UNIVERSIDAD NACIONAL AUTÓNOMA DE MÉXICO

FACULTAD DE INGENIERÍA

"SISTEMA TUTORIAL DE REDES DE DATOS Y COMPLEMENTOS DIDÁCTICOS"

T E S I S

QUE PARA OBTENER EL TÍTULO DE

INGENIERO EN COMPUTACIÓN

PRESENTA

ANTONIO MONTALVO GARCÍA

DIRECTOR DE TESIS

M. C. MA. JAQUELINA LÓPEZ BARRIENTOS

MÉXICO, D.F. **2008**

No puedo cambiar la dirección del viento,
pero sí ajustar mis velas para llegar siempre a mi destino.
James Deam

Dedico esta tesis a mis padres

Con mucho amor, cariño y todo mi agradecimiento a mi padre **Antonio Montalvo Santiago**, que ha sido mi guía en la vida, gracias padre por enseñarme el arte de la obligación y la responsabilidad. Gracias por darme la oportunidad de realizar mis estudios y sobre todo gracias por darme la vida. ¡Eres el mejor, papá!

A ti madre, que eres lo que más amo y quiero en la vida, gracias por tu calor de madre, por tus atenciones y consejos. Gracias por confiar siempre en mí y en mis estudios. Tengo muy presente que aún en los momentos difíciles, tú siempre viste la manera de apoyarnos y salir adelante. ¡Gracias Dios! por darme una vida al lado de **Silveria García Hernández**.

Deseo expresar mi gratitud, y reconocimiento a las siguientes personas e instituciones por sus valiosas opiniones, orientaciones y que en su conjunto hicieron posible arribar al resultado de ésta tesis.

A la Universidad Nacional Autónoma de México

Por su grandeza reflejada en sus docentes, estudiantes y por ser la Universidad más importante del país. Qué orgullo tener una comunidad unida y trabajadora. ¡Gracias UNAM! ¡Gracias Facultad de Ingeniería!.

A mis Maestros

Motivo de mis logros y la motivación para continuar. Gracias por la cátedra recibida, ahora en el mundo laboral, veo la importancia de sus consejos.

A mi Directora de Tesis y Maestra

Mtra. Ma. Jaquelina López Barrientos, gracias por todo su apoyo y paciencia para la realización de este trabajo. Pero sobre todo, gracias por sus consejos, ya que me llenaron de fuerza y energía para concluir el objetivo.

A mi Hermano

Sergio Montalvo García que me ha apoyado en todo momento de mi vida. Gracias por traer a este mundo a una persona que ha llenado de felicidad nuestro hogar, Sergio Iván Montalvo.

A mis Amigos y Compañeros de Trabajo

Enrique Díaz Martínez, Héctor Correa Peragallo, Christian Aguilar Díaz, Rosa María Maya Chávez, Lorena Arredondo Flores, Juan Carlos Hernández de Anda, Guadalupe Aguilar Araiza. María de la Luz Pérez, Rodolfo Cancino Gómez, Lizette Rosas Becerril, Roció Gómez, María Rosa Martínez, Reina Pérez, Laura Ramírez, Leonor Espinosa, y los que me faltaron tengan presente que los llevo en siempre mi corazón.

Igualmente quiero expresar mi agradecimiento y admiración a dos grandes personas que son un gran ejemplo a seguir, Dr. Ruperto Patiño Manffer y a la Dra. Socorro Apreza Salgado, de quienes he aprendido a valorar el trabajo, la profesión y la ética. Gracias por todo su apoyo.

A la Facultad de Derecho

Por darme la oportunidad de conocer la profesión más bella y gratificante, la docencia. ¡Gracias Facultad de Derecho!.

ÍNDICE

CAPÍTULO II REDES DE ÁREA LOCAL

CAPÍTULO III ENRUTAMIENTO Y DIRECCIONAMIENTO

CAPÍTULO IV

FUNDAMENTOS DE SISTEMAS, APLICACIONES WEB Y CONCEPTOS DE DISEÑO

CAPÍTULO V DISEÑO Y DESARROLLO DEL SISTEMA

CAPÍTULO VI IMPLEMENTACIÓN Y PRUEBAS

INTRODUCCIÓN

En la actualidad, el mundo de la informática ha crecido exponencialmente en todos sus aspectos tanto de hardware como de software. Ahí podemos observar la gran importancia de las computadoras, ya que son una herramienta utilizada para infinidad de trabajos, ya que en ellas guardamos, manipulamos, enviamos y procesamos información útil. Es por esto que el desarrollo de la computación y su integración con las telecomunicaciones han propiciado el surgimiento de nuevas formas de comunicación, que son aceptadas cada vez por más personas.

Estas nuevas formas de comunicación a las que nos referimos son las redes de datos, las cuales tiene como objetivo el poder tener la información siempre disponible, verás y segura.

Es muy interesante la evolución que han tenido las telecomunicaciones, desde la invención del telégrafo (1834) de donde surgen los principios, para la comunicación electrónica. Luego de la segunda guerra mundial comenzó el desarrollo comercial de la computadora. Estas primeras computadoras eran orientadas a lotes, no existía necesidad de interconectarse con el sistema de comunicación que abarcaba toda la nación. Sin embargo, más adelante la industria observó la importancia que tenía el poder comunicarse entre sí.

Es importante mencionar que los orígenes del Internet fueron precisamente desarrollados por proyectos del ejército de los EE.UU, al cual llamaron ARPANET, el cual tenía por objetivo sobrevivir a un ataque militar de dimensiones muy grandes y poder seguir estar comunicado. Fue ahí donde ARPANET comenzó el desarrollo en forma de comunicación entre distintas redes, pero como en aquellos años existían ya diversas empresas utilizando sus propias normas, era realmente muy problemático y costoso el poder enlazar dos redes de distinta fabricación. Es ahí cuando surge la idea de crear distintos órganos de estandarización internacional como ISA (Federación Internacional de Estandarización), IEC (Comisión electrónica Internacional), y de las últimas fue la ISO (Organización Internacional de Normalización) la cual fue la encargada de regular el problema de incompatibilidad entre redes. Fue

así como desarrolló un modelo de red que ayuda a los fabricantes a crear redes que sean compatibles con otras redes, y el resultado de este esfuerzo dio como resultado la creación de un Modelo de Referencia Interconexión de Sistemas Abiertos, llamado modelo OSI.

Es un hecho que la historia nos enseña que las redes tienen un gran propósito, el tener información relevante, verás y segura. Por tal motivo se desarrollo la presente tesis, con el fin de sistematizar herramientas para obtener un mejor conocimiento en la materia, ya que su presencia esta tanto en sectores privados y públicos; ya que cada vez se requiere contar con más profesionales de la ingeniería en computación que cuenten con los conocimientos y capacidades que les permitan afrontar los retos presentes y futuros en este campo tan demandante.

Con esto en mente, la Facultad de Ingeniería de la Universidad Nacional Autónoma de México está comprometida en formar profesionistas capaces de dar solución a los requerimientos de demande la sociedad en materia de redes y administración de redes de datos.

Ésta es una de las razones por las que se decidió llevar a cabo el diseño del Sistema Tutorial de Redes de Datos, el cual tiene por objetivo ubicarlo en el servidor web del área a fin de beneficiar a diferentes sectores:

- A los profesores brinde:
  - ✓ Material didáctico adicional para complementar sus clases.
  - ✓ Mayor cúmulo de actividades para realizar ejercicios, tareas, series, etc.
- A los alumnos permita:
  - ✓ Llevar a cabo una serie de prácticas en cuanto al conocimiento y a la administración en redes locales.
  - ✓ Estudiar y reafirmar conocimientos vistos en clase así como adquirir nuevos conocimientos referentes a las Redes Locales, su Administración y su Seguridad.

- ✓ Complementar su formación profesional en este campo del conocimiento.

- A público en general:
  - ✓ Adquirir nuevos conocimientos y habilidades en redes y seguridad a todos aquellos interesados en el área.
  - ✓ La oportunidad de actualizarse en el campo de redes y seguridad a profesionales de la computación que así lo deseen.

Así mismo este sistema fue desarrollado con aspectos fundamentales de redes de datos, prácticas, comandos y terminología de manera que sea posible acceder a ellos sin necesidad de acudir a infinidad de páginas en Internet en donde la información puede ser o no confiable. Es necesario que este tutorial le permita a los usuarios leer y visualizar ejemplos ilustrados y animados en donde encuentren información relevante, bibliografías y links a sitios confiables que les ayuden a profundizar en temas de su interés.

El diseño de la investigación realizada es un estudio de caso basado en el método de análisis cualitativo, para el cual se analizaron textos especializados que tratan sobre el tema, retomando información de manuales de redes de datos, administración linux, Cisco e Internet, así como la experiencia práctica que se ha adquirido a través del desarrollo de actividades profesionales en este campo. Dentro de las materias involucradas en el desarrollo del proyecto están sistemas operativos, redes de computadoras, ingeniería de programación, programación de sistemas, bases de datos.

Así, inicia el capítulo I, con la explicación de conceptos teóricos de los modelos de referencia (OSI y TCP/IP) y su arquitectura, la explicación de cada una de las capas y el detalle de los elementos que componen a los protocolos más utilizados. En el capítulo II se estudian los diferentes tipos de redes de acuerdo con su clasificación, cobertura, modelos de comunicación y los diversos componentes que integran a una red.

En el Capítulo III, se presentan temas de suma importancia en las redes, como son las clases de direcciones (Ipv4 e Ipv6), direccionamiento IP, la importancia de NAT para el funcionamiento de diversas direcciones IP a partir de una. Y se describen cada uno de los tipos de algoritmos de ruteo además de sus características principales.

Es importante mencionar en cualquier proyecto de software, las herramientas para su desarrollo, las fases del ciclo de vida del software y por su puesto las herramientas de implementación, ya que son parte fundamental para una buena planeación del sistema. Lo anterior se aborda en el capítulo IV.

La fase de desarrollo e implementación del sistema, se describe en el capítulo V, en donde se enriquece con imágenes, diagramas de distribución de paquetes, ventajas y desventajas de las herramientas usadas en el sistema, los diagramas de la base de datos y los esquemas de cada uno de los tutoriales, así como las pantallas finales de cada uno de ellos.

Posteriormente, la etapa de pruebas e implementación, donde se dan a conocer los tipos de pruebas que se realizaron y que son las pertinentes en el Laboratorio de Redes y Seguridad, capturando las pantallas finales y haciendo mención de las características en las que fue implementado el sistema, así como las fallas encontradas hasta ese momento y su solución.

Es relevante mencionar, que se ha creado un diccionario Web, que muestra en forma ordenada una recopilación de las palabras más utilizadas en la Administración y Seguridad, ayudando con esto a que el usuario se familiarice con el uso de la terminología.

CAPÍTULO I

MODELOS DE REFERENCIA

1. Introducción

Durante las últimas dos décadas ha habido un enorme crecimiento en la cantidad y tamaño de las redes. Muchas de ellas, sin embargo, se desarrollaron utilizando implementaciones de hardware y software diferentes. Como resultado, muchas de las redes eran incompatibles y se volvió muy difícil para las redes que utilizaban especificaciones distintas poder comunicarse entre sí. Para solucionar este problema, la Organización Internacional para la Normalización (ISO) realizó varias investigaciones acerca de los esquemas de red. La ISO reconoció que era necesario crear un modelo de red que pudiera ayudar a los diseñadores de red a implementar redes que pudieran comunicarse y trabajar en conjunto (interoperabilidad).

En sus inicios, el desarrollo de redes sucedió con desorden en muchos sentidos. A medida que las empresas tomaron conciencia de las ventajas de usar tecnología de red, agregaban o expandían a casi la misma velocidad a la que se introducían nuevas tecnologías de la misma.

Para mediados de la década de 1980, estas empresas comenzaron a sufrir las consecuencias de la rápida expansión. De la misma forma en que las personas que no hablan un mismo idioma tienen dificultades para comunicarse, las redes que utilizaban diferentes especificaciones e implementaciones tenían dificultades para intercambiar información. El mismo problema surgía con las empresas que desarrollaban tecnologías de red privadas o propietarias. "Propietario" significa que una sola empresa o un pequeño grupo de empresas controla todo uso de la tecnología, respetando reglas propietarias en forma estricta ya que no podían comunicarse con tecnologías que usaban reglas propietarias diferentes.

Para enfrentar el problema de incompatibilidad de redes, la ISO investigó modelos de redes como la red de Digital (DECnet), la Arquitectura de Sistemas de Red (SNA) y TCP/IP a fin de encontrar un conjunto de reglas aplicables de forma general a todas las redes.

1.1 Arquitectura del modelo OSI

Un modelo define una arquitectura de comunicación estructurada, de manera que el modelo OSI se presenta en siete niveles verticales. Cada nivel ejecuta un subconjunto de las funciones que se requieren para comunicar con el otro sistema. Para ello se apoya en los servicios que le ofrece el nivel inmediato inferior y ofrece sus servicios al nivel que está por encima de él (véase Figura 1.1). Idealmente, los cambios que se realicen en un nivel no deberían afectar a su nivel vecino mientras no se modifiquen los servicios que le ofrece.

