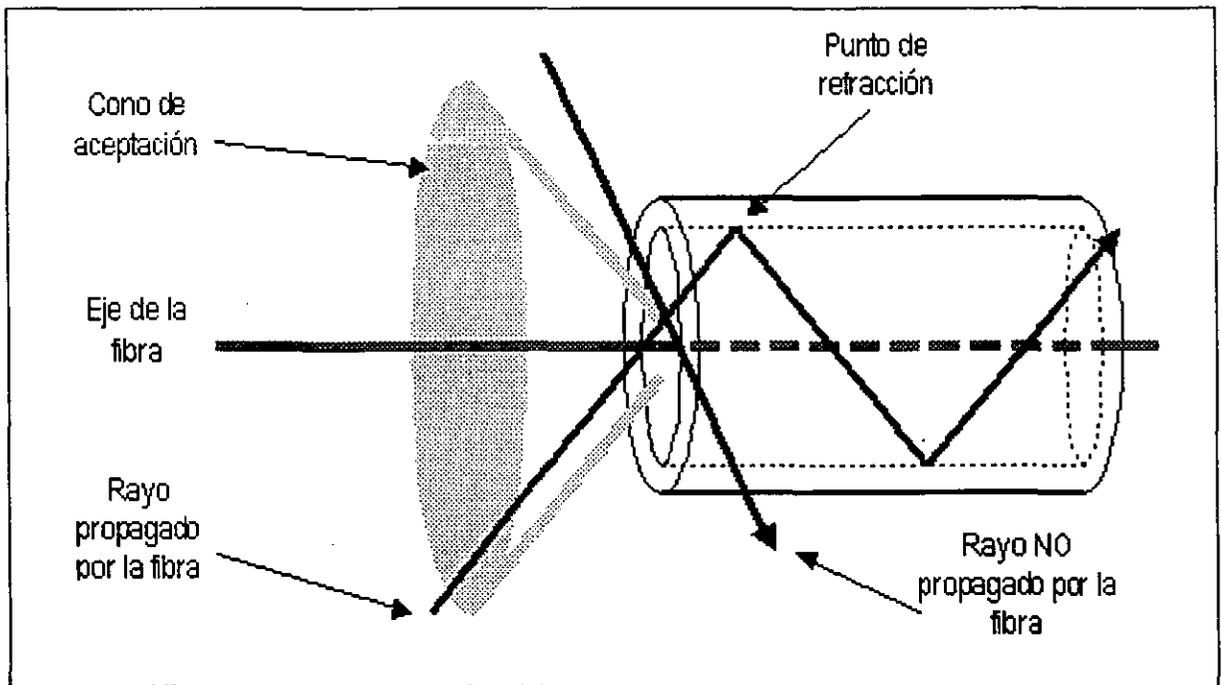


Figura 1.5.3.3 Fibra Óptica.

La ventaja de este medio de transmisión se basa en la frecuencia que tiene la luz, por lo que el ancho de banda es enorme, para un bit con valor 1: un pulso de luz, para un bit con valor 0: bastaría la ausencia de luz. Este sistema no se ve afectado por ningún tipo de interferencia y casi la única desventaja es el hecho de no poder empalmar fácilmente cables para conectarlos a nuevos nodos.

El montaje de fibra óptica está compuesto por un emisor de luz, un detector y un medio transmisor. El emisor de luz puede ser un led o bien un emisor láser, como detector sirve un fotodiodo, que detecta la ausencia o presencia de luz., mientras que el medio de transmisión está formado por una fina fibra de vidrio o silicio.

La velocidad de transmisión es muy alta, desde 10 Mbps hasta en casos especiales 500 Mbps. La longitud del cable viene limitada por las atenuaciones que recibe la señal con la distancia, pudiendo llegar a ser los segmentos de hasta 2,000 metros.

Los medios de transmisión aéreos son :

- **Las Microondas .**

Se le llaman a las ondas de radio que van de una antena parabólica a otra, sirven básicamente para comunicaciones de vídeo o telefónicas. La movilidad que pueden caracterizar estos equipos y el ahorro económico que produce el hecho de no tender cable a cada sitio en que quiera enviarse o recibir la información hace de esta técnica una de las más usadas para comunicaciones móviles.

Como la transmisión por láser o infrarrojos, las microondas, también se ven afectadas por las condiciones atmosféricas

La figura 1.5.3.4 muestra un sistema de radio comunicación vía microondas.

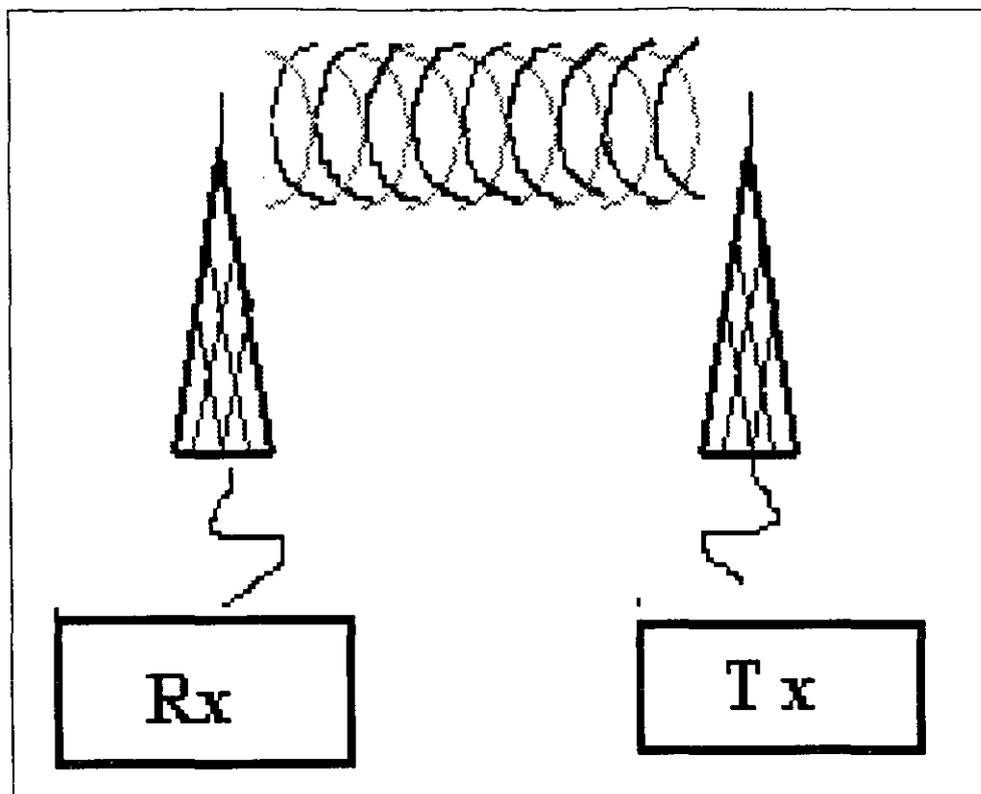


Figura 1.5.3.4 Radio Comunicación Vía Microondas.

- **Comunicaciones por Satélite.**

Los satélites son artefactos en órbitas geosincroestacionarias cuyo lanzamiento es científicamente calculado a fin de que siempre se halle cubriendo una misma porción de suelo terráqueo. La altitud promedio de un satélite es de 35,000 Km. desde la superficie terrestre, con órbitas regulares de 24 horas en la mayoría de los casos al igual que nuestro planeta.

Los satélites varían abundantemente en características como en funciones. Su peso varía entre los 50 y los 2,000 kilos. Tienen capacidades para manipular de forma simultánea, de 250 a 40,000 comunicaciones. Uno de los aspectos más interesantes de los satélites es la increíble cantidad de éstos que giran alrededor de la tierra.

Estructura de los Satélites:

Un satélite está compuesto fundamentalmente por un cuerpo o cilindro, donde se albergan todos sus equipos de control no solo de comunicaciones, sino también de control de navegación. A forma de brazos, se hallan a los lados del cilindro, los paneles solares, siempre dirigidos hacia la luz del sol, fuente de energía para el satélite y todas las funciones que debe cumplir. Apuntando siempre hacia la tierra pueden hallarse una o más antenas de transmisión y recepción de señales. La figura 1.5.3.5 muestra varios satélites.

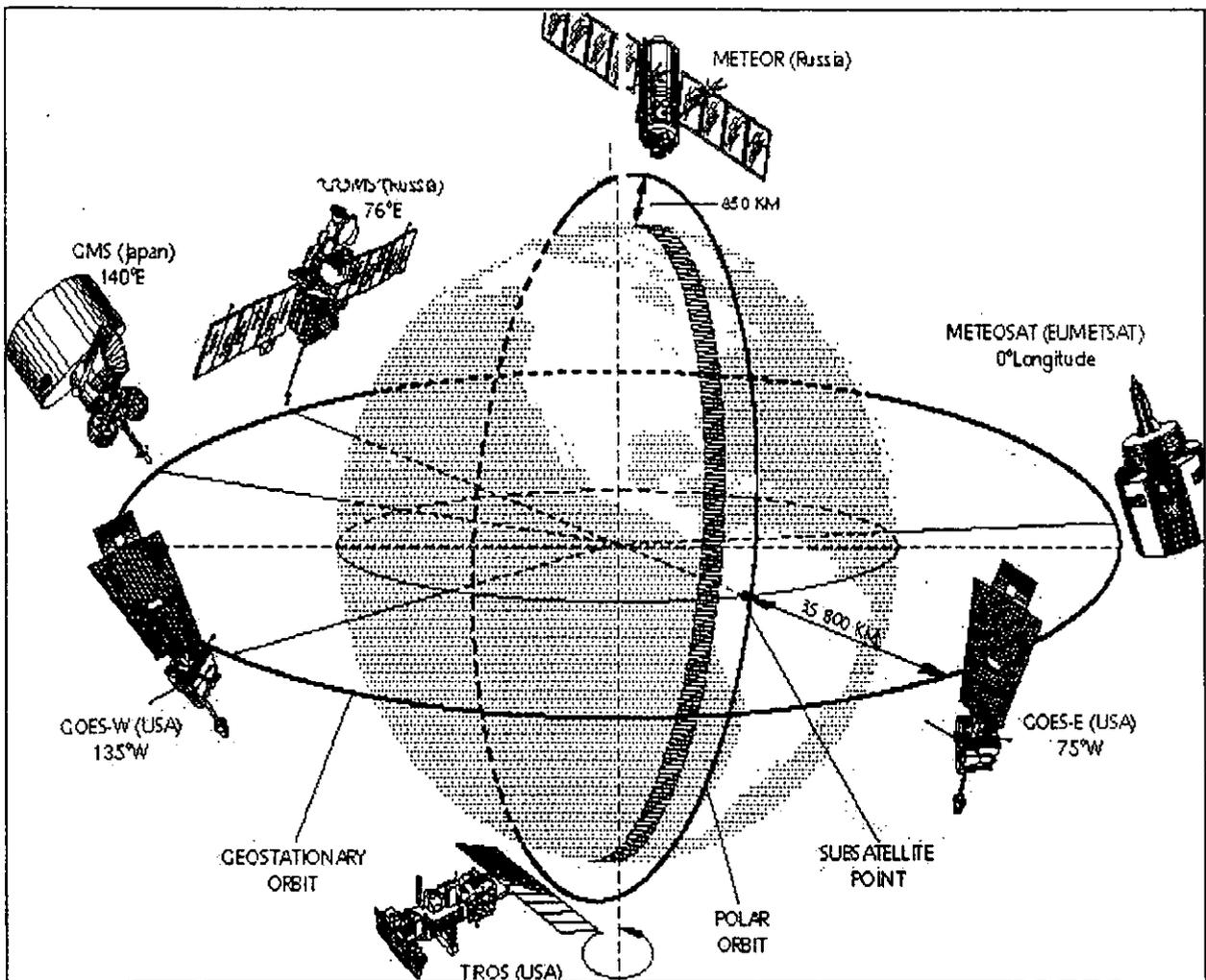


Figura 1.5.3.5 Satélites.

1.5.4 Conmutación y Transmisión.

- **Conmutación.**

Dependiendo de la modalidad con que fluye la información por el interior de la red, existen modalidades de conmutación como son por Circuitos, Mensajes y Paquetes.

1.- Conmutación de Circuitos.

Estas redes establecen un circuito o conexión (física) directa, punto a punto entre los equipos terminales de datos, análogo a una central telefónica. Aquí se requiere un tiempo para el establecimiento de la llamada y a continuación los datos pueden ser transmitidos sin introducción de retardo por los nodos. La conexión establecida es utilizada durante todo el período entre el establecimiento del circuito y el final de la llamada, caso típico: la llamada telefónica.

2.-Conmutación de Mensajes.

La unidad de transmisión es el mensaje, los nodos ubicados en el camino reciben y envían el mensaje por un camino o enlace de salida que esté libre. Si no hay un enlace disponible, el nodo almacena el mensaje en disco para mas tarde enviarlo adelante. El enlace sólo es usado durante la transmisión del mensaje y posibilita compartir enlaces y aumentar el tráfico entre ellos al multiplicar su uso por varios nodos terminales. Posibilita compartir enlaces y aumentar el tráfico en ellos al multiplicar su uso por varios nodos terminales ECD (Equipos terminales de datos).

3.-Conmutación de Paquetes.

Esta tecnología transmite una fuente de destino, bloques de datos denominados paquetes. Estos paquetes se almacenan sólo en la memoria RAM, de los nodos intermedios para continuar rápidamente su viaje hacia el nodo destino.

En esta tecnología, los paquetes de distintos usuarios comparten los mismos medios de transmisión, el tamaño es pequeño, máximo fijo (ejem.: 2 Kbits), un mensaje de tamaño mayor deberá ser segmentado en paquetes.

Un paquete que no puede avanzar es descartado en la red o se impide su entrada para evitar congestión. Sí salen en orden, pero al recorrer caminos distintos, pueden llegar en orden no correcto, el sistema debe reordenarlos. Ampliamente utilizado en redes públicas, al paquete se le agregan varios campos para control en redes tales como: circuito virtual o datagrama, control de flujo, control de errores, secuenciación y norma x.25 de la ISO

- **Transmisión.**

Para la transmisión de información, especialmente en forma digital, se tiene dos tipos:

- Multiplexación por división de frecuencia, donde la banda es dividida en varios canales y cada usuario tiene un canal exclusivo.
- Multiplexación por división de tiempo, donde cada usuario, por turnos, obtiene para su uso, todo el ancho de banda del canal.

1.5.4.1 Protocolos de Arbitraje.

Se denomina así a la posibilidad de transmitir datos por la red; hay dos formas básicas:

1.-CSMA/CD (Carrier Sense Multiple Access with Collission Detection):

En este caso, cualquier máquina puede iniciar una comunicación (acceso múltiple) con sólo verificar que no haya ninguna otra comunicación en el cable; para ello detecta la presencia de portadora (Carrier Sense). La información que se está transmitiendo tarda un cierto tiempo en recorrer la red. Una estación a la que todavía no le llegaron los

primeros bits podría iniciar una transmisión basada en que en ese momento no hay señal. La figura 1.5.4.1 muestra el proceso.

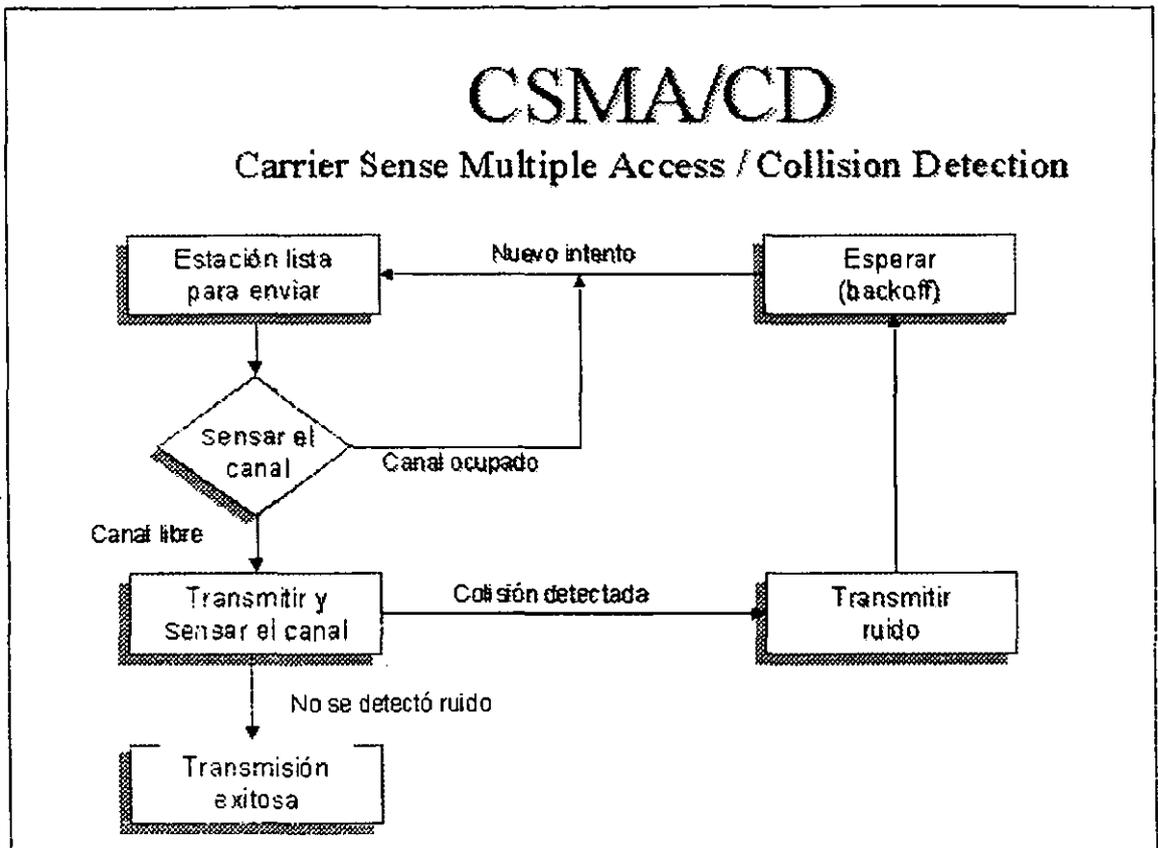


Figura 1.5.4.1 CSMA/CD.

Un instante después le empezarán a llegar dichos bits, pero como la transmisión ya había comenzado, las estaciones comprendidas entre ambas máquinas recibirán la suma de las dos señales. Esto se denomina "colisión". El segundo transmisor debe seguir transmitiendo un tiempo suficiente como para que el primero se entere de la colisión. Esta acción recibe el nombre de atascamiento (jamming).

El peor caso de colisión se produce cuando las estaciones están a la mayor distancia posible y la segunda comienza a transmitir justo antes de recibir el primer bit, pues al tiempo de propagación de la señal de la primera estación a la segunda, hay que

sumarle el de propagación del atascamiento de la segunda a la primera. La suma de esos tiempos define la "ventana de colisión". Para asegurarse la ausencia de colisiones no detectadas, se deben cumplir dos condiciones:

1. La transmisión debe durar más que la ventana de colisiones. Por ej: en Ethernet el paquete mínimo es de 46 bytes y el máximo de 1500 bytes.
2. La estación transmisora debe chequear la ausencia de colisiones durante ese tiempo; después no es necesario.

Una vez detectada la colisión, ambas estaciones deben dejar pasar un tiempo determinado aleatoriamente antes de intentar retransmitir. Si se produce otra colisión, se reintentará esperando un tiempo mayor. El tiempo promedio de demora se duplica con cada reintento. Puede haber colisiones múltiples. Es posible que una estación no pueda comunicarse durante mucho tiempo debido a una sucesión de colisiones.

2.- Token Passing.

Este sistema evita la colisión pues limita el derecho a transmitir a una máquina. Esa máquina se dice que tiene el token. En la figura 1.5.4.2, el token va pasando a intervalos fijos de una máquina a otra. La circulación del token de una máquina a la siguiente hace que, desde el punto de vista lógico, toda la red esté basada en tokens

Debe notarse que un anillo lógico no implica un anillo físico. En efecto, si bien IEEE 802.5 emplea un anillo físico, IEEE 802.4 especifica un bus y ARCnet usa una estrella. Por la red circulan dos tipos de mensajes: los "tokens" y los "frames".

Un token indica que la red está disponible. El token incluye información de prioridad, de forma tal que el control de la red lo pueda tomar sólo una estación con igual o mayor prioridad.

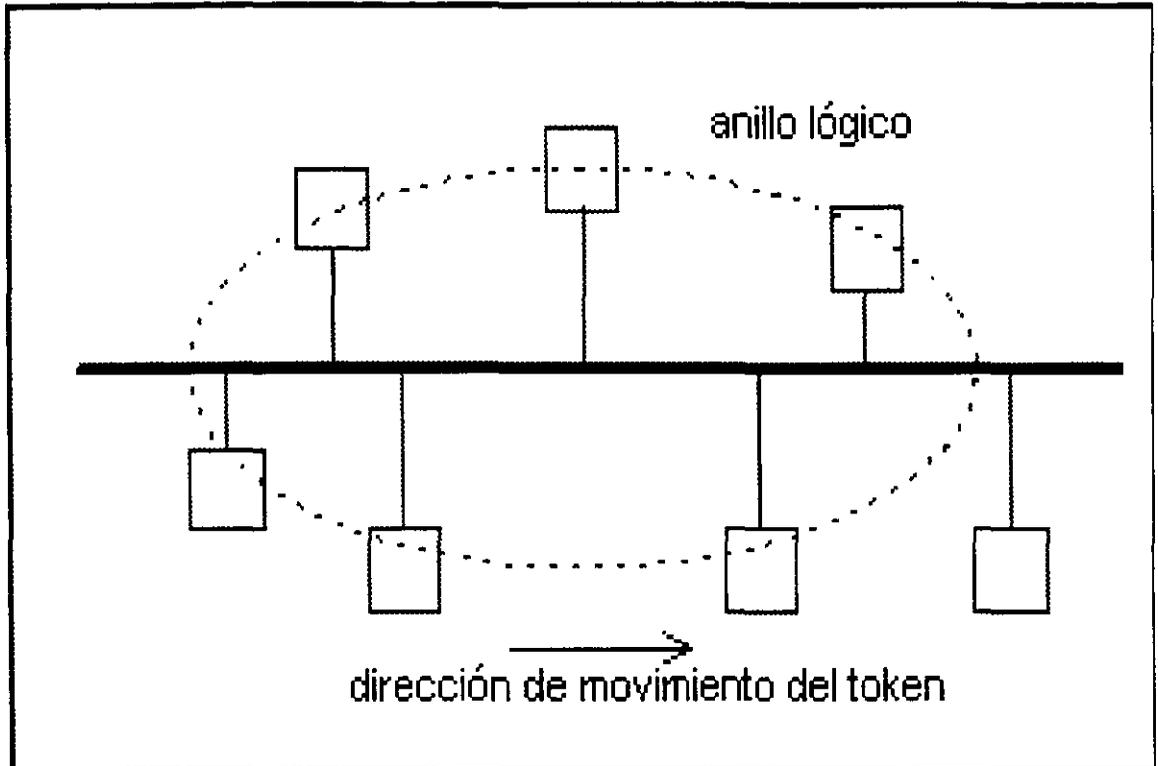


Figura 1.5.4.2 Paso del Token.

Hay un timer que asegura que ninguna estación retenga el token demasiado tiempo. Un frame (marco) es un mensaje que contiene (entre otras cosas) la información que se quiere transmitir, las direcciones de las estaciones transmisora y receptora, y un CRC para manejo de errores.

- **Comparación entre CSMA/CD y Token Passing.**

Ambos tipos de protocolo tienen uso generalizado. La ventaja del primero es que permite mayor performance, especialmente cuando hay pocas colisiones. Esto ocurre si la mayoría de las transmisiones se originan en la misma máquina o si hay relativamente poco tráfico en la red. Una ventaja del segundo es que puede asegurarse que, independientemente del tráfico en la red, una máquina va a poder transmitir antes de un tiempo predeterminado. Esto tiene dos efectos positivos: uno es que el

performance de la red no disminuye tanto al aumentar el tráfico; el otro (aunque su uso es menor) es en sistemas de control donde es importante asegurarse de que un mensaje llegue a destino antes de que pase cierto tiempo. Otra ventaja posible para el segundo es que soporta un esquema de prioridades para el uso de la red.

1.5.5 Series X del CCITT.

En la tabla 1.5.5 se muestra una lista con la serie x del CCITT:

NÚMERO	TÍTULO
x.1	Clases de servicios de usuarios internacionales en redes de datos públicas
x.2	Servicios y facilidades de usuarios internacionales en redes de datos públicas.
x.15	Definiciones de términos concernientes a las redes de datos públicas.
x.20	Interfaces entre equipos terminales de datos (ETD) y equipos terminales de datos ETCD para servicios de transmisión arranque parada en redes públicas de datos.
x.25	Interfaz entre equipos terminales de datos ETD y equipos terminales de circuito de datos (ETCD) para terminales que funcionan en modo paquete en redes de datos públicas

Tabla 1.5.5 Series x del CCITT.

1.5 6 Modelo OSI.

Es importante notar que el modelo OSI (Open System Interconnection), no es una arquitectura de red, debido a que no especifica los servicios exactos y protocolos que deben ser usados en cada capa. Sólo dice lo que cada capa debe hacer. Este modelo

fue creado por el ISO (Organización Internacional de Normalización). En la figura 1.5.6 se muestran las 7 capas de este modelo, las cuales se describen a continuación:

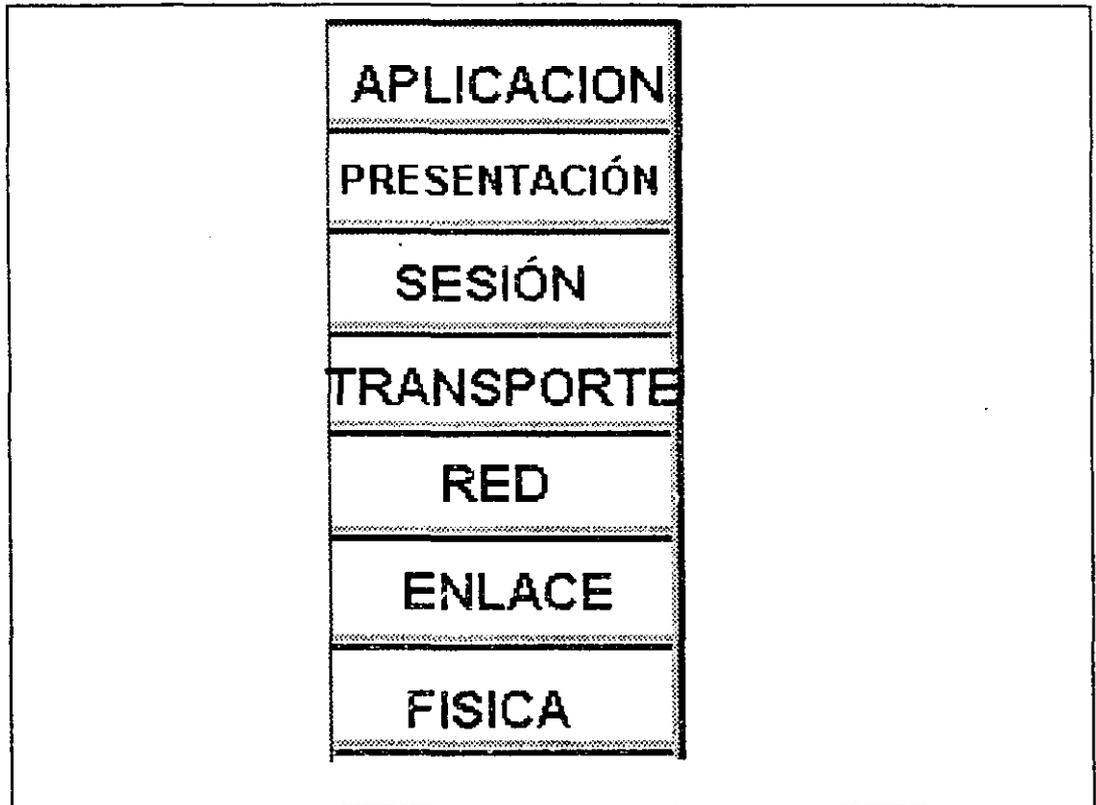


Figura 1.5 6 Modelo OSI.

- 1. Capa Física.** Se encarga de los aspectos físicos de la conexión, tales como el medio de transmisión flujo de bits a través del medio. No existe estructura alguna. Maneja voltajes y pulsos eléctricos. Especifica cables, conectores y componentes de interfaz con el medio de transmisión.
- 2. Capa Enlace de Datos** Estructura el flujo de bits bajo un formato predefinido llamado trama. Para formar una trama, el nivel de enlace agrega una secuencia especial de bits al principio y al final del flujo inicial de bits. Transfiere tramas de una forma confiable libre de errores (utiliza reconocimientos y-retransmisión de tramas). Provee control de flujo.

3. **Capa de Red.** Nivel encargado de encaminar los datos hacia su destino eligiendo la ruta más efectiva. Divide los mensajes de la capa de transporte en paquetes y los ensambla al final.
4. **Transporte.** Transporta la información de una manera fiable para que llegue correctamente a su destino.
5. **Capa de Sesión.** Permite a usuarios en diferentes máquinas establecer una sesión. Una sesión puede ser usada para efectuar un login a un sistema de tiempo compartido remoto, para transferir un archivo entre dos máquinas, etc. Controla el diálogo (quién habla, cuándo, cuánto tiempo, half duplex o full duplex). Función de sincronización.
6. **Capa de Presentación.** Establece una sintaxis y semántica de la información transmitida. Se define la estructura de los datos a transmitir (v.g. define los campos de un registro: nombre, dirección, teléfono, etc.) Define el código a usar para representar una cadena de caracteres (ASCII, EBCDIC, etc.), compresión de datos y criptografía.
7. **Capa de Aplicación.** Transferencia de archivos (ftp), login remoto (rlogin, telnet), correo electrónico (mail), acceso a bases de datos, etc.

1.5.7 Arquitecturas.

1.5.7.1 Cliente – Servidor.

El término cliente/servidor describe un sistema en el que una máquina cliente solicita a una segunda máquina llamada servidor que ejecute una tarea específica, el cliente se encarga de efectuar una petición o solicitar un servicio. El cliente no posee control sobre los recursos, sino que es el servidor el encargado de manejarlos. Por otro lado, el ordenador remoto que actúa como servidor evalúa la petición del cliente y decide aceptarla o rechazarla consecuentemente.

El servidor es, por lo general, una máquina anfitriona, como un servidor de archivos PC, un servidor de archivos de UNIX o una macrocomputadora. El programa cliente

cumple dos funciones distintas: por un lado gestiona la comunicación con el servidor, solicita un servicio y recibe los datos enviados por aquél. Por otro, maneja la interfaz con el usuario: presenta los datos en el formato adecuado y brinda las herramientas y comandos necesarios para que el usuario pueda utilizar las prestaciones del servidor de forma sencilla. El programa servidor en cambio, básicamente sólo tiene que encargarse de transmitir la información de forma eficiente. No tiene que atender al usuario. De esta forma un mismo servidor puede atender a varios clientes al mismo tiempo.

Una vez que el servidor acepta el pedido, la información requerida es suministrada al cliente que efectuó la petición, siendo este último el responsable de proporcionar los datos al usuario con el formato adecuado. Se puede precisar que cliente y servidor no tienen que estar necesariamente en ordenadores separados, sino que pueden ser programas diferentes que se ejecuten en el mismo ordenador.

Cada computadora se destina para una tarea específica, los resultados de esta computadora podrán ser utilizados por alguna otra máquina conectada a la red; esto condujo al modelo "cliente/servidor", modelo al que responde la mayoría de las aplicaciones en Internet. Servidor es el sistema que provee un servicio específico al resto, Cliente es otro sistema que usa ese servicio.

1.5.7.2 Arquitectura TCP/IP.

Las arquitecturas basadas en TCP/IP proponen cuatro capas en las que las funciones de las capas de Sesión y Presentación son responsabilidad de la capa de Aplicación y las capas de Enlace de Datos y Física son vistas como la capa de Interface a la Red. Por tal motivo para TCP/IP sólo existen las capas Interface de Red, la de Intercomunicación en Red, la de Transporte y la de Aplicación.

Como puede verse, TCP/IP presupone independencia del medio físico de comunicación, sin embargo existen estándares bien definidos a nivel de Enlace de Datos y Físico que proveen mecanismos de acceso a los diferentes medios y que en el

modelo TCP/IP deben considerarse la capa de Interface de Red; siendo los más usuales el proyecto IEEE802, Ethernet, Token Ring y FDDI.

1.5.7.3 Arquitectura de Interconexión de Redes en TCP/IP.

Las características principales son: Protocolos de no conexión en el nivel de red, conmutación de paquetes entre nodos, protocolos de transporte con funciones de seguridad, conjunto común de programas de aplicación e interconexión de redes. Para entender el funcionamiento de los protocolos TCP/IP debe tenerse en cuenta la arquitectura que ellos proponen para comunicar redes. La arquitectura ve como iguales a todas las redes a conectarse, sin tomar en cuenta el tamaño de ellas, ya sean locales o de cobertura amplia, como se muestra en la figura 1.5.7.3.

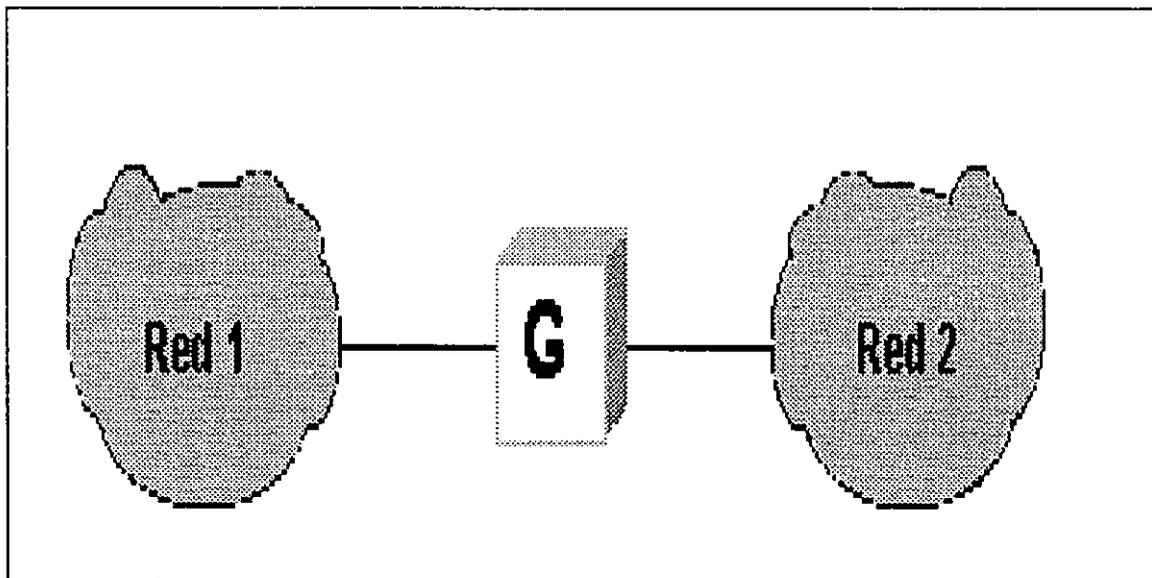


Figura 1.5.7.3 Interconexión entre dos Redes.

Define que todas las redes que intercambiarán información deben estar conectadas a una misma computadora o equipo de procesamiento (dotados con dispositivos de comunicación); a tales computadoras se les denomina compuertas, pudiendo recibir otros nombres como enrutadores o puentes.

1.5.7.4 Descripción del Modelo de Capas de TCP/IP.

En la figura 1.5.7.4 se muestran las capas del modelo TCP/IP y su relación con el modelo OSI. Para TCP/IP sólo existen cuatro capas: Interface de Red, Intercomunicación en Red o Capa Internet, Capa de Transporte y Capa de Aplicación. Estas capas se describen a continuación:

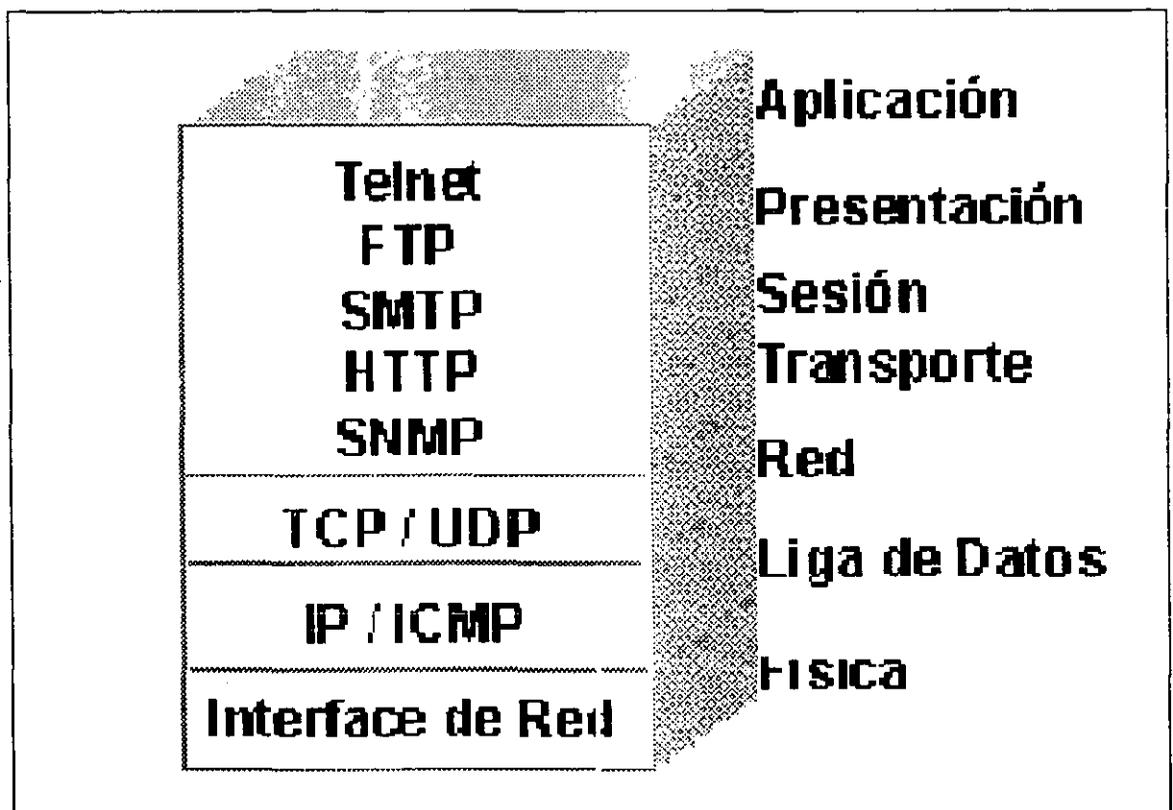


Figura 1.5.7.4 Modelo de Capas TCP/IP y su Relación con el Modelo OSI.

Capa de Aplicación. Invoca programas que acceden servicios en la red e interactúa con uno o más protocolos de transporte para enviar o recibir datos en forma de mensajes o bien en forma de flujos de bytes.

Capa de Transporte. Provee comunicación extremo a extremo desde un programa de aplicación a otro. Regula el flujo de información y puede proveer un transporte confiable asegurándose que los datos lleguen sin errores y en la secuencia correcta.

Coordina a múltiples aplicaciones que se encuentren interactuando con la red simultáneamente de tal manera que los datos que envíe una aplicación sean recibidos correctamente por la aplicación remota, esto lo hace añadiendo identificadores de cada una de las aplicaciones. Realiza además una verificación por suma para asegurar que la información no sufrió alteraciones durante su transmisión.

Capa Internet. Controla la comunicación entre un equipo y otro y decide qué rutas deben seguir los paquetes de información para alcanzar su destino. Conformar los paquetes IP que serán enviados por la capa inferior. Desencapsula los paquetes recibidos pasando a la capa superior la información dirigida a una aplicación.

Capa de Interface de Red. Emite al medio físico los flujos de bits y recibe los que de él provienen. Consiste en los manejadores de los dispositivos que se conectan al medio de transmisión.

1.5.8 Protocolos.

Los protocolos de comunicaciones definen las normas que posibilitan que se establezca una comunicación entre varios equipos o dispositivos, ya que estos equipos pueden ser diferentes entre sí. Una interfaz, sin embargo, es la encargada de la conexión física entre los equipos, definiendo las normas para las características eléctricas y mecánicas de la conexión. Exceptuando a los routers, cualquier ordenador conectado a Internet y, por tanto, capaz de compartir información con otro ordenador se conoce con el nombre de host (anfitrión). Un host debe identificarse de alguna manera que lo distinga de los demás para poder recibir o enviar datos.

Para ello todos los ordenadores conectados a Internet disponen de una dirección única y exclusiva. Esta dirección, conocida como dirección de Internet o dirección IP, es un número de 32 bits que generalmente se representa en cuatro grupos de 8 bits cada uno separados por puntos y en base decimal (esto es así en la versión número 4 del protocolo IP, pero no en la 6). Un ejemplo de dirección IP es el siguiente: 205.198.48.1.

1.5.9 Suite de Protocolos TCP / IP.

1.5.9.1 Protocolo TCP/IP.

Este modelo asume que hay un gran número de redes independientes conectadas a su vez por un "gateway".

El usuario debe ser capaz de acceder a una computadora o recurso a través de su "gateway" en cualquiera de estas redes. Los datagramas a menudo pasarán a través de una docena de redes diferentes antes de llegar a su destino final.

TCP/IP no es un único protocolo, sino que es en realidad lo que se conoce con este nombre un conjunto de protocolos que cubren los distintos niveles del modelo OSI. Los dos protocolos más importantes son el TCP (Transmission Control Protocol) y el IP (Internet Protocol), que son los que dan nombre al conjunto. Para transmitir información a través de TCP/IP, ésta debe ser dividida en unidades de menor tamaño. Esto proporciona grandes ventajas en el manejo de los datos que se transfieren y, por otro lado, esto es algo común en cualquier protocolo de comunicaciones.

En TCP/IP cada una de estas unidades de información recibe el nombre de "datagrama" (datagram), y son conjuntos de datos que se envían como mensajes independientes.

1.5.9.2 Protocolo de Transferencia de Archivos (FTP).

El protocolo FTP (File Transfer Protocol) se incluye como parte del TCP/IP, siendo éste el protocolo de nivel de aplicación destinado a proporcionar el servicio de transferencia de archivos en Internet. El FTP depende del protocolo TCP para las funciones de transporte, y guarda alguna relación con TELNET (protocolo para la conexión remota).

El protocolo FTP permite acceder a algún servidor que disponga de este servicio y realizar tareas como moverse a través de su estructura de directorios, ver y descargar

ficheros al ordenador local, enviar ficheros al servidor o copiar archivos directamente de un servidor a otro de la red. Lógicamente y por motivos de seguridad se hace necesario contar con el permiso previo para poder realizar todas estas operaciones. El servidor FTP pedirá el nombre de usuario y clave de acceso al iniciar la sesión (login), que debe ser suministrado correctamente para utilizar el servicio.

El FTP usa dos canales TCP. El puerto 20 del TCP es el canal de datos y el puerto 21 es el canal de comandos. El FTP observa el proceso de transferencia en tiempo real. Al usar el TCP, el FTP elimina la necesidad de preocuparse por la confiabilidad o la administración de la conexión, debido a que el FTP puede basarse en TCP para ejecutar estas funciones en forma adecuada. En lenguaje FTP los dos canales que existen entre las dos máquinas se llaman **protocol interpreter** (intérprete de protocolo) o PI y **data transfer process** (proceso de transferencia de datos) o DTP. El PI transfiere instrucciones entre las dos aplicaciones usando el canal 21 de comandos TCP y DTP transfiere datos en el canal 20 de datos TCP.

El FTP proporciona dos modos de transferencia de ficheros: ASCII y binario. El modo de transferencia ASCII se utiliza cuando se quiere transmitir archivos de texto, ya que cada sistema puede utilizar un formato distinto para la representación de texto. En este caso se realiza una conversión en el formato del fichero original, de manera que el fichero recibido pueda utilizarse normalmente. Cuando se establece una conexión por el FTP el proceso es el siguiente:

1. Registro: Verifica la ID del usuario y contraseña
2. Definir directorio de inicio.
3. Definir el modo de transferencia de archivos.
4. Iniciar la transferencia de datos: permite los comandos de usuario.
5. Detener la transferencia de datos: cierra la conexión.

La figura 1.5.9.2 muestra la conexión del canal FTP.

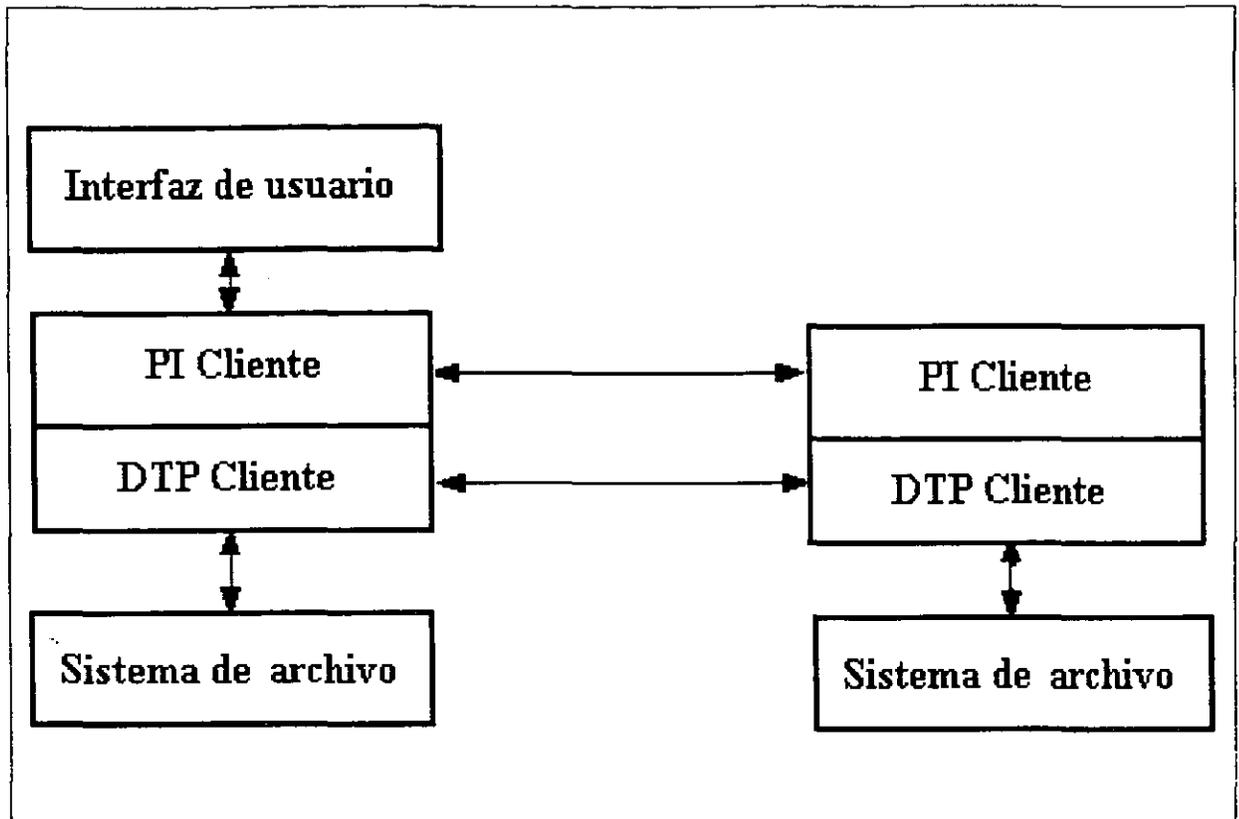


Figura 1.5.9.2. Conexión del Canal FTP.

1.5.9.3 Protocolo TFTP.

TFTP es un protocolo para transferir ficheros entre distintas máquinas conectadas a través de una red de comunicaciones. Difiere del FTP de dos maneras principales: no se registra en la máquina remota y usa un protocolo de transporte no orientado a conexión en lugar del TCP.

Consiste fundamentalmente en la lectura o escritura por parte de un cliente a un archivo de un servidor, pero no supervisa el progreso de la transferencia de archivos, aunque asegura la integridad de los datos. Es útil para cargar aplicaciones y fuentes así como para el arranque.

Sus características fundamentales son:

- Cualquier transferencia de ficheros comienza con una solicitud de lectura o escritura de un fichero por parte de un cliente. Si el servidor acepta dicha solicitud el fichero se transmite dividido en paquetes o tramas de un tamaño fijo de 512 bytes.
- Cada paquete de datos contiene uno de esos paquetes o tramas, y debe ser confirmado por un paquete de asentimiento antes de que se pueda enviar el siguiente paquete de datos. Un paquete de datos de menos de 512 bytes indica el fin de la transferencia.
- Cada paquete de datos lleva consigo un número de trama, comenzando la transferencia por la trama 1. Cada paquete de asentimiento lleva consigo el número de trozo que se asiente.

Si un paquete se pierde en la comunicación, a su destinatario le vencerá un plazo y deberá retransmitir el último paquete transmitido, lo que causará que el emisor del paquete perdido retransmita dicho paquete. Nótese que utilizan los plazos tanto el cliente como el servidor.

1.5.9.4 Protocolo de Transferencia de Correo Simple (SMTP).

El servicio de correo electrónico se proporciona a través del protocolo SMTP (Simple Mail Transfer Protocol), y permite enviar mensajes a otros usuarios de la red. A través de estos mensajes no sólo se puede intercambiar texto, sino también archivos binarios de cualquier tipo.

La transmisión de datos SMTP usa un formato sencillo. Todo el texto del mensaje se transfiere como caracteres ASCII de 7 bits. La tabla 1.5.9.4 muestra el conjunto de comandos del protocolo SMTP.

El fin del mensaje lo indica un solo punto en la línea, si por alguna razón una línea en el mensaje comienza con un punto, el protocolo agrega un segundo punto para evitar confusión con el indicador de fin de mensaje. Cuando se establece una conexión, los dos sistemas SMTP establecen códigos de autenticación.

COMANDO	DESCRIPCIÓN
DATA	Texto de mensaje.
EXPN	Expansión de una lista de distribución.
HELLO	Se usa en el establecimiento de la conexión para intercambiar identificadores.
HELP	Solicita ayuda.
MAIL	La dirección del transmisor.
NOOP	Ninguna operación.
RCPT	La dirección de destino de mensaje (puede proporcionarse más de una).
RSET	Termina la transacción actual.
SAML	Envía un mensaje a la terminal de usuario y envía correo.
SEND	Envía un mensaje a la terminal de usuario.
SOML	Envía un mensaje a la terminal de usuario o envía correo.
TURN	Cambia la dirección de envío (invierte los papeles de transmisor y receptor).
VERFY	Verifica el nombre de usuario.

Tabla 1.5.9.4 Conjunto de Comandos del Protocolo SMTP.

Se envía un RCPT (Dirección destino del mensaje), para identificar el receptor, si se identifica más de un receptor en la ubicación del receptor se envían varios mensajes RCPT, pero el mensaje en sí solo se transmite una sola vez. Después de cada RCPT hay un acuse de recibo. A un comando DATA le siguen las líneas de mensaje hasta que un solo punto en una línea indica el fin del mensaje. La conexión se cierra con el comando QUIT.

1.5.9.5 Protocolo Telnet.

El protocolo diseñado para proporcionar el servicio de conexión remota (remote login) recibe el nombre de TELNET, el cual forma parte del conjunto de protocolos TCP/IP y depende del protocolo TCP para el nivel de transporte.

El Protocolo TELNET es un emulador de terminal que permite acceder a los recursos y ejecutar los programas de un ordenador remoto en la red, de la misma forma que si se tratara de un terminal real directamente conectado al sistema remoto.

Una vez establecida la conexión, el usuario podrá iniciar la sesión con su clave de acceso. De la misma manera que ocurre con el protocolo FTP, existen servidores que permiten un acceso libre cuando se especifica "anonymous" como nombre de usuario.

Es posible ejecutar una aplicación cliente TELNET desde cualquier sistema operativo, pero hay que tener en cuenta que los servidores suelen ser sistemas VMS o UNIX por lo que, a diferencia del protocolo FTP para transferencia de ficheros donde se utilizan ciertos comandos propios de esta aplicación, los comandos y sintaxis que se utilicen en TELNET deben ser los del sistema operativo del servidor.

El sistema local que utiliza el usuario se convierte en un terminal "no inteligente" donde todos los caracteres pulsados y las acciones que se realicen se envían al host remoto, el cual devuelve el resultado de su trabajo. Para facilitar un poco la tarea a los usuarios, en algunos casos se encuentran desarrollados menús con las distintas opciones que se ofrecen.

El protocolo Telnet usa el concepto de network virtual terminal (terminal virtual de red), o NVT para definir ambos extremos de una conexión Telnet, cada NVT tiene una impresora y un teclado lógicos, la impresora lógica puede desplegar caracteres y el teclado lógico puede generar caracteres. La impresora lógica por lo general es una pantalla terminal, en tanto que el teclado lógico es el usuario, aunque podría ser un archivo u otro flujo de entrada. El protocolo trata a los dos extremos de la conexión como NVT. La figura 1.5.9.5 muestra una terminal virtual de red para Telnet.

Los programas clientes de TELNET deben ser capaces de emular los terminales en modo texto más utilizados para asegurarse la compatibilidad con otros sistemas, lo que incluye una emulación del teclado.

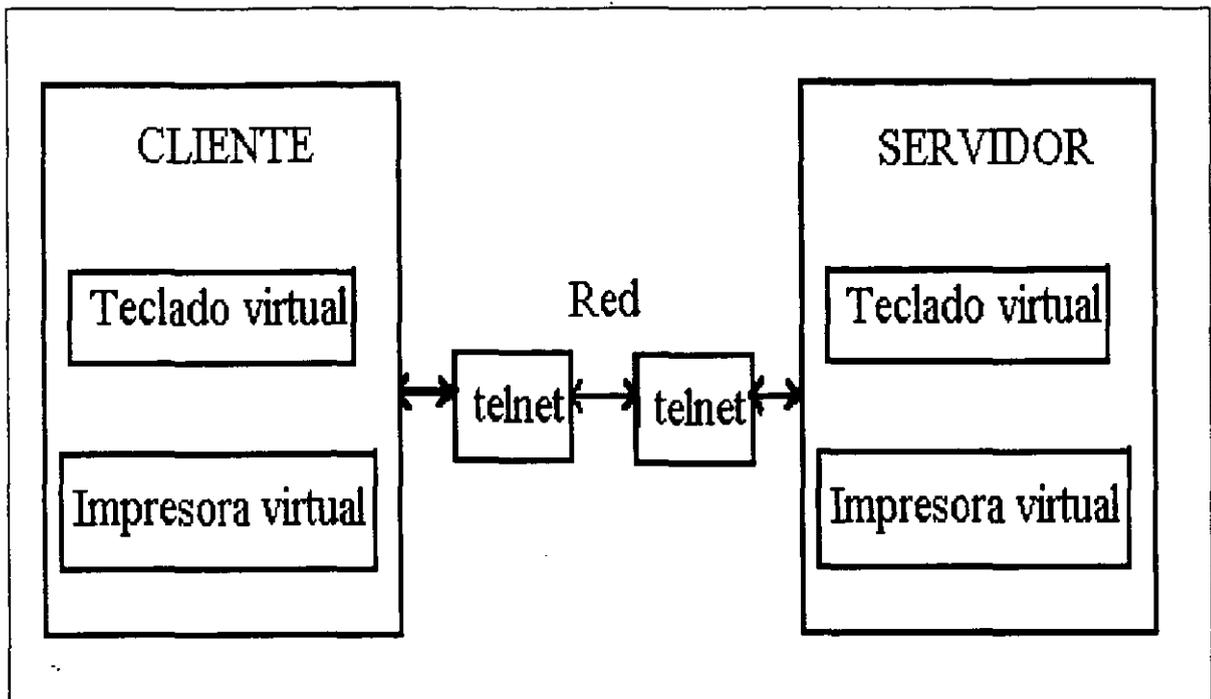


Figura 1.5.9.5 Terminal Virtual.

1.5.9.6 Protocolo IP.

A continuación se describirán los campos para el formato del mensaje IP mostrado en la figura 1.5.9.6 :

- **Versión:** Número de versión del protocolo IP utilizado. Tendrá que tener el valor 4. *Tamaño: 4 bits.*
- **Longitud de la cabecera:** (*Internet Header Length, IHL*) Especifica la longitud de la cabecera expresada en el número de grupos de 32 bits que contiene. *Tamaño: 4 bits.*
- **Tipo de servicio:** El tipo o calidad de servicio se utiliza para indicar la prioridad o importancia de los datos que se envían, lo que condicionará la forma en que éstos serán tratados durante la transmisión. *Tamaño: 8 bits.*
- **Longitud total:** Es la longitud en bytes del datagrama completo, incluyendo la cabecera y los datos. Como este campo utiliza 16 bits, el tamaño máximo del

datagrama no podrá superar los 65,535 bytes, aunque en la práctica este valor será mucho más pequeño. *Tamaño: 16 bits.*

0	8	16	32
Versión	IHL	Tipo de servicio	Longitud total
Identificación		Flags	Desplazamiento del fragmento
Tiempo de vida	Protocolo	Comprobación	
Dirección de Origen			
Dirección Destino			

Figura 1.5.9.6 Formato del mensaje IP.

- **Identificación:** Valor de identificación que se utiliza para facilitar el ensamblaje de los fragmentos del datagrama. *Tamaño: 16 bits.*
- **Flags:** Indicadores utilizados en la fragmentación. *Tamaño: 3 bits.*
- **Fragmentación:** Contiene un valor (*offset*) para poder ensamblar los datagramas que se hayan fragmentado. Está expresado en número de grupos de 8 bytes (64 bits), comenzando con el valor cero para el primer fragmento. *Tamaño: 16 bits.*
- **Límite de existencia:** Contiene un número que disminuye cada vez que el paquete pasa por un sistema. Si este número llega a cero, el paquete será descartado. Esto es necesario por razones de seguridad para evitar un bucle infinito, ya que aunque es bastante improbable que esto suceda en una red correctamente diseñada, no debe descuidarse esta posibilidad. *Tamaño: 8 bits.*
- **Protocolo:** El número utilizado en este campo sirve para indicar a qué protocolo pertenece el datagrama que se encuentra a continuación de la cabecera IP, de

manera que pueda ser tratado correctamente cuando llegue a su destino. *Tamaño: 8 bits.*

- **Comprobación:** El campo de comprobación (*checksum*) es necesario para verificar que los datos contenidos en la cabecera IP son correctos. Por razones de eficiencia este campo no puede utilizarse para comprobar los datos incluidos a continuación, sino que estos datos de usuario se comprobarán posteriormente a partir del campo de comprobación de la cabecera siguiente, y que corresponde al nivel de transporte. Este campo debe calcularse de nuevo cuando cambia alguna opción de la cabecera, como puede ser el límite de existencia. *Tamaño: 16 bits.*
- **Dirección de origen:** Contiene la dirección del *host* que envía el paquete. *Tamaño: 32 bits.*
- **Dirección de destino:** Esta dirección es la del *host* que recibirá la información. Los *routers* o *gateways* intermedios deben conocerla para dirigir correctamente el paquete. *Tamaño: 32 bits.*

1.5.9.7 Protocolo de Control de Transmisión (TCP).

El protocolo de control de transmisión (TCP) pertenece al nivel de transporte, siendo el encargado de dividir el mensaje original en datagramas de menor tamaño, y por lo tanto, mucho más manejables. Los datagramas serán dirigidos a través del protocolo IP de forma individual. El protocolo TCP se encarga además de añadir cierta información necesaria a cada uno de los datagramas. Esta información se añade al inicio de los datos que componen el datagrama en forma de cabecera.

La cabecera de un datagrama contiene al menos 160 bits que se encuentran repartidos en varios campos con diferente significado. Cuando la información se divide en datagramas para ser enviados, el orden en que éstos lleguen a su destino no tiene que ser el correcto.

Cada uno de ellos puede llegar en cualquier momento y con cualquier orden, e incluso puede que algunos no lleguen a su destino o lleguen con información errónea. Para

evitar todos estos problemas el TCP numera los datagramas antes de ser enviados, de manera que sea posible volver a unirlos en el orden adecuado.

Esto permite también solicitar de nuevo el envío de los datagramas individuales que no hayan llegado o que contengan errores, sin que sea necesario volver a enviar el mensaje completo.

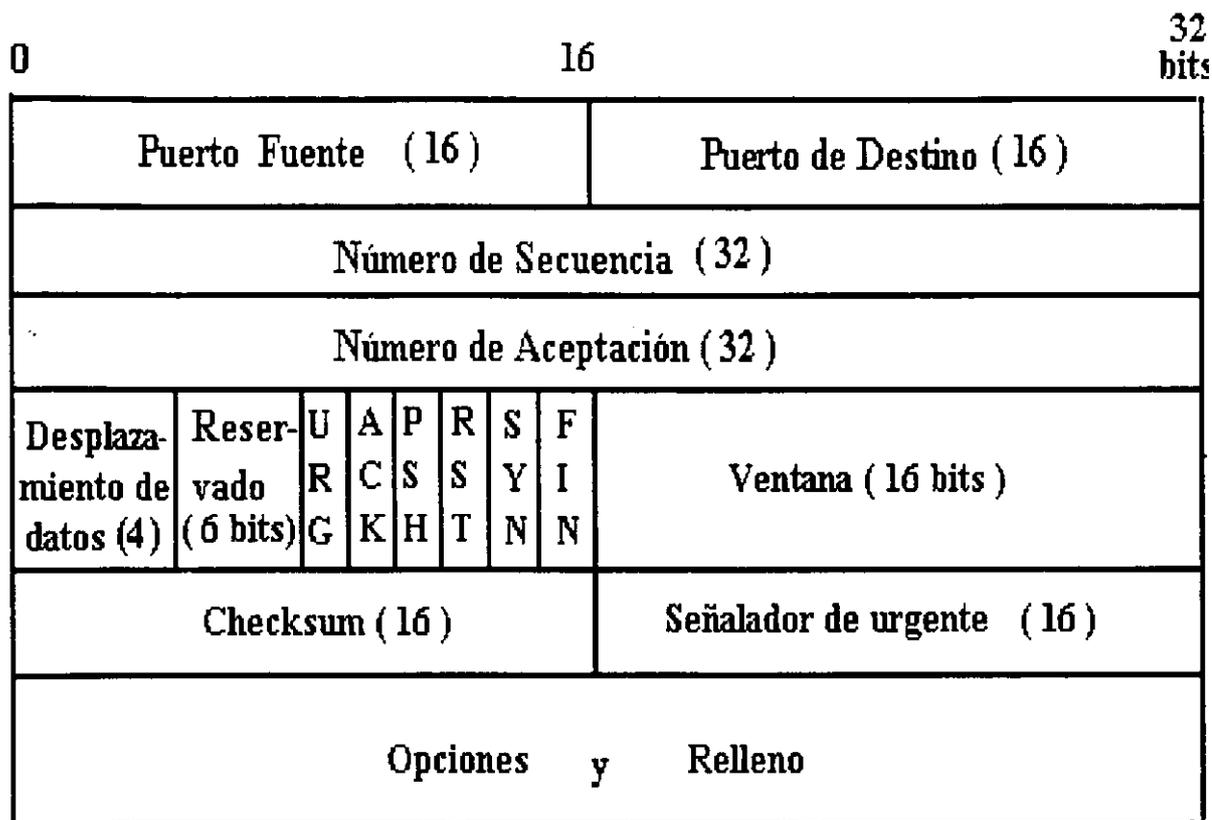


Figura 1.5.9.7 Formato del Mensaje TCP.

El segmento TCP se divide en dos partes, la parte de la cabecera y la parte de datos, la parte de datos sigue a la parte de la cabecera. Los primeros dos campos del segmento se denominan puerto fuente y puerto destino.

Estos campos de 16 bits identifican a los programas de aplicación de nivel superior que utilizan la conexión TCP. El resto de los campos se describen a continuación:

- **Número de secuencia.** Contiene el número de secuencia de la posición del bloque actual en el mensaje total. Dentro del segmento especifica el primer octeto de datos de usuario, así mismo se utiliza durante la operación de gestión de la conexión. Este número se usa también entre dos implementaciones TCP para proporcionar el número de secuencia de envío inicial.
- **Número de aceptación.** Permite aceptar los datos previamente recibidos. Este campo contiene el número de secuencia del siguiente octeto que se espera recibir del transmisor. Con esta definición permite la aceptación inclusiva, en el sentido que se permite la aceptación de todos los octetos hasta, e incluyendo, el valor de este número menos 1.
- **Desplazamiento de datos.** Especifica el número de palabras alineadas de 32 bits que están en el encabezado TCP. Este campo se usa para identificar el inicio del campo de datos.
- **Reservado.** Es un campo de 6 bits reservado para uso futuro, estos bits deben fijarse en valor de 0. Los seis bits siguientes se denominan indicadores (flags). Son bits de control TCP y se utilizan para usar ciertos servicios o utilidades que se pueden emplear durante la sesión. El valor de algunos de estos bits indica cómo interpretar otros campos de la cabecera. Los seis bits mencionados llevan la siguiente información si están activos (en valor de 1).
 - **URG.** Indica que el campo del puntero de urgencia es significativo.
 - **ACK.** Indica si el campo de aceptación es significativo.
 - **PSH.** Significa que el módulo va a utilizar la función push.
 - **RST.** Indica que la conexión se va a inicializar.
 - **SYN.** Indica que los números de secuencia deben de sincronizarse; se utiliza en los segmentos de establecimiento de conexión como indicación de que se va a realizar algunas operaciones de preparación.
 - **FIN.** Indica que el transmisor no tiene más datos que enviar. Éste es el equivalente de un marcador de fin de la transmisión.
- **Ventana.** Un número que indica cuántos bloques de datos puede aceptar la máquina receptora.

- **Checksum.** Contiene el complemento a 1 de 16 bits del complemento a 1 de la suma de todas las palabras de 16 bits del segmento, incluyendo la cabecera y el texto. El propósito de este cálculo es determinar si el segmento procedente del transmisor ha llegado libre de errores.
- **Puntero de urgente.** ó señalador de urgente. Se usa si el indicador de URG está puesto a 1, indica la parte del mensaje de datos que es urgente especificar la compensación del número de secuencia en el encabezado.
- **Opciones.** Similar al campo de opciones del encabezado IP, éste se usa para especificar opciones TCP. Cada opción consta de un número de opción (un byte), el número de bytes en esta área y los valores de la opción. Están definidas en tres opciones para el TCP como muestra la tabla 1.5.9.7.
- **Relleno.** Asegura que el encabezado es un múltiplo de 32 bits.

VALOR	DESCRIPCION
0	Fin de la lista de operaciones.
1	No operación.
2	Tamaño máximo del segmento.

Tabla 1.5.9.7 Opciones TCP.

1.5.9.8 Protocolo UDP (User Datagram Protocol).

El protocolo de datagrama de usuario (UDP) puede ser la alternativa al TCP en algunos casos en los que no sea necesario el gran nivel de complejidad proporcionado por el TCP.

Puesto que UDP no admite numeración de los datagramas, éste protocolo se utiliza principalmente cuando el orden en que se reciben los mismos no es un factor fundamental, o también cuando se quiere enviar información de poco tamaño que cabe en un único datagrama.

Cuando se utiliza UDP la garantía de que un paquete llegue a su destino es mucho menor que con TCP debido a que no se utilizan las señales de confirmación. Por todas estas características la cabecera del UDP es bastante menor en tamaño que la de TCP. Esta simplificación resulta en una mayor eficiencia en determinadas ocasiones.

Un ejemplo típico de una situación en la que se utiliza el UDP es cuando se pretende conectar con un ordenador de la red, utilizando para ello el nombre del sistema. Este nombre tendrá que ser convertido a la dirección IP que le corresponde y, por tanto, tendrá que ser enviado a algún servidor que posea la base de datos necesaria para efectuar la conversión. En este caso es mucho más conveniente el uso de UDP.

Formato de un Mensaje UDP.

Este formato es muy simple e incluye los siguientes campos representados en la figura 1.5.9.8:

- **Puerto de fuente.** Este valor identifica el puerto del proceso de aplicación remitente, este campo es opcional, si no se utiliza se pone 0.
- **Puerto de destino.** Este valor identifica el proceso de recepción en el computador de destino.
- **Longitud.** Este valor indica la longitud del datagrama de usuario, incluyendo la cabecera y los datos, la longitud mínima es de 8 octetos.
- **Checksum.** Este valor contiene el valor del complemento a 1 en 16 bits del complemento a 1 de la suma de la pseudocabecera de IP, la cabecera de UDP y los datos. Se realiza también el checksum de los campos de relleno (Si es necesario que el mensaje contenga un número de octetos que sea múltiplo de dos).
- **Datos.** Campo para los datos a transmitir.

Representa el nivel de servicio mínimo que utilizan muchos sistemas de aplicación basados en transacciones, es sin embargo muy útil en los casos que no son necesarios los servicios de TCP.

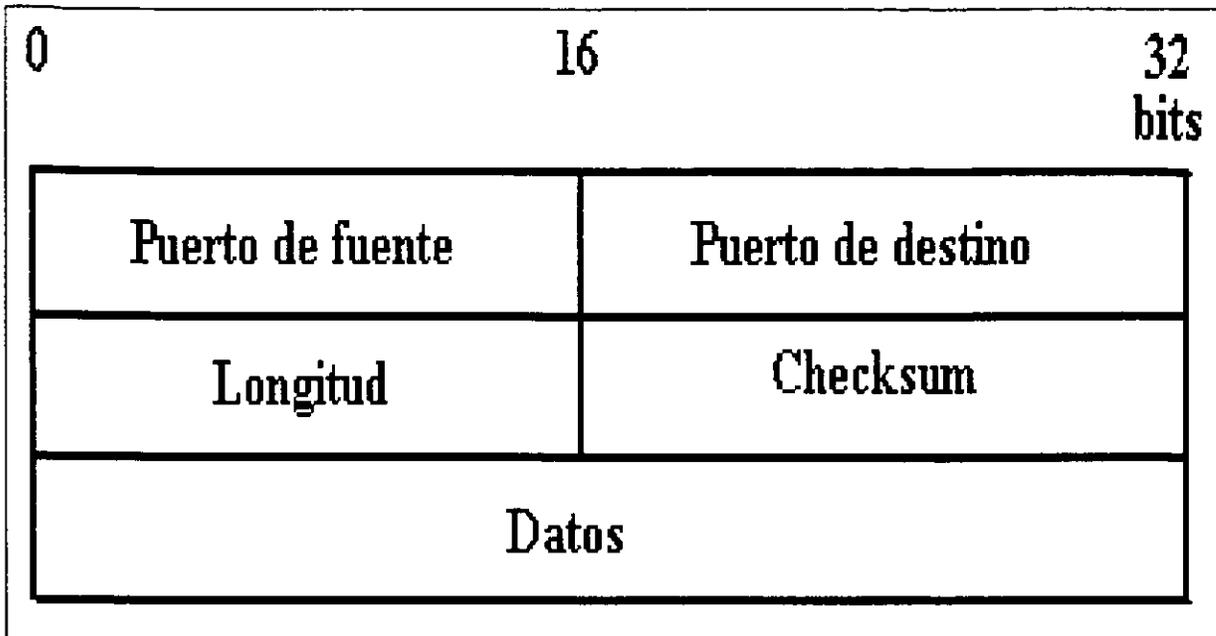


Figura 1.5.9.8 Formato del Mensaje UDP.

1.5.9.9 Protocolo ICMP (Internet Control Message Protocol).

El protocolo de mensajes de control de Internet (ICMP) es de características similares al UDP, pero con un formato aún más simple, ICMP proporciona mensajes y señales de error y control necesarios para los sistemas de la red:

- Reporta sobre destinos inalcanzables.
- Control de flujo de datagramas y congestión.
- Controla los requerimientos de cambio de rutas entre compuertas.
- Detecta rutas circulares o excesivamente largas.
- Verifica la existencia de trayectorias hacia alguna red y el estatus de la misma.

Su función es la de notificar los eventos en los que los paquetes enviados no alcanzaron su destino. ICMP no está orientado a la corrección de errores, sólo a su notificación. Por lo general, cualquier mensaje ICMP que esté reportando un problema de envío incluye también el encabezado y los primeros 64 bits del campo del datagrama donde ocurrió el problema. Esto es útil para dos cosas:

- **Primera.** Permite el dispositivo transmisor igualar el fragmento del datagrama por comparación.
- **Segunda.** Permite que la máquina receptora del mensaje ICMP realice algún diagnóstico .

El formato de ICMP cambia dependiendo de la función que realice, exceptuando los campos de Tipo, Código y de Checksum. Un 1 en el campo de protocolo del mensaje de IP indicará que se trata de un datagrama ICMP. La figura 1.5.9.9 muestra el diseño del mensaje de ICMP.

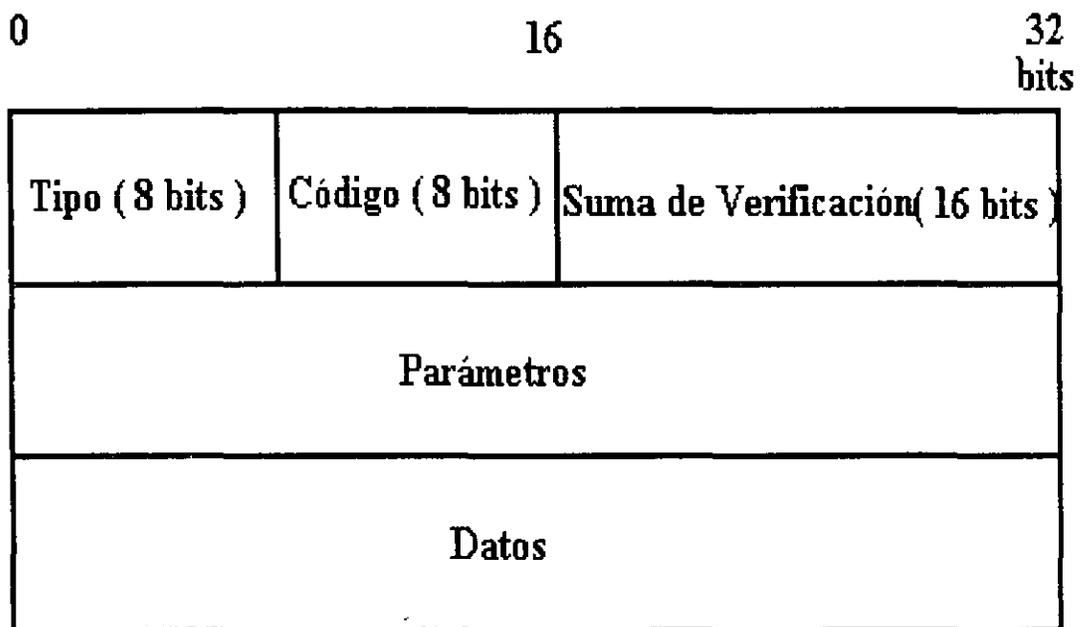


Figura 1.5.9.9 Formato del Mensaje ICMP.

La función de un mensaje determinado ICMP estará definida por el campo de Tipo; el campo de Código proporciona información adicional para realizar la función; el campo de Checksum sirve para efectuar una verificación por suma que sólo corresponde al mensaje ICMP.

El campo Tipo de mensaje de 8 bits en el encabezado ICMP puede tener uno de los valores mostrados en la tabla 1.5.9.9.

Tipo	Mensaje ICMP
0	Eco de respuesta al eco.
3	Destino Inalcanzable.
4	Fuente agotada.
5	Se requiere redireccionamiento.
8	Solicitud de Eco.
11	Tiempo del datagrama excedido.
12	Problema con los parámetros.
13	Requerimiento de hora y fecha.
14	Respuesta de host y fecha (tiempo).
15	Solicitud de información.
16	Respuesta de información.
17	Solicitud de mascara de dirección.
18	Respuesta de máscara de dirección.

Tabla 1.5.9.9. Tipos de Mensaje.

1.5.9.10 Protocolo de Resolución de Direcciones ARP (Address Resolution Protocol).

Le permite a un equipo obtener la dirección física de un equipo destino, ubicado en la misma red física, proporcionando solamente la dirección IP destino.

Las direcciones IP y física de la computadora que consulta es incluida en cada emisión general ARP, el equipo que contesta toma esta información y actualiza su tabla de conversión. ARP es un protocolo de bajo nivel que oculta el direccionamiento de la red en las capas inferiores, permitiendo asignar, a nuestra elección, direcciones IP a los equipos en una red física.

Una conversión dinámica de direcciones Internet a direcciones físicas es la más adecuada, debido a que se obtiene la dirección física por respuesta directa del nodo

que posee la dirección IP destino. Una vez que la dirección física se obtiene, ésta es guardada en una tabla temporal para subsecuentes transmisiones, de no ser así podría haber una sobrecarga de tráfico en la red debido a la conversión de direcciones por cada vez que se transmitiera un paquete.

Formato del Protocolo ARP.

La figura 1.5.9.10 muestra el formato del mensaje de ARP para cuatro octetos por línea.

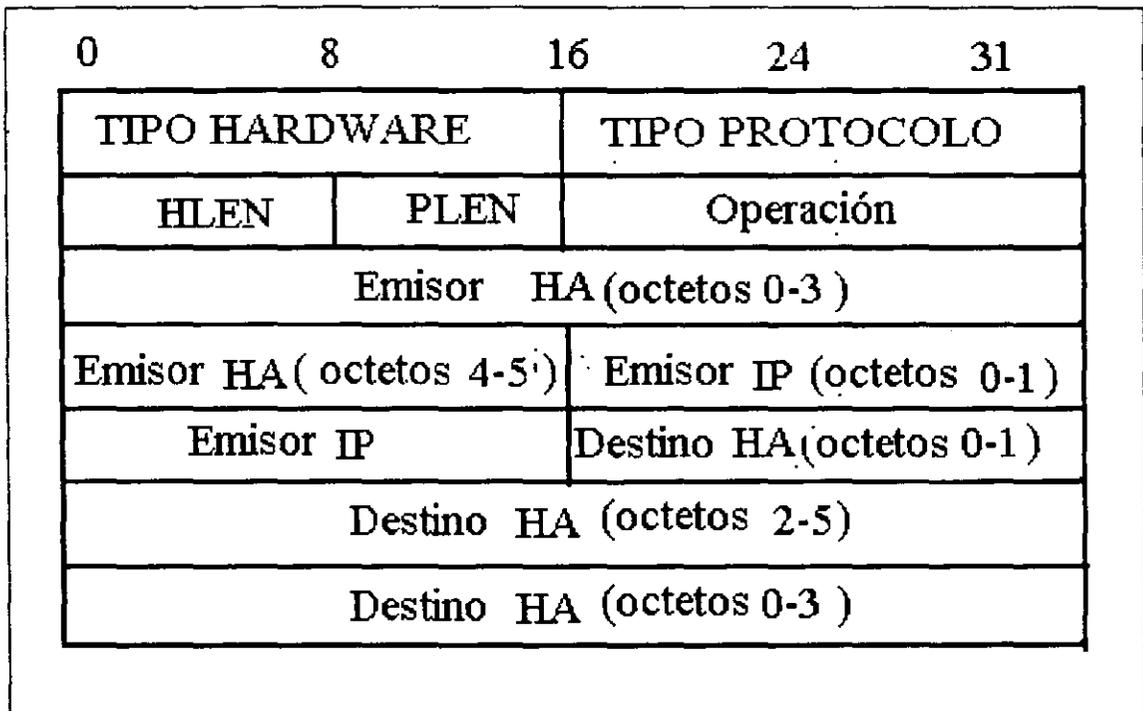


Figura 1.5.9.10 Formato del Mensaje ARP.

La interface de red recibe un datagrama IP a enviar a un equipo destino, en este nivel se coteja la tabla temporal de conversión, si existe una con la referencia adecuada ésta se incorpora al paquete y se envía.

Si no existe la referencia, un paquete ARP de emisión general con la dirección IP destino, es generado y enviado.

Todos los equipos en la red física reciben el mensaje general y comparan la dirección IP que contienen con la suya propia, enviando un paquete de respuesta que contiene su dirección IP.

La computadora origen actualiza su tabla temporal y envía el paquete IP original, y los subsecuentes, directamente a la computadora destino.

El funcionamiento de ARP no es tan simple como parece. Supóngase que en una tabla de conversión exista un mapeo de una máquina que ha fallado y se le ha reemplazado la interface de red; en este caso los paquetes que se transmitan hacia ella se perderán pues ha cambiado la dirección física, por tal motivo la tabla debe eliminar entradas periódicamente, cuya definición la muestra la tabla 1.5.9.10

CAMPO	DESCRIPCION
TIPO HARDWARE	Tipo de interfaz de hardware.
TIPO DE PROTOCOLO	Tipo de protocolo que está usando el dispositivo de hardware.
HLEN	Longitud de la dirección del hardware.
PLEN	Longitud de la dirección del protocolo.
Operación	Indica si es mensaje de consulta o de respuesta.
HA	Dirección del hardware.
IP Emisor	Dirección IP del Emisor.
HA Destino	Dirección Física del Destino.
IP Destino	Dirección IP del Destino.

Tabla 1.5.9.10 Definición del Mensaje ARP.

1.5.10 Redes Orientadas a Conexión y No Orientadas a Conexión.

La red orientada a conexión se caracteriza porque no hay conexión lógica inicial entre los ETD (Equipos terminales de datos) y la red, la conexión de red entre dos ETD está inicialmente en estado desocupado. Si dos computadoras o terminales desean

comunicarse mediante una red orientada a conexión, deben establecer la comunicación mediante un protocolo.

Una vez que se ha establecido la conexión, hay un estado de transferencia de datos utilizando un protocolo establecido, finalmente los ETD liberan la conexión, después vuelven al estado desocupado.

Las redes orientadas a la conexión cuidan mucho los datos del usuario. El procedimiento exige un reconocimiento explícito de que la conexión ha sido establecida, en caso contrario la red informa a los ETD solicitantes de que no ha sido posible establecer la conexión.

La red también debe de informarse del control del flujo, es decir que los datos lleguen correctamente en el orden apropiado, y no saturen los ETD ni los ECD (Equipos de Conmutación de Datos). Se utilizan técnicas de detección y corrección de errores.

Las redes orientadas mantienen un control constante de las secciones ETD-ETD, e intentan asegurar de que no se pierdan datos de usuario en la red. Esta seguridad impone una importante sobrecarga de trabajo en la red ya que son importantes muchas funciones adicionales de soporte.

Las redes no orientadas a conexión, también llamadas datagramas, pasan directamente al estado desocupado (los ETD no se conectan lógicamente) al estado de transferencia de datos, seguido directamente por el estado desocupado. La diferencia principal es la ausencia en la fase en la que establece la conexión y de la fase en la que se libera la conexión.

Además, la red no orientada a conexión carece de reconocimientos, control de flujo o detección/corrección de errores aunque esos servicios pueden proveerse en cada enlace concreto, obviamente las redes no orientadas a conexión tienen menos sobrecarga.

Las bases de comparación y de selección entre redes orientadas y no orientadas a conexión se centra en la sobrecarga frente a las funciones que proporcionan. Las redes orientadas a conexión aumentan el costo del sistema por tener mayores funciones.

Y las no orientadas tienen menos sobrecarga pero dan menos soporte al proceso de aplicación al usuario, muchas veces la selección final viene condicionada por la integridad y la seguridad que se desean proporcionar a los datos.

La figura 1.5.10 muestra un diagrama de las dos redes.

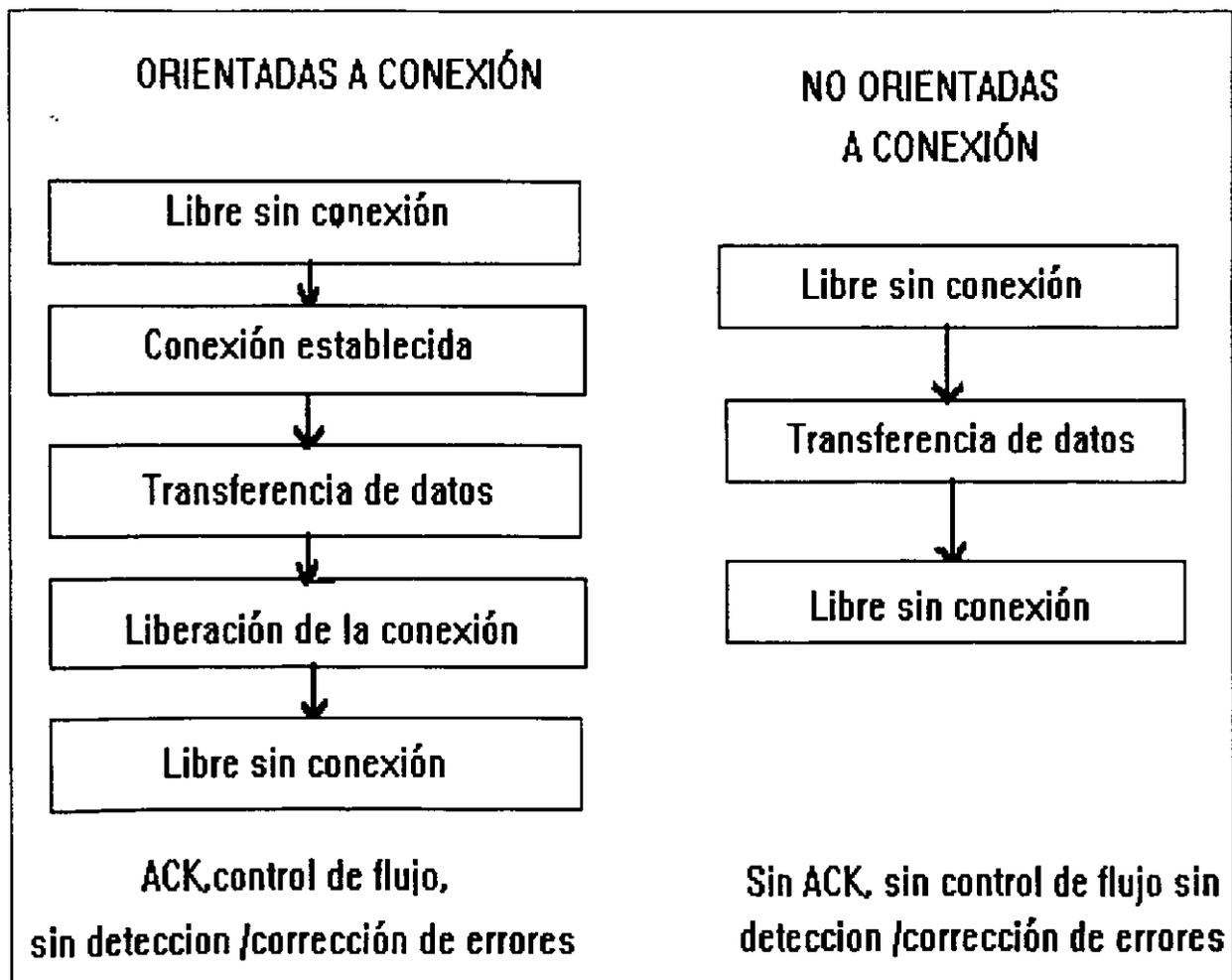


Figura 1.5.10 Redes Orientadas y no Orientadas a Conexión.

1.5.10.1 Servicio del Tipo Connection Less (No Orientado a Conexión).

Las características principales de un servicio de este tipo al nivel de red son las siguientes:

- No hay establecimiento de ninguna conexión, solo hay transferencia de datos. Se usan primitivas del tipo **Unit Data Request** (Requerimiento de Unidad de Datos) y **Unit Data Indication** (Indicación de Unidades de Datos), que contienen como parámetros: dirección del destinatario, dirección de la fuente, QOS (parámetro que determina la calidad de servicio) y datos del usuario.
- Cada paquete que se transmite se transporta de manera independiente con respecto a los paquetes predecesores.
- No debe de llevar a cabo la comprobación de errores y el control de flujo, CL es un servicio no confirmado.
- Las **UnitData** (unidades de datos) sólo pueden transmitir hasta un máximo de 64512 octetos y han de preservar la integridad de la información.
- El emisor solo se encarga de vaciar el paquete en la red, deseando que todo resulte lo mejor posible.

1.5.10.2 Servicio de Red Orientado a Conexión (CO).

Todo servicio CO tiene tres fases, y cada una de ellas presenta sus correspondientes funciones:

- **Conexión.** En esta fase se definen las siguientes funciones:
 - Connect.Request (Equivale a marcar en telefonía)
 - Indication (Equivale al tono de llamada).

- Response (Equivale a descolgar)
- Connect.Confirm (Equivale a la percepción del que llama).

- **Transferencia.** En esta fase se definen las siguientes funciones:
 - Data.Request . Data.Indication Una para enviar y otra para recibir datos.

Se observa que en la definición no hay una función que indique qué ha pasado con los datos. Por tanto, los servicios CO son servicios no confirmados (en cuanto a la transferencia de datos). En un servicio CO puede no existir limitación ni en la longitud de los datos, ni en el ritmo al que se envían (puede ser un servicio ilimitado) ya que al llegar los paquetes en orden, se pueden realizar protocolos sencillos de reconstrucción de la información.

- **Liberación.** En esta fase de un servicio CO se definen las siguientes funciones:
 - Disconnect.Request y Disconnect.Indication.

Por la definición, un servicio CO es un servicio no confirmado en cuanto a la liberación de la conexión. También se observa en la definición que la liberación la puede pedir cualquiera de las partes.

Un servicio CO es un servicio que si se invoca puede provocar pérdida de información, porque si se libera el circuito la red puede borrar los datos en tránsito. Hay que destacar que esta última característica de los servicios CO, al igual que la de ser limitados en cuanto a la transferencia de datos, no son parte de la definición de un servicio CO sino que dependen de cómo están implementadas en cada red concreta.

1.6 BASES DE DATOS RELACIONALES.

1.6.1 Context (Cartucho de Oracle 8).

Context es un cartucho (utilerías) de la Base de Datos que tiene por objetivo principal realizar búsquedas de texto y palabras en la Base de Datos o en archivos de formatos múltiples, por ejemplo: Acrobat Reader, Word, Excel, Word Perfect, entre otros. Las partes que integran este cartucho son las siguientes:

Servidores de Context.

Un Servidor de Context, es el que se encarga de atender las peticiones de los usuarios para hacer consultas o búsquedas a la Base de Datos. Cuando se inicia un servidor de Context, se dice que se ejecuta como un proceso de **background** (procesos corriendo ocultamente).

El Servidor de Context se inicia con una o más personalidades, como lo es para procesar consultas, usar términos de lingüística, utilizar sentencias **DML** (Lenguaje de Manipulación de Datos, por ejemplo: insert, update, etc.) o sentencias **DDL** (Lenguaje de Definición de Datos, por ejemplo; create table, drop table, etc.). Un punto importante es que se pueden iniciar tantos servidores como se deseen asignándole la personalidad que se requiera.

Política.

Este es uno de los conceptos más importantes para Context, ya que una política identifica la columna que almacena el texto, el método de almacenamiento del texto y también identifica las opciones que Context usará para crear el índice.

Context soporta la creación de políticas de columna para indexación de texto e indexación por tema. La política se basa en una serie de opciones que se pueden observar en la tabla 1.6.1 y en la figura 1.6.1.

OPCIÓN DE INDEXADO	PREGUNTA	VALORES POR OMISIÓN
Data Store	¿Cómo están los datos almacenados en las columnas?	El texto se almacena directamente en la columna de la tabla, cada renglón en la tabla representa un documento separado.
Filter	¿En qué formato se encuentra el texto?	El formato por omisión es ASCII.
Lexer	¿En qué lenguaje se encuentra el texto y cómo son los tokens identificados en el texto?	Inglés u otro lenguaje Europeo. Los tokens son identificados por espacios en blanco, puntuación estándar y caracteres no alfanuméricos, sensitivo a mayúsculas por omisión está deshabilitado.
Engine	¿Cómo es el índice de Context creado y dónde esta almacenado?	El índice creado usa aproximadamente 12.5 MB y es almacenado en el tablespace del usuario.
Stoplist	¿Qué tokens no son incluidos como entradas del índice?	Se proporciona una lista de palabras en español o inglés.
Wordlist	¿Cuales opciones de consulta especial están habilitadas para el índice?	Consulta de Soundex (sonidos parecidos) deshabilitada, fuzzy match (raíces de las palabras).

Tabla 1.6.1. Opciones de Indexado para crear una Política.

En las Políticas se pueden agrupar cada una de las necesidades y programarlas dentro de cualquiera de las siete siguientes áreas:

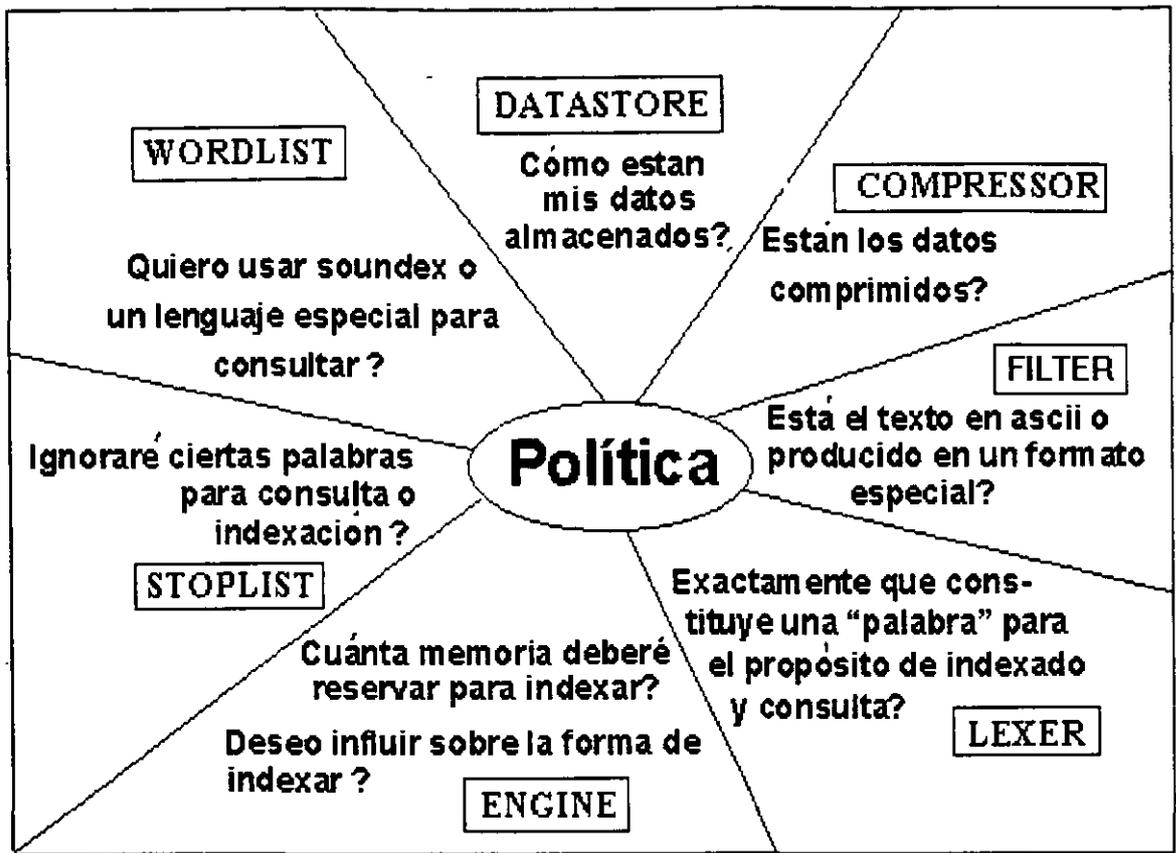


Figura 1.6.1 Áreas para la Creación de una Política.

Personalidad.

La personalidad determina las operaciones que un Servidor de Context puede atender como son instrucciones de DML, DDL, hacer consultas, entre otras. Así al iniciar un Servidor de Context hay que incluirle el tipo de personalidad que manejará en la atención de peticiones sobre el texto

Por ejemplo para iniciar un Servidor de Context con una personalidad que incluya sentencias DML, DDL y consultas en Unix podría definirse de la siguiente forma:

```
$ctxsrv -user ctxsys/ctxsys -personality QDM <Enter>.
```

Tipos de Consulta.

Las expresiones de consulta se conforman de palabras y frases combinadas con operadores y otros caracteres especiales para definir un criterio de búsqueda.

En Context, una consulta de texto es hacer una búsqueda de palabra o frases en un conjunto de documentos.

Una consulta puede tener diferentes criterios como lo son en base a sinónimos, a palabras más genéricas (como mamíferos), términos equivalentes, etc.

Context soporta tres diferentes métodos de ejecutar consultas:

- **Consulta en un paso.** Es usada en SQL*plus y es la mejor para consultas interactivas. Por ejemplo, esta consulta buscará todos los registros con la palabra "ecuaciones":

```
SQL>SELECT materia,profesor
      FROM carrera
      WHERE CONTAINS (carrera.materia,'ecuaciones')>0;
```

- **Ventajas**

- No aloja resultados en tablas intermedias.
- Usa declaraciones estándar de SQL.
- Utiliza nombres de tablas y columnas.

- **Desventajas**

- Generalmente es más lento que la consulta en dos pasos
- No hay acceso posterior a tablas intermedias

- **Consulta en dos pasos.**- Es la más adecuada para aplicaciones basadas en PL/SQL que solicita todos los resultados en una sola consulta. Es decir, ejecuta las búsquedas en los registros y los resultados los almacena en una tabla específica. El

siguiente ejemplo, busca los registros con la palabra "ecuaciones" y los resultados los deja en una tabla temporal, que el usuario ya definió anteriormente:

```
SQL execute ctx_query.contains( "politica_materia","ecuaciones ","temporal");
```

- **Ventajas**
 - Las tablas intermedias pueden ser manipuladas.
 - Generalmente más rápida para combinación de consultas.
 - Los datos estructurados pueden ser consultados como parte de un
CONTAINS

- **Desventajas**
 - Requiere almacenar resultados de la consulta en tablas.
 - Usa nombres de políticas.
 - Requiere la unión de la tabla base del texto para recuperar los detalles del documento.

- **Consultas en memoria.**- Son las más adecuadas para aplicaciones basadas en PL/SQL que pueden generar grandes cantidades de documentos recuperados pero donde solo una pequeña porción de ellos son requeridos a la vez.
 - **Ventajas**
 - No hay tablas intermedias.
 - Es mucho más rápida que las anteriores al no tener que recuperar todos los documentos dentro de la lista que nos trajo la consulta.
 - Se puede especificar el número de elementos a recuperar.

 - **Desventajas**
 - Usa nombres de políticas.
 - No puede recuperar todos los elementos a la vez.

-Todos los resultados son manejados en memoria.

Score y Contains.

Cuando se realiza una consulta sobre un texto o sobre un tema, Context regresa lista de todos aquellos documentos que satisfacen la consulta. Cada documento tiene una puntuación (**score**) que indica la relevancia del documento en la consulta que se realizó; de tal manera que el "score" más alto, será el documento más relevante en la búsqueda. Estos *scores* se pueden manipular para mostrar los documentos más relevantes primero o para hacer una filtración de los mismos.

Contains.

Es una función que especifica la expresión de la búsqueda utilizada en una consulta. Esta función va muy relacionada con el *score* ya que cada función *contains* en una consulta produce una serie de valores separados de *score*. Algunos ejemplos de estas funciones pueden verse a continuación:

- Cada consulta de texto tiene una función CONTAINS

```
Select EMPNO, ENAME, JOB, SAL
from TABLA where
CONTAINS (tabla.columna, 'palabra_buscar') > 0
```

- Cada CONTAINS tiene un SCORE, por ejemplo:

```
select SCORE (10) , ENAME, RESUME
from TABLA
where CONTAINS (tabla.columna, 'palabra_buscar',10) > 0
```

Temas.

Los temas son usados en Context con el fin de clasificar documentos, resumir el texto o hacer búsquedas, es decir, permite realizar búsquedas de documentos por sus conceptos principales, actualmente solo para inglés.

Una consulta por tema es usualmente una palabra o frase que captura el concepto general de lo que realmente se busca. Los temas a grandes rasgos mencionan acerca de que trata el documento y que tan profundo es el contenido del tema.

Resumen de Texto.

Context nos provee de maneras de resumir el texto de un documento:

Uno es a través del sentido genérico el cual da un resumen global, esto es, proporciona el párrafo o unos párrafos que representan de la mejor manera el contenido general del documento.

Y la segunda forma es la de realizar un resumen por cada tema (esto suele ser llamado un punto de vista en base al sentido) y expresarlo en uno o unos cuantos párrafos.

Lingüística.

La lingüística de Context se utiliza para manejar el contenido de los documentos en lenguaje inglés (esto debido, a que por el momento, sólo se contempla este lenguaje). La lingüística se emplea para contemplar diferentes puntos de vistas sobre los documentos que permite al usuario revisar rápidamente su contenido esencial y determinar su relevancia. Se puede incorporar análisis lingüístico y funcionalidad en una aplicación de texto. La lingüística de Context puede generar las siguientes formas de salida para los documentos. Ver tabla 1.6.2.

Tipo de salida	Descripción
Temas	Los conceptos principales de un documento.
Gist	Párrafo o párrafos en un documento que representa de la mejor manera cómo el documento habla acerca de un todo.
Resumen de tema	Párrafo o párrafos en un documento que representa de la mejor manera un tema dentro de un documento.
Sentencia a nivel Gist	La sentencia o sentencias en un documento que mejor representa el tema en el documento como un todo.
Sentencia a nivel resumen de tema	La sentencia o sentencias dentro de un documento que describen un solo tema dentro de un documento.

Tabla 1.6.2 Salidas de Lingüística de Context.

Manejando Texto.

El texto puede estar almacenado en los siguientes formatos:

- En una base de datos existente.
- En un **file system** (sistema de archivos).
- Dentro de un **URL** (Localizadores de recursos uniformes).
- En diferentes formatos de archivos tales como:

ASCII (Código Estándar Americano para Intercambio de Información), **HTML** (lenguaje de Marcación Hipertexto), **MSWord** (editor de Windows), **PDF** (Formato de Documentos Portable). La estructura lógica de Context puede verse en la figura 1.6.2.

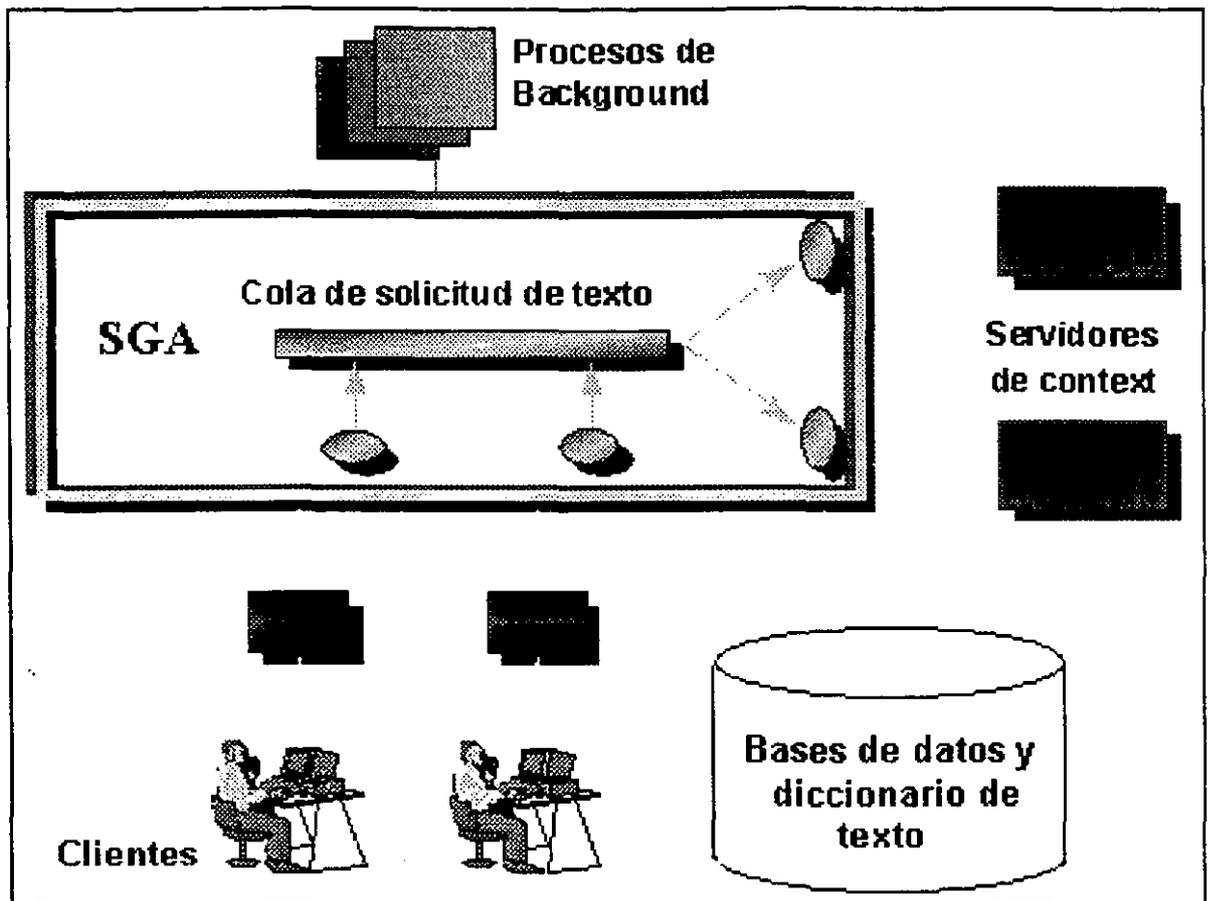


Figura 1.6.2 Estructura Lógica de Contexto.

¿Qué se puede consultar con Contexto?

- Búsqueda por palabras o frases completas.
- Combinaciones de Palabras.
 - "y", "o", "no" ...
- Palabras aproximadas:
 - máscaras , búsqueda difusa , sonido.
- Búsqueda en base a proximidad:
 - Palabras aproximadas a la específica.
- Sinónimos:
 - equivalentes, términos más amplios.

1.6.2 Oracle Enterprise Manager (Administrador Empresarial)

La Necesidad para Administración de Sistemas.

Una gran circunstancia para migrar aplicaciones de misión crítica fuera de los **Main Frame** (primeros sistemas de cómputo robustos) hacia la arquitectura cliente-servidor, ha sido la falta de poder y herramientas de administración de sistemas a bajo costo.

Los usuarios necesitan soluciones de administración de sistemas que les permitan reducir la complejidad de administrar sus ambientes heterogéneos, y les ayuden a decrementar la educación, consultoría y costos administrativos inherentes en productos de múltiples vendedores. La solución ideal necesita proporcionar una rigurosa combinación de integración y estabilidad: la integración es requerida para el control centralizado, mientras la estabilidad es requerida para el crecimiento futuro y flexibilidad. Los administradores desean estar habilitados para garantizar a sus usuarios un nivel base de confianza, disponibilidad y seguridad, así como un alto nivel de procesamiento.

Solución Oracle: Oracle Enterprise Manager (OEM).

Oracle Enterprise Manager (Administrador Empresarial de Oracle) es la clave de la estrategia la cual hará a Oracle la Base de Datos más manejable para la empresa. La familia de OEM consiste en la nueva generación de herramientas de administración de sistemas diseñado para hacer más eficaz la administración del ambiente de Oracle completamente, incluyendo sistemas, Base de Datos, redes y aplicaciones.