La tarea del subcomité ISO fue definir el conjunto de niveles y los servicios proporcionados por cada nivel. Los principios aplicados para establecer un nivel fueron los siguientes:

- Diferentes niveles deben corresponder a diferentes niveles de abstracción en el manejo de los datos (por ejemplo diferencias en la morfología, la sintaxis, la semántica).
- Cada nivel debe ejecutar una función bien definida.
- Aprovechar la experiencia de protocolos anteriores. Las fronteras de niveles deben situarse donde la experiencia ha demostrado que son convenientes.
- Establecer las divisiones de los niveles de forma que se minimice el flujo de información entre ellos.
- El número de niveles debe ser suficiente para que no agrupen funciones distintas, pero no tan grande que haga la arquitectura inmanejable.
- Permitir que las modificaciones de funciones o protocolos que se realicen en un nivel no afecten a los niveles contiguos.
- Cada nivel debe interaccionar únicamente con los niveles contiguos a él (superior e inferiormente).

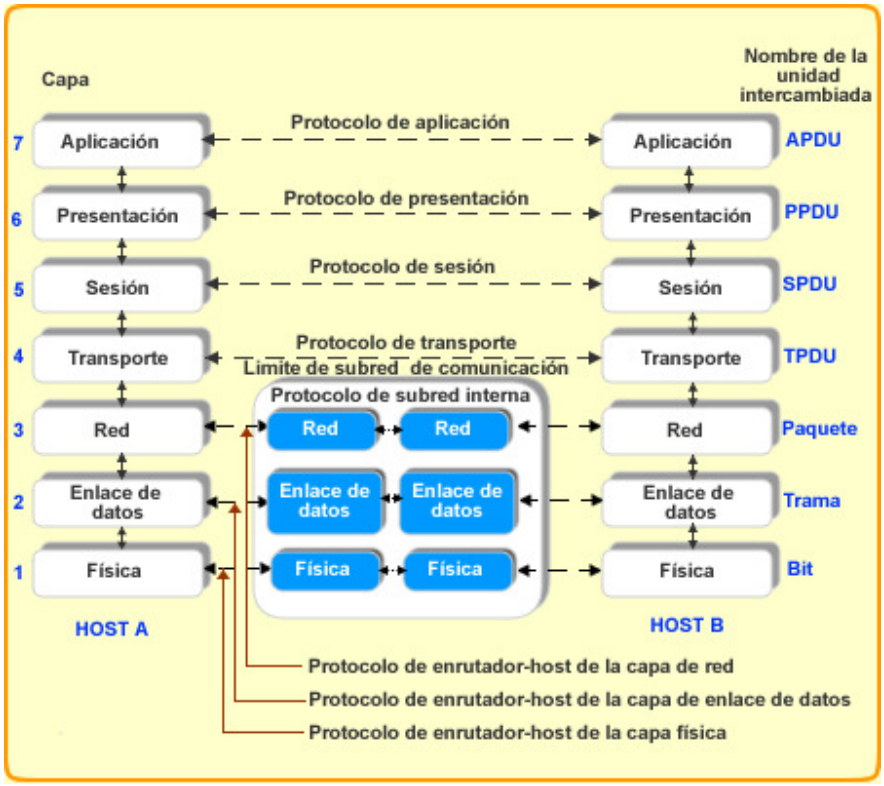


Figura 1.1. Arquitectura del modelo OSI

1.2 Conexiones del Modelo OSI

El modelo de referencia OSI es orientado a la conexión. Esto significa que, en todos los niveles, es necesario que se establezca previamente una conexión para que pueda existir intercambio de datos. Sin embargo, existen protocolos que no requieren esta condición, son los no orientados a la conexión (connectionless).

En las comunicaciones orientadas a la conexión se invierte tiempo y procesamiento en establecer y liberar la conexión entre dos nodos, pero se garantiza que el nodo remoto está escuchando. Por el contrario, en las comunicaciones no orientadas a la conexión se ahorra tiempo y procesamiento, pero a costa de no saber si el otro extremo está o no escuchando.

En cualquier nivel del modelo OSI se puede apreciar que a nivel N-1 se establece una asociación, una conexión N-1, para que dos entidades de nivel N puedan

comunicarse. La conexión N-1 es un servicio ofrecido por el nivel N-1, a través de la cual circulan unidades de información del nivel N (véase figura 1.2).

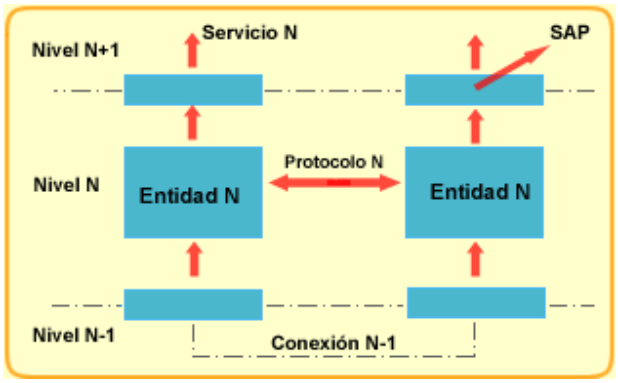


Figura 1.2. Establecimiento de conexión

El Punto de Acceso al Servicio (SAP) del nivel N identifica la dirección del nivel N a la que se conectan las entidades de nivel N+1. La relación entre direcciones de dos niveles consecutivos puede ser 1 a 1, N a 1 ó 1 a N (véase figura 1.3).

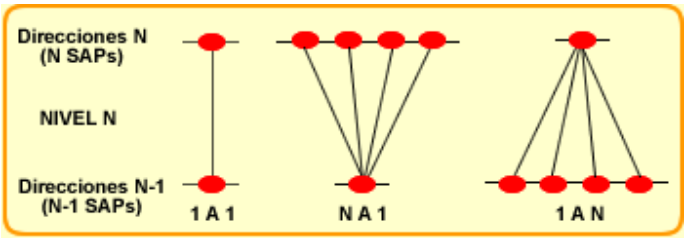


Figura 1.3. Conexión entre dos entidades de nivel N

Para que dos entidades N tengan un intercambio de información de manera eficiente y confiable es necesario que las entidades establezcan una conexión, la mantengan y finalmente la concluyan al no haber más datos que transferir.

A) Establecimiento de conexiones

Para que dos entidades N establezcan una conexión, es necesario:

- Disponer de una conexión N-1 por debajo. En los niveles bajos, es necesario establecer previamente la conexión N-1 antes de intentar establecer la conexión N, descendiendo hasta que se encuentra una disponible a nivel físico (el nivel más bajo). Sin embargo, en los niveles altos, se aprovecha la circunstancia de establecimiento de la conexión N para establecer, al mismo tiempo, la conexión N+1.
- Estar ambas entidades conformes con el establecimiento.

B) Liberación de conexiones

La liberación de una conexión N es iniciada, normalmente, por una de las entidades N+1 que la está usando. Sin embargo, esta ruptura puede ser también iniciada por una de las entidades N que la soportan.

Al contrario que ocurre en el establecimiento, la liberación de una conexión N-1 no implica la liberación de la conexión N. Esto es así para permitir que, si la conexión N-1 se ha roto por dificultades de la comunicación, pueda intentarse reestablecerla o sustituirla por otra.

La liberación de una conexión puede ser:

- Abrupta. Se libera de inmediato y los datos pendientes de envío se pierden.
- Suave o diferida. Antes de romper la conexión, se espera a no tener datos pendientes.

Adicionalmente, es importante tener presente que cuando la transferencia de información es entre redes, ésta será emitida por diferentes fuentes que deberán compartir el mismo medio de comunicación, para lo cual se requieren técnicas de multiplexión (véase figura 1.4). En donde podemos definir a la múltiplexión como la función que permite utilizar una sola conexión N-1 para soportar varias conexiones de nivel N. Todos los “tubos” de conexión N viajan por dentro del “tubo” de conexión N-1. Varias comunicaciones entre entidades pares de nivel N se realizan apoyándose en una sola conexión de nivel N-1. Es decir, las distintas entidades

usan un solo punto de acceso al servicio N-1. La función inversa realizada en el receptor se denomina demultiplexación.

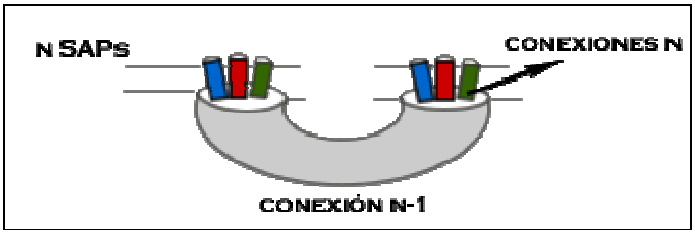


Figura 1.4. Multiplexión de conexiones

La división es la función que permite la utilización de más de una conexión N-1 por una sola conexión de nivel N. Con ello, el flujo de datos que soporta puede ser mayor (véase Figura 1.5). El flujo de datos del "tubo" correspondiente a la conexión N se reparte entre todos los "tubos" de conexiones N-1. En el extremo receptor, la función inversa se denomina recombinación y debe ser capaz de recuperar el orden en el que las PDUs (Unidades de Datos del Protocolo de Red) fueron generadas por el extremo emisor.

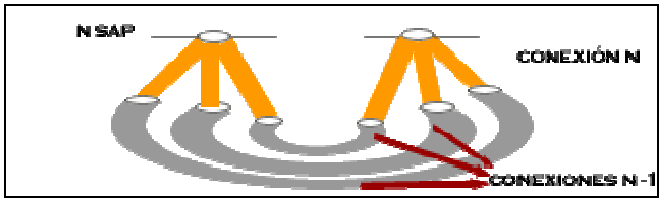


Figura 1.5. División de conexiones

1.3 Capas del Modelo OSI

Como se ha dicho en el punto anterior, el modelo de referencia OSI se divide en siete niveles o capas, para poder simplificar la implementación de la arquitectura necesaria.

Cada una de las capas del modelo OSI está dividida de acuerdo con ciertas características generales de las capas:

- Las capas poseen una estructura jerárquica.
- Cada capa desempeña funciones bien definidas.
- A cada nivel se le asigna un nombre que trata de describir las funciones que realiza.
- Los cambios en una capa son transparentes al resto de niveles.
- Los servicios proporcionados por cada nivel son utilizados por el nivel superior.
- Existe una comunicación virtual entre dos mismas capas, de manera horizontal.
- Existe una comunicación vertical entre una capa de nivel N y las capas de nivel N-1 y N+1
- La comunicación física se lleva a cabo entre las capas del nivel 1.

1.3.1 Capa Física

Esta capa proporciona los recursos eléctricos, mecánicos, procedimentales y funcionales para activar y mantener el enlace físico entre sistemas. Esta capa utiliza medios físicos, como los cables de par trenzado, coaxial y de fibra óptica.

Así mismo proporciona sus servicios a la capa de enlace de datos. Sus principales funciones son:

- Definición de características materiales (componentes y conectores mecánicos) y eléctricas (niveles de tensión, tipo de señal) que se van a utilizar en la transmisión de los datos.
- Definición de las características funcionales de la interfaz en cuanto a establecimiento, mantenimiento y liberación de enlace físico.
- Definición de reglas de procedimiento, es decir, la secuencia de eventos para transmitir.
- Transmisión de flujos de bits a través del medio.

- Manejo de voltajes y pulsos eléctricos para representar 1's o 0's.
- Especificación del medio físico de transmisión (coaxial, fibra óptica, par trenzado, etc.)
- Garantía de conexión física, pero no fiabilidad de la misma. Es decir, no se realiza ningún control de errores en este nivel. Eso corresponde al nivel superior.

1.3.2 Capa de Enlace de Datos

Puesto que la capa física sólo acepta y transmite una corriente de bits sin preocuparse por su significado o estructura, corresponde al nivel de enlace tomar el medio de transmisión en bruto y transformarlo en una línea que parezca libre de errores a los ojos de la capa de red.

A partir del canal ofrecido por el nivel físico, este segundo nivel hace que aquel parezca una línea de transmisión sin errores. Para esto, los bits transmitidos se dividen en cuadros que son confirmados por el receptor. Este nivel también es responsable del control del flujo para regular la velocidad relativa de los procesos.
Algunas de las funciones más importantes de la capa de enlace son:

- Establecimiento de medios necesarios para la comunicación fiable y eficiente entre dos máquinas en red.
- Estructuración de los datos en un formato predefinido, denominado trama, que suele ser de unos cientos de bytes, añadiendo una secuencia especial de bits al principio y al final de la misma.
- Sincronización en el envío de tramas.
- Detección y control de errores provenientes del medio físico mediante el uso de bits de paridad, CRC (Códigos Cíclicos Redundantes) y envío de acuses de recibo por parte del receptor que debe procesar el emisor.
- Utilización de número de secuencia en las tramas para evitar pérdidas y duplicidades.

- Utilización de la técnica de “piggybacking”, envío de acuses de recibo dentro de tramas de datos.
- Resolución de los problemas provocados por las tramas dañadas, perdidas o duplicadas.
- Control de la congestión de la red.
- Mecanismos de regulación de tráfico o control de flujo, para evitar que un transmisor veloz sature de datos a un receptor lento.
- Control del acceso al canal compartido en las redes de difusión.

Actividades que quedan contempladas a su vez por los dos subniveles que conforman la capa de enlace, los cuales son: el subnivel de control de enlace lógico (LLC) y el subnivel de control de acceso al medio de comunicación (MAC).

1.3.3 Capa de Red

Esta capa determina la mejor ruta de transferencia de los datos de un lugar a otro. El router funciona en esta capa, en la cual se utilizan esquemas de direccionamiento lógico que pueden ser manipulados por un administrador. Así mismo utiliza el esquema de direccionamiento del Protocolo Internet (IP) que corresponde a la pila de protocolos TCP/IP, o bien, otros esquemas de direccionamiento como AppleTalk, DECnet, VINES e IPX.

Es la responsable de la conmutación y enrutamiento de la información, y sus funciones se pueden resumir de la siguiente forma:

- Conocimiento de la topología de la red, es decir, de la forma en que están interconectados los nodos, con objeto de determinar la mejor ruta para la comunicación entre máquinas que pueden estar ubicadas en redes geográficamente distintas.
- División de los mensajes de la capa de transporte en unidades más complejas, llamadas NPDU también conocidas como paquetes, y asignación de direcciones lógicas a los mismos.

- Establecimiento, mantenimiento y liberación de las conexiones de red entre sistemas.
- Determinación del encaminamiento de los paquetes de la fuente al destino a través de dispositivos intermedios (routers).
    - Las rutas se pueden basar en tablas estáticas, previamente calculadas y configuradas.
    - Las rutas se pueden determinar al inicio de cada conversación.
    - Las rutas pueden ser dinámicas, determinándose con cada paquete en función de la carga de la red.
- Envío de paquetes de nodo a nodo usando un circuito virtual (orientado a la conexión) o datagramas (no orientado a la conexión).
    - Control de la congestión.
    - Control de flujo.
    - Control de errores.
    - Reencaminamiento de paquetes en caso de caída de un enlace.

1.3.4 Capa de Transporte

Esta capa segmenta y reensambla datos en un flujo de datos. La capa de transporte tiene el potencial de garantizar una conexión y de ofrecer un transporte fiable.

Es el responsable de la optimización de los recursos de la red, posiblemente utilizando multiplexación de canales y permitiendo la comunicación entre 2 procesos de distintas computadoras.

Su función más importante es la aceptación de datos de la capa de sesión, división en unidades más pequeñas, si es preciso, denominadas segmentos, y envío de esta información a la capa de red, asegurando que todos los datos lleguen correctamente al otro extremo de forma eficiente, donde son reensamblados.

Otras funcionalidades son:

- Cuando se inicia una conexión se determina una ruta de la fuente al destino, ruta que es usada para todo el tráfico de datos posterior.
- Determinación, en el momento del establecimiento de la sesión, del tipo de clase de servicio de transporte que se proporcionara a la capa de sesión:
  - Canal punto a punto libre de errores, que entrega los mensajes o bytes en el orden que se envían.
  - Mensajes aislados sin garantía respecto al orden de entrega.
  - Difusión de mensajes a múltiples destinos.
- El control de flujo entre nodos es distinto del control de flujo entre enrutadores, que tiene lugar en la capa de red.
- Detección y recuperación de errores de transporte.
- Control de congestión.
- Numeración de los segmentos para prevenir pérdidas y doble procesamiento de transmisiones.
- Garantía de recepción de todos los datos y en el orden adecuado, sin pérdidas ni duplicados.
- Asignación de una dirección única de transporte a cada usuario.
- Aislamiento a las capas superiores de los cambios inevitables de la tecnología del hardware.
- Contabilidad a través de la red.

En la cabecera que añade este nivel se envía la información que identifica a qué conexión pertenece cada mensaje.

1.3.5 Capa de Sesión

Esta capa proporciona sus servicios a la capa de presentación, facilitando el medio necesario para que las entidades de presentación de dos máquinas diferentes organicen y sincronicen su diálogo y procedan al intercambio de datos, mediante el establecimiento de sesiones.

Su función principal de la capa de sesión es el establecimiento, administración y finalización ordenada de sesiones entre dos máquinas. Una sesión permite el transporte ordinario de datos, como efectuar un login a un sistema remoto o transferir un archivo entre 2 nodos, además de proporcionar servicios mejorados, útiles en algunas aplicaciones, como los que se detallan a continuación:

- Manejo de control de diálogo (quién habla, cuándo, cuánto tiempo, half duplex o full duplex). Las sesiones pueden permitir que el tráfico vaya en una única dirección, comunicaciones bidireccionales alternadas (half duplex), o en ambas direcciones al mismo tiempo, comunicaciones bidireccionales simultaneas (full duplex). En las comunicaciones half duplex, la capa de sesión ayuda a llevar el control de los turnos, mediante el manejo de fichas, también llamadas testigos o token. Sólo el lado que posea la ficha puede efectuar la operación.
- Sincronización del diálogo, mediante la inserción de puntos de verificación en la corriente de datos, de modo que si se produce una interrupción sólo es necesario repetir la transferencia de los datos después del último punto de verificación. La decisión de dónde colocar los puntos de sincronización es competencia directa del nivel de aplicación. Los puntos de sincronización pueden ser de dos tipos.
    - Mayor. Necesita confirmación del otro extremo para seguir con la transferencia del siguiente bloque.
    - Menor. Se intercalan entre dos puntos de sincronización mayores.

El bloque entre el primero y el último punto de sincronización mayor se llama actividad. Cuando se establece una conexión de sesión, automáticamente se abre una actividad, para poder trabajar. Sólo un tipo de datos concreto puede enviarse fuera de una actividad, los datos de capacidades (CD), que son datos de control. Las actividades se dividen en unidades de diálogo, que es el contenido entre dos puntos de sincronización mayor consecutivos.

1.3.6 Capa de Presentación

Esta capa proporciona la representación de los datos y el formato de código, junto con la negociación de la sintaxis de la transferencia de datos. Asegura que los datos que llegan de la red pueden ser utilizados por la aplicación, y asegura también que la información enviada por la aplicación pueda ser transmitida a la red.

Esta capa ofrece a la capa de aplicación los servicios de:

- Garantía de que la información que envía la capa de aplicación de un sistema pueda ser entendida y utilizada por la capa de aplicación de otro sistema.
- Acuerdo y negociación de la sintaxis de transferencia en la fase de establecimiento de la conexión. La sintaxis elegida puede ser cambiada durante el tiempo que dure la conexión.
    - Definición del código a utilizar para representar una cadena de caracteres (ASCII, EBCDIC, etc.).
- Compresión de los datos, si es necesario.
- Aplicación de procesos criptográficos si así lo requiere. Es el nivel clave para el sistema de seguridad del sistema OSI.
- Formateo de la información para su visualización o impresión.

1.3.7 Capa de Aplicación

La capa del modelo OSI más cercana al usuario. Defiere de las demás capas en que no proporciona servicios a ninguna otra capa OSI, sino a aplicaciones que se encuentran fuera del modelo.

Las aplicaciones más importantes que hacen uso de esta capa, para que los procesos de las aplicaciones accedan al entorno OSI son, entre otras:

- Correo electrónico.
- Transferencia de archivos.
- Carga remota de trabajos.

- Servicios de directorio.
- Login remoto (rlogin, telnet).
- Acceso a bases de datos
- Sistemas operativos de red
- Aplicaciones Cliente servidor, etc.

1.4 Transmisión de datos en el modelo OSI

Una capa de una máquina no puede transferir los datos de forma directa a su capa par de otra máquina, si no que necesita los servicios de todas las capas que se encuentran por debajo de ella en la jerarquía de capas, pasándole la información hacia abajo hasta llegar al nivel físico, donde se transmiten a la máquina receptora (véase figura 1.6).

Cada capa utiliza el encapsulamiento para colocar la PDU (Unidad de Datos del Protocolo) de la capa superior en su campo de datos y agregar cualquier encabezado e información final que la capa necesite para realizar su función. De esta forma, a medida que los datos se desplazan hacia abajo a través de las capas del modelo OSI, el tamaño del mensaje va creciendo. A nivel 3, la PDU se llama **paquete** e incluye las direcciones lógicas origen y destino. A nivel 2, la **trama** incluye las direcciones físicas. Y, finalmente la capa física codifica los datos de la trama de enlace de datos en un patrón de unos y ceros para su transmisión a través del medio.

En la máquina receptora se realiza el proceso inverso, retirando los distintos encabezados, uno por uno, conforme el mensaje se propaga hacia arriba por las capas.

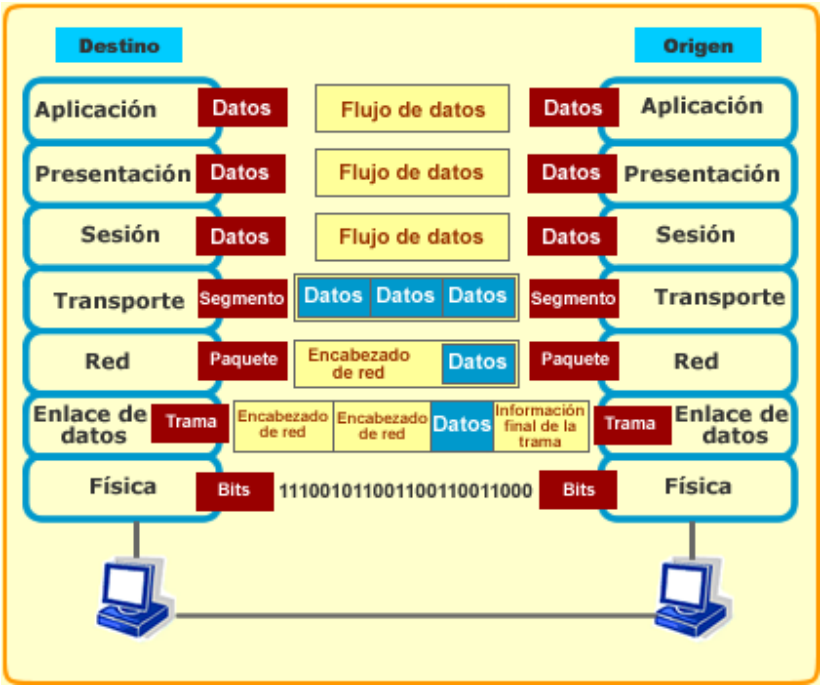


Figura 1.6. División de conexiones

Cada capa utiliza su propio protocolo de capa para comunicarse con su capa de igual del otro sistema. Cada protocolo de capa intercambia información llamada (PDU), entre capas iguales.

En la figura anterior, muestra un ejemplo de comunicación de tipo OSI. El Host A tiene información para enviar al Host B. El programa de aplicación del Host A se comunica con la capa de aplicación del Host A, y que se comunica a su vez con la capa de presentación del Host A, que a su vez se comunica con la capa de sesión del Host A, y así sucesivamente hasta que se alcanza la capa física del Host A. A la capa física coloca información (y la extrae) del medio de red físico. Una vez que la información recorre el medio de red físico y es seleccionada por el Host B, ésta asciende por las capas del Host B en orden inverso (primero la capa física, luego la capa de enlace de datos, etc.) Hasta que llega a la capa de aplicación del Host B.

Aunque cada una de las capas del Host A se comunica con sus capas adyacentes, cada una de las capas de un host tiene que llevar a cabo una tarea fundamental.

1.5 TCP/ IP

Las siglas TCP/IP, significan Transmisión Control Protocol/Internet Protocol (Protocolo de Control de Transmisión/Protocolo Internet) y es una serie basada principalmente en dos subprotocolos: el TCP, un protocolo OSI de capa 4, y el IP, un protocolo OSI de capa 3.La historia de los TCP/IP está unida al desarrollo de la ARPANET, que inicialmente estaba basada en un protocolo llamado Network Control Protocol (NCP) (Protocolo de Control de Red). El diseño original de ARPANET se basó en dos principios fundamentales: se supuso que la red física no era completamente confiable y que los protocolos de red no podían depender de ningún hardware o software patentado. La presunción de una red completamente no confiable podría parecer un poco extraña a primera vista. Sin embargo, la ARPANET era un proyecto del Departamento de Defensa y se aceptó la realidad de que la red física podría ser desorganizada por un evento catastrófico. Esto estimuló el desarrollo de los TCP/IP. El principio de no usar patentes, junto con el éxito de la temprana ARPANET, condujo a que el TCP/IP resultase disponible en una amplia variedad de plataformas.

Ayudando con el desarrollo del TCP/IP estaban Vint Cerf y Robert Kahn. A principios de los años 70, Cerf y Kahn, como parte de un programa de investigación de tecnología de redes ARPA, desarrollaron la idea de los Gateway y escribieron la primera especificación para los protocolos básicos TCP/IP usados ahora en Internet. La idea detrás del desarrollo de TCP/IP era permitir a diferentes redes de paquetes ser interconectadas de manera que las computadoras anfitrión no tuviesen que saber nada sobre las redes intermedias que las conectaban entre si. Hacia 1982, ARPA estableció TCP/IP como la serie de protocolos para ARPANET, y el Departamento de Defensa los declaró estándares para uso militar.

TCP/IP permitió la comunicación de datos a través de líneas análogas, radios en paquete, enlaces con satélite, redes Ethernet y otros.

1.5.1 Arquitectura

Para entender el funcionamiento de los protocolos TCP/IP debe tenerse en cuenta la arquitectura que ellos proponen para comunicar redes. Tal 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 (véase figura 1.7). 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 denominan compuertas, pudiendo recibir otros nombres como enrutadores o puentes.