Oracle Enterprise Manager proporciona una solución integrada para administrar ambientes heterogéneos, con una arquitectura abierta cliente-servidor. La arquitectura cliente-servidor la cual es escalable y robusta, consiste de una consola centralizada, servicios comunes y agentes inteligentes ejecutándose en los nodos manejados.

Varias aplicaciones residen en los servicios comunes, ejecutando tareas de administración de sistemas. Estas aplicaciones pueden ser proporcionadas por Oracle, por otros clientes y vendedores de software de terceros.

Esto es posible gracias a la estabilidad de los productos, resultado de la disponibilidad de múltiples **APIs** (Interfaces de Programación de Aplicaciones) en ambos casos, de los clientes y la integración de terceros. Ver figura 1.6.3.

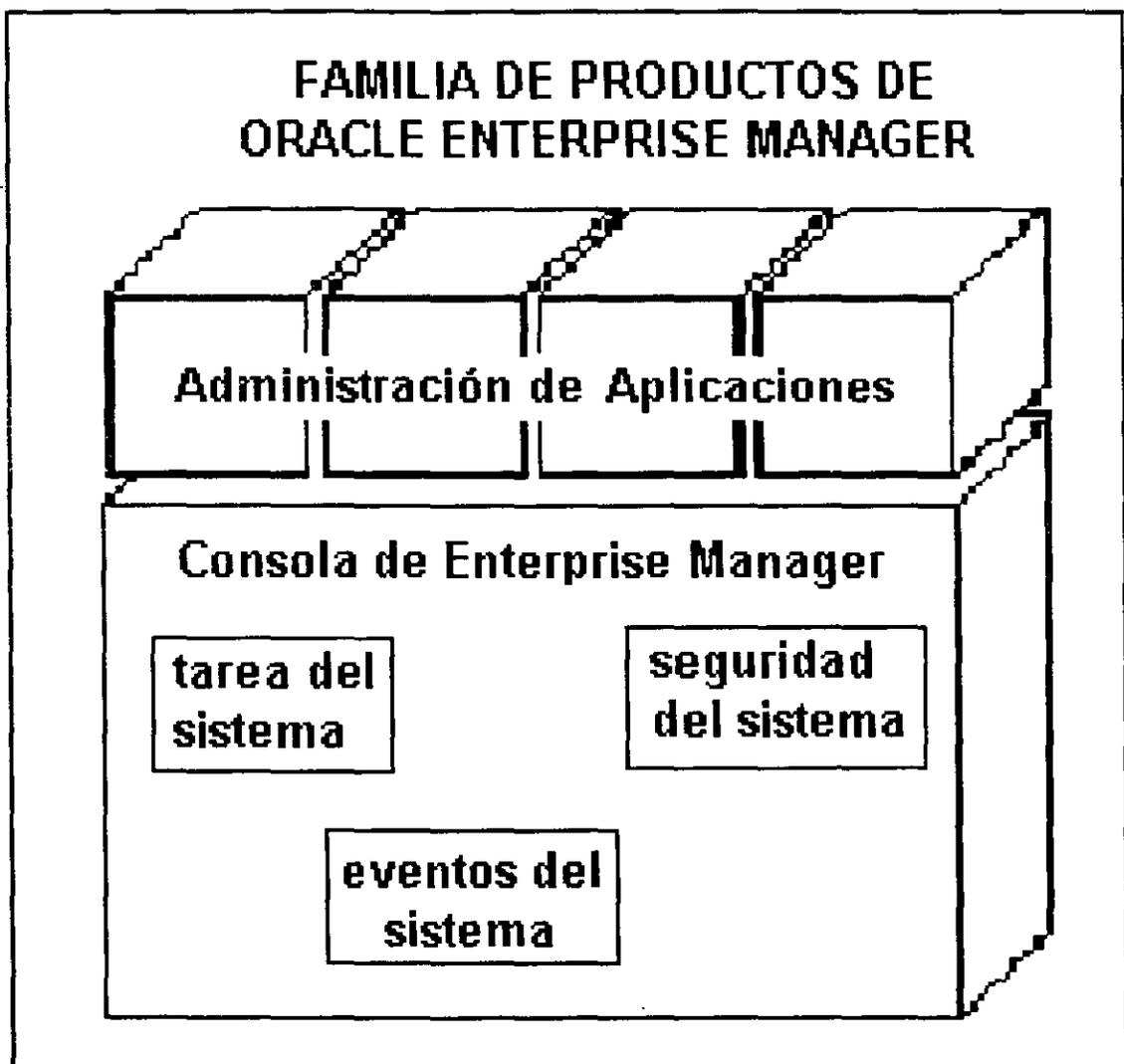


Figura 1.6.3. Estructura de Enterprise Manager.

Panorama General de Oracle Enterprise Manager (OEM).

Oracle Enterprise Manager consiste de una consola centralizada, un conjunto de servicios comunes, agentes inteligentes y un paquete de aplicaciones DBA (Administrador de Base de Datos) estándar las cuales proporcionan al DBA funcionalidad que ellos necesitan para administrar sus ambientes.

Existe además, un paquete de aplicaciones integradas, conocido como Oracle Enterprise Manager Performance Pack (Paquete de Ejecución de Administración Empresarial), el cual proporciona monitoreo avanzado y experto, y además la funcionalidad de "tuning" (afinamiento de base de datos). Ver figura 1.6.4.

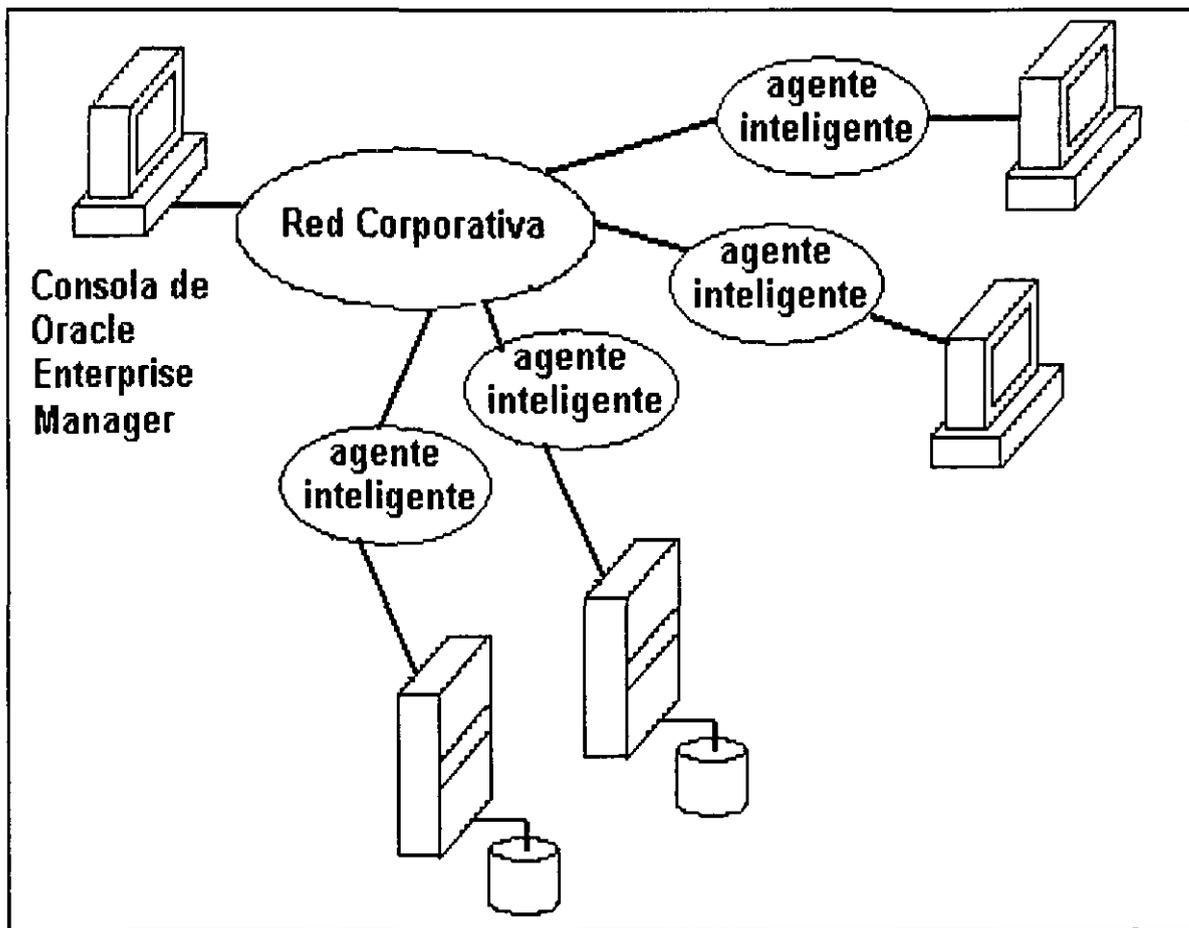


Figura 1.6.4 Arquitectura de Enterprise Manager.

Consola de Oracle Enterprise Manager.

Ofrece control centralizado. Los sistemas distribuidos y bases de datos pueden ser manejadas desde una simple consola centralizada, residiendo en el servidor del administrador. La consola proporciona una interfaz de usuario gráfica y un "punto de control" centralizado, con tareas fácilmente ejecutadas por el método de "**drag and drop**" (arrastre de mouse).

La consola consiste de cinco componentes principales o servicios comunes:

- El navegador.
- La ventana de mapa.
- El sistema de administración de eventos.
- El sistema de esquematización de trabajos.
- El sistema de seguridad.

El Agente Inteligente del OEM.

El agente inteligente es un proceso que corre en nodos remotos. Esto es eficiente y evita la intromisión de intrusos, y además funciona como el ejecutor de trabajos y eventos enviados por la consola. La alta disponibilidad es consecuencia de que el agente inteligente pueda realizar funciones que procedan de la consola.

Resumen de Oracle Enterprise Manager.

Oracle Enterprise Manager proporciona una solución integrada para las necesidades de administración de Oracle. Esto permite a los administradores manejar sus ambientes heterogéneos completamente desde una simple consola centralizada, optimiza las necesidades de administración de extensas y cuantiosas Bases de Datos.

Con esta disponibilidad para administrar complejidades y reducir costos, Oracle Enterprise Manager proporciona la solución de administración completa a los clientes que han estado esperando por ello.

1.6.3 SQL PLUS (Lenguaje Estructurado de Consultas).

Panorama general de SQL Plus

Se puede utilizar SQL Plus en conjunción con el lenguaje estándar de Base de Datos SQL y esto da como resultado la extensión del lenguaje procedural que es **PL/SQL** (lenguaje procedural SQL). El lenguaje de Base de Datos SQL permite almacenar y extraer información en Oracle. PL/SQL permite enlazar diferentes comandos de SQL a través de la lógica de procedimientos.

SQL Plus te habilita para manipular comandos SQL y bloques de PL/SQL, además te permite ejecutar muchas tareas adicionales. También, con SQL Plus se puede hacer lo siguiente:

- Editar, almacenar, extraer y ejecutar comandos SQL y bloques PL/SQL.
- Formatear, ejecutar cálculos, almacenar e imprimir consultas en la forma de reportes.
- Listar definiciones de columnas para cualquier tabla.
- Accesar y copiar datos entre bases de datos SQL.
- Enviar mensajes y aceptar respuestas desde los usuarios finales.

Conceptos Básicos.

Las siguientes definiciones explican conceptos centrales de SQL Plus:

- **command**, es una instrucción que te da Oracle y SQL Plus.

- **block**, es un grupo de comandos de SQL y PL/SQL relacionados con otros procedimientos lógicos.
- **table**, es la unidad básica de almacenamiento en Oracle.
- **query**, es un comando SQL (específicamente, un comando SELECT) que extrae información desde una o más tablas.
- **query results**, son los datos extraídos de un "query" (consulta).
- **report**, son los resultados de consultas formateadas por el usuario a través de comandos SQL*Plus.

Requerimientos para Ejecutar SQL Plus.

Para ejecutar SQL Plus, se requiere de información específica acerca del sistema operativo, **software** (programas de cómputo), **hardware** (equipo de cómputo) y además acceder a la información de las tablas a través de un **username** (nombre de usuario) y **password** (contraseña).

Hardware y Software.

Oracle y SQL Plus pueden ejecutarse desde diferentes clases de computadoras. El sistema operativo administra los recursos de la computadora y es el enlace entre el hardware del sistema y los programas tales como SQL*Plus. Diferentes computadoras usan diferentes sistemas operativos. Para la información acerca del sistema operativo, ver la documentación de la computadora.

Antes de poder iniciar SQL Plus, ambos productos Oracle y SQL Plus deben instalarse en la computadora. Si se tienen múltiples usuarios en la computadora, la empresa u organización deberá tener un Administrador de Base de Datos (DBA) quién supervisará el uso de los programas Oracle. El DBA es el responsable de instalar Oracle y SQL Plus en el sistema. Si se está actuando como DBA, revisar las

instrucciones para instalar Oracle y SQL Plus dentro del manual del usuario e Instalación de Oracle proporcionados para el sistema operativo específico.

Username y Password.

Cuando se inicializa SQL Plus, se requerirá un "username" que te identifica como un usuario autorizado y el "password" que verifica la autenticidad con username. Para sistemas Multi-usuario, las distintas personas que comparten el sistema operativo de la computadora, deberán contar con un "username" y "password" proporcionados por el DBA para trabajar en sesiones de SQL Plus. Se requerirá además un "username" y "password" para confirmar al usuario a nivel sistema operativo. En caso de ser la única persona en la computadora, se puede ejecutar las tareas de un DBA por sí mismo.

Inicialización de SQL Plus.

A continuación se muestra cómo inicializar una sesión de SQL Plus.

1. Asegurarse que Oracle ha sido instalado en la computadora.
2. Identificarse a nivel sistema operativo en la computadora.
3. Escribir en el **Prompt** (indicador del sistema) de la computadora el siguiente comando SQLPLUS. Este comando inicializa una sesión de SQL*Plus.

```
$ SQLPLUS [Enter]
```

SQL*Plus desplegará el número de versión, la fecha e información acerca de derechos de autor.

```
SQL*Plus: Release 8.0 - on Fri Jun 27 09:39:26 1997
```

```
(c) Copyright 1997 Oracle Corporation. All rights reserved.
```

```
Enter user-name:
```

4. Escribir el username y presionar [Enter]. SQL*Plus desplegará lo siguiente:
"Enter password:".

5. Escribir el password y presionar [Enter]. Como protección no se observará lo que se escribe en la pantalla.

A continuación, SQL*Plus desplegará el siguiente prompt de SQL*Plus: SQL>

Este prompt indica que SQL*Plus está listo para trabajar y recibir comandos. También se puede emplear un método de autenticación como el siguiente:

```
$ SQLPLUS USERNAME/PASSWORD [Enter]
```

Saliendo de SQL*Plus.

Cuando se desee terminar la sesión de SQL Plus se deberá teclear lo siguiente:

```
SQL> EXIT [Enter]
```

Escribiendo Comandos.

Se pueden escribir los comandos en mayúsculas o minúsculas. Existen tres clases de comandos que tu puedes ejecutar en el prompt de SQL Plus:

- Comandos SQL, para trabajar con información dentro de la Base de Datos.
- Bloques de PL/SQL (procedimientos y funciones), también para trabajar con información en la base de datos.
- Comandos de SQL Plus, para formatear resultados de consultas, colocar opciones, editar y almacenar comandos SQL y bloques de PL/SQL.

Ejecutando Comandos.

El lenguaje SQL permite manipular datos dentro de la Base de Datos. En este ejemplo, se ejecutará un comando SQL para desplegar el número de empleado, nombre, trabajo y salario de cada empleado en la tabla EMP (proporcionada como tabla de demostración al instalar Oracle).

1. En el prompt de SQL Plus, escribe la primer línea del comando:

```
SQL> SELECT EMPNO, ENAME, JOB, SAL
```

Cuando se presione [Enter] el cursor irá a la siguiente línea. Entonces SQL*Plus desplegará un "2", y se debe escribir la siguiente línea de la instrucción.

```
FROM EMP WHERE SAL < 2500; [Enter]
```

El punto y coma (;) es la forma en que se indica la terminación de la sentencia o comando. Este comando desplegará los siguientes resultados en la pantalla:

EMPNO	ENAME	JOB	SAL
-----	-----	-----	-----
7369	smith	clerk	800
7499	allen	salesman	1600
7521	ward	salesman	1250
7654	martin	salesman	1250
7782	clark	manager	2450

Después de desplegar los resultados y el número de registros encontrados, SQL Plus desplegará el "prompt" nuevamente.

Obteniendo Ayuda.

Para obtener ayuda en línea acerca de los comandos de SQL*Plus, teclear HELP en el prompt seguido del nombre del comando, por ejemplo:

SQL> HELP ACCEPT

El Buffer de SQL.

El área donde SQL Plus almacena sus más recientes comandos o bloques PL/SQL es llamado Buffer SQL. El comando o bloque es recordado hasta que se indique un nuevo comando. Si se quiere editar o re-ejecutar el actual comando SQL o bloque de PL/SQL, se podrá hacerlo sin re-escribir este. La forma de hacerlo es tecleando el carácter: (/), lo cual ejecutará nuevamente la sentencia almacenada en el buffer.

Resumen de comandos SQL Plus.

- Condiciones alternativas de búsqueda.

```
SELECT <columna1, columna2, ...>  
FROM <tabla>  
WHERE <condición> OR <condición> ;
```

- Ordenando renglones.

```
SELECT [DISTINCT] <columna1, columna2, ...>  
FROM <tabla>  
ORDER BY <columna1, columna2,...> [ASC | DESC] ;
```

- Cálculos. (operadores: * multiplicación; / división; + suma; - resta)

```
SELECT <columna1, columna2, ...>, <expresión aritmética>  
FROM <tabla>  
[ WHERE <condición_con_expresión_aritmética> ]  
[ ORDER BY <columna1, columna2,...> <expresión_aritmética> ] ;
```

- Funciones. (AVG, SUM, MIN, MAX, COUNT "puede usarse con * ")

```
SELECT función ( [DISTINCT ] <columna> )
FROM <tabla>
[ WHERE <condición> ] ;
```

- Agrupación.

```
SELECT <columna1, columna2, ...>, <función (argumento)>
FROM <tabla>
[ WHERE <condición> ]
[ GROUP BY <columna1, columna2, ...> ]
[ HAVING <condición> ]
[ ORDER BY <columna1, columna2,...> [ASC | DESC] ] ;
```

- SubConsultas.

```
SELECT <columna1, columna2, ...>
FROM <tabla> WHERE <columna> <operador | IN>
( SELECT <columna> FROM <tabla>
[ WHERE <condición> ] ) ;
```

- Joins (enlaces de tablas)

```
SELECT <columna1, columna2, ...>
FROM <tabla_1>, <tabla_2>, <.... tabla_n>
WHERE <tabla_1.columna> = <tabla_2.columna> ..... --condición Join
AND <tabla_n-1.columna> = <tabla_n.columna> --condición Join
[ AND | OR <condición> ]
[ ORDER BY <columna1, columna2 ...> ] ;
```

- Lenguaje de Manipulación de Datos.

```
INSERT INTO <tabla> [<columna1>,<columna2>...]
```

```
VALUES (<valor1>, <valor2> ...);
```

o bien:

```
INSERT INTO <tabla1> [<columna1>,<columna2>...]
```

```
SELECT <columna1>,<columna2...>
```

```
FROM <tabla2>
```

```
[WHERE <condición>];
```

- Lenguaje de Definición de Datos.

```
CREATE TABLE <tabla>
```

```
( <columna_1><tipo_de_dato><constraint_columna>, .....
```

```
<columna_n><tipo_de_dato><constraint_columna> );
```

```
DROP TABLE <table> ;
```

1.7 ARQUITECTURA NCA.

1.7.1 Ambiente Económico.

En el ambiente comercial de hoy, los profesionales encargados de manejar los asuntos de IT (Tecnología de Información) de una empresa, tienen como desafío, equilibrar las necesidades tecnológicas y operacionales con las necesidades financieras y comerciales.

Los factores a tomar en cuenta hoy en día, para lograr este equilibrio son los siguientes:

- **Digitalización:** El inmenso mundo de formas tradicionales para manejar datos, se vuelve más accesible, siguiendo un modelo no lineal. La digitalización hace que esos datos se transformen en información valiosa.
- **Globalización:** Geográficamente, la información se vuelve menos limitada gracias a la globalización.
- **Movilidad:** El poder de las redes de telecomunicación en cuanto al soporte en mano de obra, se transforma a un modo estándar de operación. Con esto se logra acceder información de manera fácil, desde cualquier parte.
- **Workgroups (Grupos de trabajo):** La información se comparte gracias a las comunicaciones lo cual favorece la formación de grupos de trabajo: compañías virtuales, e-mail, soporte en videoconferencias a través de la infraestructura de telecomunicaciones.
- **Inmediación:** Se reforzarán por medio de la operación de redes, productos **GroupWare** (de grupo), y modelos comerciales de correo electrónico.
- **Desintermediación:** Para tener un trato directo entre el proveedor y el consumidor, debe evitarse el intermediarismo.

Cada uno de estos seis elementos continúa influyendo en cómo cada empresario planea para finalmente lograr la integración de datos e información en su ambiente de trabajo.

En la figura 1.7.1 se representan los factores que influyen en el ambiente económico de la tecnología de información actual.



Figura 1.7.1 Ambiente Económico de la Tecnología de Información Actual.

Los aspectos económicos son la causa de muchas de nuestras tendencias tecnológicas. Las tendencias con un tiempo de amortización más corto serán, normalmente, las dominantes.

La explosión de Internet es un claro ejemplo de cómo funciona el aspecto económico. Hoy en día, resulta muy barato para cualquiera poner en marcha una tienda en Internet y tener acceso a 100 millones de personas. Ahora que las grandes empresas ven la Internet como una plataforma comercial viable, hay una tendencia generalizada a desarrollar estrategias Internet corporativas. El hecho de que Internet ofrezca un cliente

perfecto, con un bajo coste de instalación y mantenimiento, es la razón de esta tendencia a desarrollar aplicaciones basadas en Internet/Intranet.

Network Computing Architecture (Arquitectura de Cómputo en Red) representa la próxima evolución en arquitectura de tecnología de información para desplegar en esta economía de información.

NCA (Network Computing Architecture) puede convertirse en el medio lógico para asistir la dirección y los desafíos de la tecnología de información para el encuentro entre el hoy y el futuro y para ayudar a alinear las estrategias de despliegue de tecnología con las estrategias comerciales.

1.7.2 Evolución del Concepto Network Computing.

Para tener una visión más íntima sobre el paradigma de la informática, es necesario dar un paso adelante y aprender más sobre como el concepto de Network Computing (NC) ha evolucionado:

La aparición del **mainframe** (unidad central) no tiene mucho tiempo, y muchas compañías lo siguen usando. La arquitectura del mainframe se organiza de manera simple. Todas sus aplicaciones y lógica de los datos se centralizan en la computadora del mainframe. Este sistema proporciona un lugar único para integrar y poner al día la información. Su mantenimiento resulta caro, su manejo resulta complejo y no es muy flexible. Con el tiempo, estos inconvenientes hicieron que la industria optara por una arquitectura que representara una mejor alternativa y optaron por una arquitectura cliente/servidor.

La arquitectura cliente/servidor coincidió con la disponibilidad creciente de computadoras personales y trasladó mucha de la carga de procesamiento a las llamadas **desktops** (computadoras de escritorio). Esto significó un paso hacia delante en cuanto al tradicional manejo del mainframe, lográndose con esto para el usuario un

mayor poder y una interface gráfica. El sistema cliente es responsable de la presentación de la información y de la lógica de la aplicación. El servidor primario es usado para almacenar los datos lógicos. Aunque la arquitectura cliente/servidor concede poder al usuario de computadoras de escritorio, su implementación y mantenimiento continúan siendo caros y su manejo duro y difícil. Cada computadora resulta diferente del resto y para una empresa, resulta una pesadilla intentar actualizar sus aplicaciones a lo largo de todos sus equipos de cómputo.

NCA incorpora lo mejor de los dos tipos anteriores de arquitectura y aprovecha el avance global del Internet y la creación de dispositivos económicos como las computadoras de red.

La lógica de NCA es tan atractiva como una arquitectura de tecnología de información, ya que puede fusionar la distribución de software, con los beneficios de manejabilidad de los mainframes y con el usuario orientado a la arquitectura cliente/servidor. Además Network Computing Architecture tiene un nivel de flexibilidad que está más allá de los ofrecidos anteriormente. Este modelo, optimiza las características de las arquitecturas anteriores para lograr una funcionalidad específica.

En la figura 1.7.2 se muestra como las arquitecturas de red evoluconan del mainframe al cliente servidor, para después incorporar las características más importantes de ambas en la arquitectura de Network Computing.

NCA aún no se consolida en su totalidad con las tecnologías de Web o Internet dentro de las organizaciones ya que se requiere de una mayor infraestructura de software sofisticado para cumplir plenamente con su cometido. Este concepto parte de que todos los usuarios accesen aplicaciones conforme a las normas basadas en los **Web Browsers** (Navegadores). Esto significa que cualquier usuario (con autorización de seguridad) puede acceder cualquier información desde cualquier dispositivo. Esto aplica a los usuarios internos y externos como pueden ser proveedores, compañeros de trabajo o bien clientes.

Evolución del Network Computing

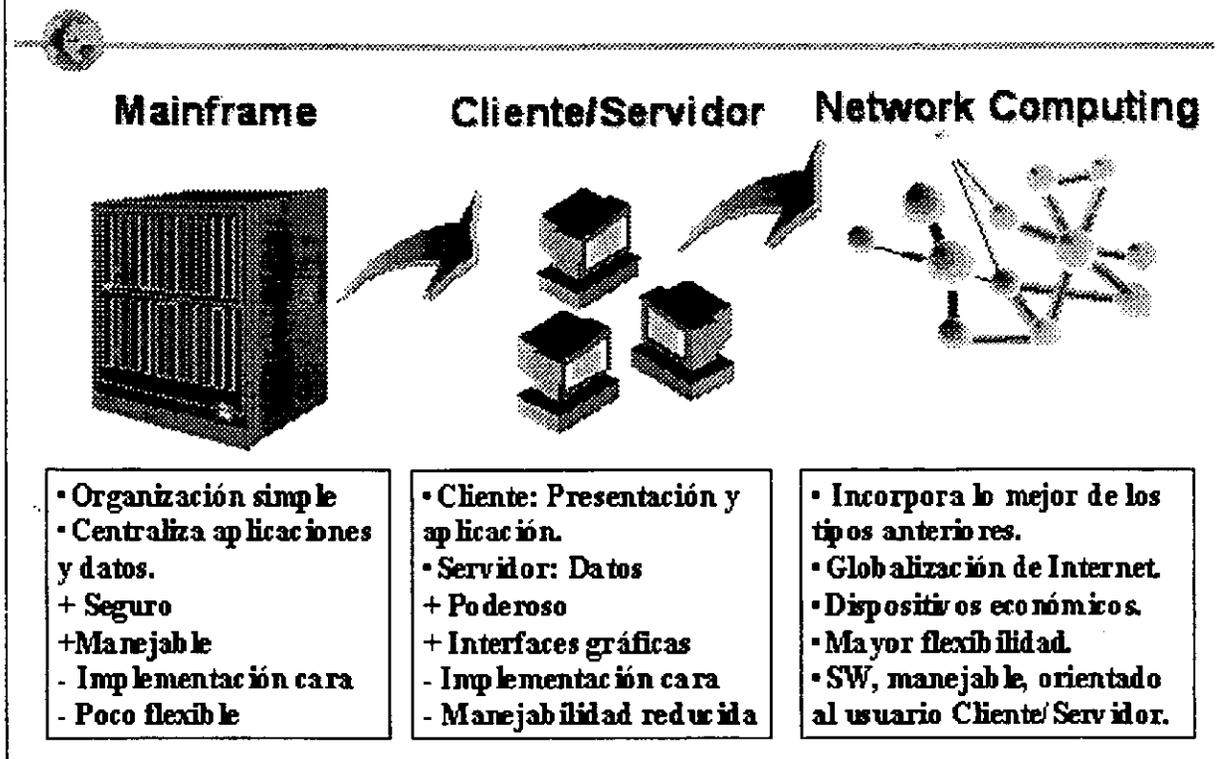


Figura 1.7.2 Evolución del Concepto Network Computing.

El uso de los Web Browsers logrará que el tiempo de instalación de software se elimine o acorte dramáticamente. Los problemas y posibles complicaciones surgidos durante la instalación de programas en sistemas personales, son eliminados al utilizar NCA donde los llamados **"thin-clients"** (clientes delgados) son descargados por el browser (típicamente para la porción gráfica de una aplicación) y las funciones principales de la aplicación corren en los servidores de la red.

Estas aplicaciones **"server-based"** (basadas en el servidor) son más fáciles de manejar que en su contraparte cliente/servidor y son menos vulnerables a las variantes de un ambiente en PC.

La productividad de una empresa se ve afectada al querer un usuario administrar su propio PC, adicionalmente la productividad sufre pérdidas al corregirse problemas causados por los usuarios que dañan sus configuraciones. Con NCA, los usuarios no necesitan convertirse en administradores de su computadora personal, esto se ve reflejado en la productividad ya que pueden invertir más tiempo para concentrarse en su trabajo.

Las aplicaciones NC se basan en la construcción de varios módulos pequeños llamados "**applets**" (componentes) en lugar de los grandes programas monolíticos. Estos componentes o "applets" son fáciles de construir y aun más importante, son fáciles de desplegar y dar mantenimiento. Los sistemas "**applet-based**" (basados en componentes) son más flexibles ya que para realizar un cambio no se requiere modificar el sistema completo, sino que los cambios son guardados en applets individuales.

1.7.3 Ventajas de NCA.

A continuación se listan las principales ventajas que ofrece la implementación de un sistema de información utilizando NCA:

- **Compra.** Los Web Browsers no cuestan y las computadoras de red están por debajo de los \$500.00 dólares.
- **Mantenimiento.** La administración y despliegue de la información son efectuados principalmente por administradores especializados.
- **Actualizaciones.** Las actualizaciones son principalmente desplegadas por el mismo usuario.
- **Integración.** El uso de protocolos estándar promueve la integración con los sistemas heredados.
- **Capacitación.** Solo usuarios capacitados utilizan los programas de aplicación (no administran) y solo administradores del sistema llevan a cabo la administración.

- **Skills** (habilidades). Cuando se implementa NC, los skills de la organización se ven enriquecidos: en lenguajes de programación, para operar sistemas, para administrar bases de datos o redes. También se aprovechan los conocimientos ya existentes en las personas al utilizar la tecnología ya existente como: mainframes, LAN's, PC's, redes y aplicaciones.
- **Preservación de inversiones ya instaladas.** Los clientes desean conservar sus inversiones en hardware y software ya existentes, incorporándolas a NCA.
- **Tiempo velocidad/transición.** El lapso de tiempo para realizar y desplegar soluciones de tecnología de información se acorta.
- **Seguridad contra Acceso Universal.** Una implementación NC también debe preocuparse por la seguridad concerniente al crecimiento natural del acceso universal. NC tiene el potencial para ir cerrando los malogrados agujeros de seguridad ya que ésta debe direccionarse por delante.
- **Fortaleza/escalabilidad.** Network Computing también debe evitar una trampa que ha plagado muchas aplicaciones implementadas en cliente/servidor: La escalabilidad, la cual es fácil de construir en sistemas pequeños, pero NCA debe poder crecer al tamaño de la empresa y aún más allá, incorporando también fiabilidad, integridad y resistencia, características asociadas a los mainframes.

1.7.4 Formas de Conexión.

Conexión **Statefull** (con estado). Los clientes permanecen conectados al servidor y utilizan recursos del mismo, estén o no realizando operaciones.

Conexión **Stateless** (sin estado). Los clientes abren su conexión con el servidor, realizan operaciones simples y se desconectan del servidor. El cliente usa solo recursos del servidor al realizar una operación.

En la figura 1.7.4 se observa de manera gráfica las ventajas que representa una conexión en modo stateless. Esta forma de conexión permite al servidor satisfacer a

más clientes, porque la mayoría de los clientes en tiempo real, están ociosos. La meta de los sistemas de información es el acceso a los datos, a la información misma.

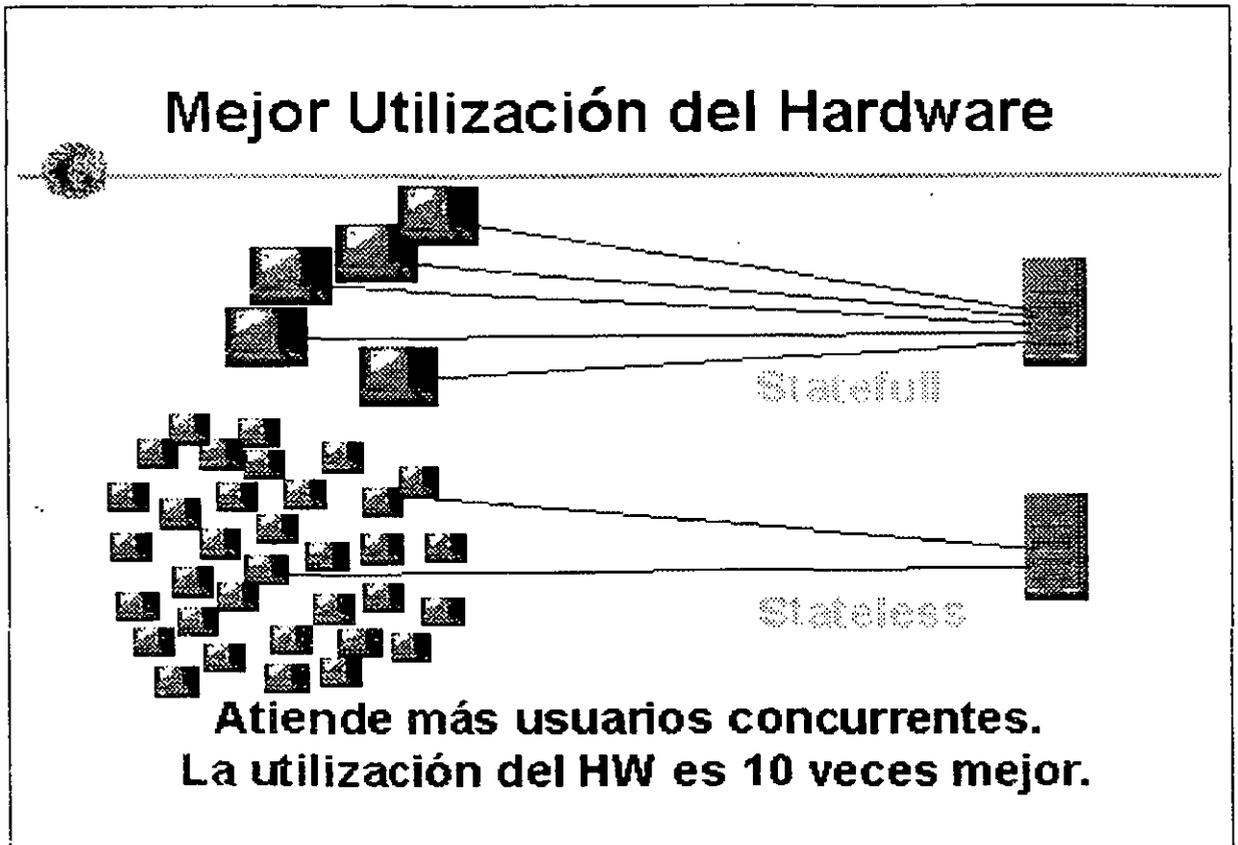


Figura 1.7.4 Optimización del Hardware Utilizando Conexión Stateless

1.7.5 Estructura de NCA.

NCA es, simplemente, lo que su propio nombre indica: una arquitectura para construir e integrar aplicaciones dentro de un entorno de computación en red. Hay que recordar que la World Wide Web es uno de estos entornos. Publicando esta arquitectura, Oracle quiere que sea lo más sencillo posible para las empresas el adoptar y desarrollar soluciones basadas en este estándar. El éxito de los navegadores universales basta, por sí mismo, para demostrar por qué tiene sentido un estándar abierto al que pueda acogerse la gente. El rápido desarrollo por parte de Oracle de esta arquitectura tan necesaria, ha situado a Oracle Corporation y sus tecnologías en el centro del

movimiento de la computación en red, y le ha permitido posicionarse como la compañía proveedora de soluciones Internet.

NCA es un modelo de tres capas o bien multicapa pero sin la complejidad. Permite la conexión de cualquier cliente (que tiene un Web Browser) a un servidor de datos, vía un servidor de aplicaciones conocido como **Web Application Server** (servidor de aplicaciones de red). De esta manera: los clientes, el web application server y el servidor universal constituyen los tres bloques fundamentales de NCA. Hay un modelo común de programación a través de la arquitectura, conectividad común, y despliegues a través de cualquier sistema operativo o infraestructura de red. El poder de NCA radica en su habilidad de construir un sistema que proporciona: Máxima expansión, escalabilidad, flexibilidad, un solo ambiente de dirección, e interfaces estándar. En la figura 1.7.5.1 se representan los tres bloques fundamentales de NCA.

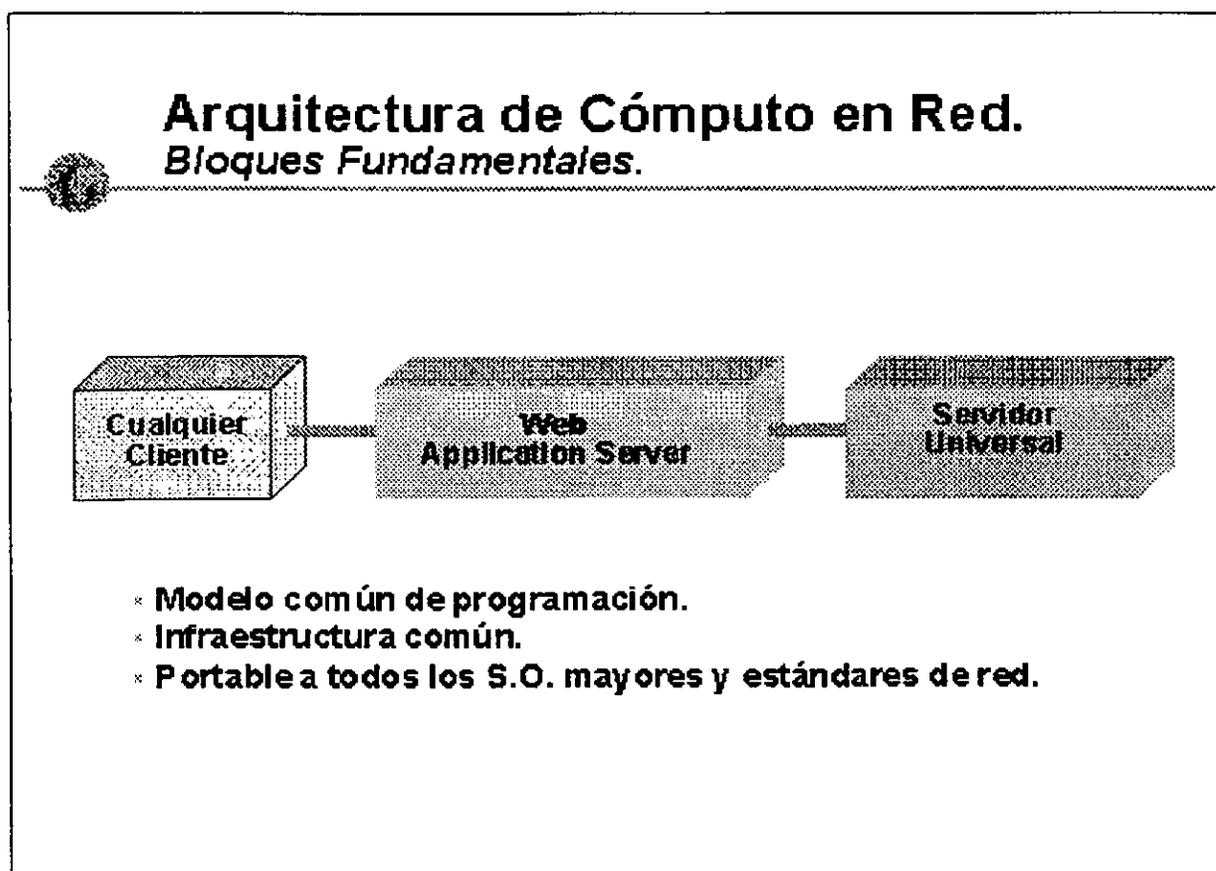


Figura 1.7.5.1 Bloques Fundamentales de NCA.

Cliente Universal.

El primer componente de NCA es el cliente universal. El dispositivo de acceso puede ir de un PC tradicional a una computadora de red. Muchas personas creen que NC significa computadoras de red, pero esto no es correcto. Muchos quieren utilizar a las computadoras de red en ambiente NCA, pero éstas no son necesarias para lograr la inmensa mayoría de los beneficios de Network Computing. Algunos usuarios necesitan todas las funciones de los PC's y algunos otros no. Para cualquiera de los dos casos, el concepto de software "**thin-client**" (cliente delgado) y acceso universal, hace conexión con un ambiente NCA, fácil, rápido y con un costo bajo.

En la figura 1.7.5.2 se describen la selección de estándares para los clientes de NCA.

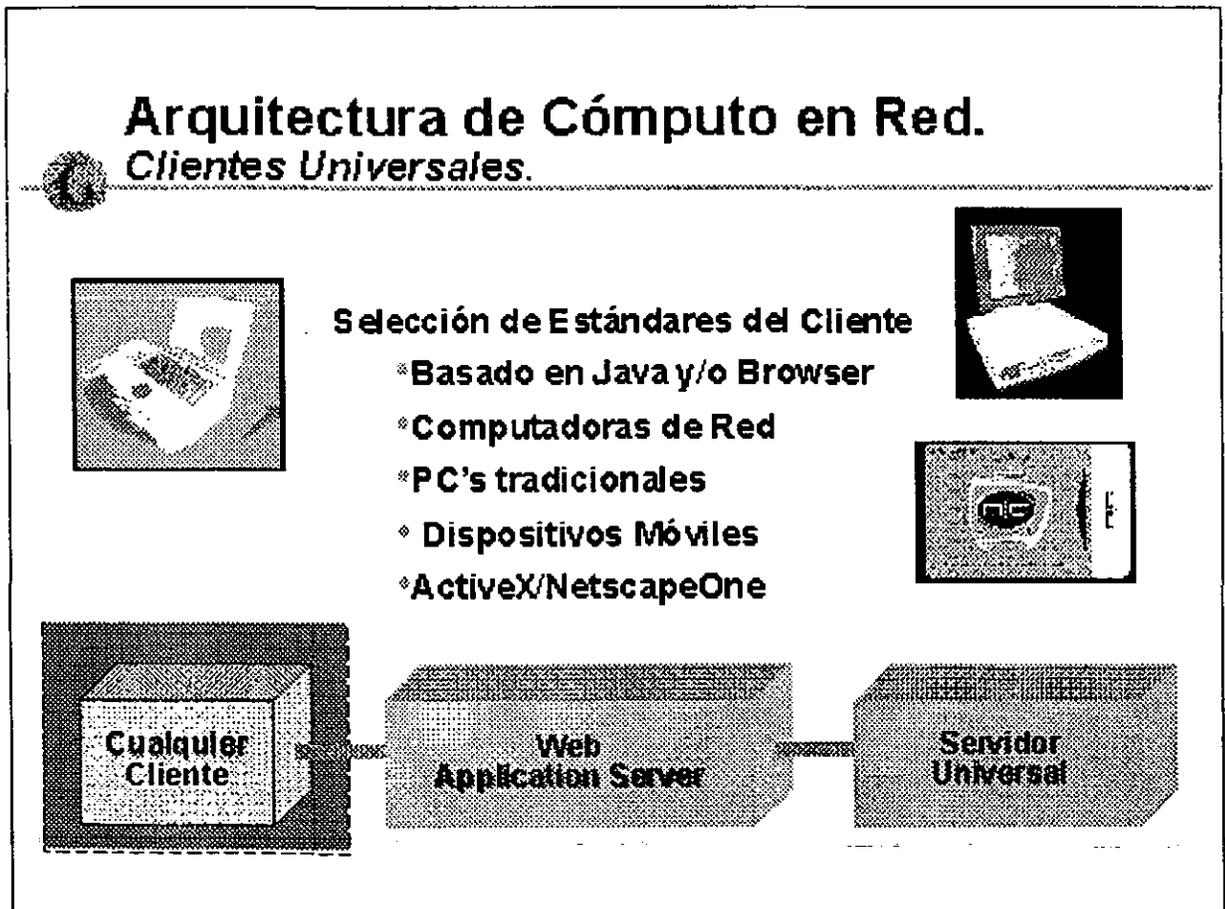


Figura 1.7.5.2 Clientes Universales en la Arquitectura NCA.

Servidor Universal.

El **Universal Server** (servidor universal) representa el poderoso **back-end** (última parte) de NCA y es el responsable de la manipulación y almacenamiento de los datos, los cuales pueden ser: texto, números o vídeo bajo demanda. El Servidor Universal proporciona la parte fuerte de los servicios de datos escalables **web-enable** (habilitado para red) los cuales son críticos para cualquier ambiente de informática de red. Los que se entregan con Oracle 8, son esto y más y conservan las inversiones que se han hecho en skills y aplicaciones para el ambiente Oracle 7.

La Figura 1.7.5.3 muestra las características del Servidor Universal de Oracle.

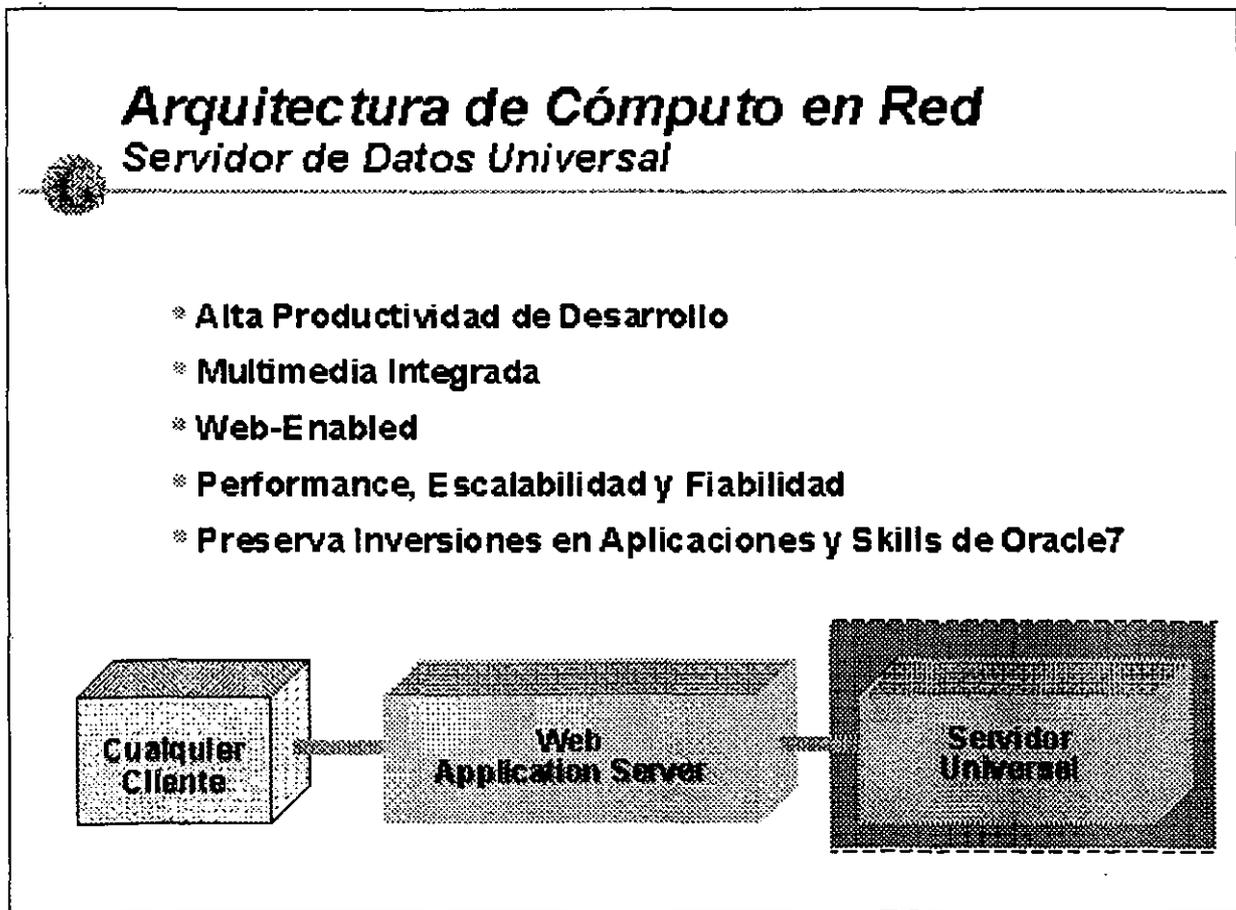


Figura 1.7.5.3 Características del Servidor Universal de Datos.

Web Application Server.

El tercer componente de NCA es el Web Application Server y constituye la nueva capa de la información. Aquí es donde las aplicaciones **line-off-business** (negocios fuera de línea) son construidas por el usuario o bien, adquiridas junto con el Web. El Web Application Server le permite desplegar aplicaciones en configuraciones muy flexibles.

El empresario puede tener servidores centrales grandes, servidores distribuidos pequeños, grupos de servidores para disponibilidad alta o cualquier combinación que tenga sentido para su negocio. Además el Web Application Server puede crecer con el uso de la aplicación. Los servidores también pueden crecer, o más servidores pueden agregarse para manejar un uso creciente de aplicaciones fuertes en el Web. En la figura 1.7.5.4 se listan las características del Web Application Server.

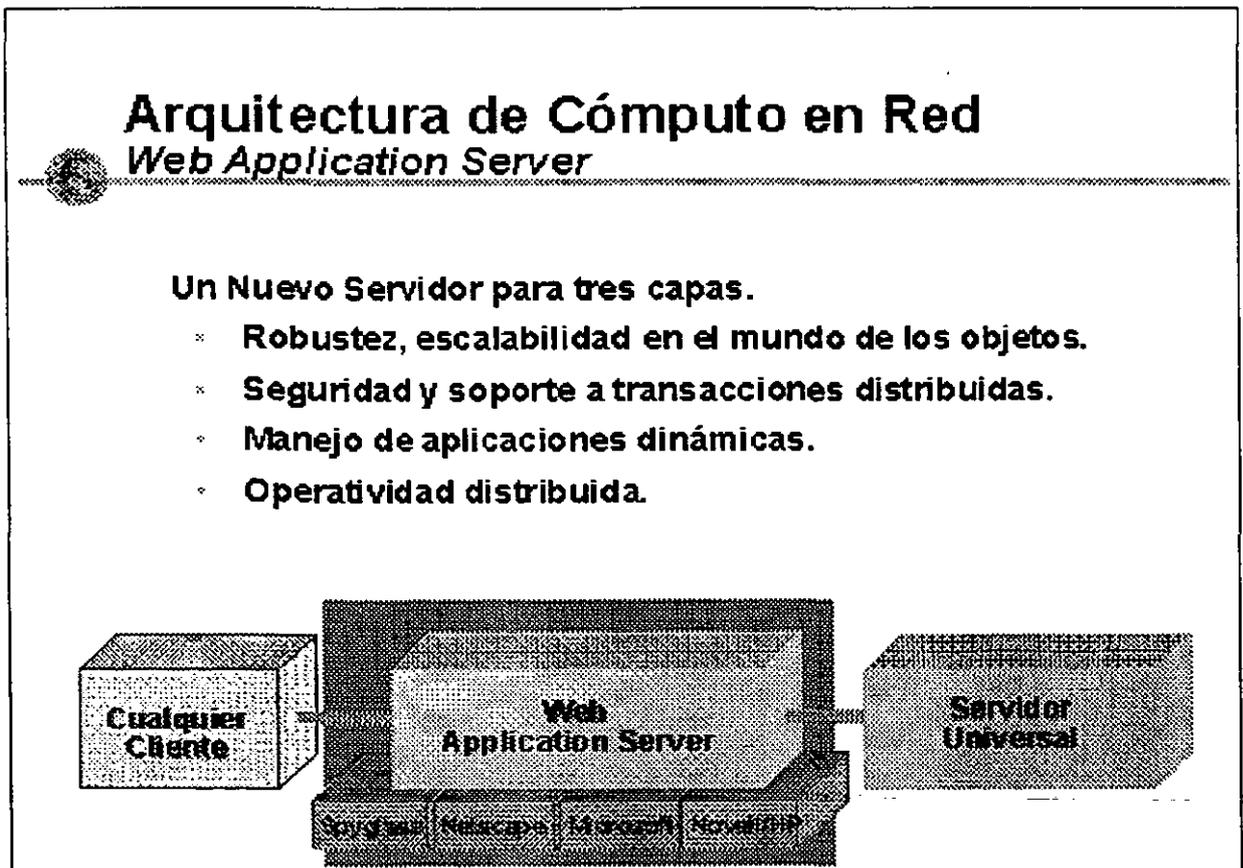


Figura 1.7.5.4 Características del Web Application Server.

El Web Application Server no duplica la función de un Web Server básico como Microsoft IIS o Netscape Webserver. Éstos también llamados HTTP-Servers son usados por el Web Application Server para acceso básico y transporte a Intranets o a Internet. Si el usuario cuenta con un HTTP-Server, el Web Application Server se instalará soportándolo. Si el usuario no cuenta con ninguno, una copia del servidor Spyglass es incluido con Web Application Server para proporcionar las funciones básicas http.

Web Application Server corre a través de una variedad de ambientes de sistemas operativos de servidor de uso popular como Unix y Windows NT.

Cartridges (Cartuchos).

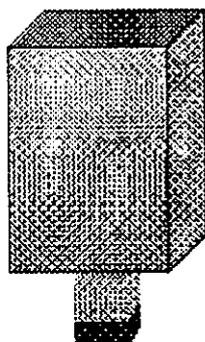
Como se mencionó antes, la Arquitectura de Computación en Red prevé un mundo en el que el servidor de aplicaciones y el de base de datos están conectados entre sí y con los clientes delgados. También prevé un mundo en que los fabricantes de software creen programas que pueden ser enchufados en el nivel del cliente, del servidor de aplicaciones o incluso del servidor de base de datos.

Los programas que corren en NCA son llamados cartridges o cartuchos. Dicho de otra manera, un cartucho es un componente de programa. Una variedad de lenguajes pueden usarse para escribir cartuchos, de este modo sus programadores no necesitan ser reentrenados. Algunos cartuchos son específicos en función del cliente; algunos son específicos en función de los datos y otros son función de la aplicación universal que puede correr a cualquier nivel de la arquitectura.

En la figura 1.7.5.5 se muestra el bloque fundamental de un cartucho

La mayoría de la gente recordará a los cartridges del cliente como una visualización del software, a los datos en los cartridges como tipos esenciales de datos extendidos y a los cartridges en los servidores de aplicaciones, como una lógica comercial.

Cartridge: Bloque Fundamental.



- ◆ Los cartridges son:
 - ◆ Aplicaciones.
 - ◆ Componentes de Aplicaciones.
- ◆ Soporta múltiples lenguajes de programación.
- ◆ Administración común y servicios a través de todas las plataformas.
- ◆ Puede combinar programas custom y off-the-shelf.

Figura 1.7.5.5 Características de los Cartridges.

Los cartridges son típicamente applets o applets múltiples, escritos para ser flexibles con el despliegue y fáciles para reemplazar, descargar o mantener.

Como se mencionó antes; habrá muchos cartridges para NCA disponibles para la inmensa serie de desarrolladores de software que escriban programas para el ambiente Oracle.

Cuando la decisión correcta sea desarrollar sus propias aplicaciones o extender aplicaciones comerciales (del estante); habrá un basto set de herramientas Oracle y herramientas de desarrollo para terceros, disponible para la construcción rápida de cartridges para NCA.

Intercambiador de Cartuchos (ICX).

La manera en que se comunica entre sí, la informática de multicapas, ha sido uno de los aspectos más complicados previos a las arquitecturas de informática. En NCA, Oracle ha usado una base de normas industriales para crear el Inter-Cartridge Exchange (intercambiador de cartuchos) de Oracle, que trabaja y escala en base a estándares y es fácil para desplegar y manejar. El soporte fundamental de esta arquitectura es precisamente el intercambiador de cartuchos (ICX), que es la capa de comunicación que admite los componentes de NCA para comunicarlos entre sí (ver figura 1.7.5.6). Es el bus que da a los cartuchos la posibilidad de "hablar" con otros. Gracias al intercambiador, un cartucho situado en cualquier punto de la red puede sacar partido de los servicios de cualquier otro cartuchos.

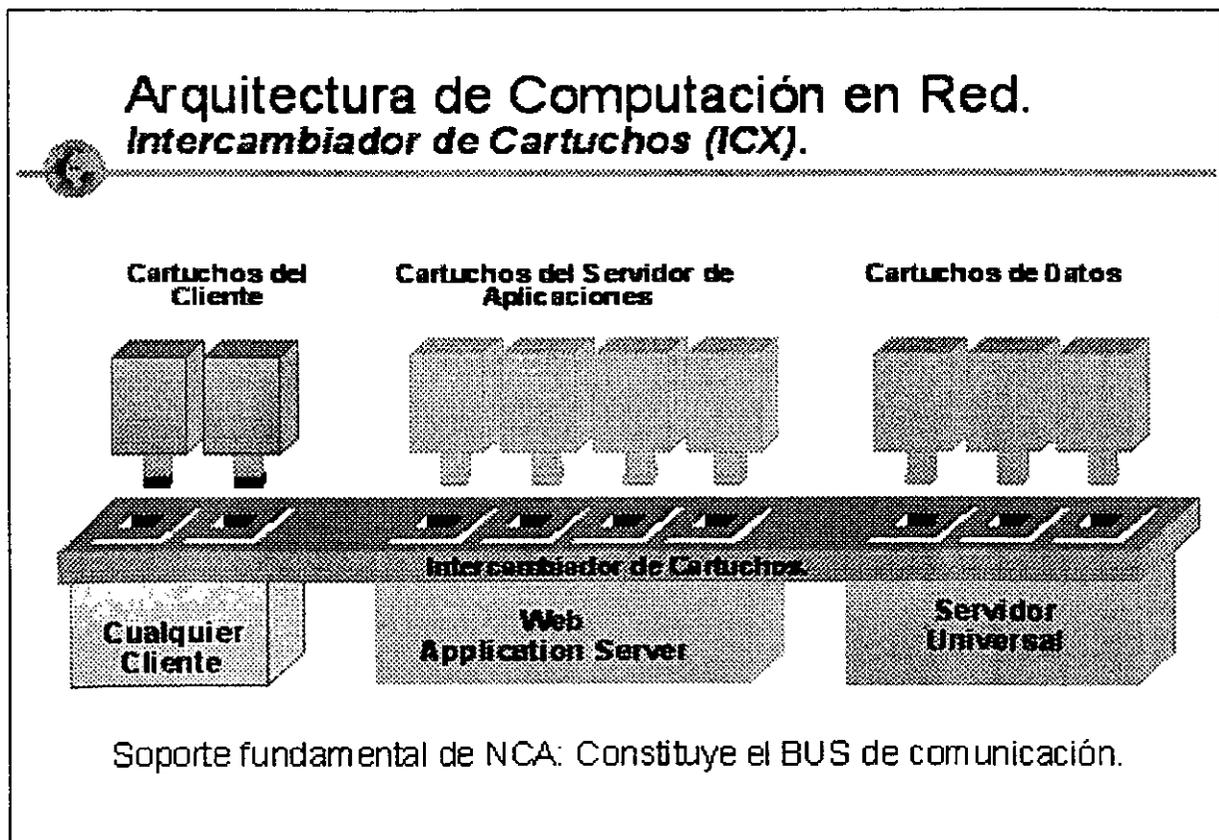


Figura 1.7.5.6 Intercambiador de Cartuchos.

1.8 WEB APPLICATION SERVER.

Uno de los campos que más auge ha tenido en los últimos tiempos es el del World Wide Web, debido a que una nueva generación de aplicaciones de negocios en Internet e Intranets están emergiendo. Estas nuevas aplicaciones incorporan transacciones de negocios reales, manejo de multimedia e información interactiva. La combinación de las bases de datos y de las aplicaciones en Web permite a las corporaciones expandir sus servicios e incrementar sus líneas de negocios mediante el uso de Internet.

Las nuevas aplicaciones del Web demandan un nuevo ambiente de trabajo que combine las características tradicionales cliente-servidor con la flexibilidad y fácil manejo de Internet.

¿Qué es un Servidor Web?

La palabra servidor implica varias cosas dependiendo de la audiencia. Para algunas personas, un servidor es hardware que siempre está administrando archivos, otros ven un servidor como un software constructor y otros lo ven como una combinación de hardware y software.

En el mundo UNIX, los servidores que contienen páginas de Web son llamados "daemons" porque siempre están ejecutando un programa gestor de comunicaciones que es lo que se suele llamar un daemon. Los daemons son programas originarios de las máquinas UNIX que se ejecutan en segundo plano (es decir, a la vez que otros programas más "normales"). La particularidad de los daemons es que se ejecutan, y nunca se detienen. Por ejemplo, para controlar las comunicaciones de correo electrónico se suele usar un daemon (el daemon sendmail) que continuamente espera que suceda algo relacionado con el envío o recepción de correo electrónico.

En un servidor de Web (sea UNIX o no) siempre se está ejecutando un daemon que controla las peticiones de páginas, y que se denomina httpd (HTTP daemon). De esto se deduce que, para usar un ordenador como servidor de WWW, éste debe correr sobre algún tipo de sistema operativo multitarea (para poder ejecutar el daemon y otras cosas a la vez). Por ello UNIX es el sistema operativo más usado para servidores Web, seguido por X-Windows, Linux (excelente adaptación de UNIX al entorno de los PC's) y Windows NT. Incluso se puede usar Windows 3.1, pero, sinceramente, no es recomendable. También es posible usarlo en Windows 95, pues este sistema operativo funciona con un mecanismo de multitarea preemptiva, es una plataforma ideal para acoger instalaciones de Internet o, en el caso que nos ocupa, de la World Wide Web.

En el caso de aplicaciones servidoras de Web (los daemons), existen varias posibilidades, de entre las cuales las dos más utilizadas son:

- CERN HTTPD

Este daemon es el original con que se ideó la Web. Es propiedad del CERN, pero su uso y copia son gratuitos (razón por la cual está muy extendido). Se puede conseguir directamente del CERN (<ftp://info.cern.ch/>) y está disponible para UNIX, Windows NT y Linux. Sus principales ventajas son su facilidad de uso y configuración y, evidentemente, su nulo costo. En la otra mano, no es aconsejable para grandes instalaciones, caso en el cual es conveniente inclinarse por algo más sofisticado (y caro).

- Netscape HTTPD

La empresa Netscape distribuye su browser gratuitamente (hasta cierto punto), pero no su software HTTPD. Este paquete es mejor que el CERN HTTPD, especialmente para servidores que vayan a soportar un tráfico intenso (caso en el cual la opción del CERN no es aconsejable). Además el daemon de Netscape tiene garantizada una

compatibilidad al cien por ciento con el navegador de Netscape Corporation, que hoy por hoy es líder indiscutible del mercado.

1.8.1 Arquitectura Conceptual del Web Application Server.

La arquitectura conceptual está dividida en tres niveles:

- Capa Baja.

La capa más baja de la red es donde se manejan las aplicaciones o conexiones de clientes HTTP. En ésta residen los servidores y despachadores; estos servidores corresponden directamente a los clientes que hacen requerimientos de documentos HTML estáticos o bien de requerimientos CGI (Common Gateway Interface). Los servidores HTTP se conectan al WRB (Web Request Broker) y proporcionan un punto de acceso para aquellos clientes que desean acceder a las aplicaciones.

- Capa Intermedia.

La capa intermedia es el WRB. Actúa como autenticador, distribuidor y coordinador de las peticiones a las aplicaciones al proporcionar la infraestructura que permite a las aplicaciones ser distribuidas a través de los servidores. Además, separa las aplicaciones de los servidores HTTP y, por lo tanto, proporcionan independencia a dichos servidores. Cualquier servidor HTTP es capaz de ejecutar cualesquiera de las aplicaciones seleccionadas, debido a que tiene los permisos de seguridad requeridos.

- Capa Alta.

La capa más alta es la de aplicación; aquí es donde la aplicación lógica y los servicios del sistema están implementados. Los cartridges (cartuchos) son extensiones del servidor, escritas a un API (Application Program Interface). Además de estar

implementados como objetos CORBA (Common Object Request Broker Architecture) y proporcionar los servicios que permiten al WRB funcionar como aplicación de la plataforma. Los cartridges del sistema ejecutan funciones tales como autenticación, resolución de nombres, etc. Contienen la aplicación lógica. Los desarrolladores que construyen aplicaciones para el Web utilizando el Oracle Web Application Server implementan sus aplicaciones como cartridges.

1.8.2 Arquitectura Física del Web Application Server.

Existen tres componentes en la implementación física al igual que en la arquitectura conceptual:

- Listener / Dispatcher,
- WRB,
- Application Cartridges.

Hay además un componente adicional llamado **Cartridge Factory** (Fábrica de Cartuchos), el cual es un subcomponente del WRB y es el responsable de iniciar y detener los Application Cartridges.

El escenario típico involucrado en el proceso que se ejecuta cuando un cliente hace un requerimiento es el siguiente:

Al recibir la petición del cliente de la capa del servidor HTTP, el dispatcher determina cual cartridge necesita manejar ese requerimiento y para ello usa el cartridge del sistema de servicios (VMP); luego, autentifica el requerimiento utilizando el cartridge de autenticación (Auth). Una vez autenticado, el dispatcher obtiene una instancia del Broker, el cual asigna un cartridge que maneja la petición. Finalmente, envía el requerimiento así como la referencia del objeto a ese Cartridge para que sea ejecutado.

A continuación se describen los tres componentes que integran la arquitectura física del Web Application Server:

1. Listener/Dispatcher/ADP.

La función de la capa del listener/dispatcher es aceptar los requerimientos HTTP de los clientes. Cada listener tiene un pequeño dispatcher y un adaptador ligados a su proceso servidor. El adaptador es conectado en el proceso servidor con el uso del API nativo de ese servidor específico, tal como el NSAPI para servidores Netscape y el ISAPI para Microsoft IIS. Los adaptadores mapean el requerimiento del servidor HTTP nativo en un protocolo independiente (Adapter/Dispatcher, ADP). El módulo del dispatcher acepta y genera llamadas ADP. Todas las comunicaciones con el WRB son manejadas a través del módulo dispatcher.

Para responder a un requerimiento, el dispatcher debe tener un cartridge de recursos disponible y libre. El dispatcher tiene dos subcomponentes: un manejador de recursos (RM) y un manejador de ejecución (EXEC).

El RM es el responsable de manejar las listas de cartridges disponibles y el estado de éstos.

El manejador EXEC toma la referencia de un objeto y es el encargado de la ejecución de los requerimientos dentro del cartridge de recursos identificado por la referencia del objeto.

Si el dispatcher recibe un requerimiento de un Application Cartridge el cual no tiene ó no existe un recurso disponible, le pide al WRB que cree una instancia del Application Cartridge. En la figura 1-8.2.1, se pueden observar los elementos que intervienen en un requerimiento al Web Server, los eventos que se disparan y la secuencia en la que se presentan.

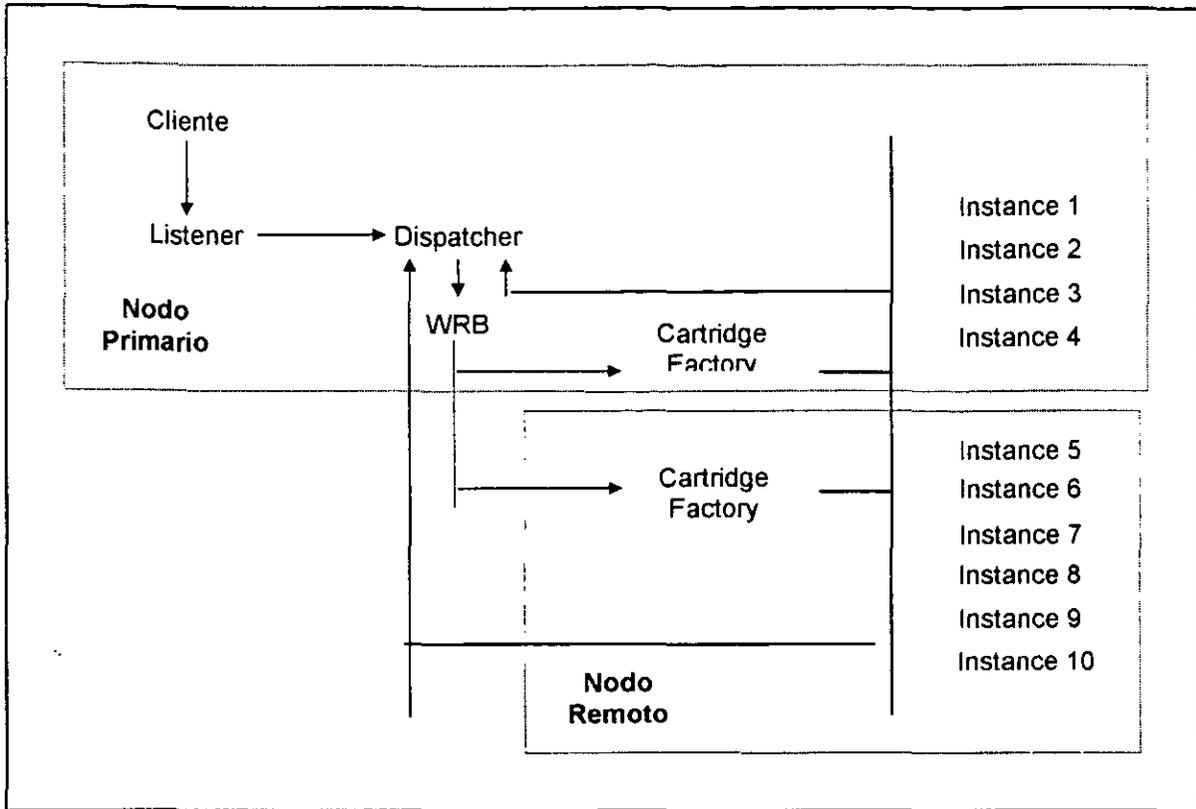


Figura 1.8.2.1 Flujo de Eventos del Web Application Server

2. Web Request Broker.

EL WRB es el corazón del sistema completo. Consta de un complemento CORBA, ORB (Object Request Broker), y un número de servicios del sistema. El WRB es la suma de esos servicios y el ORB.

EL ORB.

Al ORB en la versión 3.0 se le conoce como Oracle Media Exchange 3.3.2. El protocolo de comunicación utilizado en este ORB es 'Media Net'.

El ORB consta de tres componentes, los cuales ejecutan diferentes tareas.

- ORB address server - mnaddrsrv
- RPC name server - mnrpcmnsrv
- ORB service - mnorbsrv

El servidor de direcciones (address server) es responsable de ejecutar la resolución de nombre/dirección requerida cuando un requerimiento es hecho a través del ORB.

El dominio Media Exchange ORB es el proceso que implementa la funcionalidad del ORB. Ordena los requerimientos y respuestas de los distintos servidores y clientes, y pasa la referencia de los objetos a través del sistema.

El modo de operación distribuido del Web Application Server está habilitado a través de la capa de ORB y permite que los servicios sean implementados como objetos CORBA, esto es, que se ejecuten en diferentes máquinas perteneciendo a la misma instancia del Web Application Server.

Cada uno de los objetos distribuidos del WRB se comunican con el ORB central vía el protocolo Media Net. Este protocolo está implementado en la parte alta del protocolo UDP (Universal Data Packet). El UDP es un protocolo de comunicación el cual es más rápido; sin embargo, este tipo de protocolos no tienen mecanismos de recuperación de errores, por lo que el Media Net se encarga de reordenar y re-ensamblar así como manejar las retransmisiones de los paquetes faltantes. Mientras que el ORB proporciona la infraestructura para que los objetos y los componentes sean capaces de ejecutarse y comunicarse con su respectiva capa de red o sistema operativo.

Los tres deben ser iniciados antes de que cualquier proceso de WRB pueda ser levantado. No necesariamente deben estar corriendo en el mismo nodo que el WRB, o inclusive en el mismo nodo los tres juntos. Se comunican a través del UDP y utilizan la variable de ambiente OMN_ADDR, la cual debe especificarse como sigue: UDP:HOST_NAME:PORT. Se puede consultar en el archivo:

%OWA%/ows/site_name/wrb/config/omnaddr. El %OWA% hace referencia a la variable ORAWEB_ADMIN, la cual puede ser:

\$ORACLE_HOME/ows/admin, o bien

\$ORACLE_BASE/admin/

dependiendo del tipo de instalación Non-OFA/OFA.

En una configuración típica dichos procesos se ejecutan en el mismo nodo que el WRB. Para verificar que estos procesos se estén ejecutando, se utiliza el siguiente comando:

```
$ORAWEB_HOME/bin/mnorbis
```

Es necesario especificar la variable LD_LIBRARY_PATH a \$ORAWEB_HOME/omx/lib así como la variable OMN_ADDR antes de ejecutar el mnorbis. Esto mostrará todas las interfaces ORB definidas ejecutándose. Deben existir al menos 5 entradas clasificadas como mnorbis y 4 entradas del wrb (negociadas como wr). Si sucede algo diferente, quiere decir que hay algo extraño con el ORB y es necesario reiniciarlo.

Los siguientes servicios son parte del WRB. Son implementados como objetos CORBA y definen sus métodos mediante el empleo de IDL (Interface Definition Language).

Wrbroker. El Broker o Resource Manager, proceso maestro del wrb, también utilizado como servidor de nombres. Mantiene un registro del estado de cada recurso en la instancia completa del Web Application Server.

Este servicio es el responsable de crear las instancias del Application Cartridge. En una configuración 'multi Cartridge', este servicio se encarga de la distribución de las instancias de los cartridges a través de cada servidor disponible.

Este servicio mantiene una lista global de recursos tanto de Application Cartridge como de System Cartridge. Para llevar el control del estado de los recursos, mantiene dos listas: la de objetos libres y objetos ocupados. Estas listas son globales en el sistema. Cada dispatcher de ser registrado en el Manejador de Recursos Globales (Global RM). Este es responsable de mantener al máximo y mínimo valor de cartridges. Si el número de cartridges disponibles es mayor que el mínimo configurado del sistema, contacta al Cartridge Factory y pide que se elimine las instancias de Cartridges.

Wrbfac. Cartridge Factory, genera procesos de cartridges. El propósito de este servicio es crear y destruir las instancias físicas de los Application Cartridges. Además, es el único capaz de detener e iniciar instancias de Cartridges. En una configuración multi servidor, este servicio debe de estar presente en cada nodo.

Wrbc. Proceso cartridge instance, copia del proceso wrbfac.

Wrbypm. The Virtual Path Manager. La función de este servicio es determinar si el requerimiento que proviene de un cliente HTTP es para un Application Cartridge conocido. Mapea el path virtual recibido en el servidor HTTP a un Application Cartridge mapeando el path físico. Este servicio es utilizado primeramente por el dispatcher para determinar si el requerimiento debe ser manejado por el WRB o por el servidor HTTP.

Wrbcfg. Config Provider. Es el que proporciona los servicios de manejo de la configuración distribuida. Al iniciar lee el archivo wrb.app para cuando lo necesite, además proporciona los métodos con los cuales otros servicios pueden obtener información de configuración.

Wrblog. Log server, provee servicios de 'log' y de error tanto para los procesos de WRB y como en los Application Cartridges. Se tiene un conjunto de métodos que permiten a otros servicios escribir mensajes y errores, ya sea archivos o a la base de datos. Puede ser configurado para operar ya sea en memoria o bien para operar como

un servicio de ORB, permitiendo ser distribuido. Todos los componentes del WRB utilizan el Log Server.

Wrbahsrv. Authentication Host Adapter, implementa un método particular de autenticación y de autorización; ejemplo, filtrando la dirección IP.

Wrbasrv. Authentication Broker, autentifica y autoriza las peticiones.

Los servidores wrbahsrv y wrbasrv proporcionan los esquemas de seguridad con los cuales los recursos pueden ser protegidos. Los esquemas disponibles son:

- Basic
- Digest
- IP
- Domain
- Basic Oracle

Cada esquema de seguridad está implementado como un objeto compartido separado. Este diseño permite que esquemas adicionales de seguridad sean desarrollados e implementados como módulos. La sección del wrb.app que define cada uno de estos esquemas es: AuthService.

Wrbdm. Limpia otros procesos del wrb, a partir de la versión 3.0.1, busca y elimina procesos 'errantes'.

Oraweb. Proceso del listener, en versiones 3.0.0 existen dos de estos procesos. El primero es el del listener actual; el segundo hace direccionamientos DNS. En la versión 3.0.1, existen tres procesos, el tercero se encarga de limpiar y monitorear.

El proceso oraweb debe ser el último que se levante. El wrb dispatcher se liga con el oraweb. La primera vez que se inicia el listener de administración, usando el comando

owsctl start admin, existirán dos procesos oraweb ejecutándose. Cada vez que se definen más listeners en el sistema, se levantarán más procesos oraweb. Existe solamente un ORB y un WRB; por lo tanto, el wrb no es necesario que se detenga y se levante cada vez que un listener se levante o se detenga.

3. Application Cartridge.

En esta capa es donde las aplicaciones lógicas se desarrollan y despliegan. La aplicación es desarrollada utilizando un conjunto abierto de API's. Cuando se ejecuta, cada Application Cartridge existirá como un proceso del sistema operativo distinto, el cual es capaz de responder a los requerimientos de los clientes a través de WRB.

Gracias a esta implementación si un cartridge falla, debido a que son completamente independientes uno de otro y también de la capa del WRB, no afecta a ningún otro. Un 'core dump' en un Cartridge sólo afectará esa instancia, y no tendrá efecto en ninguna otra ya sea perteneciente a ese mismo o a otros cartridges, o cualquier componente del WRB.

En conclusión, el Oracle Web Application Server es un software utilizado para desarrollar aplicaciones para el Web.

El sistema está construido e implementado en la parte alta del ORB creando varios Cartridges de sistema los cuales actúan como servicios del Web Request Broker. Estos servicios permiten a los desarrolladores generar Applications Cartridges que son manejadas por el Web Request Broker.

En la figura 1.8.2.2 se ilustra la arquitectura del Servidor Web Application Server.

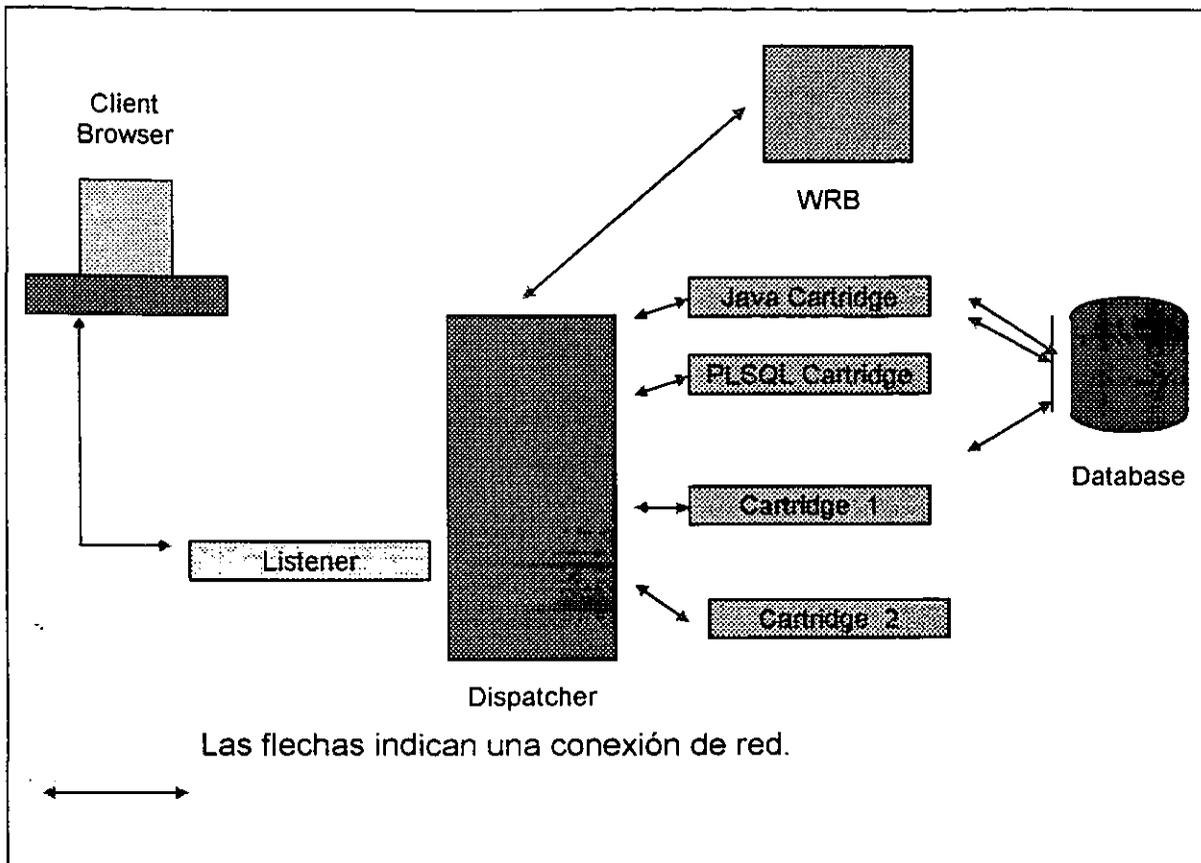


Figura 1.8.2.2 Arquitectura del Web Application Server.

1.8.3 Seleccionando un Web Server.

Para seleccionar un Servidor Web, es necesario saber cuales son los requerimientos del cliente o empresa, donde se desea instalar, y a partir de ello comenzar a tomar decisiones. La primer consideración, es qué servidores son válidos para el tipo de plataforma que se desea manejar. A continuación se listan algunas plataformas que son válidas para Servidores Web:

- UNIX (cualquier versión)
- Linux
- Windows NT
- Windows 95

- Windows 3.x
- OS/400
- Macintosh
- Novell NetWare
- OS/2
- VMS
- Amiga
- VM/CMS

UNIX es la plataforma más robusta para servidores Web. De cualquier manera, si no se tiene habilidad en UNIX, Windows NT es equiparable a UNIX en lo que a soporte para Internet se refiere y está considerado a ser una buena plataforma para servidores Web.

Windows NT tiene la ventaja de su familiaridad con el popular Windows 3.1 ó 95, que lo hacen relativamente sencillo de usar. Por otro lado, su potencia es bastante elevada. Sin embargo, sus prestaciones siguen por debajo de las de algunas implementaciones de UNIX. Probablemente Windows NT esté llamado a convertirse en el sistema operativo de los servidores de Internet montados en PC, pero no hay que olvidar otras opciones, que no por ser menos populares y recibir menos promoción son menos potentes.

Macintosh es otra opción en plataformas que permite el buen funcionamiento de los servidores Web que manejan pequeños sitios Web.

Por otro lado está el crecimiento popular de Linux, que no es más que una implementación de UNIX para ordenadores PC. Se trata de un sistema operativo gratuito, que es usado en países tan diferentes como Estado Unidos o Pakistán. Su principal ventaja es que se trata de una implementación fiel de UNIX con soporte para NFS, lo cual siempre es una garantía. Por otro lado, se trata de un sistema operativo gratuito. Linux viene con un completísimo juego de utilidades gestoras de Internet,

entre ellas servidores de casi cualquier servicio, clientes, y soporte para CGI's usando el lenguaje Perl.

En cambio, no todo es tan bueno. Si Usted instala un servidor con Windows NT y algo falla, Microsoft responde. En cambio, Linux es un proyecto sin ánimo de lucro. Si algo falla, mala suerte, pero no busque responsables, pues no existe "Linux S. A.", ni nada que se le parezca. Sin embargo, en su defensa cabe decir que Linux es más estable que la mayoría de los sistemas operativos para PC, y que su nivel de calidad es, cuanto menos, similar al de éstos.

Mientras UNIX ha sido considerado el rey en el mercado de los servidores Web, la situación esta cambiando. La IDC (International Data Corp.) reporta que el porcentaje de los 50,000 servidores Web que se equiparon con UNIX en el año de 1994 fué del 80% al 90%. Para 1996, la IDC reporta que de los 200,000 servidores Web que se compraron el 60% al 70% se equiparon con Windows NT, Windows 95 y OS/2. Una razón para esta tremenda ausencia de UNIX es que los usuarios UNIX están más decididos a cambiar de plataforma que los usuarios de Windows NT, Windows 95 u OS/2. Adicionalmente, el mercado de los servidores ha evolucionando cada vez más, presentando más y mejores alternativas, así como ha ido creciendo el número de programadores de sitios Web que no trabajan en UNIX.

En la tabla 1.8.3.1 se puede observar una lista de servidores Web que se encuentran disponibles en Internet, clasificados por plataforma.

Plataforma	Servidor	Dirección en Internet
UNIX	Apache	http://www.apache.org/apache
	CERN	http://www.navisoft.com
	NCSA	http://ftp.ncsa.edu/web/httpd
	Netscape Commerce	http://home.netscape.com
	Plexus	http://bsd1.com/server/doc/plexus.html

Plataforma	Servidor	Dirección en Internet
Windows 3.x	Secure WebServer	http://www.openmarket.com
	Spinner	http://spinner.infovav.se
	WebServer	http://www.openmarket.com
	WN	http://hopf.math.nwu.edu/docs/manual.html
	EIT http	http://wsk.eit.com/wsk/doc
	Phttpd	http://ftp.lysator.liu.se/pub/phttpd (Sun Solaris 2.x)
	WEB4HAM	http://ftp.infomatic.uni-hamburg.de/pub/net/winsock/web4ham.zip
Windows NT	Windows httpd	http://www.city.net/win-httpd
	SerWeb	http://winftp.cica.indiana.edu/pub/pc/win3/winsock/serweb03.zip
Windows NT	Alibaba	http://www.csm.co.at/csm/alibaba.htm
	FolkWeb	http://www.ilar.com/folkweb.htm
	HTTPS	ftp://emwac.ed.ac.uk/pub/https.txt
	NaviServer	http://www.navisoft.com
	NetPublisher	http://www.notis.com
	Netscape Communications	http://home.netscape.com
	Purveyor	http://www.process.com
	SerWeb	ftp://enwac.ed.ac.uk/pub/serweb_i.zip
	EMWAC	ftp://emwac.ed.ac.uk/pub/https
	WebSite	http://website.ora.com
Windows 95	Alibaba	http://www.csm.co.at/csm/alibaba.htm
	Folk Web	http://www.ilar.com/folkweb.htm
	Purveyor	http://www.process.com
	Web Site	http://website.ora.com
OS/400	Web Server/400	http://www.inetmi.com/products/webserve/webinfo.htm
OS/2	GoServer	http://www2.hursley.ibm.com/goserve
	OS2HTTPD	ftp://ftp.netcom.com/pub/kf/kfan/overview.html
Macintosh	InterServer Publisher	http://www.intercon.com
	MachHTTP	http://www.biap.com
	WebSTAR	http://www.starmine.com
	Mac Common Lisp Server	http://www.ai.mit.edu/projects/iip/doc/ci-http/home-page.html
VMS	CERN	http://delonline.cern.ch/disk\$user/duns/doc/vms/distribution.html

Plataforma	Servidor	Dirección en Internet
	Purveyor	http://www.process.com
	Region 6	http://kcgl1.eng.ohio-state.edu/Web/doc/serverinfo.html
Amiga	Amiga Web Server	ftp://max.physics.sunysb.edu/pub/amiga/amosaic
NetWare	Purveyor	http://www.process.com
	GLACI-HTTPD	http://www.glaci.com/info/glaci-httpd.html
VM/CMS	VM/CMS Server	http://ua1vm.ua.edu/-troth/software/cmshttpd.html

Tabla 1.8.3.1 Servidores Web por Plataforma.

1.8.4 Hardware Requerido.

Los requerimientos de hardware para un servidor Web varían por plataforma. Más aún, la necesidad de una computadora potente con suficiente memoria depende en gran parte en la popularidad de los sitios de Web. Otro factor a considerar es si el servidor corre sobre alguna máquina de servicios de información Internet, como son Gopher, FTP o servicio de Listas o si el servidor Web correrá sobre una computadora que actualmente es usada por la empresa. Más aún las consideraciones de seguridad, uno debe considerar que el servidor de Web correrá en una máquina que corre actualmente algún software de seguridad y que esto puede alterar el **performance** (calidad) de la máquina.

La elección del hardware es probablemente la decisión más importante de todas las referentes a Internet y su empresa. ¿Cuántas veces se ha conectado a una página de Web, y la ha dejado por culpa de la lentitud del servidor? Conviene que adquiera una máquina capaz de servir todo el tráfico que su red sea capaz de generar. Sin embargo, tampoco tiene sentido comprar una máquina por encima de sus necesidades, ya que la inversión perdería rentabilidad. Hoy por hoy, los servidores más populares suelen ser de dos tipos: PC's de alto rendimiento o estaciones de trabajo.

- PC.

Usar una PC como servidor de Web y punto de enlace con una Intranet (redes locales privadas que trabajan internamente con protocolo TCP/IP) es una decisión arriesgada, sobre todo si la instalación que ha de abastecer es mínimamente grande. Los PC's se diseñaron para usos domésticos, no para este tipo de tareas. Piense que un servidor de Internet recibe mucha carga, ya sea en peticiones de páginas de Web desde el exterior como en cuestiones internas en una red local. Por ello, si su red es de tamaño mediano a grande debería planearse la compra de un aparato más veloz.

Por otro lado, bien es cierto que los PC's de hoy en día se parecen cada vez más a ordenadores más potentes, ya que las prestaciones de los PC's modernos han subido enormemente. Un PC de gama alta le dará un rendimiento no muy inferior a algunas estaciones de trabajo, a un coste teóricamente inferior. Por ello los PC son adecuados para controlar instalaciones de tamaño reducido.

Otro inconveniente que tienen los PC's para actuar como servidores de Internet es su sistema operativo. No olvidemos que casi todo el trabajo de desarrollo de Internet se ha hecho sobre el sistema operativo UNIX o variantes. Por ello, es normal que la red esté diseñada "pensando en UNIX", y no en los sistemas operativos de PC. Quizás las dos únicas opciones recomendables sean Windows NT, que es equiparable a UNIX en lo que a soporte para Internet se refiere, y Linux.

- Estación de Trabajo.

Las estaciones de trabajo no son mas que ordenadores potentes, que pueden tener uno o más microprocesadores funcionando a la vez. Poseen una capacidad de cálculo bastante superior a los PC's (la relación va desde 2:1 a más de 20:1). Suelen emplear variantes del sistema operativo UNIX, que dependiendo del fabricante se llama de una u otra manera, y tienen sus propias características diferenciales. Su coste es, evidentemente, más elevado que el de un PC, pero también lo son sus prestaciones.

Las estaciones de trabajo más populares son las de Sun Microsystem, seguidas por un conjunto de marcas como Silicon Graphics, Digital Electric Corporation u otras. En cualquiera de ellas el equipo suele venir configurado especialmente para trabajar como servidor de Web, ya que las estaciones de trabajo no son útiles sólo para Internet: se trata de ordenadores polivalentes de gran potencia, válidos tanto para aplicaciones de cálculo intensivo como para ejecutar software gráfico.

Últimamente, han aparecido en el mercado estaciones diseñadas específicamente para ser usadas como servidores de Internet. Se trata de modelos más o menos normales, a los que se les ha añadido soporte para los diversos protocolos de red local existentes. Además suelen venir equipados con software administrador de instalaciones de Internet, tales como servidores de HTTP, FTP, NNTP, clientes de la mayoría de servicios, y utilidades de administración y control del sistema.

Evidentemente, una workstation es mucho más interesante que un PC en cuanto a prestaciones se refiere. Sin embargo, su precio aumenta en una relación casi directa. Es decir, una estación de trabajo que sea el cuádruple de rápida que un PC costará más o menos el doble que una que sea dos veces más rápida que el mismo PC. Por ello su uso es especialmente interesante para nodos que soporten mucho tráfico, ya sea interno (proveniente de una Intranet) o externo, pidiendo páginas de Web.

Por otro lado, una estación de trabajo es bastante más compleja de administrar que un PC. Se requiere personal calificado para manejarlas, ya que la mayoría de ellas trabajan con un sistema operativo UNIX, que es sumamente potente, pero nunca se ha caracterizado por su sencillez de uso.

Sea como sea, a partir de un cierto tamaño de instalación no basta con un PC, y deberá darse el salto al mercado de workstations, con todo lo bueno y lo malo que ello conlleva. Sólo una workstation es capaz de gestionar el tráfico generado por un Web y una Intranet de tamaño mediano a grande.

Una vez que se ha decidido por un hardware en particular, el siguiente paso es la configuración de la máquina. Generalmente, los vendedores nos sugieren software para configurar el hardware de los servidores. De cualquier modo, después de seguir las recomendaciones de los vendedores, es mejor preguntar a algún **Webmaster** (experto en instalaciones de servidores Web) qué configuración aplican ellos y cómo ellos perciben la calidad del servicio. Las configuraciones sugeridas por los vendedores en ocasiones varía con respecto a las configuraciones de los clientes. La tabla 1.8.4.1 compara las recomendaciones de los vendedores y críticos, y la lista de configuraciones propias de los clientes.

Recomendaciones Microsoft	Recomendaciones de Críticos	Recomendaciones Implementadas
486-based system	486-based system	486-based system
16 MB RAM	32 MB RAM	132 MB RAM
90-110 MB espacio en disco	1 Gigabyte SCSI-based en disco	2.5 GB SCSI-based en disco
486-based system	486-based system	486-based system
CD-ROM	CD-ROM	CD-ROM
No usar unidad de cinta de respaldo	Usar unidad de cinta de respaldo	Usar unidad de cinta de respaldo

Tabla 1.8.4.1 Comparación de las Recomendaciones de Hardware para Windows NT

La configuración inicial de la máquina podrá aliviar la necesidad de actualizar nuestro equipo hasta dentro de un año, a menos que el flujo de trabajo del servidor se incremente y haya la necesidad de realizar algunas modificaciones.

1.8.5 Comunicaciones Requeridas.

Más importante que el poder de una workstation es el ancho de banda de la línea de comunicaciones. De nada nos sirve una estación de millones si no tenemos una conexión a la red fiable. En esto también disponemos de diversos niveles de conexión, de coste y velocidad diferente. Por un lado, si su instalación es pequeña (por ejemplo, un PC funcionando como servidor de Web) puede confiar la comunicación a un

modem. Tiene la ventaja de ser bastante económico, pero son muy lentos. Esto se debe a que las líneas telefónicas que utilizan son las de la red telefónica básica, que son analógicas y, por lo tanto, no soportan grandes velocidades.

En el otro lado del espectro tenemos las líneas de gran velocidad, que van desde los 64 Kbits por segundo de una línea de RDSI básica hasta cables tipo T3 a 54 Mbits por segundo. Si la transmisión de la línea es de 1 MB por minuto se recomienda una línea de 512 Kbps, si la transmisión es de 300,000 bits por día se recomienda una conexión T1 de 1.544 Mbpp, pero si la carga de trabajo es ligera se recomienda una línea de 56 Kbps. En términos de costo, los precios para una conexión T1 en los Estados Unidos es de \$2,000 a \$2,700 mensuales con instalación y sistema de honorarios entre \$8,000 y 15,000, mientras que una línea de 56Kbps tiene un costo aproximado de \$300 a \$400 dólares mensuales.

Debe usted tener en cuenta lo que necesita, ya que tan erróneo sería tener un servidor de Web muy potente con un cable muy lento, como tener un servidor muy pequeño con cable T3. Como puede ver, estamos relacionando continuamente la velocidad de la conexión con el tipo de servidor. Habitualmente los servidores de tipo PC funcionan a base de uno o más modems, o RDSI básica en el mejor de los casos. En cambio, si dispone de un servidor tipo workstation lo lógico es ir a buscar cables de 128 Kb por segundo o más.

1.8.6 Seguridad Web.

La seguridad del Web fue diseñada para proveer tres capas de seguridad: por usuarios, por usuarios de grupo y para el resto de los usuarios.

La figura 1.8.6.1 muestra las capas de seguridad del servidor de Web e ilustra las tres capas que forman a este:

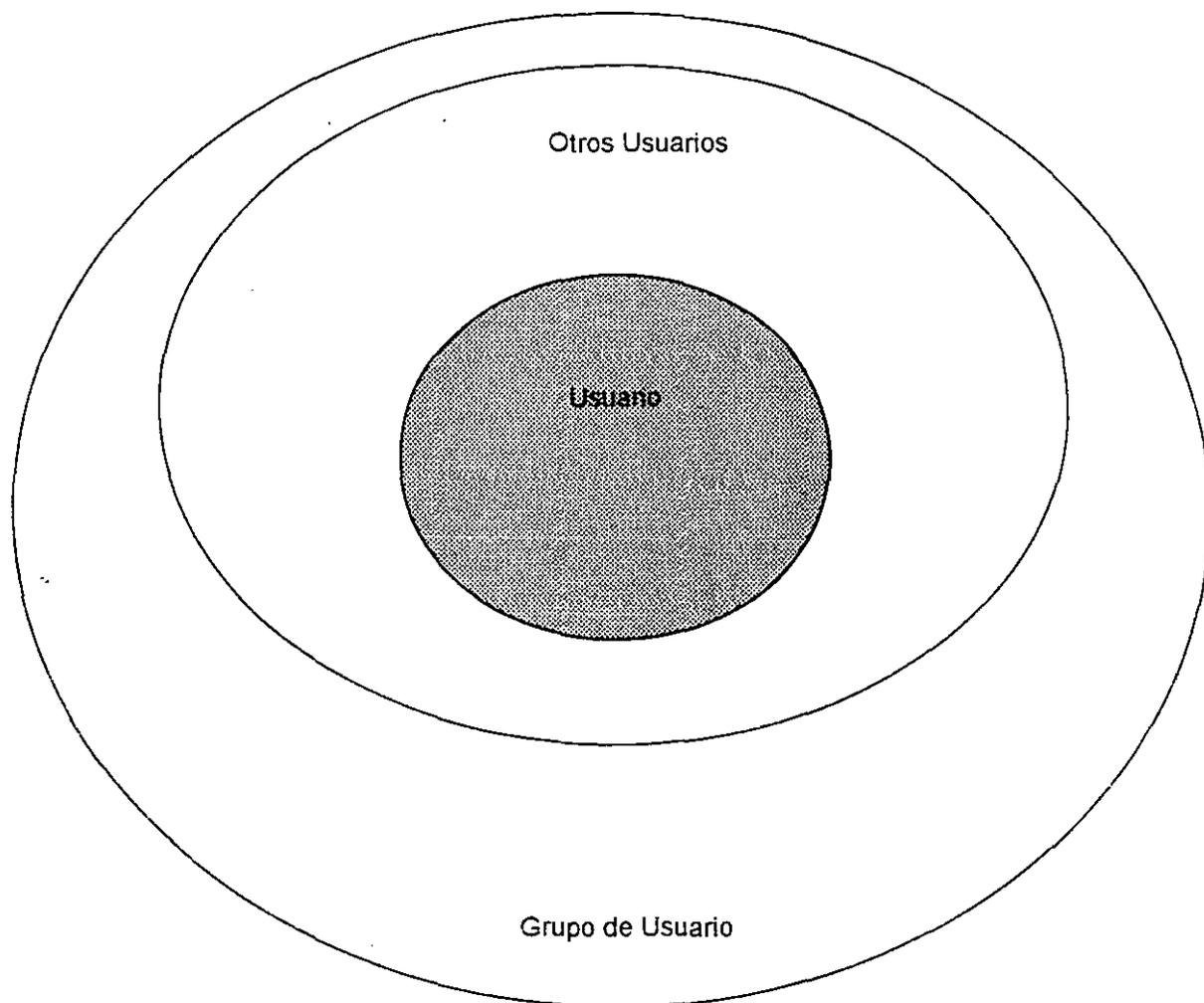


Figura 1.8.6.1 Capas de Seguridad del Web

1.9 METODOLOGÍA DE INGENIERÍA DE SOFTWARE.

La metodología del desarrollo de sistemas es el camino que siguen los analistas de sistemas al realizar su trabajo. Se emplea el término genérico de analista de sistemas para describir a la persona que tiene la responsabilidad principal de conjuntar los componentes estructurales, dándoles forma y sustancia en conformidad con las fuerzas del diseño para construir sistemas de información exitosos. En una compañía pequeña, el analista quizás no sólo diseñará el sistema de información, sino que también hará la programación y operará la computadora. En una compañía grande, el analista de sistemas puede preparar las especificaciones del diseño que se dan a los técnicos, como los programadores, ergonomistas, diseñadores de formas y especialistas en comunicaciones.

Fases Principales de la Metodología del Desarrollo de Sistemas.

Las fases principales de la metodología del desarrollo de sistemas son: el análisis, el diseño general, la evaluación, el diseño detallado, y la implementación de sistemas. En la figura 1.9.1 se muestra un esquema general de éstas. Dentro de cada fase se incluyen las principales actividades o tareas.

Las primeras cuatro fases están dirigidas a proporcionar valores específicos para los componentes estructurales. Algunos autores, compañías y consultores dividen su metodología de desarrollo de sistemas en fases o etapas diferentes a las presentadas en este trabajo, sin embargo independientemente del número o nombres de las fases o etapas, la metodología del desarrollo de sistemas racionaliza y asigna una rutina al proceso de construcción de sistemas de información. Su sello distintivo lo forman sus fases discretas.

La meta principal de la metodología del desarrollo de sistemas es reducir los inicios falsos, reciclamiento indebido, retrabajos y callejones sin salida.

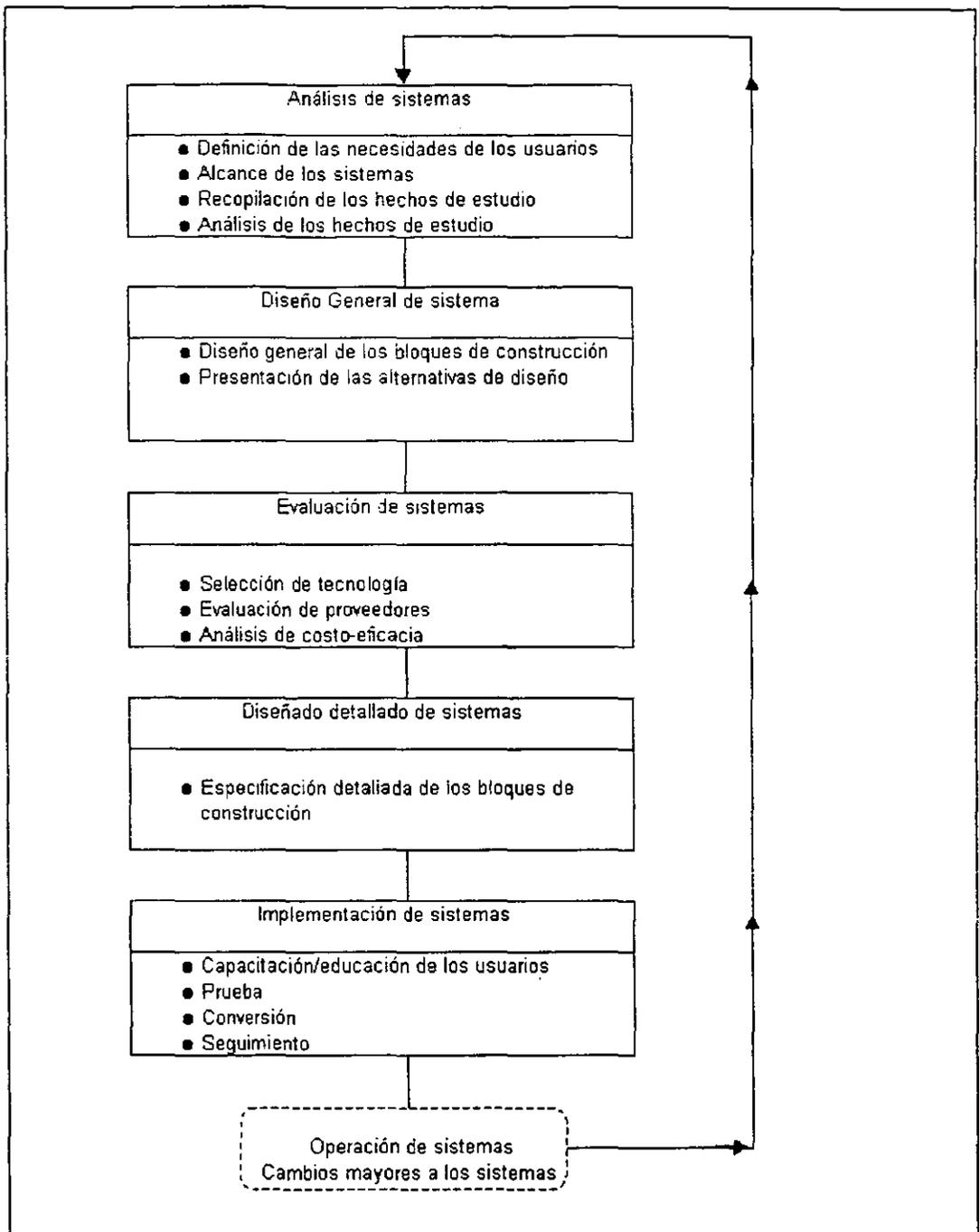


Figura 1.9.1 Metodología del Desarrollo de Sistemas.

Además, aumenta la probabilidad de que el sistema que se construya e instale finalmente sea el que los usuarios desean y necesitan.

Pero de ninguna manera la metodología del desarrollo de sistemas deberá ahogar la creatividad o la innovación.

Proporciona una guía para el recorrido a lo largo de un sistema de información. El que tan excitante, productivo y creativo desee el analista de sistemas que sea este recorrido depende, en gran medida, del propio analista.

Definición de la Metodología del Desarrollo de Sistemas.

Durante la fase del análisis de sistemas, los analistas de sistemas ayudan a los usuarios a identificar qué información se necesita. Se llevan a cabo varias entrevistas y plantean preguntas como : ¿ Qué información está usted recibiendo actualmente ? ¿Qué clase de información necesita para realizar su trabajo?.

Sin embargo, en este punto los analistas de sistemas deben ser más que escuchas pasivos o "mecanógrafos," deben ser líderes y maestros. Con frecuencia los usuarios no tienen una idea completa de cuál es la información que realmente necesitan o cómo pueden obtenerla.

En la mayoría de los casos, los analistas deben interactuar y determinar con los usuarios cuáles son sus verdaderas necesidades. Después de un buen número de entrevistas, observaciones y muestreos, los analistas de sistemas empiezan a conjuntar muchos hechos del estudio para un análisis posterior. Hacia el final de la fase de análisis, los analistas conceptualizan diseños generales y gradualmente van construyendo un modelo.

Antes de iniciar el diseño general de sistemas, la fase de análisis debe completarse y los usuarios deben estar de acuerdo con los resultados. El diseño es el proceso de traducir los requerimientos definidos durante el análisis a varias alternativas de diseño para la consideración de los usuarios. También implica descubrir todas las fuerzas de

diseño para ver cómo impactarán e influirán en los diferentes diseños. Los usuarios pueden responder a las alternativas de diseño y empezar a trabajar hacia un consenso. Sin embargo, pueden ocurrir varias iteraciones entre el análisis y el diseño, en que cada iteración implique malabear y dar nueva forma a los componentes estructurales.