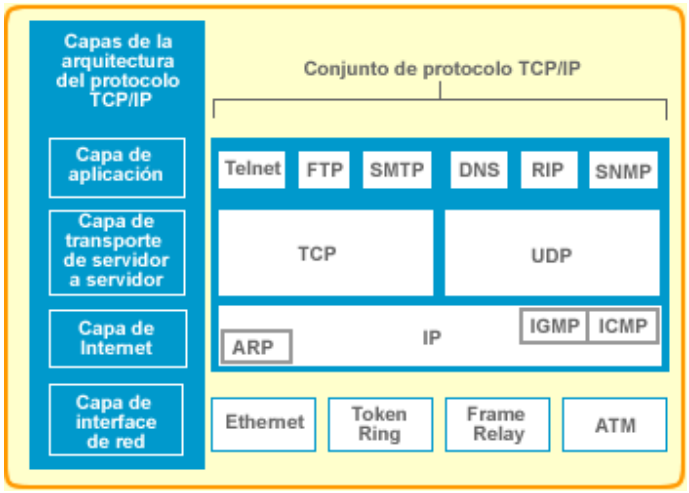


Figura 1.7. Arquitectura del modelo TCP/IP

TCP/IP es el protocolo común utilizado por todas las computadoras conectados a Internet, de manera que éstos puedan comunicarse entre sí. Hay que tener en cuenta que en Internet se encuentran conectadas computadoras de clases muy diferentes con hardware y software incompatibles en muchos casos, además de todos los medios y formas posibles de conexión. Aquí se encuentra una de las grandes ventajas del TCP/IP, pues este protocolo se encargará de que la

comunicación entre todos sea posible. TCP/IP es compatible con cualquier sistema operativo y con cualquier tipo de hardware.

La arquitectura del TCP/IP consta de cuatro niveles o capas en las que se agrupan los protocolos, y que se relacionan con los niveles OSI de la siguiente manera:

- Aplicación: Se corresponde con los niveles OSI de aplicación, presentación. Aquí se incluyen protocolos destinados a proporcionar servicios, tales como correo electrónico (SMTP), transferencia de ficheros (FTP), conexión remota (TELNET) y otros más recientes como el protocolo HTTP (Hypertext Transfer Protocol).
- Transporte: Coincide con el nivel de sesión y transporte del modelo OSI. Los protocolos de este nivel, tales como TCP y UDP, se encargan de manejar los datos y proporcionar la fiabilidad necesaria en el transporte de los mismos.
- Internet: Es el nivel de red del modelo OSI. Incluye al protocolo IP, que se encarga de enviar los paquetes de información a sus destinos correspondientes. Es utilizado con esta finalidad por los protocolos del nivel de transporte.
- Físico: Análogo a los niveles de enlace y físico del OSI. TCP/IP no específica ningún protocolo concreto, así es que corre por las interfaces conocidas, como por ejemplo: 802.2, CSMA/CD, X.25, etc.

1.6 Capas del Modelo TCP/IP

Puede pensarse en la mayor parte de las interredes, incluyendo Internet, como una arquitectura en capas, para simplificar la comprensión. Las capas muestran además, la manera como las diferentes partes del TCP/IP trabajan juntas.

1.6.1 Capa de Interface de Red

Sus funciones incluyen lograr una conexión al medio físico usando un protocolo específico para tener acceso al medio y segmentar los datos en bloques. Lleva a

cabo de manera efectiva todas las funciones de las primeras dos capas del modelo OSI:

- Conecta un anfitrión al hardware de red local
- Realiza una conexión con el medio físico
- Usa un protocolo específico para tener acceso al medio
- Coloca los datos en bloques
- Efectúa de manera efectiva todas las funciones de las dos primeras capas del modelo OSI.

1.6.2 Capa de Internet

Proporciona la funcionalidad para las comunicaciones entre redes a través de routers. Cada subred usa gateways para conectarse con las otras subredes en la interred. Además de transferir los datos de gateway a gateway hasta que alcanzan su destino y, luego, pasan a la capa de subred. La capa de interred ejecuta el Protocolo Internet (IP):

- El centro y fundamento es el Protocolo de Internet (IP).
- El usuario de transferencias envía mensajes del anfitrión fuente al anfitrión destino.
- Se trata de un servicio de datagramas sin conexión.
- La selección de la ruta se basa en alguna métrica.
- Usa direcciones Internet o IP como mapa de caminos para localizar un anfitrión dentro de la Internet.
- Depende de enrutadores o conmutadores (nodos dedicados de conectan dos o más redes disímiles).
- Parte integral es el Protocolo Internet de Control de Mensajes (ICMP), que usa un datagrama IP para llevar mensajes respecto al estado del ambiente de las comunicaciones.

1.6.3 Capa de transporte de servidor a servidor

Esta capa es la responsable de las comunicaciones globales de extremo a extremo de la red. Esta es la capa que ejecuta el (TCP) y otros protocolos como el UDP (Protocolo de Datagrama de usuario). Maneja el flujo de tráfico de datos en sí y asegura la confiabilidad para la transferencia de mensajes.

Se definen los dos protocolos de transporte:

- TCP (Tansmission Control Protocol, Protocolo de Control de Transmisión):
  - Se trata de un protocolo orientado a conexión.
  - Proporciona una transmisión confiable de datos mediante detección y corrección de datos de extremo a extremo.
  - Garantiza que los datos sean transferidos a través de una red de manera exacta y en el orden apropiado.
  - Retransmite cualesquiera datos no recibidos por el nodo destino.
  - Ofrece garantía contra duplicación de datos entre los nodos emisor y receptor.
  - Los protocolos de aplicación incluyen Telnet, FTP, SMTP y POP.
- UDP (User Datagram Protocol, Protocolo de Datagrama de Usuario):
  - No orientado a conexión.
  - Proporciona servicio de datagrama no confiable (ninguna detección o corrección de error extremo a extremo).
  - No retransmite ningún dato que no haya sido recibido.
  - Requiere poco sobrecarga.
  - Para aplicaciones cliente-servidor.

1.6.4 Capa de aplicación

Sirve como interfaz de comunicación para los usuarios proporcionando servicios de aplicación específicos. Esta capa ofrece interfaces para correo electrónico, transferencias de archivos remotos y acceso remoto. Por ejemplo:

- Terminal virtual (TELNET).
- Transferencia de archivos (FTP).

- Correo Electrónico (SMTP).
- Servicio de nombres (DNS).
- Web (HTTP).

1.7 Protocolo de Internet (IP)

El Protocolo de Internet tiene como tareas principales direccionar los datagramas de información entre computadoras y manejar el proceso de fragmentación de estos datagramas. Además es el responsable del enrutamiento de un datagrama, determinando a dónde será enviado y concibiendo rutas alternativas en caso de problemas.

IP tiene que ver con el envío no confiable de un datagrama. No confiable en el sentido del IP significa que el envío del datagrama no está garantizado, debido a que puede demorarse, enrutarse mal o mutilarse en la descomposición y reensamblaje de los fragmentos de mensaje. El IP no tiene nada que ver con el control o la confiabilidad del flujo: no tiene capacidad inherente para verificar que un mensaje enviado se reciba en forma correcta. El IP puede hacer una suposición de cuál es la mejor ruta para mover un datagrama al siguiente nodo a lo largo de una ruta, pero no verifica de manera inherente qué ruta elegida sea la más rápida o la más eficiente.

Parte del sistema IP define cómo manejan los gateways (equipos de compuerta de enlace) los datagramas, cómo y cuándo deben producir mensajes de error y cómo recuperarse de problemas que podrían surgir.

1.7.1 Principales características de IP

El protocolo IP es un ejemplo de servicio no orientado a conexión. Permite, sin establecimiento de llamada previo, el intercambio de datos entre dos computadoras (sin embargo, dos computadoras generalmente comparten un protocolo común de transporte orientado a conexión). Como IP no es orientado a conexión, se pueden perder datagramas entre las dos estaciones de usuario. Por ejemplo, las IP utilizan un tamaño máximo de cola, y si se sobrepasa, los buffers

se desbordarán. En esta situación se descartaran datagramas en la red. Por esta razón es importante un protocolo de transporte de nivel superior (como TCP) que soluciones esos problemas.

IP oculta la subred que hay debajo a los usuarios finales. Crea para ellos una red virtual. Este aspecto de IP es muy atractivo, ya que permite que diferentes redes se conecten a una pasarela IP.

Cabecera de IP

Una perspectiva muy útil en el análisis de IP consiste en examinar los campos del encabezado de IP (PDU) (ver figura 1.8).

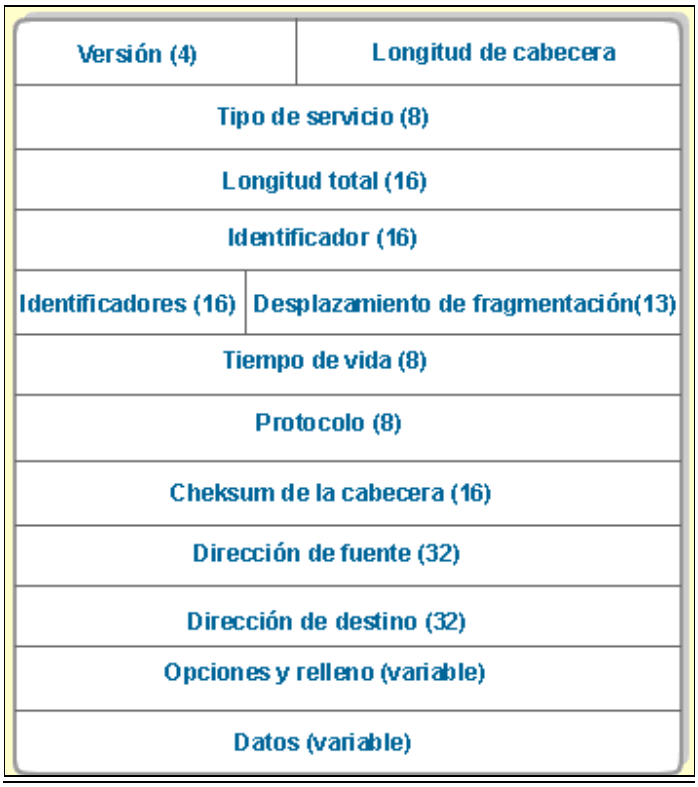


Figura 1.8. Cabecera IP

El campo de **versión** identifica la versión de IP en uso.

El campo de **longitud de cabecera** contiene cuatro bits con el valor de la longitud de la cabecera del datagrama.

EL campo de **tipo de servicio** se puede utilizar para el retardo de transito, el caudal efectivo, la precedencia y la fiabilidad.

El campo de **longitud** total especifica la longitud total del datagrama de IP. Se mide en octetos e incluye la longitud de la cabecera y de los datos.

El protocolo IP utiliza tres campos de datos en la cabecera que sirven para controlar la fragmentación y ensamblado del datagrama. Son el **identificador, los indicadores y el desplazamiento de fragmentación**. El campo de identificador se utiliza para identificar unívocamente todos los fragmentos de un datagrama original. Se utiliza junto con la dirección de fuente del computador receptor para identificar el fragmento. El campo de identificadores contiene bits que indican si el datagrama se puede fragmentar y si se puede fragmentar uno de los bits, se puede poner a 1 para indicar el último fragmento del datagrama original. El campo de desplazamiento de fragmentación contiene un valor que especifica la posición relativa del fragmento en el datagrama original. Su valor se inicializa a cero y se va poniendo al valor apropiado a medida que la pasarela fragmenta los datos. El valor se mide en valores de 8 octetos.

El **parámetro de tiempo de vida** (TTL) se utiliza para medir el tiempo que un datagrama lleva en la interred.

El campo TTL no sólo se utiliza para que la pasarela evite bucles sin fin, sino también para que las computadoras limiten el tiempo de vida de los segmentos que pasan por la interred. Si hay un computador que actúa como una pasarela, debe tratar los campos TTL con las reglas de las pasarelas. Hay que consultar al fabricante para saber si el computador descarta los datagramas utilizando el valor del campo TTL.

El campo de **protocolo** se utiliza para identificar el siguiente protocolo en la estructura de niveles por encima de IP que va a recibir el datagrama en el computador de destino. Los grupos de normalización han ideado un sistema de numeración que identifica a los protocolos de nivel superior más ampliamente

utilizados. Por ejemplo, el número 6 identifica a TCP, y el número 20 identifica al nivel de transporte de ISA.

El **checksum** de la cabecera se utiliza para detectar distorsiones en la cabecera. No se realizan comprobaciones en la cadena de datos de usuario. Algunos sectores críticos a IP indican que si se detectaran errores en los datos de usuario, las pasarelas podrían al menos notificar al computador remitente que hay problemas.

El datagrama de IP lleva dos direcciones. Se denominan **dirección de fuente y de destino** y no se modifican durante toda la vida del datagrama. Esos campos contienen las direcciones de IP.

El campo de **opciones** se emplea para identificar diversos servicios adicionales, como gestión de red y diagnósticos.

El campo de **relleno** se puede utilizar para asegurarse de que la cabecera del datagrama se alinea exactamente con una división de intervalo de 32 bits.

Finalmente el campo de **datos** contiene los datos de usuario. IP estipula que la combinación de los campos de cabecera y de datos no puede sobrepasar 65 535 octetos.

Adicionalmente es importante considerar los principales servicios que ofrece IP, los cuales son:

- Encaminamiento de fuente.
- Operaciones de encaminamiento.
- Métrica de distancia.
- Encaminamiento relajado y estricto.

1.7.2 Protocolo de Mensajes de Control Interred (ICMP)

El protocolo de Internet (IP) es un protocolo no orientado a la conexión, y, por tanto, no proporciona mecanismos de corrección ni de información de errores. Se basa en un módulo denominado Protocolo de Mensajes de Control Interred (ICMP) para:

- Informar de los errores ocurridos en el procesamiento de los datagramas.
- Proporcionar algunos mensajes de administración y de estatus.

ICMP reside en computadoras o pasarelas y acompaña a IP. Se utiliza entre computadoras y pasarelas por diversas razones, entre ellas:

- Cuando no se puedan enviar los datagramas.
- Cuando las pasarelas encaminan el tráfico por rutas más cortas.
- Cuando una pasarela no dispone de suficiente capacidad de almacenamiento (buffer) para retener y enviar unidades de datos de protocolo.

ICM notificara al computador si el destino no se puede alcanzar. Es también responsabilidad del ICMP gestionar o crear un mensaje de tiempo sobrepasado en el caso de que expire el periodo de vida de un datagrama. ICMP realiza también ciertas funciones de edición para determinar si la cabecera de IP es errónea o ininteligible.

1.8 Protocolo de Control de Transmisión (TCP)

El protocolo de Transmisión proporciona una cantidad considerable de servicios de la capa IP y las capas superiores. De mayor importancia, proporciona un protocolo orientado hacia la conexión para las capas superiores, que permite a una aplicación estar segura de que un datagrama enviado por la red fue recibido integro. En este papel, el TCP actúa como un protocolo de validación del mensaje que proporciona comunicaciones confiables. Si un datagrama se altera o pierde, por lo general el TCP maneja la retransmisión, en lugar de las aplicaciones de las capas superiores.

El TCP maneja el flujo de datagramas desde las capas superiores hasta la capa IP, así como los datagramas que llegan desde la capa IP hacia los protocolos de niveles superiores. El TCP tiene que asegurar que las prioridades y la seguridad se respeten de manera apropiada. El TCP debe ser capaz de manejar la terminación de una aplicación superior, que esta esperando la llegada de

datagramas, al igual que las fallas en las capas inferiores. El TCP también debe mantener una tabla de estado de todas las series de datos que entran y salen de la capa TCP.

El TCP reside en la capa de transporte, ubicada encima de IP pero debajo de las capas superiores y sus aplicaciones. El TCP reside sólo en dispositivos que en realidad procesan datagramas, asegurando que el datagrama ha ido desde la máquina fuente hasta la máquina destino. No reside en un dispositivo que tan solo enruta datagramas, de ahí que por lo general no haya una capa TCP en un gateway. Esto tiene sentido, debido a que en un gateway el datagrama no tiene necesidad de ir más arriba de la capa IP en el modelo en capas.

1.8.1 Principales características de TCP

TCP suministra una serie de servicios a los niveles superiores. Estos servicios son:

TCP es un **protocolo orientado a conexión**. Esto quiere decir que TCP mantiene información del estado de cada cadena de datos de usuario que circula por él. Así mismo debe asegurar que los datos se transmiten y se reciben correctamente por las computadoras atravesando las correspondientes redes.

El módulo TCP receptor utiliza una rutina de **cheksum** para comprobar la posible existencia de daños en los datos producidos en el proceso de transmisión. Si los datos son aceptables, TCP envía una aceptación positiva (ACK) al modulo TCP remitente.

Además de la capacidad de transmisión de cadenas, TCP soporta también el concepto de función **push**. Esta función se utiliza cuando una aplicación desea asegurarse de que todos los datos que han pasado al nivel inferior se han transmitido.

Además de utilizar los números de secuencia para las aceptaciones, TCP los utiliza para la **reordenación** de los segmentos que lleguen a su destino fuera de orden. Como TCP descansa sobre un protocolo no orientado a conexión, es bastante posible que la interred se creen datagramas duplicados.

El módulo TCP receptor se ocupa también de **controlar el flujo** de los datos del transmisor, lo que es muy útil para evitar el desbordamiento de los dispositivos de almacenamiento y la saturación de la máquina receptora.

TCP posee una facilidad muy útil que permite **multiplexar** varias sesiones de usuario en un mismo computador. Esta operación se realiza definiendo algunas convenciones para compartir puertos y sockets entre usuarios.

1.8.2 Conexión

Los puertos TCP pueden establecer dos tipos de conexiones. El modo de **apertura pasiva** permite que el protocolo de nivel superior (por ejemplo, un servidor) indique al TCP y al sistema operativo del computador que va a esperar la llegada de solicitudes de conexión procedentes del sistema remoto, en lugar de mandar una apertura activa. Tras recibir esta solicitud, es sistema operativo asigna un número de puerto a este extremo. Esta utilidad se puede usar para realizar comunicaciones con usuarios remotos sin tener el retardo de la apertura activa.

La segunda forma de establecimiento de conexión, es el modo de conexión activa. En esta situación, el protocolo de nivel superior designa específicamente otro socket por el que establece la conexión. Típicamente se envía la apertura activa a un puerto con apertura pasiva para establecer un **circuito virtual**.

TCP admite un escenario en el que se envían dos aperturas activas de un sistema a otro a la vez. TCP realizara la conexión. Esta característica permite que las aplicaciones envíen una apertura en cualquier momento, sin preocuparse de si la otra aplicación ha enviado otra apertura o no.

TCP establece convenciones estrictas sobre como se deben utilizar conjuntamente las aperturas activas y pasivas. En primer lugar, una apertura activa identifica un socket específico, así como sus niveles de prioridad y de seguridad. TCP garantiza una apertura si el socket remoto tiene una apertura pasiva compatible, o si ha enviado una apertura activa compatible.

El segmento TCP

La PDU que se intercambian entre dos módulos TCP se denominan segmentos (ver figura 1.9). El segmento se divide en dos partes, la parte de cabecera y la parte de datos. La parte de datos sigue a la parte de cabecera. Los primeros dos campos del segmento se denominan **puerto de fuente** y **puerto de destino**. Esos campos de 16 bits identifican a los programas de aplicación de nivel superior que utilizan la conexión TCP.

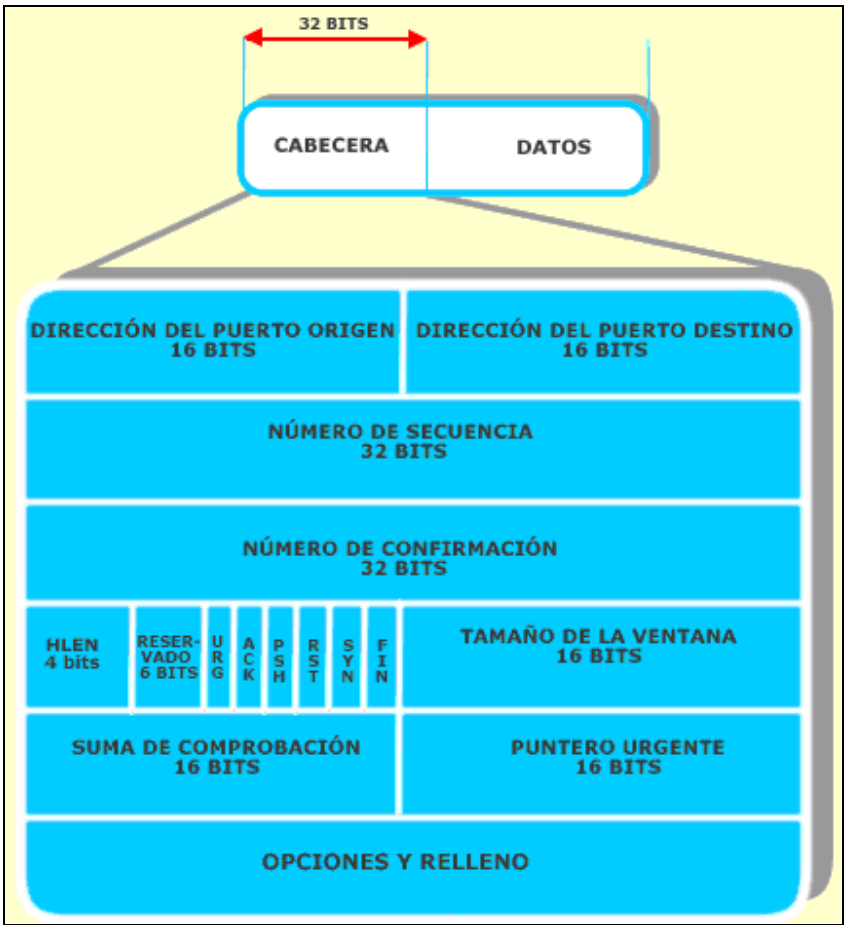


Figura 1.9. Segmento TCP

El siguiente campo se denomina **número de secuencia.** Este campo contiene el número de secuencia del primer octeto del campo de datos de usuario. Su valor

especifica la posición de la cadena de bits del modulo transmisor. Dentro del segmento especifica el primer octeto de datos de usuario.
El número de secuencia se utiliza también durante la operación de gestión de la conexión. Si dos entidades TCP utilizan el segmento de solicitud de conexión, entonces el número de secuencia especifica el número de **secuencia de envío inicial** (ISS) que se utilizará para la numeración subsiguiente de los datos de usuario.
El valor del **número de aceptación** permite aceptar los datos previamente recibidos. Este campo contiene el valor del número de secuencia del siguiente octeto que se espera recibir del transmisor. Con esa definición permite la aceptación inclusiva, en el sentido de que permite la aceptación de todos los octetos hasta, e incluyendo, el valor de este número menos 1.

El campo de **desplazamiento de datos** especifica el número de palabras alineadas de 32 bits de que consta la cabecera de TCP. Este campo se utiliza para determinar donde comienza el campo de datos.

Como puede esperarse, el campo **reservado** está reservado. Consta de 6 bits que deben valer cero. Estos bits están reservados para usos futuros.

Los seis bits siguientes se denominan **indicadores** (flags). Son bits de control de TCP y se utilizan para especificar ciertos servicios o utilidades que se pueden emplear durante la sesión. El valor de algunos de esos bits indica cómo interpretar otros campos de la cabecera. Los seis bits mencionados llevan la siguiente información:
URG, indica que el campo de puntero de urgencia es significativo.
ACK, indica si el campo de aceptación es significativo.
PSH, significa que el modulo va a utilizar la función push.
RST, indica que la conexión se va a inicializar.

SYN, indica que se van a sincronizar los números de secuencia; se utiliza en los segmentos de establecimiento de conexión como indicación de que se van a realizar algunas operaciones de preparación.

FIN, indica que el remitente no tiene más datos para enviar. Es comparable a la señal de fin de transmisión (EOT) en otros protocolos.

VENTANA, se pone a un valor que indica cuantos octetos desea aceptar el receptor. Este valor se establece teniendo en cuenta el valor del campo de aceptación (numero de aceptación).

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, se utiliza sólo si el indicador URG está a 1. El objeto de este puntero es identificar el octetos de datos al que datos urgentes. Los datos urgentes se denominan datos **fuera de banda.** TCP no dice qué hacer con los datos urgentes. El valor de este campo es un desplazamiento del número de secuencia y apunta al octeto a partir de cual siguen los datos urgentes.

OPCIONES, esta concebido para posibilitar futuras mejoras de TCP. Esta diseñado de forma semejante al campo de opción de los datagramas de IP, en el sentido de que cada opción se especifica mediante un byte que especifica el numero de opción, un campo que contiene la longitud de la opción y finalmente, los valores de la opción propiamente dichos.

RELLENO, asegura que la cabecera TCP ocupa un múltiplo de 32 bits. Finalmente, como muestra la figura siguen los datos de usuario.

1.8.3 Protocolo de Datagramas de Usuario (UDP)

Recordamos que los protocolos no orientados a conexión no proporcionan fiabilidad ni mecanismos de control de flujo. No proporcionan procedimientos de recuperación de errores. UDP es un protocolo no orientado a conexión.

UDP sirve como interfaz de aplicación simple para IP. Como no incluye mecanismos de fiabilidad, control de flujo ni medidas de recuperación de errores, sirve únicamente como multiplexor / demultiplexor del envió y recepción del trafico de IP.

El datagrama UDP contiene un número de puerto de destino y un número de puerto de fuente. El numero de destino es utilizado por el módulo de UDP para enviar el trafico al receptor adecuado (véase Figura 1.10).

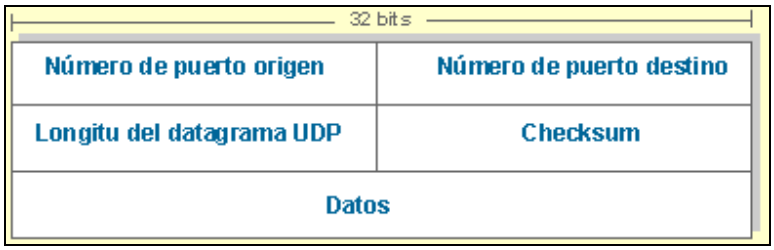


Figura 1.10. Formato del mensaje UDP

Puerto de Fuente: Este valor identifica el puerto del proceso de aplicación remitente. Este campo es opcional. Si no utiliza, se pone a 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 de 8 octetos.