En la fase de diseño, los usuarios seleccionan dos o tres de los mejores modelos o le piden al analista que regrese al restirador para desarrollar un mejor modelo. Si se recopilaron y analizaron suficientes hechos del estudio en la fase de análisis, se reduce la probabilidad de tener que regresar a dicha fase.

También se debe recalcar que durante el diseño los analistas de sistemas con frecuencia empezarán a hacer uso de técnicos como: científicos de la administración, ergonomistas, expertos en robótica, programadores, especialistas de formas y expertos en comunicaciones.

En cualquier caso, los analistas concluyen la fase del diseño general con un entendimiento bastante claro de lo que se desea y de lo que tendrán que hacer para obtenerlo. La evaluación de sistemas en realidad no es una fase discreta; en todo momento durante el trabajo en sistemas se realiza una gran cantidad de evaluaciones, pero para resaltarla, aquí se presenta como una fase separada.

Esta fase implica: la selección de la tecnología que soportará a los otros componentes estructurales, la evaluación de esta tecnología, los proveedores que la suministran y un análisis completo del costo-eficacia de cada una de las opciones propuestas del diseño de sistemas para determinar aquella con la mejor proporción de eficacia en relación al costo.

Los analistas no desean tener que recorrer el camino largo y difícil hacia el diseño detallado de sistemas teniendo varias opciones de diseño aún en el aire.

Antes de iniciar este trabajo sustancial, los analistas desean asegurarse de que todas las partes hayan llegado a un consenso final en cuanto al sistema que desean que implemente. Aquí a cada componente de construcción se le da una definición precisa y detallada.

A decir verdad, sería un desperdicio excesivo de tiempo y dinero, por no mencionar la frustración y consternación, diseñar en detalle un sistema sólo para encontrar finalmente que nadie lo quiere y que corresponde a las fuerzas del diseño. La implementación de sistemas es la capacitación y educación de los usuarios, la prueba y conversión para hacer que el sistema sea operacional.

Aquí es donde todo el trabajo de desarrollo y diseño llega a un clímax. Los analistas fijan las fechas límite o puntos de congelación, capacitan y coordinan al personal usuario, instruyen a los técnicos, prueban el nuevo sistema y eliminan errores, instalan nuevos procedimientos y formas, y ven si hay descuidos u omisiones. En algún punto, el nuevo sistema de diseño es aceptado por el personal que lo operará y administrará bajo una base cotidiana.

Poco después de su aceptación, el analista de sistemas deberá realizar un seguimiento para ver que el sistema está operando según lo esperado. En muchos casos, los analistas pueden hacer algunos ajustes menores a afinaciones que resulten en mejoras significativas. Después de que el sistema haya estado operando durante cierto tiempo, los usuarios cambiarán y los analistas de sistemas se verán involucrados en la preparación de prototipos especiales para cubrir necesidades particulares.

En ocasiones será necesario efectuar algún trabajo de mantenimiento. Finalmente, en algún momento, tiempo después de que el sistema haya estado en operación, el analista de sistemas se verá de nuevo, y entonces se repetirá el ciclo de vida de la metodología del desarrollo de sistemas.

Puede ser necesario que los analistas desarrollen un subsistema con base en una solicitud especial. Para corresponder a esta solicitud los analistas pueden trabajar en la terminal del usuario para ayudar a diseñar formatos de pantalla alternos de acuerdo a las especificaciones del usuario. Esta interacción y consulta de uno-a-uno con el usuario se denomina elaboración de prototipos y, en gran medida, es un subconjunto de la metodología del desarrollo de sistemas pero se centra en una aplicación para un usuario (o unos cuantos usuarios) y su alcance no es amplio.

La sesión de elaboración de prototipos no dura más de una hora y produce un modelo de una aplicación. Con el usuario presente e interactuando con el analista de sistemas, se puede obtener en la pantalla un bosquejo general que sea bastante parecido a lo que el usuario desea. El usuario y el analista de sistemas diseñan, prueban, modifican y rediseñan conjuntamente los prototipos hasta que el usuario elige uno para su implementación posterior.

El papel de los analistas de sistemas y su empleo de la metodología del desarrollo de sistemas es, por supuesto un cuadro compuesto. En una organización grande el sistema de información es una entidad vasta y compleja, y muchos analistas están involucrados en el diseño e implementación de sólo subsistemas de todo el sistema de información de la organización.

En un momento dado, un analista podría muy bien estar realizando actividades relacionadas con más de una fase de la metodología del desarrollo de sistemas, y estas actividades se deben coordinar adecuadamente.

Para que el sistema sea más eficaz, el analista de sistemas participará en un amplio espectro de actividades que van desde las formales cuantitativas a cualitativas, estructuradas a no estructuradas, específicas a generales. Al realizar estas actividades, el analista puede hacer uso de diagramas de flujo, diagramas de flujo de datos, tablas de decisión, matrices, gráficas, reportes narrativos, entrevistas, modelos

y prototipos. Estas técnicas se refuerzan entre sí y, cuando se ven en conjunto, proporcionan las herramientas básicas para el trabajo en sistemas.

Después de la fase de análisis de sistemas, en muchos casos se ve con bastante claridad lo que se debe hacer.

Opciones de Desarrollo.

Los analistas de sistemas cuentan por lo menos con tres opciones básicas, cada vez que se evalúa un conjunto de requerimientos de los usuarios. El analista puede recomendar que no se haga nada, que se modifique un sistema existente o que se diseñe un nuevo sistema.

1.- No hacer nada. En toda decisión de sistemas referente a la forma de satisfacer los requerimientos de información de los usuarios o las solicitudes de mejora en los sistemas, el analista tiene la oportunidad de recomendar que no se tome ninguna acción en ese momento. Las razones para elegir esta opción incluyen: (a) una pobre identificación de los requerimientos y necesidades, (b) la determinación de que no es factible desarrollar un sistema significativo o una solución a las necesidades de los usuarios, (c) otras solicitudes de sistemas con mayor prioridad o debido a que los recursos para el desarrollo ya se han asignado completamente, o (d) las necesidades de los usuarios, según se ha planteado, no son necesidades reales.

2. Modificación de un sistema existente. La mayoría de todas las investigaciones de sistemas que se realizan en las organizaciones incluyen alguna consideración de los sistemas y subsistemas existentes. Para satisfacer requerimientos nuevos de los usuarios, el analista con frecuencia recomienda la modificación de los sistemas existentes en lugar de diseñar sistemas nuevos. Dependiendo del tamaño de la organización y del subsistema en particular que se está evaluando, las modificaciones

de sistemas pueden tener un mayor impacto sobre la organización que el desarrollo de un subsistema totalmente nuevo.

Este impacto puede resultar del tamaño del esfuerzo dedicado a los sistemas o del cambio que se está dando en la organización. Cuando se aplica un sistema como soporte a la solución de un problema organizacional, el énfasis está en los resultados inmediatos. De esta forma, con frecuencia se implantan cambios a los sistemas existentes hasta que se pueda definir y desarrollar un nuevo sistema. Adicionalmente, el nivel de desarrollo de sistemas de información que existe actualmente en muchas organizaciones de tamaño mediano a grande, ha llegado a un punto en que las nuevas demandas de los usuarios con frecuencia requieren cambios relativamente pequeños a la captura de datos y elementos de almacenamiento, y el énfasis se da en el acceso a los datos disponibles en un nuevo formato o en una forma más oportuna.

3. Diseño de un nuevo sistema. La opción final a la disposición del analista consiste en que este analista recomiende un nuevo sistema para satisfacer los requerimientos de los usuarios. Esta opción es obviamente la solución más compleja y difícil de implementar. Esta puede verse como una combinación de dos elecciones de acción posteriores. Cuando un analista recomienda que se implemente un nuevo sistema, se debe tomar la decisión acerca de si el sistema se va a desarrollar desde su principio o si se va a comprar de otras fuentes un sistema aceptable. Tradicionalmente esto se reconoce como la decisión de "hacer-o-comprar".

Definición del Diseño.

El diseño general de sistemas puede definirse como el dibujo, planeación, bosquejo o arreglo de muchos elementos separados en un todo viable y unificado. En tanto que la fase de análisis de sistemas responde a las preguntas de lo que está haciendo el sistema y de lo que debería estar haciendo para satisfacer los requerimientos de los usuarios, la fase del diseño de sistemas se ocupará de la forma en que se desarrolla el

sistema para satisfacer estos requerimientos. La fase del diseño de sistemas está orientada técnicamente al punto en que los analistas deben contestar a la pregunta: ¿Cómo lo vamos a hacer?. Por otra parte, el diseño es un arte, orientado en forma creativa.

La Mente de Diseño.

En su trabajo, los analistas de sistemas poseen y analizan tres mentes: la de análisis, la de diseño y la de evaluación. Pero estas mentes no están separadas como tales, se combinan y se mezclan en donde es apropiado, una está más enfocada y activa que las otras dos. A medida que se analizan los hechos de estudio y se revisan los hallazgos, en la mente de diseño empiezan a cristalizarse conceptos claros de diseño.

El analista de sistemas formula y convierte estas ideas de diseño en modelos de diseño empleando algunas de las técnicas de modelación, como los diagramas de flujo de datos, los diagramas de entidad-relación y los diagramas de estructura. Ya sea en papel o en plantilla, estos modelos se visualizan, se evalúan y se vuelven a dibujar hasta que son apropiados y factibles. Las opciones de diseño se malaborean mentalmente y se evalúan aún más hasta que algunas de ellas empiezan a sobresalir de las demás. Los detalles que no son críticos se dejan a un lado para su solución durante la fase del diseño detallado de sistemas. Algunas de las ideas de diseño más pertinentes que se generan con lluvia de ideas y entrevistas de grupo, también pueden accederse y utilizarse en este momento, así como mezclarse y combinarse con los modelos de diseño.

En algunos casos, cuando la mente de diseño no genere ni proporcione al analista de sistemas ideas de diseño claramente definidas, si esto sucede, el analista de sistemas deberá concentrarse en los hechos de estudio y en los requerimientos de los usuarios hasta un punto de agotamiento.

Proceso de Diseño y Presentación de Opciones de Diseño.

El proceso de diseño general de sistemas se ilustra en la figura 1.9.2. El analista de sistemas conoce los requerimientos de los usuarios, el alcance de los sistemas y los recursos disponibles. Las fuerzas de diseño se consideran y se ponderan para determinar sobre los componentes estructurales y los diseños de sistemas finales.

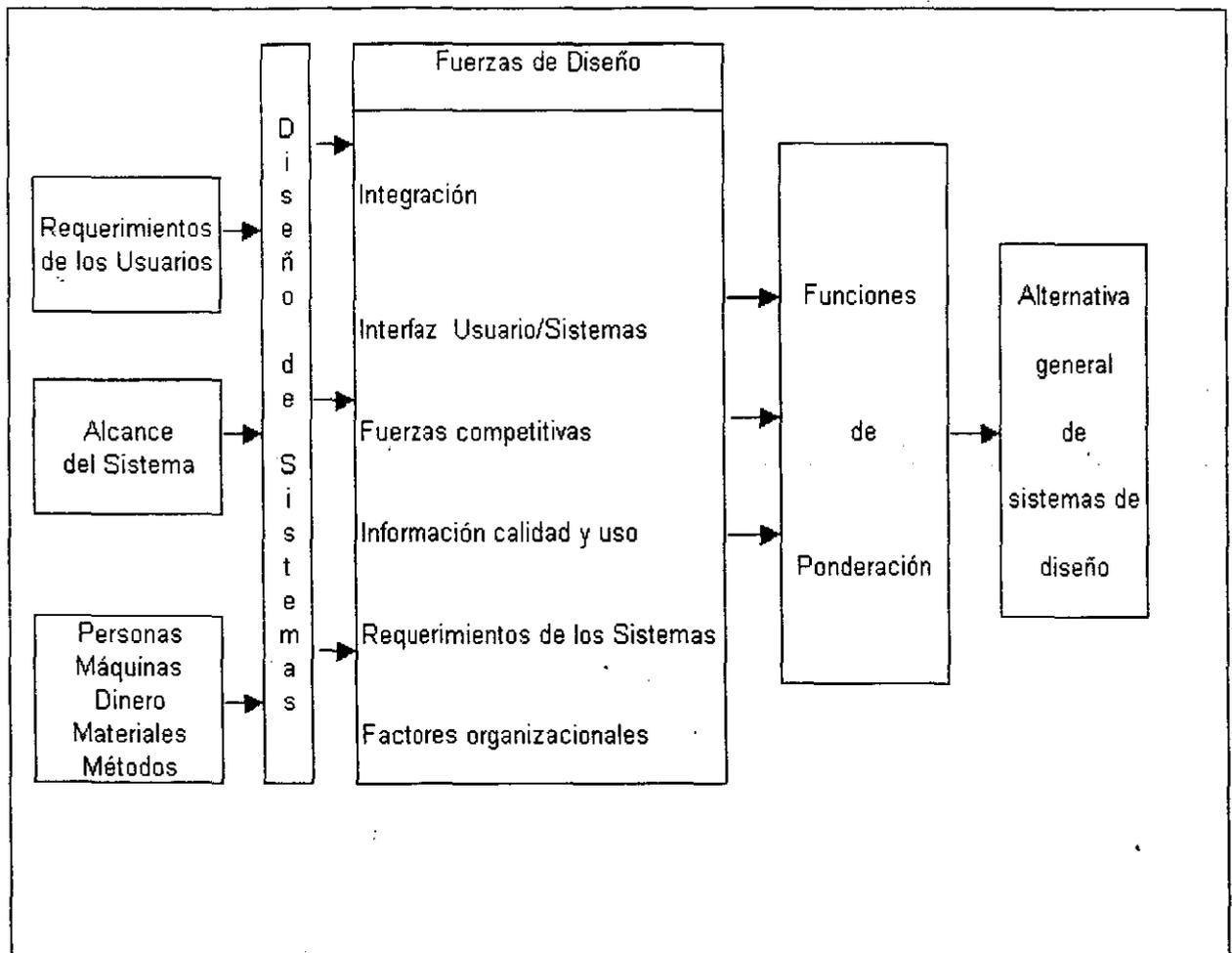


Figura 1.9.2 Proceso General del Diseño de Sistemas.

Se crean diferentes opciones de diseño para tomar en cuenta diferentes funciones de ponderación. Por ejemplo, en una opción se puede dar mucho peso a la integración, a la interfaz usuario/sistema, a las fuerzas competitivas y a los factores humanos. En

otra opción se puede dar más peso a los requerimientos de sistemas y a los requerimientos de costo-eficacia.

Con base en las fuerzas de diseño se pueden crear diversas opciones de diseño que se someten a la consideración de los usuarios. Por ejemplo, una opción puede estar basada en una topología de red en estrella, en tanto que otra puede basarse en una topología de anillo. Una opción puede extenderse en el ambiente, como en los sistemas de los clientes, y otra puede manejar los pedidos en forma tradicional. Una puede ofrecer grandes capacidades de filtrado, monitoreo e interacción; otra puede ofrecer pocas.

Una opción de diseño puede incluir un componente estructural de tecnología compuesta de muchos avances recientes, en tanto que otra sólo incluya la tecnología establecida. O bien, las opciones de diseño pueden originarse de la decisión entre rentar o comprar equipo, o de la decisión entre desarrollar paquetes de software o comprarlos a proveedores de software.

El reporte de la propuesta del diseño general de sistemas se prepara para comunicar a la gerencia y a los usuarios de la organización la forma, a nivel general, en que el sistema diseñado podrá satisfacer sus requerimientos de sistemas de información y procesamiento de datos. Las siguientes guías se ofrecen para ayudar al analista a preparar el reporte de la propuesta del diseño general del sistema.

El punto es que el analista de sistemas es un profesional y presenta a los usuarios diversas opciones de diseño de entre las cuales pueden elegirse algunas para una evaluación más profunda. El analista de sistemas trata de cubrir todas las contingencias dando a los usuarios la oportunidad de ver diferentes caminos para satisfacer sus requerimientos. La presentación de opciones aumenta la probabilidad de que finalmente se implemente el diseño correcto, ya que en muchos casos los usuarios realmente no saben lo que desean.

Preparación del Reporte de la Propuesta del Diseño General de Sistemas.

El reporte de la propuesta del diseño general de sistemas se prepara para comunicar a la gerencia y a los usuarios de la organización la forma, a nivel general, en que el sistema diseñado podrá satisfacer sus requerimientos de sistemas de información y procesamiento de datos. Las siguientes guías se ofrecen para ayudar al analista a preparar el reporte de la propuesta del diseño general de sistemas:

1. Plantear nuevamente la(s) razón(es) para iniciar el trabajo en sistemas, incluyendo los objetivos específicos. Relacionar todos los requerimientos originales de los usuarios y los objetivos con la propuesta actual de diseño de sistemas.
2. Preparar un modelo conciso pero completo del diseño de sistemas propuesto. Tratar siempre de incluir opciones de diseño de entre las cuales la gerencia pueda elegir, en vez de presentar sólo un enfoque, además con frecuencia se puede determinar que una opción particular tendrá un impacto significativamente diferente sobre la organización. Por ejemplo, la propuesta de diseño B puede cubrir un 90 por ciento de los requerimientos de la propuesta del diseño A, pero B puede costar solamente un 40 por ciento del costo de A. El analista nunca debe entrar en una situación en la que deba elegir entre un diseño particular o nada.
3. Mostrar todos los recursos requeridos para implementar y mantener cada opción.
4. Identificar toda suposición, crítica o problema no resuelto que pueda afectar al diseño final del sistema.

Ciertamente, el formato del reporte de la propuesta del diseño de sistema está sujeto a una amplia variación de una organización a otra. Sin embargo, el punto principal que debe tenerse presente al preparar una propuesta de diseño en la que las personas que deban de autorizar el desarrollo de una de las opciones o una combinación de las

mismas debe tener suficientes elementos sobre los cuales basa su decisión. En efecto, el analista de sistemas está haciendo una invitación a las críticas y sugerencias de los usuarios al presentar el reporte de la propuesta para el diseño general de sistemas.

Esta propuesta obviamente tiene una probabilidad elevada de que las opciones del diseño general de sistemas tendrán que modificarse varias veces antes de que los usuarios acepten finalmente las que consideren que cubrirán completamente sus requerimientos.

Bosquejos y Prototipos.

A menos que la clase de proyectos de sistemas que se está desarrollando sea tradicional, como un sistema básico de contabilidad, los usuarios no siempre podrán definir sus requerimientos en forma adecuada y precisa. Simplemente no pueden especificar sus requerimientos de manera previa, tienen que descubrirlos. Tres técnicas de análisis y diseño que ayudan a este proceso circunstancial son; el bosquejo de componentes estructurales, el bosquejo en papel en blanco y la elaboración de prototipos.

Bosquejos de Componentes Estructurales.

La reunión de todos los componentes estructurales en una hoja de papel, o en una pantalla para cada opción de diseño, le dan al analista un bosquejo y una vista general del sistema total.

Las opciones pueden bosquejarse para que los usuarios presenten sus reacciones y ofrezcan una retroalimentación valiosa a los analistas de sistemas, éstas hojas se convierten en una parte importante del reporte de la propuesta del diseño general de sistemas.

Bosquejo en Papel en Blanco.

El bosquejo en papel en blanco es una técnica de diseño que tiene sus raíces en los prototipos, la simulación y la lluvia de ideas. A los usuarios potenciales simplemente se les da una hoja en blanco o una pantalla y se les pide que bosquejen lo que desean como salida. Los usuarios, con una mínima ayuda y guía de los analistas, bosquejan elementos como tablas, diagramas y el contenido de la salida deseada.

Los bosquejos iniciales pueden ser rudimentarios e incompletos. Sin embargo, con el desarrollo de cada nuevo bosquejo tiene lugar un aprendizaje iterativo. De hecho, un proceso de diseño bastante complejo puede iniciarse con bosquejos y progresar hasta representaciones con mayor detalle a partir del cual se puede implementar el bosquejo final. Y lo más importante de todo, es que los analistas de sistemas pueden asegurarse en cierto modo de los sistemas que implementan serán los que los usuarios desean y necesitan.

Elaboración de Prototipos.

Una de las peores cosas que le pueden suceder a un analista de sistemas es diseñar e implementar un sistema que los usuarios no desean ni necesitan. Todas las técnicas y conceptos presentados hasta este momento ayudan a evitar que se presente este problema. Durante la planeación estratégica de sistemas de información a los analistas de sistemas se les da una guía y misiones específicas de sistemas. En el análisis de sistemas se recopilan hechos de estudios de varias fuentes. Estos hechos se analizan y se estudian a conciencia.

Se hacen presentaciones de reportes de terminación y propuestas de análisis para que los usuarios proporcionen sus críticas y evaluación. Se resuelven las diferencias y se logran acuerdos.

Los analistas de sistemas, contando con una visión aún más clara y más detalles específicos, inician la fase del diseño general de sistemas y desarrollan varias opciones de diseño con base en lo que se ha aprendido hasta ese momento. Los usuarios tienen nuevamente una oportunidad de aceptar, rechazar o combinar opciones de diseño, mostrando diferentes formas de salida, como tablas y gráficas. A través de cada iteración, los prototipos se mejoran hasta que satisfacen con precisión los requerimientos.

Se pueden crear varios prototipos en una terminal. La necesidad de una programación clásica de computadoras se minimiza gracias a la interacción en línea con las bases de datos relacionales y los lenguajes de consulta, lenguajes fáciles de usar para la creación de modelos, teclas de definición de menús y una variedad de formas de pantalla.

Obviamente, el ciclo de vida de los prototipos es mucho más corto que el ciclo de vida de las metodologías de desarrollo de sistemas (MDS). La metodología para la elaboración de prototipos generalmente no es tan formal como la de la MDS, pero aún así también sigue una metodología.

En realidad la elaboración de prototipos es un microcosmos de la MDS, excepto que la primera produce un modelo de trabajo físico del diseño propuesto, en tanto que los diseños generales de sistemas propuestos por la MDS normalmente son más conceptuales y más amplios en alcance.

Si el prototipo aceptado es parte de un sistema más grande, entonces se integra en el diseño general de sistemas y se desarrolla de acuerdo a las fases restantes de la MDS. Sirve como base para un conjunto completo y válido de requerimientos de los usuarios, y la MDS mantiene el avance del proyecto de sistemas. Por otra parte, si el prototipo aceptado es un sistema único o independiente que pueda implementarse, entonces el

analista de sistemas realiza las tareas necesarias para convertir directamente el nuevo prototipo en una operación total.

¿No es suficiente todo este trabajo en sistemas para evitar que se implemente un sistema malo? No necesariamente. ¿Porqué? Debido a que los analistas de sistemas con experiencia saben que, en algunos casos, "lo que los usuarios piden no es lo que desean y lo que desean no es realmente lo que necesitan".

A decir verdad, algunos usuarios simplemente no saben qué necesitan hasta que lo ven en términos físicos y trabajan con él. Ahí es donde entra en juego la elaboración de prototipos. Esta técnica es más avanzada que el bosquejo de los componentes estructurales y el bosquejo en papel en blanco por el hecho de que el usuario trabaja efectivamente con una intención del sistema que se va implementar.

Si a los usuarios no les gusta, entonces se construye otro modelo para su prueba y evaluación. Aplicada de esta forma, la elaboración de prototipos es esencialmente la construcción de sistemas mediante aprendizaje y descubrimiento.

En general la elaboración de prototipos no es aplicable para sistemas estándar de funciones específicas como contabilidad, pero es particularmente útil para el desarrollo de proyectos de sistemas únicos e innovadores que traen consigo un gran número de beneficios cualitativos o en el diseño de subsistemas pequeños y limitados para satisfacer las necesidades especiales de reportes y toma de decisiones de uno o unos cuantos usuarios. La elaboración de prototipos también puede utilizarse eficazmente para afinar y mejorar partes del sistema existente.

Típicamente, los analistas de sistemas desarrollan prototipos centrándose en el componente estructural de la salida, las entidades, atributos de datos y en las interfaces usuarios sistema, considerando solamente en forma secundaria otros aspectos como los controles y la tecnología.

Los analistas de sistemas, trabajando conjuntamente con los usuarios, crean diferentes reportes o salidas por pantalla empleando lenguajes de consulta o generadores de reporte. Los analistas de sistemas discuten con los usuarios las necesidades de éstos y luego desarrollan prototipos en la terminal conjuntamente con los usuarios.

Si las necesidades son complejas, los analistas pueden requerir una semana aproximadamente para producir una muestra. En cualquier caso, los prototipos deben demostrarse. Los usuarios ven, trabajan y los utilizan y con frecuencia solicitan cambios y los analistas de sistemas crean más prototipos.

Metodologías de Desarrollo.

Dentro de las metodologías de desarrollo, se pueden distinguir tres corrientes:

- La francesa, que dió como fruto la Metodología Merise, potenciada por la administración francesa a partir del año 1977.
- La inglesa, también impulsada por la administración en Gran Bretaña y que dió lugar, a partir de 1981, al SSADM (Structure System Analysis and Desing Methodd.).
- La americana, basada en las teorías de Edward Yourdon y que tiene algunas variantes aportadas por otros autores como Demarco, Gane y James Martin.

El resto de las metodologías existentes, tanto públicas como privadas, deben considerarse como adaptaciones, más o menos mejoradas, de las citadas anteriormente, así mismo, las metodologías de orientación a objetos están fuera del alcance de esta tesis.

Metodología Merise.

Surge en Francia a partir del año 1977, tras una petición del Ministerio de Industria, como un intento de definir una metodología a emplear en la Administración Pública para el desarrollo y diseño de sistemas informáticos.

Esta petición se realiza en un momento en que la gran diversidad de lenguajes y formalismos empleados para la representación de sistemas informáticos hacen necesario un esfuerzo de unificación de criterios y métodos. En un momento en que hay que tratar de eliminar los problemas derivados de la rotación de personal.

Los principios generales en que se apoya Merise son :

- Definición de documentos estándar.
- Uso del modelo Entidad/Relación y sus formalismos para la representación de datos.
- Uso de las Redes de Petri para la representación de procesos y tratamientos.
- Definición de grupos de trabajo, reparto de las responsabilidades y funciones a lo largo del desarrollo.
- Especificación del reparto de tareas y tratamientos entre los usuarios y el ordenador.
- Definición de los flujos de información entre las unidades del sistema.

El sistema se contempla desde diferentes niveles de abstracción y ésto da lugar a una descripción del mismo a tres niveles: conceptual, lógico u organizativo y físico.

En la fase de concepción se trabaja básicamente sobre dos elementos: datos y tratamientos. La descripción de los datos reflejará la información existente en el entorno y las relaciones entre ellas. La presentación de los tratamientos reflejará los procesos a realizar con los datos así como su secuencia en el tiempo. Con la

descripción de estos dos elementos habremos conseguido reflejar tanto el contenido del sistema como su funcionamiento.

Se podría establecer el siguiente cuadro en cuanto a los diferentes niveles, tanto de decisión como de descripción de datos y tratamientos como lo muestra en la tabla 1.9.3.

	TRATAMIENTOS	DATOS
Conceptual	Conceptual	Conceptual
Organizativo	Organizativo	Lógico
Técnico	Operativo	Físico

Tabla 1.9.3 Niveles de Decisión y de Descripción.

Veremos en primer lugar lo que significan estos niveles y cuál es su contenido. Empecemos por los tratamientos:

En cuanto a la **descripción de tratamientos**, en la descripción de términos de operaciones y resultados de la gestión que debe resolver el sistema independientemente de quién sea el que la realice; hombre o máquina y de qué modo.

En el **nivel organizativo**, se desglosan las operaciones descritas a nivel conceptual en procedimientos funcionales, es decir, en tareas realizadas sucesivamente en un puesto de trabajo. A este nivel se concreta: quién, cuándo y dónde se han de realizar éstas tareas.

En el **nivel operativo** se responde la pregunta de cómo hay que hacer las cosas. Se detallan, tanto para procedimientos automatizados como para los manuales y las normas para realizarlos correctamente.

Veamos a continuación como se estructura la descripción de los datos a diferentes niveles de abstracción:

En la **descripción de datos** contemplamos en primer lugar un **nivel conceptual**, en el que se observa la información del sistema en términos de objetivos o entidades, se describen sus propiedades, la información de cada uno de ellos y las relaciones entre los mismos. Este modelo conceptual de datos es, en principio, bastante estable a lo largo del ciclo de vida del sistema.

En el **nivel lógico** se traduce el modelo conceptual en agrupaciones o estructuras lógicas de datos para su tratamiento por el sistema. Todavía este nuevo modelo debe de ser independiente de la opción de elegir en cuanto a soporte de software del mismo.

En el nivel más bajo de la descripción de datos es decir, en el **nivel físico**, se concreta cuál va a ser la estructura final de los datos de acuerdo al sistema gestor elegido (base de datos, tipo de ficheros, etc.) Una vez conocido esto se podrán hacer optimizaciones del modelo lógico para mejorar rendimientos.

En estos procesos: descripción de datos y de tratamientos, se hacen necesarias validaciones para comprobar la coherencia entre unos y otros. En este sentido se podrían estructurar los pasos a seguir en el siguiente diagrama (ver figura 1.9.4) que propone un estudio paralelo de datos y tratamientos. Por otra parte, Merise como todas las metodologías, propone una serie de etapas en el desarrollo de un proyecto.

- Estudio preliminar
- Estudio detallado
- Realización
- Puesta en marcha o implantación

Esto es lo que compone el ciclo de vida del sistema, veremos ahora el conjunto de acciones a realizar en las etapas propuestas por Merise y su estructuración en distintas fases. El índice general de etapas y fases sería el siguiente, como lo muestra la figura 1.9.4.

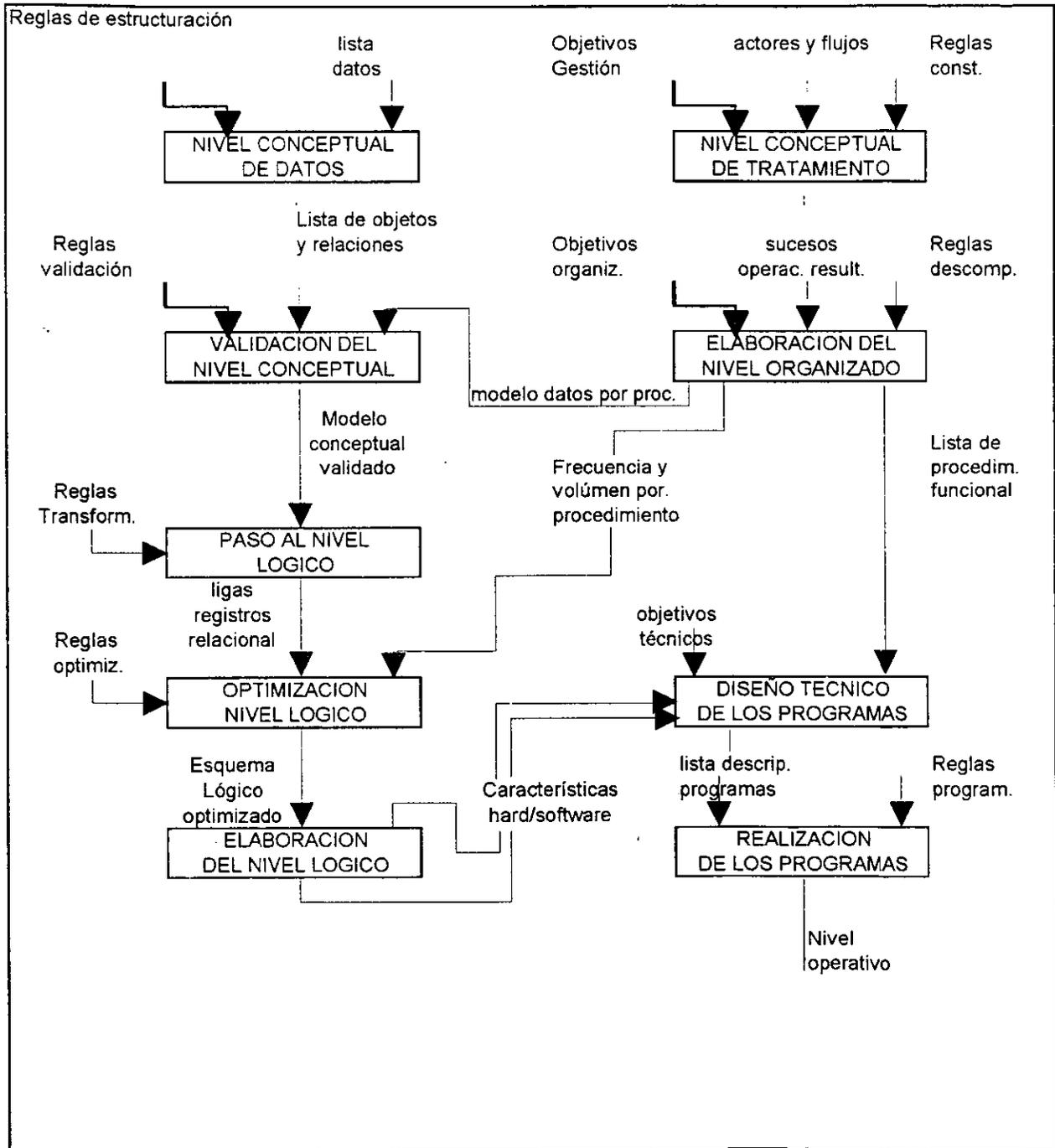


Figura 1.9.4 Niveles de Descripción de Datos y Tratamientos.

ETAPA 1 : Estudio Preliminar.

FASE 1 : Recopilación de datos.

- Recopilación inicial.
- Estudio de la situación actual.
- Síntesis y crítica de la situación actual.

FASE 2: Concepción de la nueva solución.

- Objetivos a alcanzar.
- Descripción de la solución.

FASE 3 : Evaluación y plan de desarrollo.

- Evaluación de la nueva solución
- Plan de desarrollo.

ETAPA 2 : Estudio Detallado.

FASE 1 : Concepción general.

FASE 2 : Concepción detallada de las fases.

- Realización de las especificaciones detalladas de los procesos.

FASE 3 : Plan de desarrollo.

ETAPA 3 : Realización.

FASE 1 : Estudio técnico.

FASE 2: Producción.

ETAPA 4 : Puesta en Marcha.

FASE 1: Preparación de los recursos físicos y humanos.

FASE 2 : Recepción y lanzamiento del sistema.

Otro punto contemplado por Merise es la descripción de los diferentes grupos de trabajo, es decir, el personal implicado en el desarrollo del sistema a todos los niveles.

Así , habrá:

- **Comite Director** que indicará los objetivos a cubrir por el sistema en las primeras fases de concepción.
- **Comite de Usuarios** que aportará el punto de vista de los futuros utilizadores del mismo.
- **Grupo de Desarrollo** o personal técnico encargado de satisfacer esos requerimientos a través del sistema informático.

Metodología SSADM.

Las siglas de esta metodología corresponden a las iniciales de Structure System Analysis and Desing Method. Nace en el Reino Unido a petición del Gobierno Inglés con la intención de ser un sistema de desarrollado público de Gran Bretaña.

En la elaboración de esta metodología intervienen dos empresas consultoras: Learmonth and Burchett Management Systems y Central Computer and Telecommunications Agency.

Es aceptada en 1981 en que se presenta la primera versión de la misma. Se hace uso obligatorio en toda la administración en 1983, aunque se siguen sacando sucesivas versiones hasta el modelo actual que va incorporando mejoras o contemplando fases del ciclo de vida no cubiertas en un principio.

La metodología consiste en una estructuración de los pasos a seguir en el desarrollo de un proyecto informático, en las fases iniciales del ciclo de vida del mismo y en la descripción de unas técnicas y formalismos sobre las que se basan los trabajos a realizar en cada fase.

Así, según un carácter puramente jerárquico, podíamos distinguir de la figura 1.9.5

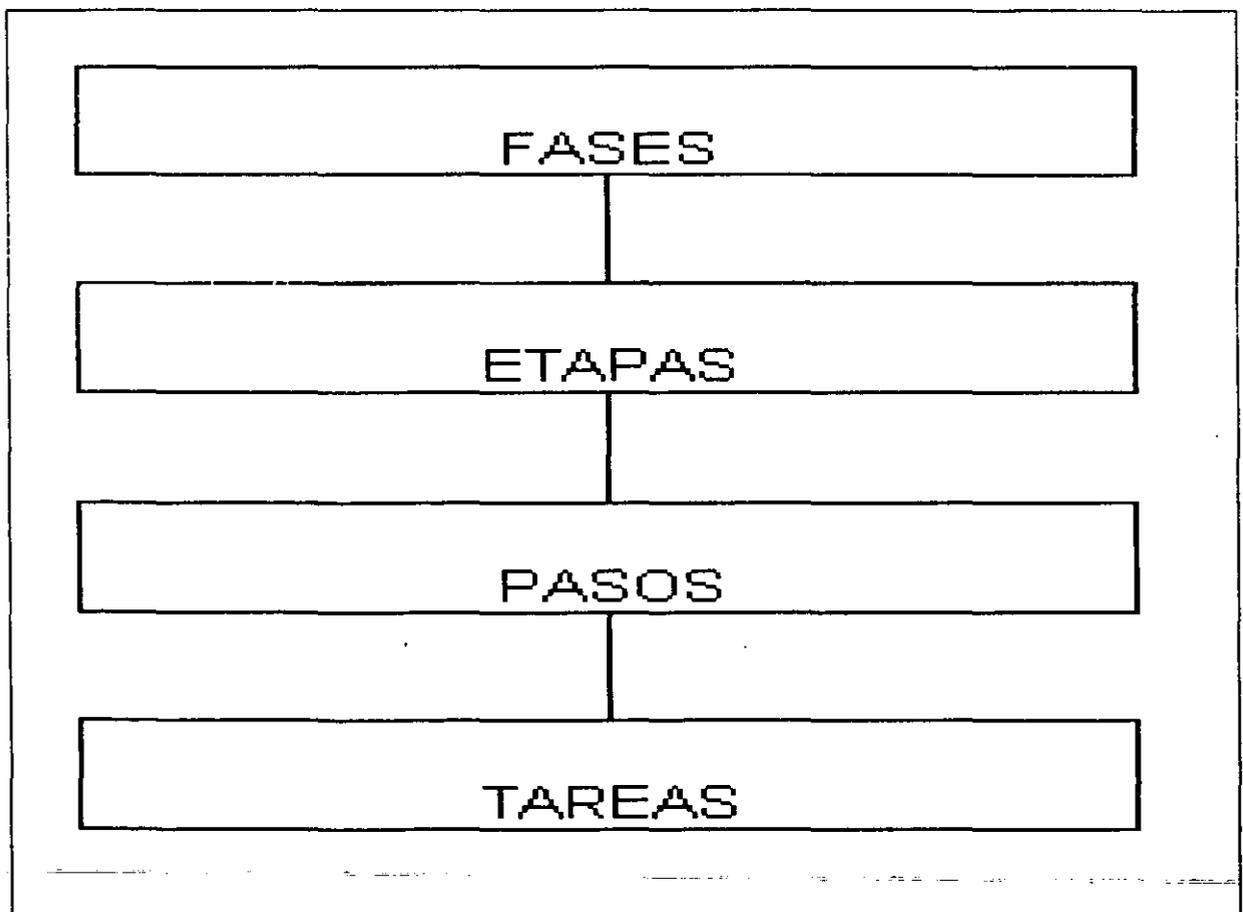


Figura 1.9.5 Estructuración de los Pasos en un Proyecto Informático.

El método contempla, en principio, las tres primeras fases del desarrollo: estudio de viabilidad, análisis y diseño, divididas a su vez en una serie de etapas, esto se muestra en la figura. 1.9.6

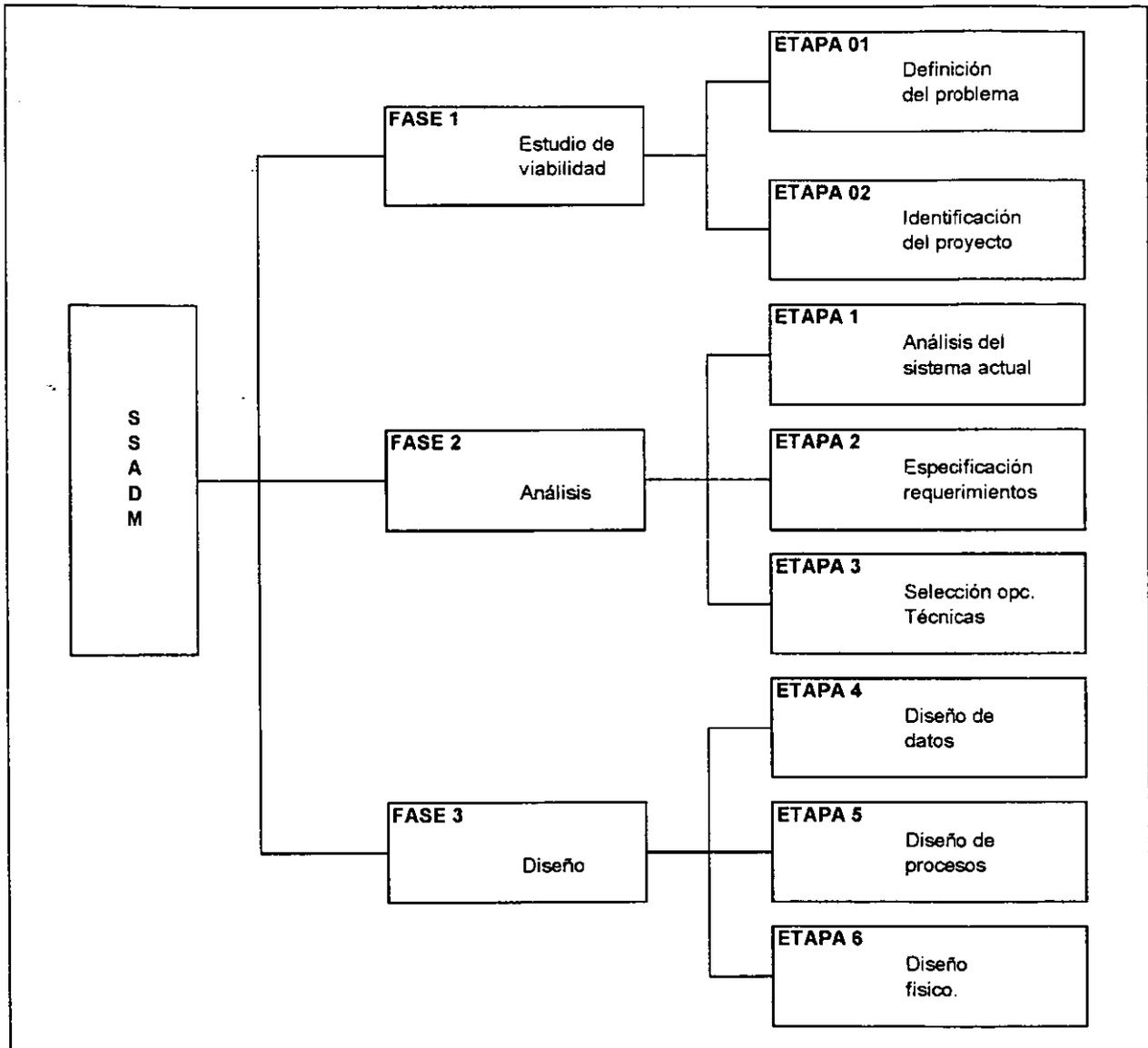


Figura 1.9.6 Fases y Etapas de SSADM.

Cronológicamente, las seis etapas, una vez comprobada la viabilidad del proyecto se desarrollarán en el orden que muestra la figura 1.9.7:

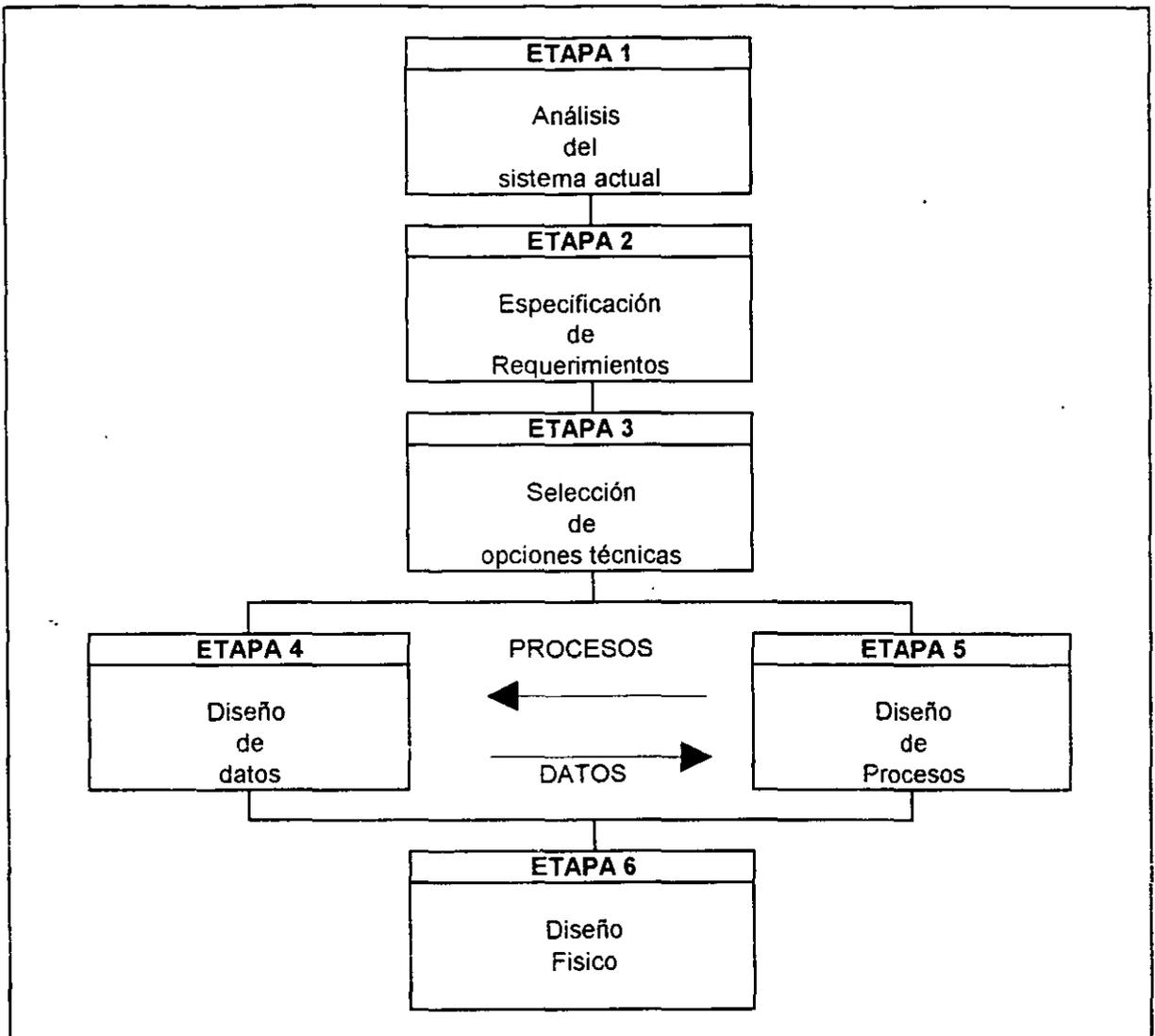


Figura 1.9.7 Orden de Ejecución de las Etapas.

Las técnicas utilizadas por SSADM en cada una de las fases y etapas son :

- Diagrama de Flujo de datos (DFD), que son una formal representación de los flujos de información en el interior del sistema contemplado y entre el sistema y el exterior, es decir sus reflexiones con otros.
- Estructura Lógica de Datos (LDS). Mediante la representación de las entidades del sistema y las relaciones entre ellas. Para ello se utilizarán los formalismos y teorías del modelo entidad-relación.

- Historia en la Vida de la Entidad (ELH). Representa la descripción de cómo las entidades descritas son afectadas por diferentes sucesos que ocurren en el sistema.
- Tercera Forma Normal (TFN), en la descripción de datos, es un método matemático para la definición de datos que ayuda a evitar inconsistencias y ambigüedades en la estructura del mismo.

Metodología Yourdon.

Es el representante de la corriente metodológica más importante de Estados Unidos, aunque hay numerosos autores que aportan variantes, matices y formalismos de representación al método de Yourdon.

A lo largo de sus obras Yourdon describe técnicas para la realización de análisis estructurado de sistemas basadas principalmente en los siguientes conceptos:

- Diagramas de flujo de datos para la representación de procesos.
- Diagramas de transición de estados para la representación estructurada de las funciones a realizar en los procesos.
- Modelo Entidad-Relación para la representación conceptual de datos.
- Diccionario de datos como base o soporte de información del sistema.
- Diagramas o mapas de estructura para la representación modular de los procesos y las variables intercambiadas entre ellos.
- Especificaciones de programas basadas en lenguaje estructurado y tablas de decisión.

Yourdon define las siguientes etapas y niveles en el ciclo de vida de los sistemas informáticos y por tanto aplica las técnicas descritas anteriormente a lo largo de estas etapas, distinguiendo las siguientes actividades a realizar:

- **Estudio de Viabilidad.** En este punto se debe identificar el proyecto a realizar, los usuarios responsables y se debe hacer un estudio de la situación actual, representando la misma a través de Diagramas de Flujo de Datos de primer nivel o Diagramas de Contexto en los que, de forma simple, se indiquen los procesos simples más relevantes. En este estudio de la situación actual se identificarán las deficiencias del mismo como puntos a resolver por el nuevo sistema. Esta actividad debe ocupar entre 5 y un 10% del total del proyecto.
- **Análisis del Sistema.** En esta fase se debe tener en cuenta los requerimientos de los usuarios relativos a cambios o funcionalidad del sistema y esta labor es facilitada por herramientas CASE de diagramación y técnicas de prototipo del sistema.
- **Diseño.** En esta fase pasa del nivel conceptual descrito anteriormente a un nivel de representación lógica de datos mediante un diseño dependiente del modelo de base de datos relacional elegida y una estructuración de los procesos utilizando diagramas de estructura de los mismos y generando las especificaciones de diagramas correspondientes.
- **Implementación o Producción** Comprende la generación de código y el ensamblaje e integración de todos los módulos.
- **Pruebas del Sistema.** A realizar con la totalidad del sistema hasta llegar a la aceptación del mismo por parte del usuario. En esta fase se harán pruebas de integración y de funcionamiento en conjunto de programas y cadenas.
- **Control de Calidad.** El objetivo de esta actividad es garantizar los controles de calidad del software que puedan estar definidos para la empresa. Esta fase complementa la anterior de forma que el producto final sea de un buen nivel de calidad y cumpla los estándares fijados.
- **Documentación** En este apartado se generará toda la documentación necesaria para la instalación del sistema: manuales de usuario, de operación, etc. La documentación interna, o sea, las especificaciones de programas, habrá sido creada ya anteriormente y utilizada por los programadores. Con ello el nuevo sistema queda completamente documentado, interna y externamente.

- **Conversión de los Datos del Sistema Anterior.** La ejecución de esta fase depende, evidentemente del estado anterior a la mecanización del entorno afectado por el proyecto. Si existía ya un sistema informatizado se deben realizar los programas de conversión de datos al nuevo sistema y si anteriormente los archivos eran manuscritos puede requerirse una grabación y carga previa a la puesta en marcha del sistema.
- **Instalación.** Comprende la puesta en marcha del sistema y en esta fase son de aplicación las consideraciones hechas acerca de aspectos tales como formación y entrenamiento del usuario, entrega de manuales, procesos paralelos, etc.

En la figura 1.9.8 se muestran las etapas y niveles del ciclo de vida utilizados por Yourdon para los sistemas informáticos.

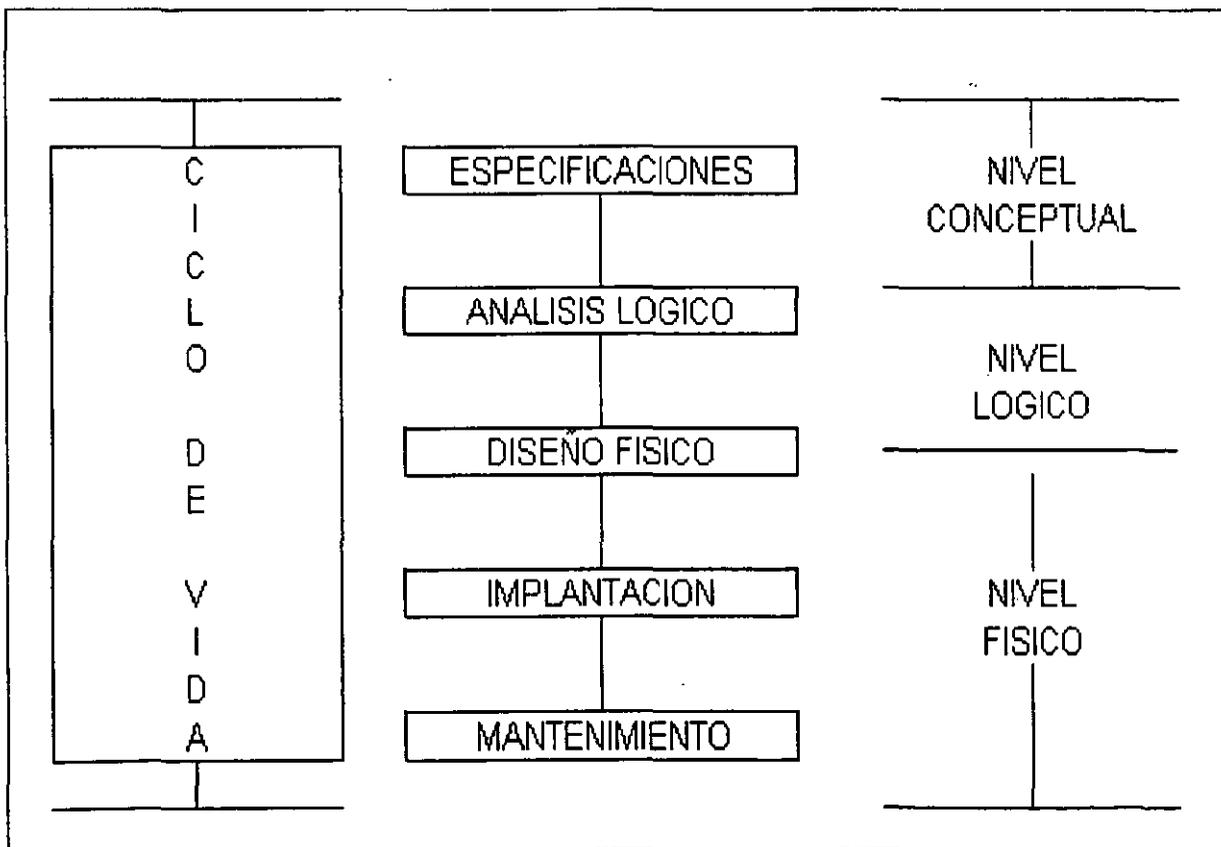


Figura 1.9.8 Ciclo de Vida Según Yourdon.

Estudio Comparativo entre las Diferentes Técnicas.

Evidentemente, en las tres metodologías expuestas hay puntos comunes y técnicas similares en determinadas fases del desarrollo. Para establecer una comparación entre ellas hay que tomar como base esos puntos comunes que toda metodología deba contemplar y después cómo son abordados estos puntos desde cada una de las metodologías. Tomaremos como referencia lo siguiente:

- Etapas de Ciclo de Vida.
- Modelado de Datos.
- Modelado de Procesos.

Etapas del Ciclo de Vida.

En general todas ellas estructuran el desarrollo en diversas etapas. Si examinamos esta estructuración se puede observar una mayor precisión en cuanto a los pasos a seguir en la metodología SSADM. Resulta ser casi un recetario, siguiéndolo tal cual podemos desarrollar cualquier proyecto.

Hay que resaltar que ninguna de las tres llega a determinar qué técnicas de programación estructurada se deben seguir, aunque todas recomiendan su uso. En el caso de SSADM incluso no se llega a esta etapa y la metodología llega hasta el diseño físico. Se podría decir que Merise es la que más trascendencia dá al estudio organizativo de la empresa como base de todo el sistema informático

La selección de las personas que van a intervenir en el proyecto a través de los comités de dirección, usuarios y equipo de desarrollo y en qué momentos o fases intervienen cada uno de ellos, hacen incidencia en el transfondo organizativo que Merise quiere aportar.

Modelado de Datos.

Es común en todas las metodologías la definición de los niveles conceptual, lógico y físico en la definición de los datos. A nivel conceptual, se admite por parte de todas el modelo entidad-relación como el más adecuado para reflejar la información del sistema. También es una tendencia general del mercado la utilización de bases de datos relacionales como soporte del nivel lógico del modelo de datos. Desde este punto de vista, todas recomiendan la normalización de estructuras siguiendo las reglas de normalización de cada uno para después desnormalizar de forma controlada y por razones de rendimiento y tiempos de respuesta del sistema.

Modelado de Procesos.

Nuevamente Merise introduce una descripción más precisa de los mismos a través de las Redes de Petri. En ellas la información acerca de los procesos del: qué hay que hacer, cómo, cuándo y quiénes es muy completa. Esta técnica puede tener el inconveniente de la dificultad no sólo para el que la realiza sino también para la validación por parte del usuario por su mayor complejidad y nivel de detalle.

Las otras dos metodologías realizan esta modelización a través de los diagramas de Flujo de Datos de diferente nivel de detalle. Los Diagramas de Flujo de Datos resultan claros para el usuario y relativamente fáciles de realizar. Evidentemente, si hay un apoyo en una herramienta CASE de diagramación, esta técnica resulta cómoda de realizar y de interpretar por el usuario. De cualquier forma, las Redes de Petri serán una opción más en las herramientas de soporte de la metodología SSADM en sus próximas versiones.

**CAPÍTULO 2.
PLANTEAMIENTO
DEL PROBLEMA Y
PROPUESTA DE
SOLUCIÓN.**

2.1 SITUACIÓN ACTUAL.

En la mayoría de los medios escritos de comunicación existen secciones denominadas anuncios clasificados, los cuales dan a conocer ofertas de bienes y servicios por parte de particulares dirigidos al público en general.

El objetivo de este punto es definir la situación actual de las personas involucradas en el proceso de publicación de un anuncio clasificado: el vendedor, el intermediario y el cliente final. Actualmente si una persona desea vender o comprar un producto, un inmueble o contratar algún servicio, es necesario darse a conocer y buscar el lugar donde puede realizarlo. La forma más accesible de publicarla es a través de algún medio publicitario impreso como es el periódico o revistas, ya que su precio lo hace accesible a la mayoría de los interesados.

A continuación se describirá la forma como se publican actualmente los anuncios clasificados, la cual se divide en tres etapas:

Primera Etapa. Es realizada por el usuario interesado en publicar el anuncio y consiste en:

1. Buscar un lugar autorizado. El usuario interesado en realizar la publicación acude a las Agencias de Publicidad que prestan este servicio.
2. Obtener una papeleta para la publicación. La papeleta es un formato que es proporcionado por la Agencia, donde se definen los datos necesarios para la publicación.
3. Llenar los formatos. El usuario llena los formatos con la información que desea dar a conocer, sin olvidar los datos requeridos y respetando los apartados a ser llenados por la Agencia.
4. Entregar los formatos. La entrega se realiza en la Agencia o en las oficinas de recepción del medio impreso.

5. Pagar la cantidad. La cual se fija de acuerdo al número de palabras, sección, tamaño y tiempo de publicación del anuncio.

La figura 2.1.1 muestra un diagrama con la primera etapa de publicación.

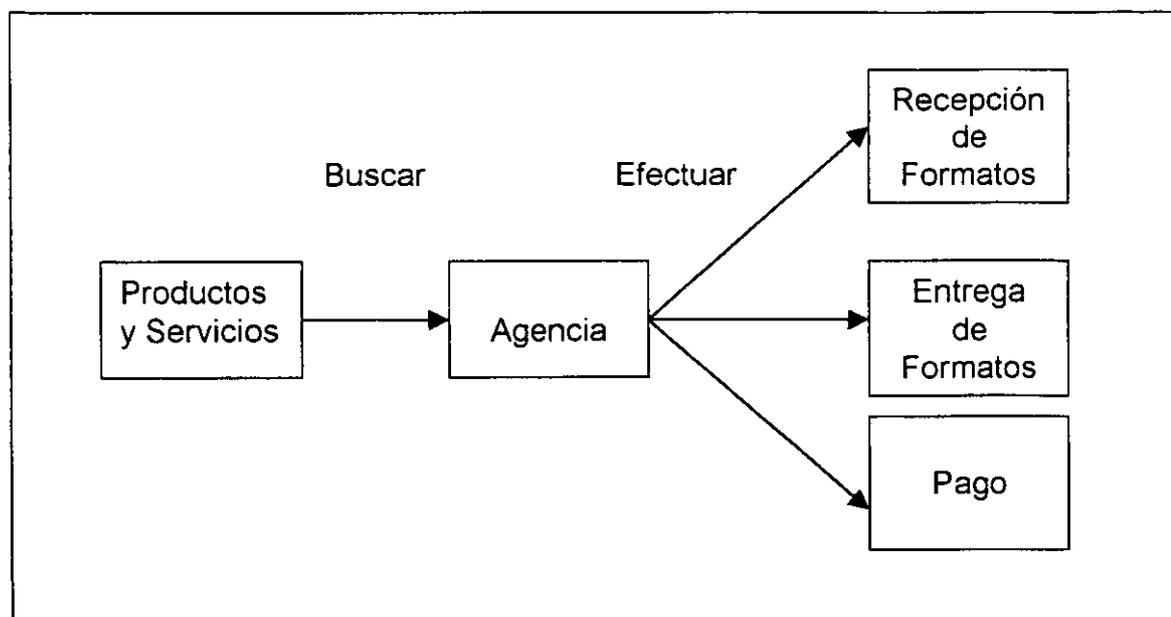


Figura 2.1.1 Primera Etapa de Publicación.

Segunda Etapa. Es realizada por la Agencia de Publicidad y consiste en:

1. Recolección de los anuncios. La Agencia se encarga de recolectar los anuncios captados durante el día en todos los centros de acopio.
2. Clasificación. Personal de la Agencia se encarga de separar los anuncios recolectados de acuerdo a las secciones (*primera plana o interiores*), clasificaciones y subclasificaciones establecidas.
3. Captura. Una vez clasificados los anuncios estos son capturados en un sistema de información.
4. Impresión. La información capturada se prepara con el formato final para su impresión.

5. Publicación. El producto final se distribuye a los diferentes puestos de periódicos y autoservicios.

La figura 2.1.2 muestra un diagrama con la segunda etapa de publicación.

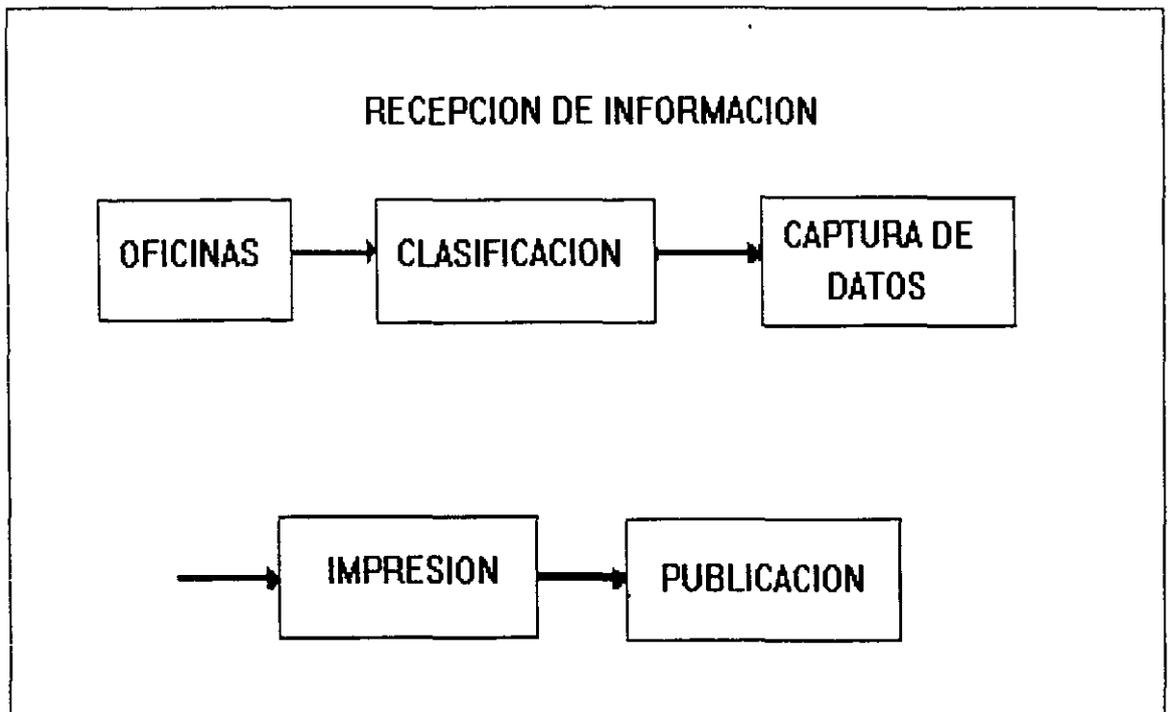


Figura 2.1.2 Segunda etapa de Publicación.

Tercera Etapa. Es realizada por los usuarios interesados en consultar los anuncios clasificados y consiste en:

1. Adquirir el periódico o revista, a través de una suscripción o bien directamente en un puesto de periódico.
2. Buscar en la sección de clasificados la sección correspondiente, al bien o servicio que desea adquirir el usuario.
3. Dentro de la sección, el interesado debe leer en forma secuencial los anuncios de la misma, hasta encontrar los que son de su interés marcándolos.
4. Elegir las opciones que le convengan.

5. Establecer la comunicación con el anunciante.

La figura 2.1.3 muestra el proceso completo de compra-venta de un producto o servicio

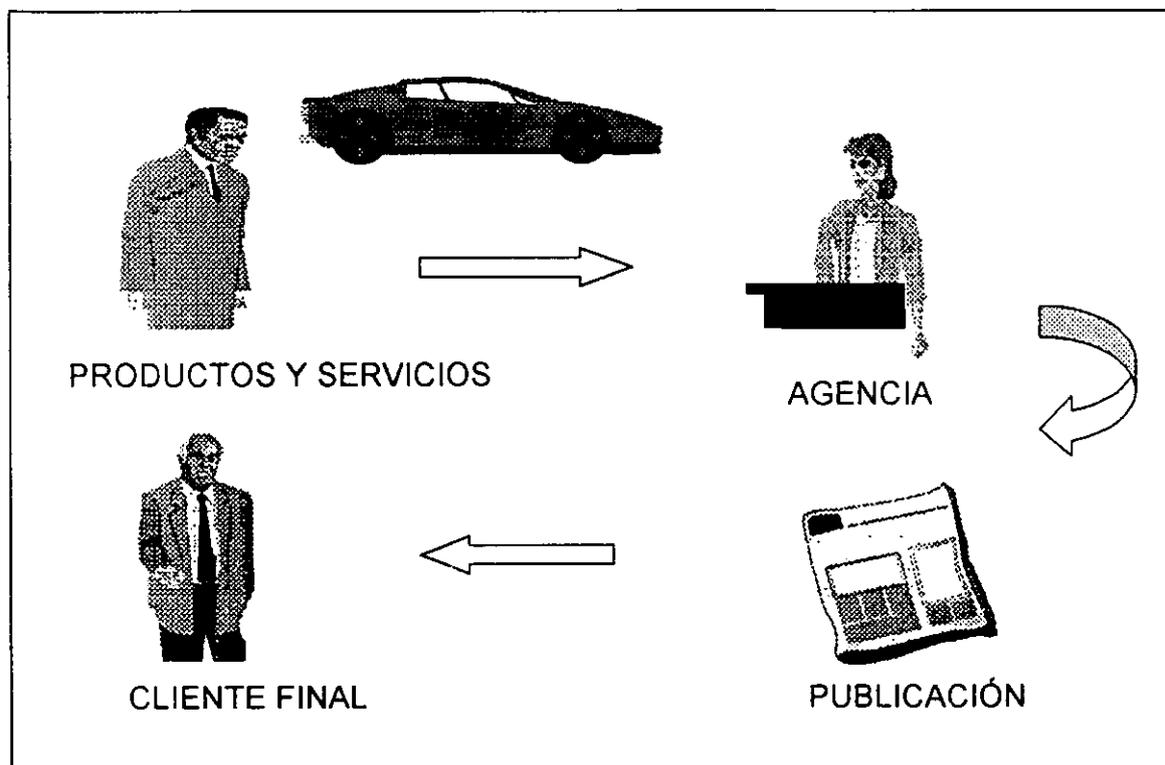


Figura 2.1.3 Proceso de Publicación de un Producto.

PLANTEAMIENTO DEL PROBLEMA Y PROPUESTA DE SOLUCIÓN El Usuario y sus Requerimientos:

opción de seleccionar dentro de un universo de información, los temas de interés de forma directa, lo que constituye un ahorro de tiempo.

- Flexibilidad de horario. El carácter matutino de los diarios, obliga a la gente interesada en adquirirlos a disponer de un tiempo por la mañana, ya que si no se cuenta con una suscripción y la demanda supera a la oferta, se corre el riesgo de no conseguir un ejemplar. El acceso a Internet puede realizarse a cualquier hora del día, lo que constituye una opción más para consultar anuncios clasificados. En la figura 2.2.2.3 se ilustra este requerimiento.

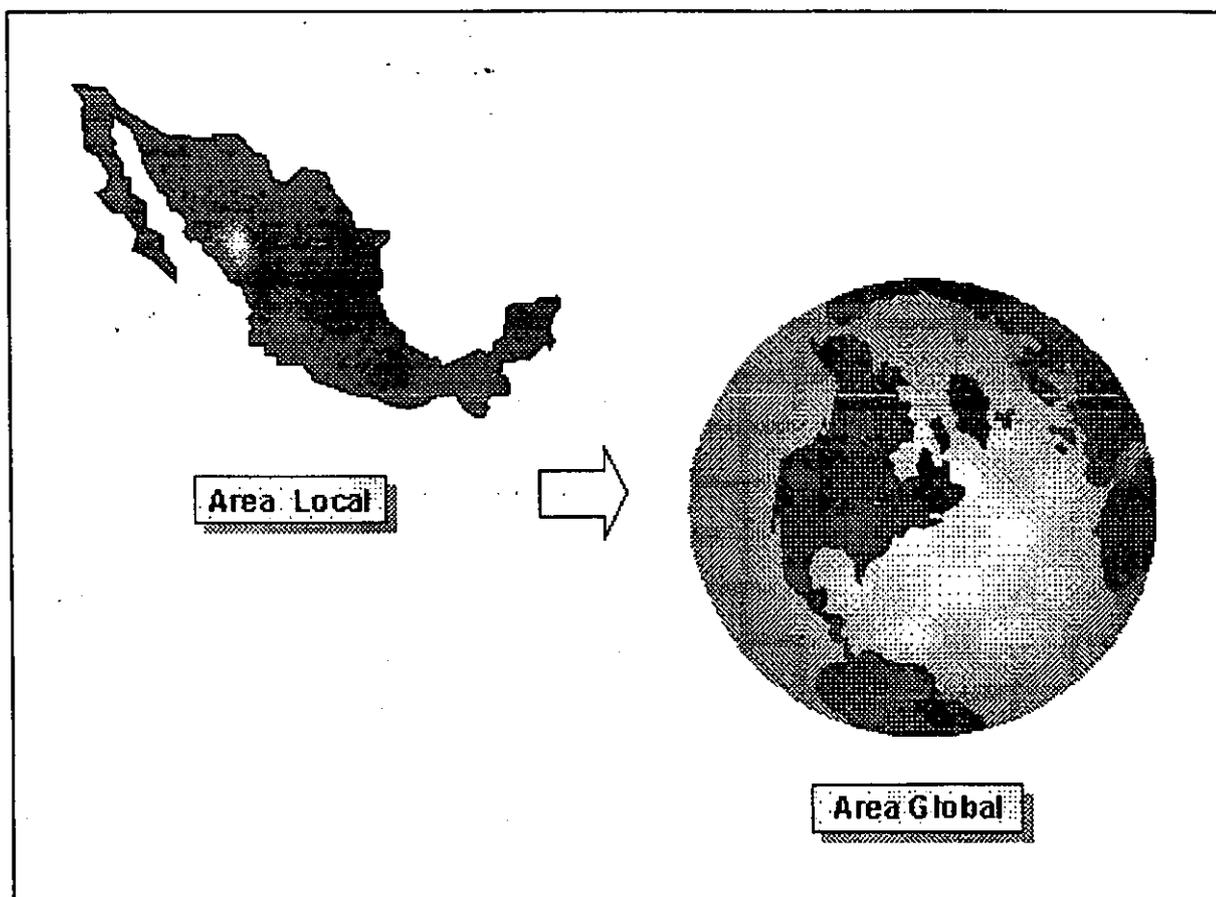


Figura 2.2.2.2 Cobertura Requerida por los Usuarios.

2. Los requerimientos particulares para los interesados en publicar productos o servicios con el objetivo de vender, se presentan a continuación:

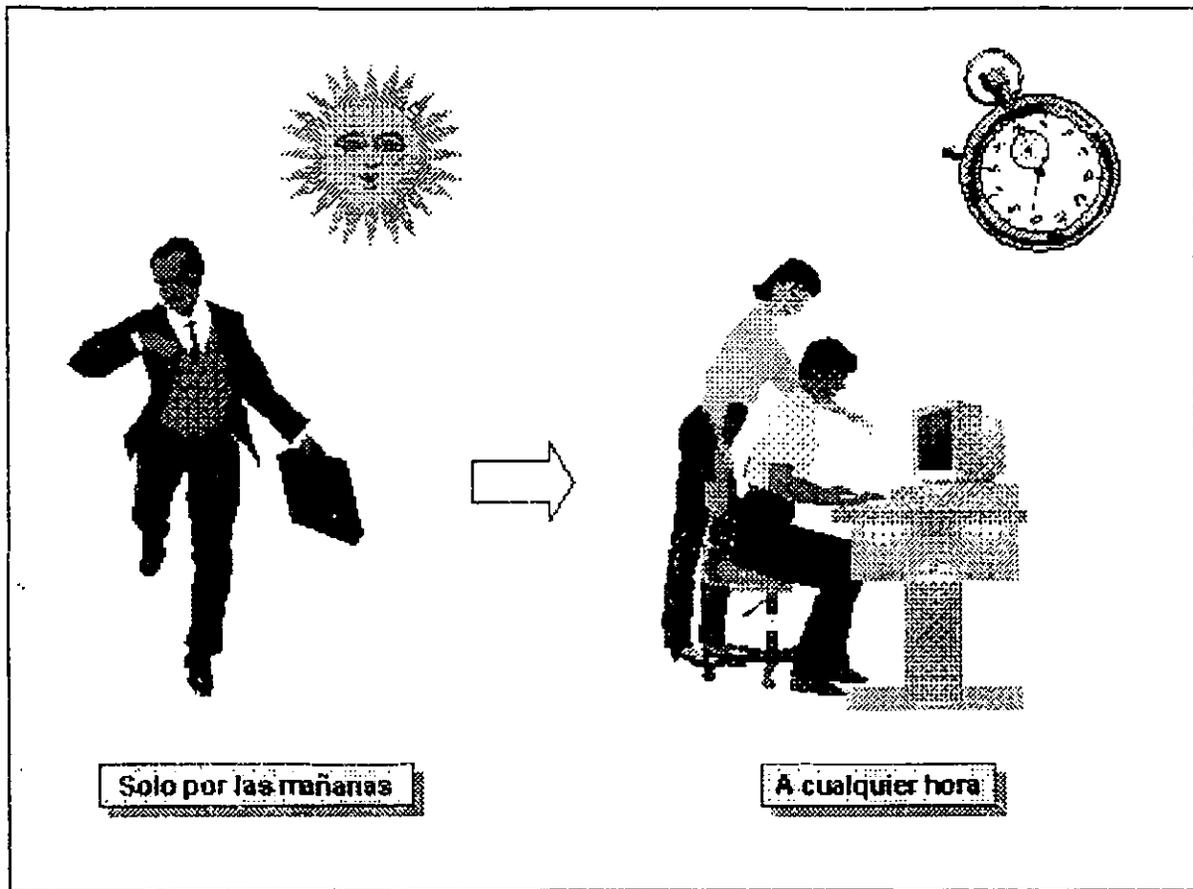


Figura 2.2.2.3 Flexibilidad de Horario.

- Que su producto llegue de la manera más rápida al público interesado. En el caso de las revistas especializadas en anuncios clasificados se tiene la desventaja de que su publicación en el mejor de los casos, es de carácter semanal. Los periódicos se consiguen diariamente, pero su edición limitada en ocasiones no cubre la demanda. El sistema de consulta vía Internet permitirá satisfacer este requerimiento.
- Que no se limite geográficamente la promoción de los que ofrecen. Al igual que los usuarios interesados en la consulta, la cobertura local que ofrecen los diarios, limita esta posibilidad.

2.3 RECOPIACIÓN Y ANÁLISIS DE INFORMACIÓN.

En secciones anteriores de este capítulo se mencionaron cuales son los procedimientos que se siguen para publicar un anuncio clasificado en el periódico; mencionamos que es necesario que las personas que desean colocar un anuncio deben acudir a las agencias de publicidad, llamar por teléfono o acudir directamente a las oficinas del periódico a realizar el trámite. Los trámites generalmente son similares en la mayoría de los periódicos de la ciudad, pero los formatos, las clasificaciones, los precios y los estilos de publicación varían dependiendo de cada periódico. Estos puntos marcan la diferencia en el servicio y la preferencia de los usuarios.

Una vez que los medios poseen el total de los anuncios a publicar, estos son concentrados en un archivo de texto que será procesado para ser desplegado en Internet.

A continuación presentamos una recopilación de la información más importante en la publicación de anuncios clasificados, que reúnen el conjunto de anuncios a publicar en Internet y un análisis de la misma.

2.3.1. Formatos.

Los formatos son impresiones de tamaño carta que los usuarios deben llenar con: la información que desean publicar, las características del anuncio en cuanto a tamaño, color y tipo de letra, los días en que ha de publicarse y alguna otra información de carácter administrativo de utilidad en caso de reclamaciones, aclaraciones o cancelaciones. La información que se requiere llenar depende del tipo de formato que se requiera según el anuncio a publicar.

Los formatos se clasifican en:

- Formato para Anuncios Clasificados.
- Formato para Anuncios Desplegados.
- Formatos Mixtos.

Anuncios Clasificados.

Los Anuncios Clasificados se caracterizan por su tamaño, generalmente pequeño, son aquellos cuyo precio se mide en proporción a la cantidad de palabras que incluyen, estas pueden ser minúsculas, mayúsculas ó mixtas. El número de palabras que se requieren para publicar un anuncio clasificado es de 10 como mínimo, de lo contrario no es posible publicarlo.

La información requerida para este tipo de formato es la siguiente:

1. Número de la Agencia de Publicidad donde se realizó el trámite.
2. Número del recibo o comprobante de pago.
3. Sección en la que ha de aparecer el anuncio.
4. Subdivisión donde ha de aparecer el anuncio.
5. Tamaño de la letra (minúscula, mayúscula o mixta).
6. Número de veces que ha de publicarse el anuncio.
7. Mes y días del mes de publicación.
8. Texto del anuncio.
9. Total de palabras.
10. Importe total del anuncio.

En la figura 2.3.1.1 se muestra el formato para un Anuncio Clasificado.

DESPLGADOS AVISOS	AGENTE No. <input style="width: 100%;" type="text"/>	MEDIDA <input style="width: 100%; height: 50px;" type="text"/>	O. INSERCIÓN No. <input style="width: 100%;" type="text"/>
	AGENCIA <input style="width: 100%;" type="text"/>		RECIBO No. <input style="width: 100%;" type="text"/>
EL DIARIO	SECCIÓN <input style="width: 100%;" type="text"/>	SUBDIVISIÓN <input style="width: 100%;" type="text"/>	No. AVISO <input style="width: 100%;" type="text"/>
	ELABORO <input style="width: 100%;" type="text"/>	CORRIGIÓ <input style="width: 100%;" type="text"/>	No. VECES <input style="width: 100%;" type="text"/>
		IMP. <input style="width: 100%;" type="text"/>	

MES MARQUE USTED LAS FECHAS EN QUE HA DE PUBLICARSE

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31
	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31

SUPPLICAMOS MARCAR LA SECCIÓN Y SUBDIVISIÓN DONDE USTED QUIERA QUE APAREZCA SU
LA CLARIDAD CON QUE ESCRIBA SU AVISO NOS PERMITIRÁ SERVIRLE MEJOR
PARA LAS SECCIONES 20 Y 38 ES NECESARIOS QUE EN EL TEXTO APAREZCA COMO PRIMERA
LA QUE ENCABEZA LA SUBDIVISIÓN RESPECTIVA.

TEXTO	
ORIGINAL	
NEGATIVO	
CORTE	
CORRECCIÓ	
REPOSICIÓN	
LAGO/MARC	

Figura 2.3.1.2 Formato para Anuncio Desplegado.

Formatos Mixtos.

En algunos periódicos se maneja un solo formato que sirve para publicar tanto anuncios clasificados como desplegados, tal es el caso del periódico el Excelsior y La Afición que manejan un solo formato. A continuación se presenta un formato mixto.

En la figura 2.3.1.3 se muestra el formato para los anuncios mixtos.

<p><i>CLASIFICADOS</i></p> <p style="text-align: center;">El Aviso de Ocasión</p>	<p style="text-align: center;">NOTA DE VENTA</p>	<p style="text-align: center;">AGENCIA / AGENTE</p> <hr/> <p style="text-align: center;">FACTURA</p>																																																																	
<p>PRIMERA PLANA ()</p>	<table border="1" style="width: 100%; border-collapse: collapse;"> <tr> <th style="width: 15%;">SECCIÓN</th> <th style="width: 20%;">PALABRAS</th> <th style="width: 20%;">TAMAÑO</th> <th style="width: 20%;">VECES</th> <th style="width: 25%;">IMPORTE</th> </tr> </table>				SECCIÓN	PALABRAS	TAMAÑO	VECES	IMPORTE																																																										
SECCIÓN	PALABRAS	TAMAÑO	VECES	IMPORTE																																																															
<p>MARQUE USTED LAS FECHAS DE PUBLICACIÓN</p>																																																																			
<p>Mes : <input style="width: 100px;" type="text"/></p>																																																																			
<table border="1" style="width: 100%; border-collapse: collapse; text-align: center;"> <tr> <td>1</td><td>2</td><td>3</td><td>4</td><td>5</td><td>6</td><td>7</td><td>8</td><td>9</td><td>10</td><td>11</td><td>12</td><td>13</td><td>14</td><td>15</td><td>16</td><td>17</td><td>18</td><td>19</td><td>20</td><td>21</td><td>22</td><td>23</td><td>24</td><td>25</td><td>26</td><td>27</td><td>28</td><td>29</td><td>30</td><td>31</td> </tr> <tr> <td> </td><td> </td> </tr> </table>					1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31																																
1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31																																					
<p>Mes : <input style="width: 100px;" type="text"/></p>																																																																			
<table border="1" style="width: 100%; border-collapse: collapse; text-align: center;"> <tr> <td>1</td><td>2</td><td>3</td><td>4</td><td>5</td><td>6</td><td>7</td><td>8</td><td>9</td><td>10</td><td>11</td><td>12</td><td>13</td><td>14</td><td>15</td><td>16</td><td>17</td><td>18</td><td>19</td><td>20</td><td>21</td><td>22</td><td>23</td><td>24</td><td>25</td><td>26</td><td>27</td><td>28</td><td>29</td><td>30</td><td>31</td> </tr> <tr> <td> </td><td> </td> </tr> </table>					1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31																																
1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31																																					
<p>LA CLARIDAD CON QUE USTED ESCRIBA SU AVISO, NOS PERMITIRÁ SERVIRLE</p> <p>POR FAVOR ESCOJA SU SECCIÓN AL REVERSO</p>																																																																			

Figura 2.3.1.3 Formato para Anuncios Mixtos.

Clasificación	Descripción	Sub- Clasificación	Descripción
18	ESCUELAS Y ACADEMIAS	1A 1B 1C	ENSEÑANZA EN ESCUELAS Y ACADEMIAS INSTITUCIONES MAESTROS
19	CONTRATACIÓN DE EJECUTIVOS (AS)	1A 1B	EJECUTIVOS (AS) SEMINARIOS Y CONVENCIONES
20	EMPLEOS (OFERTAS)	1A 1B 1C 1D 1E 1F 1G 1H 1I 1J 1K 1L 1M 1N 1O 1P 1Q 1Z	AGENTES DE VENTAS ALMACENISTAS AUXILIARES CONTABLES CAPTURISTAS COBRADORES CONTADORES COSTURERAS CHOFERES COCINERAS TÉCNICOS MECANÓGRAFAS MESERAS MÚSICOS PLANCHADORES Y LAVADORES SECRETARIAS TAQUEROS VENDEDORES OTROS EMPLEOS
21	OFICIOS		
22	SIRVIENTES		
23	PROFESIONISTAS QUE OFRECEN SUS SERVICIOS	1A 1B 1C 1D 1E 1F 1G	ABOGADOS CONTADORES DENTISTAS DETECTIVES INGENIEROS MÉDICOS QUÍMICOS

Clasificación	Descripción	Sub - Clasificación	Descripción
		1H 1I 1Z	OTROS PROFESIONISTAS SOLICITUD DE PROFESIONISTAS MATERIAL, EQUIPO Y MOB. PARA PROFESIONISTAS
24	MAQUINAS		
25	MATERIALES PARA CONSTRUCCIÓN	1A 1B 1C 1D 1E	ADAPTACIONES Y CONSTRUCCIONES HERRERÍAS MADERAS MATERIALES DE CONSTRUCCIÓN MATERIALES INDUSTRIALES
26	MAQUINARIA Y HERRAMIENTAS		
30	BANQUETES	1A 1B	SALONES PARA FIESTAS ALIMENTOS A DOMICILIO
33	INSTRUMENTOS MUSICALES (COMPRA, VENTA O COMP.)	1A 1B 1Z	ORGANOS PIANOS OTROS INSTRUMENTOS MUSICALES
34	MENSAJES		
35	SALONES DE BELLEZA	1A 1B 1C	ESTÉTICAS ESTILISTAS Y PEINADORAS MANICURISTAS
36	PERMUTAS		
37	TRASPASOS		
38	VENTA DE AUTOMÓVILES Y ACCESORIOS	1A 1B 1C 1D 1E 1F 1G 10A	CAMIONES Y AUTOBUSES CAMIONETAS GUAYINES MICROBUSES PICKUP Y PANEL TRACTOCAMIONES VANETTES CHEVROLET CAVALIER

Clasificación	Descripción	Sub - Clasificación	Descripción
		10B	CELEBRITY
		10C	CENTURY
		10D	CUTLASS
		10E	CHEVY
		10F	SUBURBAN
		10G	CADILAC
		10H	MALIBU
		10I	PONTIAC
		10J	BLEIZER
		10Z	OTROS MODELOS CHEVROLET
			CHRYSLER
		15A	DART
		15B	DODGE
		15C	NEÓN
		15D	NEW YORKER
		15E	PHANTOM
		15F	RAM CHARGER
		15G	SHADOWN
		15H	SPIRIT
		15I	STRATUS
		15J	INTREPID
		15K	VOYAGER
		15L	CONCORD
		15Z	OTROS MODELOS CHRYSLER
			FORD
		20A	COUGAR
		20B	FAIRMONT
		20C	ESCORT
		20D	GRAND MARQUIS
		20E	LINCOLN
		20F	MUSTANG
		20G	TAURUS
		20H	THUNDERBIRD
		20I	TOPAZ
		20J	COUNTUR

Clasificación	Descripción	Sub - Clasificación	Descripción
		20K	MYSTIQUE
		20Z	OTROS MODELOS FORD
		25	MERCEDES BENZ
		26	BMW
		27	HONDA
		28	AUDI
			NISSAN
		30A	DATSUN
		30B	LUCINO
		30C	ICHI BAN
		30D	MÁXIMA
		30E	TSURU
		30F	SENTRA
		30Z	OTROS MODELOS NISSAN
		35	RENAULT
			TAXIS
		40A	TAXIS
		40B	PLACAS Y TAXÍMETROS
			VAM
		45A	GREMLIN
		45B	RAMBLER
		45Z	OTROS MODELOS VAM
			VOLKSWAGEN
		50A	ATLANTIC
		50B	BRASILIA
		50C	CARIBE
		50D	COMBI
		50E	CORSAR
		50F	GOLF
		50G	JETTA
		50H	SEDAN
		50Z	OTROS MODELOS VOLKSWAGEN
		60	VENTA DE AUTOMÓVILES OTRAS MARCAS

2.5 DESCOMPOSICIÓN FUNCIONAL.

- Con el fin de comprender ampliamente la problemática de la Agencia de Publicidad, es necesario partir de la siguiente descomposición de actividades y así, poder proponer una solución óptima.

Las tareas principales que se tienen que separar y analizar, una vez que se tienen identificadas, son las siguientes:

1. Diseño e implementación de los módulos correspondientes al BACK-END (Parte interna del Sistema)
 - Generación de tablas que permitan una mejor organización de los datos de acuerdo a las clasificaciones que tenga la Agencia de Publicidad.
 - Automatización del proceso de carga de información en la Base de Datos del sistema, partiendo de la información que ellos actualmente generan para el actual proceso de impresión de desplegados.
2. Diseño e implementación de los módulos correspondientes al FRONT-END (presentación para el usuario).
 - Construcción de páginas dinámicas de Web, las cuales lleven a un diseño proporcionado por los estándares internacionales de acuerdo a sus necesidades, las cuales permitan hacer un sistema interactivo y fácil de usar por los usuarios, garantizando la rapidez de consultas como desplegado de los resultados que el usuario solicite.
3. Utilización de una arquitectura de cómputo que proporcione soporte a muchos usuarios concurrentes a través de Internet, que sea: fácil de administrar para los

encargados, escalable y totalmente abierta a los estándares actuales de cualquier modelo de cómputo.

2.5.1 Módulos Correspondientes al Back End.

Para lograr una mejor clasificación de la información, los desplegados de los usuarios deberán almacenarse bajo la siguiente organización de tablas:

- **Temporal:** Esta tabla es donde se almacenará toda la información de los desplegados. De esta tabla se parte para cargar datos a las demás tablas. También sobre esta tabla se harán consultas genéricas sobre toda la información que se tiene en los desplegados.
- **Inmuebles, negocios, permutas, empleos, cómputo, varios, vehículos y servicio:** En estas tablas es donde se tendrá toda la información clasificada de acuerdo a las secciones que presenta la Agencia de Publicidad.
- **Cata_tipo, cata_subtipo, cata_submarca:** Son las tablas donde se almacenarán los catálogos permanentes de las clasificaciones que se tienen en la agencia.
- **Tipo_hoy, subtipo_hoy, submarca_hoy:** Son las tablas de catálogos diarios donde se almacenarán las secciones que el día de hoy cuentan con desplegados.

Respecto a la carga automática de información a la base, se partirá originalmente del archivo de texto que contiene los clasificados de los clientes. Inicialmente se llevará a cabo un proceso, el cual deberá formatear los datos, también se corregirán caracteres especiales a caracteres en español, se cargarán los datos a una tabla temporal para después hacer la distribución de los datos a las tablas correspondientes, y finalmente se generarán los catálogos diarios de los clasificados dependiendo de los que en ese día existan. Este proceso se realizará diariamente.

2.5.2 Módulos Correspondientes al Front End.

En esta etapa de la aplicación se llevará a cabo la creación de procedimientos donde estarán definidos los tipos de consultas y donde se agregará la parte del diseño de las páginas Web.

Se planea un despliegue de aproximadamente 24 páginas dinámicas, las cuales cubrirán las necesidades de la aplicación. Según se analizó en la información proporcionada por la agencia, siete de las ocho clasificaciones (inmuebles, negocios, permutas, empleos, cómputo, varios y servicios) tendrán la misma estructura lógica y la única clasificación diferente a las demás será la de "autos", debido a que en esta clasificación se incluye una subclasificación extra.

En la figura 2.5.2.1, el **package** (estructura lógica que contiene procedimientos) "varios" es representativo de las siete clasificaciones comunes en estructura lógica, los cuales estarán formados por 5 procedimientos o funciones (representados por cuadros), los cuales recibirán cada uno sus parámetros correspondientes (representados por óvalos).

El procedimiento inicial del paquete "varios" es **forma_varios**, el cual pedirá las opciones del usuario y las almacenará en dos variables: "p_opción" (selección de subclasificación) y "p_variable" (palabras por las cuales el usuario realizará búsquedas en la tabla varios).

El procedimiento **vali_varios** se encargará de verificar cuál de las dos opciones insertó el usuario, para ejecutar alguno de los dos procedimientos: **busca_varios** ó **subforma_varios**.

El procedimiento **busca_varios**, ejecutará Context para poder realizar "búsquedas en memoria" con las palabras y clasificación que haya introducido el usuario.

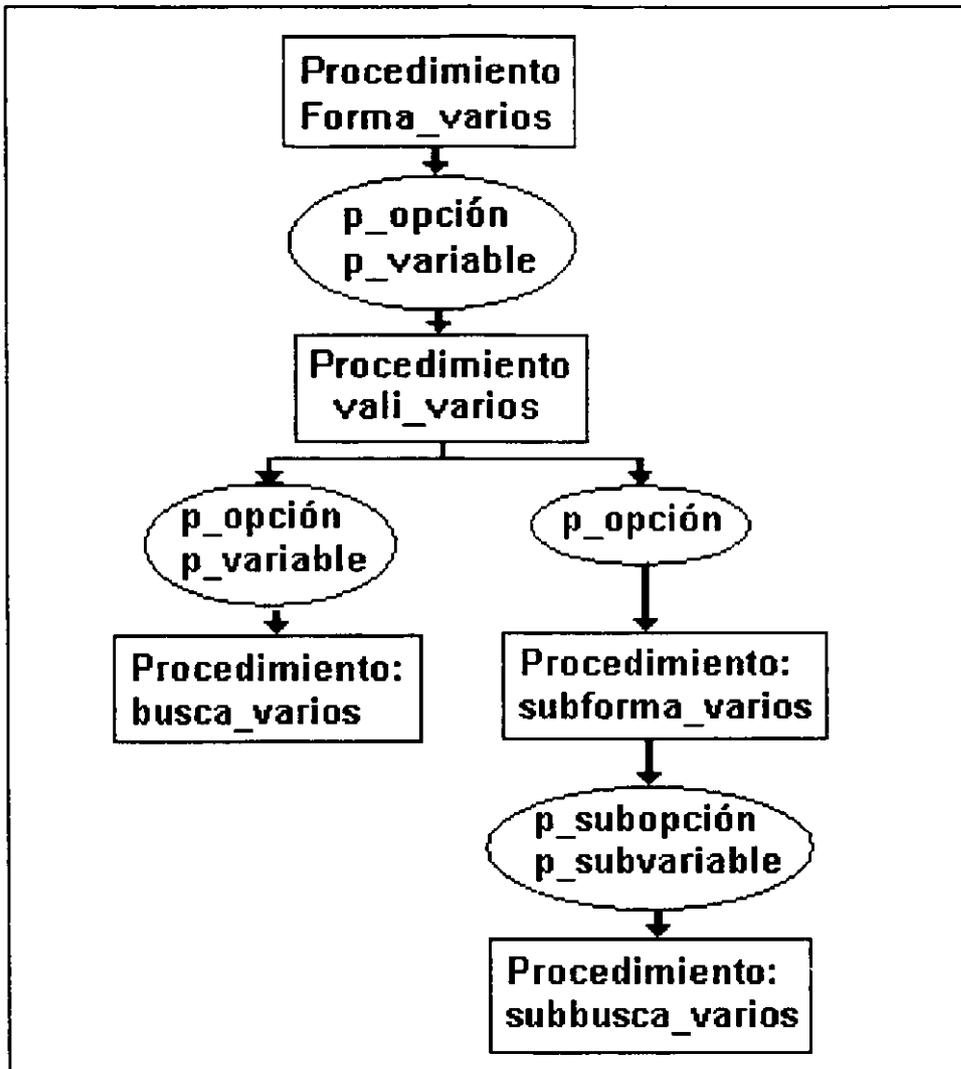


Figura 2.5.2.1 Paquete "varios".

El procedimiento **subforma_varios**, ejecutará una segunda pantalla de captura de opciones del usuario, la cual mostrará las subclasificaciones de la tabla "varios".

El procedimiento **subbusca_varios**, realizará búsquedas en memoria con Context, si el usuario introdujo una serie de palabras por las cuales necesitaba buscar.

Se emplearán las dos variables que se utilizaron en el procedimiento anterior: (p_subvariable y p_subopción).

A continuación se dará la explicación del paquete autos, el resto de los paquetes llevan la misma lógica de programación del paquete varios, antes mencionado, por lo cual no será necesario la explicación de cada uno de ellos.

En la figura 2.5.2.2 el paquete "autos" muestra la secuencia lógica de cómo será la secuencia de pantallas para que el usuario realice consultas de la clasificación de autos. A diferencia del paquete "varios" se agregará una segunda forma de captura para que el usuario elija las submarcas de los autos. Por ejemplo: la marca Ford cuenta con modelos como Mustang, Gran Marquis, Topaz, etc.; entonces el usuario podrá elegir estas marcas en la forma de captura "subforma_autos_1" y el procedimiento que desplegará los resultados será "busca_submarca".

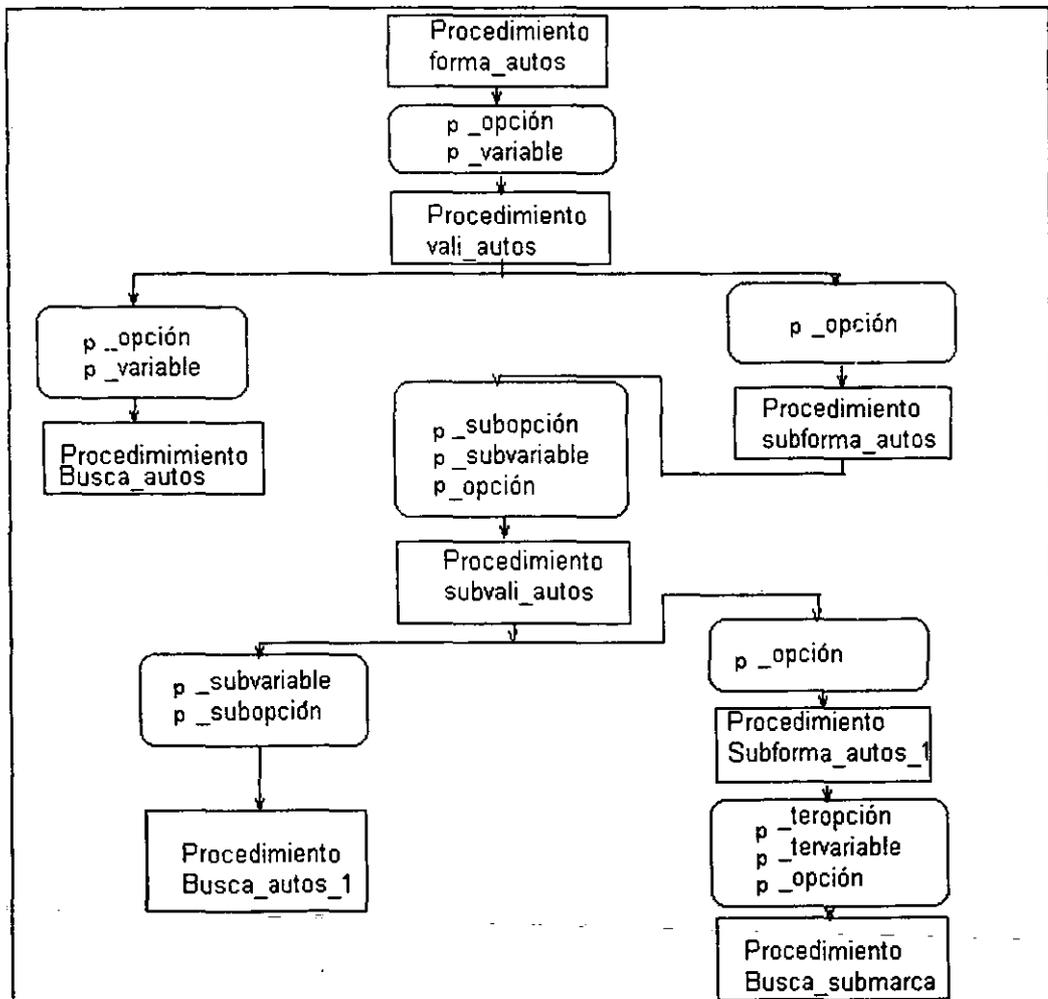


Figura 2.5.2.2 Paquete "autos".

2.5.3 Arquitectura de Cómputo.

Debido a que el sistema tendrá sus cimientos en Internet, entonces se empleará una arquitectura que sea capaz de soportar a muchos usuarios concurrentes, y que tenga contemplado ejecutar aplicaciones con estándares abiertos, de fácil actualización y respaldo, tanto en el servidor de base de datos como en el servidor de Web o aplicaciones. Un primer acercamiento a la arquitectura con estas características puede verse en la figura 2.5.3.

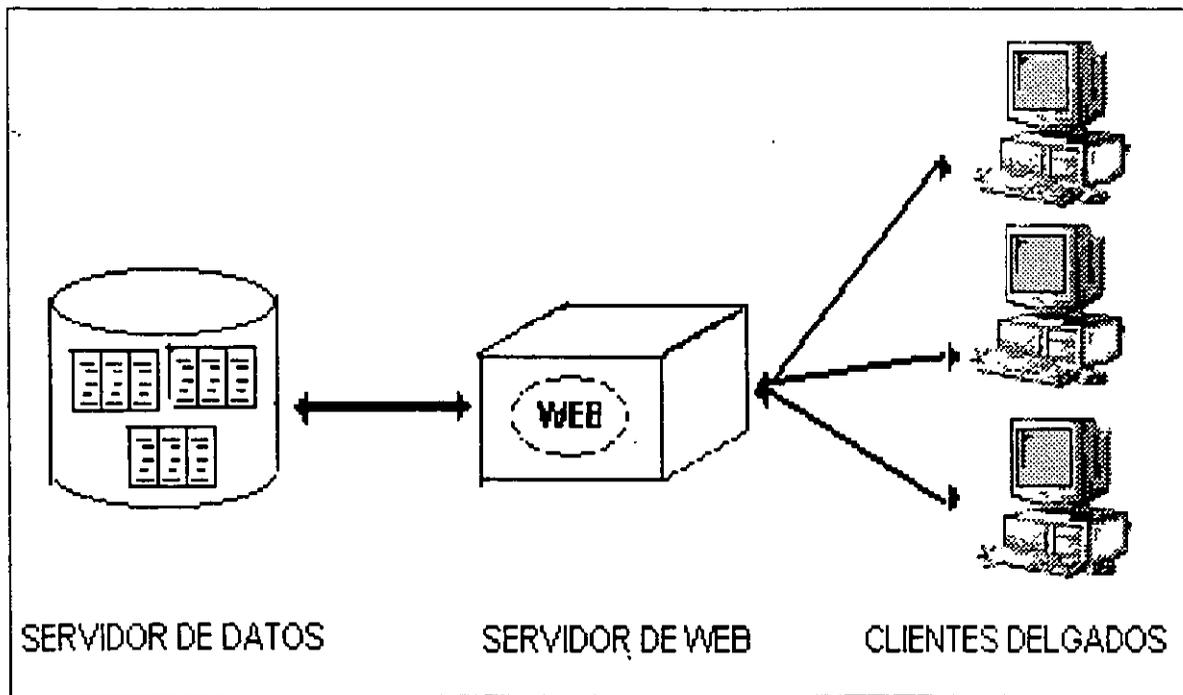


Figura 2.5.3. Probable Arquitectura de Cómputo..

2.6 OPCIONES DE SOLUCIÓN.

Además de la utilización de herramientas Oracle para el desarrollo del Sistema de Consulta, es necesario presentar opciones de solución en cuanto a sistema operativo, arquitectura y lenguaje de programación para la interfaz del usuario. En la tabla 2.6.1 se muestran estas opciones.

Sistema Operativo	UNIX	Windows NT		
Arquitectura	Dos Capas	Tres Capas		
Lenguaje para la programación de la Interfaz de usuario.	JAVA	Perl	CGI's	HTML

Tabla 2.6.1 Opciones de Solución.

A continuación se describirán las características más importantes de las opciones de solución mencionadas:

2.6.1 Sistemas Operativos.

Windows NT.

Windows NT de Microsoft es un verdadero sistema operativo de 32 bits muy poderoso, que está disponible en versiones cliente y servidor. Entre las características clave de NT está la multitarea prioritaria, procesos de multilectura o hebras, portabilidad y soporte para multiprocesamiento simétrico. Windows NT emplea el sistema de archivos NT (NTFS). Este sistema de archivos soporta nombres de archivos de hasta 256 caracteres. También permite el rastreo de transacciones. Esto significa que si el

sistema falla, NT regresa los datos al estado inmediato anterior a la caída del sistema. Microsoft diseñó Windows NT para que fuera portátil. Está compuesto por un Kernel o Núcleo, así como de diferentes subsistemas del sistema. Hay subsistemas disponibles para aplicaciones que ejecutan programas basados en OS/2 y POSIX. Un procesador DOS virtual (VDM) ejecuta MS-DOS y aplicaciones Windows de 16 bits. NT incluye software de red de punto a punto para que los usuarios de NT puedan compartir archivos y aplicaciones con otros usuarios que ejecuten NT o Windows para Trabajo en Grupo. Windows NT requiere que los usuarios introduzcan una contraseña cada vez que inician el sistema operativo, estén o no conectados a un servidor. Cada vez que inicia NT, éste solicita una contraseña. Windows NT califica para la certificación gubernamental C-2 para ambientes seguros. Una función de seguridad de NT es el administrador de usuarios. Este programa garantiza que las contraseñas se sujeten a la política de la compañía.

Windows NT Server ofrece compartición de archivos integrada, capacidad de comparación de impresoras para la computación en grupos de trabajo y una interfaz de sistema de red abierto, que incluye soporte integrado para IPX/SPX, TCP/IP, NetBEUI y otros transportes. NT Server es compatible con redes existentes como VINES, NetWare, UNIX, LAN Manager 2.x y Windows para Trabajo en Grupo. Windows NT incluye interfaces de programación de aplicación (API) que permiten que los fabricantes de sistemas operativos de red (NOS) escriban software de cliente para que sus productos puedan ejecutarse con éste.

La integración de la interfaz de usuario de Windows 95 en NT 4.0, proporciona una visión consistente a través del escritorio y el servidor, resultando en un menor tiempo de entrenamiento y un más rápido desenvolvimiento del nuevo sistema operativo de red. Herramientas como el administrador de tareas y el monitor de red simplifican la administración del servidor. El directorio de servicios de Windows NT (NTDS) soporta a 25,000 usuarios por dominio y cientos o miles por empresa. Sin importar lo centralizado o descentralizado de un negocio, NTDS permite instalar un directorio en la organización capaz de proveer un manejo completo de recursos, servicios y aplicaciones. NTDS es

un directorio de servicios que presenta seguridad, arquitectura confiable, interfaz gráfica para la administración e interoperatividad abierta con Novell NetWare.