Checksum: Este valor contiene el valor del comportamiento a 1 de 16 bits del comportamiento a 1 de la suma de la seudocabecera 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 un múltiplo de dos).

Se puede decir que UDP representa el nivel de de servicio mínimo que utilizan muchos sistemas de aplicación basados en transacciones. Es, sin embargo, muy útil en los casos en los que no son necesarios los servicios de TCP.

1.9 Comparación entre los modelos OSI y TCP/IP

Los modelos de referencia OSI y TCP/IP tienen mucho en común. Ambos se basan en el concepto de un gran número de protocolos independientes.

Es importante mencionar que vamos a comparar modelos de referencia, no pilas de protocolos correspondientes. En el modelo OSI, son tres conceptos fundamentales: Servicios, Interfaces, Protocolos. En este modelo se hace explícita esta distinción entre estos conceptos. Cada capa presta algunos servicios a la capa que se encuentra sobre ella. La definición de servicio dice lo que la capa hace, no cómo es que las entidades superiores tienen acceso a ella o cómo funciona la capa. La interfaz de una capa les dice a los procesos de arriba cómo acceder a ella; específica cuáles son los parámetros y qué resultados esperar; nada dice tampoco sobre cómo trabaja la capa por dentro.

Finalmente los protocolos pares que se usan en una capa son asunto de la capa. Ésta puede usar los protocolos que quiera, siempre que consiga que se realice el trabajo (esto es, que provea los servicios que ofrece). La capa también puede cambiar los protocolos a voluntad sin afectar el software de las capas superiores.

El modelo TCP/IP originalmente no distinguía en forma clara entre servicio, interfaz y protocolo, aunque se ha tratado de reajustarlo después a fin de hacerlo más parecido a OSI. Por ejemplo, los únicos servicios reales que ofrece la capa de interred son para enviar y recibir paquetes de IP.

El modelo de referencia OSI se desarrolló antes de que inventaran los protocolos. Este orden significa que el modelo no se orientó hacia un conjunto específico de protocolos.

Lo contrario sucedió con TCP/IP: primero llegaron los protocolos, y el modelo fue en calidad sólo una descripción de los protocolos existentes. El problema fue que el modelo no se ajustaba a ninguna otra pila de protocolos; en consecuencia no fue de mucha utilidad para describir otras redes que no fueran de tipo TCP/IP.

CAPÍTULO II

REDES DE ÁREA LOCAL

2.1 Introducción

En los últimos tres siglos, hemos podido ver el desarrollo de importantes tecnologías para la vida útil de la humanidad, por ejemplo en el siglo XVIII fue la era de los grandes sistemas mecánicos que acompañaron a la Revolución Industrial. El siglo XIX fue la invención de la máquina de vapor. Durante el siglo XX la tecnología clave fue la obtención, el procesamiento y la distribución de la información. Entre otros acontecimientos, vimos la instalación de redes mundiales de telefonía, la invención de la radio y la televisión, el nacimiento y crecimiento acelerado de la industria de la computación, así como el lanzamiento de satélites de comunicaciones.

La fusión de las computadoras y las comunicaciones ha tenido una influencia profunda en la manera en que están organizados los sistemas computacionales. El modelo antiguo de una sola computadora que realiza todas las tareas computacionales de una empresa ha sido reemplazado por otro en el que un gran número de computadoras separadas pero interconectadas hacen el trabajo. Estos sistemas se denominan redes de computadoras.

El desarrollo de las redes de área local (LAN) a mediados de la década de 1980 ayudó a cambiar nuestra forma de pensar de las computadoras a la forma en que nos comunicamos entre computadoras, y por qué.

En la actualidad muchas compañías tienen una cantidad considerable de computadoras. Por ejemplo, podría tener computadoras separadas para supervisar la producción, controlar inventarios y hacer la nómina. Al principio estas computadoras tal vez hayan trabajado por separado pero, en algún momento, la administración decidió conectarlas para extraer y correlacionar información acerca de toda la compañía. Dicho de otra manera, comparten recursos y el objetivo es hacer que todos los programas, el equipo y, en particular, los datos estén disponibles para todos los que se conecten a la red, independientemente de la ubicación física del recurso y del usuario.

2.2 Concepto de red

El objetivo de una red de datos consiste en facilitar la consecución de un incremento de la productividad vinculando todas las computadoras y redes de computadoras de manera que los usuarios pueden tener acceso a la información con independencia del tiempo, ubicación y tipo de equipo informático.

Las redes de datos han cambiado nuestra forma de ver nuestras empresas y empleados. Ya no es necesario mantener una ubicación común para todos los empleados si se quiere acceder a la información que estos necesitan para desarrollar su trabajo. Debido a esto, hay muchas organizaciones que han cambiado sus estrategias comerciales para utilizar estas redes de la forma en que llevan a cabo su actividad empresarial. La figura 2.1 muestra que la red está definida en función de agrupamientos de empleados (usuarios), siguiendo los siguientes criterios:

- La oficina principal es aquella donde todos están conectados a una LAN y donde está ubicada la mayoría de la información corporativa.
- Sucursales. Se trata de ubicaciones remotas donde trabajan grupos más reducidos de individuos. Estos usuarios se conectan entre sí por medio de una LAN. Para acceder a la oficina principal, los usuarios utilizan servicios de redes de área amplia (WAN).
- Teletrabajadores. Se trata de empleados que trabajan desde sus domicilios. Estos usuarios requieren, generalmente, conexiones puntuales (bajo demanda) con la oficina principal y la sucursal para acceder a los recursos de la red.
- Usuarios móviles. Se trata de individuos que trabajan desde distintas ubicaciones y dependen de distintos servicios para poder conectarse a la red. Cuando se encuentran fuera de la oficina, normalmente dependen de servicios de acceso telefónico para conectarse a la red corporativa.

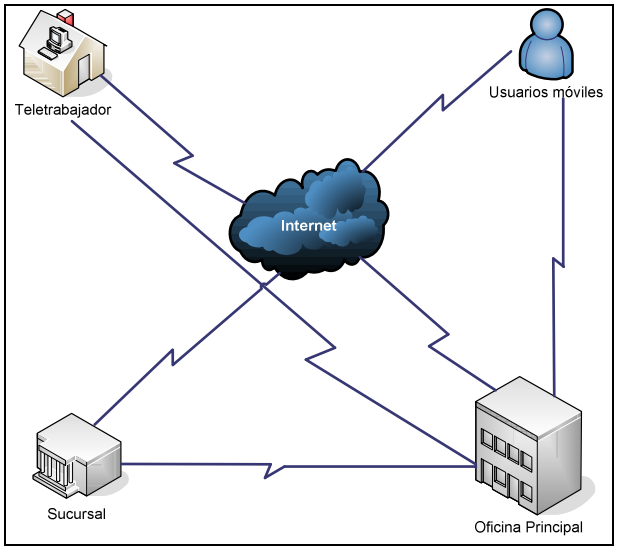


Figura 2.1. Formas de división de una red

También podemos definir una red de computadoras como un sistema de interconexión entre equipos que permite compartir recursos e información. Para ello es necesario contar con computadoras, tarjetas de red, cables de conexión, los dispositivos periféricos y el software correspondiente.

2.3 Clasificación de las redes

El concepto genérico de red incluye multitud de tipos diferentes de redes y posibles configuraciones de las mismas, por lo que desde un principio surgió la necesidad de establecer clasificaciones que permitieran identificar estructuras de red concretas.

Es por ello que existen multitud de redes, cada una de ellas con unas características especificas que las hacen diferentes del resto. Podemos clasificar a las redes en diferentes tipos, atendiendo a diferentes criterios. La clasificación se expone a continuación:

2.3.1 Titularidad de la red

Esta clasificación atiende a la propiedad de la red, por lo que se puede hacer una división en dos tipos de redes:

- Redes dedicadas: Una red dedicada es aquélla en la que sus líneas de comunicación son diseñadas e instaladas por el usuario o administrador, o bien, alquiladas a las compañías de comunicaciones que ofrecen este tipo de servicios (en caso de que sea necesario comunicar zonas geográficas alejadas), y siempre para su uso exclusivo.

- Redes compartidas: Las redes compartidas son aquéllas en las que las líneas de comunicación soportan información de diferentes usuarios. Se trata en todos los casos de redes de servicio público ofertadas por las compañías de telecomunicaciones bajo cuotas de alquiler en función de la utilización realizada o bajo tarifas por tiempo limitado.

2.3.2 Cobertura geográfica

La cobertura geográfica de la red es un factor a tener en cuenta a la hora de diseñar e instalar (véase figura 2.2). Tomando en cuenta la distancia entre estaciones o sucursales. La clasificación se describe a continuación:

Distancia entre procesadores	Procesadores ubicados en el mismo	
1 m	Metro cuadrado	Red de área personal
10 m	Cuarto	Red de área local
100 m	Edificio	
1 Km	Campus	
10 Km	Ciudad	Red de área metropolitana
100 Km	País	Red de área amplia
1,000 Km	Contiente	
10,000 Km	Planeta	Internet

Figura 2.2. Clasificación de procesadores interconectados por escala

Subred o segmento de red

Un segmento de red está formado por un conjunto de estaciones que comparten el mismo medio de transmisión. El segmento está limitado en espacio al departamento de una empresa, un aula de informática, etc. Se considera al segmento como la red de comunicación más pequeña, y todas las

redes de mayor tamaño están constituidas por la unión de varios segmentos de red.

Red de Área Local (LAN)

Como se mencionó, con la llegada de tecnología de red se obtuvieron muchos beneficios en cuanto a disposición de componentes y de información; de ahí partimos para dar una definición más formal "Una red local la podemos definir como la interconexión de varias computadoras y periféricos". Su extensión está limitada físicamente a un edificio o a un entorno de unos pocos kilómetros.

Son redes con velocidades entre 10 y 100 Mbps, tiene baja latencia y baja tasa de errores. Sin embargo las LAN más actuales funcionan hasta a 10 Gbps. Así mismo podemos decir en forma general que las LAN son diferentes a otros tipos de redes en tres aspectos: 1) Extensión, 2) Tecnología de transmisión, 3) Medio de Transmisión.

Entre algunas ventajas de las redes locales podemos destacar:

- Posibilidad de compartir periféricos costosos.
- El compartimiento de grandes cantidades de información a través de distintos programas, bases de datos, etc. De modo que sea más fácil su uso y utilización.
- Permite utilizar el correo electrónico para enviar o recibir mensajes de diferentes usuarios de la misma red e incluso de redes diferentes.

Cabe aclarar que a medida que va creciendo la red, el compartir dichos dispositivos y recursos pierden relevancia (véase figura 2.3).

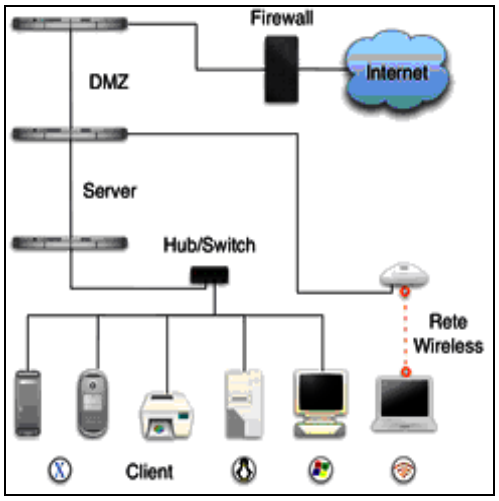


Figura 2.3. Arquitectura LAN

Existe no obstante una definición oficial, la del Comité IEEE 802, quien define una red local de la siguiente manera: Una red local es un sistema de comunicaciones que permite que un número de dispositivos independientes se comunique entre sí.

Red de Área Metropolitana (MAN)

Son redes que se extienden sobre un área geográfica extensa (véase figura 2.4). Contiene una colección de máquinas dedicadas a ejecutar los programas de usuarios (hosts). Estas LAN de host acceden a la subred de la WAN por un router.

Es una versión de mayor tamaño de la red local. Puede ser pública o privada. Una MAN puede soportar tanto voz como datos. No tiene elementos de intercambio de paquetes o conmutadores, lo cual simplifica bastante el diseño. La razón principal para distinguirla de otro tipo de redes, es que para las MAN se ha adoptado un estándar llamado DQDB (Distributed Queue Dual Bus, Bus Dual de Cola Distribuida) o IEEE 802.6. Utiliza medios de difusión al igual que las Redes de Área Local.

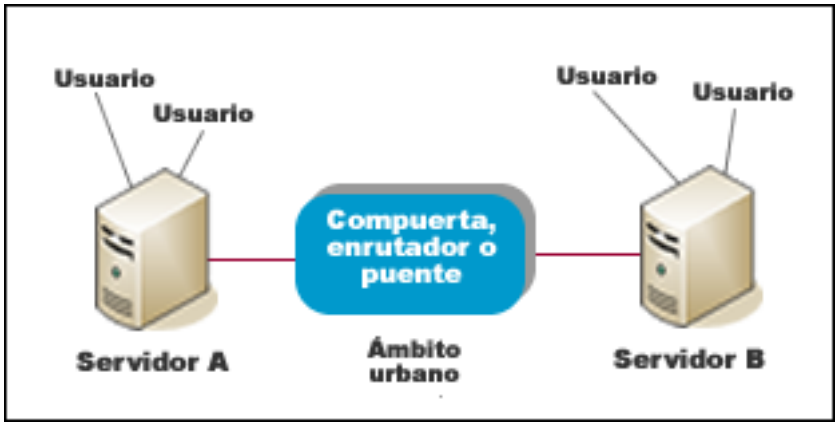


Figura 2.4. Red de Área Metropolitana (MAN).