NT 4.0 incluye un programa de diagnóstico que proporciona información acerca de los drivers y del uso de la red, minimizando los posibles errores del sistema. Esta información se presenta en forma gráfica que puede ser utilizada desde un sistema NT remoto. El desempeño y la escalabilidad del servidor se han mejorado, así como la compartición e impresión de archivos y el desempeño del servidor de Internet. Windows NT 4.0 trabaja con sistemas como NetWare, UNIX e IBM. Tiene soporte para más de 5,000 plataforma de hardware, siendo compatible para los protocolos de red más utilizados como TCP/IP, IPX/SPX, NetBEUI, AppleTalk, control de enlace de datos (Data Link Control, DLC), HTTP, arquitectura de redes de sistema (Systems Network Architecture, SNA), PPP y protocolo de punto a punto por medio de túnel (Point to Point Tunneling Protocol, PPTP). NT 4.0 es compatible para una gran variedad de sistemas clientes como Windows 3.x, Windows 95, Windows NT Workstation, IBM OS/2 y Macintosh.

Resumiendo las características de Windows NT podemos decir:

- Windows NT es un sistema operativo de 32 bits, que está disponible en versiones cliente y servidor.
- NT ofrece procesamiento multitarea, procesamiento de lectura múltiple e interrupciones prioritarias.
- Ofrece la capacidad de realizar procesamiento simétrico.
- NT califica para la certificación gubernamental C-2 para ambientes seguros.
- Incluye soporte integrado para IPX/SPX, TCP/IP, NetBEUI y otros transportes.
- El directorio de servicios de NT 4.0 (NTDS) soporta a 25,000 usuarios por dominio y cientos o miles por empresa.
- NT 4.0 incluye un programa de diagnóstico que proporciona información acerca de los drivers y del uso de red.

UNIX.

El sistema operativo UNIX ha evolucionado durante los últimos veinte años desde su invención como experimento informático hasta llegar a convertirse en uno de los sistemas operativos más populares e influyentes del mundo. UNIX es el más usado en investigación científica, pero su aplicación en otros entornos es bastante considerable. UNIX tiene una larga historia y muchas de sus ideas y metodologías se encuentran en sistemas como DOS y Windows.

Las características fundamentales del UNIX moderno son: memoria virtual, multitarea y multiusuario. La filosofía original de diseño de UNIX fue la de distribuir la funcionalidad en pequeñas partes: los programas trabajo. Las versiones modernas del sistema UNIX están organizadas para un uso de red fácil y funcional, por lo que es muy frecuente encontrar versiones del sistema UNIX sobre grandes unidades centrales sosteniendo varios cientos de usuarios al mismo tiempo.

Las herramientas de comunicación internas del sistema, la fácil aceptación de rutinas de dispositivo adicionales de bajo nivel y la organización flexible del sistema de archivos son naturales para el entorno de red de hoy en día. El sistema UNIX, con su capacidad de multitarea y su enorme base de software de comunicaciones, hace que la computación por red sea simple, permitiendo también compartir eficientemente dispositivos como impresoras y disco duro.

La versión SVR4 (Sistema V versión 4), es la versión más actualizada del sistema UNIX de AT&T. Ha sido portada a la mayoría de las máquinas computadoras centrales y es el estándar actual para la línea AT&T. SVR4 ha sido significativamente mejorado con respecto a versiones anteriores. Una de estas mejoras es la interfaz gráfica de usuario (GUI), que permite la utilización de X Windows. Los sistemas comerciales UnixWare de SCO y Solaris de Sun Microsystems están basados en el SVR4. La mejora más importante de SVR4 es la adición de soporte completo para redes de área

local. La administración de máquinas conectadas en red se ha mejorado en gran medida y la administración remota es ahora posible a través de la red.

Para poder identificar a las personas, Unix realiza un proceso denominado ingreso (login). Cada archivo en UNIX tiene asociados un grupo de permisos. Estos permisos le indican al sistema operativo quien puede leer, escribir o ejecutar como programa determinado archivo. UNIX reconoce tres tipos diferentes de individuos: primero, el propietario del archivo; segundo, el "grupo"; por último, esta el "resto" que no son ni propietarios ni pertenecen al grupo, denominados "otros". En general las máquinas UNIX están conectadas en red, es decir, que los comandos no se ejecutarán físicamente en la computadora en la cual se está tecleando, sino en la computadora en la que uno se ha conectado

Dado que UNIX es un sistema operativo de red, muchas de las computadoras con UNIX están conectadas unas a otras y a Internet. Una computadora UNIX ofrece generalmente una serie de servicios de red, mediante programas que se ejecutan continuamente llamados daemon. Estos daemon escuchan un puerto, o dirección numérica que identifica un servicio y actúan como servidores. Para usar tales servicios se usan programas clientes, ya que saben de qué puerto se trata y cual es el protocolo adecuado para hablar con ese daemon. Por supuesto, para usar estos programas hay que tener primero permiso para usar tal puerto o protocolo, y luego acceso a la máquina remota, es decir, hay que "autenticarse", o identificarse como un usuario autorizado de la máquina. Algunos de estos programas son: telnet, rlogin, rsh, ftp, etc.

Resumiendo las principales características de UNIX podemos decir:

- UNIX es el sistema más usado en investigación científica, pero su aplicación en otros entornos ha tenido gran aceptación.
- La versión SVR4, es la versión más actualizada del sistema UNIX de AT&T.
- La filosofía original de diseño de UNIX fue la de distribuir la funcionalidad en pequeñas partes: los programas.

- En general, en las máquinas UNIX, los comandos no se ejecutan físicamente en la computadora en la cual se está tecleando, sino en aquella a la que uno se ha conectado.
- Una computadora UNIX ofrece generalmente una serie de servicios de red, mediante programas que se ejecutan continuamente llamados daemon.

2.6.2 Arquitecturas de Cómputo.

La determinación de un apropiado nivel de distribución de los componentes de la aplicación es de fundamental importancia cuando se usa el modelo cliente/servidor, esta decisión impacta en la selección de la infraestructura adecuada, en el rendimiento del sistema y en las condiciones de la implementación.

El paso principal en el establecimiento de la arquitectura de la aplicación es identificar los niveles de aplicación y los mensajes entre esos niveles. Los niveles lógicos de una aplicación como: interface de usuario, procesamiento y datos, son conocidos como capas lógicas, el siguiente paso es estructurar y hacer coincidir esos niveles en una configuración física cliente/servidor. Por ejemplo, el nivel de la aplicación puede ser colocado en una sola máquina, lo que es llamado capa simple; o en una máquina cliente y una servidor, lo que es llamado dos capas físicas; o también en una máquina servidor y en otra adicional a ellas, a lo que se llama tres capas físicas. En la mayoría de los ambientes corporativos se puede encontrar el uso de todas las configuraciones, pues cada una de ellas resuelve necesidades específicas.

Arquitectura de Dos Capas.

En la arquitectura de dos capas en donde se cuenta con una máquina cliente y una servidor, se pueden dar una serie de combinaciones que dan como resultado varias arquitecturas cliente/servidor de dos capas: la primera de ellas es la configuración basada en lo que se podría llamar presentación remota, los datos y la lógica de la

aplicación residen en el servidor, sólo la interfaz de usuario esta localizada en el cliente.

Una segunda configuración podría estar basada igual que en la primera en los datos y en la programación están posicionadas en el servidor, pero a diferencia de ella, se sitúa la interface de usuario en el cliente y también en el servidor, la totalidad de los programas se comunican a través de RPC (llamadas a procedimientos remotos), esta es usada en aplicaciones más robustas, como puede ser de tipo corporativo. Hay proveedores que soportan esta arquitectura como es el caso del manejador de base de datos y herramientas Oracle; una más es que la lógica de la aplicación reside totalmente del lado del cliente y los datos en el servidor, esta aplicación es usada para aplicaciones de tipo departamental. La figura 2.6.2.1 muestra la arquitectura cliente/servidor de dos capas.

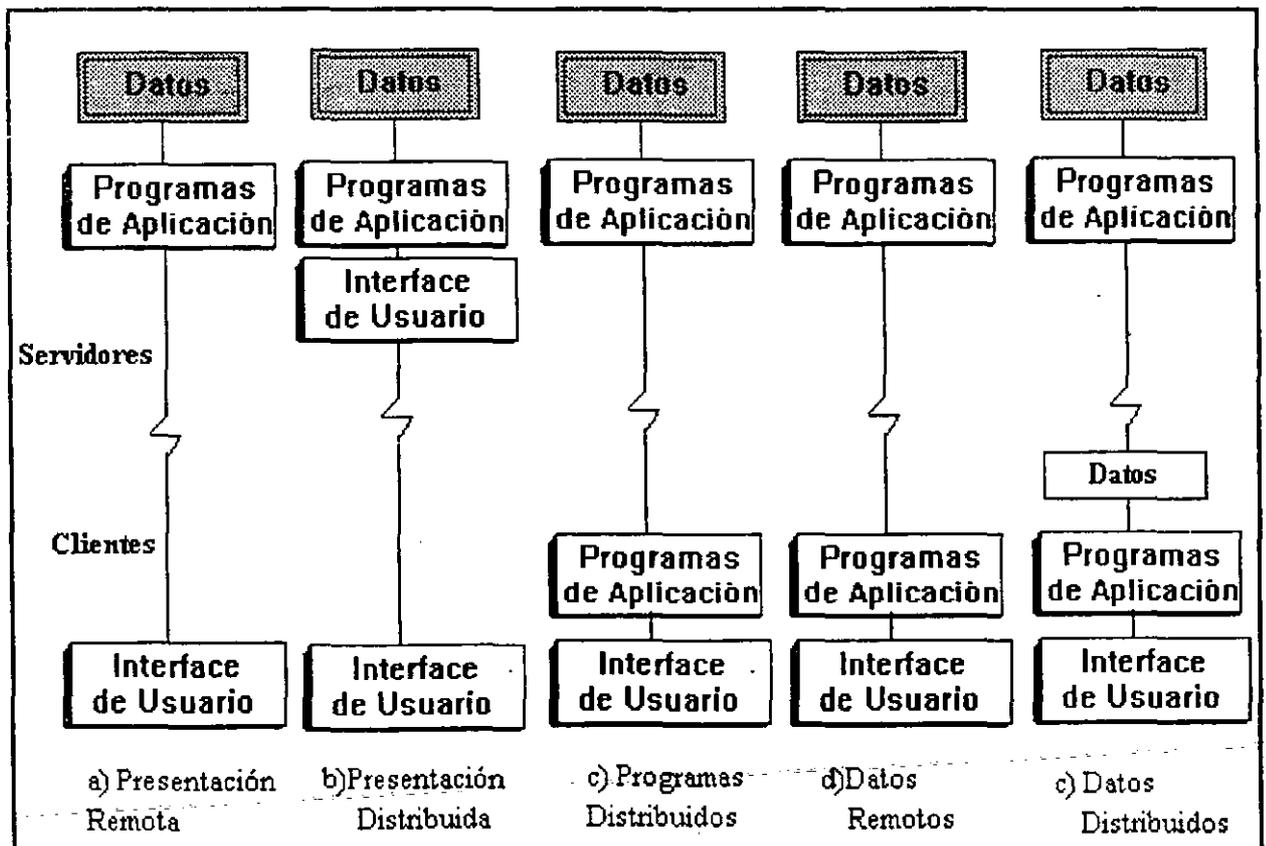


Figura 2.6.2.1 Arquitectura de Dos Capas.

Arquitectura de Tres Capas.

En un modelo cliente/servidor de tres capas, la aplicación cliente/servidor está repartida en tres tipos de máquinas: una máquina para el front-end, que puede ser una PC o NC (Computadora de Red), una máquina intermedia para el manejo de las aplicaciones y otra máquina para el back-end que puede ser incluso un mainframe. Es importante que la interacción en cada capa use un modelo cliente/servidor. Cuando sucede que la interacción entre un servidor y otro es a través de transferencia de archivos, entonces estamos hablando de una arquitectura híbrida en vez de una arquitectura de tres capas.

La figura 2.6.2.2 muestra las diferentes combinaciones de cómo acomodar las tres capas lógicas (front-end, servidor de aplicaciones y back-end) en tres capas físicas.

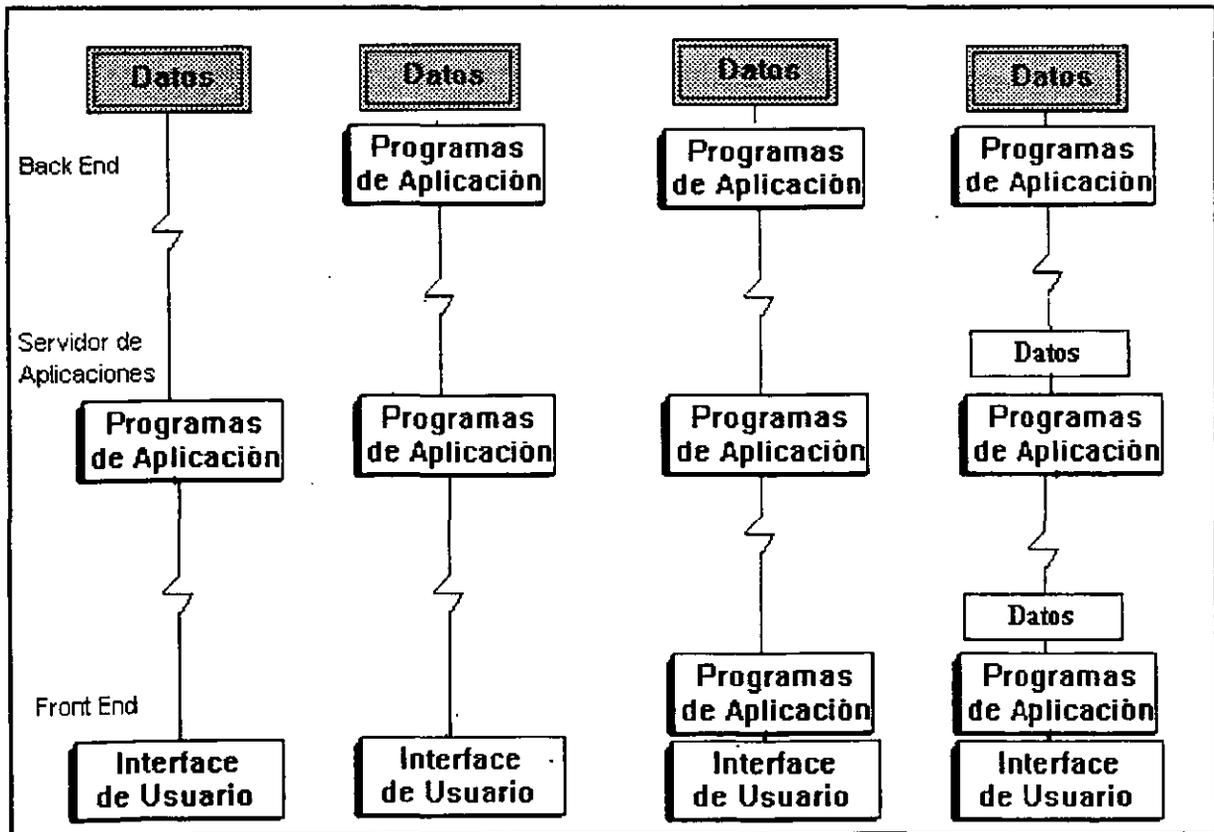


Figura 2.6.2.2 Arquitectura de Tres Capas.

Una aplicación puede ser configurada como una, dos, tres o arquitectura mixta de capas físicas, lo que se quiere mostrar son las diferencias entre las arquitecturas en términos de aplicación, flexibilidad, independencia de usuarios, rendimiento, disponibilidad, costo inicial, costo de actualización, manejabilidad y seguridad.

2.6.3 Lenguajes de Programación para la Interfaz del Usuario.

Programación de CGI's.

Los CGI's se pueden programar en varios lenguajes como Pascal, C, C++, Fortran o lenguajes de tipo script como Perl, TCL o Shell de Unix (por lo que también se les llama scripts CGI). Lo importante es que la aplicación CGI que se genere en cualquiera de los lenguajes, sea soportada en la computadora o computadoras en la que se encuentra tal aplicación, ya que cuando el cliente realice una petición para que se ejecute dicho CGI, éste ejecutará los procesos requeridos en la computadora servidor, por lo que dicha aplicación debe considerar las características de ese equipo.

HTML.

HTML sirve para desarrollar documentos de hipertexto en el WWW. Este lenguaje es un método estándar para estructurar y escribir información de tal forma que es legible para todos los browsers que circulan por Internet. Una característica de HTML es que se pueden generar documentos en este lenguaje con la ayuda de un simple editor de texto, por otro lado HTML es un lenguaje fuertemente estructurado o, mejor aun, requiere que el usuario desarrolle sus documentos utilizando una estructura específica, ya que el propósito es transmitir la estructura de los documentos entre todos los usuarios que deseen consultarlo. Otra característica importante es que los códigos de control introducidos en los documentos ocupan poco espacio, por lo que los archivos HTML suelen ser de tamaño pequeño.

HTML tiene una estructura que se define a partir de comandos básicos, estos comandos se pueden clasificar en los siguientes grupos:

- Comandos Estructurales. Estos comandos identifican a un archivo como un documento HTML y proporcionan cierta información relacionada con los datos contenidos en él.
- Comandos de Formato de Párrafo. Estos comandos especifican operaciones con los párrafos (saltos de línea, introducción de rayas separadoras, etc.) así como diferentes tipos de títulos (seis niveles distintos).
- Comandos de Formato de Caracteres. Estos le permitirán aplicar diferentes estilos (tales como negrita, cursiva o subrayado) a los caracteres contenidos en el documento.
- Comandos de especificación de listas. HTML puede trabajar con varios tipos de listas, incluyendo listas numeradas y con tablas.
- Comandos de Hiperenlaces. Estos comandos le permitirán introducir enlaces que transportarán a los usuarios a otra parte del mismo documento, a otro documento incluido en el servidor o a otro servidor de cualquier parte del mundo.
- Comandos de Integración. Estos comandos permiten acceder a información almacenada en formato multimedia. Estos comandos muestran imágenes, acceden a archivos de sonido y visualizan películas digitales.

Los documentos HTML son archivos de texto en formato ASCII estándar, que tendrán la extensión HTML en MS-DOS y HTML en UNIX

Una de las restricciones que se tienen en el formato HTML, es que aunque permite mostrar imágenes, no permite presentar una animación en tiempo real en la misma página, esto se puede conseguir con la inclusión de algún vídeo, pero se tiene la desventaja de que se presenta como una aplicación separada del documento HTML, sin embargo, el formato GIF, tiene la posibilidad de almacenar una secuencia de imágenes en un solo archivo y presentarlas consecutivamente en un browser, las imágenes pueden ser secuencias de alguna animación, además dentro del archivo gráfico, se almacena información acerca del tiempo en que se irán cambiando las

imágenes así como el número de veces que se repetirá la presentación de todas las imágenes.

JAVA.

Podría decirse que después del impacto que tuvo Mosaic en sus inicios, Java actualmente tiene un impacto equiparable. Java es un lenguaje de programación, con el cual se pueden diseñar aplicaciones que pueden ejecutarse en varias plataformas (Windows 3.x, Windows NT, UNIX con X-Windows o Motif, Windows 95), dichas aplicaciones están orientadas al WWW y el HTML, algunas de las características de las aplicaciones Java son las siguientes:

- Ejecutar sonidos en línea en tiempo real, cuando algún usuario acceda a la página que lo contiene.
- Ejecutar música de fondo durante la consulta de una página.
- Animaciones.
- Vídeo en tiempo real.
- Ejecutar operaciones que afectan el estado de la página en tiempo real (como cambiar el tipo, tamaño o color del texto; modificar colores de fondo, etc.).
- Dividir el área de visualización en el browser en varias áreas (frames).
- Ejecutar validación de información en tiempo real.

Las características descritas, son ejecutadas por los browsers sólo si estos soportan dichas aplicaciones tales como HotJava, Internet Explorer o Netscape (a partir de la versión 2.0).

Independientemente de lo anterior, Java es un lenguaje de programación para aplicaciones distribuidas. No solamente permite agregar nuevos tipos de contenido a las páginas, como lo hacen Netscape e Internet Explorer (con sus extensiones de etiquetas al HTML estándar), permite agregar tanto el contenido, como el código necesario para interactuar con dicho contenido. La ventaja que se proporciona con esta

forma de programación es la de tener independencia de las aplicaciones respecto del browser, ya que no importa la versión del browser que se este utilizando o el protocolo, ya que se envía tanto el contenido como el código para que la aplicación pueda ejecutarse.

Otra de las característica de Java es en cuanto a interacción, si se tiene una base de datos, la cual se quiere poner en el WWW, se requiere de una aplicación CGI que se ejecute por lo general en ambiente UNIX, para poder interpretar los datos que nos proporciona el usuario y de esa forma a través del CGI realizar algún proceso para poder regresar al usuario la información de la base de datos, sin embargo si la base de datos se encuentra en un mainframe, que no interpreta HTTP, resulta más difícil el acceso a los datos, ya que se requiere un proceso intermedio para poder utilizar el protocolo HTTP. Con Java se evita lo anterior, enviando el código, para que el usuario pueda interactuar directamente con la base de datos, y no necesariamente se tenga que utilizar el puerto 80 para tener acceso a la información.

Además, una aplicación Java se ejecuta de igual forma en cualquier arquitectura en la que se encuentre un browser que soporte Java, por ejemplo Netscape 2.0 el cual está disponible para Windows 95, Windows NT, MacOS, Sun OS 4.1.3, SGI-IRIX, OSF/1, HP-UX.

Java no solo es para crear aplicaciones WWW, es un lenguaje de programación de propósito general como Fortran o C++, pero sin algunos de los problemas que estos lenguajes tienen. Las características de Java como lenguaje son las siguientes:

- Es simple en su estructura sintáctica, de modo que no se tiene que agregar una gran conjunto de características adicionales.
- Es orientado a objetos. Casi todo en Java es una clase, un método o un objeto. Solamente los elementos básicos no lo son (operadores y tipos de datos).
- Es independiente de la plataforma. Los programas en Java son compilados a un formato de código - byte, que puede ser leído y ejecutado en diversas plataformas.

- Es seguro. Puede ejecutarse en ambientes que son propicios para introducción de virus.
- Alto desempeño. Java puede ser compilado a velocidades cercanas a las de C++.
- Multitarea. Un sólo programa en Java puede tener varios y diferentes procesos ejecutándose independiente y continuamente.

La dirección electrónica donde se puede obtener más información sobre ésta herramienta es: <http://www.javasoft.com/nav/whatis/index.html>.

Perl.

Perl (Practical Extraction and Report Language) conjunta varias características de otros programas, tales como: AWK, SED, TR, BASIC y C. También es conocido como un lenguaje de tipo script. Perl no compila los programas, ya que un programa escrito en este lenguaje, no genera un ejecutable. Perl en realidad es un intérprete, el procedimiento que realiza con un programa es: leer el programa, verificarlo sintácticamente y ejecutar cada una de las instrucciones, las velocidades que alcanza en la ejecución de programas es cercana a las de C, el procedimiento anterior es repetido cada vez que se ejecute el programa. La sintaxis de Perl es muy parecida a la sintaxis que se utiliza en los shell de UNIX, otra característica de Perl es la facilidad que provee para utilizar bases de datos de tipo xbase, dispone de rutinas de nivel de software de cuarta generación.

La popularidad que actualmente tiene Perl en Internet es debido a la capacidad que se tiene de aplicarlo en el diseño de formas para las páginas de Web, obtener estadísticas y manejar catálogos y bases de datos en los servidores de WWW, para lo anterior, se han diseñado librerías, las cuales permiten tener un mejor control y facilidad de uso en las aplicaciones CGI por medio de Perl. Estas librerías ofrecen funciones altamente específicas en la descodificación de datos enviados por los usuarios a través de formas, con estas rutinas el manejo de esa información así como su procesamiento se facilita en gran medida. Perl actualmente existe para plataformas UNIX y para DOS.

2.7 ELECCIÓN DE LA SOLUCIÓN ÓPTIMA.

Para la elección de la solución óptima se partirá de la propuesta de tesis de utilizar herramientas Oracle, debido a que la empresa para la que se desarrollará el sistema cuenta con las licencias de los productos de este proveedor de software.

Oracle siempre ha sido una empresa líder e innovadora en el campo de la tecnología informática. Comenzó con la tecnología portable de bases de datos relacionales, hasta la más reciente tecnología de Oracle, que permite a los actuales diseñadores y desarrolladores de sistemas poder construir aplicaciones del siglo XXI, que van desde sistemas de misión crítica para el procesamiento de transacciones interactivas que admiten miles de usuarios, hasta almacenes de datos multiterabyte para la asistencia a la toma de decisiones.

Actualmente Oracle es un manejador de base de datos objeto-relacional cuyas principales características son la portabilidad de sus aplicaciones en una gran variedad de plataformas y sistemas operativos, además de la óptima administración de sus recursos que hacen que la información almacenada cuente con seguridad, control de acceso, consistencia y protección de los datos. De la versión Enterprise Edition de Oracle 8 utilizaremos las siguientes herramientas que simplificarán y harán más fácil la implementación de este sistema:

- Enterprise Manager, nos facilitará la administración de la Base de Datos y nos apoyará en la ejecución automática de programas que depuren los datos, además de realizar la carga de datos en las tablas correspondientes.
- Cartucho de Contex, el cual nos servirá para realizar búsquedas de palabras en los datos que se encuentran almacenados en las tablas.
- PL/SQL, como lenguaje de programación para el Front End, debido a la simplicidad para integrar sentencias HTML con instrucciones de SQL estándar.

Además de las herramientas de Oracle, y partiendo de las opciones de solución presentadas en el capítulo 2.6, referentes a: sistema operativo, arquitectura y lenguajes de programación para la interfaz del usuario, se han elegido como solución óptima las siguientes:

Sistema Operativo.

La plataforma que se utilizará para el diseño del sistema será Windows NT 4.0 Server, debido a que se cuenta con el equipo adecuado para su instalación y las licencias de este sistema operativo. Aunque en un futuro se pretenderá realizar una migración hacia una plataforma Unix debido a que este sistema operativos cuenta con las mejores estadísticas de soporte a usuarios concurrentes en Internet. A continuación se muestra la tabla 2.7.1 que compara las características de estos sistemas operativos, donde se observa que la mejor opción es Unix.

Características	Windows NT	UNIX
Funcionalidad	<ul style="list-style-type: none"> • Es multiusuario. • Es multitarea. • Se utiliza en ambientes de negocio medianos. 	<ul style="list-style-type: none"> • Es multiusuario. • Es multitarea. • Se utiliza para investigación científica.
Confiabilidad	<ul style="list-style-type: none"> • Pantalla azul de la muerte. 	<ul style="list-style-type: none"> • Pánico del Kernel.
Adm. Del Sistema	<ul style="list-style-type: none"> • Posee un Graphical User Interface. 	<ul style="list-style-type: none"> • Posee un Graphical User Interface.
Desempeño	<ul style="list-style-type: none"> • Más espacio en memoria y disco. • Requiere una interface gráfica. 	<ul style="list-style-type: none"> • El Kernel se confecciona de acuerdo a las necesidades. • No requiere una interface gráfica como NT.

Tabla 2.7.1 Windows NT vs UNIX.

Arquitectura.

Debido a que el sistema propuesto tiene como objetivo trabajar en un ambiente de Internet, se optó por utilizar una arquitectura de tres capas la cual fue creada para desarrollar grandes aplicaciones en este ambiente de manera simplificada. En cambio en una arquitectura de dos capas el desarrollo de una aplicación en Internet representa un mayor problema debido a que se tiene que ejecutar la aplicación en la parte del cliente y en la parte del servidor por no contar con un servidor de aplicaciones que reduzca la carga de procesamiento en los clientes.

Características	Dos capas Físicas	Tres capas Físicas
Tipo de Aplicación	Soporte de decisiones departamentales	Aplicaciones Corporativas
Flexibilidad y crecimiento	Bueno	Muy bueno
Independencia de usuario	Bueno	Muy bueno
Rendimiento	Congestión en la red	Muchas opciones
Disponibilidad	Bueno	Muy bueno
Costo inicial	Medio	Medio
Costo de Actualización	Medio	Bajo
Control y Manejo	Bueno	Muy bueno
Seguridad	Bueno	Bueno

Tabla 2.7.2 Comparación entre Diferentes Capas.

La arquitectura de tres capas conocida en Oracle como Arquitectura de Cómputo en Red (Network Computing Architecture) está formada por un cliente delgado para el

nivel de presentación, un servidor de aplicaciones para los procedimientos operacionales y un servidor de bases de datos para el almacenamiento y manipulación de la información; lo cual permite atender a muchos usuarios concurrentes en una aplicación en Internet. Otras ventajas de la arquitectura de tres capas (NCA) sobre la arquitectura de dos capas (cliente/servidor) las podemos ver en la tabla 2.7.2.

Lenguajes de Programación para la Interfaz de Usuario.

Para la programación de la interfaz del usuario se requiere de un lenguaje diseñado especialmente para Internet, que sea rápido, sencillo en su estructura y estándar. El lenguaje que cubre estas características es HTML (Lenguaje de Marcación de Hiper Texto). Otra ventaja que ofrece este lenguaje respecto a los demás, es que integrado con PL/SQL de Oracle por medio del Web Development Kit, facilita la programación de código para acceder a la base de datos.

En la tabla 2.7.3 a manera de resumen se muestra, la elección de la solución óptima para la implementación del sistema:

MÓDULO	ELECCIÓN
Sistema Operativo.	Windows NT 4.0 Server.
Arquitectura.	Arquitectura de Tres Capas (Network Computing Architecture).
Herramientas para el manejo de la Base de Datos.	<ul style="list-style-type: none"> • Oracle Enterprise Manager. • Cartucho de Context. • PL/SQL.
Lenguaje de Programación para la Interfaz de Usuario.	HTML y PL/SQL.

Tabla 2.7.3 Elección de la Solución Óptima.

**CAPÍTULO 3.
DESARROLLO E
IMPLEMENTACIÓN
DEL PROBLEMA.**

3.1 CARACTERÍSTICAS Y CONFIGURACIÓN DE ORACLE 8.

A medida que los sistemas de información evolucionan hacia arquitecturas abiertas basadas en redes y hacia sistemas distribuidos, las empresas se enfrentan a los siguientes retos:

- Mantenimiento de la robustez y el rendimiento de los sistemas existentes en nuevos sistemas que se encuentran en fase de desarrollo.
- Gestión de la interoperatividad de los datos existentes y nuevos.
- Identificación de los mejores productos que se pueden implantar en ese entorno.
- Integración y gestión de un entorno multinivel y multiplataforma.

La Arquitectura de Cómputo en Red (NCA) de Oracle es abierta y está basada en estándares, y permite a los departamentos de tecnología de la información dedicar menos tiempo a cuestiones de interoperatividad y más tiempo a la implantación de nuevos sistemas.

Tanto para las aplicaciones empresariales tradicionales como para el comercio electrónico en World Wide Web, Oracle8 y NCA aportan la potencia, el rendimiento, la robustez, la integración en red y la flexibilidad necesarias para soportar las aplicaciones más exigentes. Oracle8 incluye importantes mejoras de rendimiento y de utilización de recursos.

Oracle8 Enterprise Edition, Release 8.0 también incluye mejoras en prácticamente todas las demás áreas del servidor Oracle, aumentando la disponibilidad global, el rendimiento, la capacidad de gestión, el soporte de tipos de datos multimedia y la funcionalidad de réplica de datos.

Oracle8 Enterprise Edition, Release 8.0 se ha diseñado para abordar las siguientes necesidades:

- Alta disponibilidad y capacidad de gestión con tablas e índices divididos en particiones.
- Paralelismo mejorado.
- Mayor rendimiento y mejor gestión de aplicaciones de data warehouse.
- Procesamiento de transacciones en línea a un nivel comparable al ofrecido por un mainframe.
- Mejor administración de la seguridad, así como mejor soporte de sistemas distribuidos, tecnología de objetos y extensibilidad, herramientas de gestión de fácil uso, perfecta migración e interoperatividad con versiones anteriores.

Nuevas Opciones en Paralelo.

Las transacciones de inserción, actualización y eliminación se pueden ejecutar ahora en paralelo con Oracle8. Estas operaciones, denominadas DML, se ejecutan en paralelo a través de varios procesos. Al ejecutar estas operaciones en paralelo, la transacción se realiza con mayor rapidez que si la misma transacción se ejecutara en serie.

Las operaciones DML en paralelo complementan las consultas en paralelo al aportar un procesamiento en paralelo tanto de las consultas como de las modificaciones. Las operaciones DML en paralelo resultan útiles en entornos de soporte a la toma de decisiones (DSS) o de data warehouse, en los que son habituales la mayoría de las operaciones DML. Sin embargo, las operaciones DML en paralelo también pueden agilizar los trabajos batch que se ejecutan en una base de datos OLTP.

La inserción en paralelo en una tabla mejora el rendimiento al dar formato y escribir los bloques de disco directamente en los archivos de datos, eludiendo los cuellos de botella que se producen en la memoria caché intermedia y en la gestión del espacio. En este caso, cada proceso de exploración de la consulta de inserción introduce datos en un segmento situado por encima del límite máximo de ocupación de la tabla.

Cuando finalizan todos los procesos de inserción, la transacción se graba en la memoria y el límite máximo de ocupación se traslada más allá de los nuevos segmentos.

Para utilizar la función de DML en paralelo, debe activarse antes de ejecutarse la operación de inserción, actualización y eliminación. Normalmente, las operaciones DML en paralelo se realizan en programas de procesamiento batch o desde dentro de una aplicación que ejecuta una inserción, actualización y eliminación en masa. Hay disponibles nuevos parámetros para especificar el paralelismo de las instrucciones DML.

Mejoras Introducidas en Servidor Paralelo.

Numerosas prestaciones introducidas en Oracle8 Parallel Server (Servidor Paralelo) mejoran el rendimiento, la escalabilidad, el uso de la memoria y la disponibilidad. Un administrador de bloqueos distribuido, común e integrado sustituye a los administradores de bloqueos suministrados por los distintos proveedores de sistemas operativos, aumentando el rendimiento y la portabilidad en la mayoría de las plataformas.

Varias mejoras del rendimiento aumentan considerablemente la rapidez de Oracle8 Parallel Server. Al integrar el DLM en Oracle8, Parallel Server queda liberado de la necesidad de aportar la funcionalidad más básica a través de todas las plataformas. Esto también permite que Oracle Parallel Server esté disponible en plataformas que anteriormente no admitían el uso compartido de recursos a través de nodos.

Asimismo, el DLM sitúa ahora los bloqueos en la memoria caché para evitar la comunicación de la información relativa a los bloqueos a través de nodos. El uso de pings, o la pugna por acceder a bloques a través de los nodos, también se ha reducido al mejorar el algoritmo que controla el acceso a los bloques requeridos.

Además, nuevas tablas globales aumentan la capacidad de gestión. Oracle Parallel Server utiliza ahora su propio administrador de bloqueos distribuido (DLM) integrado para procesar las solicitudes de recursos entre nodos, como muestra la figura 3.1.1.

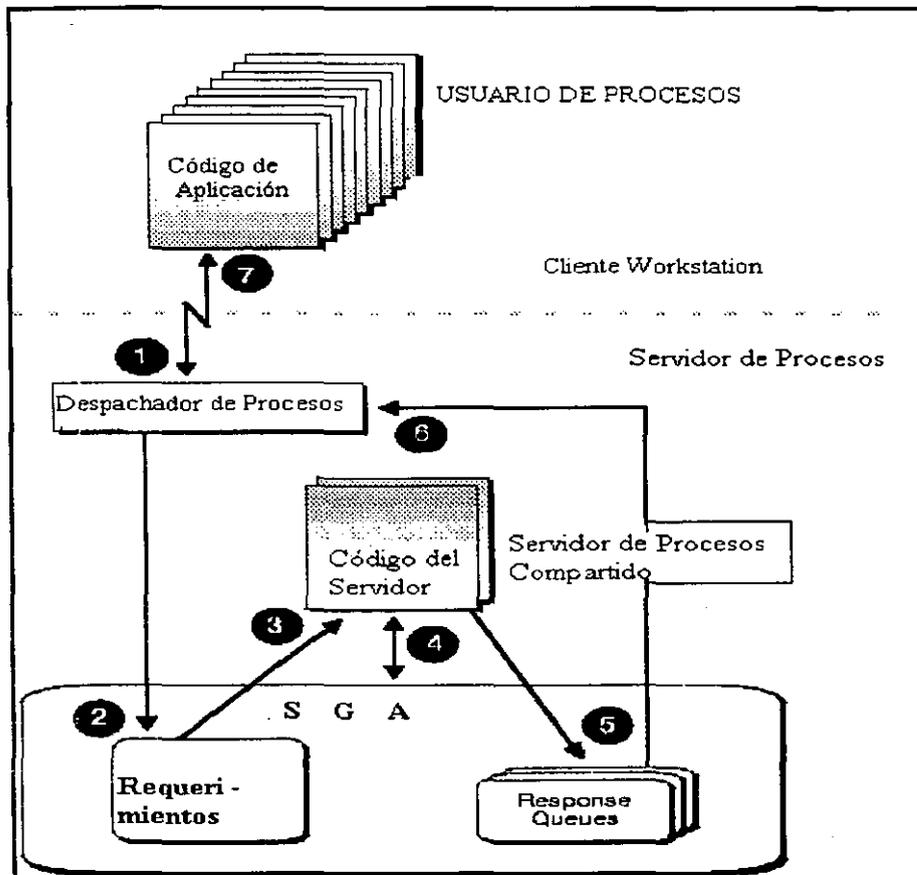


Figura 3.1.1 Servidor Paralelo.

Traslado Transparente de Aplicaciones en Caso de Fallo.

En el supuesto de que un nodo de Parallel Server experimente un fallo, la función de traslado transparente de aplicaciones realiza la migración de las conexiones de los usuarios y restablece automáticamente sus sesiones en otro nodo. Las aplicaciones de los usuarios siguen ejecutándose y los usuarios pueden incluso no ser conscientes del fallo. Esto aporta una disponibilidad continua en caso de producirse interrupciones

tanto planificadas como imprevistas. Dependiendo del nivel de actividad general que se desee desarrollar en el equipo cliente, el fallo de una instancia podría dar como resultado una migración completamente transparente de las conexiones de los usuarios al nodo que asume el control tras producirse el fallo. Es posible especificar que todas las consultas se sitúen en la memoria caché del sistema cliente de manera que se puedan regenerar las instancias en el nodo de reserva. Asimismo, se puede disponer de una sesión conectada previamente al nodo secundario, en cuyo caso se elimina el tiempo necesario para volver a conectarse a una nueva instancia.

El traslado transparente de aplicaciones en caso de fallo no sólo resulta útil para aumentar la disponibilidad, sino que se puede utilizar para compensar la carga manualmente o para efectuar un cierre controlado del sistema.

Si el número de usuarios que se conectan a una instancia es excesivo, es posible dar por finalizadas sus sesiones y realizar una migración transparente a otro nodo.

También se puede cerrar un nodo y hacer que los usuarios efectúen una migración transparente a un nodo de reserva una vez finalizada su transacción actual. Las paradas del sistema ya no tienen por qué interrumpir el trabajo de los usuarios.

Mejor Administración de la Seguridad.

Oracle8 incluye un servidor de seguridad que aporta un único entorno de inicio de sesión para permitir la administración central de usuarios y funciones. Oracle Security Server cumple con normas de seguridad, para autenticación de claves públicas/privadas.

También se admite la autenticación mutua entre cliente y servidor con el fin de ofrecer una protección contra bases de datos "ilegales" diseñadas para capturar las comunicaciones de los sistemas clientes.

Se incluye además un juego de herramientas para la generación de firmas digitales que permite crear aplicaciones diseñadas para identificar la manipulación no autorizada de los datos. Oracle8 también ofrece un nivel mejorado de mantenimiento y administración de contraseñas. Se puede definir un perfil de contraseña para aplicar un sistema de seguridad con Oracle8. Las contraseñas pueden caducar transcurrido un determinado tiempo, o bien puede verificarse su validez. Un usuario puede crear su propia contraseña o bien utilizar la función estándar almacenada para comprobar la longitud, el contenido o la reutilización de nuevas contraseñas.

Base de Datos Objeto-Relacional.

Oracle8 representa un importante avance en la tecnología de gestión de datos con la introducción de un paradigma objeto-relacional. Los esquemas y las aplicaciones de bases de datos son cada vez más complejas. A menudo, varias aplicaciones independientes con datos similares, tales como información de clientes, facturación y envío, existen en distintos esquemas de base de datos, y el departamento de sistemas de información debe gestionar su interacción.

La gestión de la información se convierte en una tarea sumamente difícil consistente en integrar distintos objetos relacionales y diferentes aplicaciones, posiblemente de distintos proveedores, para formar un modelo de datos de usuario final más uniforme y coherente.

Al mejorar la base de datos relacional con extensiones de objetos para proporcionar una completa solución objeto-relacional, Oracle aborda la necesidad de simplificar la modelización de datos y ampliar la base de datos con nuevos tipos de datos. Entre las características objeto-relacionales de Oracle cabe mencionar las siguientes:

- Tipos de objetos para ampliar el servidor de maneras específicas para cada aplicación.

- Procedimientos externos como métodos de objetos.
- Soporte de objetos en el lado del cliente para agilizar la navegación.
- Evolución de un entorno relacional existente.
- Herramientas de desarrollo para la modelización de objetos.
- Soporte de datos no estructurados (imágenes, vídeo, texto, etc.), como muestra la figura 3.1.2, así como soporte de estándares abiertos: JAVA.

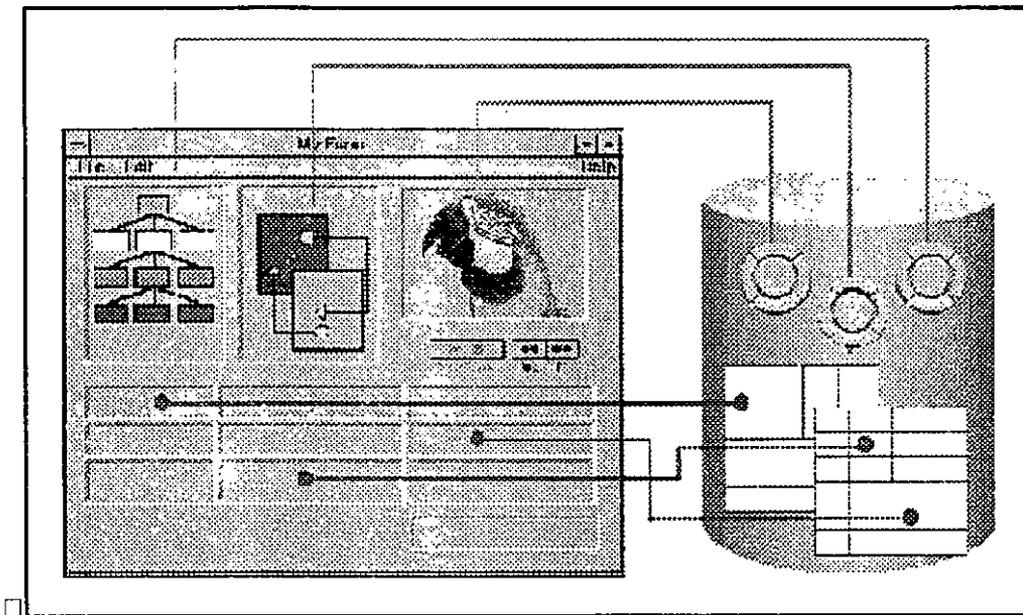


Figura 3.1.2 Soporte de Datos no Estructurados.

Los objetos se han integrado completamente en Oracle8, a todos los niveles del servidor. Todas las propiedades avanzadas del servidor Oracle se proporcionan con objetos, incluidas avanzadas posibilidades de Oracle tales como su modelo de concurrencia, alto rendimiento, escalabilidad, fiabilidad, capacidad de gestión y disponibilidad.

Tipos de Objetos.

Los tipos de objetos aportan una manera de ampliar el sistema de tipos de datos relacionales de Oracle. Las bases de datos relacionales admiten tres tipos de datos:

caracteres, números y fechas. Los tipos de objetos permiten definir nuevos tipos de datos y utilizarlos de la misma manera que se usarían los tipos de datos relacionales normales. Por ejemplo, se puede crear un nuevo tipo denominado Dirección que puede contener datos, denominados atributos, tales como la calle, la población y el código postal. El tipo de objeto también puede contener métodos, tales como Distancia, para calcular la distancia entre las direcciones. Estos métodos se pueden escribir en PL/SQL, C o Java. Los tipos de objetos permiten a los desarrolladores de aplicaciones codificar la lógica de la aplicación en la base de datos o en el servidor de aplicaciones de nivel intermedio, en lugar de utilizar código en el lado del cliente. Todas las aplicaciones podrán compartir entonces la lógica de los nuevos tipos de datos, por lo que los desarrolladores no tienen que volver a escribir el código.

Esta característica ofrece las ventajas de crear componentes de código reutilizables y una segmentación transparente de las aplicaciones, por lo que el código puede residir y ejecutarse en el nivel que mayor rendimiento aporte: cliente, servidor de aplicaciones o servidor de base de datos. Oracle8 sigue el estándar emergente SQL3 en lo relativo a la definición de tipos de objetos y las técnicas de modelización de objetos. SQL3 define la sintaxis para crear y modificar tipos de objetos, generar y almacenar identificadores de objetos (OID's), crear referencias o punteros de objetos y modelizar colecciones de objetos similares.

Llamadas a Procedimientos Externos desde la Base de Datos.

Oracle proporciona un método rápido y seguro para que la base de datos realice una llamada a un programa externo escrito en C, C++ o Java. La llamada también se puede realizar a través de protocolos abiertos como HTTP. Los procedimientos externos permiten utilizar el código existente de las aplicaciones, o bien escribir código altamente optimizado para fines específicos. Así mismo, pueden utilizarse procedimientos externos para interactuar con otras aplicaciones o con dispositivos especializados, como sistemas integrados.

Evolución de los Entornos Relacionales.

Oracle8 se ha diseñado para facilitar la evolución hacia la nueva funcionalidad orientada a objetos por parte de los usuarios, ya que todas las aplicaciones existentes serán compatibles con las versiones posteriores. Las nuevas extensiones objeto-relacional se fundamentan en la misma base que la funcionalidad relacional, por lo que los usuarios no tienen que desechar o volver a escribir sus aplicaciones relacionales existentes antes de realizar la migración a Oracle8.

A diferencia de lo que ocurre con otras bases de datos objeto-relacionales, este diseño permite que las aplicaciones relacionales más antiguas –que siguen leyendo y escribiendo filas y columnas– coexistan con las nuevas aplicaciones orientadas a objetos, que leen y escriben objetos. Oracle8 proporciona vistas de objetos para recuperar datos relacionales y representar los datos a un cliente como si fuesen un objeto, y viceversa.

Por ejemplo, un sistema relacional de introducción de pedidos ya existente podría necesitar un nuevo front-end para World Wide Web. Las aplicaciones existentes que acceden al esquema relacional pueden seguir en funcionamiento y se puede desarrollar un nuevo conjunto de vistas de objetos como una representación de objetos para el cliente Web. Las aplicaciones nuevas y antiguas se pueden basar en los mismos datos, pero cada una tiene su propia representación.

Otras Mejoras.

Las tablas organizadas mediante índices almacenan las columnas de una tabla en los nodos de hoja de la estructura de índice B-tree de Oracle8. De esta manera se reducen los requisitos globales de almacenamiento cuando la mayoría de las columnas se indexan almacenándolas una sola vez, no en un índice y en una tabla aparte.

Las tablas organizadas mediante índices también reducen el tiempo de acceso al recuperar todas las columnas desde una ubicación, en lugar de hacerlo desde dos. El rendimiento de PL/SQL aumenta, al igual que el rendimiento al realizar llamadas a SQL desde PL/SQL, y viceversa. El procesamiento de restricciones se ha mejorado significativamente en Oracle8. Ahora pueden utilizarse índices no únicos para aplicar restricciones UNIQUE (único) y PRIMARY KEY (llave principal), lo cual elimina los índices redundantes y permite que los índices sigan siendo válidos mientras las restricciones están desactivadas.

La verificación de restricciones diferidas desplaza la comprobación de las restricciones de integridad desde el final de la ejecución de la instrucción al final de la ejecución de la transacción, simplificando la codificación de determinadas operaciones relacionadas con las limitaciones de integridad.

Migración.

Una utilidad de migración sencilla y rápida reconstruye el diccionario de datos y convierte los archivos de control, los archivos de registro y los bloques de datos. La utilidad de migración convierte cualquier base de datos Oracle 7.1, 7.2 ó 7.3 en una base de datos Oracle8. Las aplicaciones Oracle7 se ejecutan sin cambio alguno con Oracle8, mientras que los comandos distribuidos de Oracle8 se ejecutan con Oracle7, y viceversa

Área Global del Sistema (SGA).

SGA es el lugar de la memoria donde la base de datos Oracle almacena la información sobre sí misma. Lo hace así dado que la memoria es la forma más rápida y más eficiente de permitir que los procesos se comuniquen. La figura 3.1.3 muestra la relación entre el área global de sistema y los procesos de los usuarios. Esta estructura de memoria es accesible a todos los procesos de usuario y de servidor. SGA es un

grupo de estructuras en memoria compartida. Si múltiples usuarios están conectados, los datos en el SGA están compartidos y monitoreados entre los usuarios.

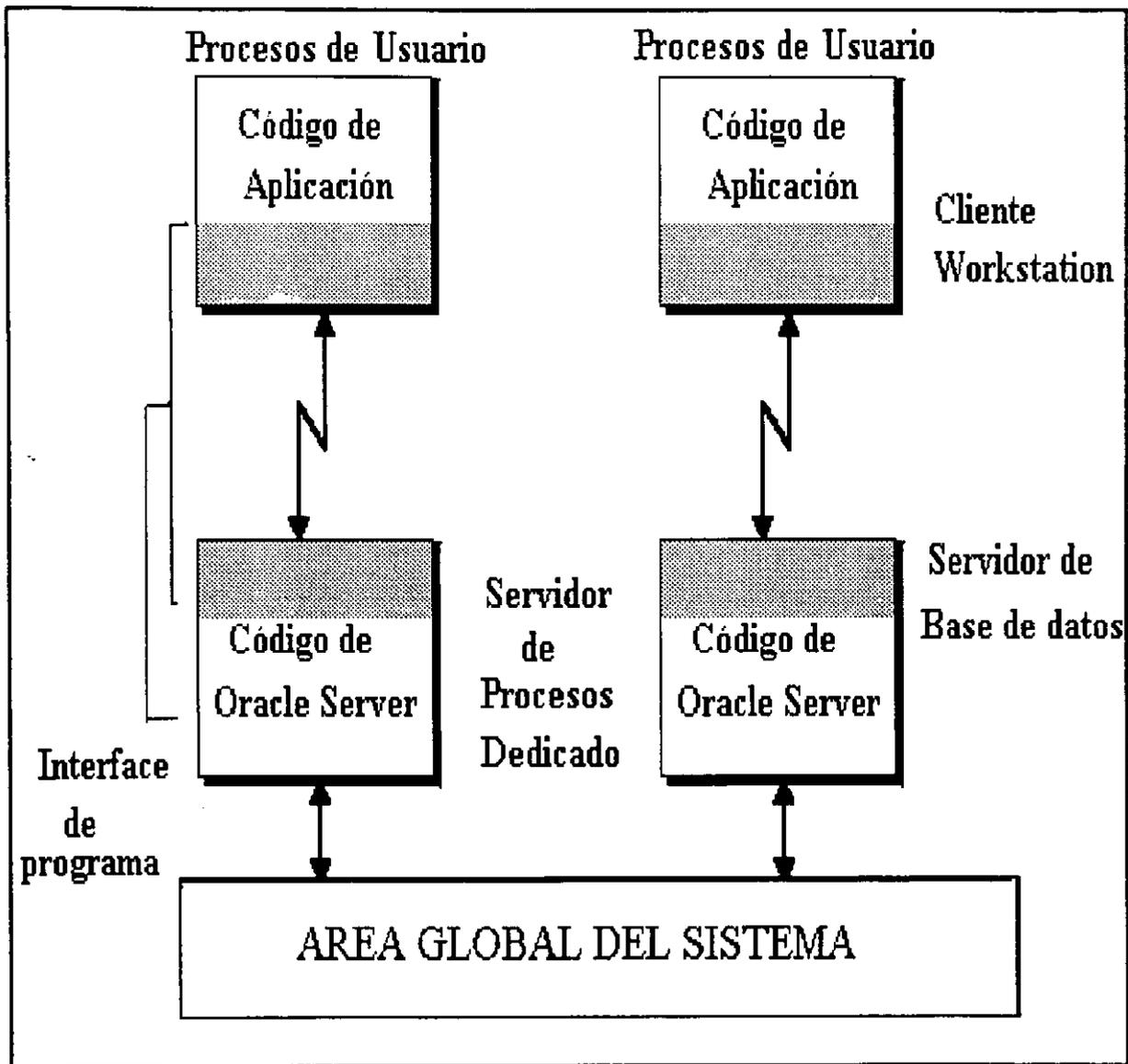


Figura 3.1.3 Relación entre el Area Global del Sistema y los Procesos de usuario.

Servidor de Procesos.

Antes que se accesen los datos, un servidor de procesos debe colocar los datos en el buffer cache de la base de datos, para procesar los enunciados, un servidor de

procesos usa memoria compartida en el SGA. Dado que el SGA es el mecanismo por el que distintos procesos de cliente y de servidor se comunican, es importante entender sus diferentes componentes. El SGA del servidor Oracle se divide en los siguientes componentes fundamentales.

- **Memoria Caché del Buffer de Datos.**

La memoria caché del buffer de datos es donde Oracle almacena los bloques de datos más recientemente utilizados. En otras palabras, es la memoria caché de datos. Cuando se introduce información en la base de datos, se almacena en bloques de datos.

La memoria caché del buffer de datos es un área de memoria en la que Oracle coloca dichos bloques de datos, para que un proceso de usuario pueda acceder a ellos. Antes de que ningún proceso de usuario pueda acceder a parte de los datos, éstos deben residir en la memoria caché del buffer de datos. El tamaño de la memoria caché del buffer de datos tiene un límite físico, por lo que cuando Oracle la ha llenado, deja los bloques de mayor uso en la memoria caché y elimina los bloques de menor uso.

Un punto importante es que si un proceso de cliente necesita una información que no está en la memoria caché, la base de datos accede a la unidad de disco físico, lee los bloques de datos necesarios y los coloca en la memoria caché del buffer de datos. Esto hace que todos los procesos de servidor y de cliente puedan aprovecharse de la lectura de disco físico.

- **Memoria Caché del Diccionario (Memoria Caché de Fila).**

Una memoria caché de diccionario está formada por registros extraídos del diccionario de datos. El diccionario de datos contiene toda la información que Oracle necesita

para gestionarse, tal como qué usuarios tienen acceso a la base de datos Oracle, qué objetos de la base de datos poseen y dónde se localizan dichos objetos.

- **Buffer del Registro de Rehacer.**

Otro nombre común que designa a los registros de rehacer es registro de transacción. Cualquier transacción que pueda registrarse en el registro de rehacer (los registros de rehacer en línea se necesitan para propósitos de recuperación) debe residir primero en el buffer del registro de rehacer.

Se trata de un área de memoria reservada para este propósito. La base de datos, periódicamente, vacía este buffer en los registros de rehacer en línea.

- **Área Compartida SQL.**

Si se piensa en el área compartida SQL como en la memoria caché de programa, es el área donde se almacenan todos los programas. Los programas de una base de datos Oracle se basan en un lenguaje estándar denominado SQL. Esta memoria caché contiene todas las órdenes SQL analizadas que están listas para ejecutarse.

Resumiendo, el SGA es el gran comunicador. Es el lugar de memoria donde se coloca la información a que pueden acceder los procesos de servidor y de cliente. Está dividido en grandes áreas: la memoria caché de datos, la memoria caché de registro de rehacer, la memoria caché de diccionario y la memoria caché compartida SQL.

La Figura 3.1.4 muestra las memorias caché que Oracle mantiene en el SGA. Denominamos sqlarea a la memoria caché SQL, pudiendo usar ambos términos indistintamente.

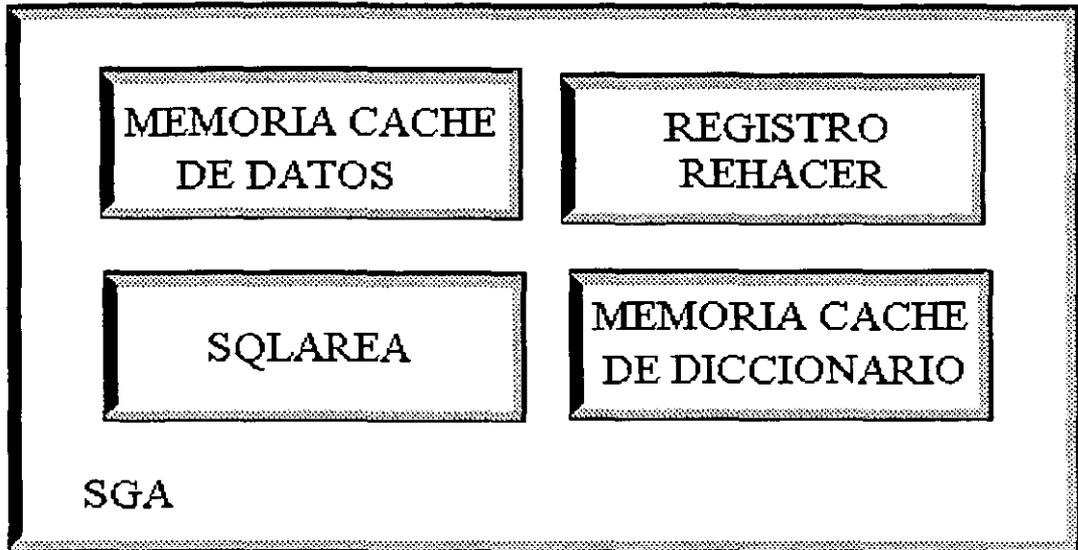


Figura 3.1.4 Memorias Caché en el SGA.

Área Global de Programa (PGA).

El PGA es un área de memoria utilizada por un único proceso Oracle. El área global de programa no se comparte; contiene datos e información de control para un único proceso. Contiene información tal como matrices internas y variables de sesión del proceso. Al igual que un sistema de intercomunicadores doméstico, las distintas partes del proceso pueden comunicarse entre si, pero no con el mundo exterior.

Instancia Oracle.

Es una parte de la memoria de la computadora y los procesos auxiliares requeridos para acceder a una base de datos Oracle. Una base de datos Oracle está formada por uno o más controles, archivos de datos y archivos **redo log** (de respaldo) nombrados en archivos de control.

Dicho sencillamente, una instancia Oracle es un conjunto de procesos de servidor Oracle que tienen su propia área global de sistema y un conjunto de archivos de base

de datos asociados con ellos. La tabla 3.1.5 muestra la estructura física del conjunto de archivos.

ESTRUCTURA FÍSICA	DEFINICIÓN
Estructura de archivos	Contiene todos los datos de la base de datos: estructuras lógicas, contiene tablas e índices, son físicamente mostrados en los archivos de datos.
Archivos Redo Log	Se refiere a los registros de todos los cambios hechos a la base de datos para propósitos específicos.
Archivos de control	Registra la estructura física y estado de la base de datos.

Tabla 3.1.5 Estructura Física.

Por ejemplo, si se tiene una computadora que contiene dos bases de datos, denominadas prd y tst. Si cada base de datos tiene su propio SGA y un conjunto separado de procesos de servidor Oracle, entonces tiene dos instancias de la base de datos.

Para que la base de datos no se confunda, cada instancia se identifica por medio de lo que conoce como SID (identificador de sistema). En la mayor parte de las computadoras UNIX, se define mediante la variable «ORACLE_SID». Luego, se nombra cada uno de los procesos de servidor de acuerdo con el SID correspondiente.

Archivos de Parametros.

La figura 3.1.6 muestra la localización de los parámetros en el SGA, y la tabla 3.1.7 los describe.

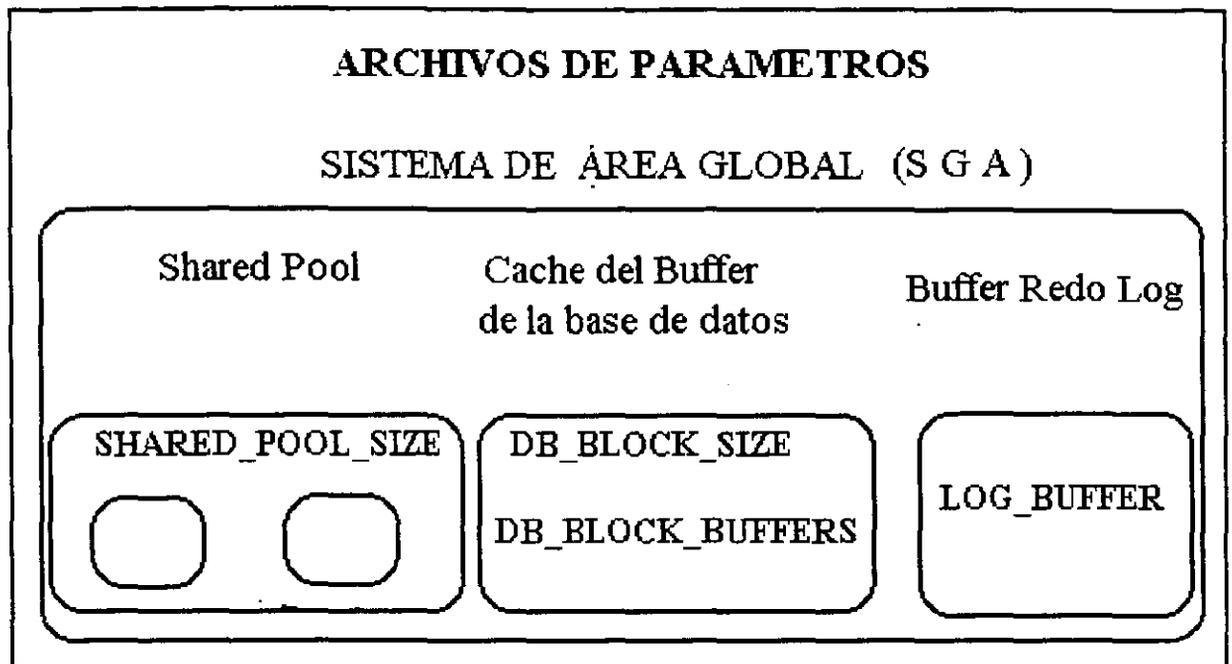


Figura 3.1.6 Localización de los Parámetros Dentro del SGA.

PARÁMETRO	DEFINICIÓN
SHARED_POOL_SIZE	Se refiere al tamaño en bytes en memoria para SQL y enunciados PL/SQL.
DB_BLOCK_SIZE	Es el tamaño en bytes de un simple bloque de datos en el buffer de la base de datos.
DB_BLOCK_BUFFERS	Es el número de buffers de la base de datos ,cada tamaño de <code>DB_BLOCK_SIZE</code> ,es destinado para el SGA.
LOG_BUFFER	Es el número de bytes destinados para el buffer del archivo redo log.

Tabla 3.1.7 Descripción de los Parámetros.

Estos elementos definen la instancia. Para iniciar una instancia Oracle debe de leer un archivo de parámetros inti < SID >.ora, un archivo de parámetros es un archivo de texto que contiene una lista de parámetros, para valores particulares de control del tamaño de memoria y procesos de una instancia Oracle, como:

- Cuanta memoria usar para la estructura de memoria en el SGA.
- Qué hacer con los archivos redo log.
- Los nombres y localizaciones de control de archivos de la base de datos.

3.1.1 Instalación y Configuración de Oracle 8.

Esta instalación se realizará bajo el sistema operativo Windows NT, antes de empezar la instalación se requiere saber lo siguiente:

- El idioma que se va a utilizar para correr Oracle8 porque existen diferentes.
- Donde se quiere que resida Oracle8, es decir el path donde se van a copiar los archivos durante el proceso de instalación.
- Conocer que productos son los que se van a instalar, porque hay diferentes tipos de instalación como: servidor, cliente y personalizada.
- Cada instalación tiene diferentes grupos de programas.

En la tabla 3.1.1 se muestran los requerimientos para la instalación de Oracle 8 en un ambiente UNIX o NT.

Instalación en el Servidor.

En la instalación del servidor, los componentes necesarios para Oracle8 Server incluyen:

- **Oracle8 Server.** Este es el Oracle8 RDBMS.

HARDWARE	REQUERIMIENTOS
Memoria	Se necesita un mínimo de 32 MB en RAM , recomendado 64MB en RAM y 128 MB si se usa ConText Cartridge.
Espacio Swap(particionado)	En general es recomendado en sistemas con grandes capacidades de memoria usar más de 1 GB, aunque puede ser reducido. Para Unix se recomiendan cuatro particiones:
Particiones de Disco	Uno para la distribución del software de Oracle, y tres para creación de bases de datos OFA-compliant.
Espacio en Disco	Es deseable un mínimo de 600 MB al instalar la distribución de Oracle Server 8.
CD - ROM	Utiliza formatos de CD-ROM High Sierra o ISO 9660 con extensión RockRidge.

Tabla 3.1.1 Requerimientos de Oracle sobre UNIX y NT.

- **Componentes de Server Networking.** (Interfaces de red). Estos incluyen el programa TNS listener, y programas SQL*Net como herramientas administrativas.
- **Oracle Enterprise Manager.** Este componente es la herramienta del nuevo sistema de administración de Oracle, proporciona una interfaz gráfica para administrar el servidor Oracle8.
- **Oracle Intelligent Agents** (Agentes Inteligentes Oracle). Estos son programas que muestran el Oracle Enterprise Manager para comunicarse con el servidor Oracle8 y utilerías. Utilizan el protocolo SMTP (Simple Management Transport Protocol) para la comunicación.
- **Documentación Oracle8** (opcional). Miscelánea de utilerías o manuales. Otras utilerías incluyen SQL*Plus, Oracle installer, SQL*Loader, **Open Database Connectivity** (conectividad abierta con base de datos, ODBC).

Instalación del Cliente.

En la instalación del cliente, el DBA (Administrador de la Base de Datos) debe de conocer las aplicaciones de los clientes. Así mismo la instalación del DBA proporciona los componentes para administrar el sistema Oracle remotamente. Estos productos incluyen:

- **Oracle Enterprise Manager** (Mencionado anteriormente).
- **SQL*Plus**. Esta es la interface primaria sobre Oracle, además de utilizarse para propósitos administrativos.
- **Oracle8 Toolbar**. (Barra de herramientas). Proporciona un rápido acceso a las aplicaciones de Oracle Enterprise Manager.
- **SQL*Net**. Los componentes SQL*Net incluyen los componentes **client-side networking** (componentes de gestión de redes del cliente) y herramientas administrativas.
- **Oracle8 Documentación** (opcional), así como misceláneas de herramientas y utilerías.

Instalación Personalizada.

Aquí se instala cualquiera de todos los productos u opciones de Oracle8 que se requieran, esta es desde CD-ROM.

Instalación Básica.

Al iniciar la instalación, colocando el CD-ROM en la unidad, se verá la siguiente pantalla de presentación, al inicio como muestra la figura 3.1.1.1.

Para comenzar la instalación simplemente posicionarse en begin y dar doble click, y el proceso comenzará a correr el programa setup.

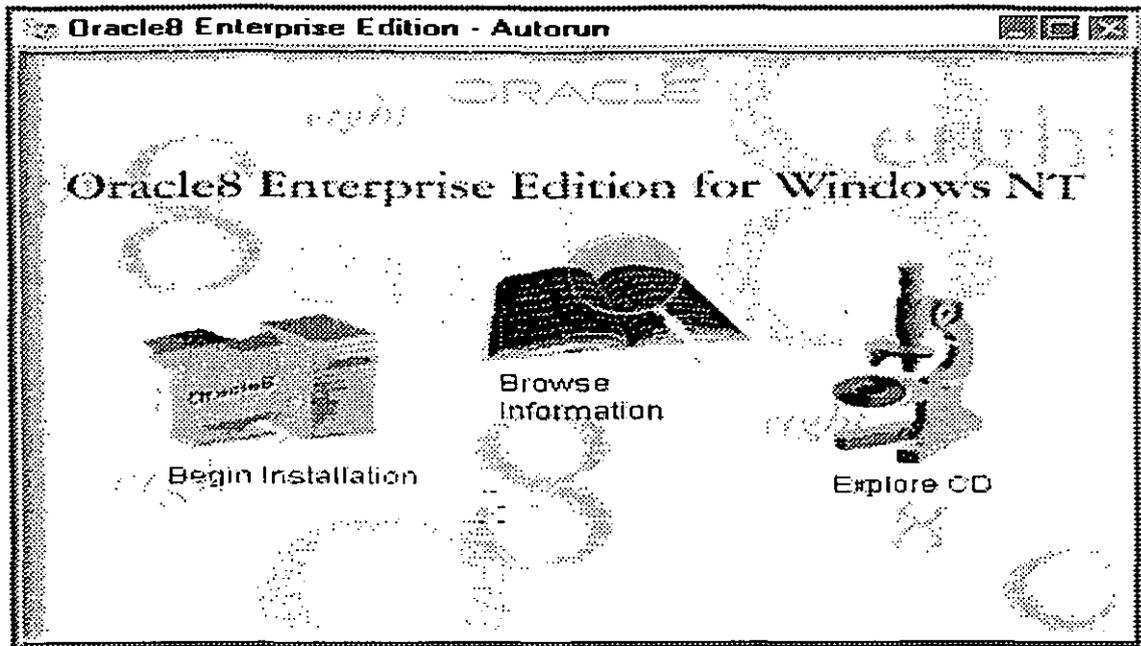


Figura 3.1.1.1 Pantalla de Presentación.

Después que el proceso ha comenzado se verá la ventana de la figura 3.1.1.2, la cual pregunta cuál idioma se quiere utilizar para correr los productos de Oracle.

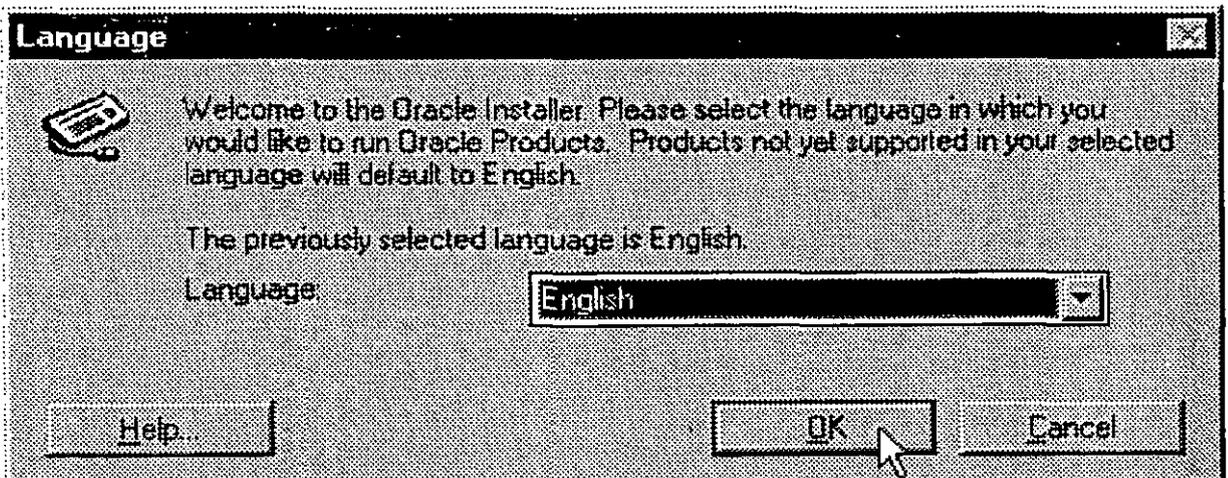


Figure 3.1.1.2 Idioma de Selección.

En la figura 3.1.1.3 se pregunta por el nombre de la compañía y el directorio donde los archivos van a ser instalados.

Antes de seleccionar el directorio, verificar que se tiene un espacio adecuado, ya que la instalación completa puede requerir más de 100MB.

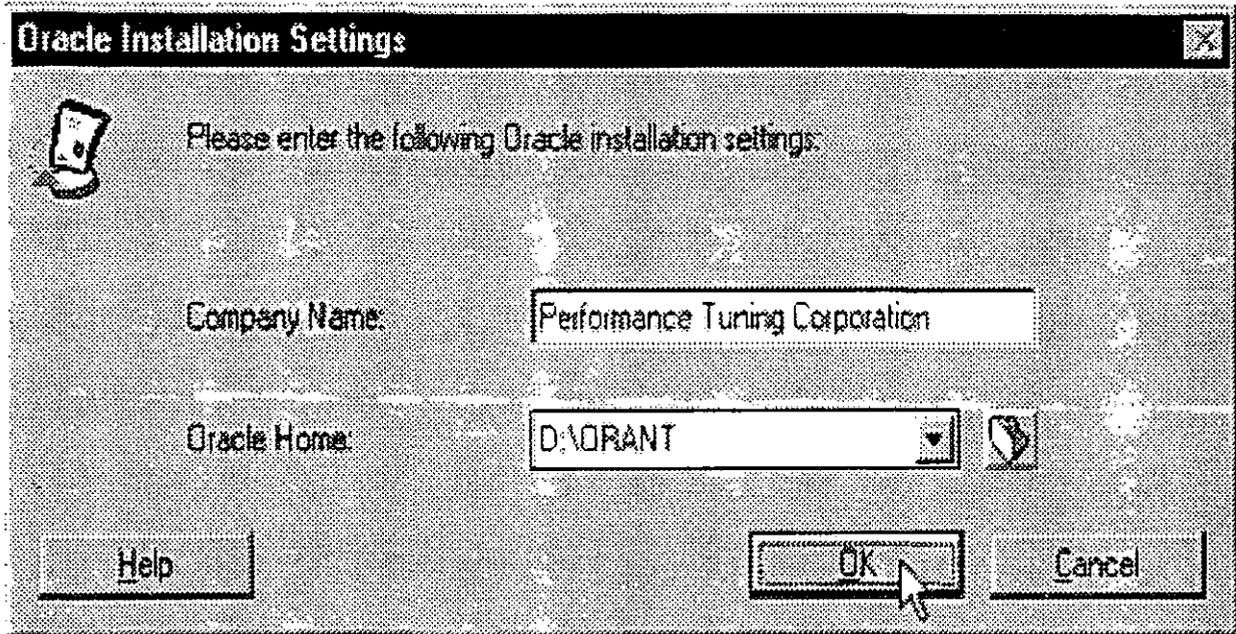


Figura 3.1.1.3 Ventana para la Compañía y Directorio Home.

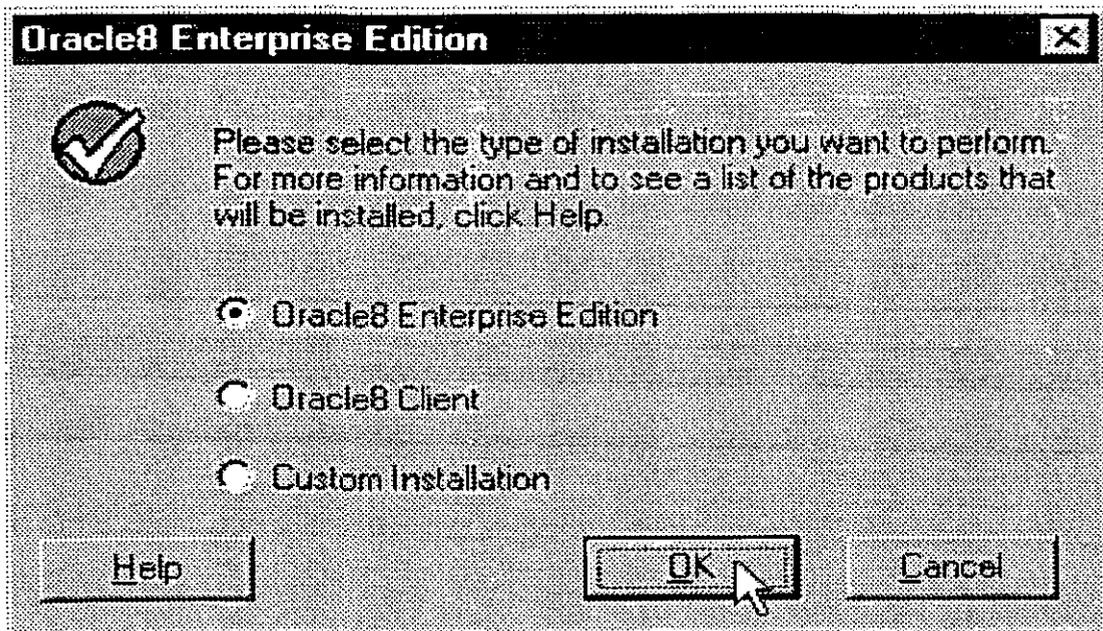


Figure 3.1.1.4 Ventana para Seleccionar el Tipo de Instalación.

Al dar un clic en el botón de OK, en la ruta especificada, los archivos y las librerías serán colocados. Después de realizado lo anterior se presenta la pantalla de la figura 3.1.1.4 la cual pregunta el tipo de instalación a realizarse, y cada una de estas se realiza por separado.

Instalación del Oracle8 Server.

Se realiza en dos secciones. La primera sección describe la instalación del software. La segunda sección describe el sistema después que la instalación ha ocurrido. Específicamente, esta sección proporciona información sobre que productos y servicios se han instalado, y cómo se ha modificado el registro.

Si se ha escogido instalar el Servidor Oracle8, pulsar el botón OK de la primera opción en el menú de la figura anterior. En este punto algunas dependencias se determinarán y se preguntará acerca de Oracle **data cartridges** (cartuchos de datos) como muestra la figura 3.1.1.5. Se preguntará después si se quiere una instalación de bases de datos típica, personalizada o ninguna como muestra la figura 3.1.1.6.

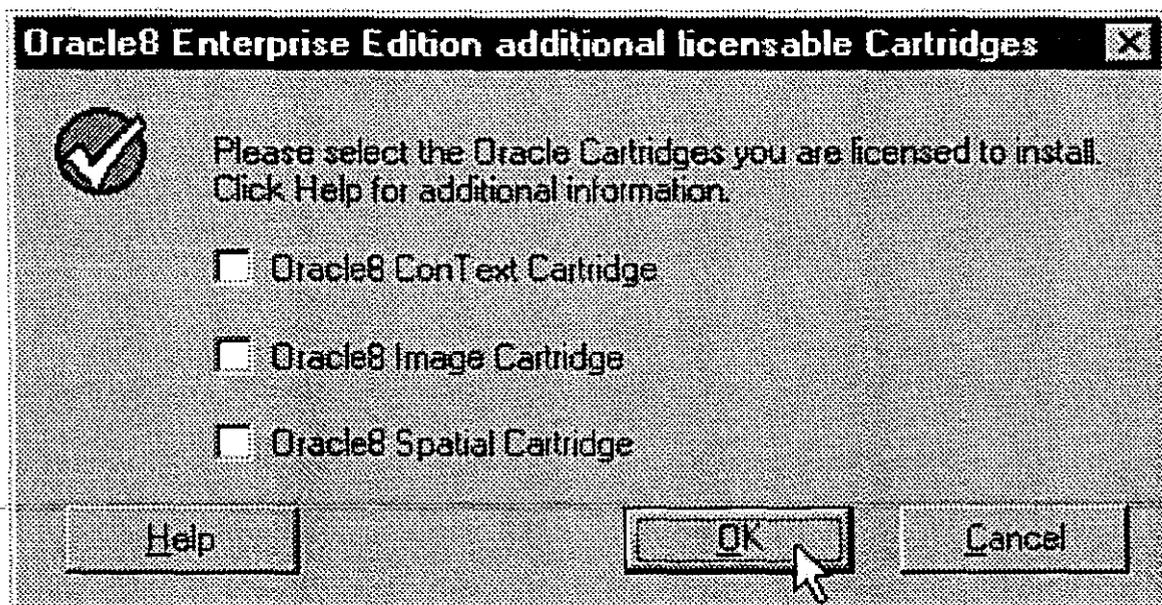


Figura 3.1.1.5 Instalación de Cartuchos de Datos de Oracle.

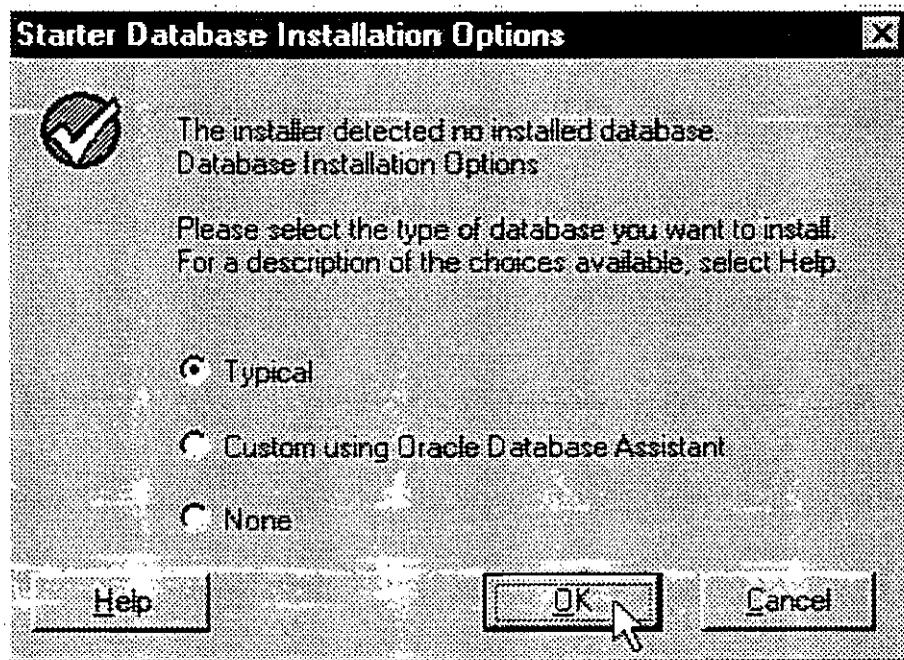


Figura 3.1.1.6 Opciones de Instalación de Base de Datos.

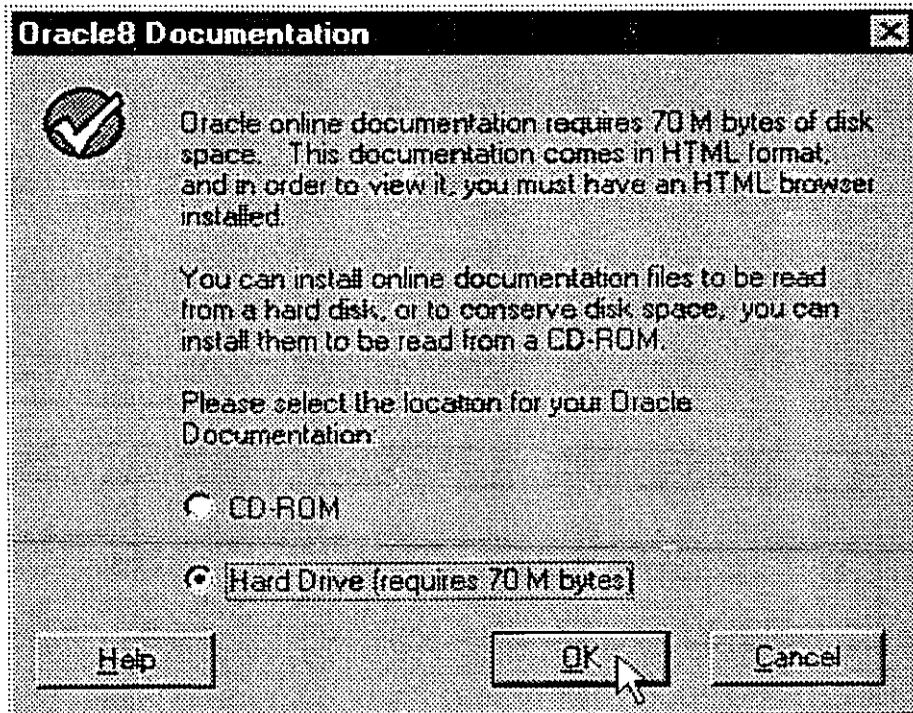


Figure 3.1.1.7 Instalación de la Documentación.

Finalmente se instalará la documentación en línea, ésta se puede acceder desde CD-ROM o instalarla en disco duro, como muestra la figura 3.1.1.7 Instalar la documentación en el disco duro requiere aproximadamente 70MB de espacio en disco. La documentación está disponible en HTML, con algunas hyperligas.

Se notificará cuando la instalación esté terminada. Además de los elementos que se han instalado, se han registrado siete servicios como se muestra la figura 3.1.1.8. Los cuales se describen a continuación:

- **Oracle Agent.** Este servicio es el agente inteligente de Oracle para el Oracle Enterprise Manager, que puede comunicarse con múltiples instancias de Oracle en un mismo sistema.

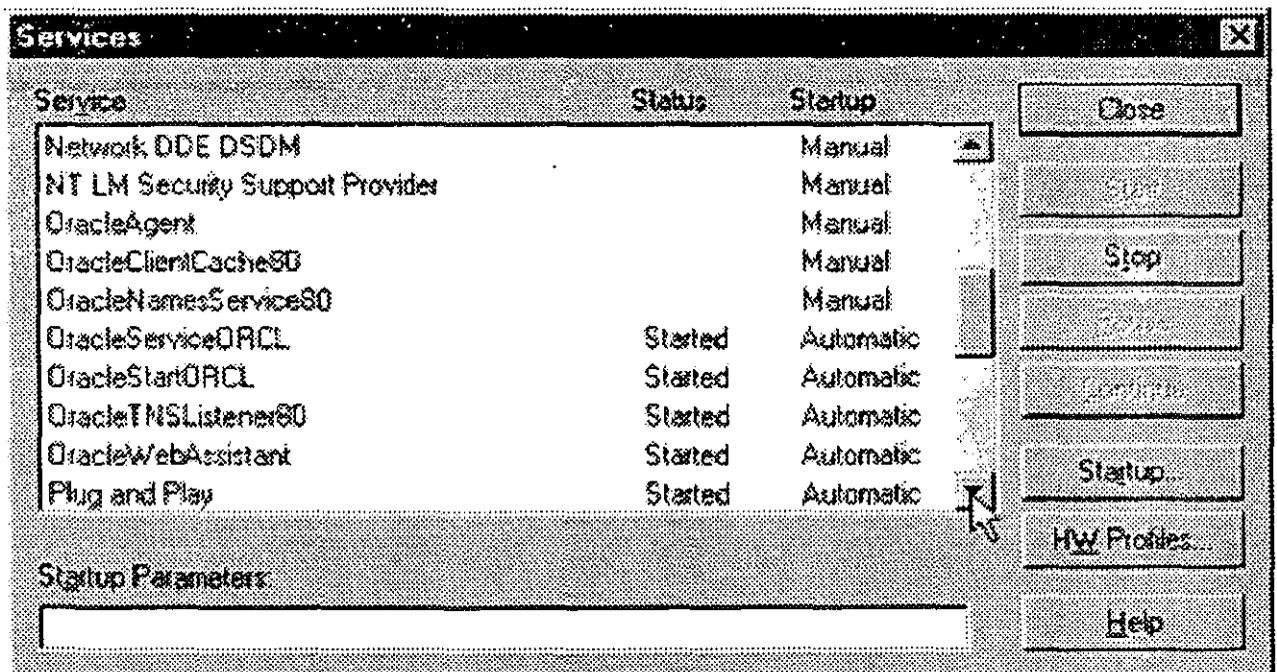


Figura 3.1.1.8 Servicios Adicionales.

- **OracleClientCache80.** Una vez que comenzó, guarda en el cache toda información recibida del servidor de nombres de Oracle.

- **OracleNamesService80.** Si se configura este sistema como un servidor de los nombres.
- **OracleServiceORCL.** Este servicio se asocia con la instancia, para SID= ORCL por default. Si una nueva base de datos es creada con un diferente SID, un nuevo servicio debe de ser creado con un nuevo SID.
- **OracleStartORCL** El servicio OracleStartORCL inicia el servicio llamado ORCL por default, cada vez que el sistema es reinicializado.
- **OracleWebAssistant.** Este servicio habilita consultas a la base de datos de Oracle para colocarlas en el WEB.

Si no se desea un servicio en particular, se puede modificar en las propiedades de servicio, e ir instalando cada uno, como muestra la figura 3.1.1.9. En automático, manual o deshabilitado. Como parte del proceso de instalación nuevos grupos de programas han sido instalados.

Remover Componentes.

Si se desea remover algún componente, simplemente llamar al instalador de Oracle a través de Oracle para Windows NT. Seleccionar cualquier componente que se quiera remover, y dar clic para remover y salir.

Descripción de los Productos.

Varios productos y componentes instalados se mostraron anteriormente, se dará una breve descripción de los productos y componentes actuales.

- **Oracle8 Server** El Oracle8 Server es el RDBMS. Con el RDBMS se encuentran todas las características del estándar de Oracle incluyendo las siguientes opciones:
- **Distribuida** Esta muestra la base de datos como va a ser distribuida a través de diferentes nodos.

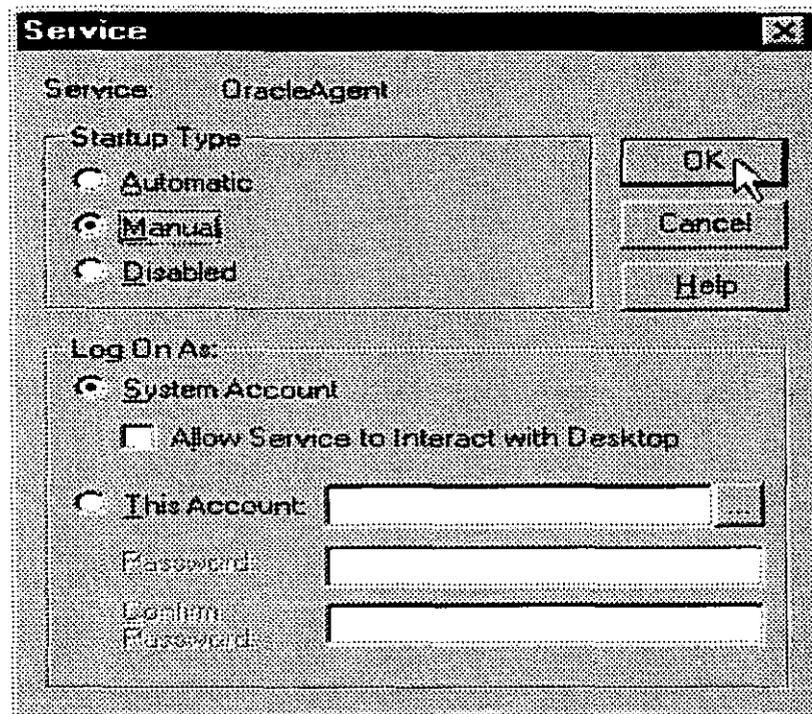


Figura 3.1.1.9 Instalación de Servicios.

- **Réplica:** Esta muestra la réplica o copia de datos para diferentes sistemas múltiples. Puede ser usado para incrementar la eficiencia, o protección en caso de algún imprevisto.
- **Queries Paralelos.** Para diferentes tipos de operaciones, las cuales pueden ser procesadas en paralelo cuando las tareas son muy largas de correr.
- **PL/SQL.** Es el lenguaje procedural, ésta opción muestra los comandos SQL para ser escritos.
- **Enterprise Manager.** Oracle Enterprise Manager es la herramienta de administración gráfica introducida recientemente por Oracle, el Enterprise Manager ayuda al DBA para simplificar algunas de las tareas necesarias, incluye una suite de herramientas incluyendo:
 - Backup Manager.** Esta opción asiste al administrador en el respaldo y recuperación de operaciones.

- Instance Manager.** Esta opción se encarga del startup (arranque), shutdown (apagado o terminación) y configuración.
- Schema Manager.** Este es usado para ver y modificar tablas y vistas.
- Security Manager.** Este se usa para el manejo de usuarios, agregarlos y asignar reglas, así como la seguridad.
- Agentes Inteligentes.** Son programas que muestra el Oracle Enterprise Manager para comunicarse con el Oracle8 Server y utilerías.
- Oracle Installer.** Es la aplicación usada para instalar y remover software.

3.1.2 Configuración del Cartucho de Context.

Para la configuración del cartucho de Context se llevaron a cabo los siguientes pasos:

- Creación de dos tablespaces para almacenar los objetos del cartucho de context.
- Un tablespace de 20M como default y un tablespace temporal de 10M.
- Ejecución de los siguientes scripts como el usuario SYS:

```
SQL> @ d:\orant\ctx80\admin\dr0crsyst ctxsys passwdctxsys context ctxtmp
```

(donde: ctxsys "usuario administrador de Context",

passwdctxsys "password del ctxsys",

context "nombre del tablespace",

ctxsys "nombre del tablespace temporal")

/* crea los objetos para usar Context */

```
SQL> connect ctxsys/ctxsys
```

```
SQL> @ d:\orant\ctx80\admin\dr0inst.sql
```

- Inicialización de los servidores de Context con la personalidad QDM.

3.1.3 Enterprise Manager.

La primera vez que se va a ejecutar OEM (Oracle Enterprise Manager), se debe de responder a la pregunta que se muestra en la siguiente figura sobre la configuración

de los repositorios OEM y software manager. Pulsar OK para invocar al programa de configuración, lo cual se lleva algunos minutos. Y creación, como se muestra en las figuras 3.1.3.1 a) y 3.1.3.1 b).

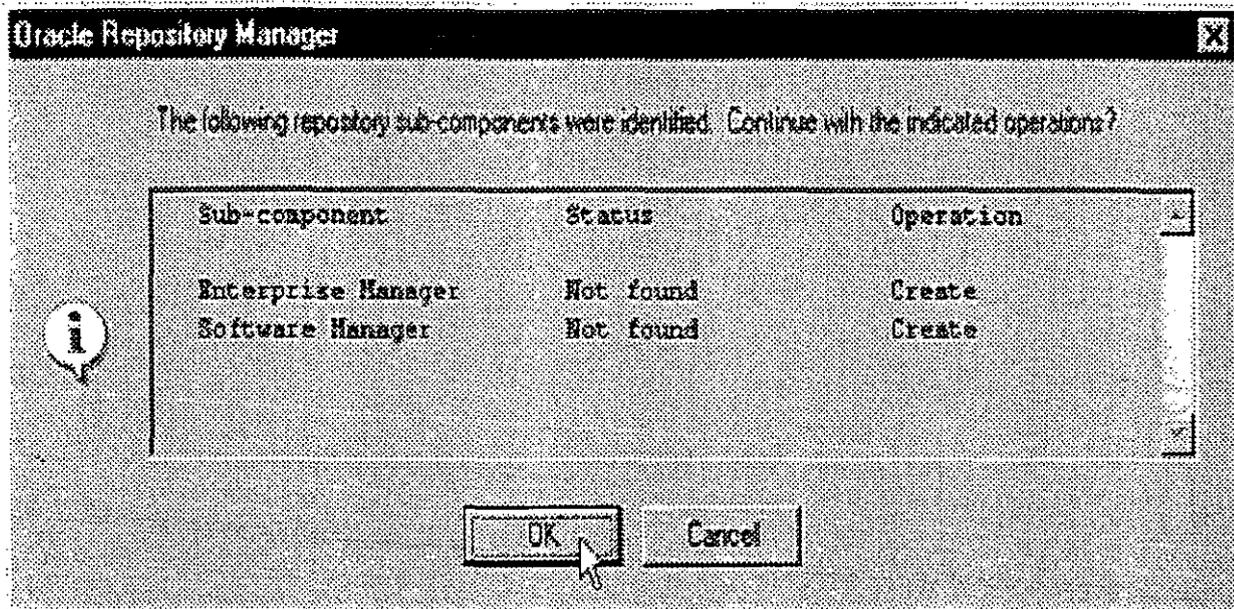


Figura 3.1.3.1 a) Cuadro de Diálogo del Repositorio Cuando se Arranca OEM.

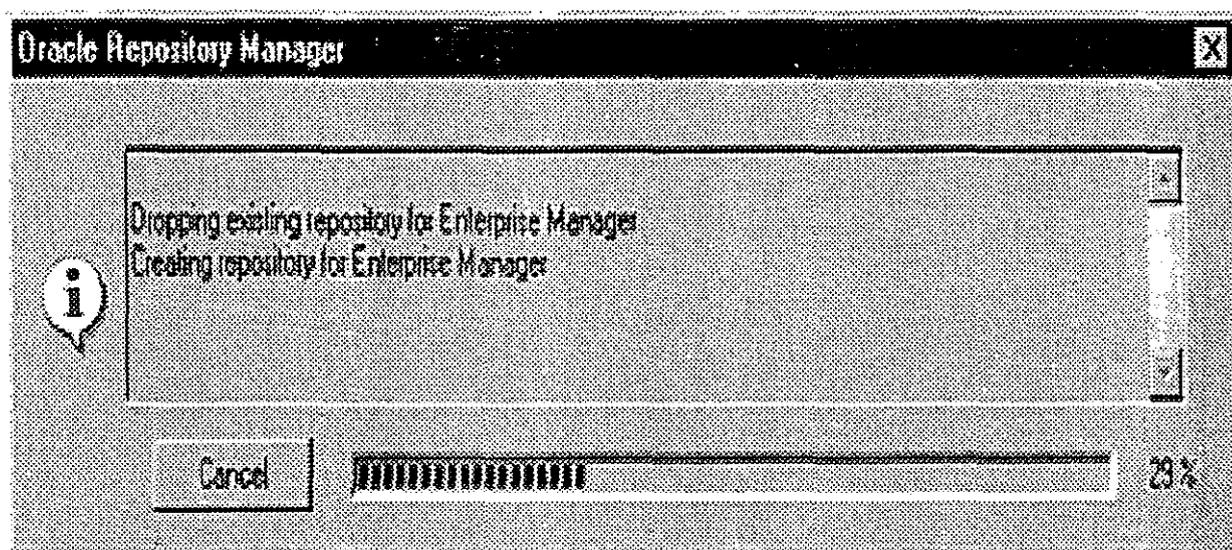


Figura 3.1.3.1 b) Creación del Repositorio de OEM.

Herramientas de Gestión OEM.

Una vez que se ha cargado OEM, se verá la ventana principal, que será una pantalla similar a la mostrada en la figura 3.1.3.2.

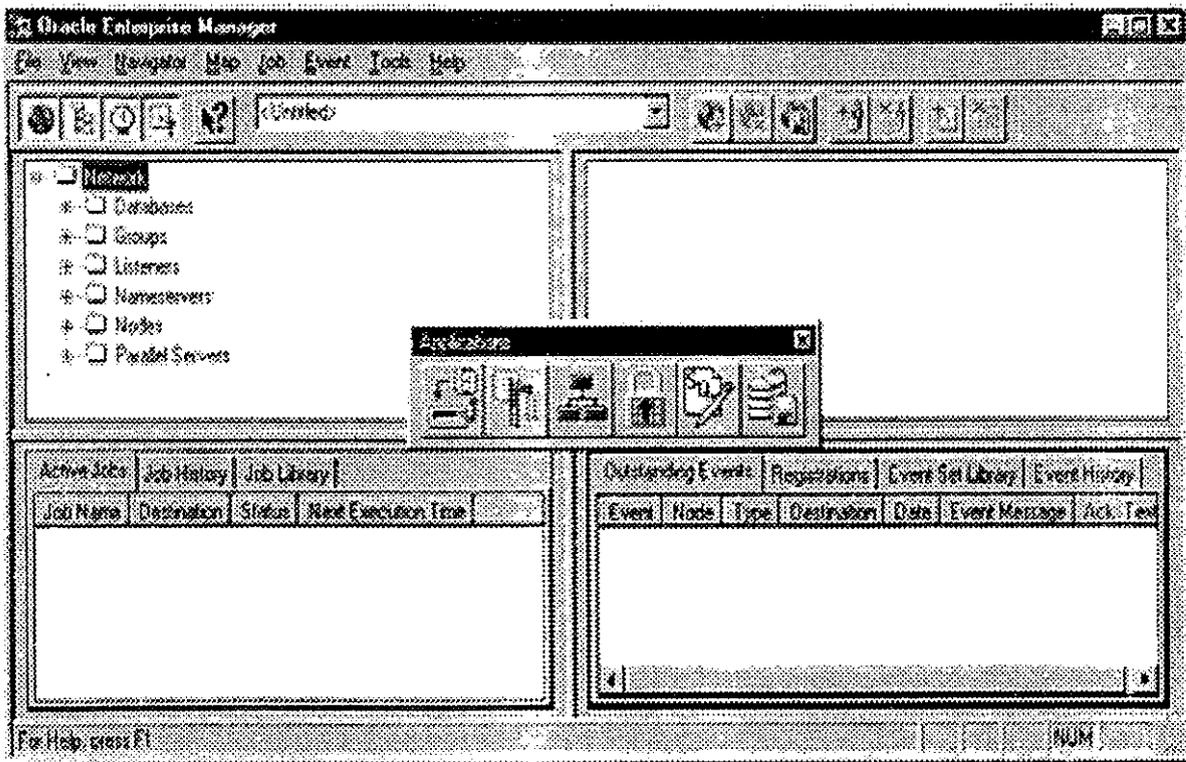


Figura 3.1.3.2 Ventana Principal.

Las herramientas que se despliegan al seleccionar Tools, en el menú de la barra superior son las siguientes:

Instance Manager (Administrador de Instancias).

Este administrador se usa para trabajar con una o más instancias Oracle, a las que se tiene acceso a través de OEM. Se puede arrancar o detener una instancia, inspeccionar o modificar la información del archivo de parámetros de inicialización y monitorear que usuarios están conectados a la base de datos.

Schema Manager (Administrador de Esquemas).

Es el corazón del conjunto de herramientas de gestión de objetos de OEM, en el que el DBA u otros usuarios privilegiados pueden trabajar con la serie de objetos de la base de datos.

El operador puede crear e inspeccionar los objetos que pertenezcan a cualquier persona de cualquier instancia disponible.

Security Manager (Administrador de Seguridad).

Desde el administrador se gestionan los usuarios y roles, estos objetos se pueden crear, modificar y eliminar.

Storage Manager (Administrador de almacenamiento).

Todas las tareas asociadas con la gestión de elementos de almacenamiento en una base de datos Oracle se controlan desde esta herramienta. Aquí se crean los archivos que almacenan los datos de la instancia; dichos archivos pueden renombrarse cuando es necesario, utilizando cuadros de diálogo de este administrador.

Data Manager (Administrador de Datos).

Aquí es donde el operador puede invocar las utilidades export, import, u Oracle Loader.

Conexión al Repositorio.

Cada usuario que se invoca a OEM debe conectarse al repositorio de OEM como lo muestra la figura 3.1.3.3.

Respecto al área de introducción de datos, nombre de usuario y contraseña siempre se piden. Servicio se refiere a una cadena de conexión que se usa para conectarse con una base de datos remota.

Connect AS (conexión como), se refiere al modo en el que se desea registrarse en el repositorio.

La mayoría de las veces se selecciona NORMAL, aunque en el menú desplegable están disponibles las opciones SYSOPER y SYSDBA para realizar, por ejemplo el arranque y la desconexión.

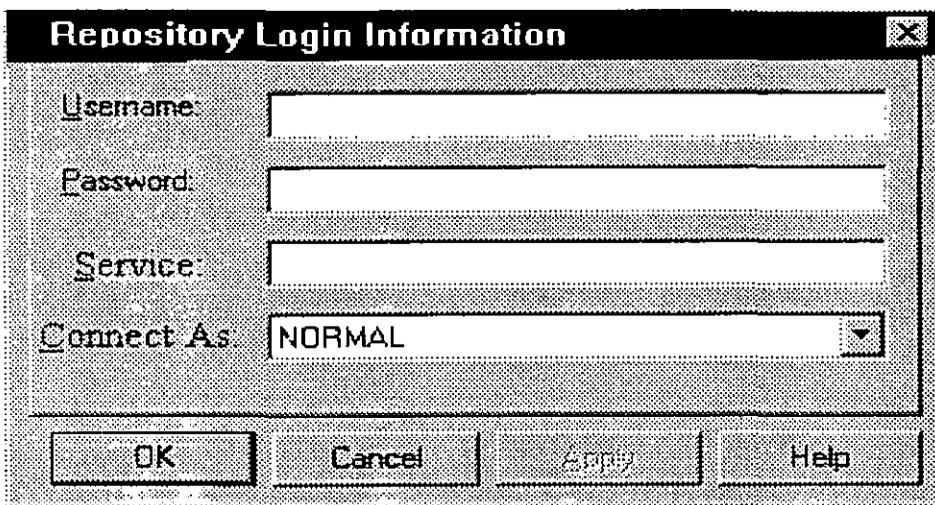


Figura 3.1.3.3 Conexión al Repositorio.

Arranque Enterprise Manager.

Después de introducir la información de registro cuando se invoca OEM, hay que hacer lo siguiente para arrancar la base de datos Oracle8:

1. Seleccionar la opción Tools de la barra de menú situada en la parte superior de la consola OEM.

2. Pulsar sobre Applications y luego seleccionar la opción Instance Manager (Administrador de Instancias) del menú desplegable.
3. Si le informa que debe de estar seleccionada una base de datos para usar esta aplicación, pulse OK para introducir dicha información .
4. Introduzca los detalles de registro apropiados, asegurándose de que se selecciona la opción SYSDBA en el área Connect As del cuadro de diálogo del registro.
5. Cuando se encuentre en Instance Manager, pulse sobre la pestaña Startup (arranque), como se muestra en la figura 3.1.3.4.

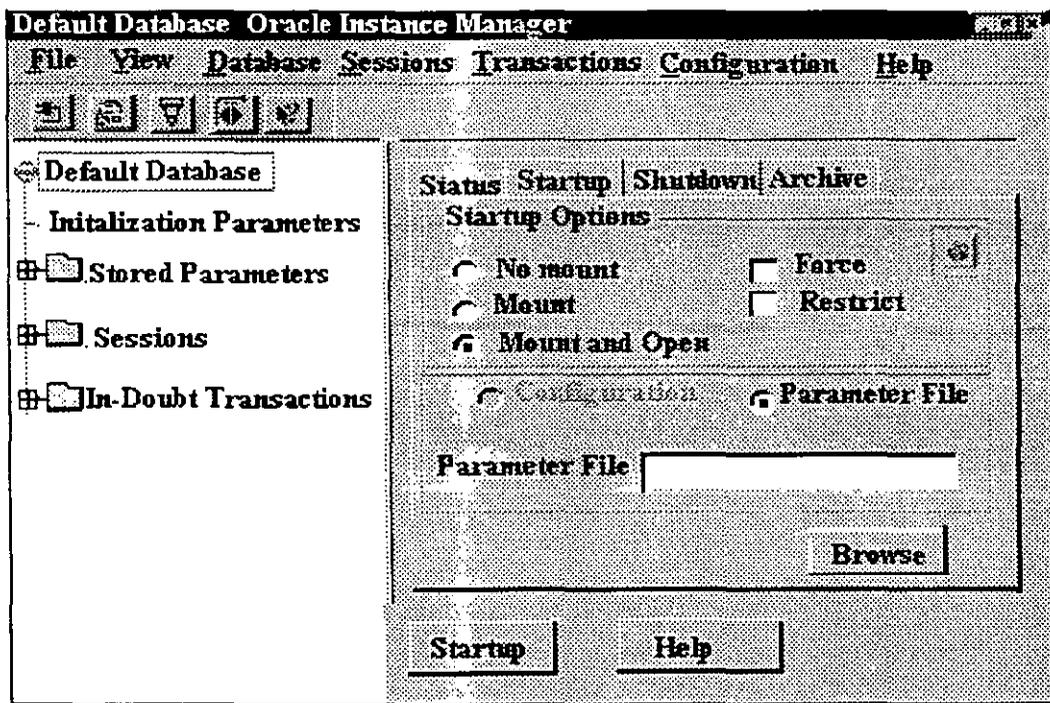


Figura 3.1.3.4 Cuadro de Diálogo de Arranque de Instance Manager.

6. Asegurarse que se encuentre seleccionada la opción Mount and Open y luego pulse el botón de Startup para iniciar la operación.
7. Pulse OK para que desaparezca el cuadro informativo Instance Started (Instancia Arrancada).
8. Salga de Instance Manager para volver a OEM.

Ahora la base de datos está abierta y disponible para las conexiones de usuario.

Desconexión del OEM.

Esta tarea es similar a la anterior excepto por la actividad en cuestión. Lleve a cabo los mismos pasos 1 a 4, pero luego haga lo siguiente:

5. Cuando se encuentre en Instance Manager, seleccione la pestaña Shutdown (desconectarse).
6. Hay tres opciones normal, immediate y about, asegúrese que la opción immediate este seleccionada en el área Shutdown y pulse OK para continuar.
7. Pulse OK para cerrar el cuadro que informa la desconexión.
8. Salga de Instance Manager para volver a OEM, pulsando sobre el cuadro de cierre de esquina.

Ahora la base de datos está cerrada e inaccesible para la comunidad habitual de usuarios.

3.2 CONFIGURACIÓN DEL WEB APPLICATION SERVER.

Como la World Wide Web evoluciona de una plataforma que atiende documentos estáticos, a una que corre distribuida, en plataforma cruzada, con aplicaciones dinámicas y protocolos para la transferencia de hipertexto (HTTP). Para lograr esto, los web servers (servidores de red) juegan un creciente rol. El Oracle Web Application Server es un servidor HTTP que habilita a sus usuarios, a crear y desplegar tales aplicaciones. Éste proporciona un framework (marco de trabajo) que abarca una arquitectura modular distribuida, una API (Interface para Programas de Aplicación) abierta que habilita al usuario a crear aplicaciones portátiles y modelos de aplicación diferentes o paradigmas.

3.2.1 Requerimientos del Sistema.

Los requerimientos mínimos del sistema necesarios para instalar y correr Web Application Server para Windows NT, se describen a continuación en las tablas 3.2.1.1 para el HW y 3.2.1.2 para el SW:

ASUNTO (HARDWARE)	REQUERIMIENTO
CPU	Un procesador Pentium o superior.
Memoria	64 MB (32 MB sólo para una máquina cliente)
Espacio en Disco Duro	315 MB de espacio en disco duro son requeridos para una instalación completa. El total es distribuido como sigue: Archivos de descarga = 62 MB. Descompresión de los archivos de descarga = 151 MB. Instalación completa del Software = 65 MB. Nota: Si se requiere liberar espacio en disco duro. Se podrán borrar los archivos de descarga (downloaded) después de su descompresión.

Tabla 3.2.1.1 Requerimientos de Hardware.

SOFTWARE	VERSIÓN
Sistema Operativo	<ul style="list-style-type: none"> • Windows NT Workstation 3.51 o 4.0. • Windows NT Server 3.51 o 4.0.
Web Browser	Web Browser que soporte tablas.

Tabla 3.2.1.2 Requerimientos de Software

3.2.2 Configuración Estándar del Web Application Server.

La siguiente configuración es común para toda instalación:

- **Language** -- El idioma por default es el Inglés.
- **Company Name** -- Una cadena de texto identifica el nombre de la compañía.
- **Oracle Home Directory** -- El directorio para el ambiente variable %ORACLE HOME%. El default es C:\ORANT.
- **Oracle Web Home Directory** -- El directorio para el ambiente variable Oracle Web Home. El default es C:\ORANT\ows\3.0.
- **Site Name** -- El nombre del **site** (sitio) para el servidor. El default es "WEBSITE30".
- **Host Name** -- Se usa para el nodo primario en una instalación multi-nodo. El default es el nombre de la máquina local como está definido en el registro.
- **Remote List** -- Esta es una lista con los nombres de los nodos remotos que formarán este **site** (sitio) y es requerida cuando se instale el nodo primario en una instalación multi-nodo.
- **UDP Service Port** -- El puerto de servicio UDP es usado para el proceso Object Request Broker (Agente de Requerimiento de Objeto) y Web Request Broker (Agente de Requerimiento de Red). El default es 2649. Se puede usar cualquier número entre 1024 y 65535.
- **Shared Key** (en hexadecimal) -- Esta llave es usada para encriptar en una configuración multi-nodo. Se requerirá introducir la misma llave en todos los nodos. Este setting se mantendrá en blanco para un nodo simple.

3.2.2.1 Configuración del Administrador Listener (Oyente).

Esta información es requerida para un nodo simple o para la instalación del nodo primario (en una instalación multi-nodo).

- **Port Number** (número de puerto) – El default es 8888.
- **User Name** (nombre del usuario) – El default es "Admin".
- **User Password** (clave del usuario) – El password que se usará para el nodo.

En la figura 3.2.2.1 Se muestra la pantalla de Windows NT donde se configura el administrador listener.

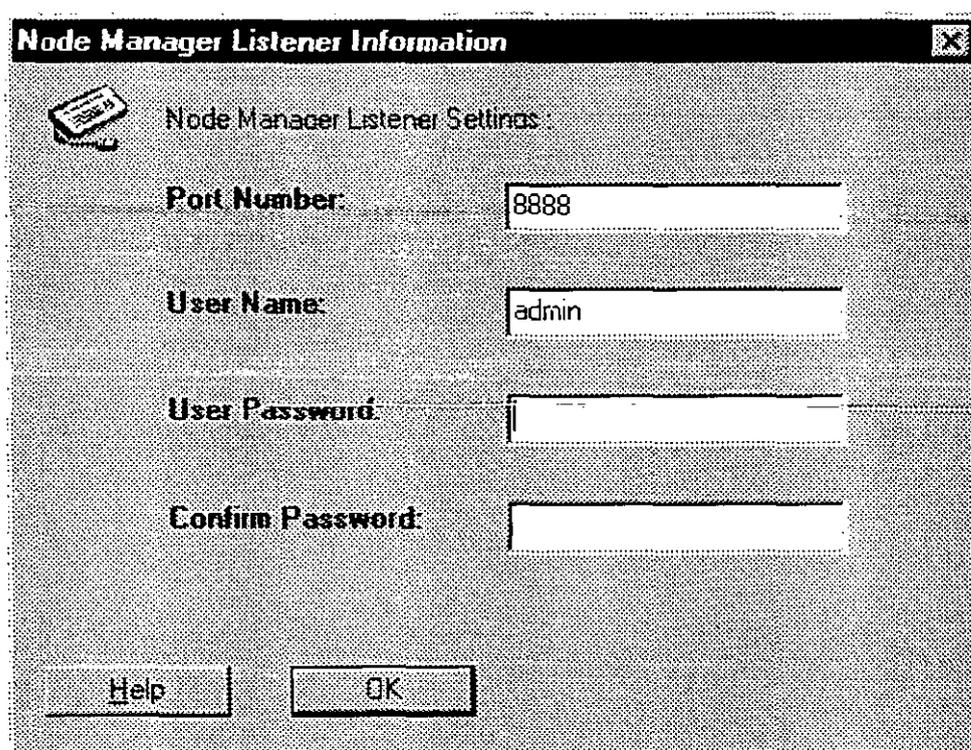


Figura 3.2.2.1 Pantalla de Configuración para el Administrador Listener.

3.2.2.2 Configuración del Web Listener.

La siguiente información es requerida para el uso general del Web Listener. Se debe

usar el Oracle listener o un **third-party** (tres capas) HTTP listener tal como FastTrack o Microsoft IIS.

- **Web Listener Name** – El default es “www” para el Oracle listener.
- **Port Number** – El default es 80.

En la figura 3.2.2.2 Se muestra el panel de windows NT para llevar a cabo esta configuración.

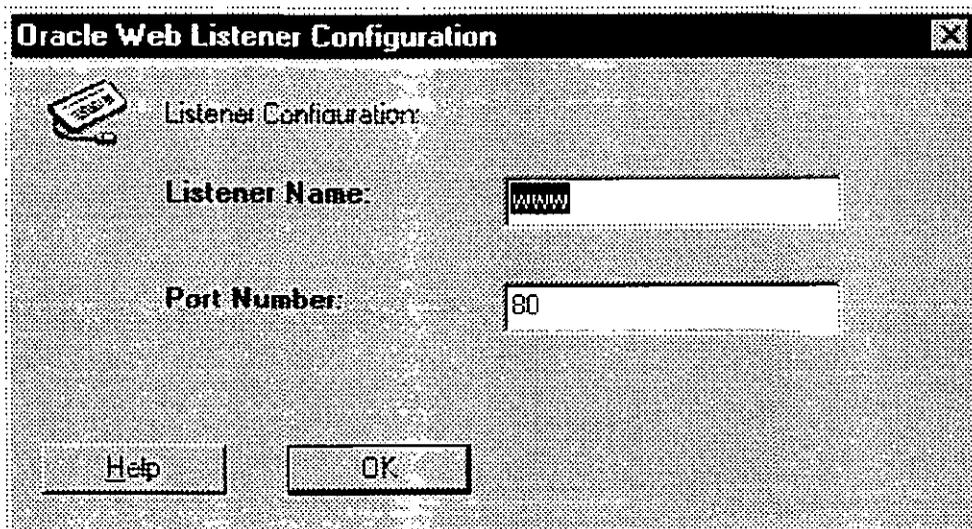


Figura 3.2.2.2 Configuración del Oracle Web Listener.

Nota: El Microsoft IIS listener no tiene un listener name . Sólo se necesita especificar el número de puerto.

3.2.3 Inicializando el Web Application Server.

Inicializar el Web Request Broker (WRB) y el Administrator Listener usando el comando `owsctl` para el prompt de DOS.

Escribir:

```
owsctl start wrb
```

Esto inicializa el Web Request Broken.

Para inicializar el Administration Listener, escribir lo siguiente en el prompt de DOS:

```
owsctl start admin
```

Listeners.

El proceso de instalación crea dos listeners usando los valores de entrada al tiempo de instalar, el Administration Listener y el Default Web Listener. Si los defaults son cambiados, los valores son mostrados debajo como una lista. Si otros valores fueron metidos, se deben anotar.

Administration Listener.

El Administration Listener es usado por las tareas de configuración y administración del Web Application Server.

- **Host Name:** hostname.domainname
- **Web Listener Name:** ADMIN
- **Port Number:** 8888
- **Username:** admin

Default Web Listener.

Un web listener de uso general es creado por la instalación. Este puede ser el Oracle listener o un third-party HTTP listener.

- **Web Listener Name:** www
- **Port Number:** 80

3.2.4 Configuración Para una Instalación Multi-Nodo.

Después de instalar los archivos en el nodo primario y en los nodos remotos, se necesita configurar el Web Application Server para ser llamado desde los nodos remotos.

Tareas a Realizar Para el Nodo Primario.

En el nodo primario se necesita:

- Levantar el WRB y los procesos listener, en ese orden.
- Configurar el cartridge que se requiere correr remotamente.

Para levantar el WRB:

```
owsctl start wrb
```

Para levantar el listener:

```
owsctl start listenerName
```

donde el listenerName especifica el nombre del listener.

Para configurar el cartridge que se va a correr remotamente, se utiliza el Web Application Server Manager de la siguiente manera:

- 1 Sobre la página del Administration home, seleccionar el Web Application Server para pasar al Administration page.
- 2 Seleccionar Cartridge Administration.
- 3 Seleccionar el cartridge que se quiere correr sobre los nodos remotos.

- 4 Sobre la página Cartridge Configuration, el campo "List of Hosts" especifica el nombre de las máquinas que pueden correr el cartridge. Se debe teclear el calificativo del nombre completo de la máquina. Por ejemplo: `hal.us.oracle.com`.

Tareas a Realizar para el Nodo Remoto.

En cada uno de los nodos remotos se necesita levantar el cartridge factory:

```
owsctl start cartridge
```

El cartridge factory maneja los cartridges que corren en la máquina. Este se comunica con el Dispatcher (que corre en el nodo primario), llamando este sobre nuevas instancias de cartridge que están dados de alta, o bien, que se mantienen desocupados. El Dispatcher entonces reconoce donde está el requerimiento directo.

3.3 DISEÑO Y CONSTRUCCIÓN DEL BACK END.

3.3.1. Diagrama de Contexto.

Una técnica común utilizada para definir el alcance de un sistema incluye el empleo del diagrama de contexto. En las técnicas que utilizan este diagrama, el sistema queda representado por un solo símbolo en medio del diagrama. A continuación se identifican los *actores* así como sus interacciones requeridas para dichos sistemas. Llegado a este punto, el diagrama muestra el contexto de un sistema determinado que está basado en los requerimientos externos. Dado este contexto, es posible entonces identificar la funcionalidad básica requerida por el sistema propuesto. En este paso final se especifica una operación para manejar cada interacción dentro del sistema. Jacobson llama a las operaciones identificadas de esta manera *casos de uso*. En la figura 3.3.1.1 se ilustra el diagrama para nuestro sistema. Aún cuando muchas metodologías orientadas a objetos emplean el diagrama de contexto, el trabajo de Jacobson es actualmente la fuente preferida. La técnica paso a paso para la representación de diagramas de contexto se podrá encontrar allí.

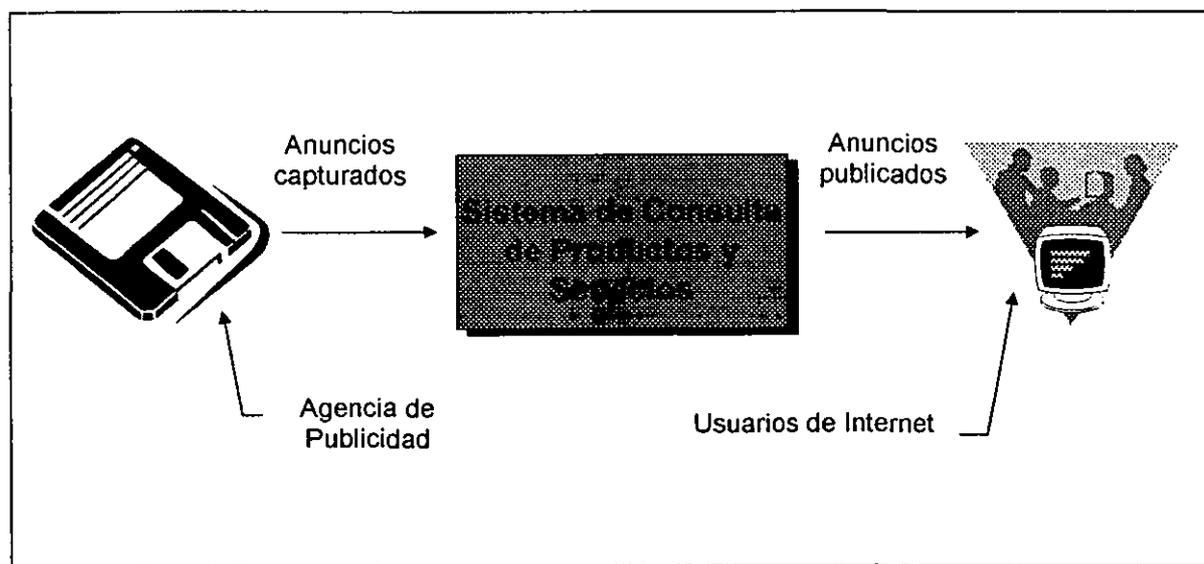


Figura 3.3.1.1 Diagrama de Contexto.

3.3.2. Diagrama de Flujo de Datos (DFD).

Los diagramas de flujo de datos que se utilizan para describir un diseño de sistemas de alto nivel, muestran cómo se transforman los datos al pasar de un componente del sistema a otro. La notación de los diagramas de flujo de datos descrita aquí se deriva del trabajo de Constantine y Yourdon (1979), Yourdon (1975) y Myers (1975). Estos diagramas documentan cómo los datos de entrada se transforman en datos de salida, donde cada etapa del diagrama representa una transformación diferente.

Componentes de un DFD.

Los diagramas de flujo de datos constan de tres componentes:

1. Procesos
2. Flujo de datos
3. Almacenamiento de datos

Procesos.

Los procesos cambian el flujo de datos de entrada en un flujo de datos de salida y se representan por medio de círculos como se muestra en la figura 3.3.2.1.

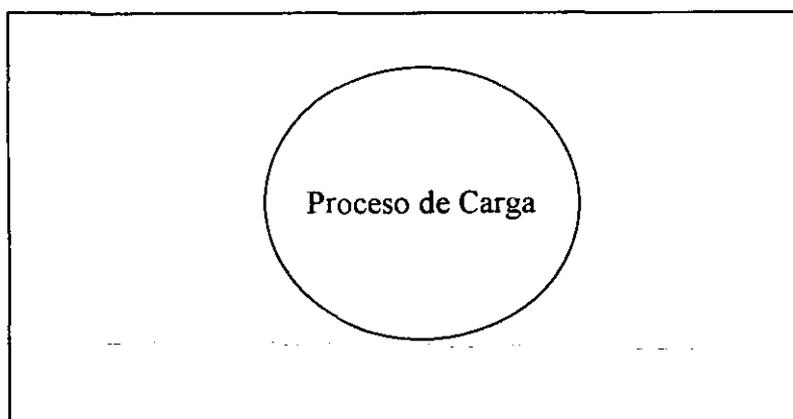


Figura 3.3.2.1 Proceso.

Un proceso representa un centro de transformación en el que la notación o nombre del proceso especifica la transformación.

Flujo de Datos.

El flujo de datos es un paquete de información en movimiento, y se representa por medio de una flecha, la cual muestra la dirección de movimiento del paquete hacia adentro y afuera de los centros de transformación (ver figura 3.3.2.2). Algunas características relevantes del mismo son:

- Un paquete puede contener datos.
- Cada flujo solo puede llevar un paquete.
- El nombre del flujo describe el significado del paquete.
- Los flujos conectados a un almacenamiento de datos, sólo pueden llevar paquetes del tipo de datos que el almacenamiento puede contener.

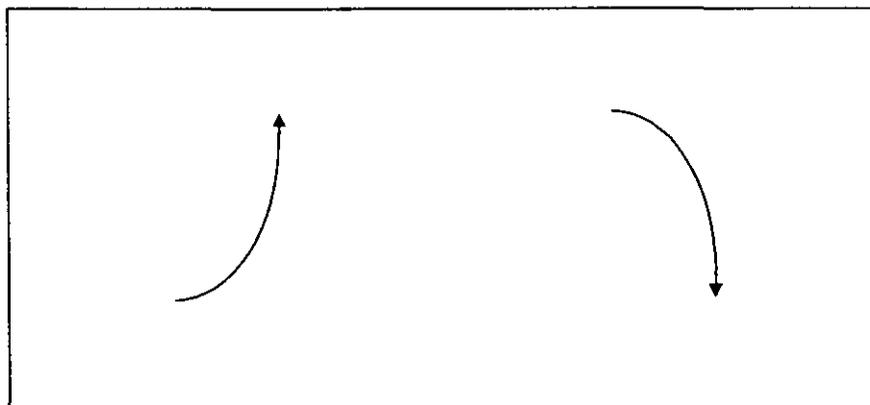


Figura 3.3.2.2 Flujo de Datos.

Flujos con Doble Dirección.

Un flujo con doble flecha, una en dirección contraria con respecto a la otra como muestra la figura 3.3.2.3, es considerado como conversacional; es decir, que uno

espera respuesta del otro, por ejemplo: teniendo dos paquetes, uno de ida y otro de regreso, se tiene el siguiente comportamiento:

- Para la pregunta de un flujo se tiene la respuesta por parte del otro.
- Para el estímulo que da un flujo se obtiene una respuesta por el flujo contrario.
- Los paquetes en ambas direcciones deben ser nombrados.

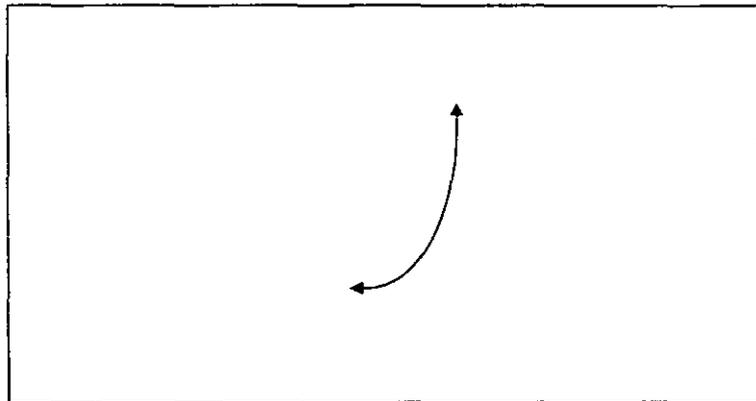


Figura 3.3.2.3 Flujo con Doble Dirección.

Almacenamiento de Datos.

El almacenamiento de datos es una colección de paquetes en reposo; un buen nombre para dicho almacenamiento, generalmente, es el plural del nombre del mismo paquete. El almacenamiento se representa por un par de líneas paralelas (ver figura 3.3.2.4).

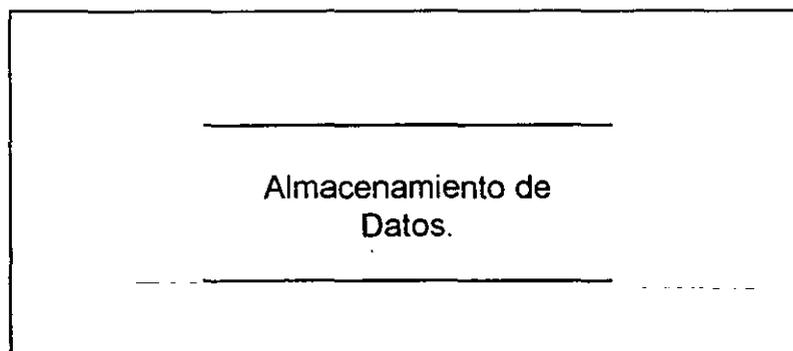


Figura 3.3.2.4 Almacenamiento de Datos.

Diagramas de Flujo de Datos.

Los diagramas de flujo de datos pueden ser usados para representar un sistema de software a cualquier nivel de abstracción. De hecho, un diagrama de flujo de datos puede ser dividido en niveles que representen a detalle el crecimiento del flujo de información y el detalle funcional de la misma.

Un nivel 0, también llamado *modelo fundamental del sistema* o *modelo de contexto*, representa los elementos del software como un simple proceso con una entrada y una salida. En la figura 3.3.2.5, se muestra el diagrama de flujo de datos nivel 0 para nuestro sistema. En el nivel 1 se puede observar más detalladamente la representación de nuestro sistema como lo muestra la figura 3.3.2.6.

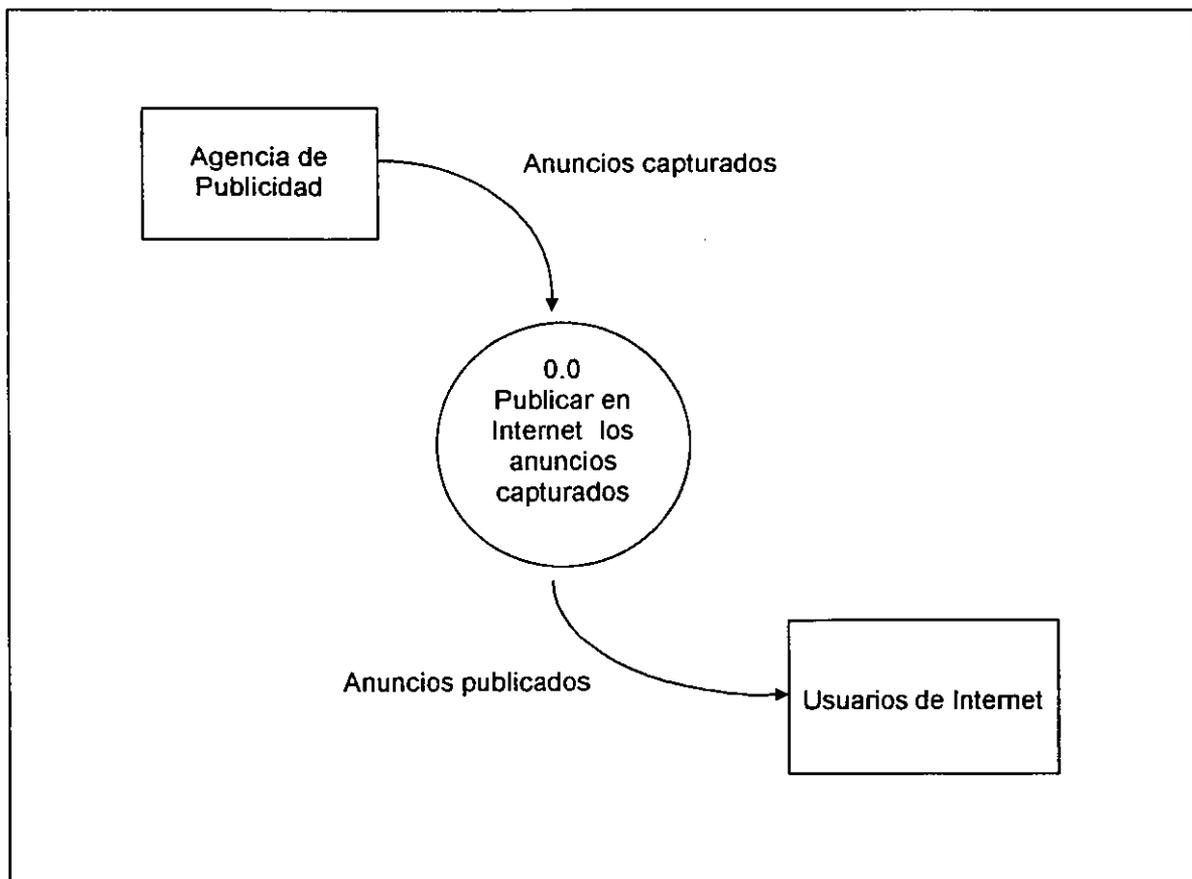


Figura 3.3.2.5 DFD Nivel 0.

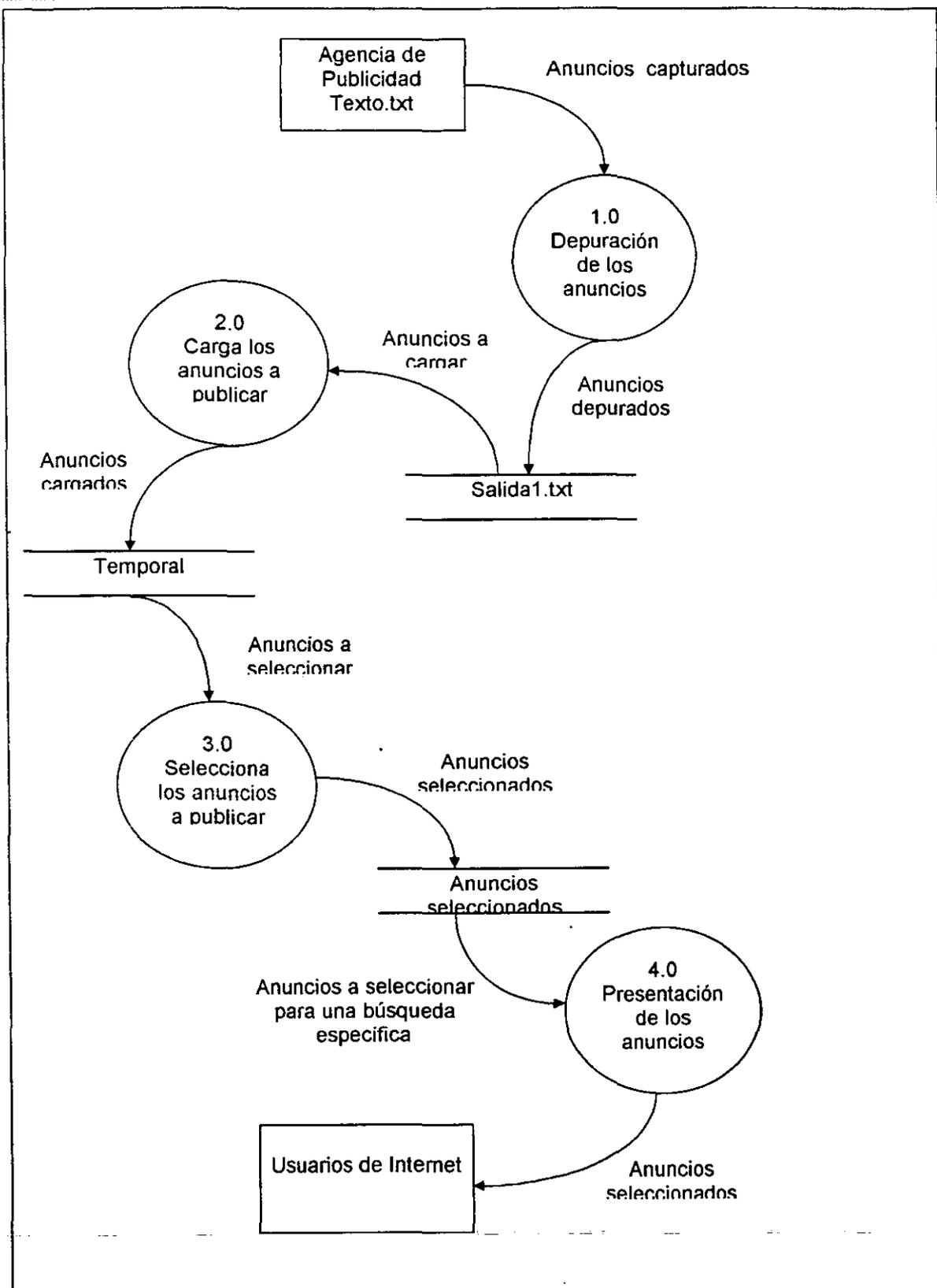


Figura 3.3.2.6 Diagrama de Flujo de Datos Nivel 1.

En el módulo de depuración se le solicita a la Agencia de Publicidad proporcione los anuncios que se desean publicar para que estos sean sujetos a un proceso de depuración. La figura 3.3.2.7 muestra el DFD Nivel 2 para este módulo, el cual se compone de 2 submódulos en los que la información se depura y se deja en el formato adecuado para poder ser cargada en la base de datos:

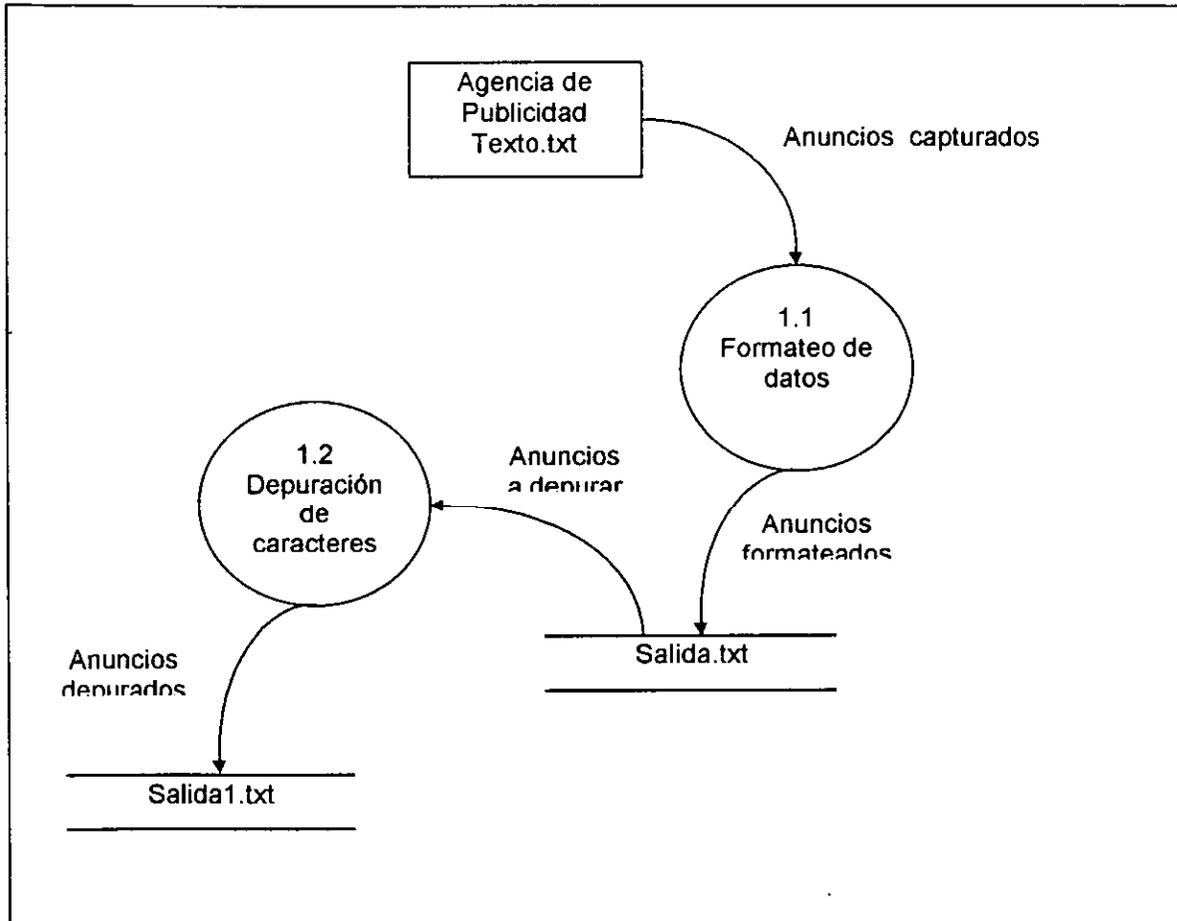


Figura 3.3.2.7 Diagrama de Flujo de Datos Nivel 2 Para el Módulo de Depuración.

- **Formateo de los datos.** Este módulo desarrollado en lenguaje C formatea la información eliminando todos los encabezados y agregando espacios donde sea necesario, para dejar exclusivamente el texto de los desplegados así como la clasificación de los mismos. Este programa obtiene como entrada un archivo *texto.txt* y obtiene como salida el archivo *salida.txt*.

El archivo de *texto.txt* contiene toda la información de los anuncios diarios a ser desplegados. Contiene encabezados y formatos que no son útiles al momento de cargar la información en la base de datos, motivo por el cual se desarrolla este módulo. El archivo *salida.txt* cuenta con todos los desplegados y su clasificación para ser desplegados en la base de datos, sin embargo contiene ciertos problemas con el set de caracteres debido a la generación propia del archivo *texto.txt*.

- Depuración de caracteres. Este módulo desarrollado en lenguaje C cambia los caracteres que están en inglés por sus correspondientes caracteres en español, como Ñ, acentos, etc. para su correcto despliegado en las páginas de Internet y correcto almacenamiento en la base de datos. Este programa obtiene como entrada el archivo *salida.txt* y obtiene como salida el archivo *salida1.txt*.

El archivo *salida1.txt* contiene la misma información que el archivo *salida.txt*, ya con los caracteres corregidos. Esta información está lista para hacer la carga a las tablas de la base de datos.

En el módulo de carga se requiere de la información ya depurada para realizar el proceso de carga (ver figura 3.3.2.8). Este proceso está compuesto de dos submódulos que realizan las siguientes actividades:

- Archivo de Control. Una vez que se tiene lista la información, es necesario generar un archivo de control *carga.cfl* en el cual se indica el nombre y el formato del archivo a cargar, el nombre de la tabla y los campos donde se realizará la carga, y finalmente el formato como va a ser cargada la información a la base de datos.
- Carga del Archivo. Una vez generado el archivo de control, se realiza la carga del archivo *Salida1.txt* con la ayuda del 'SQLloader' de Oracle. Esta utilidad subirá la información a la tabla *temporal*.

La tabla *temporal* almacena todos los desplegados así como la clasificación de los mismos. De aquí se podrán hacer consultas de carácter genérico en la aplicación, además de poder utilizar dicha tabla para distribuir la información al resto de las tablas definidas en la base de datos.

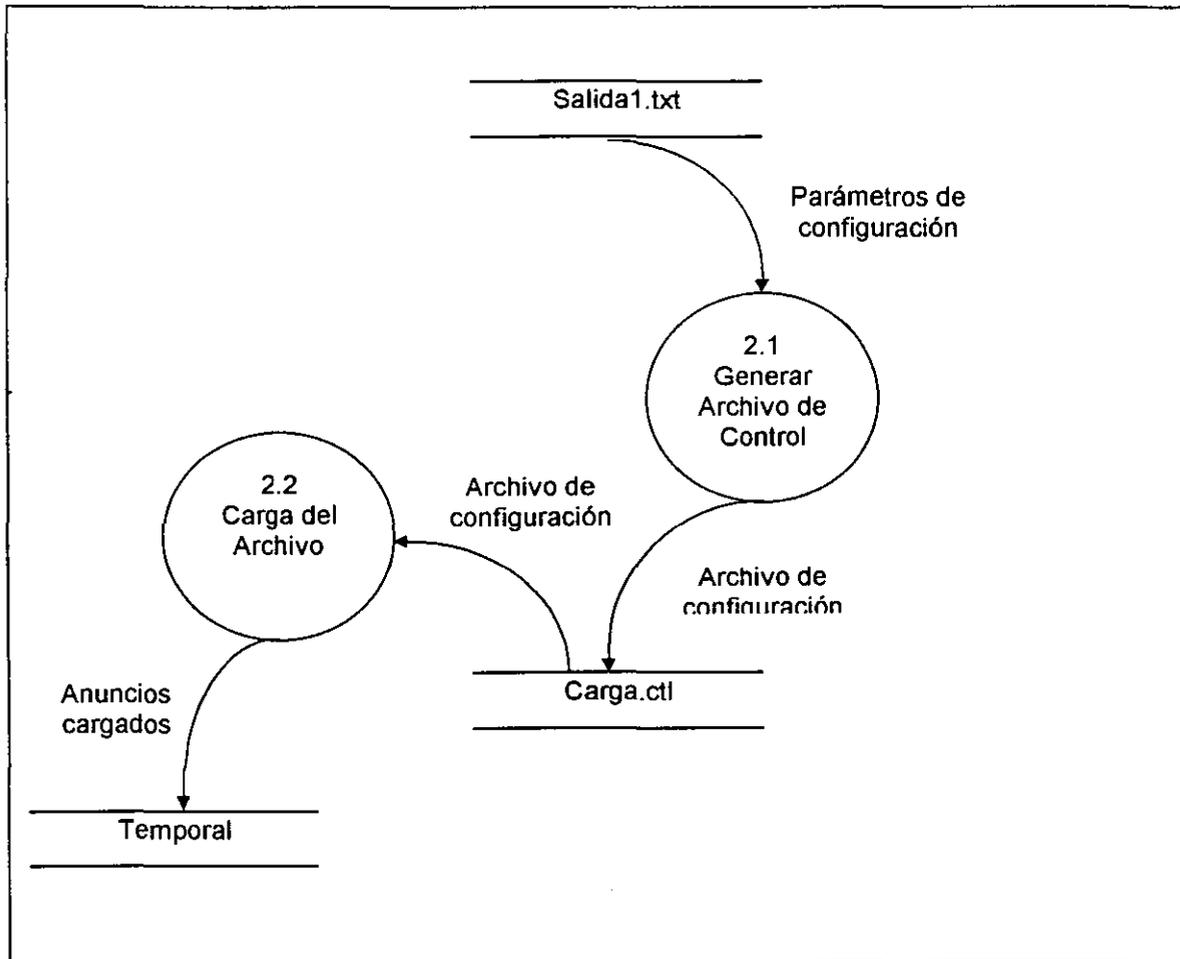


Figura 3.3.2.8 Diagrama de Flujo de Datos Nivel 2 Para el Módulo de Carga.

A continuación se presenta un ejemplo del archivo de control y la forma en la que se realiza la carga del archivo *Salida1.txt*.

Archivo de Control.

LOAD DATA

REPLACE

INTO TABLE TEMPORAL

```
(
  ID          SEQUENCE(1,1),
  TIPO        POSITION(9:10),
  SUBTIPO     POSITION(11:13),
  IMAGEN      POSITION(15:21),
  TEXTO       POSITION(30) CHAR(4500) TERMINATED BY "@"
)
```

Carga del Archivo.

```
Sqlldr80 userid=noticias/noticias control=carga.ctl data=salida1.txt
```

```
bindsize=20000000 errors=500
```

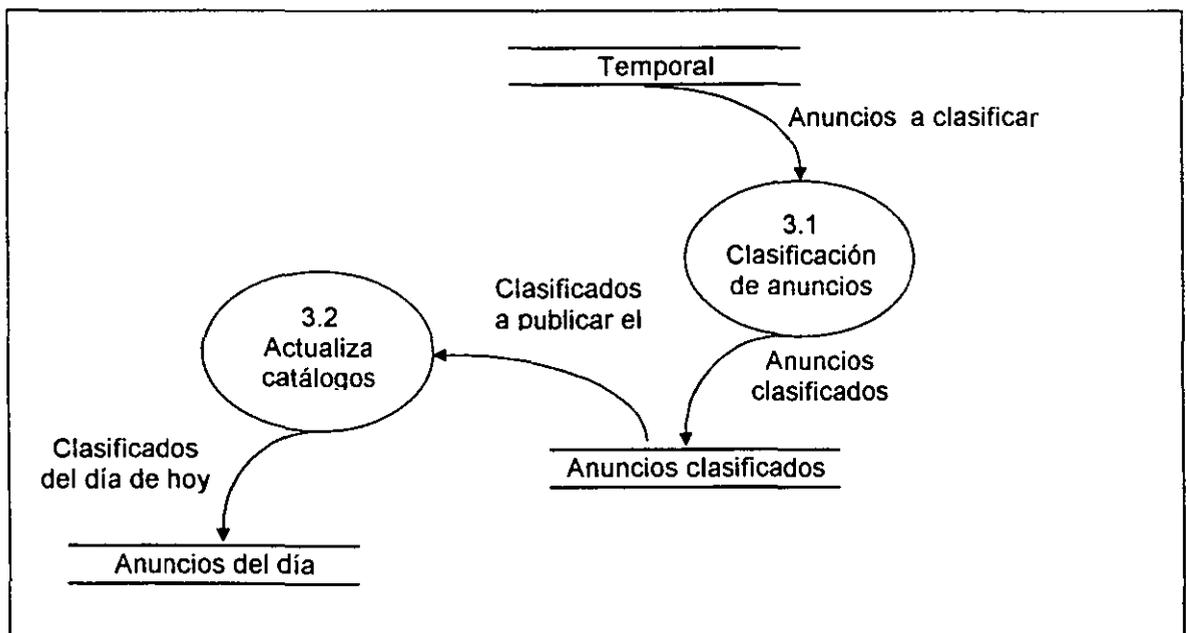


Figura 3.3.2.9 Diagrama de Flujo de Datos Nivel 2 Para el Módulo de Selección.

Como se muestra en la figura-3.3.2.9, el módulo de selección se encarga de escoger la información que ha de publicarse en Internet el día de hoy. Este módulo se compone de dos submódulos que a continuación se mencionan:

- Clasificación. La función que realiza este submódulo consiste en separar la información de la tabla temporal, en los diferentes clasificados que maneja la Agencia y almacenarlos en las tablas correspondientes según los siguientes criterios:
 - Inmuebles: 1,3,4,5,6,7,8,9,39,40,42.
 - Negocios: 10,11,12,14,15.
 - Permutas: 36,37.
 - Empleos: 16,17,18,19,20,21,22,23.
 - Cómputo: 43,47.
 - Varios: 24,25,26,33,42,44,45,46,49,50,51,52,53.
 - Vehículos: 34,38.
 - Servicios: 34,35,13,30.
- Actualización. Este proceso ejecuta un script *cata_hoy.sql* que actualiza los catálogos de tipos, subtipos y submarcas con los clasificados que cuentan con desplegados para el día de hoy; por ejemplo, si hoy la sección de permutas no tiene desplegados, esta sección no aparecerá en la aplicación, hasta el día que tenga desplegados.

Ambos procesos serán ejecutados por el Oracle Enterprise Manager.

Nota: Los números descritos anteriormente corresponden a las secciones establecidas por la Agencia.

En etapa del desarrollo de la aplicación correspondiente al módulo de presentación descrito en la figura 3.3.2.10, se lleva a cabo la creación de **packages** (paquetes) y procedimientos donde se definen los tipos de consultas y donde se define parte del diseño de páginas de Web que el usuario ha de utilizar.

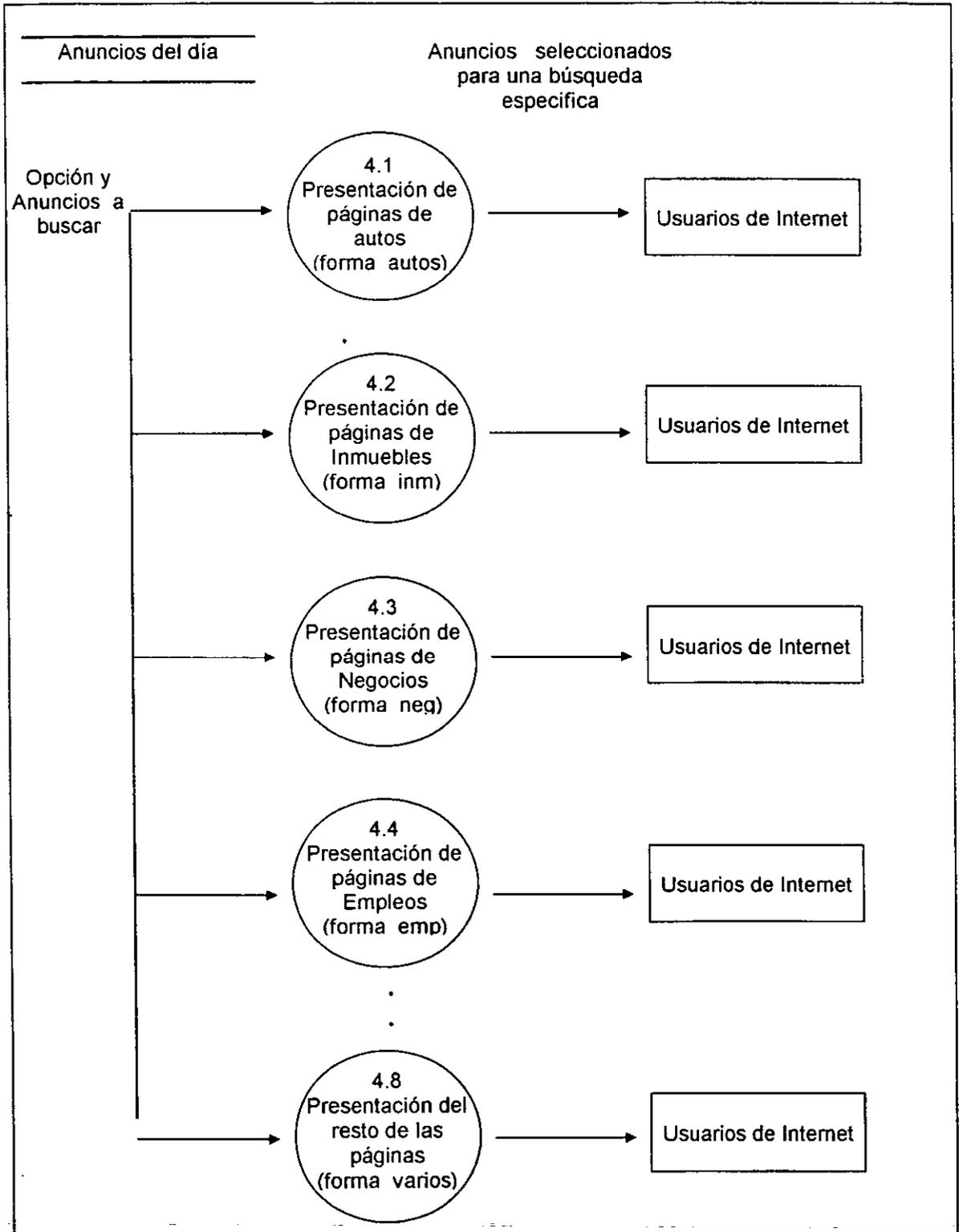


Figura 3.3.2.10 Diagrama de Flujo de Datos Nivel 2 Para el Módulo de Presentación.

Para el despliegue de las 24 páginas dinámicas que satisfacen las necesidades de la aplicación se crearon 8 packages. Siete de ellos poseen la misma estructura lógica y el package de autos que se diferencia con respecto a los otros, debido a que en este package se incluye una subclasificación extra (modelos de autos). Por tal motivo, solo se explicarán el package de autos y un package de varios que encierra el resto de los packages del sistema. (4.1 Forma_autos y 4.8 Forma_varios).

Descripción del Package 'autos'.

El procedimiento inicial del package 'autos' se llama **forma_autos**, el cual se encarga de desplegar la pantalla principal, donde el usuario puede introducir una consulta por palabra específica o ir a una segunda pantalla donde se encuentra una subclasificación. Este procedimiento almacena dos opciones del usuario y las almacena en dos variables 'opcion' (selección de subclasificación) y 'variable' (son las palabras por el cual el usuario realiza búsquedas en las tablas). En la figura 3.3.2.11 se muestra el DFD nivel 3 para el módulo de presentación de Autos.

El procedimiento **vali_autos** se encarga de verificar cuál de las dos opciones insertó el usuario, para mandar a ejecutar alguno de los dos procedimientos siguientes: 'busca_autos' ó 'subforma_autos'.

El procedimiento **busca_autos** ejecuta el cartucho Contex para poder realizar búsquedas con las palabras que haya introducido el usuario.

El procedimiento **subforma_autos** ejecuta una segunda pantalla de captura de opciones del usuario, la cual muestra las subclasificaciones de la tabla de autos.

El procedimiento **subvali_autos** recibe las opciones que el usuario ha insertado en esta segunda forma de captura ('p_subvariable' y 'p_subopcion'), y una tercer variable que ya introdujo desde la primer forma de captura ('p_opcion').

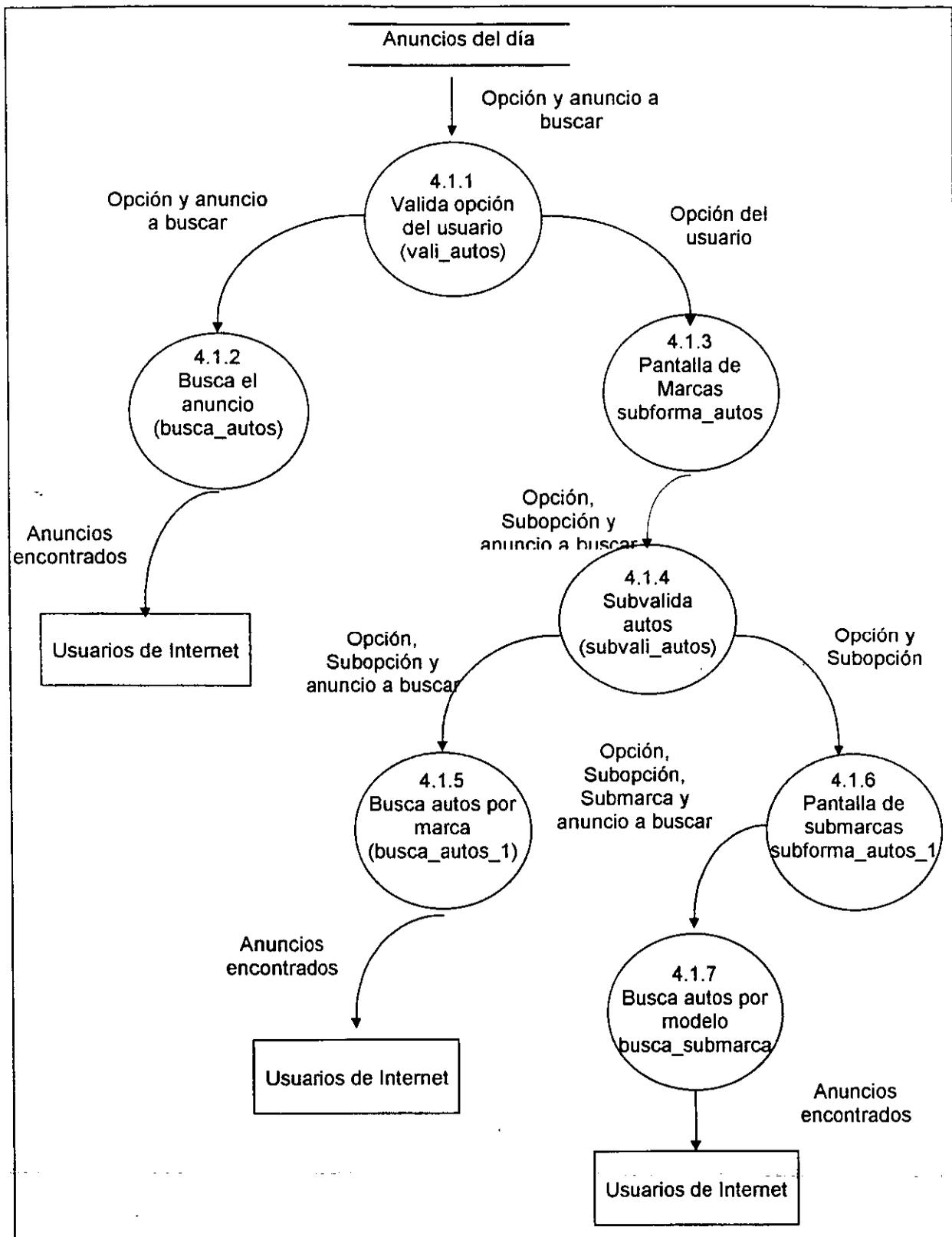


Figura 3.3.2.11 DFD Nivel 3 Para el Módulo de Presentación de Autos.

El procedimiento **busca_autos_1** realiza búsquedas en Memoria con el cartucho de Context, si el usuario introdujo una serie de palabras por las cuales necesitaba buscar. Se emplean las tres variables que se utilizaron en el anterior procedimiento ('p_subvariable', 'p_subopcion' y 'p_opcion').

El procedimiento **subforma_autos_1** ejecuta una tercer pantalla de captura de opciones del usuario, la cual muestra las submarcas de la subclasificación anteriormente seleccionada por el usuario.

El procedimiento **busca_submarca** realiza búsquedas en Memoria con el cartucho Context, si el usuario introdujo una serie de palabras por las cuales necesitaba buscar. Se emplean las cuatro variables que se han utilizado en los procedimientos anteriores ('p_tervariable', 'p_teropcion', 'p_opcion' y 'v_id_subtipo').

Descripción del Package 'varios'.

El procedimiento inicial del package 'varios' se llama **forma_varios**, el cual pide las opciones que determinó el usuario y las almacena en dos variables 'opcion' y 'variable' que permitirán realizar la búsqueda necesaria. En la figura 3.3.2.12 se presenta el DFD nivel 3 para el módulo de presentación de Varios.

El procedimiento **vali_varios** se encarga de verificar cuál de las dos opciones insertó el usuario, para mandar a ejecutar alguno de los dos procedimientos siguientes: 'busca_varios' ó 'subforma_varios'.

El procedimiento **busca_varios** ejecuta el cartucho Context para poder realizar búsquedas en Memoria con las palabras y clasificación que haya introducido el usuario.

El procedimiento **subforma_varios** ejecuta una segunda pantalla de captura de opciones del usuario, la cual muestra las subclasificaciones de las tablas 'varios'

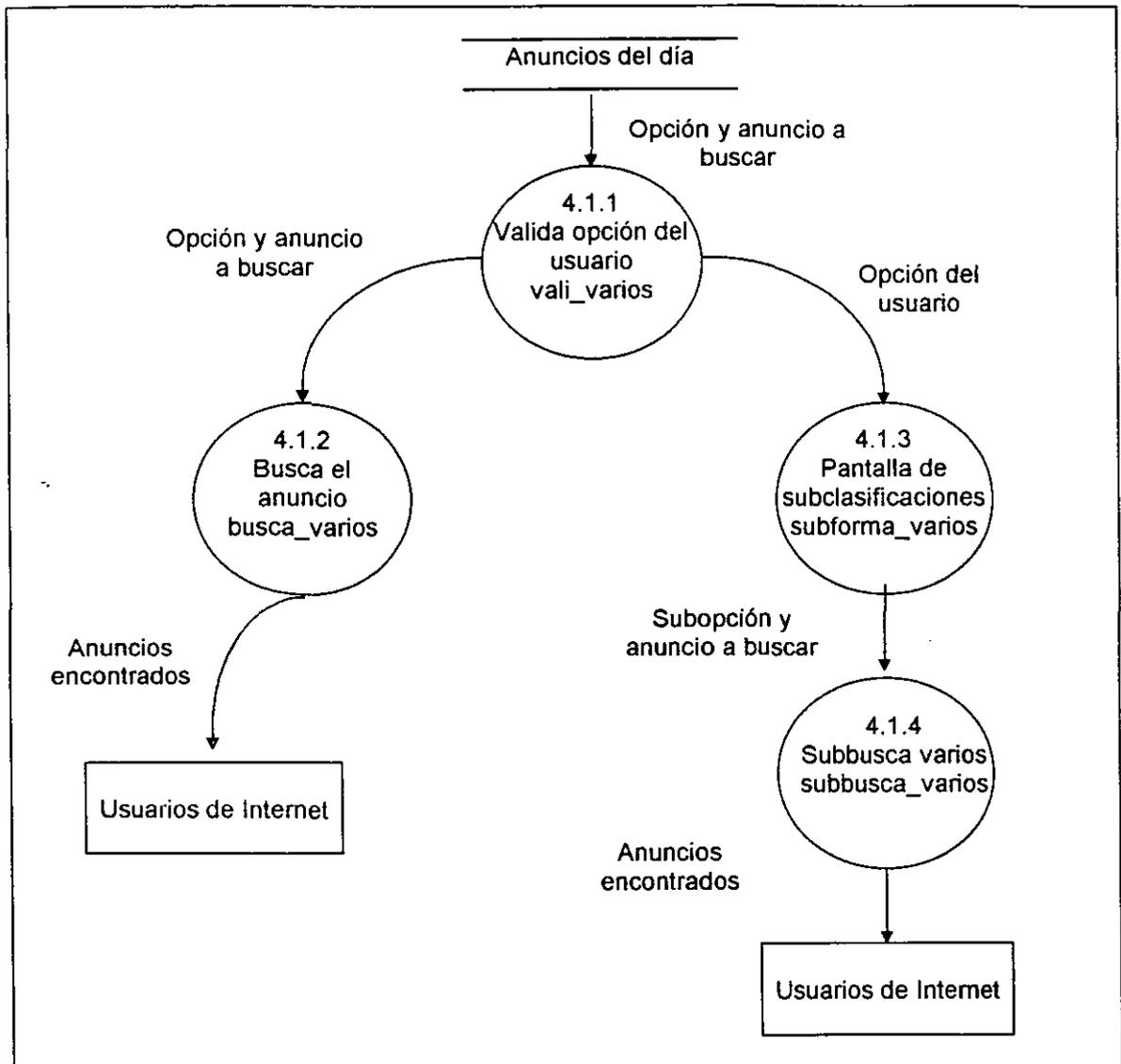


Figura 3.3.2.12 DFD Nivel 3 Para el Módulo de Presentación de Varios.

El procedimiento **subbusca_varios** realiza búsquedas en Memoria con el cartucho Context, si el usuario introdujo una serie de palabras por las cuales necesitaba buscar. Se emplean las dos variables que se utilizaron en el anterior procedimiento ('p_subvariable' y 'p_subopcion').

3.3.3. Diagrama de Entidad-Relación.

Para obtener un prototipo de la base de datos que se implantará, se hará uso de una técnica de modelado llamada Entidad-Relación.

Un modelo es una abstracción de un hecho real o representación de una situación. El modelo de datos relacional es una colección finita de tablas (de dos dimensiones; formadas de columnas y renglones) que representan una situación.

Características del modelo relacional:

1. **Simplicidad:** Las tablas son una forma familiar y explicables por sí mismas para representar datos. La mayoría de la gente ha utilizado datos en forma de tabla, no se requiere de un entrenamiento especial para entender o utilizar los datos que se representan en las tablas. En pocas palabras podemos decir que son amigables al usuario.
2. **Precisión.** Las tablas correctamente diseñadas mantienen un rigor matemático, dicen lo que significan y significan lo que dicen. Pueden ser implantadas y procesadas por una gran variedad de configuraciones de *software* y *hardware*.
3. **Flexibilidad.** Las tablas no solamente muestran la estructura de los datos, sino pueden mostrar los datos también. Esto nos permite manejar el modelo antes de implementarlo. En otras palabras, las tablas son apropiadas no sólo para modelar datos sino para procesarlos también.

Para hacer uso de esta técnica de modelado es necesario definir algunos conceptos necesarios.

Situación.

Conjunto de circunstancias bien definidas que pueden ser descritas utilizando un lenguaje natural suficientemente complejo. Un lenguaje natural suficientemente complejo incluye al menos los siguientes 3 constructores gramaticales:

- *Sujeto*. Un sujeto es el nombre de una persona, de un tipo de persona, de un animal, una planta, un lugar, una cosa, una substancia o una idea. Un sujeto propio es el nombre de una ocurrencia particular o instancia de un sujeto. Un pronombre es una palabra utilizada como sustituto de un sujeto que hace referencia a un sujeto que ha sido nombrado y que se sobre entiende de acuerdo al contexto en el cual se utiliza. Los sujetos pueden ser tangibles o intangibles y es imposible describir alguna situación sin al menos el uso de un sujeto.
- *Verbos*. Un verbo es una palabra que describe un modo de ser una asociación, una acción o un evento. Los verbos describen el estado de los sujetos y relacionan a los sujetos dentro de las situaciones. Los verbos pueden ser activos o pasivos:

Pasivo: El alumno es enseñado por el profesor.

Activo: El profesor enseña al alumno.

- *Modificadores*. Un modificador es una palabra que califica a un sujeto o a un verbo de acuerdo a sus características, cantidad, extensión, etc. Los modificadores de los sujetos se llaman adjetivos y los modificadores de los verbos se llaman adverbios.

En la figura 3.3.3.1 podemos identificar los conceptos de un modelo Entidad-Relación.

El modelo Entidad-Relación nos dice que podemos dividir el análisis en 3 fases.

1. Modelo de Entidades.

2. Modelo de Relaciones.
3. Modelos de Atributos.

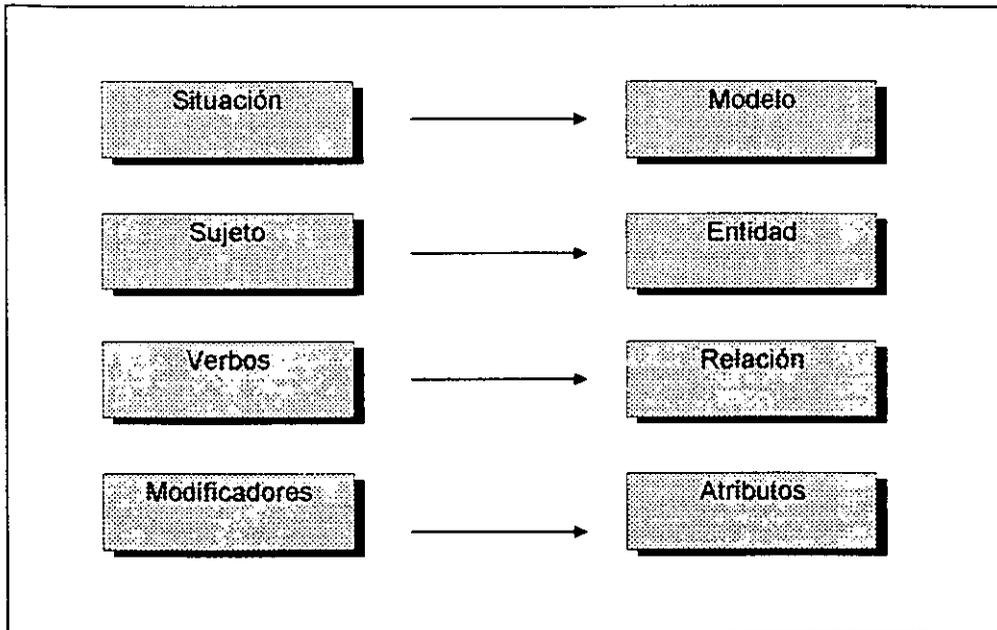


Figura 3.3.3.1 Conceptos de un Modelo Entidad/Relación.

Modelado de Entidades.

El modelado de entidades se basa en los siguientes pasos:

- 1) Descubrir entidades. Concentrarse en sujetos, las entidades existen por sí mismas.
- 2) Definir el alcance de la entidad. Definir si esa entidad es de interés al sistema.
- 3) Definir una llave primaria. Una llave primaria es un atributo de la entidad, que permitirá identificar cada renglón dentro de la tabla que formará, de manera única; sus características son:
 - ND: No duplicada, garantiza la existencia única de ese elemento.
 - NN: No nula, garantiza que este dato siempre existe.
 - NC: No cambia, no permite cambiar este dato.
- 4) Documentar.

5) Incluirla en le diagrama Entidad-Relación.

Las entidades de un diagrama Entidad-Relación se representan como rectángulos. En el caso de nuestro sistema podemos identificar las siguientes entidades, las cuales se representan en la figura 3.3.3.2:

Temporal: Esta entidad es donde se almacena toda la información de los desplegados. De esta tabla se parte para llenar las demás tablas. Sobre esta entidad se hacen consultas genéricas sobre la información que se tiene de los desplegados.

Inmuebles, negocios, permutas, empleos, cómputo, varios, vehículos y servicios: En estas entidades es donde se tiene toda la información clasificada de acuerdo a las secciones que presenta la Agencia de Publicidad.

Cata_tipo, cata_subtipo, cata_submarca: Son las entidades donde se almacenan los catálogos permanentes de las clasificaciones que se tienen en la Agencia.

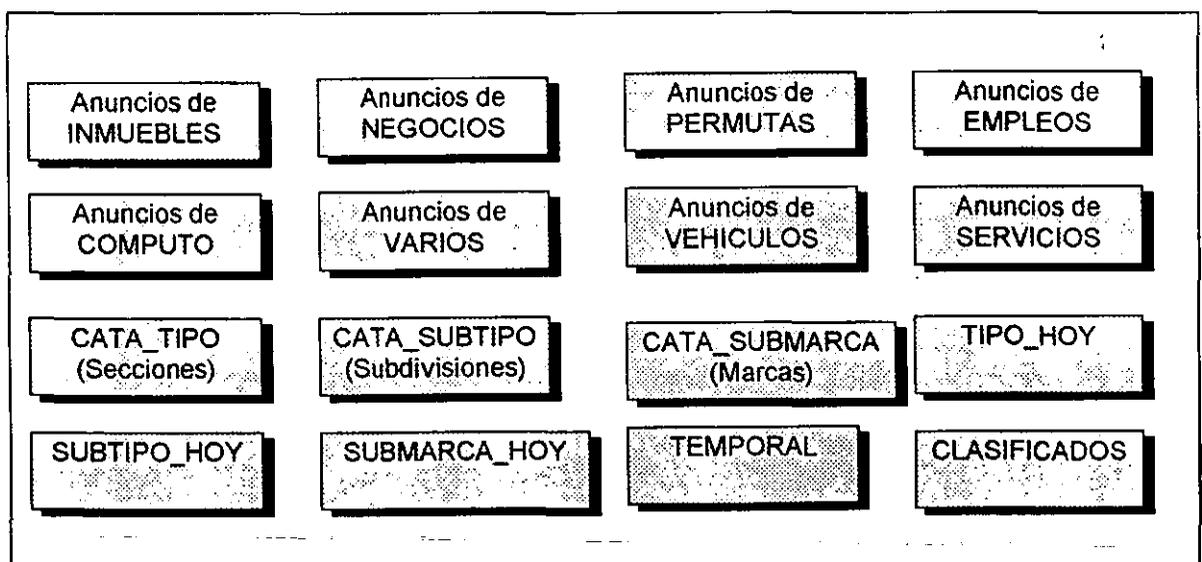


Figura 3.3.3.2 Entidades del Sistema.

Tipo_hoy, subtipo_hoy, submarca_hoy: Son las entidades de catálogos diarios donde se almacenan las secciones que el día de hoy cuentan con desplegados.

Las entidades identificadas pasarán a ser las tablas de la base de datos y para cada una de ellas se definirá un atributo que fungirá como llave primaria.

En la tabla 3.3.3.3 se muestran la llave primaria para cada entidad definida.

ENTIDAD	LLAVE PRIMARIA
INMUEBLES	ID_INMUEBLES
NEGOCIOS	ID_NEGOCIOS
PERMUTAS	ID_PERMUTAS
EMPLEOS	ID_EMPLEOS
COMPUTO	ID_COMPUTO
VARIOS	ID_VARIOS
VEHICULOS	ID_VEHICULOS
SERVICIOS	ID_SERVICIOS
CATA_TIPO	ID_TIPO
CATA_SUBTIPO	ID_TIPO
CATA_SUBMARCA	ID_SUBMARCA
TIPO_HOY	ID_TIPO
SUBTIPO_HOY	ID_SUBTIPO
SUBMARCA_HOY	ID_SUBMARCA
TEMPORAL	ID, TIPO, SUBTIPO
CLASIFICADOS	ID

Tabla 3.3.3.3 Llaves Primarias.

Cabe mencionar que después de realizar el modelado de relaciones, las llaves primarias pueden ser modificadas.

Modelado de Relaciones.

Una vez identificadas las entidades, las relaciones las obtenemos a partir de los siguientes pasos:

- 1) Descubrir relaciones, concentrarse en verbos.
- 2) Definir el alcance de la relación, definir si la relación es importante para el sistema.
- 3) Definir el tipo de relación.
- 4) Documentar en el diagrama Entidad-Relación.
- 5) Documentar en tablas.

En las figuras 3.3.3.4 y 3.3.3.5 se muestran los diagramas de entidades con las relaciones que se encontraron para nuestro sistema:

Las relaciones de acuerdo al número de elementos que se involucran de ambos lados de las entidades que participan (cardinalidad) se clasifican en 3.

Relación 1 a 1.

Relación 1 a M.

Relación M a M.

Donde M se refiere a muchos.

Para modelar las relaciones de acuerdo a su tipo se debe emplear la llave primaria de las entidades involucradas, esta llave al ser colocada en otra entidad pasa a llamarse foránea.

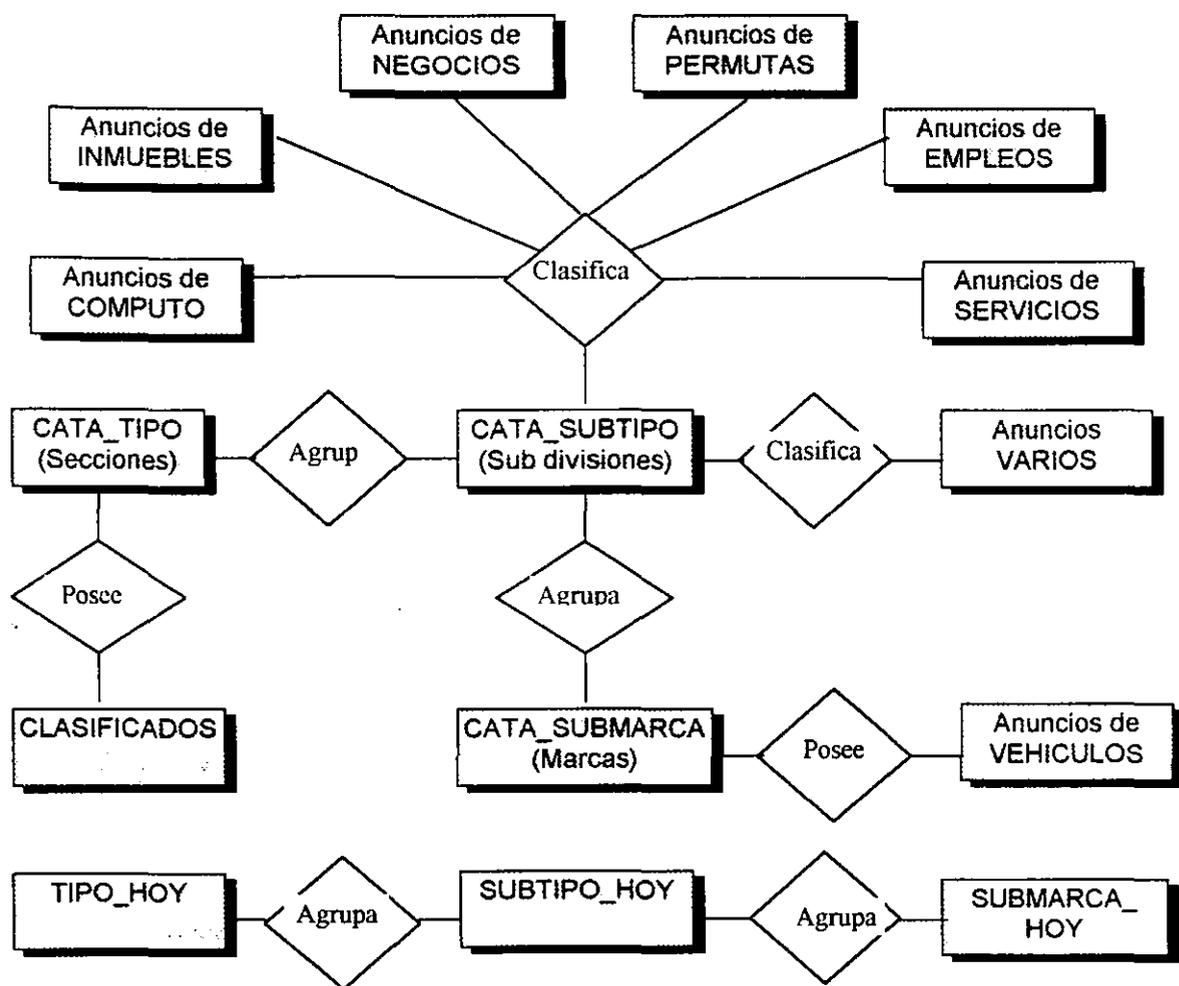


Figura 3.3.3.4. Diagrama de Relaciones 1.

Relación 1 a 1: La llave primaria de cualquiera de las 2 entidades involucradas debe migrar hacia la otra entidad, es indistinto cuál de las entidades migra su llave primaria, sin embargo de acuerdo a la utilización de los datos esto puede cambiar.

Relación 1 a M: La llave primaria de la entidad donde sólo se involucra 1 elemento se migra hacia la entidad donde se involucran muchos elementos (M).

Relación M a M: Se debe generar una nueva entidad cuya llave primaria es la combinación de las llaves primarias de las entidades que la originaron.

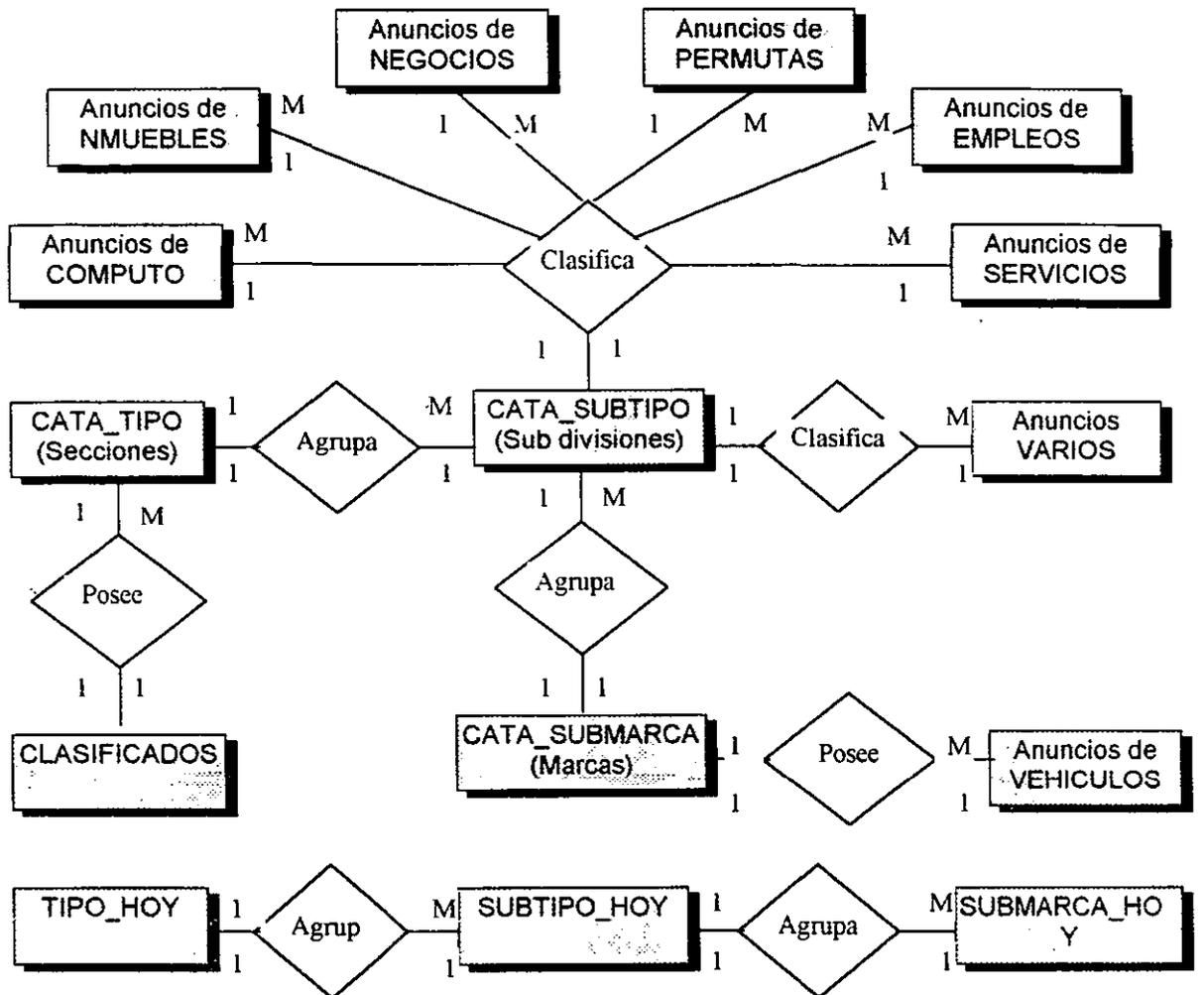
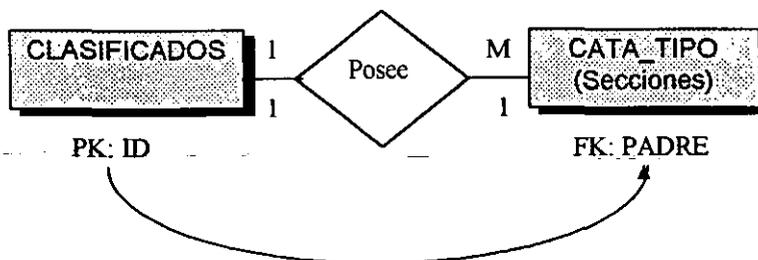
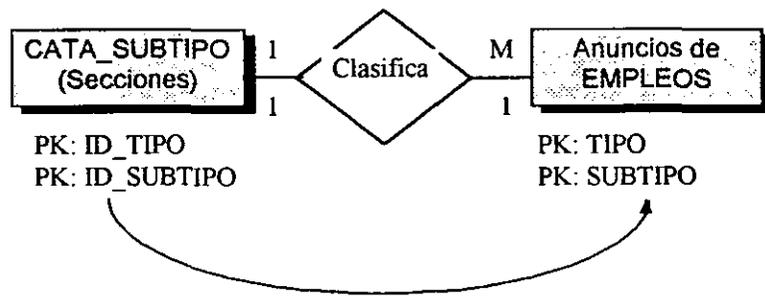
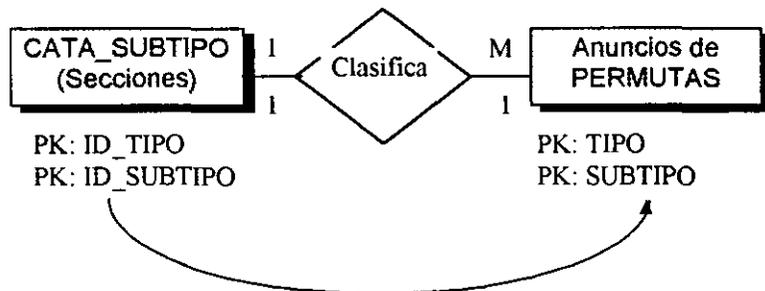
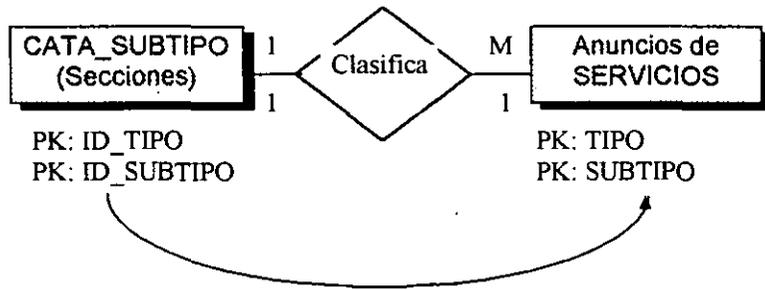
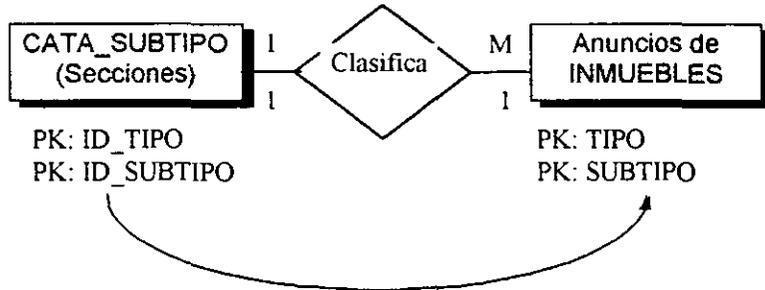
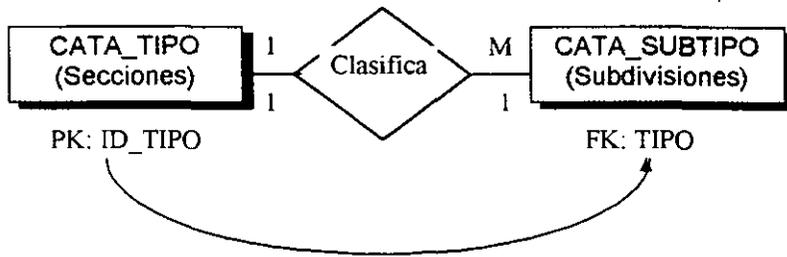
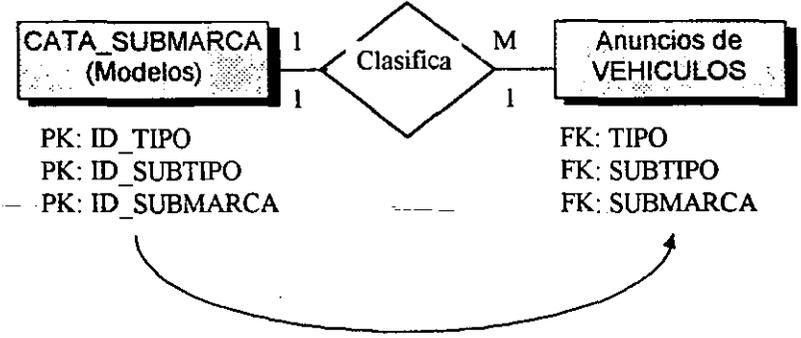
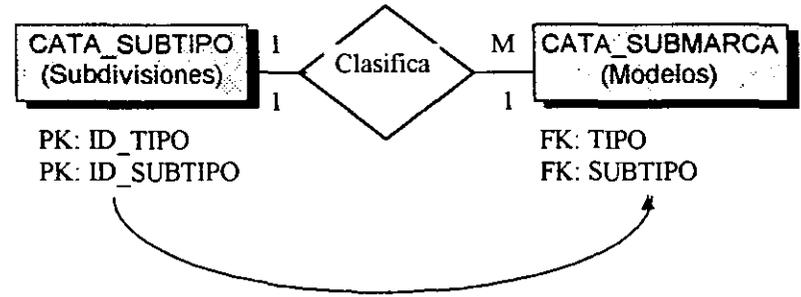
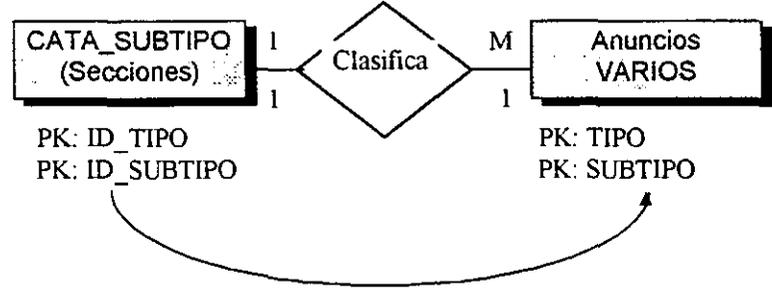
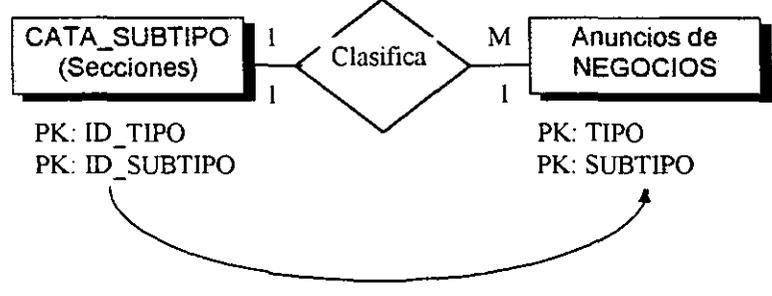
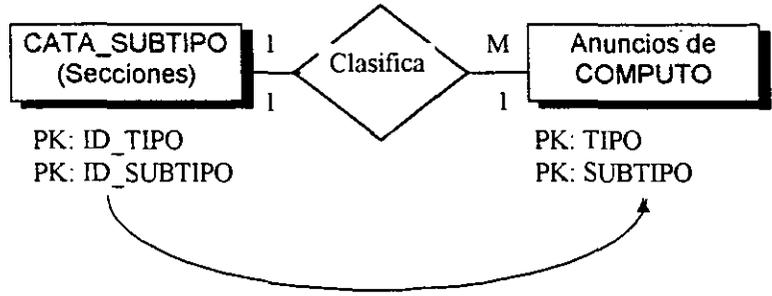


Figura 3.3.3.5. Diagrama de Relaciones 2.

En la figura 3.3.3.6. se muestran los diagramas con los cambios que sufre nuestro modelo de acuerdo a las definiciones antes mencionadas:







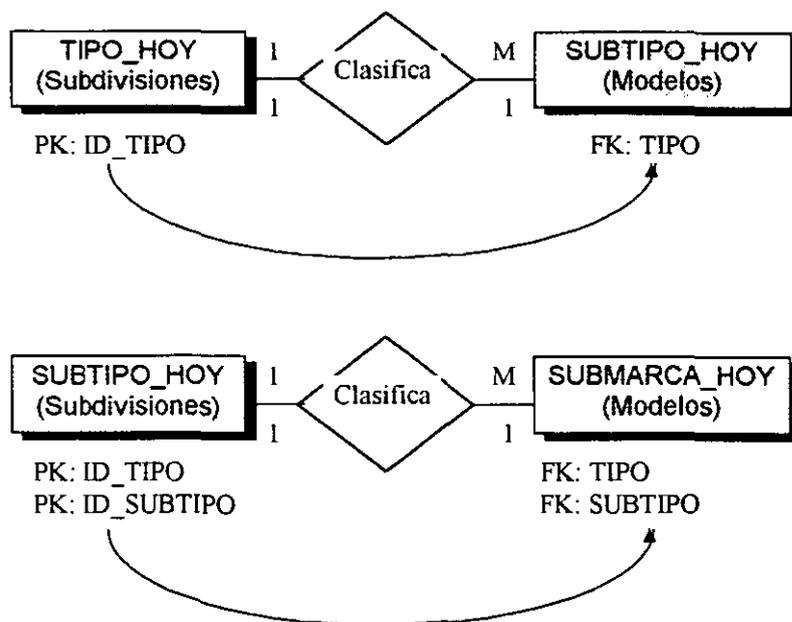


Figura 3.3.3.6. Diagramas de Migración de PK.

Modelado de Atributos.

- 1) Descubrir atributos, concentrarse en adjetivos y adverbios.
- 2) Definir el alcance del atributo.
- 3) Documentar el atributo en la entidad; para colocar los atributos de manera correcta en una entidad se utiliza el método de normalización que permite verificar si nuestro modelo es correcto o bien si funcionará al ser implementado.

Una vez encontrados los atributos para cada entidad se procede a normalizar.

En la tabla 3.3.3.7. se muestran los atributos de las entidades de nuestro sistema:

Entidad (s)	Atributo	Descripción
CLASIFICADOS	ID	Identificador del clasificado 91 = Negocios 92 = Permutas 93 = Servicios 94 = Empleos 95 = Bienes Inmuebles 96 = Equipo fotográfico y de Computo 97 = Varios 98 = Vehículos
	TEMPORAL	
	ID	Identificador
	TIPO	Tipo de anuncio
	SUBTIPO	Subtipo de anuncio
	SUBMARCA	Submarca del anuncio
	IMAGEN	Imagen del objeto que se promociona
	TEXTO	Texto del anuncio
	CATA_TIPO, TIPO_HOY	ID_TIPO
CATA_SUBTIPO, SUBTIPO_HOY	PADRE (FK)	Identificador del clasificado que se anuncia
	TITULO	Titulo de clasificado
	CLASI	Indica si el tipo de anuncio posee mas subclasificaciones 1 = No tiene 2 = Si tiene
	TIPO	Identificador del tipo
	ID_SUBTIPO	Identificador del subtipo

CATA_SUBMARCA, SUBMARCA_HOY	TITULO	Titulo de clasificado
	CLASI	Indica si el tipo de anuncio posee mas subclasificaciones 1 = No tiene 2 = Si tiene
	TIPO	Identificador del tipo
	SUBTIPO	Identificador del subtipo
INMUEBLES, NEGOCIOS, PERMUTAS, EMPLEOS, COMPUTO, VARIOS, SERVICIOS	ID_SUBMARCA	Identificador de la submarca
	TITULO	Titulo de clasificado
	ID	Identificador
	TIPO (FK)	Tipo de anuncio
VEHICULOS	SUBTIPO (FK)	Subtipo de anuncio
	IMAGEN	Imagen del objeto que se promociona
	TEXTO	Texto del anuncio
	FECHA	Fecha de publicación
	ID	Identificador
	TIPO (FK)	Tipo de anuncio
	SUBTIPO (FK)	Subtipo de anuncio
	SUBMARCA (FK)	Submarca del vehículo
	IMAGEN	Imagen del objeto que se promociona
	TEXTO	Texto del anuncio
FECHA	Fecha de publicación	

Tabla 3.3.3.7 Relación de Atributos.

3.3.4. Normalización.

La normalización permite verificar si nuestro modelo es correcto o bien si funcionará al ser implementado.

Primera Forma Normal.

Dada una tabla T y con una llave primaria P y un atributo A , se dice que T está en primera forma normal sí y sólo sí, el valor del atributo A en cualquier renglón depende del valor de la llave primaria P en ese renglón. En otras palabras, no se permiten valores constantes dentro de las tablas, bajo una misma columna.

Segunda Forma Normal.

Dada una tabla T en primera forma normal, con una llave primaria P en varias columnas, con componentes $P1$ y $P2$ y un atributo A , se dice que T está en segunda forma normal, sí y sólo sí, el valor del atributo A en cualquier renglón depende de dos valores $P1$ y $P2$ en ese renglón. Las tablas con llaves primarias en una columna, siempre están en segunda forma normal. En este caso se habla de la dependencia de cualquier atributo con respecto a las columnas que fungen como llave primaria compuesta, tampoco se permiten valores constantes.

Tercera Forma Normal.

Dada una tabla T en segunda forma normal con una llave primaria P y dos atributos $A1$ y $A2$, se dice que T está en tercera forma normal sí y sólo sí, el valor del atributo $A1$ en cualquier renglón no depende del atributo $A2$, a menos que $A2$ esté marcado como no duplicado (ND) y el valor del atributo $A2$ en cualquier renglón no depende $A1$, a menos que $A1$ este marcado como ND.

Si existe dependencia entre atributos regularmente se debe crear una nueva tabla con los atributos que dependen entre sí y se establece una relación con la tabla original. Revisando cada uno de los atributos para las diversas tablas ya identificadas no se encontró ninguna incongruencia por lo que podemos pasar a la definición de cada una de ellas en Oracle.

El modelo Entidad-Relación Normalizado puede ser generado mediante el uso de herramientas CASE (Computer Added Software Engineer), que son programas especializados en la creación de modelos y la normalización de estos, así como la generación de los diccionarios de datos que siempre resultan ser muy tediosas para los diseñadores.

En nuestro caso la herramienta CASE que se utilizó para elaborar el diagrama Entidad-Relación fue Erwin. A continuación se muestran los pasos para realizar un diagrama Entidad-Relación con Erwin.

1. Crear las entidades necesarias con ayuda del 'Erwin Toolbox' como se muestra en la figura 3.3.4.1.

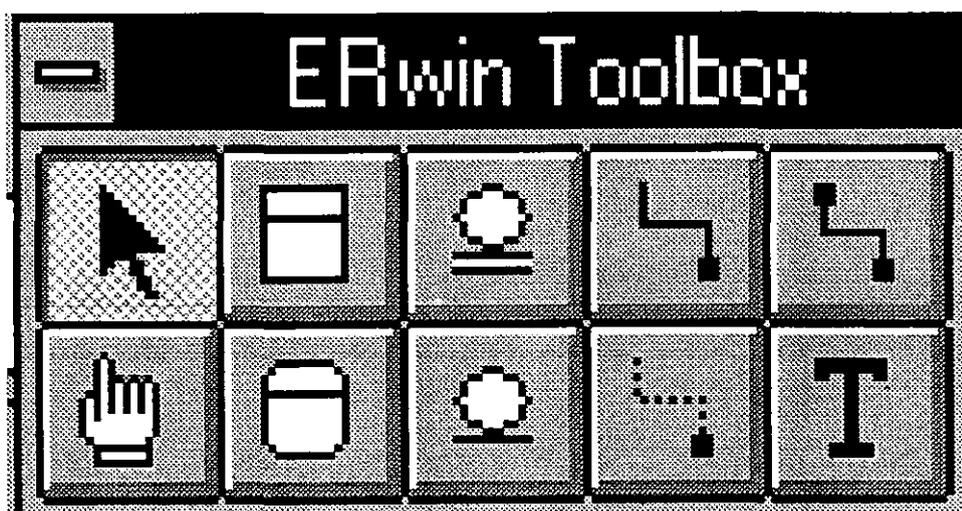


Figura 3.3.4.1. Erwin Toolbox.

Existen dos tipos de entidades a crear (figura 3.3.4.2):

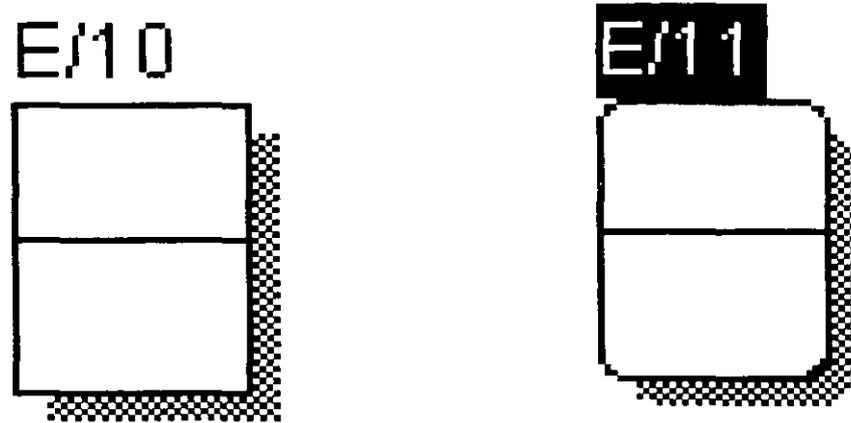


Figura 3.3.4.2. Entidad 1(E/10) y Entidad 2 (E/11).

- Una vez creadas las entidades que se necesitan, se procede a capturar los atributos de cada una de ellas. Para capturar los atributos es necesario seleccionar la entidad a editar y del menú seleccionar la opción 'Edit entity attribute' como se muestra en las figuras 3.3.4.3.

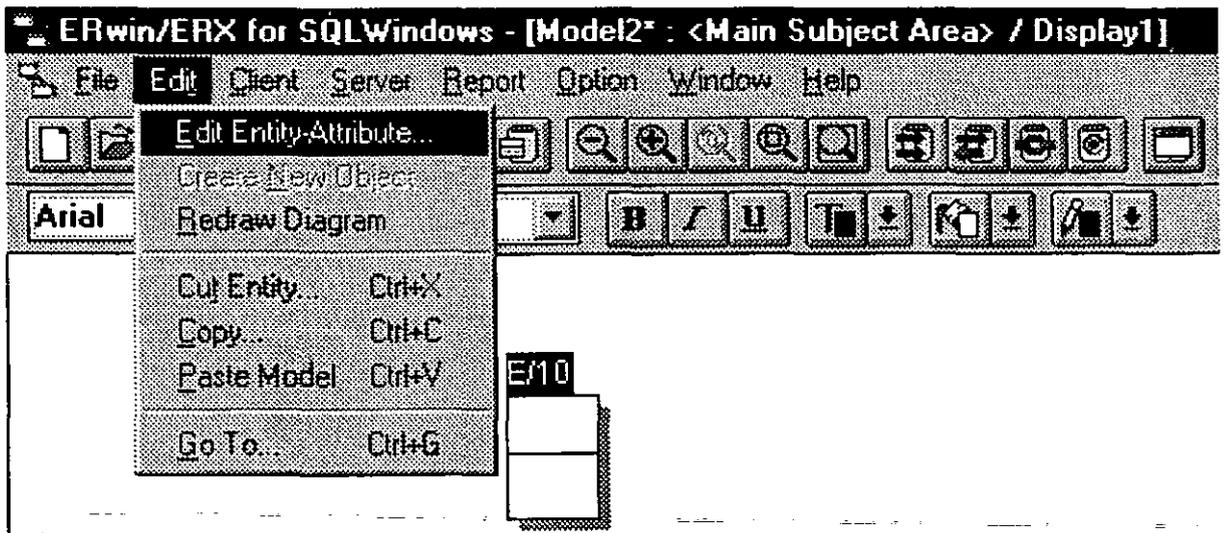


Figura 3.3.4.3 Editor de Entidades.

Y se desplegará la pantalla donde se capturan los atributos de la entidad como se muestra en la figura 3.3.4.4.

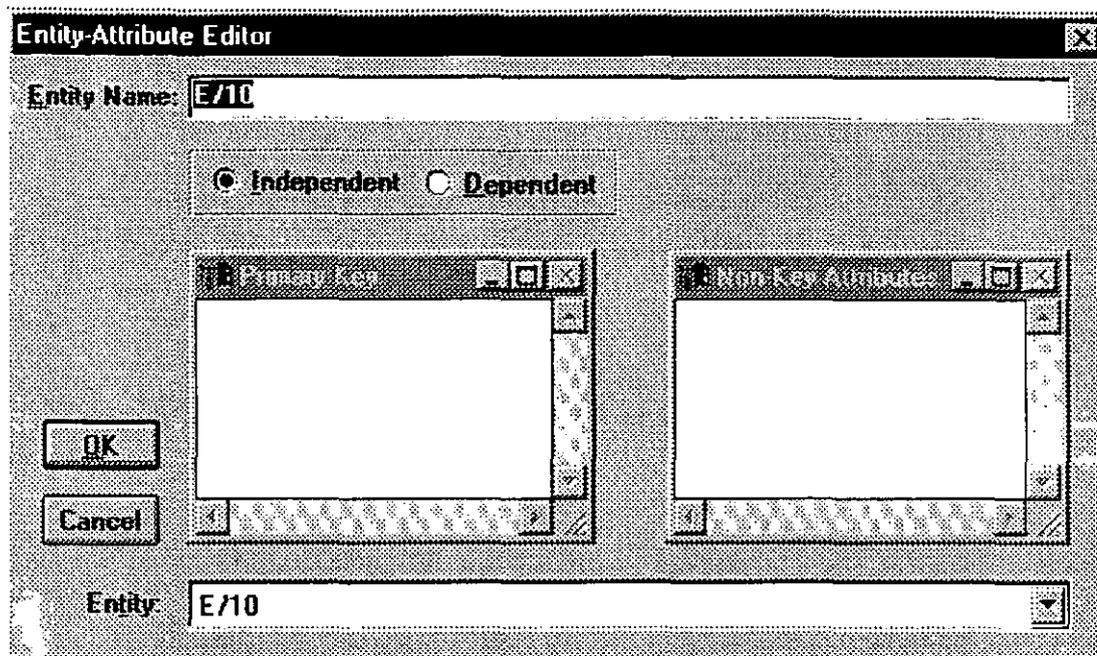


Figura 3.3.4.4 Captura de Atributos.

3. Una vez capturados los atributos de nuestras entidades se crean las ligas de relación entre entidades. Las relaciones entre entidades se realizan con ayuda del 'Erwin Toolbox'. Existen dos tipo de relación:

- *Identifying*, donde la llave que foránea pasa a formar parte de la llave primaria de la entidad a donde se exporta. En la figura 3.3.4.5 se muestra un ejemplo de una relación identifying.
- *Non_identifying*, donde la llave foránea NO pasa a formar parte de la llave primaria de la entidad a donde se exporta la llave. En la figura 3.3.4.6 se muestra un ejemplo de este tipo de relación.

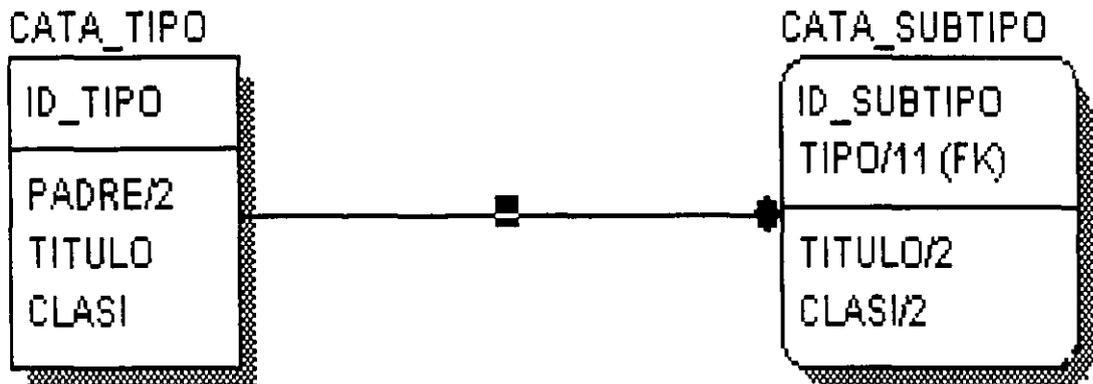


Figura 3.3.4.5 Relación Identifying.

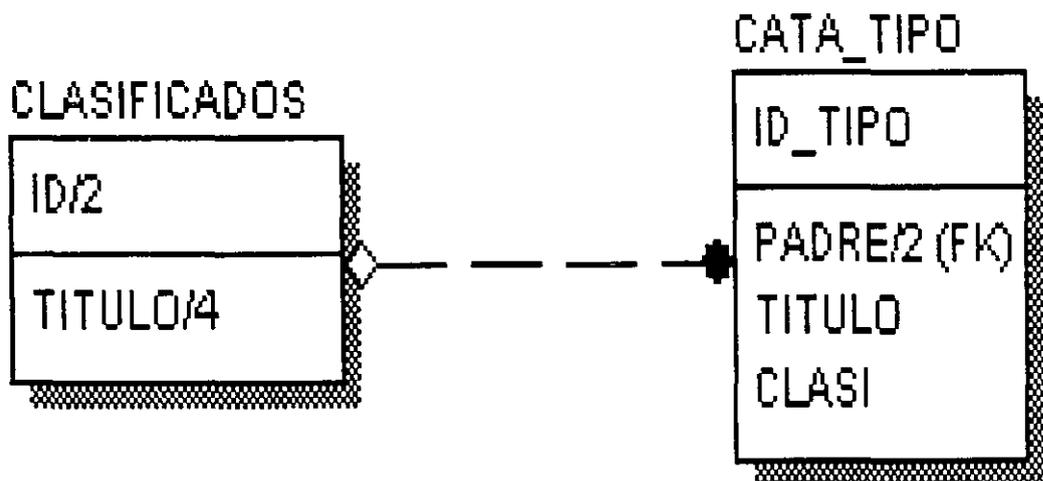


Figura 3.3.4.6 Relación Non_identifying.

- Al terminar de generar las relaciones el diagrama de Entidad-Relación queda listo para ser generado en Oracle.

En la figura 3.3.4.7. se muestra el diagrama Entidad-Relación Normalizado.

Diagrama de Entidad - Relación

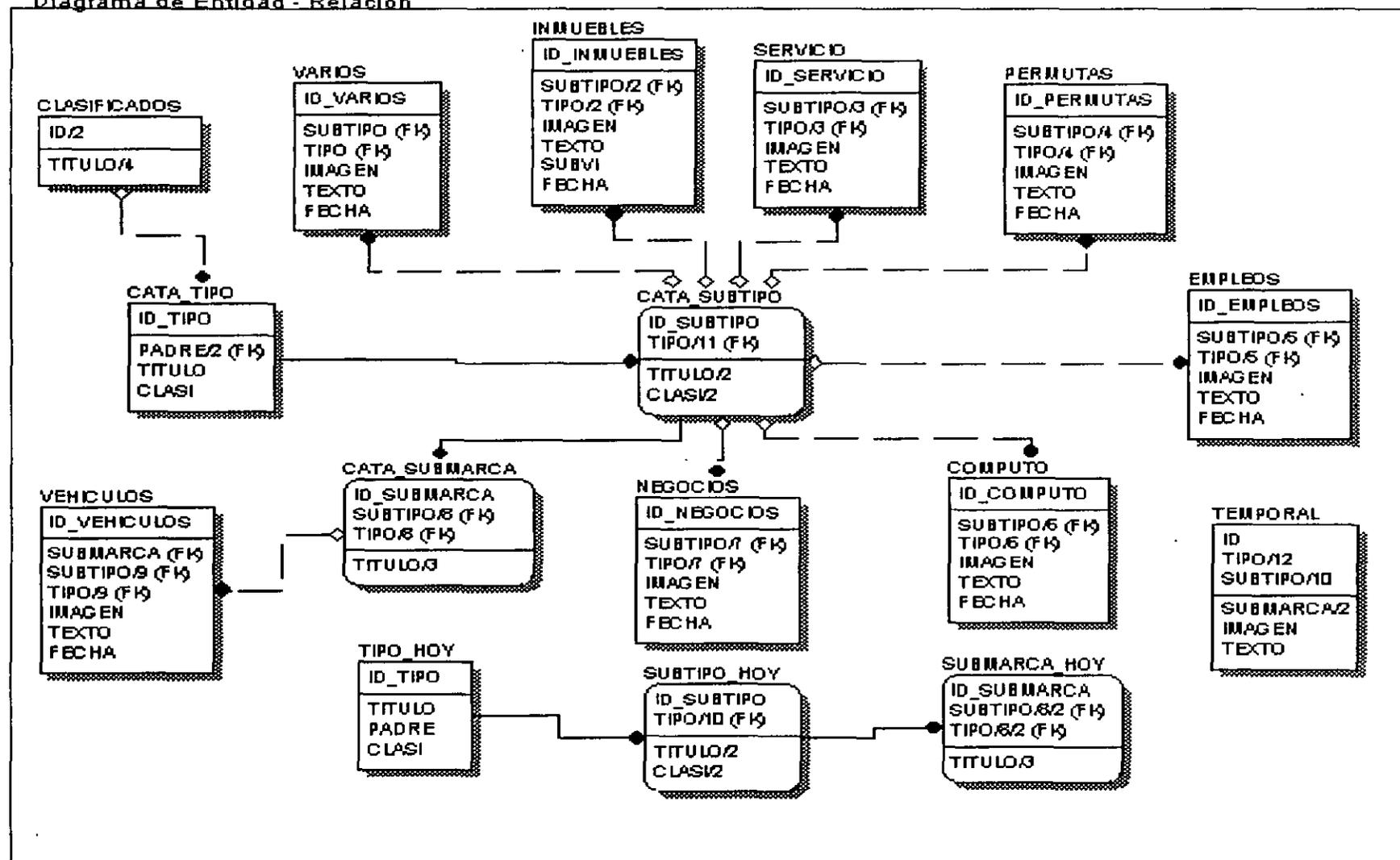


Figura 3.3.4.7 Diagrama Entidad-Relación

Una vez que se ha concluido con el diseño del sistema, se procede a generar el Back End de la Base de Datos. A continuación se mencionan los pasos a seguir:

Configuración de Context (Búsquedas de Texto).