Red de Área Extensa (WAN) y Redes Globales

Una WAN, abarca una gran área geográfica. Con frecuencia un país o un continente. Contiene un conjunto de computadoras (hosts) que están conectados por una subred de comunicación que, por lo general, las compañías telefónicas o los proveedores de servicios de Internet poseen y operan la subred. La función de una subred es llevar mensajes de un host a otro, como lo hace el sistema telefónico con las palabras del que habla al que escucha.

En la mayoría de las redes de área amplia la subred consta de dos componentes distintos: líneas de transmisión y elementos de conmutación. Las líneas de transmisión mueven bits entre máquinas. Los elementos de conmutación son computadoras especializadas que conectan tres o más líneas de transmisión. Cuando los datos llegan a una línea de entrada, el elemento de conmutación debe elegir una línea de salida en la cual reenviarlos. Estas computadoras de conmutación reciben varios nombres; conmutadores y enrutadores son los más comunes.

En este modelo, que se muestra en la figura 2.5, cada host está conectado frecuentemente a una LAN en la que existe un enrutador, aunque en algunos casos un host puede estar conectado de manera directa a un enrutador. El conjunto de líneas de comunicación y enrutadores forman la subred.

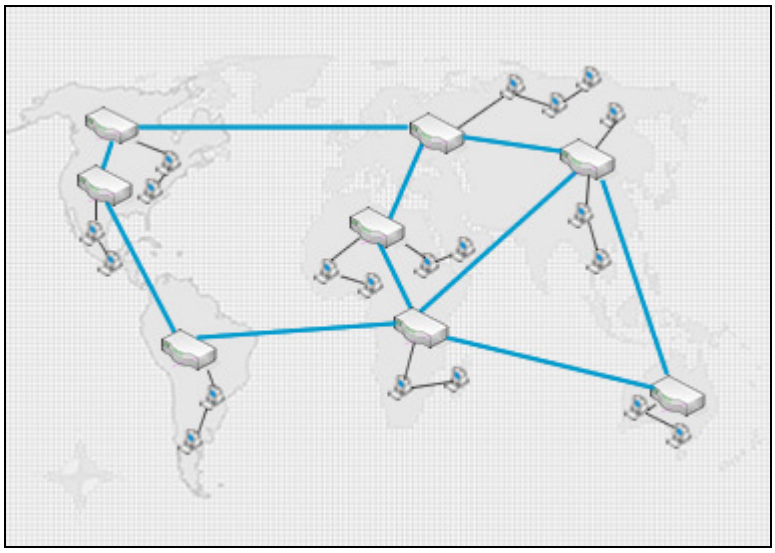


Figura 2.5. Relación entre hosts de LANs y la subred.

Cabe mencionar que cuando un paquete es recibido por un enrutador, se almacena ahí en su totalidad hasta que la línea de salida requerida esté libre y, por último se reenvía. Así mismo se divide el mensaje en paquetes, los cuales tienen un número de secuencia y se envían por la red de uno en uno en una rápida sucesión. Los paquetes se transportan de forma individual a través de la red y se depositan en el host receptor, donde se reensamblan en el mensaje original.

Redes inalámbricas

La comunicación inalámbrica digital no es una idea nueva. A principios de 1901, el físico italiano Guillermo Marconi demostró un telégrafo inalámbrico desde un barco a tierra utilizando el código Morse. Los sistemas inalámbricos digitales de la actualidad tienen un mejor desempeño, pero la idea básica es la misma.

Como primera aproximación, las redes inalámbricas se pueden dividir en tres categorías principales:

- Interconexión de sistemas. La cual se refiere a la interconexión de componentes de una computadora que utiliza radio de corto alcance (Bluetooth).

- LAN inalámbricas son sistemas en los que cada computadora tiene un módem de radio y una antena mediante los que se puede comunicar con otros sistemas (véase Figura 2.6).
- WAN inalámbricas, este tipo de red de radio utilizada para teléfonos celulares es un sistema inalámbrico de banda ancha baja. La primera generación era analógica y sólo para voz, la segunda era digital y solo para voz y la tercera generación es digital para voz y datos.

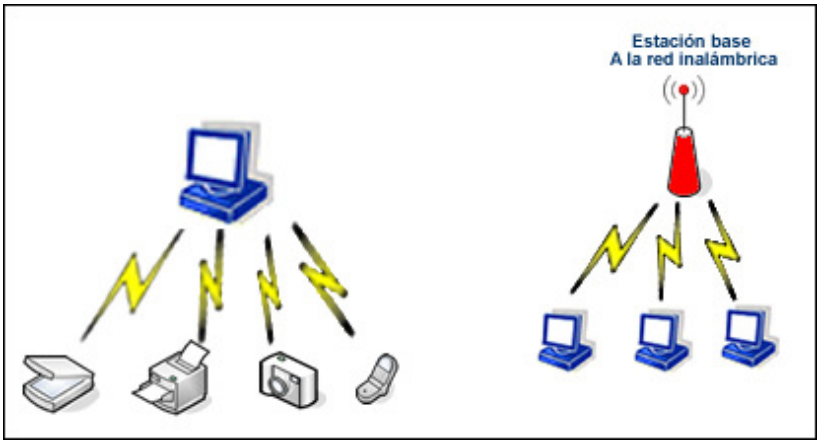


Figura 2.6. LAN inalámbrica

2.4 Modelos de Comunicación

Redes cliente servidor

En este modelo, los datos son almacenados en computadoras poderosas que se llaman servidores. Con frecuencia, estos se encuentran alojados en una central, en donde es administrado. En contraste, los empleados tienen en sus escritorios máquinas más sencillas, llamadas clientes, con las que pueden acceder a datos remotos.

Las máquinas cliente y servidor están conectadas por una red, como se ilustra en la figura 2.7.

Existen distintas formas de compartir los periféricos con otras computadoras. Estos pueden ser de varios tipos, entre ellos: servidor de archivos, servidor de impresión, servidor de comunicaciones, servidor de correo electrónico, servidor Web, servidor FTP, servidor Proxy, etc.

Existen sistemas operativos que pueden ejecutarse en una misma computadora o bien pueden estar distribuidos entre aquellos que forman la red.

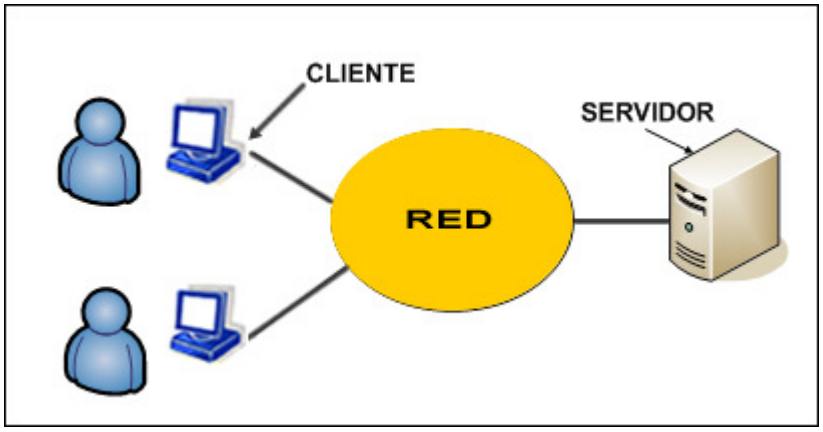


Figura 2.7. Una red con dos clientes y un servidor

Si vemos el modelo cliente-servidor en detalle, nos daremos cuenta de que hay dos procesos involucrados, uno en la máquina cliente y otro en la máquina servidor. La comunicación toma la siguiente forma: el proceso cliente envía una solicitud a través de la red al proceso servidor y espera una respuesta. Cuando el proceso servidor recibe la solicitud, realiza el trabajo que se le pide o busca los datos solicitados y devuelve una respuesta.

Redes point to point (Punto a Punto)

En una red punto a punto, los dispositivos en red actúan como socios iguales, o pares entre sí. Como pares, cada dispositivo puede tomar el rol de esclavo o la función de maestro.

Así mismo este modelo permite la comunicación entre usuarios (computadoras) directamente, sin tener que pasar por un equipo central para la transferencia, son aquellas redes en las que existen muchas conexiones entre parejas individuales de host o computadoras. Para poder transmitir la información desde un host hasta otro a veces es necesario que estos pasen por máquinas intermedias, siendo obligados en tales casos un trazado de rutas mediante dispositivos llamados enrutadores. Las redes punto a punto permiten a cualquier equipo de cómputo tanto solicitar como proporcionar servicios de red, obviamente el software está diseñado para que los host desempeñen las mismas o similares funciones que cualquier otro.

2.5 Modelos de transmisión de datos

Los sistemas de comunicaciones electrónicas pueden diseñarse para manejar la transmisión solamente en una dirección, en ambas direcciones pero sólo uno a la vez, o en ambas direcciones al mismo tiempo. Estos se llaman modos de transmisión y el determinar cuál es el más conveniente a utilizar depende de la aplicación en particular para la cual se requiera la comunicación, hay cuatro modos de transmisión posibles:

Redes de transmisión simple o simplex (SX)

Este tipo de redes, se identifican por su tipo de transmisión que es en un solo sentido. Los sistemas simplex son, algunas veces, llamados sistemas de un sentido o sólo para recibir, o sólo para transmitir. Una ubicación puede ser un transmisor o un receptor pero no ambos. Un ejemplo de la transmisión simplex es la radiodifusión de radio comercial o de televisión; la estación de radio siempre transmite y el usuario siempre recibe.

Redes Half-Duplex (HDX)

Las transmisiones HALF-DUPLEX, las transmisiones pueden ocurrir en ambas direcciones, pero no al mismo tiempo. A los sistemas half-duplex, algunas veces se les llaman sistemas con alternativa de dos sentidos, cualquier sentido o cambio y fuera. Una ubicación puede ser un transmisor o un receptor, pero no los dos al mismo tiempo. Los sistemas de radio de doble sentido como los radios de banda civil y de banda policíaca son ejemplos de este tipo de transmisión.

Redes Full - Duplex (FDX)

Con este sistema de operación, se pueden llevar acabo transmisiones en ambas direcciones al mismo tiempo. A los sistemas full-duplex algunas veces se les llama líneas simultáneas de doble sentido, duplex o de ambos sentidos. Una ubicación puede transmitir y recibir simultáneamente; sin embargo, la estación a la que está transmitiendo también debe ser la estación de la cual está recibiendo. Un sistema telefónico estándar es un ejemplo de una transmisión full-duplex.

Redes Full/Full-Duplex (F/FDX)

Con una operación full/full-duplex, es posible transmitir y recibir simultáneamente, pero no necesariamente entre las mismas dos ubicaciones (es decir, una estación puede transmitir a una segunda estación y recibir de una tercera estación al mismo tiempo). Las transmisiones full/full-duplex se utilizan casi exclusivamente con circuitos de comunicaciones de datos.

2.6 Topologías de red

Se denomina topología a la forma geométrica en que están distribuidas las estaciones de trabajo y los cables que las conectan.

Las estaciones de trabajo de una red se comunican entre sí mediante una conexión física, y el objeto de la topología es buscar la forma más económica y eficaz de conectarlas para, al mismo tiempo facilitar la fiabilidad del sistema, evitar los tiempos de espera en la transmisión de los datos, permitir un mejor control de la red y permitir de forma eficiente el aumento de las estaciones de trabajo.

Podemos distinguir dos aspectos diferentes a la hora de considerar una topología: la topología física y la topología lógica.

2.6.1 La topología física

En el diseño de redes, es importante contar con dispositivos como son máquinas, cableado (los medios) en la red y los dispositivos de conexión. Actualmente existen distintas formas de diseñar una red, a consecuencia de las diferentes necesidades que limitan la instalación de estas.

Topología irregular

En este tipo de topología no existe un patrón obvio de enlaces y nodos. El cableado no sigue un modelo determinado; de los nodos salen cantidades variables de cables. Las redes que se encuentran en las primeras etapas de construcción, o se encuentran mal planificadas, a menudo se conectan de esta manera.

Topología en estrella

En una topología en estrella, todos los dispositivos están conectados a una ubicación central común, normalmente un hub o un switch. Cuando un nodo envía datos a la ubicación central, el dispositivo central retransmite la información y la envía al destino (véase figura 2.8).

Debido a que el cableado está conectado a un dispositivo central, si un enlace falla, sólo fallara una parte de la red. El resto de la red no se verá afectada.

Sin embargo, si el dispositivo central falla, la totalidad de la red también fallará.

Una topología en estrella puede tener una máximo de 1024 nodos en una LAN y se puele usar en redes Ethernet 10 BaseT (IEEE 802.3) y 100 BaseTX (IEEE 802.12).