- Crear 2 Tablespace para el uso del Cartucho Context:

```
SQL> create tablespace name context
      data_file 'd:\orant\database\context.ora' size 20M;
```

```
SQL> create tablespace name ctxtmp
      data_file 'd:\orant\database\ctxtmp.ora' size 10M;
```

- Crear los usuarios y privilegios para Context mediante los siguientes scripts:

```
SQL> connect sys/change_on_install
```

```
SQL> @ d:\orant\ctx80\admin\dr crsyst ctxsys passwdctxsys context ctxtmp
```

donde:

ctxsys "usuario administrador de Context",

passwdctxsys "password del ctxsys",

context "nombre del tablespace",

ctxsys "nombre del tablespace temporal"

- Crea los objetos para usar Context

```
SQL> connect ctxsys/ctxsys
```

```
SQL> @ d:\orant\ctx80\admin\dr0inst
```

Verificar Parámetros de la Base de Datos en el Archivo "initORCL.ora".

Para un buen funcionamiento del Cartucho de Context se recomiendan los siguientes valores:

```
share_pool_size = 30 000 000
```

```
process = 80
```

```
open_cursors = 250
```

```
text_enable = TRUE
```

(en caso de modificar alguno de los parámetros anteriores se debe levantar y tirar nuevamente la BD).

Creación del Usuario de la Base de Datos que Almacenará el Prototipo.

- Crear tablespace's para tablas del sistema.

```
SQL> connect system/manager
```

```
SQL> create tablespace noticias  
      datafile 'D:\ORANT\DATABASE\NOTIORCL.ORA' size 50M;
```

```
SQL> create tablespace notitmp  
      datafile 'D:\ORANT\DATABASE\NOTITMP.ORA' size 15M;
```

- Verifica que estén creados los nuevos Tablespaces.

```
SQL> select file_name,tablespace_name,bytes from dba_data_files;
```

- Crea el usuario de Base de Datos.

```
SQL> create user noti identified by noti
```

```
default tablespace noticias
```

```
temporary tablespace notitmp;
```

```
SQL> alter user noti quota unlimited on noticias;
```

```
SQL> grant connect, resource, dba to noti;
```

- Verifica que este creado el nuevo Usuario de BD.

```
SQL> select username from dba_users;
```

- Otorgar privilegios al usuario para utilizar Context.

```
SQL> connect ctxsys/ctxsys
```

```
SQL> grant ctxadmin to noti;
```

```
SQL> grant execute on ctx_query to noti;
```

Cambiar en la Base de Datos el Set de Caracteres a Español.

```
SQL> connect sys/change_on_install
```

```
SQL> update sys.props$ set value$ = 'SPANISH'  
       where name = 'NLS_LANGUAGE';
```

```
SQL> update sys.props$ set value$ = 'MEXICO'  
       where name = 'NLS_TERRITORY';
```

- Verificar si los nuevos valores se actualizaron.

```
SQL> select * from NLS_DATABASE_PARAMETERS;
```

Inicialización de “Servidores” para Context.

En el prompt de DOS, ejecutar la siguiente línea y minimizar la ventana de DOS.

```
C:\ sqlldr80 -user ctxsys/ctxsys -personality QDM
```

Creación de las Tablas y Objetos en la Base de Datos.

```
SQL> connect noti/noti  
SQL> @ c:\archivo\crea_base.sql
```

Insertar Algunos Valores Fijos del Catálogo en las Tablas.

```
SQL> connect noti/noti  
SQL> @ c:\archivo\tipo.sql  
SQL> @ c:\archivo\subtipo.sql  
SQL> @ c:\archivo\submarca.sql  
SQL> @ c:\archivo\falta_cata.sql
```

Insertar Datos Reales ya Formateados en la Tabla Temporal.

Desde el prompt de DOS ejecutar la siguiente línea:

```
C:\ sqlldr80 userid=noti/noti control=carga.ctl data=salida1.txt bindsizes= 20 000 000
```

Almacenar Datos en las Tablas Correspondientes.

```
SQL> connect noti/noti  
SQL> @ c:\archivo\l_t.sql  
SQL> @ c:\archivo\cata_hoy.sql
```

En el anexo 2 se incluye una muestra del código de algunos de los programas que se mencionaron con anterioridad.

3.4 DISEÑO Y PROGRAMACIÓN DEL FRONT END USANDO PL-SQL.

A continuación se muestra el código de los ocho paquetes (estructura lógica de programación) que sirven para publicar en Internet los clasificados de los usuarios divididos en las secciones: autos, varios, permutas, inmuebles, cómputo, servicios, empleos y negocios. El lenguaje de programación es PL/SQL, que permite combinar sentencias de tipo SQL y comandos de HTML (lenguaje de marcación hipertexto).

El primer paquete de programación es el correspondiente a la clasificación de "autos", y como se mencionó en el capítulo 2.5, está formado por 8 procedimientos (módulos de programación) que realizan acciones específicas. Para hacer una descripción más detallada del código de programación, los comentarios que se hacen en ciertas líneas del código están encerradas por los siguientes símbolos: `/** comentario **/`

El siguiente código puede estar almacenado en cualquier directorio, de cualquier máquina cliente que tenga conexión a la base de datos. Pueden hacerse las modificaciones necesarias en cualquier editor de palabras que guarde los cambios en formato texto. La forma de compilar estos paquetes se hace de la siguiente manera:

1. Abrir una sesión de SQL Plus, en una máquina cliente que tenga instalado el software: Oracle 8 client y además esté conectada al servidor donde reside la base de datos Oracle. También puede hacerse directamente en el servidor mediante la siguiente sentencia:

```
C:\> sqlplus usuario/password <enter>
```

2. Si el acceso fué correcto, cambiará el **prompt** (indicador del sistema) al de SQL y se tendrá que compilar el paquete dando la trayectoria donde se encuentre el archivo que contenga el código del paquete, por ejemplo si el archivo se llama: autos.sql se deberá ejecutar el siguiente comando:

```
SQL> @ c:\programas\autos.sql <enter>
```

3. Si el paquete no tuvo errores se indicará con un mensaje satisfactorio, en caso contrario mostrará la línea del programa en que hubo errores, por lo cual se deberá editar el programa y se deberán hacer las correcciones necesarias.

4. Para ver los resultados del paquete se deberá abrir un navegador de internet (explorer ó netscape) y se deberá dar la localización de la página, previamente configurada con Web Application Server de la siguiente manera:

```
http://nombre_máquina.dominio:9999/ruta_virtual/paquete.procedimiento
```

donde:

- nombre_máquina, es el nombre en el cual esta instalado el Web Application Server, por ejemplo: "pruebas".
- dominio, es el dominio donde esté localizada la máquina, por ejemplo: "unam.mx".
- 9999, es el puerto por el que Web Application Server va estar 'escuchando' las peticiones de los usuarios exclusivamente para ésta aplicación. La configuración de puertos puede hacerse desde 1024 hasta 64000.
- ruta_virtual, es la ruta que se configura en el Web Application Server para mapear las rutas físicas donde van a almacenarse los archivos html, de imagenes, etc. que requiera la aplicación, por ejemplo si los archivos que se utilizarán en la aplicación se encuentran en una estructura de directorios a partir de "c:\proyectos\clasificados" entonces en nuestra página Web puede mapearse como "/uni/" como se muestra a continuación.

Ruta física (en la máquina):

c:\proyectos\clasificados\

Ruta virtual (en Web):

/uni/

ej. c:\proyectos\clasificados\imagenes\logo.gif /uni/imagenes/logo.gif

- paquete.procedimiento, es el nombre del paquete y procedimiento al cual estamos haciendo referencia, por ejemplo: autos.forma_autos, donde "autos" es el nombre del paquete y "forma_autos" es el nombre del procedimiento.

Finalmente en el navegador se tendrá que escribir la siguiente dirección para ver los resultados de la compilación del primer paquete: "autos".

http://pruebas.unam.mx:9999/uni/autos.forma_autos

El código del paquete "autos" se muestra a continuación.

/** Declaración del paquete AUTOS, y los ocho procedimientos que lo conforman. *****/**

```
CREATE OR REPLACE PACKAGE autos AS
```

```

PROCEDURE forma_autos;
PROCEDURE vali_autos (p_opcion in NUMBER, p_variable in VARCHAR2);
PROCEDURE busca_autos (p_variable in VARCHAR2, p_opcion in NUMBER);
PROCEDURE subforma_autos (p_opcion in NUMBER);
PROCEDURE subvali_autos (p_subopcion in NUMBER, p_subvariable in VARCHAR2, p_opcion in
number);
PROCEDURE busca_autos_1 (p_subvariable in VARCHAR2, p_subopcion in NUMBER,
v_opcion in NUMBER);
PROCEDURE subforma_autos_1 (v_id_subtipo in VARCHAR2, p_opcion in NUMBER);
PROCEDURE busca_submarca (p_teropcion in NUMBER, p_tervariable in VARCHAR2,
v_id_subtipo in VARCHAR2, p_opcion in NUMBER);

```

```
END autos;
```

```
/
```

```
CREATE OR REPLACE PACKAGE BODY autos AS
```

```

/****** Inicia el Procedimiento FORMA_AUTOS, que muestra la forma captura *****/
/****** para que el usuario escriba sus opciones búsqueda. Ver figura 3.4.1 *****/

```

```
PROCEDURE forma_autos IS
```

```
BEGIN
```

```
http.p('<html><head><title>VEHICULOS</title></head>
```

```

<body bgcolor="#FFFFFF"><div align="CENTER">
<table border="0" width="468">
<tr><td colspan="2"><font color="#666666">
    </font>
    </td>
</tr></tr>
    <td colspan="2">
    </td>
</tr></table></div>
<CENTER><FONT COLOR="#CC0000" face="Arial" SIZE=2><B>Vehiculos</B></FONT>
</CENTER><CENTER>
<table border="0" width="500">
<tr><td>
    <FONT COLOR="#000000" SIZE=2>Seleccione un tema de su interés, aquellos temas
    marcados con ** contienen subtemas, (los cuales pueden verse sólo dando click en el botón BUSCAR)
    </td>
</tr> </FONT>
</table></CENTER>');

```

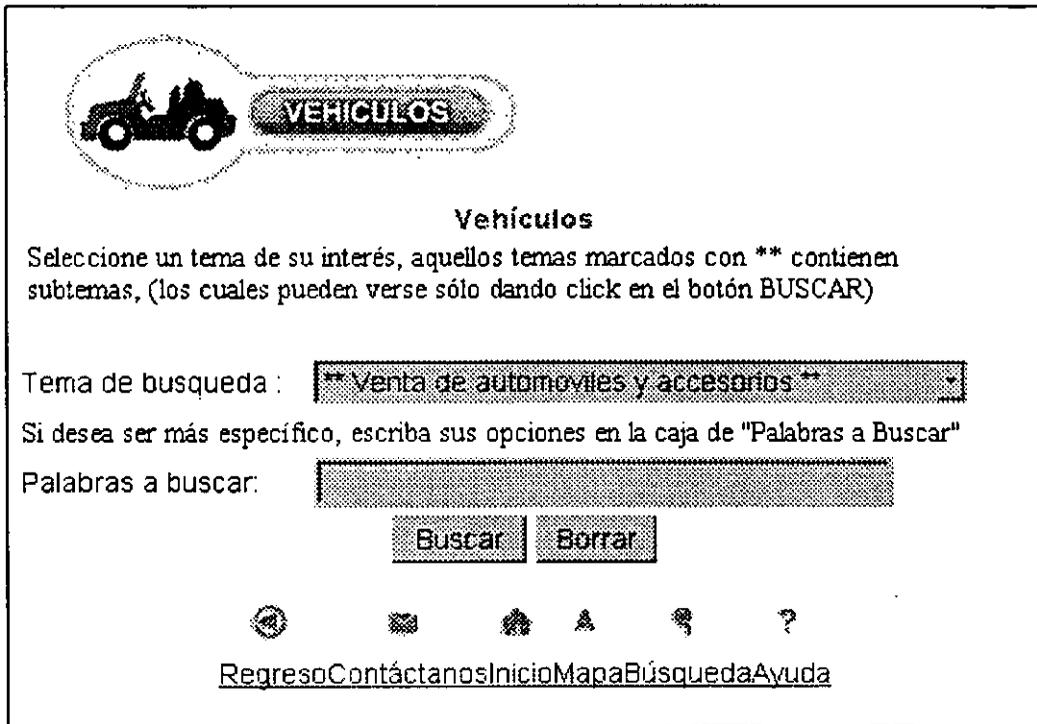


Figura 3.4.1 Forma de Captura del Procedimiento Autos.

*/*Abre una forma de captura que llama al procedimiento autos.vali_autos con las opciones del */*
*/** usuario y para desplegar las opciones actuales realiza "selects" a las tablas **/*
*/** corerspondientes. Ver figura 3.4.2 *****/*

```

htp.formOpen('autos.vali_autos');
htp.p(<CENTER>
<table cellspacing="0">
<tr><td><FONT COLOR="#000000" face="Arial" SIZE=2>Tema de busqueda :</font></td><td>');
    htp.formSelectOpen(cname=>'p_opcion');

```

```

FOR cur_op in (SELECT TITULO,ID_TIPO FROM tipo_hoy WHERE padre=98) LOOP
  htp.FormSelectOption(cur_op.titulo,'SELECTED',cattributes=>'VALUE=""||cur_op.id_tipo||"');
END LOOP;
  htp.p('</select></FONT>
</td></tr>
<tr><td width="500" colspan="2"><font color="#000000" SIZE=2>Si desea ser más específico,
escriba sus opciones en la caja de "Palabras a Buscar"</font>
</td></tr>
<tr><td>
  <font color="#000000" face="Arial" SIZE=2>Palabras a buscar:</font>
</td><td><input type="text" size="30" name="p_variable">
</td></tr>
</table><CENTER>
<font color="#000000" face="Arial" SIZE=2><input type="submit" value="Buscar"></font>
<font color="#000000" face="Arial" SIZE=2><input type="reset" value="Borrar"></font>
</CENTER>');
  htp.formClose;

```

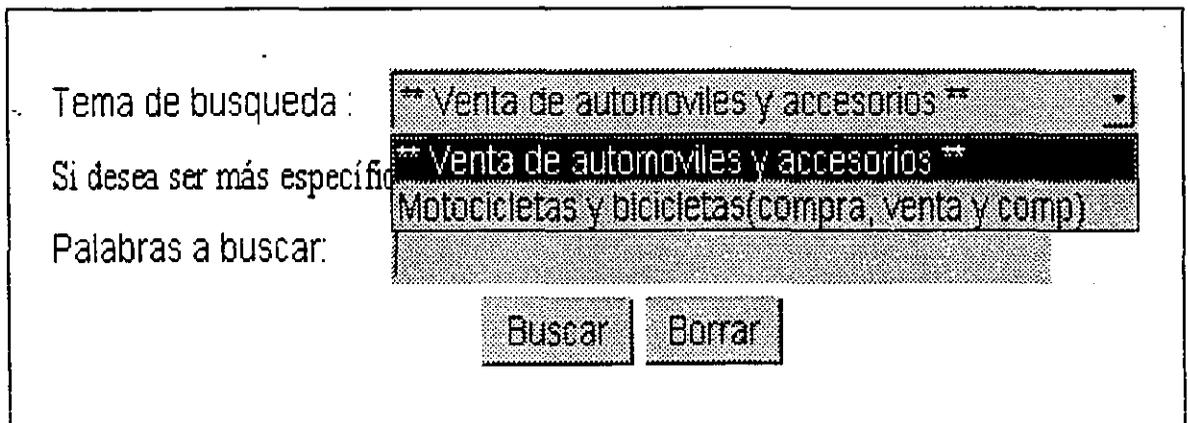


Figura 3.4.2 Despliega "select" de la Base de Datos.

/ Despliega el menú de pie de página para navegar en la página actual. Ver figura 3.4.3 ****/**

```

  htp.p('<div align="CENTER">
  <table border="0" width="280" cellspacing="0" cellpadding="0" vspace="0" hspace="0">
  <tr><td><div align="CENTER">
    <a href="http://ingenieria.unam.mx" target="_top">
    </a>
  </div></td><td><div align="CENTER">
    <a href="mailto:contacta@aguila.el-universal.com.mx">
    </a>
  </div></td><td><div align="CENTER">
    <a href="/uni/index.html" target="_top">
    </a>
  </div></td> <td><div align="CENTER">
    <a href="/uni/pages/mapasitio.htm" target="main">
    </a>
  </div></td> <td><div align="CENTER">
    <a href="busca.forma_busca" target="main">
    </a>

```

```

</div></td> <td><div align="CENTER">
  <a href="/uni/pages/ayuda.htm" target="main">
    </a>
</div></td> </tr><tr><td><div align="CENTER">
  <a href="http://ingenieria.unam.mx" target="_top">
    <font size="2" face="Arial">Regreso</font></a>
</div></td> <td><div align="CENTER">
  <font size="2" face="Arial"><a href="mailto:contacta@aguila.el-universal.com.mx">
    Contáctanos</a>
  </font>
</div></td> <td><div align="CENTER">
  <font face="Arial" size="2"><a href="/uni/index.html" target="_top">Inicio</a></font>
</div></td> <td><div align="CENTER">
  <font face="Arial" size="2"><a href="/uni/pages/mapasitio.htm"
target="main">Mapa</a></font>
</div></td> <td><div align="CENTER">
  <font face="Arial" size="2"><a href="busca.forma_busca" target="main">
Búsqueda</a></font>
</div></td> <td><div align="CENTER">
  <font size="2" face="Arial"><a href="/uni/pages/ayuda.htm" target="main">Ayuda</a></font>
</div> </td></tr></table></div>);
oraclelogo;
http.p('</BODY></HTML>');
END forma_autos;

```



Figura 3.4.3 Menú de Pie de Página.

```

/** El procedimiento VALI_AUTOS valida las opciones que el usuario introdujo para ejecutar ***/
/**** el procedimiento adecuado *****/

```

```

PROCEDURE vali_autos (p_opcion in NUMBER, p_variable in VARCHAR2) IS

```

```

v_clasi tipo_hoy.clasi%TYPE;          /**** declara una variable *****/

```

```

BEGIN

```

```

  select clasi into v_clasi from tipo_hoy where id_tipo=p_opcion;

```

```

  IF (v_clasi=1) or ((v_clasi=2) and (p_variable is NOT NULL)) THEN
    autos.busca_autos(p_variable, p_opcion); /** ejecuta el paquete.procedimiento **/

```

```

  ELSE

```

```

    autos.subforma_autos(p_opcion);          /** ejecuta el paquete.procedimiento **/

```

```

  END IF;

```

```
END vali_autos;
```

```
/** El procedimiento BUSCA_AUTOS utiliza el Cartucho de ConText para realizar búsquedas ***/
/** (consultas en memoria) con las palabras que el usuario introdujo. Ver figura 3.4.4 ****/
```

```
PROCEDURE busca_autos(p_variable in VARCHAR2, p_opcion in NUMBER) IS
```

```
score char(5);
pk char(5);
curid number;
texto varchar2(4000);
v_contador NUMBER:=0;
v_titulo tipo_hoy.titulo%TYPE;
CURSOR cur1 IS          /** declara un "cursor" para un almacenamiento temporal ***/
    SELECT texto
    FROM vehiculos
    WHERE tipo=p_opcion;
```

```
BEGIN
```

```
select titulo into v_titulo from tipo_hoy where id_tipo=p_opcion;
```

```
IF (p_variable is NOT NULL) THEN  /** Ejecuta instrucciones para consultas en memoria ***/
```

```
dbms_output.enable(1000000);
curid := ctx_query.open_con(policy_name => 'poli_vehiculos',
    text_query => p_variable,
    score_sorted => true,
    other_cols => 'texto',
    struct_query => 'tipo='||p_opcion||');
```

```
END IF;
```

```
/** Despliega los resultados acerca de la consulta realizada ****/
```

```
http.p('<html><head><title>VEHICULOS</title></head>
<body bgcolor="#FFFFFF"><div align="CENTER">
<table border="0" width="468">
<tr> <td colspan="2">
<font color="#666666"> </font>
</td></tr>
<tr><td colspan="2">

</td> </tr>
</table></div>
<center> <BR>Tema: <font size=3 color="#CC0000"> <B>'||v_titulo||'</B></FONT> <BR><BR>');
http.tableOpen;
IF (p_variable is NOT NULL) THEN
http.p('Palabra: <FONT SIZE=3 COLOR="#FF2211"><B>'||p_variable||'</B></FONT><BR>');
```

```
/** abre estructura en memoria para desplegar los resultados usando Context****/
```

```
while (ctx_query.fetch_hit(curid,pk,score,texto)>0)
loop
http.p('<tr><TD><IMG SRC = "/uni/images/fig13.gif"><BR></td><TD>'||texto||'</td></tr> ');
v_contador:=v_contador+1;
end loop;
ctx_query.close_con(curid);
```

/** invoca al cursor "cur1" para desplegar resultados sin utilizar Context **/

```
ELSE
  FOR i IN cur1 loop
    http.p('<tr><td><img src = "/uni/images/fig13.gif"><br></td><td>||i.texto||</td></tr> ');
    v_contador:=v_contador+1;
  END LOOP;
END IF;
http.tableClose;
http.p('<br>Número de Registros Encontrados: <font size=4 color="#2255FF"><b>
||v_contador||</b> </font><br>');
```



Tema: ** Venta de automoviles y accesorios **

Palabra: microbus

- ★ >MICROBUS. 2 minibuses, turísticos Chevrolet 1990 y 1991, 24 asientos ejecutivos, clima, detalles, 517-12-74.
- ★ >MICROBUS Alfa 1992, Ford fuel cromado tenencias. Tratamos \$89,000.- 5597-19-29.
- ★ >MICROBUS Casavan 1991, R 14, Portales. Magnífico. Tomaría coche. 5693-89-02.
- ★ >MICROBUS Chevrolet 1991, enganche \$17,600.- 2 años, para pagar. 5575-08-61.
- ★ >MICROBUS Chevrolet 1992, carrocería Rey Midas, excelente estado, \$90,000. 5781-34-18, 5781-55-37.
- ★ >MICROBUS Chevrolet 87, siempre particular, \$58,000.00. Kilómetro 43 carretera México Toluca Jajalpa. Teléfono: 01-728 515-61.

Figura 3.4.4. Despliegue de Registros Encontrados.

/** Despliega menú de pie de página para navegar en la página actual **/

```
http.p('<br><div align="CENTER">
<table border="0" width="280" cellspacing="0" cellpadding="0" vspace="0" hspace="0">
<tr><td><div align="CENTER">
  <a href="autos.forma_autos">
    </a>
</div></td> <td><div align="CENTER">
  <a href="mailto:contacta@aguila.el-universal.com.mx">
    </a>
</div></td> <td><div align="CENTER">
  <a href="/uni/index.html" target="_top">
    </a>
</div></td> <td><div align="CENTER">
  <a href="/uni/pages/mapasitio.htm" target="main">
    </a>
</div></td></tr></table>');
```

```

</div></td> <td><div align="CENTER">
  <a href="busca.forma_busca" target="main">
    </a>
</div></td> <td><div align="CENTER">
  <a href="/uni/pages/ayuda.htm" target="main">
    </a>
</div></td> </tr>
<tr><td><div align="CENTER">
  <a href="autos.forma_autos"> <font size="2" face="Arial">Regreso</font></a>
</div></td> <td><div align="CENTER">
  <font size="2" face="Arial">
    <a href="mailto:contacta@aguila.el-universal.com.mx">Cont&aacute;ctanos</a></font>
</div></td> <td><div align="CENTER">
  <font face="Arial" size="2">
    <a href="/uni/index.html" target="_top">Inicio</a></font>
</div></td> <td><div align="CENTER">
  <font face="Arial" size="2"><a href="/uni/pages/mapasitio.htm"
target="main">Mapa</a></font>
</div></td> <td><div align="CENTER">
  <font face="Arial" size="2">
    <a href="busca.forma_busca" target="main">B&uacute;squeda</a></font>
</div></td> <td><div align="CENTER">
  <font size="2" face="Arial"><a href="/uni/pages/ayuda.htm" target="main">Ayuda</a></font>
</div> </td></tr></table></div>);
oraclelogo;
htp.p('</BODY></HTML>');
END busca_autos;

```

/* El procedimiento SUBFORMA_AUTOS despliega una subforma de captura para que el usuario */

/* introduzca nuevas opciones de búsqueda. Ver figura 3.4.5 *****/**

PROCEDURE subforma_autos (p_opcion in NUMBER)IS

v_titulo subtipo_hoy.titulo%TYPE; **/*** declara una variable ***/**

BEGIN

select titulo into v_titulo from tipo_hoy where id_tipo=p_opcion;

htp.p('<html><head><title>VEHICULOS</title></head>

<body bgcolor="#FFFFFF"><div align="CENTER">

<table border="0" width="468">

<tr><td colspan="2">

</td></tr>

<tr> <td colspan="2">

</td></tr>

</table></div>

<center>||v_titulo||</center>

<center><table border="0" width="500"><tr><td>

Seleccione un tema de su interés, aquellos temas marcados con ** contienen subtemas,(los cuales pueden verse sólo dando click en el botón BUSCAR)

</td></tr>

</table></center>);

/ Abre una forma para ejecutar "autos.subvali_autos" con las nuevas opciones del usuario **/**

```

htp.formOpen('autos.subvali_autos');
htp.p('<center>
<table cellspacing="0">
<tr><td>
<font color="#000000" face="Arial" SIZE=2>Tema de busqueda :</font>
</td><td>);
htp.formSelectOpen(cname=>'p_subopcion');

```

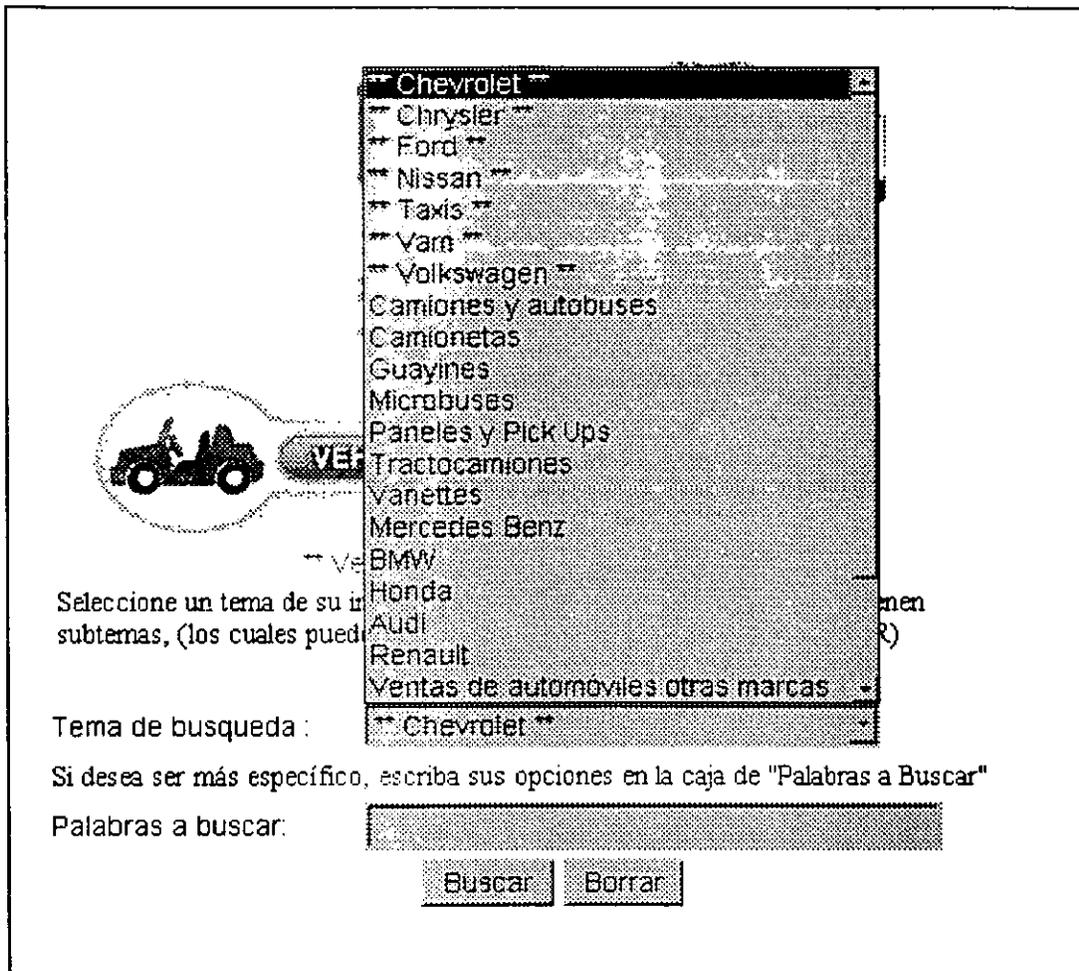


Figura 3.4.5 Despliega Submenú de Autos.

/ Realiza "selects" a las tablas correspondientes para desplegar sólo los valores actuales ***/**

```

for cur_opp in (SELECT TITULO.ID FROM subtipo_hoy WHERE tipo=p_opcion)
LOOP
    htp.FormSelectOption(cur_opp.titulo,'SELECTED',cattributes=>'VALUE=""||cur_opp.id||"');
END LOOP;
htp.p('</select></FONT>
</td></tr>

```

```

<tr><td width="500" colspan="2">
  <font color="#000000" SIZE=2>Si desea ser más específico, escriba sus opciones en la caja de
  "Palabras a Buscar"</font>
</td></tr>
<tr><td>
  <font color="#000000" face="Arial" SIZE=2>Palabras a buscar:</font>
</td><td>
  <input type="text" size="30" name="p_subvariable">
</td></tr></table>
  <INPUT TYPE="HIDDEN" NAME="p_opcion" SIZE="20" VALUE="||p_opcion||"><center>
  <font color="#000000" face="Arial" SIZE=2>
  <input type="submit" value="Buscar"></font>
  <font color="#000000" face="Arial" SIZE=2><input type="reset"
value="Borrar"></font></center>');
  http.formClose;

```

****** despliega menú de pie de página para navegar en la página actual ****/**

```

http.p('<div align="center">
  <table border="0" width="280" cellspacing="0" cellpadding="0" vspace="0" hspace="0">
  <tr><td><div align="center">
    <a href="autos.forma_autos">
    </a>
  </div></td> <td><div align="center">
    <a href="mailto:contacta@aguila.el-universal.com.mx">
    </a>
  </div></td> <td><div align="center">
    <a href="/uni/index.html" target="_top">
    </a>
  </div></td> <td><div align="center">
    <a href="/uni/pages/mapasitio.htm" target="main">
    </a>
  </div></td> <td><div align="center">
    <a href="busca.forma_busca" target="main">
    </a>
  </div></td> <td><div align="center">
    <a href="/uni/pages/ayuda.htm" target="main">
    </a>
  </div></td> </tr>
  <tr><td><div align="center">
    <a href="autos.forma_autos"> <font size="2" face="Arial">Regreso</font></a>
  </div></td> <td><div align="center">
    <font size="2" face="Arial">
    <a href="mailto:contacta@aguila.el-universal.com.mx">Cont&aacute;tanos</a></font>
  </div></td> <td><div align="center">
    <font face="Arial" size="2"><a href="/uni/index.html" target="_top">Inicio</a></font>
  </div></td> <td><div align="center">
    <font face="Arial" size="2"><a href="/uni/pages/mapasitio.htm"
target="main">Mapa</a></font>
  </div></td> <td><div align="center">
    <font face="Arial" size="2"><a href="busca.forma_busca"
target="main">B&uacute;squeda</a></font>
  </div></td> <td><div align="center">
    <font size="2" face="Arial"><a href="/uni/pages/ayuda.htm" target="main">Ayuda</a></font>
  </div> </td></tr></table></div>');
  oraclelogo;

```

```

http.p('</BODY></HTML>');
END subforma_autos;

```

/ El procedimiento SUBVALI_AUTOS valida las opciones que el usuario introdujo *****/**
/ para ejecutar el procedimiento correspondiente *****/**

```

PROCEDURE subvali_autos(p_subopcion in number, p_subvariable in varchar2,p_opcion in number) IS

```

```

v_clasi      tipo_hoy.clasi%TYPE;          /** declara variables del mismo tipo**/
v_id_subtipo subtipo_hoy.id_subtipo%TYPE; /** que la tabla.columna **/

```

```

BEGIN

```

```

    select clasi,id_subtipo
    into v_clasi,v_id_subtipo
    from subtipo_hoy
    where id=p_subopcion;

```

```

    IF (v_clasi=1) or ((v_clasi=2) and (p_subvariable is NOT NULL)) THEN
        autos.busca_autos_1(p_subvariable,p_subopcion,p_opcion);

```

```

    ELSE

```

```

        autos.subforma_autos_1(v_id_subtipo,p_opcion);

```

```

    END IF;

```

```

END subvali_autos;

```

/*El procedimiento BUSCA_AUTOS_1 realiza una búsqueda con las opciones nuevas del usuario.
Ver figura 3.4.4, en donde la forma de despliegue es similar */**

```

PROCEDURE busca_autos_1 ( p_subvariable in VARCHAR2, p_subopcion in NUMBER,
                          v_opcion in NUMBER) IS

```

```

score char(5);          /** declaración de variables ***/

```

```

pk char(5);

```

```

curid number;

```

```

curid2 number;

```

```

score2 char(5);

```

```

pk2 char(5);

```

```

texto varchar2(4000);

```

```

v_contador NUMBER:=0;

```

```

v_tipo subtipo_hoy.tipo%TYPE;

```

```

v_subtipo subtipo_hoy.id_subtipo%TYPE;

```

```

v_titulo subtipo_hoy.titulo%TYPE;

```

```

v_id_subtipo varchar2(2);

```

```

CURSOR cur1 IS          /** declara un cursor para consultas directas en tablas ***/

```

```

    SELECT texto

```

```

    FROM vehiculos

```

```

    WHERE subtipo = (select id_subtipo from subtipo_hoy where id=p_subopcion)

```

```

    AND tipo = (select tipo from subtipo_hoy where id=p_subopcion);

```

```

BEGIN

```

```

    SELECT titulo into v_titulo FROM subtipo_hoy

```

```

    WHERE id=p_subopcion;

```

```

http.p('<html><head><title>VEHICULOS</title></head>

```

```

<body bgcolor="#FFFFFF"><div align="center">
<table border="0" width="468">
<tr><td colspan="2">
<font color="#666666"> </font>
</td></tr>
<tr><td colspan="2">

</td> </tr> </table></div>
<center> <BR>Tema: <FONT SIZE=4 COLOR="#CC0000"><B>'||v_titulo||'</B></FONT><BR>);

```

****** ejecuta instrucciones para realizar consultas en memoria con Context ****/**

```

IF (p_subvariable is NOT NULL) THEN
dbms_output.enable(1000000);
IF ( (p_subopcion >=133) and (p_subopcion <= 139)) THEN
select tipo,id_subtipo into v_tipo,v_subtipo from subtipo_hoy where id=p_subopcion;
curid := ctx_query.open_con(policy_name => 'poli_vehiculos',
text_query => p_subvariable,
score_sorted => true,
other_cols => 'texto',
struct_query => 'tipo='||v_tipo||' and subtipo='''||v_subtipo||''');
ELSE
select tipo,substr(id_subtipo,1,2) into v_tipo,v_id_subtipo from subtipo_hoy where id=p_subopcion;
curid2 := ctx_query.open_con(policy_name => 'poli_vehiculos',
text_query => p_subvariable,
score_sorted => true,
other_cols => 'texto',
struct_query => 'tipo='||v_tipo||' and submarca='''||v_id_subtipo||''');
END IF;
END IF;

```

/ Despliega resultados de la consulta en memoria con Context **/**

```

htp.tableOpen;
IF (p_subvariable is NOT NULL) THEN
htp.p('Palabra: <FONT SIZE=3 COLOR="#FF2211"><B>'||p_subvariable||'</B></FONT><BR>');
IF ( (p_subopcion >=133) and (p_subopcion <= 139)) THEN
while (ctx_query.fetch_hit(curid,pk,score,texto)>0)
loop
htp.p('<tr><td><IMG SRC = "/uni/images/fig13.gif"><BR></td><td>'||texto||'</td></tr> ');
v_contador:=v_contador+1;
end loop;
ctx_query.close_con(curid);
ELSE
while (ctx_query.fetch_hit(curid2,pk2,score2,texto)>0)
loop
htp.p('<tr><td><IMG SRC = "/uni/images/fig13.gif"><br></td><td>'||texto||'</td></tr> ');
v_contador:=v_contador+1;
end loop;
ctx_query.close_con(curid2);
END IF;

```

/ despliega resultados de la búsqueda directamente en tablas, sin utilizar Context **/**

```

ELSE
  FOR i IN cur1 loop
    http.p('<tr><td><IMG SRC = "/uni/images/fig13.gif"><BR></td><td>'||i.texto||'</td></tr> ');
    v_contador:=v_contador+1;
  END LOOP;
END IF;
http.tableClose;
http.p('<br>Número de Registros Encontrados: <FONT SIZE=4 COLOR="#2255FF"><B>
'||v_contador||' </B></FONT>');

```

/ Despliega menú de pie de página para navegar en la página actual **/**

```

http.p('<br><BR><div align="center">
<table border="0" width="280" cellspacing="0" cellpadding="0" vspace="0" hspace="0">
<tr><td><div align="center">
  <a href="autos.subforma_autos?p_opcion=||v_opcion||">
    </a>
</div></td><td><div align="center">
  <a href="mailto:contacta@aguila.el-universal.com.mx">
    </a>
</div></td><td><div align="center">
  <a href="/uni/index.html" target="_top">
    </a>
</div></td><td><div align="center">
  <a href="/uni/pages/mapasitio.htm" target="main">
    </a>
</div></td><td><div align="center">
  <a href="busca.forma_busca" target="main">
    </a>
</div></td><td><div align="center">
  <a href="/uni/pages/ayuda.htm" target="main">
    </a>
</div></td></tr>
<tr><td><div align="center">
  <a href="autos.subforma_autos?p_opcion=||v_opcion||">
    <font size="2" face="Arial">Regreso</font></a>
</div></td><td><div align="center">
  <font size="2" face="Arial">
  <a href="mailto:contacta@aguila.el-universal.com.mx">Contáctanos</a></font>
</div></td><td><div align="center">
  <font face="Arial" size="2"><a href="/uni/index.html" target="_top">Inicio</a></font>
</div></td><td><div align="center">
  <font face="Arial" size="2"><a href="/uni/pages/mapasitio.htm"
target="main">Mapa</a></font>
</div></td><td><div align="center">
  <font face="Arial" size="2"><a href="busca.forma_busca" target="main">Búsqueda</a></font>
</div></td><td><div align="center">
  <font size="2" face="Arial"><a href="/uni/pages/ayuda.htm" target="main">Ayuda</a></font>
</div></td></tr></table></div>');
oraclelogo;
http.p('</BODY></HTML>');
END busca_autos_1;

```

**/* El procedimiento SUBFORMA_AUTOS_1 muestra una forma de captura para las nuevas */
 /* opciones que el usuario indique, la forma muestra las submarcas de autos que */
 /* existen actualmente. Ver figura 3.4.6 **/**

```
PROCEDURE subforma_autos_1(v_id_subtipo in VARCHAR2,p_opcion in NUMBER)IS
```

```
v_titulo SUBTIPO_HOY.titulo%TYPE;          /** declaración de variable **/
```

```
BEGIN
```

```
select titulo into v_titulo from subtipo_hoy where
  tipo=p_opcion and
  id_subtipo=v_id_subtipo;
```

```
htp.p('<html><head><title>VEHICULOS</title></head>
```

```
<body bgcolor="#FFFFFF"><div align="center">
```

```
<table border="0" width="468">
```

```
<tr><td colspan="2">
```

```
<font color="#666666">
```

```
 </font>
```

```
</td></tr>
```

```
<tr> <td colspan="2">
```

```

```

```
</td> </tr> </table></div>
```

```
<center><font color="#CC0000" face="Arial" SIZE=2>||v_titulo||</FONT></center>
```

```
<center><table border="0" width="500">
```

```
<tr><td>
```

```
<font color="#000000" SIZE=2>Seleccione un tema de su interés, aquellos temas marcados
con ** contienen subtemas, (los cuales pueden verse sólo dando click en el botón BUSCAR)
```

```
</td></tr></FONT>
```

```
</table></center>');

```

```
/** Abre una forma para ejecutar "autos.busca_submarca" con las nuevas opciones del usuario
**/
```

```
htp.formOpen('autos.busca_submarca');
```

```
htp.p('<center>
```

```
<table cellspacing="0">
```

```
<tr><td>
```

```
<font color="#000000" face="Arial" SIZE=2>Tema de búsqueda :</font>
```

```
</td><td>');

```

```
htp.formSelectOpen(cname=>'p_teropcion');
```

```
FOR cur_opp in (SELECT TITULO,ID FROM submarca_hoy WHERE id_subtipo=v_id_subtipo)
```

```
LOOP
```

```
htp.FormSelectOption(cur_opp.titulo,'SELECTED',cattributes=>'VALUE="||cur_opp.id||"');
```

```
END LOOP;
```

```
htp.p('</select></FONT>
```

```
</td></tr>
```

```
<tr><td width="500" colspan="2">
```

```
<font color="#000000" SIZE=2>Si desea ser más específico, escriba sus opciones en la caja de
"Palabras a Buscar"</font>
```

```
</td></tr>
```

```
<tr><td>
```

```
<font color="#000000" face="Arial" SIZE=2>Palabras a buscar:</font>
```

```
</td><td>
```

```
<input type="text" size="30" name="p_tervariable">
```

```
</td></tr></table>
```

```
<INPUT TYPE="HIDDEN" NAME="v_id_subtipo" SIZE="20" VALUE="||v_id_subtipo||">
```

```
<INPUT TYPE="HIDDEN" NAME="p_opcion" SIZE="20" VALUE="||p_opcion||">
```

```
<font color="#000000" face="Arial" SIZE=2><input type="submit" value="Buscar"></font>
```

```

<font color="#000000" face="Arial" SIZE=2><input type="reset"
value="Borrar"></font></center>);
http.formClose;

```

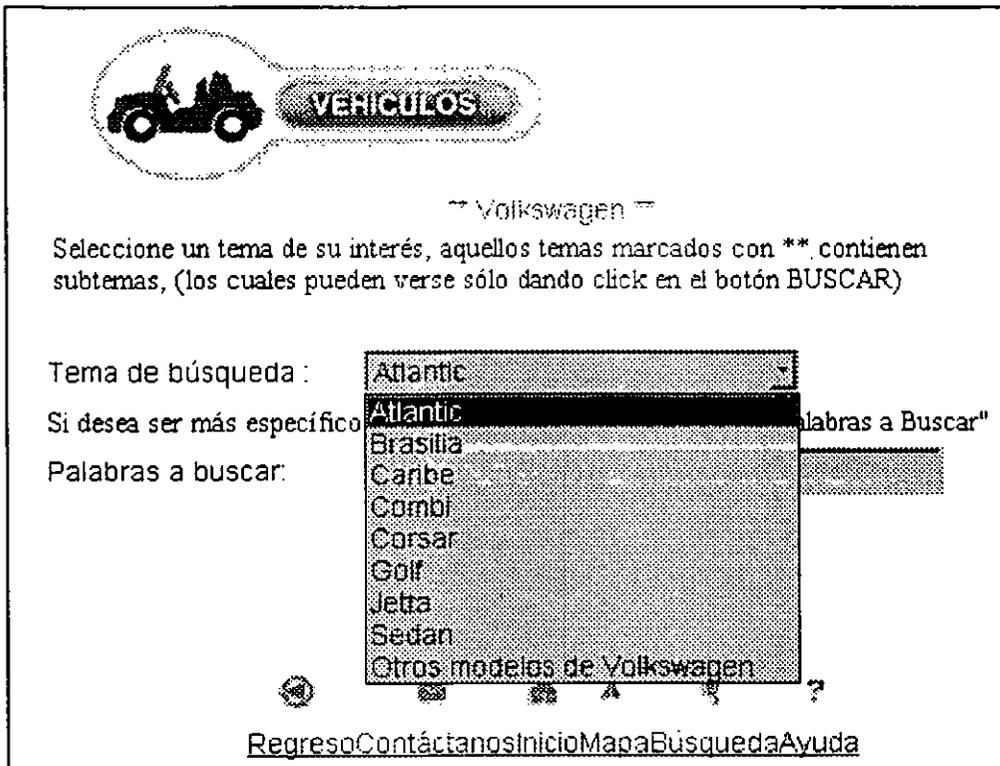


Figura 3.4.6 Forma de Subcaptura para Autos.

/** Despliega menú de pie de página para navegar en la página actual **/

```

http.p(<br><BR><div align="center">
<table border="0" width="280" cellspacing="0" cellpadding="0" vspace="0" hspace="0">
<tr><td><div align="center">
<a href="autos.subforma_autos?p_opcion=||p_opcion||">
</a>
</div></td><td><div align="center">
<a href="mailto:contacta@aguila.el-universal.com.mx">
</a>
</div></td><td><div align="center">
<a href="/uni/index.html" target="_top">
</a>
</div></td><td><div align="center">
<a href="/uni/pages/mapasitio.htm" target="main">
</a>
</div></td><td><div align="center">
<a href="busca.forma_busca" target="main">
</a>
</div></td><td><div align="center">
<a href="/uni/pages/ayuda.htm" target="main">
</a>

```

```

</div></td>
</tr><tr><td align="center">
  <a href="autos.subforma_autos?p_opcion=||p_opcion||">
    <font size="2" face="Arial">Regreso</font></a>
</div></td> <td><div align="center">
  <font size="2" face="Arial">
    <a href="mailto:contacta@aguila.el-universal.com.mx">Contáctanos</a></font>
</div></td> <td><div align="center">
  <font face="Arial" size="2"><a href="/uni/index.html" target="_top">Inicio</a></font>
</div></td> <td><div align="center">
  <font face="Arial" size="2"><a href="/uni/pages/mapasitio.htm"
target="main">Mapa</a></font>
</div></td> <td><div align="center">
  <font face="Arial" size="2">
    <a href="busca.forma_busca" target="main">B&uacute;squeda</a></font>
</div></td> <td><div align="center">
  <font size="2" face="Arial"><a href="/uni/pages/ayuda.htm" target="main">Ayuda</a></font>
</div></td></tr></table></div>");
oraclelogo;
http.p('</BODY></HTML>');
END subforma_autos_1;

```

/* El procedimiento BUSCA_SUBMARCA realiza búsquedas con las opciones del usuario. */
/* Despliega los registros como en la figura 3.4.4 */**

```

PROCEDURE busca_submarca(p_teropcion in NUMBER,p_tervariable in VARCHAR2,
v_id_subtipo in VARCHAR2,p_opcion in NUMBER ) IS

```

```

score char(5);          /** declaración de variables **/
pk char(5);
curid number;
texto varchar2(4000);
v_contador NUMBER:=0;
v_tipo submarca_hoy.tipo%TYPE;
v_id_submarca submarca_hoy.id_submarca%TYPE;
v_titulo submarca_hoy.titulo%TYPE;

```

```

CURSOR cur1 IS          /** utiliza un cursor para búsquedas en tablas **/
  SELECT texto
  FROM vehiculos
  WHERE subtipo =(select id_submarca from submarca_hoy where id=p_teropcion)
  AND tipo = (select tipo from submarca_hoy where id=p_teropcion);
BEGIN
  SELECT titulo,id_submarca,tipo
  into v_titulo, v_id_submarca,v_tipo from submarca_hoy where id=p_teropcion;

```

/ utiliza Context para hacer "consultas en memoria" con las opciones del usuario ***/**

```

IF (p_tervariable is NOT NULL) THEN
  curid := cx_query.open_con(policy_name => 'poli_vehiculos',
  text_query => p_tervariable,
  score_sorted => true,
  other_cols => 'texto',
  struct_query => 'tipo=||v_tipo||' and subtipo=""||v_id_submarca||"");
END IF;

```

```

http.p ('<html><head><title>VEHICULOS</title> </head>
<body bgcolor="#FFFFFF"><div align="center">
<table border="0" width="468">
<tr><td colspan="2">
<font color="#666666"> 
</font>
</td></tr>
<tr> <td colspan="2">

</td> </tr> </table></div>
<center> <BR>Tema: <FONT SIZE=4 COLOR="#CC0000"><B>'||v_titulo||'</B></FONT><BR>');
http.tableOpen;

```

/ Despliega resultados de la búsqueda con Context **/**

```

IF (p_tervariable is NOT NULL) THEN
while (ctx_query.fetch_hit(curid,pk,score,texto)>0)
loop
http.p('<tr><td><IMG SRC = "/uni/images/fig13.gif"><BR></td><td>'||texto||'</td></tr> ');
v_contador:=v_contador+1;
end loop;
ctx_query.close_con(curid);

```

/ Despliega resultados de la búsqueda directamente de las tablas, usando un cursor **/**

```

ELSE
FOR i IN cur1 loop
http.p('<tr><td><IMG SRC = "/uni/images/fig13.gif"><BR></td><td>'||i.texto||'</td></tr> ');
v_contador:=v_contador+1;
END LOOP;
END IF;
http.tableClose;
http.p('<BR>Número de Registros Encontrados: <FONT SIZE=4 COLOR="#2255FF"> '||v_contador||'
</FONT>');

```

/ Despliega el menú de pie de página para navegar en la página actual **/**

```

http.p('<br><br><div align="center">
<table border="0" width="280" cellspacing="0" cellpadding="0" vspace="0" hspace="0">
<tr><td><div align="center">
<a href="autos.subforma_autos_1?v_id_subtipo='||v_id_subtipo||'&p_opcion='||p_opcion||'">
</a>
</div></td>
<td><div align="center">
<a href="mailto:contacta@aguila.el-universal.com.mx">
</a>
</div></td> <td><div align="center">
<a href="/uni/index.html" target="_top">
</a>
</div></td> <td><div align="center">
<a href="/uni/pages/mapasitio.htm" target="main">
</a>
</div></td> <td><div align="center">
<a href="busca.forma_busca" target="main">
</a>
</div></td> <td><div align="center">

```

```

    <a href="/uni/pages/ayuda.htm" target="main">
    </a>
</div></td> </tr>
<tr><td><div align="center">
    <a href="autos.subforma_autos_1?v_id_subtipo=||v_id_subtipo||&p_opcion=||p_opcion||">
    <font size="2" face="Arial">Regreso</font></a>
</div></td> <td><div align="center">
    <font size="2" face="Arial">
    <a href="mailto:contacta@aguila.el-universal.com.mx">Cont&aacute;ctanos</a></font>
</div></td> <td><div align="center">
    <font face="Arial" size="2"><a href="/uni/index.html" target="_top">Inicio</a></font>
</div></td> <td><div align="center">
    <font
        face="Arial"
        size="2"><a
            href="/uni/pages/mapasitio.htm"
target="main">Mapa</a></font>
</div></td> <td><div align="center">
    <font face="Arial" size="2">
    <a href="busca.forma_busca" target="main">B&uacute;squeda</a></font>
</div></td> <td><div align="center">
    <font size="2" face="Arial"><a href="/uni/pages/ayuda.htm" target="main">Ayuda</a></font>
</div> </td></tr></table></div>');
oraclelogo;
http.p('</BODY></HTML>');

END busca_submarca;          /** fin del procedimiento */

END autos;                   /** fin del paquete */
/

```

Para más información respecto al código de los demás módulos consulte el anexo 3.

3.5 INTEGRACIÓN Y DEPURACIÓN DEL SISTEMA.

Hasta el momento se tienen concluidas las dos partes importantes que conforman el sistema de consulta de clasificados en Internet: el Back End y el Front End. Recordando lo fundamental de cada una de estas partes, tenemos que el Back End está formado por la configuración y diseño de la Base de Datos que conforma al sistema, es decir, la construcción de tablas y la relación entre ellas para el almacenamiento de los clasificados.

En la parte del Front End se tiene la programación final con la que el usuario podrá hacer la consulta de clasificados que más le interese; en esta programación se han incluido referencias a las tablas de donde se obtendrá la información para mostrarla a los usuarios. Estos programas se encuentran en formato de texto y podrán ser utilizados cuando el administrador del sistema lo requiera.

El esquema hasta ahora de estas partes puede verse en la Figura 3.5.1.

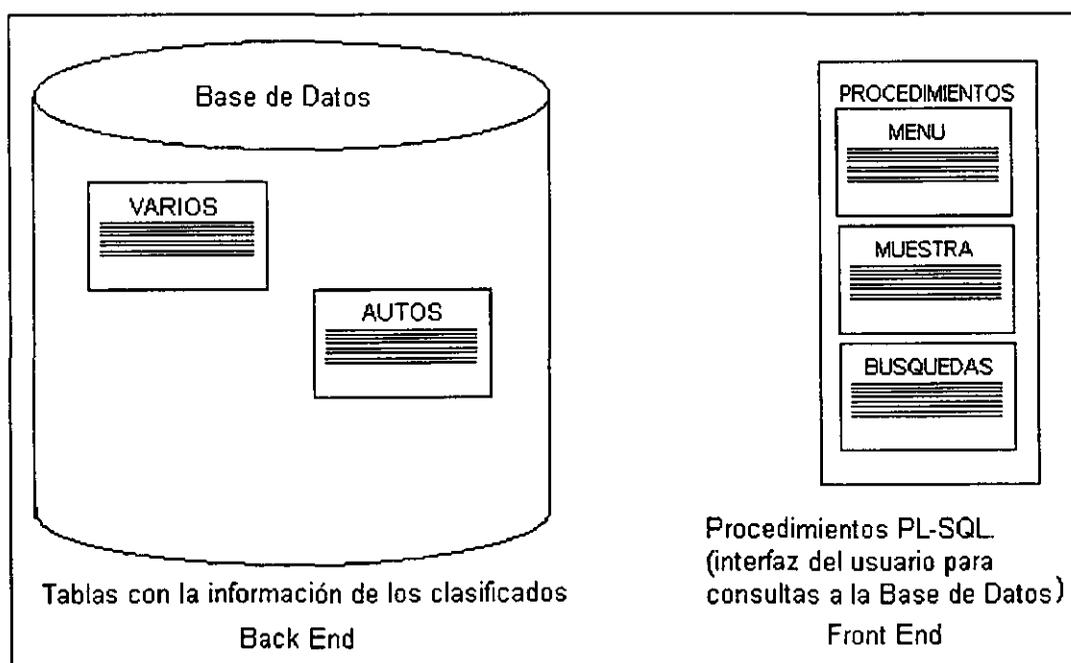


Figura 3.5.1 Esquema del Front End y Back End.

Entonces, la parte de la integración del sistema tiene como finalidad incorporar el diseño de la base de datos junto con la programación de los procedimientos PL-SQL, (Lenguaje Procedural - Lenguaje de Consultas Estructurado), que dan la facilidad de incorporar sentencias HTML y SQL en la misma programación.

Esta integración puede verse en la figura 3.5.2.

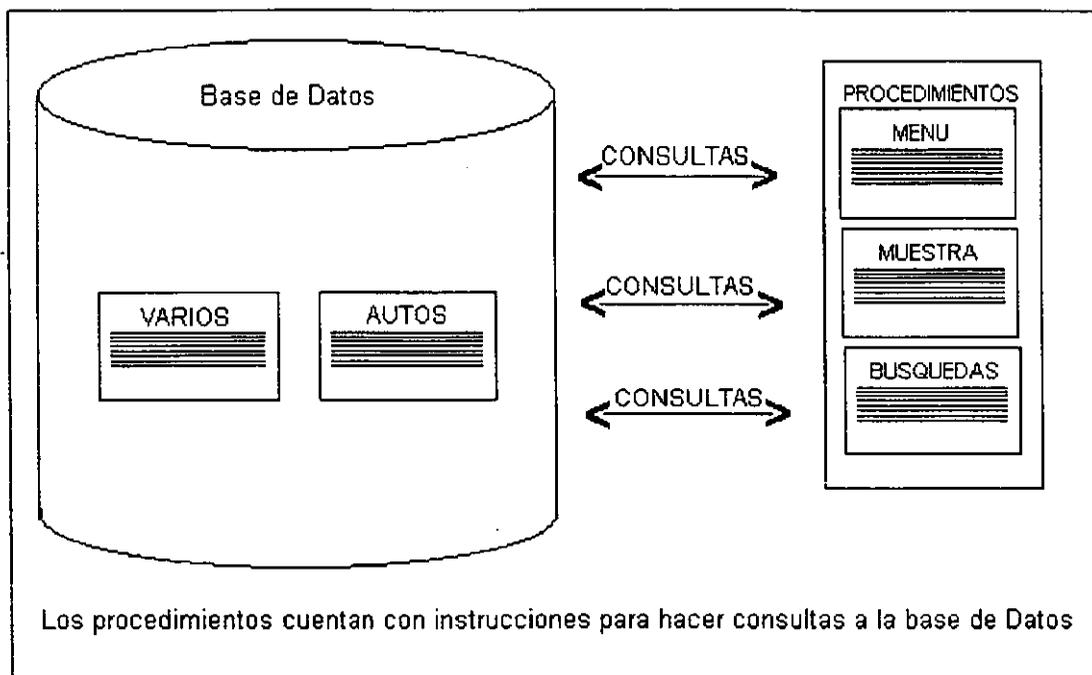


Figura 3.5.2 Integración de Procedimientos PL-SQL con la Base de Datos.

Para lograr esta integración se requieren compilar los programas PL-SQL (del Front End) para que éstos queden almacenados dentro de la base de datos y lograr con ello que las consultas a las tablas sean más rápidas y eficientes. La compilación se hace desde una sesión de SQL-Plus conectada con el usuario dueño de la información y ejecutando la siguiente instrucción:

```
SQL> @ nombre_programa.sql <enter>
```

Esta representación se muestra en la figura 3.5.3.

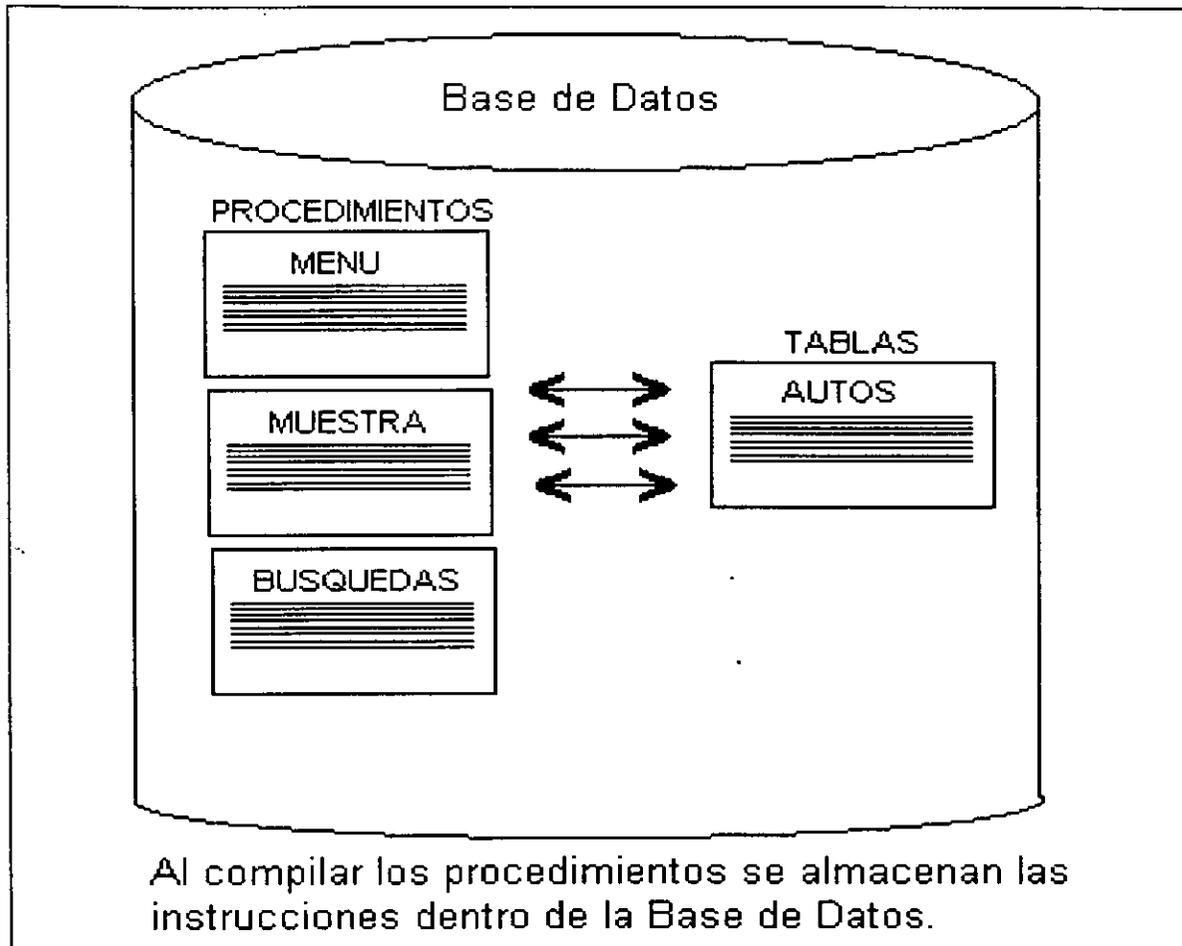


Figura 3.5.3 Procedimientos Almacenados en la Base de Datos.

Una vez que se tienen compilados cada uno de los ocho procedimientos correspondientes a las secciones de los clasificados, se procede a configurar el OWAS (Oracle Web Application Server), para mostrar el sistema a los usuarios de Internet. La parte fundamental de la configuración, consiste en especificar que el usuario de la base de datos (dueño de los datos y de las tablas), tendrá una conexión con el OWAS para desplegar la información en Internet. Esto se realiza usando la opción del OWAS conocida como: Data Base Access Descriptor (DAD).

Esta configuración puede verse en la figura 3.5.4.

También para llevar a cabo la fase de integración del sistema fue necesario la configuración de "Listeners" en el OWAS. En esta parte se declara un "listener" que es el puerto por el cual el OWAS va a estar recibiendo las peticiones de los usuarios de Internet para consultar el sistema.

Es importante decir que un Listener, soporta 358 conexiones simultáneas de los usuarios en la versión 3 del OWAS y más de 700 conexiones simultáneas en la versión 4 del OWAS. Por lo que es muy difícil que un Listener llegue a saturarse.

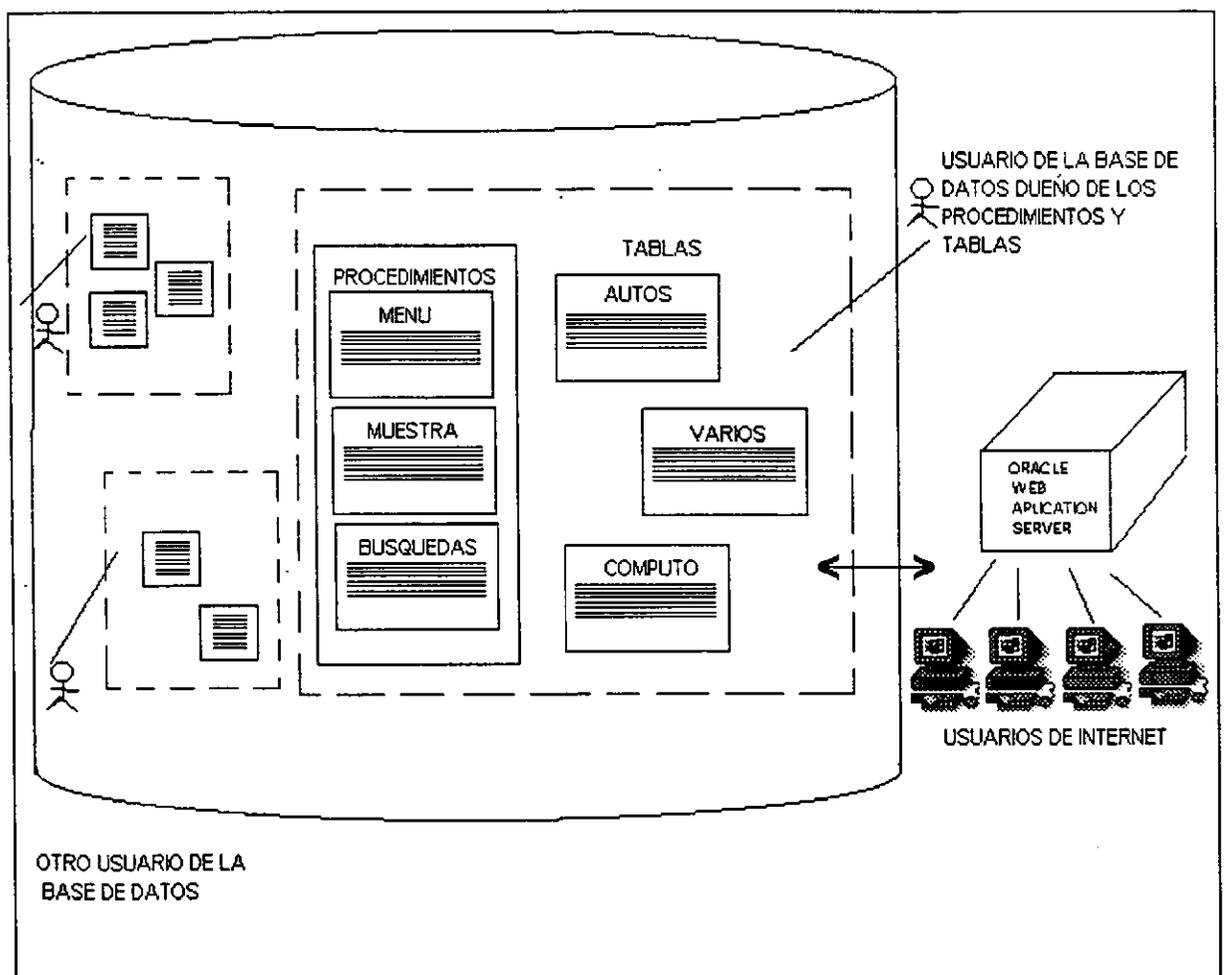


Figura 3.5.4 Configuración del Web Application Server para Desplegar los Procedimientos Almacenados en la Base de Datos.

Por otro lado, en esta parte se declararon las rutas virtuales que representarían a los directorios físicos en la computadora donde se almacenarán imágenes y páginas estáticas de HTML.

Depuración del Sistema.

Para la parte de depuración del sistema se contemplaron ambas partes importantes del sistema: el Front End y el Back End.

En la parte del Front End, es decir la programación PL/SQL con HTML se revisaron a fondo los programas, de forma que no existiera redundancia en el código de cada uno de los "paquetes" de cada una de las secciones. También se examinaron todas las referencias (ligas) hacia otros sitios de la página y hacia otros "paquetes", y conforme se iban corrigiendo estos detalles en los programas, se iban cambiando los nombres de éstos, para evitar errores de actualización en los programas, es decir, si el código del programa "vivienda01" tenía alguna modificación de corrección de errores, entonces se renombraba a "vivienda02" con el fin de evitar confusiones al momento de la compilación de ellos.

Respecto a la parte del Back End se depuró la redundancia de datos en las tablas, de tal forma que sólo se almacenarán los datos más importantes. También se depuró la declaración de las tablas con el fin de hacer más rápidas las consultas, lo que propició que las tablas estuvieran declaradas en memoria "cache" para facilitar el rápido acceso a los datos. Además se depuraron las consultas hechas por "CONTEXT" para hacer búsquedas de palabras, lo cual obligó a utilizar consultas en memoria para acceder más rápido a la información.

3.6 PRUEBAS DE RENDIMIENTO, RAPIDEZ Y OPERATIVIDAD DE LA PUBLICACIÓN EN INTERNET.

De acuerdo con los avances del sistema, se fueron realizando diversas pruebas para evitar posibles errores durante el fin de cada uno de los módulos y también al final cuando todos los módulos estaban terminados. Para hacer el análisis del sistema se contemplaron dos tipos de pruebas para examinar el sistema.

La primer estrategia de pruebas se le conoce como "Prueba Ascendente", que consiste en ir probando los módulos de programación desde los más pequeños hasta seguir con unidades más grandes de programación.

La segunda estrategia de pruebas recibe el nombre de "Prueba Descendente", y consiste en analizar los módulos de programación desde el nivel más alto, considerando que los módulos pequeños se encuentren vacíos.

La estrategia de prueba que más se adecuaba a las circunstancias del sistema era la "Prueba Ascendente", debido a que el sistema contaba con módulos de programación pequeños conocidos como "procedimientos" que a su vez estaban unidos por módulos de programación más grandes que reciben el nombre de "paquetes". Por lo tanto, las pruebas correspondientes de acuerdo con la estrategia de pruebas seleccionada, se realizaron inicialmente en cada uno de los procedimientos que conformaban cada Paquete. Recordemos que existen ocho paquetes que representan a cada una de las subclasificaciones que componen al sistema: autos, vivienda, empleos, permutas, varios, cómputo, negocios y varios.

La estrategia de Prueba Ascendente, se subdivide en cada una de las pruebas que se mencionan a continuación:

- Prueba Funcional.
- Prueba de Desempeño.

Prueba Funcional.

Esta prueba consiste en asegurar que el sistema ejecute cada una de las funciones determinadas de manera correcta. Para lo cual, se proponen diversas entradas a los módulos y se verifica que las salidas sean correctas, con el fin de evitar errores en estos módulos.

Se hicieron pruebas de funcionalidad en el proceso de carga de información a la base de datos y también pruebas de funcionalidad en la programación de cada uno de los módulos que se desarrollaron para el sistema. Iniciaremos explicando las pruebas que se realizaron para lograr la integración del archivo de texto para almacenar los clasificados en la base de datos.

Pruebas en la Carga de Datos.

Inicialmente las pruebas para cargar los datos a las tablas a partir del archivo de texto dado (texto.txt), fueron el formateo de los datos, ya que estos venían con códigos que se utilizaban para publicar en medios impresos. Por ejemplo, el siguiente fragmento muestra el contenido del archivo original donde se proporcionaba la información.

```
</GALLEY></clshd,0100,380></CLSHD1>>1>  
>A>>L>>Q>>.>> >>P>>E>>R>>M>>U>>T>>A>>S>> >>D>>E>  
>T>>I>>E>>M>>P>>O>>S>> >>C>>O>>M>>P>>A>>R>>T>>I>>D>>O>>S>  
>Y>> >>V>>I>>A>>J>>E>>S>  
8  
</clsad,0100,0,0,77297></CLSI><cl0>  
8  
</clsad,0100,0,1,3617803></CLSI><cl0>  
8  
</clscd,9656513,C9656513,K04,L75,,2>8  
</clscd,9309698,C9309698,K04,L70,,3>8  
</GALLEY></clshd,0101B,100></CLSHD2><QJ>  
  
>H>>o>>t>>e>>I>>e>>s>  
8
```

</clsad,0101B,420,0,9724649></CLSI>ACAPULCO, semana de s£per lujo del 23 <QJ>
al 29 de enero, Hotel Acapulco Plaza, <QJ>
dos habitaciones (para 4 a 8 perso-<QJ>
nas) acceso al Club Privado Oasis. <QJ>
800 d¢lares, total por las 7 noches. <QJ>
Tel,fono (01-83) 00-55-38, Monterrey <QJ>
N. L.

8

</clsad,0101B,180,1,9727748></CLSI>DE puente Huatulco Zaashila. Suites lujo, <QJ>
alberca integrada. 633-40-19, <QJ>
670-15-33.

8

</clsad,0101B,240,2,9726150></CLSI>ESTRESADO, PLAYA VIRGEN TUXPAN, <QJ>
VER. 8 CUARTOS, TODOS LOS SERVI-<QJ>
CIOS, LAGUNA, ALBERCA, MAR. <QJ>
678-07-81.

8

</clsad,0101B,300,3,9719546></CLSI>GANGA. SUITE EN HOTEL 5 ESTRELLAS
<QJ>
PARA EL DA EN POCA QUE USTED <QJ>
ELIJA EN PUERTO VALLARTA A PRECIO <QJ>
DE 1994. 515-52-01, LUNES A VIER-<QJ>
NES, HORAS HABLES.

Para eliminar los caracteres sin sentido para la carga de la información a la base de datos, se empleó el programa: "formato.cpp", el cual puede verse en el apéndice 2. El resultado al ejecutar este programa se muestra a continuación.

</clsad,0101B,9724649></CLSI>ACAPULCO, semana de s£per lujo del 23 al 29 de enero, Hotel Acapulco Plaza, dos habitaciones (para 4 a 8 personas) acceso al Club Privado Oasis. 800 d¢lares, total por las 7 noches. Tel,fono (01-83) 00-55-38, Monterrey N. L.@

</clsad,0101B,9727748></CLSI>DE puente Huatulco Zaashila. Suites lujo, alberca integrada. 633-40-19, 670-15-33.@

</clsad,0101B,9726150></CLSI>ESTRESADO, PLAYA VIRGEN TUXPAN, VER. 8 CUARTOS, TODOS LOS SERVICIOS, LAGUNA, ALBERCA, MAR. 678-07-81.@

</clsad,0101B,9719546></CLSI>GANGA. SUITE EN HOTEL 5 ESTRELLAS PARA EL DA EN POCA QUE USTED ELIJA EN PUERTO VALLARTA A PRECIO DE 1994. 515-52-01, LUNES A VIERNES, HORAS HABLES.@

DESARROLLO E IMPLEMENTACIÓN DEL PROBLEMA Pruebas de Rendimiento, Rapidez y Operación

De esta parte, nos interesa la clasificación del anuncio, por ejemplo 0101B, y el texto del contenido para almacenar los registros en la Base de Datos. Posteriormente se aplicó un segundo programa llamado: "corrige.cpp", que sustituía acentos en la forma adecuada, ya que los acentos eran representados por otros símbolos, por ejemplo en el primer registro la palabra "s£per" al pasar por este programa se sustituía por "súper". El código de este programa puede verse en el apéndice 2, y el resultado al ejecutar este segundo programa es el siguiente:

```
</clsad,0101B,9724649></CLSI>ACAPULCO, semana de súper lujo del 23 al 29 de enero, Hotel Acapulco Plaza, dos habitaciones (para 4 a 8 personas) acceso al Club Privado Oasis. 800 dólares, total por las 7 noches. Teléfono (01-83) 00-55-38, Monterrey N. L.@
```

```
</clsad,0101B,9727748></CLSI>DE puente Huatulco Zaashila. Suites lujo, alberca integrada. 633-40-19, 670-15-33.@
```

```
</clsad,0101B,9726150></CLSI>ESTRESADO, PLAYA VIRGEN TUXPAN, VER. 8 CUARTOS, TODOS LOS SERVICIOS, LAGUNA, ALBERCA, MAR. 678-07-81.@
```

```
</clsad,0101B,9719546></CLSI>GANGA. SUITE EN HOTEL 5 ESTRELLAS PARA EL DA EN POCA QUE USTED ELIJA EN PUERTO VALLARTA A PRECIO DE 1994. 515-52-01, LUNES A VIERNES, HORAS HABILES.@
```

Una vez que se tienen los datos formateados en este esquema, se comprueba que los registros queden totalmente insertados en las tablas correspondientes de la base de datos. La forma de verificar si todos los registros quedaron dentro de la base de datos, es observando la pantalla del programa SQL-Loader (herramienta para introducir datos a la base), que se visualiza a la hora de ejecutar el proceso automático de carga de datos.

Recordamos que con la herramienta Oracle Enterprise Manager, que automatiza el proceso de carga de datos, es en donde se utiliza el programa SQL-Loader para realizar la carga de datos que previamente han sido formateados. En este momento es donde se especifica que el archivo "Salida1.txt" hará uso del programa SQL-loader para subir los datos de este archivo a la tabla "temporal" previamente definida.

En esta parte, el administrador del sistema observa en pantalla los mensajes del SQL-Loader para verificar si hubo errores o no en la carga de datos. En caso de no haber errores el proceso continúa por las fases que se definieron. En caso contrario, el programa muestra cuántos registros no pudieron cargarse a la base e indica los motivos por los cuales no pudo hacerlo, y los deja en un archivo definido como "errores.log".

A partir de este archivo, el administrador puede revisar cuál fue el error en los registros, por ejemplo, que falte un campo en ese registro o algún otro motivo. Una vez que se corrigió el error el administrador puede usar nuevamente el programa SQL-loader para cargar sólo los datos que faltaron y lo puede hacer desde la línea de comandos tecleando la siguiente instrucción:

```
C:\> sqlldr80 userid=usuario/password control=carga.ctl data=salida1.txt
```

donde:

- userid, es el usuario dueño de la información con su respectivo password.
- data, toma los datos ya formateados que se encuentran en el archivo salida1.txt.
- control, es el archivo donde se define las características de la tabla en donde se almacenarán los datos, por ejemplo, este código es parte del archivo "carga.ctl" :

```
LOAD DATA  
REPLACE  
INTO TABLE TEMPORAL  
(ID SEQUENCE(1,1),  
TIPO POSITION(9:10),  
SUBTIPO POSITION(11:13),  
IMAGEN POSITION(15:21),  
TEXTO POSITION(30) CHAR(4500) TERMINATED BY "@" )
```

Observamos que la tabla temporal está formada por cinco campos, el primero es un ID que se va llenar con una secuencia empezando desde 1; el campo TIPO va tomar los datos de la posición 9 a la 10 del archivo de datos, el campo SUBTIPO tomará la

información de la posición 11 a la 13, el campo IMAGEN de la posición 15 a la 21, y el campo TEXTO de la posición 30 hasta encontrar el símbolo: "@".

Ya que los registros quedaron almacenados en la tabla temporal, se tiene que ejecutar nuevamente el programa que distribuye los clasificados a las tablas correspondientes, según el número de clasificación. El programa que realiza esta función se llama: "l_t.sql". El código de este programa puede verse en el apéndice 2.

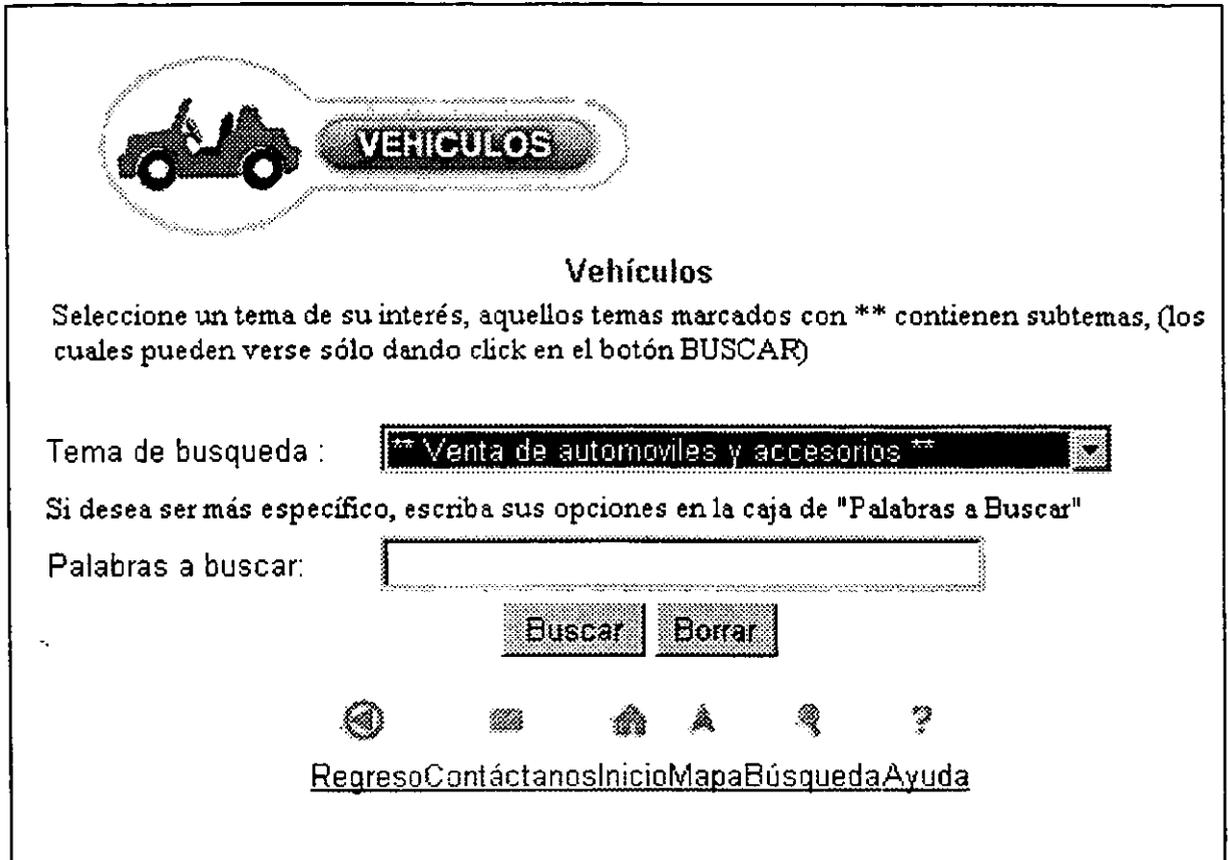
Por otro lado, las pruebas correspondientes a los módulos de programación del Front End, se dividen en los siguientes apartados:

- Procedimiento de captura de opciones del usuario.
- Procedimiento de validación de opciones.
- Procedimiento de búsqueda.

Procedimiento de Captura de Opciones del Usuario.

En esta etapa de pruebas se verificaron algunas palabras probables que un usuario escribiría cuando tratara de indicar sus opciones de búsqueda. En esta etapa se le permite al usuario realizar consultas utilizando operadores lógicos (por ejemplo OR, AND, comodín*) para facilitar su búsqueda. Las pruebas en esta etapa consistieron en introducir varios ejemplos utilizando palabras junto con los operadores lógicos. Cabe hacer la aclaración que el Cartucho de Context utiliza ciertos caracteres especiales como %, | &, que causan problemas si el usuario solo escribe estos caracteres; pero es poco probable que un usuario haga una búsqueda con estos símbolos. Previendo esta situación, se configura en el Web Application Server una opción para enviar una página de error cuando suceda este caso.

Las dos posibles opciones de búsqueda para un usuario son por palabras o seleccionando un tema en especial, como se muestra en la figura 3.6.1.



The screenshot shows a search interface for 'VEHICULOS'. At the top left is a logo with a car icon and the word 'VEHICULOS' in a rounded rectangle. Below this is the title 'Vehículos'. A text instruction reads: 'Seleccione un tema de su interés, aquellos temas marcados con ** contienen subtemas, (los cuales pueden verse sólo dando click en el botón BUSCAR)'. There is a dropdown menu for 'Tema de búsqueda' with the selected option '*** Venta de automoviles y accesorios ***'. Below that is a text input field for 'Palabras a buscar' with the placeholder text 'Si desea ser más específico, escriba sus opciones en la caja de "Palabras a Buscar"'. At the bottom of the search area are two buttons: 'Buscar' and 'Borrar'. Below the buttons are six navigation icons: a left arrow, a square, a house, an upward arrow, a magnifying glass, and a question mark. At the very bottom are six underlined text links: 'Regreso', 'Contáctanos', 'Inicio', 'Mapa', 'Búsqueda', and 'Ayuda'.

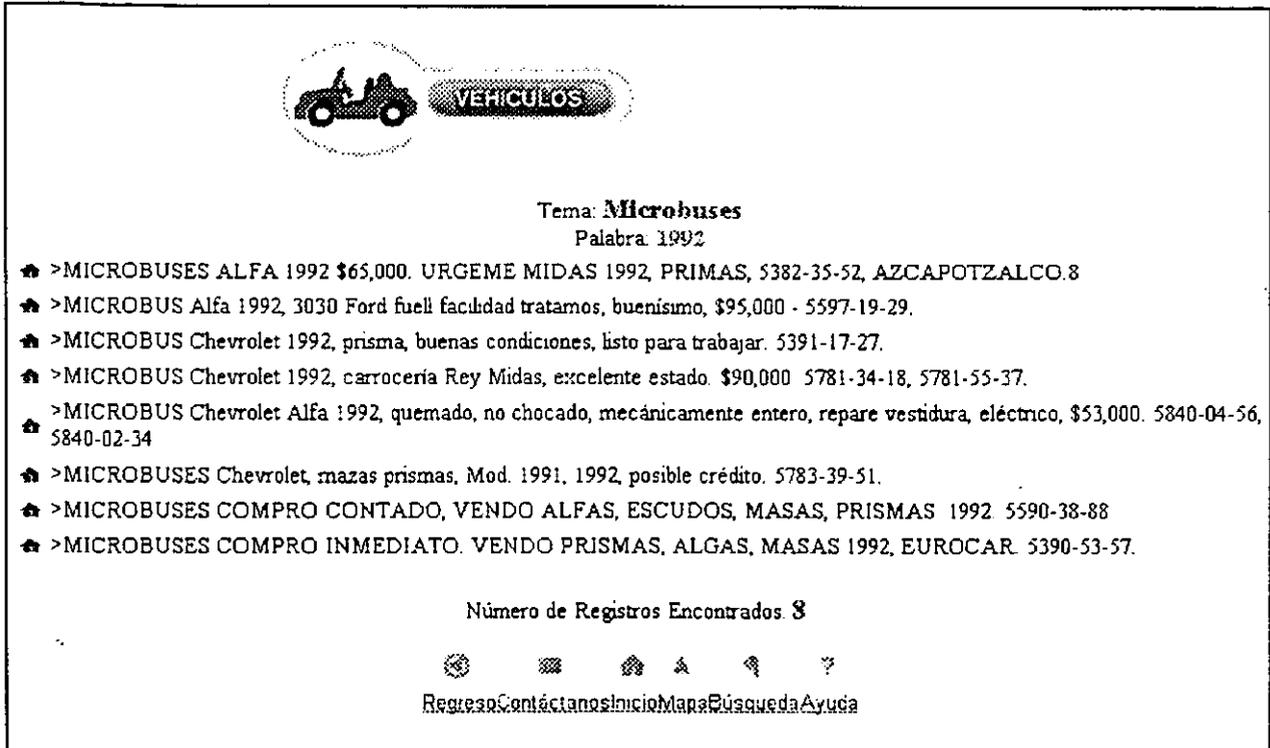
Figura 3.6.1 Forma de Captura.

Procedimiento de Validación de Opciones.

En esta etapa, las pruebas consistieron en verificar si el usuario seleccionó alguno de los temas mostrados, o bien, seleccionó búsqueda por palabras, para enviar sus peticiones al siguiente nivel de búsqueda o mostrar sus resultados en pantalla.

Procedimiento de Búsqueda.

En este procedimiento las pruebas se realizaron respecto a la forma en que el usuario vería sus resultados en la pantalla, y además de probar las formas de navegar en esa página de despliegue de resultados. La figura 3.6.2 muestra los resultados de una búsqueda y las opciones de navegación en esa pantalla de despliegue de resultados.



 **VEHICULOS**

Tema: Microbuses
Palabra: 1992

- ♣ >MICROBUSES ALFA 1992 \$65,000. URGEME MIDAS 1992, PRIMAS, 5382-35-52, AZCAPOTZALCO. 8
- ♣ >MICROBUS Alfa 1992, 3030 Ford fueli facilidad tratamos, buenísimo, \$95,000 - 5597-19-29.
- ♣ >MICROBUS Chevrolet 1992, prisma, buenas condiciones, listo para trabajar. 5391-17-27.
- ♣ >MICROBUS Chevrolet 1992, carrocería Rey Midas, excelente estado. \$90,000 5781-34-18, 5781-55-37.
- ♣ >MICROBUS Chevrolet Alfa 1992, quemado, no chocado, mecánicamente entero, repare vestidura, eléctrico, \$53,000. 5840-04-56, 5840-02-34
- ♣ >MICROBUSES Chevrolet, mazas prismas, Mod. 1991, 1992, posible crédito. 5783-39-51.
- ♣ >MICROBUSES COMPRO CONTADO, VENDO ALFAS, ESCUDOS, MASAS, PRISMAS 1992 5590-38-88
- ♣ >MICROBUSES COMPRO INMEDIATO. VENDO PRISMAS, ALGAS, MASAS 1992, EUROCAR. 5390-53-57.

Número de Registros Encontrados: 8



[Regreso](#) [Contáctanos](#) [Inicio](#) [Mapa](#) [Búsqueda](#) [Ayuda](#)

Figura 3.6.2 Forma de Despliegue.

Las opciones de navegación al pie de la pagina, se probaron en su totalidad para llamar al procedimiento original que las mando a ejecutar. Además de probar que todas las "ligas" de esa pagina estuvieran apuntando al lugar adecuado.

Prueba de Recuperación.

Con esta prueba se pretende evitar cualquier circunstancia que pueda alterar la correcta ejecución del sistema. Por ejemplo, la pérdida de la energía eléctrica en las instalaciones donde se encuentre el servidor de la base de datos y el servidor de Web; y para simular este caso se apagó intencionalmente el servidor que contenía el sistema para poder conocer cuál iba a ser el estado de la Base de Datos. Se observó que el sistema mostraba consistencia en la base de datos, ya que las transacciones en el sistema sólo realizan consultas a la base de datos.

Prueba de Desempeño.

Debido a la importancia del sistema de ofrecer información de los clasificados a los usuarios vía Internet, esta prueba es fundamental para dar una buena imagen a los usuarios, ya que trata de reflejar el buen planteamiento y dimensionamiento del sistema manejando gran cantidad de información y un gran número de peticiones de los usuarios, además de que el tiempo de respuesta del sistema debe ser el menor posible.

En esta parte es donde se realiza una mayor serie de pruebas de rendimiento con el fin de evitar la saturación de conexiones de los usuarios y además evitar la escasez de memoria por parte del servidor establecido para este proyecto. En principio el Web Application Server tiene la capacidad de soportar 338 conexiones simultáneas de usuarios por cada listener, recordemos que un listener puede estar definido por un puerto que puede tener un valor desde el puerto 1024 hasta 64000 aproximadamente, es decir, pueden definirse tantos listeners como soporten los recursos del servidor donde estemos trabajando.

Además, en caso de no ser suficiente con este valor de 338 conexiones simultáneas de usuarios, se pueden "redirigir" las peticiones de los usuarios a otros listeners previamente configurados para atender la misma aplicación.

De acuerdo a las versiones de los encargados del sistema, las conexiones de los usuarios simultáneamente no sobrepasarían este valor. Por lo cual el sistema quedó configurado para utilizar un solo listener que está soportando un total máximo de 338 conexiones simultáneas de usuarios.

Por otro lado, la prueba de consumo de memoria con cada conexión de usuario, si fue preciso llevarla a la práctica, es decir, se tomaron lecturas de consumo de memoria de las siguientes actividades:

DESARROLLO E IMPLEMENTACIÓN DEL PROBLEMA Pruebas de Rendimiento, Rapidez y Operación

- Memoria para levantar los servicios de la Base de Datos.
- Memoria para levantar los servicios del Web Application Server.
- Memoria para levantar los servicios del cartucho de Context.
- Memoria por cada conexión a la página Web y hacer uso de Context.

Se tomaron lecturas en el servidor para tomar un estimado de recursos en memoria, con el fin de evitar el uso de **SWAP** en memoria (proceso de utilizar el disco duro cuando la memoria física se encuentra saturada) que retrasaría mucho la atención a los usuarios. Las lecturas arrojaron los siguientes resultados:

- Memoria para levantar los servicios de la Base de Datos = 20 MegaBytes (aprox).
- Memoria para levantar los servicios del Web Application Server = 18 MegaBytes (aprox).
- Memoria para levantar los servicios del cartucho de Context = 5 MegaBytes (aprox).
- Memoria por cada conexión a la página y hacer uso de Context = 0.8 MegaBytes (aprox).

En este punto es necesario informar que si un usuario hace la petición de búsqueda de palabras utilizando Context en la página Web, el tiempo que el sistema tarda en liberar esa memoria es de aproximadamente 30 segundos, es decir que si el usuario está utilizando la búsqueda de palabras empleando Context puede seguir haciéndolo las veces que quiera y no se incrementará el consumo de memoria para ese usuario, hasta que el usuario tarde 30 segundos aproximadamente, el sistema liberará la memoria que él estaba utilizando.

También es importante señalar que teniendo conectados alrededor de 300 usuarios concurrentes y levantando los servicios de: la Base de Datos, el cartucho de Context y el Web Application Server, el sistema estaría consumiendo aproximadamente 263 MegaBytes de memoria **RAM** (memoria de acceso aleatorio).

Mantenimiento.

La fase de mantenimiento se centra en el cambio que va asociado a la corrección de errores, a las adaptaciones requeridas por la evolución del entorno del software y las modificaciones debidas a los cambios de los requisitos del cliente dirigidos a reforzar o ampliar el sistema. Durante la fase de mantenimiento se encuentran tres tipos de cambios:

Mantenimiento Correctivo.

Incluso llevando a cabo las mejores actividades de garantía de calidad, es muy probable que el cliente descubra defectos en los programas. El mantenimiento correctivo trata de modificar el software para corregir esos defectos, o bien, para modificar el sistema cuando se llegara a detectar cualquier situación anormal. Si consideramos que el sistema esta clasificado como un sistema de alta disponibilidad, entonces el tiempo que llegase a estar fuera de línea, ocasionaría retrasos en la información de los clasificados de los usuarios. Obviamente el sistema no escapa a este tipo de problemas que por alguna circunstancia se pudieran presentar. Tal es el caso, por ejemplo, de la saturación del disco duro del servidor de Web. Este problema se presentó cuando se generaban los archivos "log" de las conexiones hechas a la página Web del sistema. Este archivo va creciendo notablemente en el transcurso de los días, y por consecuencia el disco duro del servidor se saturó y detuvo momentáneamente al sistema de Internet. El problema se detectó cuando se puso a operar el sistema en línea. Los numerosos accesos de los usuarios, provocaron esta falla en el sistema.

Se tuvieron aproximadamente 720,000 accesos durante el mes de marzo y éste fue el motivo que provocó la falla en el sistema. Afortunadamente se contaba con la disposición del personal en ese momentos para direccionar la ruta que tenía este archivo para almacenarlo en otro disco duro.

Mantenimiento Adaptativo.

Con el paso del tiempo es probable que cambie el entorno original, por ejemplo el sistema operativo o algún periférico, para el que se desarrolló el sistema. El mantenimiento adaptativo consiste en modificar el software para acomodarlo a los cambios de su entorno externo. Puede darse el caso de tener que migrar el sistema a otra plataforma más robusta, como por ejemplo Unix. Para estos casos tanto el diseño del Back-end como el diseño del Front-end no sufren modificaciones en sus códigos. Por lo que solamente hay que preocuparse por instalar adecuadamente los productos en la plataforma que se haya seleccionado.

Mantenimiento Perfectivo.

Conforme se utilice el software, el cliente o el usuario pueden descubrir funciones adicionales que podrían interesar para integrarse al sistema. El mantenimiento Perfectivo amplía el software más allá de su funcionalidad original. Tal es el caso de que se quiera cambiar el formato de las imágenes por algún formato con movimientos o algún otro tipo de formato, entonces para integrar estas nuevas correcciones se tienen que editar los programas fuente (del back-end o front-end) para integrar estas nuevas mejoras; por lo que es necesario documentar bien el código fuente de estos programas para que estas integraciones sean más rápidas y sencillas (ver anexo 3).

Mantenimiento Preventivo.

Este tipo de mantenimiento se refiere a contemplar la mayoría de los aspectos para prever situaciones que puedan alterar el comportamiento del sistema. En este punto, la programación SQL por módulos y además con comentarios, hace más flexible el detectar errores en el sistema o atacar más rápidamente el problema cuando éste llegue a ocurrir.

MANUAL TÉCNICO.

Este apéndice recopila el manual técnico para el Sistema de Consulta de Productos y Servicios vía Internet diseñado para una Agencia de Publicidad.

Introducción.

El objetivo de este sistema es el de facilitar el proceso de consulta de avisos clasificados manejados por una agencia de publicidad de una manera más rápida y eficaz aprovechando los alcances de Internet.

Este manual permitirá al administrador del sistema conocer los procedimientos y actividades para mantener el sistema funcionando y disponible para los usuarios finales, así como la estructura de sus componentes para poder resolver los problemas que llegaran a presentarse.

Conocimientos Requeridos.

El administrador del sistema deberá contar con los siguientes conocimientos básicos:

- Administración del sistema operativo Windows NT.
- Administración de Bases de Datos.
- Administración de Web Application Server.
- Familiarizado con conceptos de generación de aplicaciones utilizando Oracle 8 (Enterprise Edition) que involucra: Context cartridge, Enterprise Manager y PLSQL.

Software Necesario Para la Ejecución de la Aplicación.

- Navegador de Internet (como por ejemplo "Internet Explorer" o "Netscape").

- Web Application Server. Servidor de aplicaciones Web que permite la generación de páginas HTML a partir de los objetos de la base de datos, las cuales son presentadas al usuario por medio del navegador.
- Base de Datos. Repositorio de los objetos e información que conforman la aplicación mantenida por el manejador de la base de datos Oracle.

Servicios y Configuraciones.

La Base de Datos y el Web Application Server están instalados en una máquina con Windows NT por lo que se explicará a continuación de manera breve la forma de iniciar y detener, tanto los servicios de la base de datos, como el Web Application Server.

Servicios.

Para el sistema operativo Windows NT se requieren levantar procesos de multitarea, para dar de alta una base de datos Oracle, estos procesos son comúnmente conocidos como servicios. Los servicios necesarios para que la aplicación pueda trabajar son los siguientes:

- OracleServiceORCL.
- OracleStartORCL.
- OracleTNSListener80.
- OracleWWWListener30ADMIN.
- OracleWRBPrimaryService.
- OracleWWWListener30www.

Los servicios anteriores son iniciados desde el Panel de Control después de seleccionar el ícono "Services". Su descripción se indica a continuación:

- OracleServiceORCL: Este servicio habilita los procesos de **background** (fondo) de la base de datos.
- OracleStartORCL: Permite iniciar la base de datos en forma automática.
- OracleTNSListener80: Este servicio inicia un proceso que atiende conexiones de SQL*Net Versión 2.
- OracleWWWListener30ADMIN: Permite la administración del Web Application Server.
- OracleWRBPrimaryService: Permite iniciar un proceso que atiende conexiones a la aplicación Web Request Broker.
- OracleWWWListener30www: Este servicio habilita el funcionamiento del Web Application Server.

Los servicios anteriores pueden ser iniciados y terminados de dos maneras, las cuales a continuación se describen:

1. Desde el Panel de Control de Windows NT, a través del ícono "Services" se procederá a seleccionar el servicio; por ejemplo: OracleServiceORCL y se presionará el botón "Start". Si se quiere detener el servicio, se selecciona el servicio de interés, digamos OracleStartORCL y posteriormente presionar el botón "Stop".
2. Desde el prompt de MS-DOS a través de los comandos que se describen a continuación:

Iniciar Servicios de Base de Datos:

Teclear desde la línea de comandos lo siguiente:

```
C:\> svrmgr30 <enter>
```

mostrará el prompt de Server Manager (svrmgr) y se deberá teclear la conexión como administrador:

```
svrmgr> connect internal <enter>  
password> oracle <enter>
```

teclear el siguiente comando para levantar los servicios de base de datos:

```
svrmgr> startup <enter>
```

Con esto quedarán activos los servicios OracleServiceORCL y OracleStartORCL.

Detener Servicios de la Base de Datos.

Para detener el servicio se teleará el comando:

```
Svrmgr> shutdown <enter>
```

Y se detendrán los servicios OracleServiceORCL y OracleStartORCL

Para salir del Server Manager, telear lo siguiente:

```
svrmgr> exit <enter>
```

Levantar el “listener” de la Base de Datos.

En la línea de comandos del sistema, escribir lo siguiente:

```
C:\> lsnrctl80 <enter>
```

Y telear posteriormente:

```
lsnrctl> start <enter>
```

Con esto se activará el servicio OracleTNSListener80.

Detener el "Listener".

Para detener el "Listener" de la Base de Datos (TNSListener80), teclear:

```
Isnrctl> stop <enter>
```

Para salir del prompt, escribir:

```
Isnrctl> exit <enter>
```

Levantar los Servicios de Web Application Server.

Primero se levantan los servicios del Web Request Broker (despachador de peticiones del Web) de la siguiente forma:

```
C:\> owscctl start wrb <enter>
```

Posteriormente se levanta el listener del admin del Web Server:

```
C:\> owscctl start admin <enter>
```

Por último se inicializa el listener que contiene la aplicación del sistema:

```
C:\> owscctl start www <enter>
```

Con estos comandos se activan los servicios OracleWRBPrimaryService, OracleWWWListener30Admin y OracleWWWListener30www.

Detener los Servicios del Web Application Server.

Para detener los servicios del Web Application Server se darán los siguientes comandos, respetando el orden:

```
C:\> owscctl stop www
```

```
C:\> owscctl stop admin
```

```
C:\> owscctl stop wrb
```

Levantar Servicios de ConText.

Al igual que los otros servicios, se pueden inicializar los servicios de Context de dos formas; pero es recomendable en este caso, hacerlo desde la línea de comandos con el fin de que puedan elegirse las opciones correctas. Para ello, se teclea el siguiente comando:

```
C:\> ctxsrv80 -user ctxsys/ctxsys -personality QDM <enter>
```

Con esta única instrucción se asegura que el servicio para escuchar peticiones de búsqueda con el cartucho de Context queda satisfecha. Debe leerse el mensaje de "server startup completed" para asegurarse que se encuentra activo el servidor.

Nota importante: La ventana de MS-DOS no debe cerrarse (se puede minimizar) o dar "control-c" al servicio, ya que con esto, el servicio quedará eliminado y no se podrán escuchar peticiones de Context.

Configuración de Context.

Se crean dos "tablespaces" en SQL para el uso de este cartucho, a través de los comandos:

```
SQL> create tablespace context
        datafile 'd:\orant\database\context.ora' size 20M;
```

```
SQL> create tablespace cxttmp
        datafile 'd:\orant\database\cxttmp.ora' size 10M;
```

Se ejecutan los siguientes scripts:

```
/* crea los usuarios y roles para Context */
```

```
SQL> connect sys/change_on_install
```

```
SQL> @ d:\orant\ctx80\admin\drctrsyst ctxsys passwdctxsys context cxttmp
```

```
(donde: ctxsys      "usuario administrador de context"
        passwdctxsys "password del ctxsys"
        context     "nombre del tablespace"
        ctxsys      "nombre del tablespace temporal" )
```

```
/* crea los objetos para usar Context */
```

```
SQL> connect ctxsys/ctxsys
```

```
SQL> @ d:\orant\ctx80\admin\dr0inst
```

Verificar parámetros de la Base de Datos, en el archivo "initORCL.ora".

Para un buen funcionamiento del Cartucho de Context se recomiendan los siguientes valores:

```
share_pool_size = 30 000 000
process = 80
```

```
open_cursors = 250
```

```
text_enable = TRUE
```

(en caso de modificar alguno de los parámetros anteriores se debe tirar y levantar nuevamente la base de datos).

Creación del Usuario de la Base de Datos que Almacenará el Prototipo.

```
/* Crear tablespace's para tablas del prototipo */
```

```
SQL> connect system/manager
```

```
SQL> create tablespace noticias
```

```
    datafile 'D:\ORANT\DATABASE\NOTIORCL.ORA' size 50M;
```

```
SQL> create tablespace notitmp
```

```
    datafile 'D:\ORANT\DATABASE\NOTITMP.ORA' size 15M;
```

```
/* Verifica que estén creados los nuevos tablespaces */
```

```
SQL> select file_name,tablespace_name,bytes from dba_data_files;
```

```
/* Crea el usuario de Base de Datos */
```

```
SQL> create user noti identified by noti
```

```
    default tablespace noticias
```

```
    temporary tablespace notitmp;
```

```
SQL> alter user noti quota unlimited on noticias;
```

```
SQL> grant connect,resource,dba to noti;
```

```
/* Verifica que esté creado el nuevo Usuario de la Base de Datos */
```

```
SQL> select username from dba_users;
```

```
/* Otorga privilegios al usuario para utilizar Context */
```

```
SQL> connect ctxsys/ctxsys
```

```
SQL> grant ctxadmin to noti;
```

```
SQL> grant execute on ctx_query to noti;
```

Configuración del Web Application Server.

Para la parte de configuración del Web Application Server, se instala una nueva "Aplicación" en la parte de OAS Manager con los siguientes directorios mapeados:

Directorio Físico	Directorio Virtual
C:\universalsite	/
C:\universalsite	/uni/

Además hay que mapear el siguiente PATH en: Website => Applications => universal => configuración => virtual.

```
PATH nuevo: /noti/plsql/
```

Cambiar en la Base de Datos el Set de Caracteres a Español.

```
SQL> connect sys/change_on_install
```

```
SQL> update sys.props$ set value$ = 'SPANISH'
      where name = 'NLS_LANGUAGE';
```

```
SQL> update sys.props$ set value$ = 'MEXICO'
      where name = 'NLS_TERRITORY';
```

/* Verificar si los nuevos valores se actualizaron */

```
SQL> select * from NLS_DATABASE_PARAMETERS;
```

Creación de las Tablas y Objetos en la Base de Datos.

```
SQL> connect noti/noti
```

```
SQL> @ c:\archivo\crea_base.sql
```

Nota: Pueden verse los programas en el anexo 2.

Insertar algunos valores fijos del Catálogo en las tablas.

```
SQL> conect noti/noti
```

```
SQL> @ c:\archivo\tipo.sql
```

```
SQL> @ c:\archivo\subtipo.sql
```

```
SQL> @ c:\archivo\submarca.sql
```

```
SQL> @ c:\archivo\falta_cat.sql
```

Insertar datos reales ya formateados en la tabla temporal.

Desde el prompt de MS-DOS ejecutar la siguiente línea:

```
C:\ sqlldr80      userid=noti/noti      control=carga.ctl      data=salida1.txt  
bindsize=20000000
```

Almacenar datos en las tablas correspondientes.

```
SQL> connect noti/noti
```

```
SQL> @ c:\archivo\l_t.sql
```

```
SQL> @ c:\archivol\cata_hoy.sql
```

Ejecutar los Scripts Para Desplegar en WEB.

```
SQL> connect noti/noti
SQL> @ c:\activos\web092_autos.sql
SQL> @ c:\activos\web092_computo.sql
SQL> @ c:\activos\web092_empleado.sql
SQL> @ c:\activos\web092_negocios.sql
SQL> @ c:\activos\web092_permutas.sql
SQL> @ c:\activos\web092_servicios.sql
SQL> @ c:\activos\web092_varios.sql
SQL> @ c:\activos\web092_vivenda.sql
```

Actualizaciones, RespalDOS y Depuración de la Base de Datos.

Estructura Física de la Base de Datos.

Con el fin de entender algunas definiciones de respaldo y recuperación de la base de datos es necesario mencionar la estructura física de una base de datos Oracle.

Control File (Archivo de Control).

Cada base de datos Oracle posee un *control file*. Un *control file* es un archivo de datos extremadamente importante que contiene las especificaciones de la estructura física de la base de datos, y provee información importante al momento de hacer una recuperación de la base de datos. A continuación se presenta un ejemplo del tipo de información que contiene un control file:

- Nombre de la base de datos.

- Nombre y localización de los *datafiles* y *redo log files*.
- Momento de creación de la base de datos.
- Información del respaldo.

EL uso del *control file* se puede observar cuando se monta un servidor de una base de datos Oracle, estos control file son usados para identificar los *datafile* y *redo log files* que deben de activarse para que la base de datos entre en operación.

Oline Redo Log Files.

Cada base de datos posee un conjunto de 2 o más *redo log files*. El conjunto de *redo log files* de una base de datos es comúnmente conocido como el redo log de la base de datos. Oracle utiliza el redo log para registrar todos los cambios realizados en la base de datos.

Datafiles.

Cada base de datos posee uno o más datafiles físicos. Un datafile contiene todos los datos almacenados en la base de datos. En los datafiles asociados con una base de datos se almacenan dos tipos de información: los datos del usuario y los datos del sistema:

- *Los datos del usuario* son los datos de la aplicación, junto con toda la información relativa a las aplicaciones.
- *Los datos del sistema* constituyen la información que la base de datos necesita para gestionar los datos del usuario y así misma. Por ejemplo, con los datos del sistema, Oracle sabe que el campo de Seguridad Social es numérico y no puede ser nulo. Los datos del sistema también dicen a Oracle cuales son los usuarios válidos de la base de datos.

Rollback Segments.

Cada base de datos contiene uno o más rollback segments, que son porciones de la base de datos que registran la acción de la transacción en el evento en el que una transacción es cancelada (rollback).

Actualizaciones.

Cuando se desee actualizar el software ya sea del Web Application Server o del manejador de Oracle es importante verificar que las versiones se encuentren certificadas, es decir que esté comprobado que no existen problemas de compatibilidad que afecten a la funcionalidad de los productos y de la aplicación. Esta información de certificación es proporcionada por Oracle lo que se sugiere ponerse en contacto con la gente de Soporte Técnico antes de decidir actualizar alguno de los productos anteriores.

Estas son las versiones para los productos que componen la aplicación:

- Oracle 8.0.4 (enterprise edition):
 - Contex Cartridge.
 - Enterprise Manager.
 - PLSQL.
- Web Application Server 3.0.
- Windows NT 4.0.

Respaldos.

Un respaldo de una base de datos es una copia representativa de datos. Cuando los datos originales se pierden, podemos usar el respaldo para reconstruir la información perdida. Imaginemos la magnitud del problema (sin mencionar el tremendo disgusto de

los clientes o usuarios) al dañarse la base de datos de una compañía de servicios, de un banco o de cualquier empresa que tiene que suspender sus servicios por un problema en el sistema y no se puede restablecer debido a que no se cuenta con un respaldo. Para evitar este tipo de problemas es necesario hacer un respaldo de la base de datos que nos permita restablecer rápidamente las operaciones del sistema.

Para planear adecuadamente los respaldos de la base de datos es necesario definir la estrategia de respaldo más conveniente de acuerdo a las necesidades del sistema y de la empresa. A continuación se propone un esquema de respaldos el cual puede ser modificado por el administrador del sistema de acuerdo a las nuevas necesidades de la aplicación.

En Oracle los tipos de respaldos se pueden clasificar en respaldos lógicos y respaldos físicos.

Los respaldos lógicos se encargan de respaldar los objetos de los usuarios de la base de datos (tablas, índices, package, vistas, secuencias, etc.) sin incluir información de cómo están almacenados físicamente. La utilidad más común es el EXPORT la cual nos permite poder recuperar información de un objeto particular o bien de la base de datos en su totalidad.

Los respaldos físicos se encargan de respaldar los archivos que componen una base de datos Oracle como son los datafiles, los archivos de redo log, los archivos de control y el archivo de parámetros init.ora. Para hacer esto se cuenta con los comandos de Oracle ALTER TABLESPACE BEGIN y ALTER TABLESPACE END BACKUP; así como los comandos de sistema operativo CP (copy) de UNIX y COPY para Windows NT.

De acuerdo a lo anterior se propone el siguiente esquema de respaldo:

Respaldos Físicos.

Realizar un respaldo en frío cada mes, esto consiste en copiar los archivos de la base de datos a un dispositivo de almacenamiento masivo extremo mientras la base de datos se encuentre abajo.

A. Para obtener los archivos de control:

```
SELECT NAME FROM V$CONTROLFILE;
```

Teniendo como resultado de la consulta lo siguiente:

```
NAME
```

```
-----
C:\ORANT\DATABASE\CTL1ORCL.ORA
C:\ORANT\DATABASE\CTL2ORCL.ORA
```

B. Para obtener los archivos de redo log:

```
SELECT MEMBER FROM V$LOGFILE;
```

Con lo cual se obtiene:

```
MEMBER
```

```
-----
C:\ORANT\DATABASE\LOG1ORCL.ORA
C:\ORANT\DATABASE\LOG2ORCL.ORA
```

C. Para obtener información de los datafiles:

```
SELECT NAME FROM V$DATAFILE;
```

Con el cual se obtiene:

NAME

```
-----  
C:\ORANT\DATABASE\SYS1ORCL.ORA  
C:\ORANT\DATABASE\USR1ORCL.ORA  
C:\ORANT\DATABASE\RBS1ORCL.ORA  
C:\ORANT\DATABASE\TMP1ORCL.ORA  
C:\ORANT\DATABASE\SYS2ORCL.ORA  
C:\ORANT\DATABASE\DES2DATA.ORA  
C:\ORANT\DATABASE\DES2IDAT.ORA  
C:\ORANT\DATABASE\DES2IDA2.ORA  
C:\ORANT\DATABASE\DES2DAT2.ORA  
C:\ORANT\DATABASE\USR2ORCL.ORA  
C:\ORANT\DATABASE\DES2DATA2.ORA  
C:\ORANT\DATABASE\DES2IDA3.ORA
```

D. Archivo de parámetros:

El archivo de parámetros es:

```
C:\ORANT\DATABASE\INITORCL.ORA
```

Es conveniente que estos archivos que componen la estructura física de la base de datos sean respaldados en uno o dos discos más.

No es recomendable que se respalden en el mismo disco donde se encuentra la base de datos ya que si se presentan fallas de disco, la base de datos no podrá ser recuperada.

Respaldos Lógicos.

Realizar un respaldo lógico de la base de datos los días Martes y Viernes de cada semana empleando la utilería EXPORT. Para el sistema operativo de Windows NT esta utilería se encuentra en C:\ORANT\BIN llamándose el programa ejecutable EXP73 o EXP80.

Nota: La base de datos debe estar abierta en modo restringido sin que haya ningún usuario o proceso realizando algún tipo de actividad ya que esto podría ocasionar que la información respaldada no fuera consistente.

Se recomienda utilizar este comando en línea con las siguientes opciones:

```
C:\EXP73 SYSTEM/PASSWD FULL=Y FILE=APLICA_SISTEMA.DMP LOG=APS.LOG
```

Depuración de la Base de Datos.

La depuración de la base de datos se realizará periódicamente mediante un proceso de depuración de nombre 'Depura' que se ejecutará de manera automática mediante un cron de UNIX de nombre 'Act_dep.sh'. El periodo de depuración esta sujeto a modificaciones según el flujo de información que posea la base de datos, a fin de no afectar el **performance** (desempeño) de la misma.

MANUAL DE USUARIO.

El objetivo del presente material es apoyar al usuario en el manejo y consulta de la página electrónica diseñada para este el tema de tesis, la cual pretende facilitar el proceso de consulta de avisos clasificados manejados por una Agencia de Publicidad de manera más rápida y eficaz aprovechando los alcances de Internet.

Como primer paso el usuario necesita establecer la conexión con Internet, para posteriormente, con la ayuda de un navegador universal como Netscape o Internet Explorer, ingresar a esta página electrónica tecleando en el campo de dirección: <http://www.aviso-oportuno.com.mx/> tal como se muestra en la figura 1.

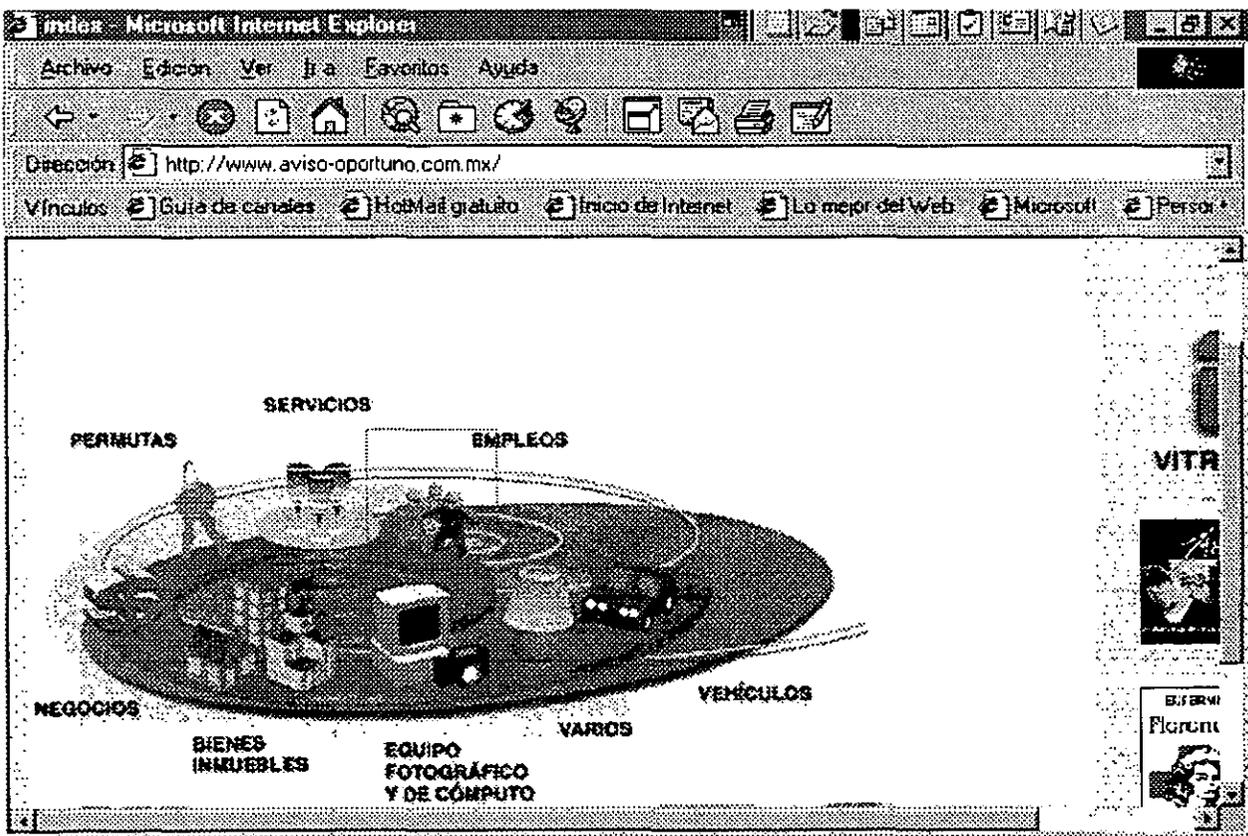


Figura 1 Página Principal del Sistema.

Esta página electrónica contiene varias secciones, por lo cual daremos una breve explicación sobre su uso, y la forma más eficaz de obtener provecho de las mismas.

En la página principal se cuenta con dos menús:

El primero localizado en la parte superior derecha de la página y el cual permanece visible durante todo el tiempo que se navega en el servicio. Este menú consta de los siguientes temas:

- **Contáctanos:** Para enviar correo electrónico al encargado de la publicación de la página electrónica.
- **Inicio:** Se utiliza para regresar a la página principal mostrada en la figura 1.
- **Mapa de Sitio:** Desde el cual se puede navegar directamente a cualquiera de las páginas del mapa tales como: inicio, servicios, empleados, vehículos, varios, equipo fotográfico y de cómputo, bienes inmuebles, negocios y permutas.
- **Búsqueda.** Donde se muestra una caja titulada Palabras a Buscar para realizar búsquedas por palabra.
- **Ayuda.** Aquí se presenta una guía o tutorial para hacer uso de esta página electrónica.

El segundo menú, que es el principal, está integrado por los 8 temas de anuncios clasificados de la página electrónica, representados cada uno por un dibujo o ícono tal como lo muestra la figura 2.

Los temas principales representados en los dibujos son: servicios, empleados, vehículos, varios, equipo fotográfico y de cómputo, bienes inmuebles, negocios y permutas. Para seleccionar alguno de estos temas, solo hay que posicionarse sobre el dibujo correspondiente y dar un clic con el botón izquierdo del mouse. De esta manera se pasa a una segunda página con el contenido necesario para realizar búsquedas subsecuentes.

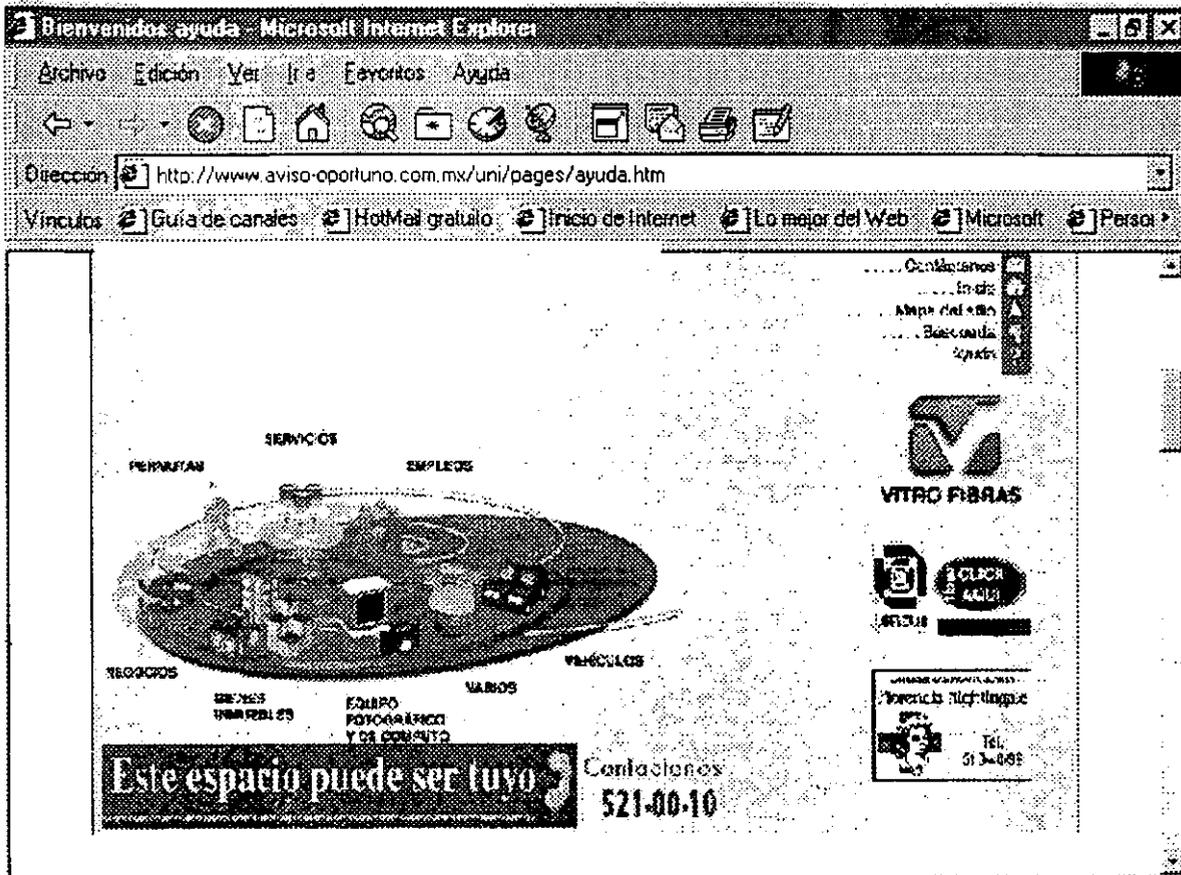


Figura 2 Menú Principal del Mapa de Sitio WEB.

Búsquedas.

Una vez que se ha seleccionado uno de los ocho temas del menú principal, se pasa a una segunda página según el tema escogido. En esta página se muestra el título y las instrucciones para realizar consultas. Más adelante aparecen dos "cajas" para facilitar las búsquedas: Una titulada "Tema de Búsqueda" y la otra "Palabras a Buscar". En la figura 4 se muestran estas características tomando como ejemplo el grupo Empleos.

Tema de Búsqueda. En esta "caja", aparecen los subtemas del grupo seleccionado en la página principal. Para visualizar el total de temas, es necesario dar un enter sobre el botón que aparece al final de la caja. En la figura 3 se muestran los subtemas pertenecientes al grupo Empleos.

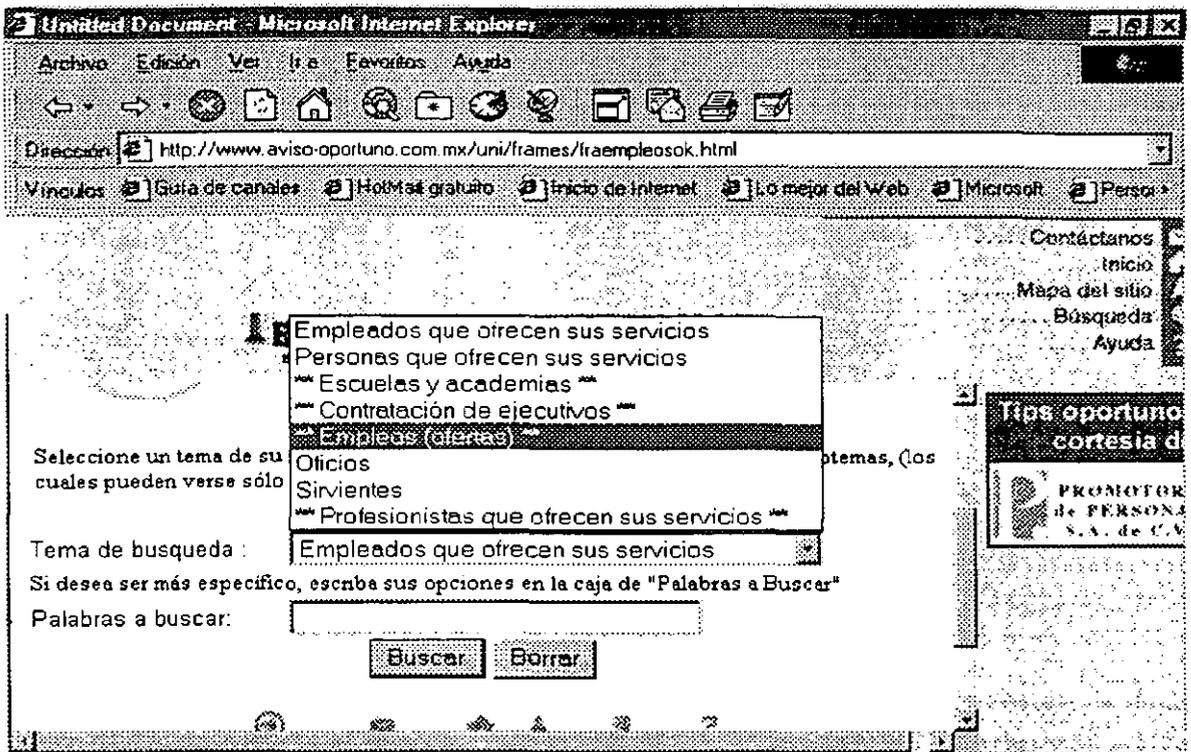


Figura 3 Subtemas del Grupo Empleos.

Si aparece un subtema entre dos asteriscos, significa que existe una segunda clasificación de temas; para poder verlos basta con seleccionar el subtema con asteriscos y teclear sobre el boton "Buscar". Finalmente, después de haber seleccionado el tema de búsqueda, solo hay que dar un clic con el boton izquierdo del mouse sobre la caja "Buscar". Si se cambia de opinión se utilizará el botón "Borrar" para limpiar la caja para posteriormente seleccionar otro tema.

Palabras a Buscar. Si se quiere ser más preciso con una búsqueda en especial, existe una segunda "caja" donde se escribirá la palabra de interés dentro del tema de búsqueda. Por ejemplo; si sólo interesan los anuncios que contengan la palabra "ingenieros", bastara con teclear esta palabra en esta caja antes de digitar el botón "Buscar" (ver figura 4) y a continuación aparecerán solo los anuncios que contengan esa palabra. El resultado de la consulta del ejemplo, aparece en la figura 5.

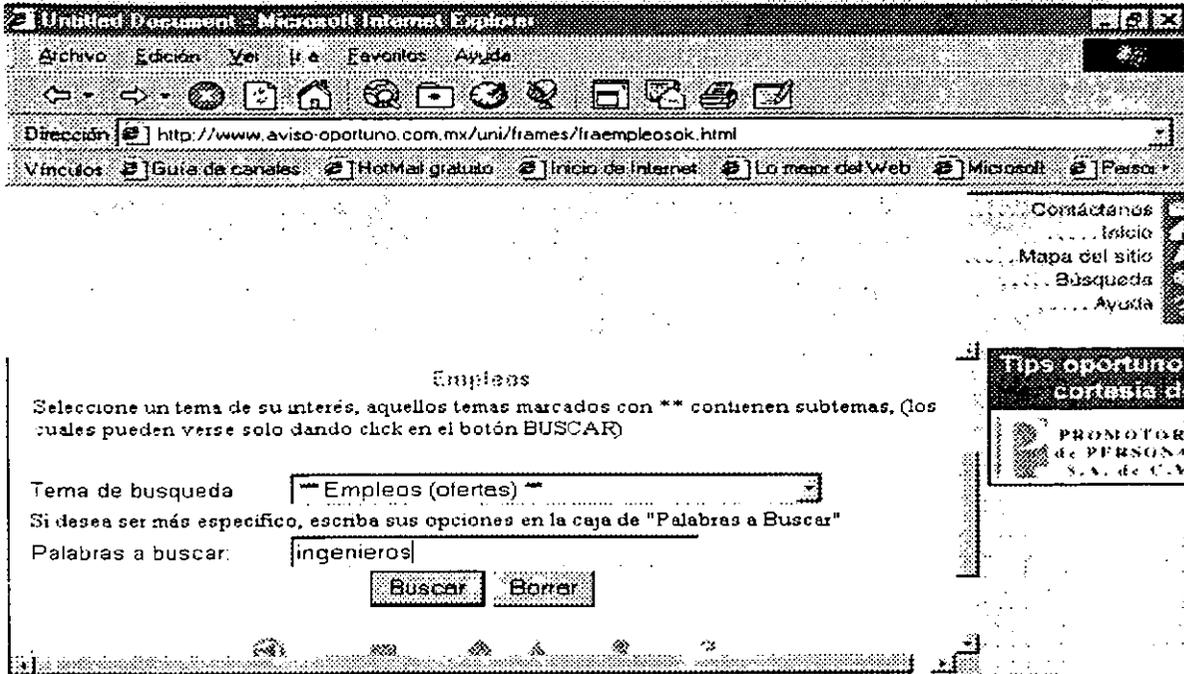


Figura 4 Búsqueda por Palabras.

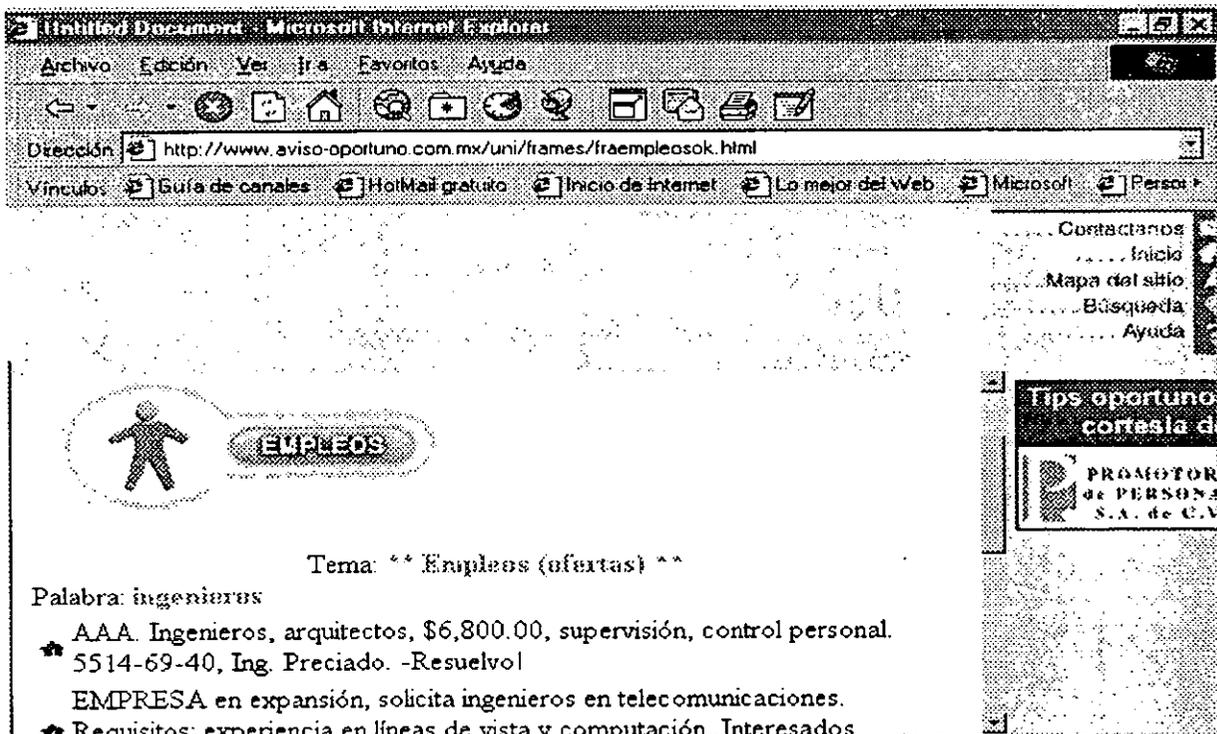


Figura 5 Resultado de la Consulta.

Búsquedas Complejas.

Para hacer búsquedas de una palabra u otra se utilizará el caracter "|" o la palabra "or" es decir si deseo buscar en autos todos los "Tsurus" además de todos los "Taurus" en el cuadro en blanco debo poner:

Tsurus | Taurus o Tsurus or Taurus

En la Figura 6 podemos observar la forma en que será buscada la información.

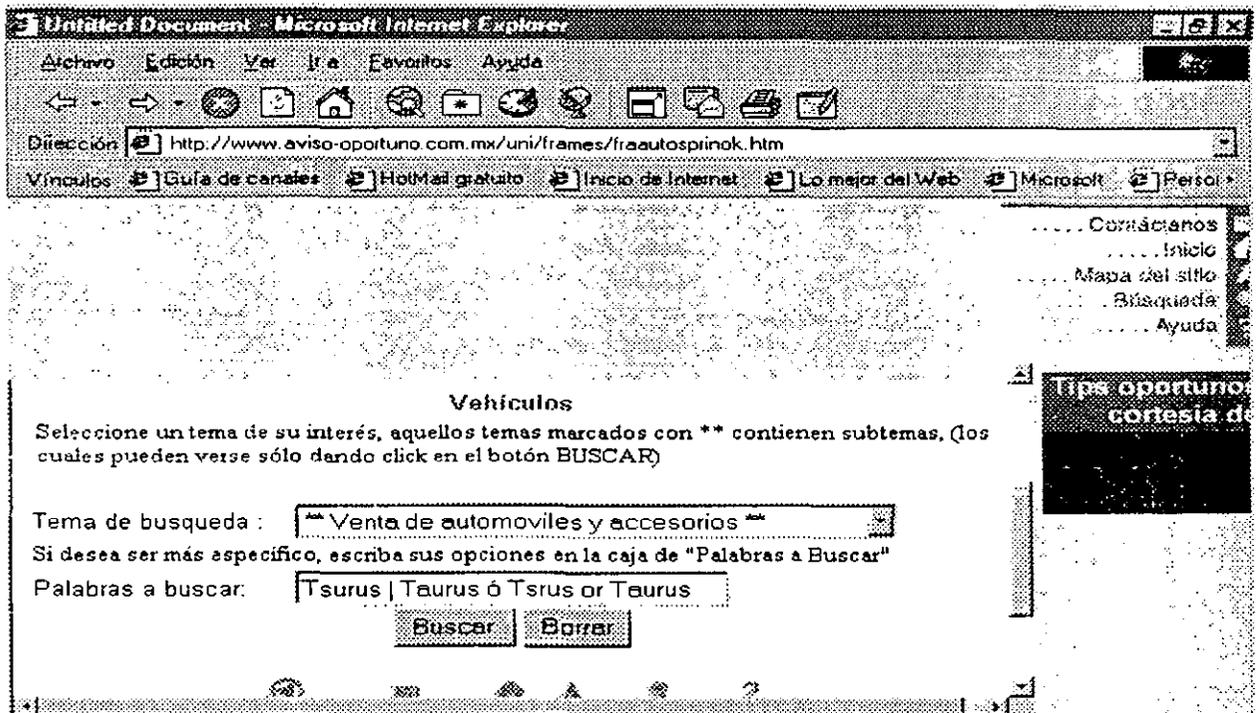


Figura 6 Formato para Realizar Búsquedas Especiales.

En la figura 7 se muestra la salida de dicha consulta, la cual se realizó por pantalla.

El resultado total de la búsqueda de Tsuru | Taurus o Tsuru or Taurus se puede observar en la tabla 1. Para este ejemplo se encontraron 20 registros.



VEHICULOS

Tema: ** Venta de automoviles y accesorios **

Palabra: Tsurus | Taurus o Tsrus or Taurus

ACCIDENTADOS: Jettas 1993, 1996, 1997; Golf 1995, Tsurus: 1993, 1994. Recuperados: Jetta 1995, Neon 1995, Voyager 1993. 5642-57-84.

TAURUS 1986, americano, legalizado, buenas comisiones, enllantado.

Figura 7 Desplegado de la Información Solicitada.

Palabra: Tsurus Taurus ó Tsurus or Taurus	ACCIDENTADOS: Jettas 1993, 1996, 1997; Golf 1995, Tsurus: 1993, 1994. Recuperados: Jetta 1995, Neon 1995, Voyager 1993. 5642-57-84
	TAURUS 1986, americano, legalizado, buenas comisiones, enllantado. \$18,900.00. 5712-36-19.
	TAURUS 1988, buenísimas condiciones, digital, clima, rines, cualquier prueba, \$22,990. Filipinas 184, esquina Eluterio Méndez, Portales.
	TAURUS 1988, factura original, negro, sin detalles, llantas 3/4 de vida, tenencia 98, muy bien cuidado, \$33,000 a tratar. 5577-72-51.
	TAURUS 1988, flamante. Enganche \$8,000.- resto 24 meses. Recibo auto. 5581-72-47.
	TAURUS 1988, totalmente original, enllantado, aire, \$25,500, recibo auto. 5579-14-64.
	TAURUS 1989, 80,000 kilómetros, impecable estado, cambiaría. Mayores informes 5605-26-90.
	TAURUS 1989, TOTALMENTE ORIGINAL, COLOR VINO, CUALQUIER PRUEBA. \$27,500.00 5397-76-72.
	TAURUS 1989, vestiduras velour, color azul original, \$29,000. 5781-

	28-62, 5781-85-60.
	TAURUS 1990, excelente estado, piel, eléctrico, clima. Urge, \$34,000.00. 5763-10-11. Señor Bejar.
	TAURUS 1990, totalmente equipado, algunas facilidades. Cuauhtémoc 1087, Glorieta Rivera.
	TAURUS 1990 de piel, se cambia por departamento o carro. 691-07-34.
	TAURUS 1990 y 1989, ambos buenas condiciones. Informan 601-00-45 Cambiaria.
	TAURUS 88, perfecto estado, exigentes. Remato. \$26,500. 795-74-46.
	TAURUS 90, aire, eléctrico, piel, buen estado. Urge. \$35,000.00. 5341-67-17.
	TAURUS 91, eléctrico, aire, cuidado, factura original, rines, tratamos 5752-34-47.
	TAURUS guayín 1989, único dueño, completamente original, nueva. Exigentes. 362-85-06, Satélite.
	TAURUS Guayín 1991, excelente estado, totalmente equipada, oportunidad, \$47,800.00. 653-04-54.
	ACCIDENTADOS NO DESARMO PIEZAS, MOTOR, CAJA, INTERIORES COALICION, POR PIEZA O LOTE. EXPLORER XLT 98, RAM CHARGER 93, JETTA GLX 96 - 98, NEON 98, TSURU 95, STRATUS 98, TAURUS 86, THUNDERBIRD 92. PRECIOS ESPECIALES DESHUESADEROS, TALLERES. FACTURADO JJ AUTOBOUTIQUE. 5526-76-50, 5526-68-53.
	ACCIDENTADOS Taurus 1989, facilito, \$21,800.00. Recuperados Cavalier 1991, \$25,000.00, Tsuru 1993, completito, \$45,000.00. 632-42-49.

Tabla 1. Resultado Total de la Consulta Tsuru o Taurus.

Si deseo encontrar dos palabras es decir todos los anuncios de casas en "Polanco" y que sean "Penthouse" debo utilizar "&" o la palabra "and" es decir:

Polanco & Penthouse ó Polanco and Penthouse

Si deseo una búsqueda de todos los anuncios de autos que contengan la palabra "Camion" con cualquier variante como "camioneta", "camioncito", etc la forma de búsqueda deberá quedar como sigue:

Camion%

De igual forma se puede realizar la consulta para cualquier otro tema o palabra a buscar.

CONCLUSIONES

1. El acceso al Sistema de Consultas de Productos y Servicios se realiza de una manera fácil y rápida a través del ambiente gráfico con el que opera y a las opciones de búsqueda que ofrece y con las cuales se logra filtrar la información y encontrar más rápidamente los temas a consultar. Con esto se obtiene una pantalla con el reporte de sólo lo que interesa al usuario, evitando llenarlo de información que no le sirve, por lo que el objetivo principal se cumple; pero, como todas las páginas de Internet, tiene el inconveniente de que está limitado el acceso por la capacidad del proveedor que presta los servicios de conectividad a la World Wide Web, ya que en ocasiones no se logra establecer una conexión vía modem por encontrarse las líneas saturadas.
2. El modelo de tres capas utilizado para la implementación del sistema en Internet y que en Oracle se conoce como Arquitectura de Cómputo en Red (Network Computing Architecture, NCA), permitió soportar más adecuadamente la concurrencia de muchos usuarios interesados en consultar esta página Web. Esto se logró gracias a que NCA está compuesta por tres elementos que optimizan el funcionamiento de una red: un servidor de datos para el almacenamiento y manipulación de información y un servidor de aplicaciones que permite la conectividad de muchos usuarios finales que constituyen el tercer elemento conocido como clientes delgados. Como el sistema tiene características de alta disponibilidad, esta arquitectura facilitó y eficientó la toma de respaldos y actualizaciones al sistema, ya que los recursos de hardware y software se optimizaron con el modelo seleccionado.
3. Aprendimos que el utilizar herramientas de software escalables y abiertas, hace más fácil la integración de módulos al sistema. Por otro lado, la portabilidad de estas herramientas hacia otras plataformas, nos permitió realizar de manera eficaz trasladar el sistema de Windows NT a Sun Solaris. Estas características permitirán en el futuro incorporar de manera fácil, nuevos módulos al sistema que faciliten

interactuar de manera más directa al usuario con los prestadores de servicios o vendedores de los productos anunciados.

4. Considerando que este sistema se encuentra clasificado como un “sistema de alta disponibilidad”, es decir, que cualquier tipo de error provocaría no tener en línea el sistema y por lo tanto que los usuarios no podrían consultar los clasificados, entonces el planear y dimensionar bien la Base de Datos, así como configurar correctamente los cartuchos y el Web Application Server, fue fundamental para que el sistema se encuentre operando adecuadamente.
5. El presente trabajo aporta los conocimientos básicos y necesarios para guiar al lector desde la instalación del Servidor Web hasta la creación de la página en Internet, con lo cual el lector no requerirá de complicados y grandes volúmenes de información.
6. La metodología propuesta en este estudio consideró como hardware un equipo Compaq Pentium II a 350 Mhz con sistema operativo Windows NT 4.0 Server y el software Oracle 8.0.5 (Enterprise Edition que incluye: Context Cartridge, Enterprise Manager PLSQL), Web Application Server 3.0 y permitió la creación del Web prototipo del Aviso Oportuno del Periódico el Universal, el cual hasta el momento ofrece varios servicios de interés para la comunidad en general, ampliando la difusión de empleos y servicios.
7. Cabe señalar que las herramientas que ofrece este trabajo solo contemplan la instalación y desarrollo de un Web, por lo que también deberá considerarse como aspecto importante del mismo, la presentación visual de la página, así como la calidad de la información, por que el producto final es la página en sí, que es lo que el usuario recibe, consideramos que para lograr una página de calidad se requiere de diferentes disciplinas tales como el diseño gráfico, la publicidad y otras ciencias de la comunicación.

8. Consideramos que cada uno de nosotros aprendimos a utilizar Oracle, así como sus herramientas, ya que actualmente el manejo de base de datos es de suma importancia y cada vez más en el mercado es necesario contar con el conocimiento de manejadores de base de datos cada vez más potentes.
9. La manipulación de los datos es más fácil con el uso de herramientas Oracle, porque al utilizar el lenguaje SQL, se ahorra tiempo al realizar las consultas, y se puede tener control sobre las tablas y vistas. Con la utilización de HTML para la creación de las páginas en combinación con el PL/SQL se logró que el sistema fuera más amigable para el usuario.
10. Los objetivos de realizar una tesis no es solo cubrir con un requisito más que la Facultad de Ingeniería y nuestra Universidad solicitan, sino concluir con la meta de terminar una carrera profesional y no olvidar que cuando algo se empieza se debe concluir; un hábito que no debemos olvidar .
11. En la actualidad, la demanda de empleos es cada vez mayor y la oferta es cada vez menor. Por tal motivo, los profesionistas tenemos el compromiso de actualizarnos y prepararnos cada vez más en los temas más importantes y de interés para aquellos que solicitan nuestros servicios. Desafortunadamente pocos seguimos el consejo. En este panorama, el trabajo de tesis fue una oportunidad de ampliar, aprender e investigar temas de gran interés en el área de Ingeniería en Computación, que no dudamos nos serán de gran ayuda en nuestro desempeño profesional.

BIBLIOGRAFÍA

- Abbey Michael, Corey Michael J., **ORACLE 8, Guía de Aprendizaje**, Mc Graw Hill, España, 1998.
- Alin Frederic, Lafont Denis, Macary Jean Francois, **El Proyecto Internet**, Ed. Gestión 2000 S.A. Eyrolles, HD30.385, A5518.
- Armitage Tim, Pountney Jason, **Oracle Web Application Server: Develop Web-Based Applications**, Oracle Corporation, 1997.
- Austin David, Caindec Laie, **Oracle Server: Database Administration**, Oracle Corporation, 1996.
- Black Uyles D., **Redes de Computadoras Protocolos, Normas e Interfaces**, México Macrobit Ra –Ma, 1995.
- *Colin McGregor, **ORACLE8 Context Cartridge Application Developer's Guide**, Oracle Corporation, 1998.*
- Comer Douglas E., **Redes Globales de Información con Internet y TCP/IP, Principios Básicos, Protocolos y Arquitectura**, Prentice - Hall Hispanoamericana, México, 1998.
- Connor Deni, Mark Anderon, **Networking the desktop: Cabling, Configuration and Communications**, A.P Proffesional, Boston, 1995.
- De Wire Dawna Travis **Client / Server Computing** Mc Graw Hill, New York 1995.
- Silvatech, **Programación en HTML**, Division Educational, 1997.

- Educational Department, **Oracle Context Cartridge**, Oracle Corporation, 1996.
- Ellsworth Jill H., **The Internet Bussines Book**, New York, Edit.Wiley, 1996.
- Ferreyra Cortés Gonzalo, **Internet Gráfico, Herramientas de World Wide Web**, Ed. Computec, México, TK5105.875157, F467.
- Gburzynsky Pawel, Englewood Clifs , **Protocol design for local and Metropolitan Area Networks**, New Jersey, Prentice Hall, 1995.
- Goldman ,Jamer E., **Local Area Networks: Client /Server Approach**, New York, J.Wiley, 1997.
- Gundavaram Shishir, **CGI Programming on the World Wide Web**. 1a ed. Estados Unidos: O`Rally & Associates, 1996. 433 p.
- Hallsal Fred, **Data Communications Computer Networks and Open Systems**, 4 Edición, Harlow England, Addison Wesley, México, 1995.
- Heslop Brent D., **HTML Publishing on the Internet for Windows: Create Great-Looking Documents On Line: Home Pages, Newsletters, Catalogs, ads & Forums**, 1a ed., Chepel Hill, NC: Ventana Press, c1995. 480 p.
- James Gardner, **Aprendiendo UNIX** , Segunda Edición, Editorial Prentice Hall.
- José Luis Raya, **Windows NT Server Versión 4**, Editorial Ra-Ma, México, 1997.
- José Mopin, **Interconexión de Periféricos**, Publicaciones Macombo. Serie: Mundo Electrónico.

- Karney James, **"SGML y HTML Maestros de la Etiqueta."** p. 85-96. En PC Magazine en español. Vol. 6 no. 6 (jun. 1995).
- Khan Sharaaz, Taylor Larry, **Oracle Enterprise Manager**, Oracle Corporation, 1996.
- Kochhar Neena, Kramer Debby, **Introduction to Oracle and PL/SQL Using Procedure Builder**, Oracle Corporation, 1996.
- Krol Ed, **Conectate al Mundo Internet**, Mc Graw Hill, México, TK5105.875157.K7618.
- Laura Lemay, **Aprendiendo HTML para Web en una Semana**, Sams Publishing Prentice Hall Hispanoamericana.
- Majocchi Rita y Attanasio Franco, **Cómo Hacer Publicidad**, Ediciones DEUSTO, Bilbao España, 1973.
- Martin James y Odell James J., **Métodos Orientados a Objetos**, Ed. Prentice Hall, QA76.64. M3691.
- Martínez Ramos Emilio, **El Uso de los Medios de Comunicación en Marketing y Publicidad**, Akal / Comunicación, HF5415.123.M37
- **Microsoft Windows NT Server Resource Kit ver 4**, 1996, Microsoft Professional Editions.
- **Miller Mark Troubleshooting Internetwork: Tools ,Techniques and Protocols**. San Mateo California Prentice Hall 1996.

- Perlman Radia, **Interconnection Bridges and Routers**, México City, Addison Wesley, 1995.
- Pressman Roger S., **Ingeniería de Software: Un Enfoque Práctico**, ed. Mc-Graw Hill.
- Ramón M. Chorda, **XENIX**, Editorial RA-MA, México, 1995.
- Rosen Kennet H., **UNIX Sistema V**, Versión 4, Primera Edición, Madrid; México: Osborne . Mc-Graw Hill, 1995.
- Russel Charlie Running, **Microsoft Windows NT Server 4 The Essential guide for Administrators, Systems Engineers and is professionals**, Washington, Microsoft, 1997.
- Sammerville Lan, **Ingeniería de Software**, Ed. Wesley Iberoamericana, QA76.6, S8618.
- Sánchez Crespo Daniel, Sánchez Crespo Ramiro, **Internet en la Empresa**, Ed. Inforbooks, S. L., HD30.37, S33.
- *Santifaller Michel, **Internetworking a Unix Enviroment** , Colección Informática de Gestión.*
- Schildt Herbert, **Turbo C/C++**, Manual de Referencia, Mc Graw Hill.
- Schwartz M., **Cableado de Redes**, Traducción Fernando Diaz Hillan, Paraninfo, Madrid, 1996.

- Shelton Tom . **Soluciones para Interconectividad en Red**, Traducción José Angel Vallejo Pinto, Mc Graw Hill, México, 1995.
- Sterne Jim, **World Wide Web Marketing**, Promociones Jumerca, HF5415.1265.
- Sun Education Service, **Solaris 2.X Network Administration**, Sun Microsystems, 1995.

ANEXO 1

Formatos de Salida de Archivos de Publicidad.

Formato del archivo: Salida.

</clsad,0101B,9724649></CLSI>ACAPULCO, semana de s£per lujo del 23 al 29 de enero, Hotel Acapulco Plaza, dos habitaciones (para 4 a 8 personas) acceso al Club Privado Oasis. 800 d¢lares, total por las 7 noches. Tel,fono (01-83) 00-55-38, Monterrey N. L.@

</clsad,0101B,9727748></CLSI>DE puente Huatulco Zaashila. Suites lujo, alberca integrada. 633-40-19, 670-15-33.@

</clsad,0101B,9726150></CLSI>ESTRESADO, PLAYA VIRGEN TUXPAN, VER. 8 CUARTOS, TODOS LOS SERVICIOS, LAGUNA, ALBERCA, MAR. 678-07-81.@

</clsad,0101B,9719546></CLSI>GANGA. SUITE EN HOTEL 5 ESTRELLAS PARA EL DA EN POCA QUE USTED ELIJA EN PUERTO VALLARTA A PRECIO DE 1994. 515-52-01, LUNES A VIERNES, HORAS HABILES.@

</clsad,0101C,9725967></CLSI>REMATO TIEMPO COMPARTIDO, nacional, internacional, CancEn, Puerto Vallarta, Acapulco, Estados Unidos, Canad . Vale 40,000; remato en 20,000. Urge. Sra. Patricia Alfaro. 398-42-46.@

</clsad,0101D,9717436></CLSI>TRASPASO membres;a Santa Mar;a Tlalmanlco. (01-597) 3-12-06. Nos arreglamos. facilidades.@

</clsad,0101D,9723222></CLSI>VENDO acci£n Club alem n Tepepan por viaje. \$27,300.- (91-74) 82-27-72.@

</clsad,0301A,9721220></CLSI>ACAPULCO, condominio lujo, vendo departamento amueblado, dos rec maras, playa, albercas, chapoteaderos, jardines, inmejorables servicios. 658-37-12.@

</clsad,0301A,9712857></CLSI>ACAPULCO, CUERNAVACA. CASA LUJO, 4, 5 RECAMARAS, ALBERCA, JARDIN. 630-32-70.@

</clsad,0301A,9721409></CLSI>ACAPULCO, frente mar, vendo amplio departamento, terraza, alberca, estacionamiento. \$360,000.- 674-16-59.@

Formato del archivo: Salida1.

</clsad,0301A,9728041></CLSI>ACAPULCO, Penthouse amueblado, 2 recámaras, alberca, garage, por la Diana, frente a SAMS. \$415,000. 517-70-92.@

</clsad,0301A,9660649></CLSI>ACAPULCO, rento/ vendo casa amueblada, tres recámaras, alberca, vista. 652-53-20, 514-60-97.@

</clsad,0301A,9703900></CLSI>ACAPULCO, Villas Princess, Las Brisas, departamentos, playa, casas grandes. 363-16-51.@

</clsad,0301A,9714207></CLSI>ACAPULCO. Rento preciosísima casa, 6 recámaras, alberca, limpieza. \$3,000.00 diarios. Enero descuentos. 540-40-30.@

</clsad,0301A,9727156></CLSI>ACAPULCO JOYAS DE BRISAMAR, PRECIOSO DEPARTAMENTO VISTA A LA BAHIA. 45,000 DOLARES. 562-63-48.@

</clsad,0301A,9658887></CLSI>ACAPULCO Princess, rento villa de lujo, aparte semana santa o verano. 333-02-25 clave 325-96-54, 343-87-95.@

</clsad,0301A,9725309></CLSI>ACAPULCO rento casas, cuatro, cinco recámaras, alberca, vista. 294-27-53, 294-27-93.@

</clsad,0301A,9725354></CLSI>ACAPULCO zona dorada, Suite familiar, vista mar, alberca, cocineta; disponible enero, \$350.00 noche. 578-05-18.@

</clsad,0301A,9719854></CLSI>DEPARTAMENTO 3 RECAMARAS, COMPLETAMENTE AMUEBLADO, COSTA AZUL, ALBERCA, \$1,500 DIARIOS. 602-48-49, 643-65-16.@

</clsad,0301A,9726226></CLSI>FIN semana, Villas, Condesa, jardín, alberca, servicios, \$800 noche. 564-01-09.@

ANEXO 2

Código de Programas para la Creación de la Base de Datos.

Archivo: formato.ccp

Elimina los caracteres sin sentido para la carga de información.

```
#include <stdio.h>
#include <stdlib.h>
#include <string.h>
#include <conio.h>

char *formatea(char *s);

int main(void)
{
    FILE *en,*sa;
    char cad1[350],cad2[5000],*p;
    int dummy,i,k;

    cad2[0]='\0';
    p=NULL;

    clrscr();

    if ((en=fopen("c:\\archivo\\texto.txt","r"))==NULL)
    {
        printf("no se puede leer el archivo");
        exit(1);
    }

    if ((sa=fopen("c:\\archivo\\salida.txt","w"))==NULL)
    {
        printf("no se puede abrir el archivo");
        exit(1);
    }

    while(!feof(en))
    {
        fgets(cad1,300,en);
        if
        ((p=strstr(cad1,"GALL"))||(p=strstr(cad1,"<cl>"))||(p=strstr(cad1,"clscd"))||(cad1[0]=='>'))||(strlen(cad1)<=2)
        )
        {
        }
        else
        {
            if (!(p=strstr(cad1,"<QJ>")))
            {
                if (cad2[0]=='\0')
                {
```

```

        k=strlen(cad1);
        cad1[k-1]='@';
        strcpy(cad2,cad1);
        fputs(formatea(cad2),sa);
        fputs("\n",sa);
        cad2[0]='\0';
    }
    else
    {
        k=strlen(cad1);
        cad1[k-1]='@';
        strcat(cad2,cad1);
        fputs(formatea(cad2),sa);
        fputs("\n",sa);
        cad2[0]='\0';
    }
}
else
{
    i=strlen(cad1);
    if (cad1[i-6]=='-')
    {
        cad1[i-6]='\0';
        if (cad2[0]!='\0')
            strcat(cad2,cad1);
        else
            strcpy(cad2,cad1);
    }
    else
    {
        cad1[i-5]='\0';
        if (cad2[0]!='\0')
            strcat(cad2,cad1);
        else
            strcpy(cad2,cad1);
    }
}
}
} /* llave del while */
fclose(en);
fclose(sa);
return 0;
}
char *formatea (char *s)
{
    char c[5000],c2[5000];
    int j,conta=0,k=0,m=0;

    if (s[12]!=';')
    {
        for(j=0;j<=11;j++)
            c[j]=s[j];
        c[12]='\0';
        for(j=13;j<=strlen(s)+1;j++)
            c[j]=s[j-1];
    }
}

```

```

    c[j+1]='\0';
}
else
{
strcpy(c,s);
}
while(k<=strlen(c))
{
if (c[k]==' ')
    conta=conta+1;
if ((conta!=2) && (conta!=3))
    {
        c2[m]=c[k];
        m++;
    }
    k++;
}
return c2;
}

```

Archivo: corrige.ccp.

Sustituye acentos en la forma adecuada.

```

#include <stdio.h>
#include <stdlib.h>
#include <string.h>
#include <conio.h>

char corrige(char s);

int main(void)
{
FILE *en,*sa,*err;
char cad1[5000],s,*p,*q;
int i;

clrscr();

if ((en=fopen("c:\\archivo\\salida.txt","r"))==NULL)
{
printf("no se puede leer el archivo\n");
exit(1);
}

if ((sa=fopen("c:\\archivo\\salida1.txt","w"))==NULL)
{
printf("no se puede abrir el archivo\n");
exit(1);
}

if ((err=fopen("c:\\archivo\\salida.err","w"))==NULL)
{
printf("no se puede abrir el archivo\n");
exit(1);
}
}

```

```

while(!feof(en))
{
    fgets(cad1,5000,en);
    if (((p=strstr(cad1,"</clsad") && (q=strstr(cad1,"</CLSI>")) ||
        ((p=strstr(cad1,"</clsad") && (q=strstr(cad1,"</KLSI>")))))
        {
            for(i=0;i<=strlen(cad1);i++)
            {
                s=corrige(cad1[i]);
                cad1[i]=s;
            }
            fputs(cad1,sa);
        }
    else
    {
        fputs(cad1,err);
    }
}

fclose(en);
fclose(sa);
fclose(err);
return 0;
}

```

```

char corrige(char s)

```

```

{
    if (s=='µ')
        return 'Á';
    else if (s==' ' )
        return 'á';
    else if (s=='□')
        return 'É';
    else if (s==',')
        return 'é';
    else if (s=='Ö')
        return 'Í';
    else if (s=='i')
        return 'í';
    else if (s=='à')
        return 'Ó';
    else if (s=='ø')
        return 'ó';
    else if (s=='é')
        return 'Ú';
    else if (s=='£')
        return 'ú';
    else if (s=='Ÿ')
        return 'Ñ';
    else if (s=='¤')
        return 'ñ';
    else if (s=='š')
        return 'Ů';
    else if (s=='□')
        return 'ü';
    else if (s=='Ó')

```

```

return 'ë';
else if (s=='%')
return 'è';
else return s;
}

```

Archivo: crea_base.sql

Genera la estructura de la base de datos.

```
DROP TABLE CATA_SUBMARCA CASCADE CONSTRAINTS;
```

```

CREATE TABLE CATA_SUBMARCA
  (ID_SUBMARCA      VARCHAR2(4) NOT NULL,
   SUBTIPO          VARCHAR2(4) NOT NULL,
   TIPO             NUMBER(3) NOT NULL,
   TITULO           VARCHAR2(80) NULL
);

```

```

CREATE UNIQUE INDEX XPKCATA_SUBMARCA
  ON CATA_SUBMARCA
  (
   ID_SUBMARCA      ASC,
   SUBTIPO          ASC,
   TIPO             ASC
);

```

```

ALTER TABLE CATA_SUBMARCA
  ADD ( CONSTRAINT XPKCATA_SUBMARCA PRIMARY KEY (ID_SUBMARCA, SUBTIPO, TIPO) );

```

```
DROP TABLE INMUEBLES CASCADE CONSTRAINTS;
```

```

CREATE TABLE INMUEBLES
  (ID_EMPLEOS      NUMBER(4) NOT NULL,
   SUBTIPO          VARCHAR2(4) NULL,
   TIPO             NUMBER(3) NULL,
   IMAGEN           VARCHAR2(20) NULL,
   TEXTO            VARCHAR2(4000) NULL,
   FECHA            DATE NULL
);

```

```

CREATE UNIQUE INDEX XPKINMUEBLES
  ON INMUEBLES
  (
   ID_EMPLEOS      ASC
);

```

```

ALTER TABLE INMUEBLES
  ADD ( CONSTRAINT XPKINMUEBLES PRIMARY KEY (ID_EMPLEOS) );

```

```
DROP TABLE INMUEBLES CASCADE CONSTRAINTS;
```

```

CREATE TABLE INMUEBLES
  (ID_COMPUTO      NUMBER(4) NOT NULL,

```

```

SUBTIPO          VARCHAR2(4) NULL,
TIPO             NUMBER(3) NULL,
IMAGEN          VARCHAR2(20) NULL,
TEXTO           VARCHAR2(4000) NULL,
FECHA           DATE NULL
);

CREATE UNIQUE INDEX XPKINMUEBLES
ON INMUEBLES
(
    ID_COMPUTO          ASC
);

ALTER TABLE INMUEBLES
ADD ( CONSTRAINT XPKINMUEBLES PRIMARY KEY (ID_COMPUTO) );

DROP TABLE CATA_SUBMARCA CASCADE CONSTRAINTS;

CREATE TABLE CATA_SUBMARCA
(ID_SUBMARCA      VARCHAR2(4) NOT NULL,
SUBTIPO          VARCHAR2(4) NOT NULL,
TIPO             NUMBER(3) NOT NULL,
TITULO           VARCHAR2(80) NULL
);

CREATE UNIQUE INDEX XPKCATA_SUBMARCA
ON CATA_SUBMARCA
(
    ID_SUBMARCA          ASC,
    SUBTIPO              ASC,
    TIPO                 ASC
);

ALTER TABLE CATA_SUBMARCA
ADD ( CONSTRAINT XPKCATA_SUBMARCA PRIMARY KEY (ID_SUBMARCA, SUBTIPO, TIPO) );

```

Aarchivo: tipo.sql

Carga los catálogos de los clasificados y tipos de anuncios.

```

insert into clasificados
    values(90,'** Raiz **');
insert into clasificados
    values(91,'** Negocios **');
insert into clasificados
    values(92,'** Permutas **');
insert into clasificados
    values(93,'** Servicios **');
insert into clasificados
    values(94,'** Empleos **');
insert into clasificados
    values(95,'** Bienes inmuebles **');
insert into clasificados
    values(96,'** Equipo fotográfico y de cómputo **');

```

```

insert into clasificados
  values(97,'** Varios **');
insert into clasificados
  values(98,'** Vehículos **');

insert into cata_tipo
  values(1,95, '** Alquiler, permutas de tiempos compartidos y viajes **',2);
insert into cata_tipo
  values(3,95,'** Alquiler y venta casas en provincia **',2);
insert into cata_tipo
  values(4,95,'** Alquiler y venta casas en el Estado de México **',2);
insert into cata_tipo
  values(5,95,'**Alquiler de viviendas e inmuebles en el D.F. **',2);
insert into cata_tipo
  values(6,95,'Fianzas y avales ',1);
insert into cata_tipo
  values(7,95,'Alquiler cuartos Amueblados',1);
insert into cata_tipo
  values(8,95,'Alquiler despachos(solicitudes y ofertas)',1);
insert into cata_tipo
  values(9,95,'Alquiler varios (solicitudes y ofertas)',1);
insert into cata_tipo
  values(39,95,'** Venta casas, condominios y edificios **',2);
insert into cata_tipo
  values(40,95,'**Venta de terrenos **',2);
insert into cata_tipo
  values(41,95,'Granjas y Ranchos (Compra venta)',1);

```

Archivo: Subtipo.sql

Carga el catalogo de subtipos de anuncios

```

insert into cata_subtipo
  values(1,'01A','Condominios',1);
insert into cata_subtipo
  values(1,'01B','Hoteles',1);
insert into cata_subtipo
  values(1,'01C','Permutas',1);
insert into cata_subtipo
  values(1,'01D','Acciones de club',1);
insert into cata_subtipo
  values(1,'01E','Agencias de viajes',1);
insert into cata_subtipo
  values(1,'000','No clasificados',1);
insert into cata_subtipo
  values(3,'01A','Acapulco',1);
insert into cata_subtipo
  values(3,'01B','Cancún',1);
insert into cata_subtipo
  values(3,'01C','Cuautla',1);
insert into cata_subtipo
  values(3,'01D','Cuemavaca',1);
insert into cata_subtipo
  values(3,'01E','Guadalajara',1);

```

Archivo: submarca.sql

Carga el catalogo de submarcas de vehículos

```

insert into cata_submarca
  values(38,'10A','100','Cavalier');
insert into cata_submarca
  values(38,'10B','100','Celebrity');
insert into cata_submarca
  values(38,'10C','100','Century');
insert into cata_submarca
  values(38,'10D','100','Cutlass');
insert into cata_submarca
  values(38,'10E','100','Chevy');
insert into cata_submarca
  values(38,'10F','100','Suburban');
insert into cata_submarca
  values(38,'10G','100','Cadillac');

```

Archivo: falta_cata

Scripts para el caso de una nueva categoría y que no se haya definido en tablas.

```

create or replace procedure CARGA_TIPO_HOY
is
  cursor c1 is select * from cata_tipo;
begin
  for cur in c1 loop
    insert into tipo_hoy values (cur.id_tipo,cur.titulo,cur.padre,cur.clasi);
  end loop;
  commit;
end;
/
--*****
create or replace procedure CARGA_SUBTIPO_INMUEBLES
is
  cursor c1 is select * from cata_subtipo
  where tipo in (1,3,4,40);
begin
  for cur in c1 loop
    insert into subtipo_hoy values (cur.id_subtipo,cur.titulo,cur.tipo,cur.clasi);
  end loop;
  commit;
end;
/
--*****
create or replace procedure CARGA_SUBTIPO_EMPLEOS
is
  cursor c1 is select * from cata_subtipo
  where tipo in (18,19,20,23);
begin
  for cur in c1 loop
    insert into subtipo_hoy values (cur.id_subtipo,cur.titulo,cur.tipo,cur.clasi);

```

```

end loop;
commit;
end;
/
--*****
create or replace procedure CARGA_SUBTIPO_COMPUTO
is
  cursor c1 is select * from cata_subtipo
  where tipo in (47);
begin
  for cur in c1 loop
    insert into subtipo_hoy values (cur.id_subtipo,cur.titulo,cur.tipo,cur.clasi);
  end loop;
  commit;
end;
/
--*****
create or replace procedure CARGA_SUBTIPO_VARIOS
is
  cursor c1 is select * from cata_subtipo
  where tipo in (25,33,42,44);
begin
  for cur in c1 loop
    insert into subtipo_hoy values (cur.id_subtipo,cur.titulo,cur.tipo,cur.clasi);
  end loop;
  commit;
end;
/
--*****
create or replace procedure CARGA_SUBTIPO_SERVICIO
is
  cursor c1 is select * from cata_subtipo
  where tipo in (30,35);
begin
  for cur in c1 loop
    insert into subtipo_hoy values (cur.id_subtipo,cur.titulo,cur.tipo,cur.clasi);
  end loop;
  commit;
end;
/
--*****
create or replace procedure CARGA_SUBTIPO_SUBMARCA
is
  cursor c1 is select * from cata_submarca;
begin
  for cur in c1 loop
    insert into submarca_hoy values
      (cur.id_submarca,cur.titulo,cur.tipo,cur.id_subtipo);
  end loop;
  commit;
end;
/
--*****
create or replace procedure CARGA_SUBTIPO_VEHICULO
is
  cursor c1 is select * from cata_subtipo

```

```

where tipo=38 and
      substr(id_subtipo,1,2) in ('01','25','26','27','28','35','60','61','62','63','64');
begin
  for cur in c1 loop
    insert into subtipo_hoy values (cur.id_subtipo,cur.titulo,cur.tipo,cur.clasi);
  end loop;
  commit;
end;
/
--*****
create or replace procedure CARGA_SUBTIPO_SEC5_39
is
  cursor c1 is select * from cata_subtipo
  where tipo IN (5,39) and
        substr(id_subtipo,1,2) in ('10','20','30','40','50','60','00');
begin
  for cur in c1 loop
    insert into subtipo_hoy values (cur.id_subtipo,cur.titulo,cur.tipo,cur.clasi);
  end loop;
  commit;
end;
/
commit;

```

Archivo: l_t.sql

Carga las tablas de anuncios (Inmuebles, Negocios, Permutas, etc..) con los anuncios cargados en la tabla temporal.

```

-- BORRA LOS REGISTROS ANTERIORES DE LA TABLA
DELETE FROM INMUEBLES;
DELETE FROM NEGOCIOS;
DELETE FROM PERMUTAS;
DELETE FROM EMPLEOS;
DELETE FROM COMPUTO;
DELETE FROM VARIOS;
DELETE FROM VEHICULOS;
DELETE FROM SERVICIO;
COMMIT;

-- INSERTA DE LA TABLA GLOBAL "PRUEBA" A LAS TABLAS CORRESPONDIENTES

INSERT INTO INMUEBLES(ID_INMUEBLE,TIPO,SUBTIPO,IMAGEN,TEXTO,SUBVI,FECHA)
SELECT ID,TIPO,SUBTIPO,IMAGEN,TEXTO,SUBSTR(SUBTIPO,1,2),(SYSDATE+1)
FROM TEMPORAL
WHERE tipo in (1,3,4,5,6,7,8,9,39,40,41);

INSERT INTO NEGOCIOS(ID_NEGOCIO,TIPO,SUBTIPO,IMAGEN,TEXTO,FECHA)
SELECT ID,TIPO,SUBTIPO,IMAGEN,TEXTO,(SYSDATE+1)
FROM TEMPORAL
WHERE tipo in (10,11,12,14,15);
INSERT INTO PERMUTAS(ID_PERMUTA,TIPO,SUBTIPO,IMAGEN,TEXTO,FECHA)
SELECT ID,TIPO,SUBTIPO,IMAGEN,TEXTO,(SYSDATE+1)

```

```
FROM TEMPORAL
WHERE tipo in (36,37);
```

```
INSERT INTO EMPLEOS(ID_EMPLEO, TIPO, SUBTIPO, IMAGEN, TEXTO, FECHA)
SELECT ID, TIPO, SUBTIPO, IMAGEN, TEXTO, (SYSDATE+1)
FROM TEMPORAL
WHERE tipo in (16,17,18,19,20,21,22,23);
```

```
INSERT INTO COMPUTO(ID_COMPUTO, TIPO, SUBTIPO, IMAGEN, TEXTO, FECHA)
SELECT ID, TIPO, SUBTIPO, IMAGEN, TEXTO, (SYSDATE+1)
FROM TEMPORAL
WHERE tipo in (43,47);
```

```
INSERT INTO VARIOS(ID_VARIOS, TIPO, SUBTIPO, IMAGEN, TEXTO, FECHA)
SELECT ID, TIPO, SUBTIPO, IMAGEN, TEXTO, (SYSDATE+1)
FROM TEMPORAL
WHERE tipo in (24,25,26,33,42,44,45,46,49,50,51,52,53);
```

```
INSERT INTO VEHICULOS(ID_VEHICULO, TIPO, SUBTIPO, IMAGEN, TEXTO, SUBMARCA, FECHA)
SELECT ID, TIPO, SUBTIPO, IMAGEN, TEXTO, SUBSTR(SUBTIPO,1,2), (SYSDATE+1)
FROM TEMPORAL
WHERE tipo in (38,48);
```

```
INSERT INTO SERVICIO(ID_SERVICIO, TIPO, SUBTIPO, IMAGEN, TEXTO, FECHA)
SELECT ID, TIPO, SUBTIPO, IMAGEN, TEXTO, (SYSDATE+1)
FROM TEMPORAL
WHERE tipo in (35,13,30);
```

```
DELETE FROM TEMPORAL
WHERE TIPO=34;
```

```
commit;
```

Archivo: cata_hoy.sql

Prepara la tabla de catálogos del día

```
declare
cursor c1 is select distinct(tipo) resulta from temporal
  where tipo !=34;
t_rec cata_tipo%ROWTYPE;
begin
delete from submarca_hoy;
delete from subtipo_hoy;
delete from tipo_hoy;
commit;
insert into tipo_hoy values(90,'Raiz',null,2);
insert into tipo_hoy values(91,'Negocios',90,2);
insert into tipo_hoy values(92,'Permutas',90,2);
insert into tipo_hoy values(93,'Servicios',90,2);
insert into tipo_hoy values(94,'Empleos',90,2);
insert into tipo_hoy values(95,'Bienes inmuebles',90,2);
insert into tipo_hoy values(96,'Equipo fotográfico y de cómputo',90,2);
insert into tipo_hoy values(97,'Varios',90,2);
```

```

insert into tipo_hoy values(98,'Vehiculos',90,2);
for cur in c1 loop
  select * into t_rec from cata_tipo
  where id_tipo=cur.resulta;
  insert into tipo_hoy values(t_rec.id_tipo, t_rec.titulo, t_rec.padre, t_rec.clasi);
end loop;
commit;
EXCEPTION
  WHEN NO_DATA_FOUND THEN
    rollback;
    CARGA_TIPO_HOY;
end;
/
--*****
--***** carga los demas valores de el catalogo
--*****

-- ***** Inmuebles ***
declare
cursor c1 is select distinct(tipo) tipo_a ,subtipo from inmuebles
where tipo in (1,3,4,40);
s_rec cata_subtipo%ROWTYPE;
begin
for cur in c1 loop
  select * into s_rec from cata_subtipo
  where id_subtipo=cur.subtipo and
  tipo=cur.tipo_a;
  insert into subtipo_hoy values(s_rec.id,s_rec.id_subtipo, s_rec.titulo, s_rec.tipo, s_rec.clasi);
end loop;
commit;
EXCEPTION
  WHEN NO_DATA_FOUND THEN
    rollback;
    CARGA_SUBTIPO_INMUEBLES;
end;
/
--***** Negocios no aplica porque no tiene subclasificaciones
--***** Permutas no aplica porque no tiene subclasificaciones
--***** Empleos *****
declare
cursor c1 is select distinct(tipo) tipo_a ,subtipo from empleos
where tipo in (18,19,20,23);
s_rec cata_subtipo%ROWTYPE;
begin
for cur in c1 loop
  select * into s_rec from cata_subtipo
  where id_subtipo=cur.subtipo and
  tipo=cur.tipo_a;
  insert into subtipo_hoy values(s_rec.id,s_rec.id_subtipo, s_rec.titulo, s_rec.tipo, s_rec.clasi);
end loop;
commit;
EXCEPTION
  WHEN NO_DATA_FOUND THEN
    rollback;
    CARGA_SUBTIPO_EMPLEOS;
end;

```

```

/
--***** Computo *****
declare
cursor c1 is select distinct(tipo) tipo_a ,subtipo from computo
where tipo in (47);
s_rec cata_subtipo%ROWTYPE;
begin
for cur in c1 loop
select * into s_rec from cata_subtipo
where id_subtipo=cur.subtipo and
tipo=cur.tipo_a;
insert into subtipo_hoy values(s_rec.id,s_rec.id_subtipo, s_rec.titulo, s_rec.tipo, s_rec.clasi);
end loop;
commit;
EXCEPTION
WHEN NO_DATA_FOUND THEN
rollback;
CARGA_SUBTIPO_COMPUTO;
end;
/
--***** varios *****
declare
cursor c1 is select distinct(tipo) tipo_a ,subtipo from varios
where tipo in (25,33,42,44);
s_rec cata_subtipo%ROWTYPE;
begin
for cur in c1 loop
select * into s_rec from cata_subtipo
where id_subtipo=cur.subtipo and
tipo=cur.tipo_a;
insert into subtipo_hoy values(s_rec.id,s_rec.id_subtipo, s_rec.titulo, s_rec.tipo, s_rec.clasi);
end loop;
commit;

EXCEPTION
WHEN NO_DATA_FOUND THEN
rollback;
CARGA_SUBTIPO_VARIOS;
end;
/
--*****Servicios*****
declare
cursor c1 is select distinct(tipo) tipo_a ,subtipo from servicio
where tipo in (35,30);
s_rec cata_subtipo%ROWTYPE;
begin
for cur in c1 loop
select * into s_rec from cata_subtipo
where id_subtipo=cur.subtipo and
tipo=cur.tipo_a;
insert into subtipo_hoy values(s_rec.id,s_rec.id_subtipo, s_rec.titulo, s_rec.tipo, s_rec.clasi);
end loop;
commit;
EXCEPTION
WHEN NO_DATA_FOUND THEN
rollback;

```

```

    CARGA_SUBTIPO_SERVICIO;
end;
/
--***** vehiculos *****
--**** llena la table de submarca ****
declare
cursor c1 is select distinct(tipo) tipo_a ,subtipo from vehiculos
where tipo in (38) and
    substr(subtipo,1,2) NOT in ('01','25','26','27','28','35','60','61','62','63','64');
s_sub cata_submarca%ROWTYPE;
begin
for cur in c1 loop
select * into s_sub from cata_submarca
where id_submarca=cur.subtipo and
    tipo=cur.tipo_a;
insert into submarca_hoy values(s_sub.id,s_sub.id_submarca, s_sub.titulo, s_sub.tipo,
s_sub.id_subtipo);
end loop;
commit;
EXCEPTION
WHEN NO_DATA_FOUND THEN
rollback;
CARGA_SUBTIPO_SUBMARCA;
end;
/
--***** llena la tabla de subtipo *****
declare
cursor c1 is select distinct(id_subtipo) subtipo_a from submarca_hoy;
s_rec cata_subtipo%ROWTYPE;
begin
for cur in c1 loop
select * into s_rec from cata_subtipo
where id_subtipo= cur.subtipo_a;
insert into subtipo_hoy values(s_rec.id,s_rec.id_subtipo, s_rec.titulo, s_rec.tipo, s_rec.clasi);
end loop;
commit;
end;
/
--***** llena con los demas valores subtipo *****
declare
cursor c1 is select distinct(tipo) tipo_a ,subtipo from vehiculos
where tipo in (38) and
    substr(subtipo,1,2) in ('01','25','26','27','28','35','60','61','62','63','64');
s_rec cata_subtipo%ROWTYPE;
begin
for cur in c1 loop
select * into s_rec from cata_subtipo
where id_subtipo=cur.subtipo and
    tipo=cur.tipo_a;
insert into subtipo_hoy values(s_rec.id,s_rec.id_subtipo, s_rec.titulo, s_rec.tipo, s_rec.clasi);
end loop;
commit;
EXCEPTION
WHEN NO_DATA_FOUND THEN
rollback;
CARGA_SUBTIPO_VEHICULO;

```

```
end;
/

--***** Inmuebles
--*****carga seccion 5 y 39 de inmuebles *****
declare
cursor c1 is select distinct(tipo) tipo_a,substr(subtipo,1,2) subtipo_a from INMUEBLES
  where tipo in (5,39);
s_rec cata_subtipo%ROWTYPE;
begin
for cur in c1 loop
  select * into s_rec from cata_subtipo
  where id_subtipo=cur.subtipo_a and
  tipo=cur.tipo_a;
  insert into subtipo_hoy values(s_rec.id,s_rec.id_subtipo, s_rec.titulo, s_rec.tipo, s_rec.clasi);
end loop;
commit;
EXCEPTION
  WHEN NO_DATA_FOUND THEN
    rollback;
    CARGA_SUBTIPO_SEC5_39;
end;
/
commit;
CATA_HOY
```

ANEXO 3

Código fuente de los módulos correspondientes al Front End:

- Cómputo
- Empleados
- Negocios
- Permutas
- Servicios
- Varios
- Vivienda
- Autos (ver capítulo 3.4)

/** Código fuente del módulo Cómputo **/

```
CREATE OR REPLACE PACKAGE compu AS
```

```
    PROCEDURE forma_compu;
    PROCEDURE vali_compu (p_opcion in NUMBER, p_variable in VARCHAR2);
    PROCEDURE busca_compu (p_variable in VARCHAR2, p_opcion in NUMBER);
    PROCEDURE subforma_compu (p_opcion in NUMBER);
    PROCEDURE subbusca_compu(p_subopcion in NUMBER,p_subvariable in VARCHAR2 );
```

```
END compu;
```

```
/
```

```
CREATE OR REPLACE PACKAGE BODY compu AS
```

```
/**-----*/
```

```
PROCEDURE forma_compu IS
```

```
BEGIN
```

```
  htp.p('<html><head>
```

```
    <title>EQUIPO FOTOGRAFICO Y DE COMPUTO</title>
```

```
    <meta name="author" content="lweb S.A. de C.V.1998 " Calzada de las Aguilas 101
```

```
    tel 5936110 y 5936233 Mxico D.F."></head>
```

```
    <body bgcolor="#FFFFFF"><div align="CENTER">
```

```
    <table border="0" width="468">
```

```
    <tr>
```

```
      <td colspan="2"><font color="#666666"></font><font color="#666666">
```

```
      </font></td>
```

```
    </tr>
```

```
    <tr>
```

```
      <td colspan="2"></td>
```

```
    </tr>
```

```
    </table></div>
```

```
      <CENTER><FONT COLOR="#000000" face="Arial, Helvetica, sans-serif" SIZE=2><B>Equipo
Fotográfico y de Cómputo</B></FONT></CENTER>
```

```
      <CENTER><table border="0" width="500"><TR><TD>
```

```
        <FONT COLOR="#000000" SIZE=2>Seleccione un tema de su interés, aquellos temas marcados
con ** contienen subtemas,
```

```

(los cuales pueden verse sólo dando click en el botón BUSCAR)</TD></TR></FONT>
</table></CENTER>);
http.formOpen('compu.vali_compu');
http.p('<CENTER>
<table cellspacing="0">
  <tr><td><FONT COLOR="#000000" face="Arial, Helvetica, sans-serif" SIZE=2>Tema de
busqueda :</td><td>);
  http.formSelectOpen(cname=>'p_opcion');
  for cur_op in (SELECT TITULO,ID_TIPO FROM tipo_hoy WHERE padre=96) LOOP

  http.FormSelectOption(cur_op.titulo,'SELECTED',cattributes=>'VALUE="||cur_op.id_tipo||"');
  END LOOP;
  http.p('</select></FONT></td></tr>
  <tr><td width="500" colspan="2"><FONT COLOR="#000000" SIZE=2>Si desea ser más
específico, escriba sus opciones en la caja de "Palabras a Buscar"</font></td></tr>
  <tr><td>
  <FONT COLOR="#000000" face="Arial, Helvetica, sans-serif" SIZE=2>Palabras a
buscar:</font></td>
  <td><input type="text" size="30" name="p_variable">
  </td></tr></table><CENTER>
  <FONT COLOR="#000000" face="Arial, Helvetica, sans-serif" SIZE=2><input
type="submit" value="Buscar"></font>
  <FONT COLOR="#000000" face="Arial, Helvetica, sans-serif" SIZE=2><input
type="reset" value="Borrar"></font></CENTER>);
http.formClose;
http.p('<div align="CENTER">
<table border="0" width="280" cellspacing="0" cellpadding="0" vspace="0" hspace="0">
<tr><td><div align="CENTER">
  <a href="http://ns1.avis-oportuno.com.mx" target="_top">
  </a>
</div></td>
<td><div align="CENTER">
  <a href="mailto:contacta@aguila.el-universal.com.mx"></a>
</div></td>
<td><div align="CENTER">
  <a href="/uni/index.html" target="_top"></a>
</div></td>
<td><div align="CENTER">
  <a href="/uni/pages/mapasitio.htm" target="main"></a>
</div></td>
<td><div align="CENTER">
  <a href="busca.forma_busca" target="main"></a>
</div></td>
<td><div align="CENTER">
  <a href="/uni/pages/ayuda.htm" target="main"></a>
</div></td>
</tr>
<tr><td><div align="CENTER">
  <a href="http://ns1.avis-oportuno.com.mx" target="_top">
  <font size="2" face="Arial, Helvetica, sans-serif">Regreso</font></a>
</div></td>

```

```

        <td><div align="CENTER">
          <font size="2" face="Arial, Helvetica, sans-serif"><a href="mailto:contacta@aguila.el-
universal.com.mx">Cont&aacute;ctanos</a></font>
        </div></td>
        <td><div align="CENTER">
          <font face="Arial, Helvetica, sans-serif" size="2"><a href="/uni/index.html"
target="_top">Inicio</a></font>
        </div></td>
        <td><div align="CENTER">
          <font face="Arial, Helvetica, sans-serif" size="2"><a href="/uni/pages/mapasitio.htm"
target="main">Mapa</a></font>
        </div></td>
        <td><div align="CENTER">
          <font face="Arial, Helvetica, sans-serif" size="2"><a href="busca.forma_busca"
target="main">B&uacute;squeda</a></font>
        </div></td>
        <td><div align="CENTER">
          <font size="2" face="Arial, Helvetica, sans-serif"><a href="/uni/pages/ayuda.htm"
target="main">Ayuda</a></font>
        </div>
      </td></tr></table></div>');
      oraclelogo;
      http.p('</BODY></HTML>');
END forma_compu;

```

```

/*****/
PROCEDURE vali_compu (p_opcion in NUMBER, p_variable in VARCHAR2) IS

```

```

v_clasi tipo_hoy.clasi%TYPE;

```

```

BEGIN

```

```

  select clasi into v_clasi from tipo_hoy where id_tipo=p_opcion;

```

```

  IF (v_clasi=1) or ((v_clasi=2) and (p_variable is NOT NULL)) THEN

```

```

    compu.busca_compu(p_variable, p_opcion);

```

```

  ELSE

```

```

    compu.subforma_compu(p_opcion);

```

```

  END IF;

```

```

END vali_compu;

```

```

/*****/
PROCEDURE busca_compu(p_variable in VARCHAR2, p_opcion in NUMBER) IS

```

```

score char(5);

```

```

pk char(5);

```

```

curid number;

```

```

texto varchar2(4000);

```

```

v_contador NUMBER:=0;

```

```

v_titulo tipo_hoy.titulo%TYPE;

```

```

CURSOR cur1 IS

```

```

  SELECT texto

```

```

  FROM computo

```

```

  WHERE tipo=p_opcion;

```

```

BEGIN

```

```

select titulo into v_titulo from tipo_hoy where id_tipo=p_opcion;
IF (p_variable is NOT NULL) THEN
  dbms_output.enable(1000000);
  curid := ctx_query.open_con(policy_name => 'poli_computo',
    text_query => p_variable,
    score_sorted => true,
    other_cols => 'texto',
    struct_query => 'tipo=||p_opcion||');
END IF;
http.p('<html><head>
<title>EQUIPO FOTOGRAFICO Y DE COMPUTO</title>
<meta name="author" content="lweb S.A. de C.V.1998 " Calzada de las Aguilas 101
tel 5936110 y 5936233 M□xico D.F."></head>
<body bgcolor="#FFFFFF"><div align="CENTER">
<table border="0" width="468">
<tr>
<td colspan="2"><font color="#666666"></font><font color="#666666">
</font></td>
</tr>
<tr>
<td colspan="2"></td>
</tr>
</table></div>');
http.p('<center>
<BR>Tema:
<FONT SIZE=3
COLOR="#000000"><B>||v_titulo||</B></FONT><BR><BR>');
http.tableOpen;
IF (p_variable is NOT NULL) THEN
  http.p('Palabra: <FONT SIZE=3 COLOR="#FF2211"><B>||p_variable||</B></FONT><BR>');
  while (ctx_query.fetch_hit(curid,pk,score,texto)>0)
  loop
    http.p('<TR><TD><IMG SRC = "/uni/images/fig13.gif"><BR></TD><TD>||texto||</TD></TR>');
    v_contador:=v_contador+1;
  end loop;
  ctx_query.close_con(curid);
ELSE
  FOR i IN cur1 loop
    http.p('<TR><TD><IMG SRC = "/uni/images/fig13.gif"><BR></TD><TD>||i.texto||</TD></TR>');
  END LOOP;
END IF;
http.tableClose;
http.p('<BR>Número de Registros Encontrados:
<FONT SIZE=4 COLOR="#FF2211"><B>
||v_contador||</B> </FONT><br>');
http.p('<br><div align="CENTER">
<table border="0" width="280" cellspacing="0" cellpadding="0" vspace="0" hspace="0">
<tr><td><div align="CENTER">
<a href="http://ns1.aviso-oportuno.com.mx/noti/plsql/compu.forma_compu">
</a>
</div></td>
<td><div align="CENTER">
<a
href="mailto:contacta@aguila.el-universal.com.mx"></a>
</div></td>
</tr></table>');

```

```

<td><div align="CENTER">
  <a href="/uni/index.html" target="_top"></a>
</div></td>
<td><div align="CENTER">
  <a href="/uni/pages/mapasitio.htm" target="main"></a>
</div></td>
<td><div align="CENTER">
  <a href="busca.forma_busca" target="main"></a>
</div></td>
<td><div align="CENTER">
  <a href="/uni/pages/ayuda.htm" target="main"></a>
</div></td>
</tr>
<tr><td><div align="CENTER">
  <a href="http://ns1.avisooportuno.com.mx/noti/plsql/compu.forma_compu">
  <font size="2" face="Arial, Helvetica, sans-serif">Regreso</font></a>
</div></td>
  <td><div align="CENTER">
  <font size="2" face="Arial, Helvetica, sans-serif"><a href="mailto:contacta@aguila.el-
universal.com.mx">Contactanos</a></font>
</div></td>
  <td><div align="CENTER">
  <font face="Arial, Helvetica, sans-serif" size="2"><a href="/uni/index.html"
target="_top">Inicio</a></font>
</div></td>
  <td><div align="CENTER">
  <font face="Arial, Helvetica, sans-serif" size="2"><a href="/uni/pages/mapasitio.htm"
target="main">Mapa</a></font>
</div></td>
  <td><div align="CENTER">
  <font face="Arial, Helvetica, sans-serif" size="2"><a href="busca.forma_busca"
target="main">Búsqueda</a></font>
</div></td>
  <td><div align="CENTER">
  <font size="2" face="Arial, Helvetica, sans-serif"><a href="/uni/pages/ayuda.htm"
target="main">Ayuda</a></font>
</div>
</td></tr></table></div>);
oraclelogo;
http.p('</BODY></HTML>');
END busca_compu;

```

```

/*****

```

```

PROCEDURE subforma_compu (p_opcion in NUMBER)IS

```

```

v_titulo subtipo_hoy.titulo%TYPE;

```

```

BEGIN

```

```

select titulo into v_titulo from tipo_hoy where id_tipo=p_opcion;

```

```

http.p('<html><head>

```

```

<title>EQUIPO FOTOGRAFICO Y DE COMPUTO</title>

```

```

<meta name="author" content="lweb S.A. de C.V.1998 " Calzada de las Aguilas 101
tel 5936110 y 5936233 Mexico D.F."></head>

```

```

<body bgcolor="#FFFFFF"><div align="CENTER">

```

```

<table border="0" width="468">
<tr>
<td colspan="2"><font color="#666666"></font><font color="#666666">
</font></td>
</tr>
<tr>
<td colspan="2"></td>
</tr>
</table></div>
<CENTER><FONT COLOR="#000000" face="Arial, Helvetica, sans-serif"
SIZE=2><B>||v_titulo||</B></FONT></CENTER>
<CENTER><table border="0" width="500"><TR><TD>
<FONT COLOR="#000000" SIZE=2>Seleccione un tema de su interés, aquellos temas marcados
con "*" contienen subtemas,
(los cuales pueden verse sólo dando click en el botón BUSCAR)</TD></TR></FONT>
</table></CENTER>');
http.formOpen('compu.subbusca_compu');
http.p(' <CENTER>
<table cellpadding="0">
<tr><td><FONT COLOR="#000000" face="Arial, Helvetica, sans-serif" SIZE=2>Tema de
busqueda :</td><td>');
http.formSelectOpen(cname=>'p_subopcion');
for cur_opp in (SELECT TITULO,ID FROM subtipo_hoy WHERE tipo=p_opcion)
LOOP

http.FormSelectOption(cur_opp.titulo,'SELECTED',cattributes=>'VALUE="||cur_opp.id||"');
END LOOP;
http.p('</select></FONT></td></tr>
<tr><td width="500" colspan="2"><FONT COLOR="#000000" SIZE=2>Si desea ser más
específico, escriba sus opciones en la caja de "Palabras a Buscar"</font></td></tr>
<tr><td>
<FONT COLOR="#000000" face="Arial, Helvetica, sans-serif" SIZE=2>Palabras a
buscar:</font></td>
<td><input type="text" size="30" name="p_subvariable">
</td></tr></table><CENTER>
<FONT COLOR="#000000" face="Arial, Helvetica, sans-serif" SIZE=2><input
type="submit" value="Buscar"></font>
<FONT COLOR="#000000" face="Arial, Helvetica, sans-serif" SIZE=2><input
type="reset" value="Borrar"></font></CENTER>');
http.formClose;
http.p('<div align="CENTER">
<table border="0" width="280" cellpadding="0" cellspacing="0" vspace="0" hspace="0">
<tr><td><div align="CENTER">
<a href="compu.forma_compu">
</a>
</div></td>
<td><div align="CENTER">
<a href="mailto:contacta@aguila.el-universal.com.mx"></a>
</div></td>
<td><div align="CENTER">
<a href="/uni/index.html" target="_top"></a>
</div></td>
<td><div align="CENTER">

```

```

        <a href="/uni/pages/mapasitio.htm" target="main"></a>
    </div></td>
    <td align="CENTER">
        <a href="busca.forma_busca" target="main"></a>
    </div></td>
    <td align="CENTER">
        <a href="/uni/pages/ayuda.htm" target="main"></a>
    </div></td>
</tr>
<tr><td align="CENTER">
    <a href="compu.forma_compu">
    <font size="2" face="Arial, Helvetica, sans-serif">Regreso</font></a>
    </div></td>
    <td align="CENTER">
    <font size="2" face="Arial, Helvetica, sans-serif"><a href="mailto:contacta@aguila.el-
universal.com.mx">Contactanos</a></font>
    </div></td>
    <td align="CENTER">
    <font face="Arial, Helvetica, sans-serif" size="2"><a href="/uni/index.html"
target="_top">Inicio</a></font>
    </div></td>
    <td align="CENTER">
    <font face="Arial, Helvetica, sans-serif" size="2"><a href="/uni/pages/mapasitio.htm"
target="main">Mapa</a></font>
    </div></td>
    <td align="CENTER">
    <font face="Arial, Helvetica, sans-serif" size="2"><a href="busca.forma_busca"
target="main">Búsqueda</a></font>
    </div></td>
    <td align="CENTER">
    <font size="2" face="Arial, Helvetica, sans-serif"><a href="/uni/pages/ayuda.htm"
target="main">Ayuda</a></font>
    </div>
</td></tr></table></div>');
    oracielogo;
    http.p('</BODY></HTML>');
END subforma_compu;
/*****/
PROCEDURE subbusca_compu(p_subopcion in NUMBER,p_subvariable in VARCHAR2) IS

```

```

score char(5);
pk char(5);
curid number;
texto varchar2(4000);
v_contador NUMBER:=0;
v_tipo subtipo_hoy.tipo%TYPE;
v_id_subtipo subtipo_hoy.id_subtipo%TYPE;
v_titulo subtipo_hoy.titulo%TYPE;

```

```

CURSOR cur1 IS
SELECT texto,tipo,subtipo
FROM computo
WHERE subtipo = (select id_subtipo from subtipo_hoy where id=p_subopcion)

```

```

AND tipo = (select tipo from subtipo_hoy where id=p_subopcion);

BEGIN
SELECT tipo,id_subtipo,titulo
into v_tipo,v_id_subtipo,v_titulo from subtipo_hoy where id=p_subopcion;

IF (p_subvariable is NOT NULL) THEN
  curid := ctx_query.open_con(policy_name => 'poli_computo',
    text_query => p_subvariable,
    score_sorted => true,
    other_cols => 'texto',
    struct_query => 'tipo='||v_tipo||' and subtipo='''||v_id_subtipo||''');
END IF;

htp.p('<html><head>
<title>EQUIPO FOTOGRAFICO Y DE COMPUTO</title>
<meta name="author" content="Iweb S.A. de C.V.1998 " Calzada de las Aguilas 101
tel 5936110 y 5936233 Mxico D.F."></head>
<body bgcolor="#FFFFFF"><div align="CENTER">
<table border="0" width="468">
<tr><td colspan="2"><font color="#666666"></font> <font color="#666666">
</font></td>
</tr><tr>
<td colspan="2"></td>
</tr></table></div>');
htp.p('<center> <BR>Tema: <FONT SIZE=3
COLOR="#000000"><B>'||v_titulo||'</B></FONT><BR><BR>');
htp.tableOpen;
IF (p_subvariable is NOT NULL) THEN
  htp.p('Palabra: <FONT SIZE=3 COLOR="#FF2211">'||p_subvariable||'</FONT><BR>');
  while (ctx_query.fetch_hit(curid,pk,score,texto)>0)
  loop
    htp.p('<TR><TD><IMG SRC = "/uni/images/fig13.gif"><BR></TD><TD>'||texto||'</TD></TR>');
    v_contador:=v_contador+1;
  end loop;
  ctx_query.close_con(curid);
ELSE
  FOR i IN cur1 loop
    htp.p('<TR><TD><IMG SRC = "/images/fig13.gif"><BR></TD><TD>'||i.texto||'</TD></TR> ');
    v_contador:=v_contador+1;
  END LOOP;
END IF;
htp.tableClose;
htp.p('<BR>mero de Registros Encontrados: <FONT SIZE=4 COLOR="#FF2211"><B>
' ||v_contador||'</B> </FONT><br>');
htp.p('<br><div align="CENTER">
<table border="0" width="260" cellspacing="0" cellpadding="0" vspace="0" hspace="0">
<tr><td><div align="CENTER">
<a href="compu.subforma_compu?p_opcion='||v_tipo||'">
</a>
</div></td>
<td><div align="CENTER">
<a href="mailto:contacta@aguila.el-universal.com.mx"></a>
</div></td>

```

```

        <td><div align="CENTER">
          <a href="/uni/index.html" target="_top"></a>
        </div></td>
        <td><div align="CENTER">
          <a href="/uni/pages/mapasitio.htm" target="main"></a>
        </div></td>
        <td><div align="CENTER">
          <a href="busca.forma_busca" target="main"></a>
        </div></td>
        <td><div align="CENTER">
          <a href="/uni/pages/ayuda.htm" target="main"></a>
        </div></td>
      </tr>
    <tr><td><div align="CENTER">
      <a href="compu.subforma_compu?p_opcion=||v_tipo||">
        <font size="2" face="Arial, Helvetica, sans-serif">Regreso</font></a>
      </div></td>
    <td><div align="CENTER">
      <font size="2" face="Arial, Helvetica, sans-serif"><a href="mailto:contacta@aguila.el-
universal.com.mx">Cont&acute;tanos</a></font>
    </div></td>
    <td><div align="CENTER">
      <font face="Arial, Helvetica, sans-serif" size="2"><a href="/uni/index.htm"
target="_top">Inicio</a></font>
    </div></td>
    <td><div align="CENTER">
      <font face="Arial, Helvetica, sans-serif" size="2"><a href="/uni/pages/mapasitio.htm"
target="main">Mapa</a></font>
    </div></td>
    <td><div align="CENTER">
      <font face="Arial, Helvetica, sans-serif" size="2"><a href="busca.forma_busca"
target="main">B&uacute;squeda</a></font>
    </div></td>
    <td><div align="CENTER">
      <font size="2" face="Arial, Helvetica, sans-serif"><a href="/uni/pages/ayuda.htm"
target="main">Ayuda</a></font>
    </div>
  </td></tr></table></div>);
  oraclelogo;
  http.p('</BODY></HTML>');
END subbusca_compu;

END compu;
/

```

/ C3digo fuente del m3dulo Empleados**/**

CREATE OR REPLACE PACKAGE empleado AS

```

  PROCEDURE forma_empleado;
  PROCEDURE vali_empleados (p_opcion in NUMBER, p_variable in VARCHAR2);

```

```

PROCEDURE busca_emp (p_variable in VARCHAR2, p_opcion in NUMBER);
PROCEDURE subforma_emp (p_opcion in NUMBER);
PROCEDURE subbusca_emp(p_subopcion in NUMBER,p_subvariable in VARCHAR2 );

```

```

END empleado;
/

```

```

CREATE OR REPLACE PACKAGE BODY empleado AS
/*****

```

```

PROCEDURE forma_empleado IS

```

```

BEGIN

```

```

http.p('<HTML><head><title>EMPLEOS</title>

```

```

<meta name="author" content="Iweb S.A. de C.V.1998 " Calzada de las Aguilas 101

```

```

tel 5936110 y 5936233 MZxico D.F."></head>

```

```

<body bgcolor="#FFFFFF"><div align="CENTER">

```

```

<table border="0" width="468">

```

```

<tr>

```

```

<td colspan="2"><font color="#666666"></font><font color="#666666">

```

```

</font></td>

```

```

</tr>

```

```

<tr>

```

```

<td colspan="2"></td>

```

```

</tr>

```

```

</table></div>

```

```

<CENTER><FONT COLOR="#339900" face="Arial, Helvetica, sans-serif"
SIZE=2><B>Empleos</B></FONT></CENTER>

```

```

<CENTER><table border="0" width="500"><TR><TD>

```

```

<FONT COLOR="#000000" SIZE=2>Seleccione un tema de su interés, aquellos temas marcados
con ** contienen subtemas,

```

```

(los cuales pueden verse sólo dando click en el botón BUSCAR)</TD></TR></FONT>

```

```

</table></CENTER>');

```

```

http.formOpen('empleado.vali_empleados');

```

```

http.p('<CENTER>

```

```

<table cellspacing="0">

```

```

<tr><td><FONT COLOR="#000000" face="Arial, Helvetica, sans-serif" SIZE=2>Tema de
busqueda :</td><td>');

```

```

http.formSelectOpen(cname=>'p_opcion');

```

```

FOR cur_op IN (SELECT TITULO,ID_TIPO FROM tipo_hoy WHERE padre=94) LOOP

```

```

http.FormSelectOption(cur_op.titulo,'SELECTED',cattributes=>'VALUE=""||cur_op.id_tipo||"');

```

```

END LOOP;

```

```

http.p('</select></FONT></td></tr>

```

```

<tr><td width="500" colspan="2"><FONT COLOR="#000000" SIZE=2>Si desea ser más
especifico, escriba sus opciones en la caja de "Palabras a Buscar"</font></td></tr>

```

```

<tr><td>

```

```

<FONT COLOR="#000000" face="Arial, Helvetica, sans-serif" SIZE=2>Palabras a
buscar:</font></td>

```

```

<td><input type="text" size="30" name="p_variable">

```

```

</td></tr></table><CENTER>

```

```

<FONT COLOR="#000000" face="Arial, Helvetica, sans-serif" SIZE=2><input

```

```

type="submit" value="Buscar"></font>

```

```

<FONT COLOR="#000000" face="Arial, Helvetica, sans-serif" SIZE=2><input

```

```

type="reset" value="Borrar"></font></CENTER>');

```

```

http.formClose;

```

```

http.p('<div align="CENTER">

```

```

<table border="0" width="280" cellspacing="0" cellpadding="0" vspace="0" hspace="0">

```

```

<tr><td><div align="CENTER">

```

```

    <a href="http://ns1.avis-oportuno.com.mx" target="_top">
    </a>
</div></td>
<td><div align="CENTER">
    <a href="mailto:contacta@aguila.el-universal.com.mx"></a>
</div></td>
<td><div align="CENTER">
    <a href="/uni/index.html" target="_top"></a>
</div></td>
<td><div align="CENTER">
    <a href="/uni/pages/mapasitio.htm" target="main"></a>
</div></td>
<td><div align="CENTER">
    <a href="busca.forma_busca" target="main"></a>
</div></td>
<td><div align="CENTER">
    <a href="/uni/pages/ayuda.htm" target="main"></a>
</div></td>
</tr>
<tr><td><div align="CENTER">
    <a href="http://ns1.avis-oportuno.com.mx" target="_top">
    <font size="2" face="Arial, Helvetica, sans-serif">Regreso</font></a>
</div></td>
<td><div align="CENTER">
    <font size="2" face="Arial, Helvetica, sans-serif"><a href="mailto:contacta@aguila.el-
universal.com.mx">Cont&acute;tanos</a></font>
</div></td>
<td><div align="CENTER">
    <font face="Arial, Helvetica, sans-serif" size="2"><a href="/uni/index.html"
target="_top">Inicio</a></font>
</div></td>
<td><div align="CENTER">
    <font face="Arial, Helvetica, sans-serif" size="2"><a href="/uni/pages/mapasitio.htm"
target="main">Mapa</a></font>
</div></td>
<td><div align="CENTER">
    <font face="Arial, Helvetica, sans-serif" size="2"><a href="busca.forma_busca"
target="main">B&uacute;squeda</a></font>
</div></td>
<td><div align="CENTER">
    <font size="2" face="Arial, Helvetica, sans-serif"><a href="/uni/pages/ayuda.htm"
target="main">Ayuda</a></font>
</div>
</td></tr></table></div>);
oraclelogo;
http.p('</BODY></HTML>');

```

END forma_empleado;

/******

PROCEDURE vali_empleados (p_opcion in NUMBER, p_variable in VARCHAR2) IS

v_clasi tipo_hoy.clasi%TYPE;

```

BEGIN
  select clasi into v_clasi from tipo_hoy where id_tipo=p_opcion;

  IF (v_clasi=1) or ((v_clasi=2) and (p_variable is NOT NULL)) THEN
    empleado.busca_emp(p_variable, p_opcion);
  ELSE
    empleado.subforma_emp(p_opcion);
  END IF;

END vali_empleados;

/*****
PROCEDURE busca_emp(p_variable in VARCHAR2, p_opcion in NUMBER) IS

score char(5);
pk char(5);
curid number;
texto varchar2(4000);
v_contador NUMBER:=0;
v_titulo tipo_hoy.titulo%TYPE;

CURSOR cur1 IS
  SELECT texto
  FROM empleos
  WHERE tipc=p_opcion;

BEGIN
  select titulo into v_titulo from tipo_hoy where id_tipo=p_opcion;

  IF (p_variable is NOT NULL) THEN
    dbms_output.enable(1000000);
    curid := ctx_query.open_con(policy_name => 'poli_empleos',
      text_query => p_variable,
      score_sorted => true,
      other_cols => 'texto',
      struct_query => 'tipo=||p_opcion||');
  END IF;
  http.p ('<HTML><head><title>EMPLEOS</title>
  <meta name="author" content="Iweb S.A. de C.V.1998 " Calzada de las Aguilas 101
  tel 5936110 y 5936233 MZxico D.F."></head>
  <body bgcolor="#FFFFFF"><div align="CENTER">
  <table border="0" width="468">
  <tr>
  <td colspan="2"><font color="#666666"></font><font color="#666666">
  </font></td>
  </tr>
  <tr>
  <td colspan="2"></td>
  </tr>
  </table></div>');
  http.p ('<center><BR>Tema: <FONT COLOR="#339900" SIZE=3><B>'||v_titulo||</B></FONT><BR>');
  http.tableOpen;

  IF (p_variable is NOT NULL) THEN
    http.p('Palabra: <FONT SIZE=3 COLOR="#FF2211"><B>'||p_variable||</B></FONT><BR>');

```

```

while (ctx_query.fetch_hit(curid,pk,score,texto)>0)
loop
    http.p('<TR><TD><IMG SRC = "/uni/images/fig13.gif"><BR></TD><TD>||texto||</TD></TR>
);
    v_contador:=v_contador+1;
end loop;
ctx_query.close_con(curid);
ELSE
FOR i IN cur1 loop
    http.p('<TR><TD><IMG SRC = "/uni/images/fig13.gif"><BR></TD><TD>||i.texto||</TD></TR>
);
    v_contador:=v_contador+1;
END LOOP;
END IF;
http.tableClose;
http.p('<BR>Número de Registros Encontrados:<FONT SIZE=4 COLOR="#FF2211"><B>
'||v_contador||</B></FONT><BR>');
http.p('<br><div align="CENTER">
<table border="0" width="280" cellspacing="0" cellpadding="0" vspace="0" hspace="0">
<tr><td><div align="CENTER">
    <a href="http://ns1.aviso-oportuno.com.mx/noti/plsql/empleado.forma_empleado">
    </a>
</div></td>
<td><div align="CENTER">
    <a href="mailto:contacta@aguila.el-universal.com.mx"></a>
</div></td>
<td><div align="CENTER">
    <a href="/uni/index.html" target="_top"></a>
</div></td>
<td><div align="CENTER">
    <a href="/uni/pages/mapasitio.htm" target="main"></a>
</div></td>
<td><div align="CENTER">
    <a href="busca.forma_busca" target="main"></a>
</div></td>
<td><div align="CENTER">
    <a href="/uni/pages/ayuda.htm" target="main"></a>
</div></td>
</tr>
<tr><td><div align="CENTER">
    <a href="http://ns1.aviso-oportuno.com.mx/noti/plsql/empleado.forma_empleado">
    <font size="2" face="Arial, Helvetica, sans-serif">Regreso</font></a>
</div></td>
<td><div align="CENTER">
    <font size="2" face="Arial, Helvetica, sans-serif"><a href="mailto:contacta@aguila.el-
universal.com.mx">Cont&acutecanos</a></font>
</div></td>
<td><div align="CENTER">
    <font face="Arial, Helvetica, sans-serif" size="2"><a href="/uni/index.html"
target="_top">Inicio</a></font>
</div></td>

```

```

        <td><div align="CENTER">
          <font face="Arial, Helvetica, sans-serif" size="2"><a href="/uni/pages/mapasitio.htm"
target="main">Mapa</a></font>
        </div></td>
        <td><div align="CENTER">
          <font face="Arial, Helvetica, sans-serif" size="2"><a href="busca.forma_busca"
target="main">B&uacute;squeda</a></font>
        </div></td>
        <td><div align="CENTER">
          <font size="2" face="Arial, Helvetica, sans-serif"><a href="/uni/pages/ayuda.htm"
target="main">Ayuda</a></font>
        </div>
      </td></tr></table></div>');
      oraclelogo:
      http.p('</BODY></HTML>');

```

END busca_emp;

```

/*****
PROCEDURE subforma_emp (p_opcion in NUMBER)IS

```

```

v_titulo  subtipo_hoy.titulo%TYPE;

```

```

BEGIN

```

```

  select titulo into v_titulo from tipo_hoy where id_tipo=p_opcion;

```

```

  http.p('<HTML><head><title>EMPLEOS</title>
    <meta name="author" content="Iweb S.A. de C.V.1998 " Calzada de las Aguilas 101 tel 5936110 y
5936233 MZxico D.F."></head>
    <body bgcolor="#FFFFFF"><div align="CENTER">
      <TABLE border="0" width="468">
        <tr>
          <td colspan="2"><font color="#666666"></font><font color="#666666">
            </font></td>
        </tr>
        <tr>
          <td colspan="2"></td>
        </tr>
      </table></div>
      <CENTER><FONT COLOR="#339900" face="Arial, Helvetica, sans-serif"
SIZE=2><B>||v_titulo||</B></FONT></CENTER>
      <CENTER><table border="0" width="500"><TR><TD>
        <FONT COLOR="#000000" SIZE=2>Seleccione un tema de su interés, aquellos temas marcados
con ** contienen subtemas,
        (los cuales pueden verse sólo dando click en el botón BUSCAR)</TD></TR></FONT>
      </table></CENTER>');
      http.formOpen('empleado.subbusca_emp');
      http.p(' <CENTER>
        <table cellspacing="0">
          <tr><td><FONT COLOR="#000000" face="Arial, Helvetica, sans-serif" SIZE=2>Tema de
busqueda :</td><td>');
      http.formSelectOpen(cname=>'p_subopcion');
      FOR cur_op IN (SELECT TITULO,ID FROM subtipo_hoy WHERE tipo=p_opcion) LOOP
      http.FormSelectOption(cur_op.titulo,'SELECTED',cattributes=>'VALUE=""|cur_op.id||"');
      END LOOP;

```

```

htp.p('</select></FONT></td></tr>
      <tr><td width="500" colspan="2"><FONT COLOR="#000000" SIZE=2>Si desea ser más
específico, escriba sus opciones en la caja de "Palabras a Buscar"</font></td></tr>
      <tr><td>
        <FONT COLOR="#000000" face="Arial, Helvetica, sans-serif" SIZE=2>Palabras a
buscar:</font></td>
      <td><input type="text" size="30" name="p_subvariable">
</td></tr></table><CENTER>
<FONT COLOR="#000000" face="Arial, Helvetica, sans-serif" SIZE=2><input
type="submit" value="Buscar"></font>
      <FONT COLOR="#000000" face="Arial, Helvetica, sans-serif" SIZE=2><input
type="reset" value="Borrar"></font></CENTER>');
htp.formClose;
htp.p('<div align="CENTER">
  <table border="0" width="280" cellspacing="0" cellpadding="0" vspace="0" hspace="0">
  <tr><td><div align="CENTER">
    <a href="empleado.forma_empleado">
      </a>
    </div></td>
    <td><div align="CENTER">
      <a href="mailto:contacta@aguila.el-universal.com.mx"></a>
    </div></td>
    <td><div align="CENTER">
      <a href="/uni/index.html" target="_top"></a>
    </div></td>
    <td><div align="CENTER">
      <a href="/uni/pages/mapasitio.htm" target="main"></a>
    </div></td>
    <td><div align="CENTER">
      <a href="busca.forma_busca" target="main"></a>
    </div></td>
    <td><div align="CENTER">
      <a href="/uni/pages/ayuda.htm" target="main"></a>
    </div></td>
  </tr>
  <tr><td><div align="CENTER">
    <a href="empleado.forma_empleado">
      <font size="2" face="Arial, Helvetica, sans-serif">Regreso</font></a>
    </div></td>
    <td><div align="CENTER">
      <font size="2" face="Arial, Helvetica, sans-serif"><a href="mailto:contacta@aguila.el-
universal.com.mx">Cont&acute;tanos</a></font>
    </div></td>
    <td><div align="CENTER">
      <font face="Arial, Helvetica, sans-serif" size="2"><a href="/uni/index.html"
target="_top">Inicio</a></font>
    </div></td>
    <td><div align="CENTER">
      <font face="Arial, Helvetica, sans-serif" size="2"><a href="/uni/pages/mapasitio.htm"
target="main">Mapa</a></font>
    </div></td>
  </tr>
  </table>
</div>

```

```

<td><div align="CENTER">
  <font face="Arial, Helvetica, sans-serif" size="2"><a href="busca.forma_busca"
target="main">B&uacute;squeda</a></font>
</div></td>
<td><div align="CENTER">
  <font size="2" face="Arial, Helvetica, sans-serif"><a href="/uni/pages/ayuda.htm"
target="main">Ayuda</a></font>
</div>
</td></tr></table></div>');
oraclelogo;
htp.p('</BODY></HTML>');

```

END subforma_emp;

PROCEDURE subbusca_emp(p_subopcion in NUMBER,p_subvariable in VARCHAR2) IS

```

score char(5);
pk char(5);
curid number;
texto varchar2(4000);
v_contador NUMBER:=0;
v_tipo subtipo_hoy.tipo%TYPE;
v_id_subtipo subtipo_hoy.id_subtipo%TYPE;
v_titulo subtipo_hoy.titulo%TYPE;

```

```

CURSOR cur1 IS
  SELECT texto,tipo,subtipo
  FROM empleos
  WHERE subtipo = (select id_subtipo from subtipo_hoy where id=p_subopcion)
  AND tipo = (select tipo from subtipo_hoy where id=p_subopcion);

```

```

BEGIN
  SELECT tipo,id_subtipo,titulo
  into v_tipo,v_id_subtipo,v_titulo from subtipo_hoy where id=p_subopcion;

```

```

IF (p_subvariable is NOT NULL) THEN
  curid := ctx_query.open_con(policy_name => 'poli_empleos',
  text_query => p_subvariable,
  score_sorted => true,
  other_cols => 'texto',
  struct_query => 'tipo=||v_tipo|| and subtipo="||v_id_subtipo||"');

```

END IF;

```

htp.p('<HTML><head><title>EMPLEOS</title>
<meta name="author" content="Iweb S.A. de C.V.1998 " Calzada de las Aguilas 101
tel 5936110 y 5936233 MZxico D.F."></head>
<body bgcolor="#FFFFFF"><div align="CENTER">
<table border="0" width="468">
  <tr><td colspan="2"><font color="#666666"></font><font color="#666666">
</font></td>
</tr><tr>
<td colspan="2"></td>
</tr></table></div>');
htp.p ('<center> <BR>Tema: <FONT COLOR="#339900"
SIZE=3><B>||v_titulo||</B></FONT><BR>');
htp.tableOpen;

```

```

IF (p_subvariable is NOT NULL) THEN
  http.p('Palabra: <FONT SIZE=3 COLOR="#FF2211">||p_subvariable||</FONT><BR>');
  while (ctx_query.fetch_hit(curid,pk,score,texto)>0)
    loop
      http.p('<TR><TD><IMG SRC = "/uni/images/fig13.gif"><BR></TD><TD>||texto||</TD></TR>
');
      v_contador:=v_contador+1;
    end loop;
  ctx_query.close_con(curid);
ELSE
  FOR i IN cur1 loop
    http.p('<TR><TD><IMG SRC = "/uni/images/fig13.gif"><BR></TD><TD>||i.texto||</TD></TR> ');
    v_contador:=v_contador+1;
  END LOOP;
END IF;
http.tableClose;
http.p('<BR>Número de Registros Encontrados:<FONT SIZE=4 COLOR="#FF2211"><B>
||v_contador||</B></FONT><BR>');
  http.p('<br><div align="CENTER">
<table border="0" width="280" cellspacing="0" cellpadding="0" vspace="0" hspace="0">
<tr><td><div align="CENTER">
  <a href="empleado.subforma_emp?p_opcion=||v_tipo||">
  </a>
</div></td>
<td><div align="CENTER">
  <a href="mailto:contacta@aguila.el-universal.com.mx"></a>
</div></td>
<td><div align="CENTER">
  <a href="/uni/index.html" target="_top"></a>
</div></td>
<td><div align="CENTER">
  <a href="/uni/pages/mapasitio.htm" target="main"></a>
</div></td>
<td><div align="CENTER">
  <a href="busca.forma_busca" target="main"></a>
</div></td>
<td><div align="CENTER">
  <a href="/uni/pages/ayuda.htm" target="main"></a>
</div></td>
</tr>
<tr><td><div align="CENTER">
  <a href="empleado.subforma_emp?p_opcion=||v_tipo||">
  <font size="2" face="Arial, Helvetica, sans-serif">Regreso</font></a>
</div></td>
<td><div align="CENTER">
  <font size="2" face="Arial, Helvetica, sans-serif"><a href="mailto:contacta@aguila.el-
universal.com.mx">Cont&acute;ctanos</a></font>
</div></td>
<td><div align="CENTER">
  <font face="Arial, Helvetica, sans-serif" size="2"><a href="/uni/index.html"
target="_top">Inicio</a></font>

```

```

    </div></td>
    <td><div align="CENTER">
      <font face="Arial, Helvetica, sans-serif" size="2"><a href="/uni/pages/mapasitio.htm"
target="main">Mapa</a></font>
    </div></td>
    <td><div align="CENTER">
      <font face="Arial, Helvetica, sans-serif" size="2"><a href="busca.forma_busca"
target="main">B&uacute;squeda</a></font>
    </div></td>
    <td><div align="CENTER">
      <font size="2" face="Arial, Helvetica, sans-serif"><a href="/uni/pages/ayuda.htm"
target="main">Ayuda</a></font>
    </div>
  </td></tr></table></div>');
  oraclelogo;
  http.p('</BODY></HTML>');

```

END subbusca_emp;

END empleado;

/

/** C3digo fuente del m3dulo Negocios **/

CREATE OR REPLACE PACKAGE nego AS

```

  PROCEDURE forma_nego;
  PROCEDURE vali_nego (p_opcion in NUMBER, p_variable in VARCHAR2);
  PROCEDURE busca_nego (p_variable in VARCHAR2, p_opcion in NUMBER);
  PROCEDURE subforma_nego (p_opcion in NUMBER);
  PROCEDURE subbusca_nego(p_subopcion in NUMBER,p_subvariable in VARCHAR2 );

```

END nego;

/

CREATE OR REPLACE PACKAGE BODY nego AS

/*****

PROCEDURE forma_nego IS

BEGIN

```

  http.p('<html><head><title>NEGOCIOS</title>
  <meta name="author" content="Iweb S.A. de C.V.1998 " Calzada de las Aguilas 101
  tel 5936110 y 5936233 MZxico D.F."></head>
  <body bgcolor="#FFFFFF"><div align="CENTER">
  <table border="0" width="468">
    <tr>
      <td colspan="2"><font color="#666666"></font><font color="#666666">
      </font></td>
    </tr>
    <tr>
      <td colspan="2"></td>
    </tr>
  </table></div>
  <CENTER><FONT COLOR="#663300" face="Arial, Helvetica, sans-serif"
  SIZE=2><B>Negocios</B></FONT></CENTER>
  <CENTER><table border="0" width="500"><TR><TD>

```

```

<FONT COLOR="#000000" SIZE=2>Seleccione un tema de su interés, aquellos temas marcados
con ** contienen subtemas,
  (los cuales pueden verse sólo dando click en el botón BUSCAR)</TD></TR></FONT>
</table></CENTER>');
http.formOpen('nego.vali_nego');
http.p('<CENTER>
  <table cellspacing="0">
    <tr><td><FONT COLOR="#000000" face="Arial, Helvetica, sans-serif" SIZE=2>Tema de
busqueda :</td><td>');
    http.formSelectOpen(cname=>'p_opcion');
      for cur_op in (SELECT TITULO,ID_TIPO FROM tipo_hoy WHERE padre=91) LOOP
http.FormSelectOption(cur_op.titulo,'SELECTED',cattributes=>'VALUE=""|cur_op.id_tipo||"');
      END LOOP;
      http.p('</select></FONT></td></tr>
    <tr><td width="500" colspan="2"><FONT COLOR="#000000" SIZE=2>Si desea ser más
especifico, escriba sus opciones en la caja de "Palabras a Buscar"</font></td></tr>
    <tr><td>
      <FONT COLOR="#000000" face="Arial, Helvetica, sans-serif" SIZE=2>Palabras a
buscar:</font></td>
      <td><input type="text" size="30" name="p_variable">
      </td></tr></table><BR><CENTER>
      <FONT COLOR="#000000" face="Arial, Helvetica, sans-serif" SIZE=2><input
type="submit" value="Buscar"></font>
      <FONT COLOR="#000000" face="Arial, Helvetica, sans-serif" SIZE=2><input
type="reset" value="Borrar"></font></CENTER>');
http.formClose;
http.p('<div align="CENTER">
  <table border="0" width="280" cellspacing="0" cellpadding="0" vspc="0" hspace="0">
    <tr><td><div align="CENTER">
      <a href="http://ns1.avis-oportuno.com.mx" target="_top">
      </a>
      </div></td>
      <td><div align="CENTER">
        <a href="mailto:contacta@aguila.el-universal.com.mx"></a>
        </div></td>
      <td><div align="CENTER">
        <a href="/uni/index.html" target="_top"></a>
        </div></td>
      <td><div align="CENTER">
        <a href="/uni/pages/mapasitio.htm" target="main"></a>
        </div></td>
      <td><div align="CENTER">
        <a href="busca.forma_busca" target="main"></a>
        </div></td>
      <td><div align="CENTER">
        <a href="/uni/pages/ayuda.htm" target="main"></a>
        </div></td>
      </tr>
    <tr><td><div align="CENTER">
      <a href="http://ns1.avis-oportuno.com.mx" target="_top">

```

```

        <font size="2" face="Arial, Helvetica, sans-serif">Regreso</font></a>
    </div></td>
    <td><div align="CENTER">
        <font size="2" face="Arial, Helvetica, sans-serif"><a href="mailto:contacta@aguila.el-
universal.com.mx">Cont&acute;ctanos</a></font>
    </div></td>
    <td><div align="CENTER">
        <font face="Arial, Helvetica, sans-serif" size="2"><a href="/uni/index.html"
target="_top">Inicio</a></font>
    </div></td>
    <td><div align="CENTER">
        <font face="Arial, Helvetica, sans-serif" size="2"><a href="/uni/pages/mapasitio.htm"
target="main">Mapa</a></font>
    </div></td>
    <td><div align="CENTER">
        <font face="Arial, Helvetica, sans-serif" size="2"><a href="busca.forma_busca"
target="main">B&uacute;squeda</a></font>
    </div></td>
    <td><div align="CENTER">
        <font size="2" face="Arial, Helvetica, sans-serif"><a href="/uni/pages/ayuda.htm"
target="main">Ayuda</a></font>
    </div>
</td></tr></table></div>');
    oraclelogo;
    http.p('</BODY></HTML>');

```

END forma_nego;

```

/*****
PROCEDURE vali_nego (p_opcion in NUMBER, p_variable in VARCHAR2) IS

```

```

v_clasi tipo_hoy.clasi%TYPE;

```

BEGIN

```

    select clasi into v_clasi from tipo_hoy where id_tipo=p_opcion;

```

```

    IF (v_clasi=1) or ((v_clasi=2) and (p_variable is NOT NULL)) THEN
        nego.busca_nego(p_variable, p_opcion);

```

```

    ELSE

```

```

        nego.subforma_nego(p_opcion);

```

```

    END IF;

```

END vali_nego;

```

/*****
PROCEDURE busca_nego(p_variable in VARCHAR2, p_opcion in NUMBER) IS

```

```

score char(5);

```

```

pk char(5);

```

```

curid number;

```

```

texto varchar2(4000);

```

```

v_contador NUMBER:=0;

```

```

v_titulo tipo_hoy.titulo%TYPE;

```

CURSOR cur1 IS

```

    SELECT texto

```

```

    FROM negocios

```

```

    WHERE tipo=p_opcion;

```

```
BEGIN
```

```
select titulo into v_titulo from tipo_hoy where id_tipo=p_opcion;
```

```
IF (p_variable is NOT NULL) THEN
```

```
  dbms_output.enable(1000000);
```

```
  curid := ctx_query.open_con(policy_name => 'poli_negocios',
```

```
    text_query => p_variable,
```

```
    score_sorted => true,
```

```
    other_cols => 'texto',
```

```
    struct_query => 'tipo=||p_opcion||');
```

```
END IF;
```

```
htp.p ('<html><head><title>NEGOCIOS</title>
```

```
  <meta name="author" content="lweb S.A. de C.V.1998 " Calzada de las Aguilas 101
```

```
  tel 5936110 y 5936233 MZxico D.F."></head>
```

```
  <body bgcolor="#FFFFFF"><div align="CENTER">
```

```
  <table border="0" width="468">
```

```
    <tr>
```

```
      <td colspan="2"><font color="#666666"></font><font color="#666666">
```

```
  </font></td>
```

```
  </tr>
```

```
  <tr>
```

```
    <td colspan="2"></td>
```

```
  </tr>
```

```
  </table></div>');
```

```
htp.p ('<center>
```

```
  <BR>Tema:
```

```
  <FONT
```

```
    SIZE=3
```

```
COLOR="#663300"><B>'||v_titulo||'</B></FONT><BR>');
```

```
htp.tableOpen;
```

```
IF (p_variable is NOT NULL) THEN
```

```
  htp.p('Palabra: <FONT SIZE=3 COLOR="#FF2211"><B>'||p_variable||'</B></FONT><BR>');
```

```
  while (ctx_query.fetch_hit(curid,pk,score,texto)>0)
```

```
    loop
```

```
      htp.p('<TR><TD><IMG SRC = "/uni/images/fig13.gif"><BR></TD><TD>'||texto||'</TD></TR>
```

```
');
```

```
      v_contador:=v_contador+1;
```

```
    end loop;
```

```
    ctx_query.close_con(curid);
```

```
ELSE
```

```
  FOR i IN cur1 loop
```

```
    htp.p('<TR><TD><IMG SRC = "/uni/images/fig13.gif"><BR></TD><TD>'||i.texto||'</TD></TR>
```

```
');
```

```
    v_contador:=v_contador+1;
```

```
  END LOOP;
```

```
END IF;
```

```
htp.tableClose;
```

```
htp.p('<BR>Número de Registros Encontrados: <FONT SIZE=4 COLOR="#FF2211"><B>
```

```
'||v_contador||'</B></FONT><BR>');
```

```
htp.p('<br><div align="CENTER">
```

```
  <table border="0" width="280" cellspacing="0" cellpadding="0" vspace="0" hspace="0">
```

```
  <tr><td><div align="CENTER">
```

```
    <a href="http://ns1.aviso-oportuno.com.mx/noti/plsql/nego.forma_nego">
```

```
    </a>
```

```
  </div></td>
```

```
  <td><div align="CENTER">
```

```
    <a
```

```
      href="mailto:contacta@aguila.el-universal.com.mx"></a>
```

```

</div></td>
<td><div align="CENTER">
  <a href="/uni/index.html" target="_top"></a>
</div></td>
<td><div align="CENTER">
  <a href="/uni/pages/mapasitio.htm" target="main"></a>
</div></td>
<td><div align="CENTER">
  <a href="busca.forma_busca" target="main"></a>
</div></td>
<td><div align="CENTER">
  <a href="/uni/pages/ayuda.htm" target="main"></a>
</div></td>
</tr>
<tr><td><div align="CENTER">
  <a href="http://ns1.avisooportuno.com.mx/noti/plsql/nego.forma_nego">
  <font size="2" face="Arial, Helvetica, sans-serif">Regreso</font></a>
</div></td>
<td><div align="CENTER">
  <font size="2" face="Arial, Helvetica, sans-serif"><a href="mailto:contacta@aguila.el-
universal.com.mx">Cont&acutectanos</a></font>
</div></td>
<td><div align="CENTER">
  <font face="Arial, Helvetica, sans-serif" size="2"><a href="/uni/index.html"
target="_top">Inicio</a></font>
</div></td>
<td><div align="CENTER">
  <font face="Arial, Helvetica, sans-serif" size="2"><a href="/uni/pages/mapasitio.htm"
target="main">Mapa</a></font>
</div></td>
<td><div align="CENTER">
  <font face="Arial, Helvetica, sans-serif" size="2"><a href="busca.forma_busca"
target="main">B&uacute;squeda</a></font>
</div></td>
<td><div align="CENTER">
  <font size="2" face="Arial, Helvetica, sans-serif"><a href="/uni/pages/ayuda.htm"
target="main">Ayuda</a></font>
</div>
</td></tr></table></div>");
oraclelogo;
http.p('</BODY></HTML>');
END busca_nego;

```

```

/*****
PROCEDURE subforma_nego (p_opcion in NUMBER)IS

```

```

v_titulo  subtipo_hoy.titulo%TYPE;

```

```

BEGIN

```

```

  select titulo into v_titulo from tipo_hoy where id_tipo=p_opcion;

```

```

  http.p('<html><head><title>NEGOCIOS</title>

```

```

    <meta name="author" content="Iweb S.A. de C.V.1998 ~ Calzada de las Aguilas 101

```

```

tel 5936110 y 5936233 MZxico D.F."></head>
<body bgcolor="#FFFFFF"><div align="CENTER">
<table border="0" width="468">
<tr>
<td colspan="2"><font color="#666666"></font><font color="#666666">
</font></td>
</tr>
<tr>
<td colspan="2"></td>
</tr>
</table></div>
<CENTER><FONT COLOR="#000000" face="Arial, Helvetica, sans-serif"
SIZE=2><B>'||v_titulo||'</B></FONT></CENTER>
<CENTER><table border="0" width="500"><TR><TD>
<FONT COLOR="#000000" SIZE=2>Seleccione un tema de su interés, aquellos temas marcados
con ** contienen subtemas,
(los cuales pueden verse sólo dando click en el botón BUSCAR)</TD></TR></FONT>
</table></CENTER>');
http.formOpen('nego.subbusca_nego');
http.p(' <CENTER>
<table cellpadding="0">
<tr><td><FONT COLOR="#663300" face="Arial, Helvetica, sans-serif" SIZE=2>Tema de
busqueda :</td><td>');
http.formSelectOpen(cname=>'p_subopcion');
for cur_opp in (SELECT TITULO,ID FROM subtipo_hoy WHERE tipo=p_opcion)
LOOP

http.FormSelectOption(cur_opp.titulo,'SELECTED',cattributes=>'VALUE=""||cur_opp.id||"');
END LOOP;
http.p('</select></FONT></td></tr>
<tr><td width="500" colspan="2"><FONT COLOR="#000000" SIZE=2>Si desea ser más
específico, escriba sus opciones en la caja de "Palabras a Buscar"</font></td></tr>
<tr><td>
<FONT COLOR="#663300" face="Arial, Helvetica, sans-serif" SIZE=2>Palabras a
buscar:</font></td>
<td><input type="text" size="30" name="p_subvariable">
</td></tr></table><CENTER>
<FONT COLOR="#000000" face="Arial, Helvetica, sans-serif" SIZE=2><input
type="submit" value="Buscar"></font>
<FONT COLOR="#000000" face="Arial, Helvetica, sans-serif" SIZE=2><input
type="reset" value="Borrar"></font></CENTER>');
http.formClose;
http.p('<div align="CENTER">
<table border="0" width="280" cellpadding="0" cellspacing="0" vspace="0" hspace="0">
<tr><td><div align="CENTER">
<a href="nego.forma_nego">
</a>
</div></td>
<td><div align="CENTER">
<a
href="mailto:contacta@aguila.el-universal.com.mx"></a>
</div></td>
<td><div align="CENTER">
<a href="/uni/index.html" target="_top"></a>

```

```

</div></td>
<td><div align="CENTER">
  <a href="/uni/pages/mapasitio.htm" target="main"></a>
</div></td>
<td><div align="CENTER">
  <a href="busca.forma_busca" target="main"></a>
</div></td>
<td><div align="CENTER">
  <a href="/uni/pages/ayuda.htm" target="main"></a>
</div></td>
</tr>
<tr><td><div align="CENTER">
  <a href="nego.forma_nego">
  <font size="2" face="Arial, Helvetica, sans-serif">Regreso</font></a>
</div></td>
<td><div align="CENTER">
  <font size="2" face="Arial, Helvetica, sans-serif"><a href="mailto:contacta@aguila.el-
universal.com.mx">Cont&aacute;ctanos</a></font>
</div></td>
<td><div align="CENTER">
  <font face="Arial, Helvetica, sans-serif" size="2"><a href="/uni/index.html"
target="_top">Inicio</a></font>
</div></td>
<td><div align="CENTER">
  <font face="Arial, Helvetica, sans-serif" size="2"><a href="/uni/pages/mapasitio.htm"
target="main">Mapa</a></font>
</div></td>
<td><div align="CENTER">
  <font face="Arial, Helvetica, sans-serif" size="2"><a href="busca.forma_busca"
target="main">B&uacute;squeda</a></font>
</div></td>
<td><div align="CENTER">
  <font size="2" face="Arial, Helvetica, sans-serif"><a href="/uni/pages/ayuda.htm"
target="main">Ayuda</a></font>
</div>
</td></tr></table></div>');
oraclelogo;
http.p('</BODY></HTML>');

```

END subforma_nego;

/*****/

PROCEDURE subbusca_nego(p_subopcion in NUMBER,p_subvariable in VARCHAR2) IS

```

score char(5);
pk char(5);
curid number;
texto varchar2(4000);
v_contador NUMBER:=0;
v_tipo subtipo_hoy.tipo%TYPE;
v_id_subtipo subtipo_hoy.id_subtipo%TYPE;
v_titulo subtipo_hoy.titulo%TYPE;

```

CURSOR cur1 IS

```

SELECT texto,tipo,subtipo
FROM negocios
WHERE subtipo = (select id_subtipo from subtipo_hoy where id=p_subopcion)
AND tipo = (select tipo from subtipo_hoy where id=p_subopcion);

BEGIN
SELECT tipo,id_subtipo,titulo
into v_tipo,v_id_subtipo,v_titulo from subtipo_hoy where id=p_subopcion;

IF (p_subvariable is NOT NULL) THEN
  curid := ctx_query.open_con(policy_name => 'poli_negocios',
    text_query => p_subvariable,
    score_sorted => true,
    other_cols => 'texto',
    struct_query => 'tipo='||v_tipo||' and subtipo='||v_id_subtipo||'');
END IF;
http.p ('<html><head><title>NEGOCIOS</title>
  <meta name="author" content="lweb S.A. de C.V.1998 " Calzada de las Aguilas 101
  tel 5936110 y 5936233 MZxico D.F."></head>
  <body bgcolor="#FFFFFF"><div align="CENTER">
  <table border="0" width="468">
  <tr><td colspan="2"><font color="#666666"></font><font color="#666666">
  </font></td>
  </tr><tr>
  <td colspan="2"></td>
  </tr></table></div>');
http.p ('<center> <BR>Tema: <FONT SIZE=3
COLOR="#663300"><B>'||v_titulo||'</B></FONT><BR>');
http.tableOpen;
IF (p_subvariable is NOT NULL) THEN
  http.p('Palabra: <FONT SIZE=3 COLOR="#FF2211">'||p_subvariable||'</FONT><BR>');
  while (ctx_query.fetch_hit(curid,pk,score,texto)>0)
  loop
    http.p('<TR><TD><IMG SRC = "/uni/images/fig13.gif"><BR></TD><TD>'||texto||'</TD></TR>
');
    v_contador:=v_contador+1;
  end loop;
  ctx_query.close_con(curid);
ELSE
  FOR i IN cur1 loop
    http.p('<TR><TD><IMG SRC = "/uni/images/fig13.gif"><BR></TD><TD>'||i.texto||'</TD></TR> ');
    v_contador:=v_contador+1;
  END LOOP;
END IF;
http.tableClose;
http.p('<BR>Número de Registros Encontrados: <FONT SIZE=4 COLOR="#FF2211"><B>
'v_contador||'</B> </FONT><BR>');
http.p('<br><div align="CENTER">
  <table border="0" width="280" cellspacing="0" cellpadding="0" vspace="0" hspace="0">
  <tr><td><div align="CENTER">
  <a href="nego.subforma_nego?p_opcion='||v_tipo||'">
  </a>
  </div></td>
  <td><div align="CENTER">

```

```

        <a href="mailto:contacta@aguila.el-universal.com.mx"></a>
    </div></td>
    <td><div align="CENTER">
        <a href="/uni/index.html" target="_top"></a>
    </div></td>
    <td><div align="CENTER">
        <a href="/uni/pages/mapasitio.htm" target="main"></a>
    </div></td>
    <td><div align="CENTER">
        <a href="busca.forma_busca" target="main"></a>
    </div></td>
    <td><div align="CENTER">
        <a href="/uni/pages/ayuda.htm" target="main"></a>
    </div></td>
</tr>
<tr><td><div align="CENTER">
    <a href="nego.subforma_nego?p_opcion=||v_tipo||">
    <font size="2" face="Arial, Helvetica, sans-serif">Regreso</font></a>
</div></td>
    <td><div align="CENTER">
    <font size="2" face="Arial, Helvetica, sans-serif"><a href="mailto:contacta@aguila.el-
universal.com.mx">Cont&acute;tanos</a></font>
    </div></td>
    <td><div align="CENTER">
    <font face="Arial, Helvetica, sans-serif" size="2"><a href="/uni/index.html"
target="_top">Inicio</a></font>
    </div></td>
    <td><div align="CENTER">
    <font face="Arial, Helvetica, sans-serif" size="2"><a href="/uni/pages/mapasitio.htm"
target="main">Mapa</a></font>
    </div></td>
    <td><div align="CENTER">
    <font face="Arial, Helvetica, sans-serif" size="2"><a href="busca.forma_busca"
target="main">B&uacute;squeda</a></font>
    </div></td>
    <td><div align="CENTER">
    <font size="2" face="Arial, Helvetica, sans-serif"><a href="/uni/pages/ayuda.htm"
target="main">Ayuda</a></font>
    </div>
</td></tr></table></div>);
oraclelogo;
http.p('</BODY></HTML>');
END subbusca_nego;

END nego;
/
/** C3digo fuente del m3dulo Permutas **/

CREATE OR REPLACE PACKAGE permu AS

```

```

PROCEDURE forma_permu;
PROCEDURE vali_permu (p_opcion in NUMBER, p_variable in VARCHAR2);
PROCEDURE busca_permu (p_variable in VARCHAR2, p_opcion in NUMBER);
PROCEDURE subforma_permu (p_opcion in NUMBER);
PROCEDURE subbusca_permu(p_subopcion in NUMBER,p_subvariable in VARCHAR2 );

END permu;
/
CREATE OR REPLACE PACKAGE BODY permu AS
/*****
PROCEDURE forma_permu IS
BEGIN
  http.p('<html><head><title>PERMUTAS</title>
    <meta name="author" content="lweb S.A. de C.V.1998 " Calzada de las Aguilas 101
    tel 5936110 y 5936233 MZxico D.F."></head>
    <body bgcolor="#FFFFFF"><div align="CENTER">
    <table border="0" width="468">
    <tr>
      <td colspan="2"><font color="#666666"></font><font color="#666666">
      </font></td>
    </tr>
    <tr>
      <td colspan="2"></td>
    </tr>
    </table></div>
    <CENTER><FONT COLOR="#666666" face="Arial, Helvetica, sans-serif"
SIZE=2><B>Permutas</B></FONT></CENTER>
    <CENTER><table border="0" width="500"><TR><TD>
    <FONT COLOR="#000000" SIZE=2>Seleccione un tema de su interés, aquellos temas marcados
con ** contienen subtemas,
    (los cuales pueden verse sólo dando click en el botón BUSCAR)</TD></TR></FONT>
    </table></CENTER>');
  http.formOpen('permuto.vali_permu');
  http.p('<CENTER>
    <table cellspacing="0">
      <tr><td><FONT COLOR="#000000" face="Arial, Helvetica, sans-serif" SIZE=2>Tema de
busqueda :</td><td>);
      http.formSelectOpen(cname=>'p_opcion');
      for cur_op in (SELECT TITULO,ID_TIPO FROM tipo_hoy WHERE padre=92) LOOP

        http.formSelectOption(cur_op.titulo,'SELECTED',cattributes=>'VALUE=""|cur_op.id_tipo||"');
        END LOOP;
        http.p('</select></FONT></td></tr>
          <tr><td width="500" colspan="2"><FONT COLOR="#000000" SIZE=2>Si desea ser más
específico, escriba sus opciones en la caja de "Palabras a Buscar"</font></td></tr>
          <tr><td>
            <FONT COLOR="#000000" face="Arial, Helvetica, sans-serif" SIZE=2>Palabras a
buscar.</font></td>
            <td><input type="text" size="30" name="p_variable">
            </td></tr></table><CENTER>
            <FONT COLOR="#000000" face="Arial, Helvetica, sans-serif" SIZE=2><input
type="submit" value="Buscar"></font>
            <FONT COLOR="#000000" face="Arial, Helvetica, sans-serif" SIZE=2><input
type="reset" value="Borrar"></font></CENTER>');
  http.formClose;

```

```

htp.p('<div align="CENTER">
  <table border="0" width="280" cellspacing="0" cellpadding="0" vspace="0" hspace="0">
  <tr><td><div align="CENTER">
    <a href="http://ns1.avis-oportuno.com.mx" target="_top">
    </a>
  </div></td>
  <td><div align="CENTER">
    <a href="mailto:contacta@aguila.el-universal.com.mx"></a>
  </div></td>
  <td><div align="CENTER">
    <a href="/uni/index.html" target="_top"></a>
  </div></td>
  <td><div align="CENTER">
    <a href="/uni/pages/mapasitio.htm" target="main"></a>
  </div></td>
  <td><div align="CENTER">
    <a href="busca.forma_busca" target="main"></a>
  </div></td>
  <td><div align="CENTER">
    <a href="/uni/pages/ayuda.htm" target="main"></a>
  </div></td>
  </tr>
  <tr><td><div align="CENTER">
    <a href="http://ns1.avis-oportuno.com.mx" target="_top">
    <font size="2" face="Arial, Helvetica, sans-serif">Regreso</font></a>
  </div></td>
  <td><div align="CENTER">
    <font size="2" face="Arial, Helvetica, sans-serif"><a href="mailto:contacta@aguila.el-
universal.com.mx">Cont&aacute;ctanos</a></font>
  </div></td>
  <td><div align="CENTER">
    <font face="Arial, Helvetica, sans-serif" size="2"><a href="/uni/index.html"
target="_top">Inicio</a></font>
  </div></td>
  <td><div align="CENTER">
    <font face="Arial, Helvetica, sans-serif" size="2"><a href="/uni/pages/mapasitio.htm"
target="main">Mapa</a></font>
  </div></td>
  <td><div align="CENTER">
    <font face="Arial, Helvetica, sans-serif" size="2"><a href="busca.forma_busca"
target="main">B&uacute;squeda</a></font>
  </div></td>
  <td><div align="CENTER">
    <font size="2" face="Arial, Helvetica, sans-serif"><a href="/uni/pages/ayuda.htm"
target="main">Ayuda</a></font>
  </div>
  </td></tr></table></div>');
  oraclelogo;
  htp.p('</BODY></HTML>');
END forma_permu;
/*****/

```

```
PROCEDURE vali_permu (p_opcion in NUMBER, p_variable in VARCHAR2) IS
```

```
v_clasi tipo_hoy.clasi%TYPE;
```

```
BEGIN
```

```
select clasi into v_clasi from tipo_hoy where id_tipo=p_opcion;
```

```
IF (v_clasi=1) or ((v_clasi=2) and (p_variable is NOT NULL)) THEN
    permu.busca_permu(p_variable, p_opcion);
```

```
ELSE
```

```
    permu.subforma_permu(p_opcion);
```

```
END IF;
```

```
END vali_permu;
```

```
/*-----*/
```

```
PROCEDURE busca_permu(p_variable in VARCHAR2, p_opcion in NUMBER) IS
```

```
score char(5);
```

```
pk char(5);
```

```
curid number;
```

```
texto varchar2(4000);
```

```
v_contador NUMBER:=0;
```

```
v_titulo tipo_hoy.titulo%TYPE;
```

```
CURSOR cur1 IS
```

```
SELECT texto
```

```
FROM permutas
```

```
WHERE tipo=p_opcion;
```

```
BEGIN
```

```
select titulo into v_titulo from tipo_hoy where id_tipo=p_opcion;
```

```
IF (p_variable is NOT NULL) THEN
```

```
    dbms_output.enable(1000000);
```

```
    curid := ctx_query.open_con(policy_name => 'poli_permutas',
```

```
        text_query => p_variable,
```

```
        score_sorted => true,
```

```
        other_cols => 'texto',
```

```
        struct_query => 'tipo=||p_opcion||');
```

```
END IF;
```

```
htp.p ('<html><head><title>PERMUTAS</title>
```

```
<meta name="author" content="lweb S.A. de C.V.1998 " Calzada de las Aguilas 101
```

```
tel 5936110 y 5936233 MZxico D.F."></head>
```

```
<body bgcolor="#FFFFFF"><div align="CENTER">
```

```
<table border="0" width="468">
```

```
<tr>
```

```
<td colspan="2"><font color="#666666"></font><font color="#666666">
```

```
</font></td>
```

```
</tr>
```

```
<tr>
```

```
<td colspan="2"></td>
```

```
</tr>
```

```
</table></div>');
```

```
htp.p ('<center><BR>Tema: <FONT SIZE=3 COLOR="#666666"><B>||v_titulo||</B></FONT><BR>');
```

```
htp.tableOpen;
```

```

IF (p_variable is NOT NULL) THEN
  http.p('Palabra: <FONT SIZE=3 COLOR="#FF2211"><B>||p_variable||</B></FONT><BR>');
  while (ctx_query.fetch_hit(curid,pk,score,texto)>0)
    loop
      http.p('<TR><TD><IMG SRC = "/uni/images/fig13.gif"><BR></TD><TD>||texto||</TD></TR>');
      v_contador:=v_contador+1;
    end loop;
  ctx_query.close_con(curid);
ELSE
  FOR i IN cur1 loop
    http.p('<TR><TD><IMG SRC = "/uni/images/fig13.gif"><BR></TD><TD>||i.texto||</TD></TR>');
    v_contador:=v_contador+1;
  END LOOP;
END IF;
http.tableClose;
http.p('<BR>Número de Registros Encontrados: <FONT SIZE=4 COLOR="#FF2211"><B>||v_contador||</B> </FONT><BR>');
http.p('<br><div align="CENTER">
  <table border="0" width="280" cellspacing="0" cellpadding="0" vspace="0" hspace="0">
    <tr><td><div align="CENTER">
      <a href="http://ns1.aviso-oportuno.com.mx/noti/plsql/permu.forma_permu">
      </a>
    </div></td>
    <td><div align="CENTER">
      <a href="mailto:contacta@aguila.el-universal.com.mx"></a>
    </div></td>
    <td><div align="CENTER">
      <a href="/uni/index.html" target="_top"></a>
    </div></td>
    <td><div align="CENTER">
      <a href="/uni/pages/mapasitio.htm" target="main"></a>
    </div></td>
    <td><div align="CENTER">
      <a href="busca.forma_busca" target="main"></a>
    </div></td>
    <td><div align="CENTER">
      <a href="/uni/pages/ayuda.htm" target="main"></a>
    </div></td>
  </tr>
  <tr><td><div align="CENTER">
    <a href="http://ns1.aviso-oportuno.com.mx/noti/plsql/permu.forma_permu">
    <font size="2" face="Arial, Helvetica, sans-serif">Regreso</font></a>
  </div></td>
  <td><div align="CENTER">
    <font size="2" face="Arial, Helvetica, sans-serif"><a href="mailto:contacta@aguila.el-universal.com.mx">Cont&acutec;ctanos</a></font>
  </div></td>
  <td><div align="CENTER">

```

```

        <font face="Arial, Helvetica, sans-serif" size="2"><a href="/uni/index.html"
target="_top">Inicio</a></font>
    </div></td>
    <td><div align="CENTER">
        <font face="Arial, Helvetica, sans-serif" size="2"><a href="/uni/pages/mapasitio.htm"
target="main">Mapa</a></font>
    </div></td>
    <td><div align="CENTER">
        <font face="Arial, Helvetica, sans-serif" size="2"><a href="busca.forma_busca"
target="main">B&uacute;squeda</a></font>
    </div></td>
    <td><div align="CENTER">
        <font size="2" face="Arial, Helvetica, sans-serif"><a href="/uni/pages/ayuda.htm"
target="main">Ayuda</a></font>
    </div>
</td></tr></table></div>');
    oraclelogo;
    http.p('</BODY></HTML>');
END busca_permu;

```

```

/*****
PROCEDURE subforma_permu (p_opcion in NUMBER)IS

```

```

v_titulo  subtipo_hoy.titulo%TYPE;

```

```

BEGIN

```

```

    select titulo into v_titulo from tipo_hoy where id_tipo=p_opcion;
    http.p('<html><head><title>PERMUTAS</title>
    <meta name="author" content="lweb S.A. de C.V.1998 " Calzada de las Aguilas 101
    tel 5936110 y 5936233 MZxico D.F."></head>
    <body bgcolor="#FFFFFF"><div align="CENTER">
    <table border="0" width="468">
    <tr>
    <td colspan="2"><font color="#666666"></font><font color="#666666">
    </font></td>
    </tr>
    <tr>
    <td colspan="2"></td>
    </tr>
    </table></div>
    <CENTER><FONT COLOR="#666666" face="Arial, Helvetica, sans-serif"
SIZE=2><B>||v_titulo||</B></FONT></CENTER>
    <CENTER><table border="0" width="500"><TR><TD>
    <FONT COLOR="#000000" SIZE=2>Seleccione un tema de su interés, aquellos temas marcados
con ** contienen subtemas,
    (los cuales pueden verse sólo dando click en el botón BUSCAR)</TD></TR></FONT>
    </table></CENTER>');
    http.formOpen('permu.subbusca_permu');
    http.p(' <CENTER>
    <table cellspacing="0">
    <tr><td><FONT COLOR="#000000" face="Arial, Helvetica, sans-serif" SIZE=2>Tema de
busqueda :</td><td>');
    http.formSelectOpen(cname=>'p_subopcion');
    for cur_opp in (SELECT TITULO,ID FROM subtipo_hoy WHERE tipo=p_opcion)
    LOOP

```

```

http.FormSelectOption(cur_opp.titulo,'SELECTED',cattributes=>'VALUE=""|cur_opp.id|');
END LOOP;
http.p('</select></FONT></td></tr>
<tr><td width="500" colspan="2"><FONT COLOR="#000000" SIZE=2>Si desea ser más
específico, escriba sus opciones en la caja de "Palabras a Buscar"</font></td></tr>
<tr><td>
<FONT COLOR="#000000" face="Arial, Helvetica, sans-serif" SIZE=2>Palabras a
buscar:</font></td>
<td><input type="text" size="30" name="p_subvariable">
</td></tr></table><CENTER>
<FONT COLOR="#000000" face="Arial, Helvetica, sans-serif" SIZE=2><input
type="submit" value="Buscar"></font>
<FONT COLOR="#000000" face="Arial, Helvetica, sans-serif" SIZE=2><input
type="reset" value="Borrar"></font></CENTER>');
http.formClose;
http.p('<div align="CENTER">
<table border="0" width="280" cellspacing="0" cellpadding="0" vspace="0" hspace="0">
<tr><td><div align="CENTER">
<a href="permu.forma_permu">
</a>
</div></td>
<td><div align="CENTER">
<a href="mailto:contacta@aguila.el-universal.com.mx"></a>
</div></td>
<td><div align="CENTER">
<a href="/uni/index.html" target="_top"></a>
</div></td>
<td><div align="CENTER">
<a href="/uni/pages/mapasitio.htm" target="main"></a>
</div></td>
<td><div align="CENTER">
<a href="busca.forma_busca" target="main"></a>
</div></td>
<td><div align="CENTER">
<a href="/uni/pages/ayuda.htm" target="main"></a>
</div></td>
</tr>
<tr><td><div align="CENTER">
<a href="permu.forma_permu">
<font size="2" face="Arial, Helvetica, sans-serif">Regreso</font></a>
</div></td>
<td><div align="CENTER">
<font size="2" face="Arial, Helvetica, sans-serif"><a href="mailto:contacta@aguila.el-
universal.com.mx">Cont&acutecanos</a></font>
</div></td>
<td><div align="CENTER">
<font face="Arial, Helvetica, sans-serif" size="2"><a href="/uni/index.html"
target="_top">Inicio</a></font>
</div></td>
<td><div align="CENTER">

```

```

        <font face="Arial, Helvetica, sans-serif" size="2"><a href="/uni/pages/mapasitio.htm"
target="main">Mapa</a></font>
    </div></td>
    <td><div align="CENTER">
        <font face="Arial, Helvetica, sans-serif" size="2"><a href="busca.forma_busca"
target="main">B&uacute;squeda</a></font>
    </div></td>
    <td><div align="CENTER">
        <font size="2" face="Arial, Helvetica, sans-serif"><a href="/uni/pages/ayuda.htm"
target="main">Ayuda</a></font>
    </div>
</td></tr></table></div>');
    oraclelogo;
    http.p('</BODY></HTML>');
END subforma_permu;
/*****/
PROCEDURE subbusca_permu(p_subopcion in NUMBER,p_subvariable in VARCHAR2) IS

score char(5);
pk char(5);
curid number;
texto varchar2(4000);
v_contador NUMBER:=0;
v_tipo subtipo_hoy.tipo%TYPE;
v_id_subtipo subtipo_hoy.id_subtipo%TYPE;
v_titulo subtipo_hoy.titulo%TYPE;

CURSOR cur1 IS
    SELECT texto,tipo,subtipo
    FROM permutas
    WHERE subtipo = (select id_subtipo from subtipo_hoy where id=p_subopcion)
    AND tipo = (select tipo from subtipo_hoy where id=p_subopcion);

BEGIN
    SELECT tipo,id_subtipo,titulo
    into v_tipo,v_id_subtipo,v_titulo from subtipo_hoy where id=p_subopcion;

    IF (p_subvariable is NOT NULL) THEN
        curid := ctx_query.open_con(policy_name => 'poli_permutas',
            text_query => p_subvariable,
            score_sorted => true,
            other_cols => 'texto',
            struct_query => 'tipo=||v_tipo|| and subtipo=||v_id_subtipo||');
    END IF;
    http.p('<html><head><title>PERMUTAS</title>
    <meta name="author" content="lweb S.A. de C.V.1998 " Calzada de las Aguilas 101
    tel 5936110 y 5936233 MZxico D.F."></head>
    <body bgcolor="#FFFFFF"><div align="CENTER">
    <table border="0" width="468">
        <tr><td colspan="2"><font color="#666666"></font><font color="#666666">
        </font></td>
        </tr><tr>
        <td colspan="2"></td>
        </tr></table></div>');

```

```

htp.p ('<center> <BR>Tema: <FONT SIZE=3
COLOR="#666666"><B>||v_titulo||</B></FONT><BR>');
htp.tableOpen;
IF (p_subvariable is NOT NULL) THEN
  htp.p('Palabra: <FONT SIZE=3 COLOR="#FF2211">||p_subvariable||</FONT><BR>');
  while (ctx_query.fetch_hit(curid,pk,score,texto)>0)
    loop
      htp.p('<TR><TD><IMG SRC = "/uni/images/fig13.gif"><BR></TD><TD>||texto||</TD></TR>
');
      v_contador:=v_contador+1;
    end loop;
  ctx_query.close_con(curid);
ELSE
  FOR i IN cur1 loop
    htp.p('<TR><TD><IMG SRC = "/uni/images/fig13.gif"><BR></TD><TD>||i.texto||</TD></TR> ');
    v_contador:=v_contador+1;
  END LOOP;
END IF;
htp.tableClose;
htp.p('<BR>Número de Registros Encontrados: <FONT SIZE=4 COLOR="#FF2211"><B>
||v_contador||</B> </FONT><BR>');
htp.p('<br><div align="CENTER">
<table border="0" width="280" cellspacing="0" cellpadding="0" vspace="0" hspace="0">
<tr><td><div align="CENTER">
  <a href="permu.subforma_permu?p_opcion=||v_tipo||">
  </a>
</div></td>
<td><div align="CENTER">
  <a href="mailto:contacta@aguila.el-universal.com.mx"></a>
</div></td>
<td><div align="CENTER">
  <a href="/uni/index.html" target="_top"></a>
</div></td>
<td><div align="CENTER">
  <a href="/uni/pages/mapasitio.htm" target="main"></a>
</div></td>
<td><div align="CENTER">
  <a href="busca.forma_busca" target="main"></a>
</div></td>
<td><div align="CENTER">
  <a href="/uni/pages/ayuda.htm" target="main"></a>
</div></td>
</tr>
<tr><td><div align="CENTER">
  <a href="permu.subforma_permu?p_opcion=||v_tipo||">
  <font size="2" face="Arial, Helvetica, sans-serif">Regreso</font></a>
</div></td>
<td><div align="CENTER">
  <font size="2" face="Arial, Helvetica, sans-serif"><a href="mailto:contacta@aguila.el-
universal.com.mx">Cont&acute;ctanos</a></font>
</div></td>

```

```

        <td><div align="CENTER">
          <font face="Arial, Helvetica, sans-serif" size="2"><a href="/uni/index.html"
target="_top">Inicio</a></font>
        </div></td>
        <td><div align="CENTER">
          <font face="Arial, Helvetica, sans-serif" size="2"><a href="/uni/pages/mapasitio.htm"
target="main">Mapa</a></font>
        </div></td>
        <td><div align="CENTER">
          <font face="Arial, Helvetica, sans-serif" size="2"><a href="busca.forma_busca"
target="main">B&uacute;squeda</a></font>
        </div></td>
        <td><div align="CENTER">
          <font size="2" face="Arial, Helvetica, sans-serif"><a href="/uni/pages/ayuda.htm"
target="main">Ayuda</a></font>
        </div>
      </td></tr></table></div>);
      oraclelogo;
      http.p('</BODY></HTML>');
END subbusca_permu;

END permu;
/

```

/** C3digo fuente del m3dulo Servicios **/

```
CREATE OR REPLACE PACKAGE servi AS
```

```

  PROCEDURE forma_servi;
  PROCEDURE vali_servi (p_opcion in NUMBER, p_variable in VARCHAR2);
  PROCEDURE busca_servi (p_variable in VARCHAR2, p_opcion in NUMBER);
  PROCEDURE subforma_servi (p_opcion in NUMBER);
  PROCEDURE subbusca_servi (p_subopcion in NUMBER, p_subvariable in VARCHAR2);

```

```
END servi;
```

```
/
```

```
CREATE OR REPLACE PACKAGE BODY servi AS
```

```
/******
```

```
PROCEDURE forma_servi IS
```

```
BEGIN
```

```
http.p('<html><head>
```

```
<title>SERVICIOS</title>
```

```
<meta name="author" content="lweb S.A. de C.V.1998 " Calzada de las Aguilas 101
```

```
tel 5936110 y 5936233 M3xico D.F.">
```

```
</head><body bgcolor="#FFFFFF">
```

```
<div align="CENTER">
```

```
<table border="0" width="468">
```

```
<tr>
```

```
<td colspan="2" height="61"><font color="#666666"></font><font color="#666666">
```

```
</font></td>
```

```
</tr>
```

```
<tr>
```

```
<td colspan="2"></td>
```

```
</tr>
```

```

</table></div>
<CENTER><FONT COLOR="#336699" face="Arial, Helvetica, sans-serif"
SIZE=2><B>Servicios</B></FONT></CENTER>
<CENTER><table border="0" width="500"><TR><TD>
<FONT COLOR="#000000" SIZE=2>Seleccione un tema de su interés, aquellos temas marcados
con ** contienen subtemas.
(los cuales pueden verse sólo dando click en el botón BUSCAR)</TD></TR></FONT>
</table></CENTER>;
htp.formOpen('servi.vali_servi');
htp.p('<CENTER>
<table cellspacing="0">
<tr><td><FONT COLOR="#000000" face="Arial, Helvetica, sans-serif" SIZE=2>Tema de
busqueda :</td><td>);
htp.formSelectOpen(cname=>'p_opcion');
for cur_op in (SELECT TITULO,ID_TIPO FROM tipo_hoy WHERE padre=93) LOOP

htp.FormSelectOption(cur_op.titulo,'SELECTED',cattributes=>'VALUE="||cur_op.id_tipo||"');
END LOOP;
htp.p('</select></FONT></td></tr>
<tr><td>
<FONT COLOR="#000000" face="Arial, Helvetica, sans-serif" SIZE=2>Palabras a
buscar:</font></td>
<td><input type="text" size="30" name="p_variable">
</td></table> <BR><BR> <CENTER>
<FONT COLOR="#000000" face="Arial, Helvetica, sans-serif" SIZE=2><input
type="submit" value="Buscar"></font>
<FONT COLOR="#000000" face="Arial, Helvetica, sans-serif" SIZE=2><input
type="reset" value="Borrar"></font></CENTER>');
htp.formClose;
htp.p('<div align="CENTER">
<table border="0" width="280" cellspacing="0" cellpadding="0" vspace="0" hspace="0">
<tr><td><div align="CENTER">
<a href="http://ns1.avisooportuno.com.mx" target="_top">
</a>
</div></td>
<td><div align="CENTER">
<a href="mailto:contacta@aguila.el-universal.com.mx"></a>
</div></td>
<td><div align="CENTER">
<a href="/uni/index.html" target="_top"></a>
</div></td>
<td><div align="CENTER">
<a href="/uni/pages/mapasitio.htm" target="main"></a>
</div></td>
<td><div align="CENTER">
<a href="busca.forma_busca" target="main"></a>
</div></td>
<td><div align="CENTER">
<a href="/uni/pages/ayuda.htm" target="main"></a>
</div></td>
</tr>

```

```

<tr><td><div align="CENTER">
  <a href="http://ns1.aviso-oportuno.com.mx" target="_top">
    <font size="2" face="Arial, Helvetica, sans-serif">Regreso</font></a>
  </div></td>
  <td><div align="CENTER">
    <font size="2" face="Arial, Helvetica, sans-serif"><a href="mailto:contacta@aguila.el-
universal.com.mx">Cont&aacute;tanos</a></font>
  </div></td>
  <td><div align="CENTER">
    <font face="Arial, Helvetica, sans-serif" size="2"><a href="/uni/index.html"
target="_top">Inicio</a></font>
  </div></td>
  <td><div align="CENTER">
    <font face="Arial, Helvetica, sans-serif" size="2"><a href="/uni/pages/mapasitio.htm"
target="main">Mapa</a></font>
  </div></td>
  <td><div align="CENTER">
    <font face="Arial, Helvetica, sans-serif" size="2"><a href="busca.forma_busca"
target="main">B&uacute;squeda</a></font>
  </div></td>
  <td><div align="CENTER">
    <font size="2" face="Arial, Helvetica, sans-serif"><a href="/uni/pages/ayuda.htm"
target="main">Ayuda</a></font>
  </div>
</td></tr></table></div>');
  oraclelogo;
  http.p('</BODY></HTML>');
END forma_servi;

```

```

/*****/
PROCEDURE vali_servi (p_opcion in NUMBER, p_variable in VARCHAR2) IS

v_clasi tipo_hoy.clasi%TYPE;

BEGIN
  select clasi into v_clasi from tipo_hoy where id_tipo=p_opcion;

  IF (v_clasi=1) or ((v_clasi=2) and (p_variable is NOT NULL)) THEN
    servi.busca_servi(p_variable, p_opcion);
  ELSE
    servi.subforma_servi(p_opcion);
  END IF;

END vali_servi;

/*****/
PROCEDURE busca_servi(p_variable in VARCHAR2, p_opcion in NUMBER) IS

score char(5);
pk char(5);
curid number;
texto varchar2(4000);
v_contador NUMBER:=0;
v_titulo tipo_hoy.titulo%TYPE;

CURSOR cur1 IS
  SELECT texto

```

```

FROM servicio
WHERE tipo=p_opcion;

BEGIN
select titulo into v_titulo from tipo_hoy where id_tipo=p_opcion;
IF (p_variable is NOT NULL) THEN
dbms_output.enable(1000000);
curid := ctx_query.open_con(policy_name => 'poli_servicio',
text_query => p_variable,
score_sorted => true,
other_cols => 'texto',
struct_query => 'tipo=||p_opcion||');
END IF;
http.p('<html><head>
<title>SERVICIOS</title>
<meta name="author" content="Iweb S.A. de C.V.1998 " Calzada de las Aguilas 101
tel 5936110 y 5936233 Mxico D.F.">
</head><body bgcolor="#FFFFFF">
<div align="CENTER">
<table border="0" width="468">
<tr>
<td colspan="2" height="61"><font color="#666666"></font><font color="#666666">
</font></td>
</tr>
<tr>
<td colspan="2"></td>
</tr>
</table></div>');
http.p('<center> <BR>Tema: <FONT SIZE=3 COLOR="#336699"><B>'||v_titulo||'</B></FONT><BR><BR>');
http.tableOpen;
IF (p_variable is NOT NULL) THEN
http.p('Palabra: <FONT SIZE=3 COLOR="#FF2211"><B>'||p_variable||'</B></FONT><BR>');
while (ctx_query.fetch_hit(curid,pk,score,texto)>0)
loop
http.p('<TR><TD><IMG SRC = "/uni/images/fig13.gif"><BR></TD><TD>'||texto||'</TD></TR>');
v_contador:=v_contador+1;
end loop;
ctx_query.close_con(curid);
ELSE
FOR i IN cur1 loop
http.p('<TR><TD><IMG SRC = "/uni/images/fig13.gif"><BR></TD><TD>'||i.texto||'</TD></TR>');
v_contador:=v_contador+1;
END LOOP;
END IF;
http.tableClose;
http.p('<BR>mero de Registros Encontrados: <FONT SIZE=4 COLOR="#FF2211"><B>'||v_contador||'</B></FONT>');
http.p('<br><div align="CENTER">
<table border="0" width="280" cellpadding="0" cellspacing="0" vspace="0" hspace="0">
<tr><td align="CENTER">
<a href="servi.forma_servi">
</a>

```

```

</div></td>
<td><div align="CENTER">
  <a href="mailto:contacta@aguila.el-universal.com.mx"></a>
</div></td>
<td><div align="CENTER">
  <a href="/uni/index.html" target="_top"></a>
</div></td>
<td><div align="CENTER">
  <a href="/uni/pages/mapasitio.htm" target="main"></a>
</div></td>
<td><div align="CENTER">
  <a href="busca.forma_busca" target="main"></a>
</div></td>
<td><div align="CENTER">
  <a href="/uni/pages/ayuda.htm" target="main"></a>
</div></td>
</tr>
<tr><td><div align="CENTER">
  <a href="/noti/plsql/servi.forma_servi">
  <font size="2" face="Arial, Helvetica, sans-serif">Regreso</font></a>
</div></td>
<td><div align="CENTER">
  <font size="2" face="Arial, Helvetica, sans-serif"><a href="mailto:contacta@aguila.el-
universal.com.mx">Contactanos</a></font>
</div></td>
<td><div align="CENTER">
  <font face="Arial, Helvetica, sans-serif" size="2"><a href="/uni/index.html"
target="_top">Inicio</a></font>
</div></td>
<td><div align="CENTER">
  <font face="Arial, Helvetica, sans-serif" size="2"><a href="/uni/pages/mapasitio.htm"
target="main">Mapa</a></font>
</div></td>
<td><div align="CENTER">
  <font face="Arial, Helvetica, sans-serif" size="2"><a href="busca.forma_busca"
target="main">Búsqueda</a></font>
</div></td>
<td><div align="CENTER">
  <font size="2" face="Arial, Helvetica, sans-serif"><a href="/uni/pages/ayuda.htm"
target="main">Ayuda</a></font>
</div>
</td></tr></table></div>);
oraclelogo;
http.p('</BODY></HTML>');

```

END busca_servi;

```

/*****
PROCEDURE subforma_servi (p_opcion in NUMBER)IS

```

```

v_titulo  subtipo_hoy.titulo%TYPE;

```

BEGIN

```

select titulo into v_titulo from tipo_hoy where id_tipo=p_opcion;
htp.p('<html><head>
  <title>SERVICIOS</title>
  <meta name="author" content="Iweb S.A. de C.V.1998 " Calzada de las Aguilas 101
  tel 5936110 y 5936233 Mxico D.F.">
</head><body bgcolor="#FFFFFF">
<div align="CENTER">
  <table border="0" width="468">
  <tr>
    <td colspan="2" height="61"><font color="#666666"></font><font color="#666666">
    </font></td>
  </tr>
  <tr>
    <td colspan="2"></td>
  </tr>
</table></div>
  <CENTER><FONT COLOR="#336699" face="Arial, Helvetica, sans-serif"
SIZE=2><B>||v_titulo||</B></FONT></CENTER>
  <CENTER><table border="0" width="500"><TR><TD>
  <FONT COLOR="#000000" SIZE=2>Seleccione un tema de su inters, aquellos temas marcados
con ** contienen subtemas,
  (los cuales pueden verse slo dando click en el botn BUSCAR)</TD></TR></FONT>
  </table></CENTER>');
htp.formOpen('servi.subbusca_servi');
htp.p(' <CENTER>
  <table cellspacing="0">
  <tr><td><font size="2" face="Arial, Helvetica, sans-serif" SIZE=2>Tema de busqueda
:</td><td>');
htp.formSelectOpen(cname=>'p_subopcion');
for cur_opp in (SELECT TITULO,ID FROM subtipo_hoy WHERE tipo=p_opcion)
LOOP

htp.FormSelectOption(cur_opp.titulo,'SELECTED',cattributes=>'VALUE="||cur_opp.id||"');
END LOOP;
htp.p('</select></FONT></td></tr>
<tr><td width="500" colspan="2"><FONT COLOR="#000000" SIZE=2>Si desea ser ms
especfico, escriba sus opciones en la caja de "Palabras a Buscar"</font></td></tr>
<TR><TD>
  <FONT COLOR="#000000" face="Arial, Helvetica, sans-serif" SIZE=2>Palabras a
buscar:</font> </td>
  <td><input type="text" size="30" name="p_subvariable">
  </td></tr></table><CENTER>
  <FONT COLOR="#000000" face="Arial, Helvetica, sans-serif" SIZE=2><input
type="submit" value="Buscar"></font>
  <FONT COLOR="#000000" face="Arial, Helvetica, sans-serif" SIZE=2><input
type="reset" value="Borrar"></font></CENTER>');
htp.formClose;
htp.p('<div align="CENTER">
  <table border="0" width="280" cellspacing="0" cellpadding="0" vspace="0" hspace="0">
  <tr><td><div align="CENTER">
    <a href="servi.forma_servi">
    </a>
  </div></td>
  <td><div align="CENTER">

```

```

        <a                href="mailto:contacta@aguila.el-universal.com.mx"></a>
        </div></td>
        <td><div align="CENTER">
        <a href="/uni/index.html" target="_top"></a>
        </div></td>
        <td><div align="CENTER">
        <a                href="/uni/pages/mapasitio.htm"                target="main"></a>
        </div></td>
        <td><div align="CENTER">
        <a href="busca.forma_busca" target="main"></a>
        </div></td>
        <td><div align="CENTER">
        <a href="/uni/pages/ayuda.htm" target="main"></a>
        </div></td>
</tr>
<tr><td><div align="CENTER">
        <a href="servi.forma_servi">
        <font size="2" face="Arial, Helvetica, sans-serif">Regreso</font></a>
        </div></td>
        <td><div align="CENTER">
        <font size="2" face="Arial, Helvetica, sans-serif"><a href="mailto:contacta@aguila.el-
universal.com.mx">Cont&aacute;tanos</a></font>
        </div></td>
        <td><div align="CENTER">
        <font face="Arial, Helvetica, sans-serif" size="2"><a href="/uni/index.html"
target="_top">Inicio</a></font>
        </div></td>
        <td><div align="CENTER">
        <font face="Arial, Helvetica, sans-serif" size="2"><a href="/uni/pages/mapasitio.htm"
target="main">Mapa</a></font>
        </div></td>
        <td><div align="CENTER">
        <font face="Arial, Helvetica, sans-serif" size="2"><a href="busca.forma_busca"
target="main">B&uacute;squeda</a></font>
        </div></td>
        <td><div align="CENTER">
        <font size="2" face="Arial, Helvetica, sans-serif"><a href="/uni/pages/ayuda.htm"
target="main">Ayuda</a></font>
        </div>
</td></tr></table></div>');
        oraclelogo;
        http.p(</BODY></HTML>');

```

END subforma_servi;

/******

PROCEDURE subbusca_servi(p_subopcion in NUMBER,p_subvariable in VARCHAR2) IS

```

score char(5);
pk char(5);
curid number;
texto varchar2(4000);

```

```
v_contador NUMBER:=0;
v_tipo      subtipo_hoy.tipo%TYPE;
v_id_subtipo subtipo_hoy.id_subtipo%TYPE;
v_titulo    subtipo_hoy.titulo%TYPE;
```

```
CURSOR cur1 IS
  SELECT texto,tipo,subtipo
  FROM servicio
  WHERE subtipo = (select id_subtipo from subtipo_hoy where id=p_subopcion)
  AND tipo = (select tipo from subtipo_hoy where id=p_subopcion);
```

```
BEGIN
  SELECT tipo,id_subtipo,titulo
  into v_tipo,v_id_subtipo,v_titulo from subtipo_hoy where id=p_subopcion;
```

```
IF (p_subvariable is NOT NULL) THEN
  curid := ctx_query.open_con(policy_name => 'poli_servicio',
    text_query => p_subvariable,
    score_sorted => true,
    other_cols => 'texto',
    struct_query => 'tipo=||v_tipo||' and subtipo="||v_id_subtipo||"');
```

```
END IF;
```

```
htp.p ('<html><head>
  <title>SERVICIOS</title>
  <meta name="author" content="lweb S.A. de C.V.1998 " Calzada de las Aguilas 101
  tel 5936110 y 5936233 Mxico D.F.">
</head><body bgcolor="#FFFFFF">
  <div align="CENTER">
  <table border="0" width="468">
  <tr>
    <td colspan="2" height="61"><font color="#666666"></font><font color="#666666">
  </font></td>
  <tr>
  <td colspan="2"></td>
  </tr>
  </table></div>');
```

```
htp.p ('<center> <BR>Tema: <FONT SIZE=3 COLOR="#336699"><B>||v_titulo||</B></FONT><BR>');
```

```
htp.tableOpen;
```

```
IF (p_subvariable is NOT NULL) THEN
  htp.p('Palabra: <FONT SIZE=3 COLOR="#FF2211">||p_subvariable||</FONT><BR>');
  while (ctx_query.fetch_hit(curid,pk,score,texto)>0)
  loop
    htp.p('<TR><TD><IMG SRC = "/uni/images/fig13.gif"><BR></TD><TD>||texto||</TD></TR>');
  );
```

```
v_contador:=v_contador+1;
```

```
end loop;
```

```
ctx_query.close_con(curid);
```

```
ELSE
```

```
FOR i IN cur1 loop
```

```
htp.p('<TR><TD><IMG SRC = "/uni/images/fig13.gif"><BR></TD><TD>||i.texto||</TD></TR> ');
```

```
v_contador:=v_contador+1;
```

```
END LOOP;
```

```
END IF;
```

```

http.tableClose;
http.p('<BR>Número de Registros Encontrados: <FONT SIZE=4 COLOR="#FF2211"><B>
' ||v_contador||'</B></FONT><BR>');
http.p('<br><div align="CENTER">
<table border="0" width="280" cellspacing="0" cellpadding="0" vspace="0" hspace="0">
<tr><td><div align="CENTER">
<a href="servi.subforma_servi?p_opcion=||v_tipo||">
</a>
</div></td>
<td><div align="CENTER">
<a href="mailto:contacta@aguila.el-universal.com.mx"></a>
</div></td>
<td><div align="CENTER">
<a href="/uni/index.html" target="_top"></a>
</div></td>
<td><div align="CENTER">
<a href="/uni/pages/mapasitio.htm" target="main"></a>
</div></td>
<td><div align="CENTER">
<a href="busca.forma_busca" target="main"></a>
</div></td>
<td><div align="CENTER">
<a href="/uni/pages/ayuda.htm" target="main"></a>
</div></td>
</tr>
<tr><td><div align="CENTER">
<a href="servi.subforma_servi?p_opcion=||v_tipo||">
<font size="2" face="Arial, Helvetica, sans-serif">Regreso</font></a>
</div></td>
<td><div align="CENTER">
<font size="2" face="Arial, Helvetica, sans-serif"><a href="mailto:contacta@aguila.el-
universal.com.mx">Cont&acute;tanos</a></font>
</div></td>
<td><div align="CENTER">
<font face="Arial, Helvetica, sans-serif" size="2"><a href="/uni/index.html"
target="_top">Inicio</a></font>
</div></td>
<td><div align="CENTER">
<font face="Arial, Helvetica, sans-serif" size="2"><a href="/uni/pages/mapasitio.htm"
target="main">Mapa</a></font>
</div></td>
<td><div align="CENTER">
<font face="Arial, Helvetica, sans-serif" size="2"><a href="busca.forma_busca"
target="main">B&uacute;squeda</a></font>
</div></td>
<td><div align="CENTER">
<font size="2" face="Arial, Helvetica, sans-serif"><a href="/uni/pages/ayuda.htm"
target="main">Ayuda</a></font>
</div>
</td></tr></table></div>');
oraclelogo;

```

```

        http.p('</BODY></HTML>');
END subbusca_servi;

```

```

END servi;
/

```

/** Código fuente del módulo Varios **/

```

CREATE OR REPLACE PACKAGE vario AS

```

```

    PROCEDURE forma_varios;
    PROCEDURE vali_varios (p_opcion in NUMBER, p_variable in VARCHAR2);
    PROCEDURE busca_varios (p_variable in VARCHAR2, p_opcion in NUMBER);
    PROCEDURE subforma_varios (p_opcion in NUMBER);
    PROCEDURE subbusca_varios(p_subopcion in NUMBER,p_subvariable in VARCHAR2 );

```

```

END vario;
/

```

```

CREATE OR REPLACE PACKAGE BODY vario AS

```

```

/*****/

```

```

PROCEDURE forma_varios IS

```

```

BEGIN

```

```

    http.p('<html><head><title>EMPLEOS</title>
        <meta name="author" content="Iweb S.A. de C.V.1998 " Calzada de las Aguilas 101
        tel 5936110 y 5936233 MZxico D.F."></head>
        <body bgcolor="#FFFFFF"><div align="CENTER">
        <table border="0" width="468">
        <tr>
        <td colspan="2"><font color="#666666"></font><font color="#666666">
        </font></td>
        </tr>
        <tr>
        <td colspan="2"></td>
        </tr>
        </table></div>

```

```

        <CENTER><FONT COLOR="#006699" face="Arial, Helvetica, sans-serif"
        SIZE=2><B>Varios</B></FONT></CENTER>

```

```

        <CENTER><table border="0" width="500"><TR><TD>
        <FONT COLOR="#000000" SIZE=2>Seleccione un tema de su interés, aquellos temas marcados
        con ** contienen subtemas,
        (los cuales pueden verse sólo dando click en el botón BUSCAR)</TD></TR></FONT>
        </table></CENTER>');

```

```

    http.formOpen('vario.vali_varios');

```

```

    http.p('<CENTER>

```

```

        <table cellspacing="0">
        <tr><td><FONT COLOR="#000000" face="Arial, Helvetica, sans-serif" SIZE=2>Tema de
        busqueda :</td><td>');

```

```

        http.formSelectOpen(cname=>'p_opcion');

```

```

        for cur_op in (SELECT TITULO,ID_TIPO FROM tipo_hoy WHERE padre=97) LOOP

```

```

    http.FormSelectOption(cur_op.titulo,'SELECTED',cattributes=>'VALUE=""||cur_op.id_tipo||"');

```

```

    END LOOP;

```

```

    http.p('</select></FONT></td></tr>

```

```

<tr><td width="500" colspan="2"><FONT COLOR="#000000" SIZE=2>Si desea ser más
específico, escriba sus opciones en la caja de "Palabras a Buscar"</font></td></tr>

```

```

<tr><td>

```

```

<FONT COLOR="#000000" face="Arial, Helvetica, sans-serif" SIZE=2>Palabras a
buscar:</font></td>

```

```

<td><input type="text" size="30" name="p_variable">

```

```

</td></tr></table><CENTER>

```

```

<FONT COLOR="#000000" face="Arial, Helvetica, sans-serif" SIZE=2><input
type="submit" value="Buscar"></font>

```

```

<FONT COLOR="#000000" face="Arial, Helvetica, sans-serif" SIZE=2><input
type="reset" value="Borrar"></font></CENTER>);

```

```

http.formClose;

```

```

http.p(<div align="CENTER">

```

```

<table border="0" width="280" cellspacing="0" cellpadding="0" vspace="0" hspace="0">

```

```

<tr><td><div align="CENTER">

```

```

<a href="http://ns1.aviso-oportuno.com.mx" target="_top">

```

```

</a>

```

```

</div></td>

```

```

<td><div align="CENTER">

```

```

<a href="mailto:contacta@aguila.el-universal.com.mx"></a>

```

```

</div></td>

```

```

<td><div align="CENTER">

```

```

<a href="/uni/index.html" target="_top"></a>

```

```

</div></td>

```

```

<td><div align="CENTER">

```

```

<a href="/uni/pages/mapasitio.htm" target="main"></a>

```

```

</div></td>

```

```

<td><div align="CENTER">

```

```

<a href="busca.forma_busca" target="main"></a>

```

```

</div></td>

```

```

<td><div align="CENTER">

```

```

<a href="/uni/pages/ayuda.htm" target="main"></a>

```

```

</div></td>

```

```

</tr>

```

```

<tr><td><div align="CENTER">

```

```

<a href="http://ns1.aviso-oportuno.com.mx" target="_top">

```

```

<font size="2" face="Arial, Helvetica, sans-serif">Regreso</font></a>

```

```

</div></td>

```

```

<td><div align="CENTER">

```

```

<font size="2" face="Arial, Helvetica, sans-serif"><a href="mailto:contacta@aguila.el-
universal.com.mx">Cont&acutes;ctanos</a></font>

```

```

</div></td>

```

```

<td><div align="CENTER">

```

```

<font face="Arial, Helvetica, sans-serif" size="2"><a href="/uni/index.html"
target="_top">Inicio</a></font>

```

```

</div></td>

```

```

<td><div align="CENTER">

```

```

<font face="Arial, Helvetica, sans-serif" size="2"><a href="/uni/pages/mapasitio.htm"
target="main">Mapa</a></font>

```

```

</div></td>

```

```

<td><div align="CENTER">

```

```

        <font face="Arial, Helvetica, sans-serif" size="2"><a href="busca.forma_busca"
target="main">B&uacute;squeda</a></font>
    </div></td>
    <td><div align="CENTER">
        <font size="2" face="Arial, Helvetica, sans-serif"><a href="/uni/pages/ayuda.htm"
target="main">Ayuda</a></font>
    </div>
    </td></tr></table></div>');
    oraclelogo;
    http.p('</BODY></HTML>');
END forma_varios;
/*****/
PROCEDURE vali_varios (p_opcion in NUMBER, p_variable in VARCHAR2) IS

v_clasi tipo_hoy.clasi%TYPE;

BEGIN
    select clasi into v_clasi from tipo_hoy where id_tipo=p_opcion;

    IF (v_clasi=1) or ((v_clasi=2) and (p_variable is NOT NULL)) THEN
        vario.busca_varios(p_variable, p_opcion);
    ELSE
        vario.subforma_varios(p_opcion);
    END IF;

END vali_varios;

/*****/
PROCEDURE busca_varios(p_variable in VARCHAR2, p_opcion in NUMBER) IS

score char(5);
pk char(5);
curid number;
texto varchar2(4000);
v_contador NUMBER:=0;
v_titulo tipo_hoy.titulo%TYPE;

CURSOR cur1 IS
    SELECT texto
    FROM varios
    WHERE tipo=p_opcion;

BEGIN
    select titulo into v_titulo from tipo_hoy where id_tipo=p_opcion;

    IF (p_variable is NOT NULL) THEN
        dbms_output.enable(1000000);
        curid := ctx_query.open_con(policy_name => 'poli_varios',
            text_query => p_variable,
            score_sorted => true,
            other_cols => 'texto',
            struct_query => 'tipo=||p_opcion||');
    END IF;
    http.p('<html><head><title>VARIOS</title>
    <meta name="author" content="Iweb S.A. de C.V.1998 " Calzada de las Aguilas 101
    tel 5936110 y 5936233 MZxico D.F."></head>

```

```

<body bgcolor="#FFFFFF"><div align="CENTER">
<table border="0" width="468">
<tr>
<td colspan="2"><font color="#666666"></font><font color="#666666">
</font></td>
</tr>
<tr>
<td colspan="2"></td>
</tr>
</table></div>);
htp.p ('<center> <BR>Tema: <FONT SIZE=3
COLOR="#006699"><B>||v_titulo||</B></FONT><BR>');
htp.tableOpen;
IF (p_variable is NOT NULL) THEN
htp.p('Palabra: <FONT SIZE=3 COLOR="#FF2211">||p_variable||</FONT><BR>');
while (ctx_query.fetch_hit(curid,pk,score,texto)>0)
loop
htp.p('<TR><TD><IMG SRC = "/uni/images/fig13.gif"><BR></TD><TD>||texto||</TD></TR>
');
v_contador:=v_contador+1;
end loop;
ctx_query.close_con(curid);
ELSE
FOR i IN cur1 loop
htp.p('<TR><TD><IMG SRC = "/uni/images/fig13.gif"><BR></TD><TD>||i.texto||</TD></TR>
');
v_contador:=v_contador+1;
END LOOP;
END IF;
htp.tableClose;
htp.p('<BR>Número de Registros Encontrados: <FONT SIZE=4 COLOR="#FF2211"><B>
||v_contador||</B> </FONT><BR>');
htp.p('<br><div align="CENTER">
<table border="0" width="280" cellspacing="0" cellpadding="0" vspace="0" hspace="0">
<tr><td><div align="CENTER">
<a href="http://ns1.avisooportuno.com.mx/noti/plsql/vario.forma_varios">
</a>
</div></td>
<td><div align="CENTER">
<a href="mailto:contacta@aguila.el-universal.com.mx"></a>
</div></td>
<td><div align="CENTER">
<a href="/uni/index.html" target="_top"></a>
</div></td>
<td><div align="CENTER">
<a href="/uni/pages/mapasitio.htm" target="main"></a>
</div></td>
<td><div align="CENTER">
<a href="busca.forma_busca" target="main"></a>
</div></td>
<td><div align="CENTER">

```

```

        <a href="/uni/pages/ayuda.htm" target="main"></a>
    </div></td>
</tr>
<tr><td><div align="CENTER">
    <a href="http://ns1.aviso-oportuno.com.mx/noti/plsql/vario.forma_varios">
    <font size="2" face="Arial, Helvetica, sans-serif">Regreso</font></a>
    </div></td>
<td><div align="CENTER">
    <font size="2" face="Arial, Helvetica, sans-serif"><a href="mailto:contacta@aguila.el-
universal.com.mx">Cont&aacute;tanos</a></font>
    </div></td>
<td><div align="CENTER">
    <font face="Arial, Helvetica, sans-serif" size="2"><a href="/uni/index.html"
target="_top">Inicio</a></font>
    </div></td>
<td><div align="CENTER">
    <font face="Arial, Helvetica, sans-serif" size="2"><a href="/uni/pages/mapasitio.htm"
target="main">Mapa</a></font>
    </div></td>
<td><div align="CENTER">
    <font face="Arial, Helvetica, sans-serif" size="2"><a href="busca.forma_busca"
target="main">B&uacute;squeda</a></font>
    </div></td>
<td><div align="CENTER">
    <font size="2" face="Arial, Helvetica, sans-serif"><a href="/uni/pages/ayuda.htm"
target="main">Ayuda</a></font>
    </div>
</td></tr></table></div>');
oraclelogo;
http.p('</BODY></HTML>');
END busca_varios;

```

```

/*****

```

```

PROCEDURE subforma_varios (p_opcion in NUMBER)IS

```

```

v_titulo  subtipo_hoy.titulo%TYPE;

```

```

BEGIN

```

```

select titulo into v_titulo from tipo_hoy where id_tipo=p_opcion;

```

```

http.p('<html><head><title>VARIOS</title>

```

```

<meta name="author" content="Iweb S.A. de C.V.1998 " Calzada de las Aguilas 101
tel 5936110 y 5936233 MZxico D.F."></head>

```

```

<body bgcolor="#FFFFFF"><div align="CENTER">

```

```

<table border="0" width="468">

```

```

<tr>

```

```

    <td colspan="2"><font color="#666666"></font><font color="#666666">

```

```

    </font></td>

```

```

</tr>

```

```

<tr>

```

```

    <td colspan="2"></td>

```

```

</tr>

```

```

</table></div>

```

```

<CENTER><FONT COLOR="#006699" FACE="Arial, Helvetica, sans-serif"
SIZE=2><B>||v_titulo||</B></FONT></CENTER>

```

```

<CENTER><table border="0" width="500"><TR><TD>
<FONT COLOR="#000000" SIZE=2>Seleccione un tema de su interés, aquellos temas marcados
con ** contienen subtemas,
  (los cuales pueden verse sólo dando click en el botón BUSCAR)</TD></TR></FONT>
</table></CENTER>);
http.formOpen('vario.subbusca_varios');
http.p(' <CENTER>
  <table cellspacing="0">
    <tr><td><FONT COLOR="#000000" face="Arial, Helvetica, sans-serif" SIZE=2>Tema de
busqueda :</td><td>');
    http.formSelectOpen(cname=>'p_subopcion');
    for cur_opp in (SELECT TITULO,ID FROM subtipo_hoy WHERE tipo=p_opcion)
    LOOP

    http.FormSelectOption(cur_opp.titulo.'SELECTED',cattrributes=>'VALUE=""|{cur_opp.id}|');
    END LOOP;
    http.p('</select></FONT></td></tr>
    <tr><td width="500" colspan="2"><FONT COLOR="#000000" SIZE=2>Si desea ser más
específico, escriba sus opciones en la caja de "Palabras a Buscar"</font></td></tr>
    <tr><td>
    <FONT COLOR="#000000" face="Arial, Helvetica, sans-serif" SIZE=2>Palabras a
buscar:</font></td>
    <td><input type="text" size="30" name="p_subvariable">
    </td></tr></table><CENTER>
    <FONT COLOR="#000000" face="Arial, Helvetica, sans-serif" SIZE=2><input
type="submit" value="Buscar"></font>
    <FONT COLOR="#000000" face="Arial, Helvetica, sans-serif" SIZE=2><input
type="reset" value="Borrar"></font></CENTER>);
http.formClose;
http.p('<div align="CENTER">
  <table border="0" width="280" cellspacing="0" cellpadding="0" vspace="0" hspace="0">
  <tr><td><div align="CENTER">
    <a href="vario.forma_varios">
    </a>
  </div></td>
  <td><div align="CENTER">
    <a href="mailto:contacta@aguila.ei-universal.com.mx"></a>
  </div></td>
  <td><div align="CENTER">
    <a href="/uni/index.html" target="_top"></a>
  </div></td>
  <td><div align="CENTER">
    <a href="/uni/pages/mapasitio.htm" target="main"></a>
  </div></td>
  <td><div align="CENTER">
    <a href="busca.forma_busca" target="main"></a>
  </div></td>
  <td><div align="CENTER">
    <a href="/uni/pages/ayuda.htm" target="main"></a>
  </div></td>
  </tr>

```

```
|  |  |
| --- | --- |
| Regreso | Contactanos |
| Inicio | Mapa |
| Búsqueda | Ayuda |

```

END subforma_varios;

*****/

PROCEDURE subbusca_varios(p_subopcion in NUMBER,p_subvariable in VARCHAR2) IS

```

score char(5);
pk char(5);
curid number;
texto varchar2(4000);
v_contador NUMBER:=0;
v_tipo subtipo_hoy.tipo%TYPE;
v_id_subtipo subtipo_hoy.id_subtipo%TYPE;
v_titulo subtipo_hoy.titulo%TYPE;

```

CURSOR cur1 IS

```

SELECT texto,tipo,subtipo
FROM varios
WHERE subtipo = (select id_subtipo from subtipo_hoy where id=p_subopcion)
AND tipo = (select tipo from subtipo_hoy where id=p_subopcion);

```

BEGIN

```

SELECT tipo,id_subtipo,titulo
into v_tipo,v_id_subtipo,v_titulo from subtipo_hoy where id=p_subopcion;

```

IF (p_subvariable is NOT NULL) THEN

```

curid := ctx_query.open_con(policy_name => 'poli_varios',
text_query => p_subvariable,
score_sorted => true,
other_cols => 'texto',
struct_query => 'tipo=||v_tipo||' and subtipo=||v_id_subtipo||');

```

```

END IF;
http.p ('<html><head><title>VARIOS</title>
<meta name="author" content="lweb S.A. de C.V.1998 " Calzada de las Aguilas 101
tel 5936110 y 5936233 MZxico D.F."></head>
<body bgcolor="#FFFFFF"><div align="CENTER">
<table border="0" width="468">
<tr><td colspan="2"><font color="#666666"></font><font color="#666666">
</font></td>
</tr><tr>
<td colspan="2"></td>
</tr></table></div>');
http.p ('<center> <BR>Tema: <FONT SIZE=3
COLOR="#006699"><B>||v_titulo||</B></FONT><BR>');
http.tableOpen;
IF (p_subvariable is NOT NULL) THEN
http.p('Palabra: <FONT SIZE=3 COLOR="#FF2211">||p_subvariable||</FONT><BR>');
while (ctx_query.fetch_hit(curid,pk,score,texto)>0)
loop
http.p('<TR><TD><IMG SRC = "/uni/images/fig13.gif"><BR></TD><TD>||texto||</TD></TR>
');
v_contador:=v_contador+1;
end loop;
ctx_query.close_con(curid);
ELSE
FOR i IN cur1 loop
http.p('<TR><TD><IMG SRC = "/uni/images/fig13.gif"><BR></TD><TD>||i.texto||</TD></TR> ');
v_contador:=v_contador+1;
END LOOP;
END IF;
http.tableClose;
http.p('<BR>Número de Registros Encontrados: <FONT SIZE=4 COLOR="#FF2211"><B>
||v_contador||</B> </FONT><BR>');
http.p('<br><div align="CENTER">
<table border="0" width="280" cellspacing="0" cellpadding="0" vspace="0" hspace="0">
<tr><td><div align="CENTER">
<a href="vario.subforma_varios?p_opcion=||v_tipo||">
</a>
</div></td>
<td><div align="CENTER">
<a href="mailto:contacta@aguila.el-universal.com.mx"></a>
</div></td>
<td><div align="CENTER">
<a href="/uni/index.html" target="_top"></a>
</div></td>
<td><div align="CENTER">
<a href="/uni/pages/mapasitio.htm" target="main"></a>
</div></td>
<td><div align="CENTER">
<a href="busca.forma_busca" target="main"></a>
</div></td>
<td><div align="CENTER">

```

```

        <a href="/uni/pages/ayuda.htm" target="main"></a>
    </div></td>
</tr>
<tr><td><div align="CENTER">
    <a href="vario.subforma_varios?p_opcion=||v_tipo||">
    <font size="2" face="Arial, Helvetica, sans-serif">Regreso</font></a>
    </div></td>
    <td><div align="CENTER">
    <font size="2" face="Arial, Helvetica, sans-serif"><a href="mailto:contacta@aguila.el-
universal.com.mx">Cont&acute;tanos</a></font>
    </div></td>
    <td><div align="CENTER">
    <font face="Arial, Helvetica, sans-serif" size="2"><a href="/uni/index.html"
target="_top">Inicio</a></font>
    </div></td>
    <td><div align="CENTER">
    <font face="Arial, Helvetica, sans-serif" size="2"><a href="/uni/pages/mapasitio.htm"
target="main">Mapa</a></font>
    </div></td>
    <td><div align="CENTER">
    <font face="Arial, Helvetica, sans-serif" size="2"><a href="busca.forma_busca"
target="main">B&uacute;squeda</a></font>
    </div></td>
    <td><div align="CENTER">
    <font size="2" face="Arial, Helvetica, sans-serif"><a href="/uni/pages/ayuda.htm"
target="main">Ayuda</a></font>
    </div>
    </td></tr></table></div>);
oraclelogo;
http.p('</BODY></HTML>');
END subbusca_varios;

END vario;
/

```

/ C3digo fuente del m3dulo Vivienda **/**

CREATE OR REPLACE PACKAGE vivienda AS

```

PROCEDURE forma_vivienda;
PROCEDURE vali_vivienda (p_opcion in NUMBER, p_variable in VARCHAR2);
PROCEDURE busca_vivienda (p_variable in VARCHAR2, p_opcion in NUMBER);
PROCEDURE subforma_vivienda (p_opcion in NUMBER);
PROCEDURE subbusca_vivienda(p_subopcion in NUMBER,p_subvariable in VARCHAR2 );

```

END vivienda;

CREATE OR REPLACE PACKAGE BODY vivienda AS

PROCEDURE forma_vivienda IS

BEGIN

http.p('<html><head>

<title>BIENES INMUEBLES</title>

<meta name="author" content="Iweb S.A. de C.V.1998 " Calzada de las Aguilas 101

```

tel 5936110 y 5936233 Mxico D.F.">
</head><body bgcolor="#FFFFFF">
<div align="CENTER"><table border="0" width="468">
<tr>
<td colspan="2"><font color="#666666"></font><font color="#666666">
</font></td>
</tr>
<tr>
<td colspan="2"></td>
</tr>
</table></div><BR>
<CENTER><FONT COLOR="#330033" face="Arial, Helvetica, sans-serif" SIZE=2><B>Bienes
Inmuebles</B></FONT></CENTER>
<CENTER><table border="0" width="500"><TR><TD>
<FONT COLOR="#000000" SIZE=2>Seleccione un tema de su interés, aquellos temas marcados
con ** contienen subtemas,
(los cuales pueden verse sólo dando click en el botón BUSCAR)</TD></TR></FONT>
</table></CENTER>');
http.formOpen('vivienda.vali_vivienda');
http.p('<CENTER>
<table cellspacing="0">
<tr><td><FONT COLOR="#000000" face="Arial, Helvetica, sans-serif" SIZE=2>Tema de
busqueda :</td><td>');
http.formSelectOpen(cname=>'p_opcion');
FOR cur_op in (SELECT TITULO,ID_TIPO FROM tipo_hoy WHERE padre=95) LOOP
http.FormSelectOption(cur_op.titulo,'SELECTED',cattributes=>'VALUE="||cur_op.id_tipo||"');
END LOOP;
http.p('</select></FONT></td></tr>
<tr><td>
<FONT COLOR="#000000" face="Arial, Helvetica, sans-serif" SIZE=2>Palabras a
buscar:</font></td>
<td><input type="text" size="30" name="p_variable">
</td></tr></table><CENTER>
<FONT COLOR="#000000" face="Arial, Helvetica, sans-serif" SIZE=2><input
type="submit" value="Buscar"></font>
<FONT COLOR="#000000" face="Arial, Helvetica, sans-serif" SIZE=2><input
type="reset" value="Borrar"></font></CENTER>');
http.formClose;
http.p('<div align="CENTER">
<table border="0" width="280" cellspacing="0" cellpadding="0" vspace="0" hspace="0">
<tr><td><div align="CENTER">
<a href="http://ns1.avisooportuno.com.mx" target="_top">
</a>
</div></td>
<td><div align="CENTER">
<a href="mailto:contacta@aguila.el-universal.com.mx"></a>
</div></td>
<td><div align="CENTER">
<a href="/uni/index.html" target="_top"></a>
</div></td>
<td><div align="CENTER">
<a href="/uni/pages/mapasitio.htm" target="main"></a>

```

```

</div></td>
<td><div align="CENTER">
  <a href="busca.forma_busca" target="main"></a>
</div></td>
<td><div align="CENTER">
  <a href="/uni/pages/ayuda.htm" target="main"></a>
</div></td>
</tr>
<tr><td><div align="CENTER">
  <a href="http://ns1.avisooportuno.com.mx" target="_top">
  <font size="2" face="Arial, Helvetica, sans-serif">Regreso</font></a>
</div></td>
<td><div align="CENTER">
  <font size="2" face="Arial, Helvetica, sans-serif"><a href="mailto:contacta@aguila.el-
universal.com.mx">Cont&acute;tanos</a></font>
</div></td>
<td><div align="CENTER">
  <font face="Arial, Helvetica, sans-serif" size="2"><a href="/uni/index.html"
target="_top">Inicio</a></font>
</div></td>
<td><div align="CENTER">
  <font face="Arial, Helvetica, sans-serif" size="2"><a href="/uni/pages/mapasitio.htm"
target="main">Mapa</a></font>
</div></td>
<td><div align="CENTER">
  <font face="Arial, Helvetica, sans-serif" size="2"><a href="busca.forma_busca"
target="main">B&uacute;squeda</a></font>
</div></td>
<td><div align="CENTER">
  <font size="2" face="Arial, Helvetica, sans-serif"><a href="/uni/pages/ayuda.htm"
target="main">Ayuda</a></font>
</div>
</td></tr></table></div>);
oraclelogo;
http.p('</BODY></HTML>');

```

END forma_vivienda;

/*****/

PROCEDURE vali_vivienda (p_opcion in NUMBER, p_variable in VARCHAR2) IS

v_clasi tipo_hoy.clasi%TYPE;

BEGIN

select clasi into v_clasi from tipo_hoy where id_tipo=p_opcion;

IF (v_clasi=1) or ((v_clasi=2) and (p_variable is NOT NULL)) THEN

vivienda.busca_vivienda(p_variable, p_opcion);

ELSE

vivienda.subforma_vivienda(p_opcion);

END IF;

END vali_vivienda;

/*****/

PROCEDURE busca_vivienda(p_variable in VARCHAR2, p_opcion in NUMBER) IS

```

score char(5);
pk char(5);
curid number;
texto varchar2(4000);
v_contador NUMBER:=0;
v_titulo tipo_hoy.titulo%TYPE;

CURSOR cur1 IS
  SELECT texto
  FROM inmuebles
  WHERE tipo=p_opcion;

BEGIN
  select titulo into v_titulo from tipo_hoy where id_tipo=p_opcion;
  IF (p_variable is NOT NULL) THEN
    dbms_output.enable(1000000);
    curid := ctx_query.open_con(policy_name => 'poli_inmuebles',
                               text_query => p_variable,
                               score_sorted => true,
                               other_cols => 'texto',
                               struct_query => 'tipo=||p_opcion||');
  END IF;
  http.p('<html><head>
  <title>BIENES INMUEBLES</title>
  <meta name="author" content="Iweb S.A. de C.V.1998 " Calzada de las Aguilas 101
  tel 5936110 y 5936233 Mxico D.F.">
  </head><body bgcolor="#FFFFFF">
  <div align="CENTER"><table border="0" width="468">
  <tr>
    <td colspan="2"><font color="#666666"></font><font color="#666666">
    </font></td>
  </tr>
  <tr>
    <td colspan="2"></td>
  </tr>
  </table></div>');
  http.p('(<center> <BR>Tema: <FONT COLOR="#330033"
  SIZE=3><B>'||v_titulo||</B></FONT><BR>');
  http.tableOpen;
  IF (p_variable is NOT NULL) THEN
    http.p('Palabra: <FONT SIZE=3 COLOR="#FF2211"><B>'||p_variable||</B></FONT><BR><BR>');
    while (ctx_query.fetch_hit(curid,pk,score,texto)>0)
      loop
        http.p('<TR><TD><IMG SRC = "/uni/images/fig13.gif"><BR></TD><TD>'||texto||</TD></TR>');
      v_contador:=v_contador+1;
    end loop;
    ctx_query.close_con(curid);
  ELSE
    FOR i IN cur1 loop
      http.p('<TR><TD><IMG SRC = "/uni/images/fig13.gif"><BR></TD><TD>'||i.texto||</TD></TR>');
    v_contador:=v_contador+1;
  END LOOP;
  END IF;

```

```

http.tableClose;
http.p('<BR>Número de Registros Encontrados: <FONT SIZE=4 COLOR="#FF2211"><B>
'v_contador'|</B></FONT><BR>');
http.p('<br><div align="CENTER">
<table border="0" width="280" cellspacing="0" cellpadding="0" vspace="0" hspace="0">
<tr><td><div align="CENTER">
<a href="vivienda.forma_vivienda">
</a>
</div></td>
<td><div align="CENTER">
<a
href="mailto:contacta@aguila.el-universal.com.mx"></a>
</div></td>
<td><div align="CENTER">
<a href="/uni/index.html" target="_top"></a>
</div></td>
<td><div align="CENTER">
<a href="/uni/pages/mapasitio.htm" target="main"></a>
</div></td>
<td><div align="CENTER">
<a href="busca.forma_busca" target="main"></a>
</div></td>
<td><div align="CENTER">
<a href="/uni/pages/ayuda.htm" target="main"></a>
</div></td>
</tr>
<tr><td><div align="CENTER">
<a href="vivienda.forma_vivienda">
<font size="2" face="Arial, Helvetica, sans-serif">Regreso</font></a>
</div></td>
<td><div align="CENTER">
<font size="2" face="Arial, Helvetica, sans-serif"><a href="mailto:contacta@aguila.el-
universal.com.mx">Cont&aacute;ctanos</a></font>
</div></td>
<td><div align="CENTER">
<font face="Arial, Helvetica, sans-serif" size="2"><a href="/uni/index.html"
target="_top">Inicio</a></font>
</div></td>
<td><div align="CENTER">
<font face="Arial, Helvetica, sans-serif" size="2"><a href="/uni/pages/mapasitio.htm"
target="main">Mapa</a></font>
</div></td>
<td><div align="CENTER">
<font face="Arial, Helvetica, sans-serif" size="2"><a href="busca.forma_busca"
target="main">B&uacute;squeda</a></font>
</div></td>
<td><div align="CENTER">
<font size="2" face="Arial, Helvetica, sans-serif"><a href="/uni/pages/ayuda.htm"
target="main">Ayuda</a></font>
</div>
</td></tr></table></div>');
oraclelogo;

```

```

    http.p('</BODY></HTML>');
END busca_vivienda;

```

```

/*****
PROCEDURE subforma_vivienda (p_opcion in NUMBER)IS

```

```

v_titulo  subtipo_hoy.titulo%TYPE;

```

```

BEGIN

```

```

    select titulo into v_titulo from tipo_hoy where id_tipo=p_opcion;

```

```

    http.p('<html><head>

```

```

        <title>BIENES INMUEBLES</title>

```

```

        <meta name="author" content="lweb S.A. de C.V.1998 " Calzada de las Aguilas 101
        tel 5936110 y 5936233 MEXICO D.F.">

```

```

    </head><body bgcolor="#FFFFFF">

```

```

    <div align="CENTER"><table border="0" width="468">

```

```

    <tr>

```

```

        <td colspan="2"><font color="#666666"></font><font color="#666666">

```

```

        </font></td>

```

```

    </tr>

```

```

    <tr>

```

```

        <td colspan="2"></td>

```

```

    </tr>

```

```

    </div></table></div>

```

```

        <CENTER><FONT COLOR="#330033" face="Arial, Helvetica, sans-serif"

```

```

        SIZE=2><B>||v_titulo||</B></FONT><BR>

```

```

        <CENTER><table border="0" width="500"><TR><TD>

```

```

        <FONT COLOR="#000000" SIZE=2>Seleccione un tema de su interés, aquellos temas marcados
        con ** contienen subtemas,

```

```

        (los cuales pueden verse sólo dando click en el botón BUSCAR)</TD></TR></FONT>

```

```

        </table></CENTER>");

```

```

    http.formOpen('vivienda.subbusca_vivienda');

```

```

    http.p(' <CENTER>

```

```

        <table cellspacing="0">

```

```

        <tr><td><FONT COLOR="#000000" face="Arial, Helvetica, sans-serif" SIZE=2>Tema de
        busqueda :</td><td>);

```

```

        http.formSelectOpen(cname=>'p_subopcion');

```

```

        for cur_opp in (SELECT TITULO,ID FROM subtipo_hoy WHERE tipo=p_opcion)

```

```

            LOOP

```

```

                http.FormSelectOption(cur_opp.titulo,'SELECTED',cattributes=>'VALUE="||cur_opp.id||"');

```

```

            END LOOP;

```

```

        http.p('</select></FONT></td></tr>

```

```

        <tr><td width="500" colspan="2"><FONT COLOR="#000000" SIZE=2>Si desea ser más
        específico, escriba sus opciones en la caja de "Palabras a Buscar"</font></td></tr>

```

```

        <tr><td>

```

```

        <FONT COLOR="#000000" face="Arial, Helvetica, sans-serif" SIZE=2>Palabras a
        buscar:</font> </td>

```

```

        <td><input type="text" size="30" name="p_subvariable">

```

```

        </td></tr></table><CENTER>

```

```

        <FONT COLOR="#000000" face="Arial, Helvetica, sans-serif" SIZE=2><input
        type="submit" value="Buscar"></font>

```

```

        <FONT COLOR="#000000" face="Arial, Helvetica, sans-serif" SIZE=2><input
        type="reset" value="Borrar"></font></CENTER>);

```

```

    http.formClose;

```

```

    http.p('<div align="CENTER">

```

```

<table border="0" width="280" cellspacing="0" cellpadding="0" vspace="0" hspace="0">
<tr><td><div align="CENTER">
  <a href="vivienda.forma_vivienda">
    </a>
  </div></td>
<td><div align="CENTER">
  <a href="mailto:contacta@aguila.el-universal.com.mx"></a>
  </div></td>
<td><div align="CENTER">
  <a href="/uni/index.html" target="_top"></a>
  </div></td>
<td><div align="CENTER">
  <a href="/uni/pages/mapasitio.htm" target="main"></a>
  </div></td>
<td><div align="CENTER">
  <a href="busca.forma_busca" target="main"></a>
  </div></td>
<td><div align="CENTER">
  <a href="/uni/pages/ayuda.htm" target="main"></a>
  </div></td>
</tr>
<tr><td><div align="CENTER">
  <a href="vivienda.forma_vivienda">
    <font size="2" face="Arial, Helvetica, sans-serif">Regreso</font></a>
  </div></td>
<td><div align="CENTER">
  <font size="2" face="Arial, Helvetica, sans-serif"><a href="mailto:contacta@aguila.el-
universal.com.mx">Cont&acute;tanos</a></font>
  </div></td>
<td><div align="CENTER">
  <font face="Arial, Helvetica, sans-serif" size="2"><a href="/uni/index.html"
target="_top">Inicio</a></font>
  </div></td>
<td><div align="CENTER">
  <font face="Arial, Helvetica, sans-serif" size="2"><a href="/uni/pages/mapasitio.htm"
target="main">Mapa</a></font>
  </div></td>
<td><div align="CENTER">
  <font face="Arial, Helvetica, sans-serif" size="2"><a href="busca.forma_busca"
target="main">B&uacute;squeda</a></font>
  </div></td>
<td><div align="CENTER">
  <font size="2" face="Arial, Helvetica, sans-serif"><a href="/uni/pages/ayuda.htm"
target="main">Ayuda</a></font>
  </div>
</td></tr></table></div>);
oraclelogo;
http.p('</BODY></HTML>');

```

END subforma_vivienda;

*****/

```
PROCEDURE subbusca_vivienda(p_subopcion in NUMBER,p_subvariable in VARCHAR2) IS
```

```
score char(5);
pk char(5);
curid number;
curid2 number;
score2 char(5);
pk2 char(5);
texto varchar2(4000);
v_contador NUMBER:=0;
v_tipo subtipo_hoy.tipo%TYPE;
v_id_subtipo subtipo_hoy.id_subtipo%TYPE;
v_titulo subtipo_hoy.titulo%TYPE;
```

```
CURSOR cur1 IS
  SELECT texto,tipo,subtipo
  FROM inmuebles
  WHERE subtipo = (select id_subtipo from subtipo_hoy where id=p_subopcion)
  AND tipo = (select tipo from subtipo_hoy where id=p_subopcion);
```

```
CURSOR cur2 IS
  SELECT texto,tipo,subtipo
  FROM inmuebles
  WHERE subvi = (select substr(id_subtipo,1,2) from subtipo_hoy where id=p_subopcion)
  AND tipo = (select tipo from subtipo_hoy where id=p_subopcion);
```

```
BEGIN
  SELECT titulo,tipo,id_subtipo into v_titulo,v_tipo,v_id_subtipo from subtipo_hoy where id=p_subopcion;
```

```
IF (p_subvariable is NOT NULL) THEN
  IF (v_tipo=39) or (v_tipo=5) THEN
    curid2 := ctx_query.open_con(policy_name => 'poli_inmuebles',
      text_query => p_subvariable,
      score_sorted => true,
      other_cols => 'texto',
      struct_query => 'tipo='||v_tipo||' and subvi='||v_id_subtipo||'');
```

```
ELSE
  curid := ctx_query.open_con(policy_name => 'poli_inmuebles',
    text_query => p_subvariable,
    score_sorted => true,
    other_cols => 'texto',
    struct_query => 'tipo='||v_tipo||' and subtipo='||v_id_subtipo||'');
```

```
END IF;
```

```
END IF;
```

```
htp.p('<html><head>
```

```
  <title>BIENES INMUEBLES</title>
```

```
  <meta name="author" content="Iweb S.A. de C.V.1998 " Calzada de las Aguilas 101
```

```
  tel 5936110 y 5936233 Mxico D.F.">
```



```
-----
  <div align="CENTER"><table border="0" width="468">
```

```
    <tr>
```

```
      <td colspan="2"><font color="#666666"></font><font color="#666666">
```

```
      </font></td>
```

```
    </tr>
```

```

<tr>
<td colspan="2"></td>
</tr>
</table></div>');
http.p ('<center> <BR>Tema: <FONT SIZE=3
COLOR="#330033"><B>'||v_titulo||'</B></FONT><BR><BR>');
http.tableOpen;
IF (p_subvariable is NOT NULL) THEN
http.p('Palabra: <FONT SIZE=3 COLOR="#FF2211">'||p_subvariable||'</FONT><BR>');
IF (v_tipo=39) or (v_tipo=5) THEN
while (ctx_query.fetch_hit(curid2,pk2,score2,texto)>0)
loop
http.p('<TR><TD><IMG SRC =
"/uni/images/fig13.gif"><BR></TD><TD>'||texto||'</TD></TR> ');
v_contador:=v_contador+1;
end loop;
ctx_query.close_con(curid2);
ELSE
while (ctx_query.fetch_hit(curid,pk,score,texto)>0)
loop
http.p('<TR><TD><IMG SRC =
"/uni/images/fig13.gif"><BR></TD><TD>'||texto||'</TD></TR> ');
v_contador:=v_contador+1;
end loop;
ctx_query.close_con(curid);
END IF;
ELSE
IF (v_tipo=39) or (v_tipo=5) THEN
FOR i IN cur2 loop
http.p('<TR><TD><IMG SRC = "/uni/images/fig13.gif"><BR></TD><TD>'||i.texto||'</TD></TR>
');
v_contador:=v_contador+1;
END LOOP;
ELSE
FOR i IN cur1 loop
http.p('<TR><TD><IMG SRC = "/uni/images/fig13.gif"><BR></TD><TD>'||i.texto||'</TD></TR>
');
v_contador:=v_contador+1;
END LOOP;
END IF;
END IF;
http.tableClose;
http.p('<BR><CENTER>Número de Registros Encontrados:<FONT SIZE=4 COLOR="#FF2211"><B>
'v_contador||'</B></FONT></CENTER><BR>');

http.p('<br><div align="CENTER">
<table border="0" width="280" cellspacing="0" cellpadding="0" vspace="0" hspace="0">
<tr><td><div align="CENTER">
<a href="vivienda.subforma_vivienda?p_opcion='v_tipo||'">
</a>
</div></td>
<td><div align="CENTER">
<a href="mailto:contacta@aguila.el-universal.com.mx"></a>
</div></td>
<td><div align="CENTER">

```

```

        <a href="/uni/index.html" target="_top"></a>
    </div></td>
    <td><div align="CENTER">
        <a href="/uni/pages/mapasitio.htm" target="main"></a>
    </div></td>
    <td><div align="CENTER">
        <a href="busca.forma_busca" target="main"></a>
    </div></td>
    <td><div align="CENTER">
        <a href="/uni/pages/ayuda.htm" target="main"></a>
    </div></td>
</tr>
<tr><td><div align="CENTER">
    <a href="vivienda.subforma_vivienda?p_opcion=||v_tipo||">
    <font size="2" face="Arial, Helvetica, sans-serif">Regreso</font></a>
</div></td>
<td><div align="CENTER">
    <font size="2" face="Arial, Helvetica, sans-serif"><a href="mailto:contacta@aguila.el-
universal.com.mx">Cont&aacute;tanos</a></font>
</div></td>
<td><div align="CENTER">
    <font face="Arial, Helvetica, sans-serif" size="2"><a href="/uni/index.html"
target="_top">Inicio</a></font>
</div></td>
<td><div align="CENTER">
    <font face="Arial, Helvetica, sans-serif" size="2"><a href="/uni/pages/mapasitio.htm"
target="main">Map&aacute;s</a></font>
</div></td>
<td><div align="CENTER">
    <font face="Arial, Helvetica, sans-serif" size="2"><a href="busca.forma_busca"
target="main">B&uacute;squeda</a></font>
</div></td>
<td><div align="CENTER">
    <font size="2" face="Arial, Helvetica, sans-serif"><a href="/uni/pages/ayuda.htm"
target="main">Ayuda</a></font>
</div>
</td></tr></table></div>);
    oraclelogo;
    hip.p("</BODY></HTML>");
END subbusca_vivienda;

```

END vivienda;

/