Entre las ventajas de la topología en estrella se incluyen la fiabilidad y la facilidad de mantenimiento e instalación. El control y la solución de problemas se puede mantener en el dispositivo central, haciendo que el mantenimiento sea más fácil. Las topologías de estrella permiten una mayor fiabilidad, ya que cada nodo está conectado al dispositivo central por un segmento. Si un segmento pierde la conexión, solo ese nodo perderá acceso a la red, por lo que el resto de la red no verá afectado. Dado que cada nodo está conectado al dispositivo central, las topologías en estrella permiten que halla una distribución más sencilla de la red, proporcionando al administrador una instalación más fácil que las demás topologías. Un inconveniente de esta topología es el coste.

Estando cada dispositivo conectado a una ubicación central, se necesita más cableado que en otras topologías. Además, también está el costo del dispositivo central.

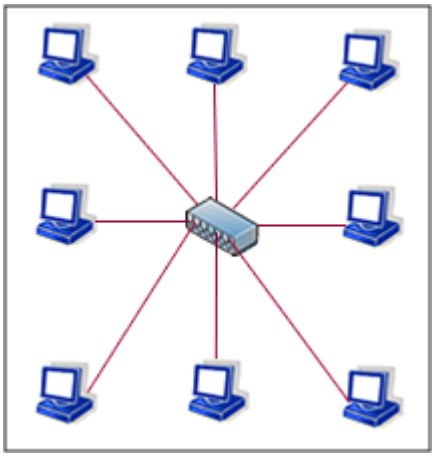


Figura 2.8. Topología en Estrella

Topología en bus

Una topología en bus, conecta múltiples dispositivos a un cable principal y, que a veces se denomina backbone, enlace troncal o segmento (véase figura 2.9). Es necesario conectar terminadores a cada extremo de la topología para absorber las señales que se reflejen. Si se usa cable coaxial sin terminadores, las señales que se reflejen se repetirán por la red, dejando la red inutilizable.

Las ventajas de una topología en bus son el costo y la facilidad de instalación. Dado que esta topología utiliza una distribución de cableado sencilla, cuesta menos y es más fácil de implementar que otras topologías.

Uno de los inconvenientes es que si un segmento de cable o de backbone se rompe o falla, la red también fallará. Otra desventaja es que sólo un nodo puede transmitir datos a la red al mismo tiempo. Si dos o más nodos tratan de enviar datos al mismo tiempo, se producirá una colisión; esto requerirá un procedimiento de recuperación, con lo que se alentará la red. Cuando se produce una colisión, todos los datos deben de ser reenviados. Un proceso llamado acceso múltiple con detección de carrier y detección de colisiones (CSMA/CD) impide que se produzca otra colisión. CSMA/CD es un proceso en virtud del cual cada nodo espera su turno para transmitir datos.

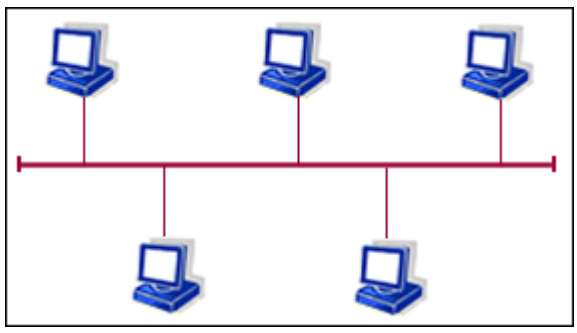


Figura 2.9. Topología de bus

Topología en malla

Normalmente utilizada en las WAN, una topología en malla conecta todos los dispositivos de la red y proporciona una ruta a cada dispositivo y desde este (véase figura 2.10). Una de las ventajas es que, como todos los dispositivos están conectados entre sí, la red tiene una tolerancia a fallos y una fiabilidad más altas. Si se interrumpe un segmento de cable en la red, los dispositivos

encontrarán la ruta más rápida para volver a enrutar el paquete a su destino. En consecuencia, los datos casi siempre llegan a su destino.

Las desventajas de esta topología son el coste y la dificultad de administración. Dado que existen numerosas conexiones con cada dispositivo, hay una cantidad mayor de requisitos de cableado, lo que hace que una topología en malla sea algo cara.

Si un segmento se interrumpe en la red, con el diseño complejo de la topología en malla, la localización del problema exacto puede resultar extremadamente difícil. Por tanto, el mantenimiento de la red puede ser muy complejo.

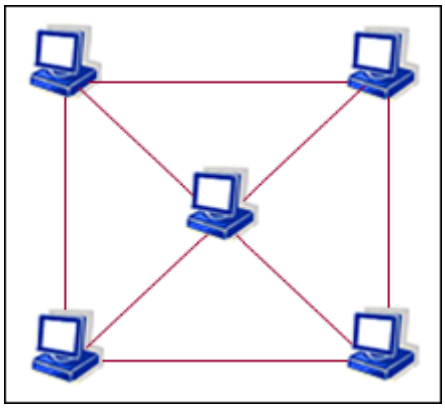


Figura 2.10. Topología en malla

Topología de anillo

En las topologías en anillo, cada dispositivo de la red está conectado con otros dos dispositivos (véase figura 2.11). El cable no tiene principio ni fin. Esta topología concreta forma un anillo. Los dispositivos de esta red utilizan un transceptor para comunicarse con sus vecinos. Los transceptores también actúan como repetidores con el fin de regenerar cada señal cuando pasa por el dispositivo.

La ventaja es un mejor rendimiento, ya que cada dispositivo recibe un turno para transmitir señales y tiene un acceso equivalente a la red. Una ventaja adicional es que la señal es regenerada por cada dispositivo que atraviesa, lo que impide que la señal se degrade.

Una de las desventajas de usar esta topología es que si falla uno de los dispositivos del cable, también fallará la totalidad de la red. A veces, la localización del fallo puede resultar difícil.

Otra desventaja es que si se realizan algunos cambios en la red, como la incorporación o el traslado de dispositivos, la interrupción hará que la red falle.

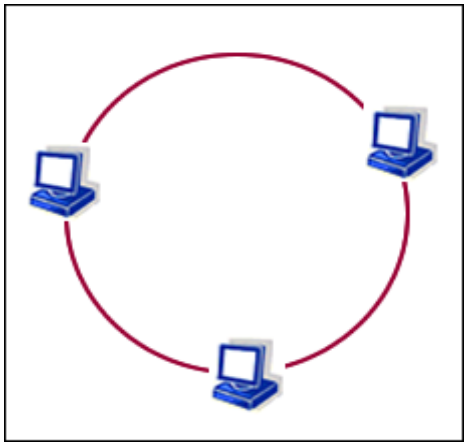


Figura 2.11. Topología de anillo

Topología de anillo doble

Este tipo de topología esta diseñada con dos anillos concéntricos, en donde cada host de la red está conectado a ambos anillos, aunque los dos anillos no están conectados directamente entre sí. Este tipo de topología incrementa la confiabilidad y flexibilidad de la red en comparación con la topología común de anillo.

Topología en estrella/bus

En esta configuración mixta, un multiplexor de señal ocupa el lugar del computador central de la configuración en estrella, estando determinadas estaciones de trabajo conectadas a él, y otras conectadas en bus junto con los multiplexores.

Esta red ofrece ventajas en edificios que cuentan con grupos de trabajo separados por grandes distancias. La utiliza la red ARCNET con control de flujo de paso de testigo.

Topología en árbol

Este tipo de topología es muy similar a la topología en estrella, salvo que no tiene un nodo central (véase figura 2.12). La forma en que se interconectan es

por medio de un enlace troncal, que es generalmente ocupado por un equipo concentrador, desde el que se ramifican los demás nodos.

El enlace troncal es un cable con varias capas de ramificaciones, y el flujo de información es jerárquico. Conectado en el otro extremo del enlace troncal generalmente se encuentra un host.

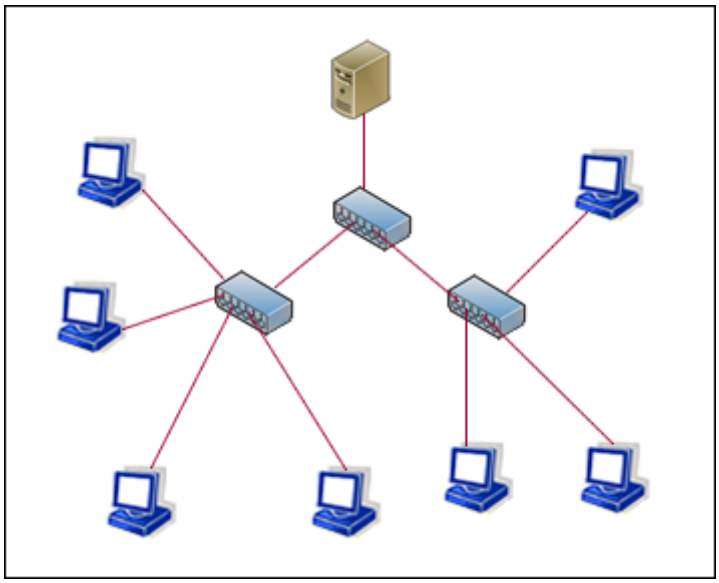


Figura 2.12. Topología en árbol

Topología de red celular

La topología celular esta compuesta por áreas circulares o hexagonales, cada una de las cuales tiene un nodo individual en el centro (véase figura 2.13).

La topología celular es un área geográfica dividida en regiones (celdas) para los fines de la tecnología inalámbrica. En esta tecnología no existen enlaces físicos; solo hay ondas electromagnéticas.

La gran ventaja de este tipo de redes es que no existe ningún medio tangible aparte de la atmósfera terrestre o el del vació del espacio exterior (y de los satélites). Las ventajas son que las señales se encuentran presentes en cualquier lugar de la celda y de ese modo pueden sufrir disturbios y violaciones de seguridad.

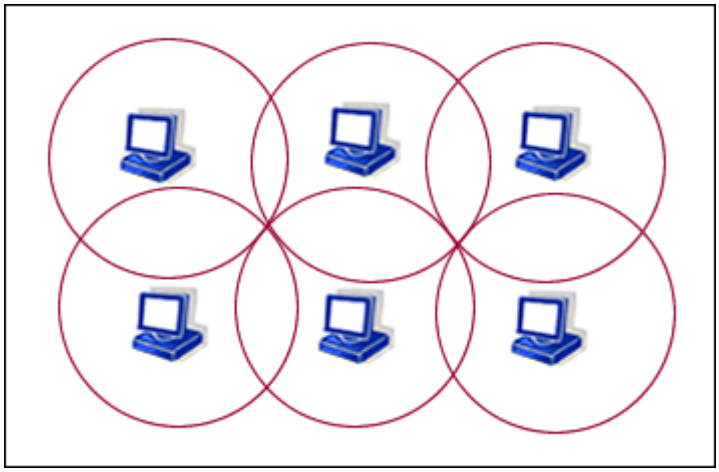


Figura 2.13. Topología celular

2.6.2 Topologías lógicas

Este tipo de topología describe la manera en que los datos son convertidos a un formato de trama específico y la manera en que los pulsos eléctricos son transmitidos a través del medio de comunicación, por lo que esta topología está relacionada con la capa física y la capa de enlace. Entre las más populares se encuentran la Ethernet, Token Ring y FDDI.

IEEE 802.3 (ETHERNET)

Ethernet es el nombre de una tecnología de redes de computadoras de área local (LAN) basada en tramas de datos. Esta tecnología fue estandarizada por la especificación IEEE 802.3, que define la forma en que los puestos de la red envían y reciben datos sobre un medio físico compartido que se comporta como un bus lógico, independientemente de su configuración física. Originalmente fue diseñada para enviar datos a 10 Mbps, aunque posteriormente ha sido perfeccionada para trabajar a 100 Mbps, 1 Gbps o 10 Gbps y se habla de versiones futuras de 40 Gbps y 100 Gbps. En sus versiones de hasta 1 Gbps utiliza el protocolo de acceso al medio CSMA/CD, actualmente Ethernet es el estándar más utilizado en redes locales/LAN.

Ethernet fue creado por Robert Metcalfe y otros en Xerox Parc, centro de investigación de Xerox para interconectar computadoras Alto. El diseño original funcionaba a 1 Mbps sobre cable coaxial grueso con conexiones vampiro (que

"muerden" el cable) en 10Base5. Para la norma de 10 Mbps se añadieron las conexiones en coaxial fino (10Base2, también de 50 ohmios, pero más flexible), con tramos conectados entre sí mediante conectores BNC; par trenzado categoría 3 (10BaseT) con conectores RJ45, mediante el empleo de hubs y con una configuración física en estrella; e incluso una conexión de fibra óptica (10BaseF).

Los estándares sucesivos (100 Mbps o Fast Ethernet, Gigabit Ethernet, y 10 Gigabit Ethernet) abandonaron los coaxiales dejando únicamente los cables de par trenzado sin apantallar (UTP - Unshielded Twisted Pair), de categorías 5 y superiores y la fibra óptica.

Ethernet es popular porque permite un buen equilibrio entre velocidad, costo y facilidad de instalación. Estos puntos fuertes, combinados con la amplia aceptación en el mercado y la habilidad de soportar virtualmente todos los protocolos de red populares, hacen a Ethernet la tecnología ideal para la red de la mayoría de usuarios de la informática actual.

Hardware utilizado en una Red Ethernet

Los elementos en una red Ethernet son los nodos de red y el medio de interconexión. Dichos nodos de red se pueden clasificar en dos grandes grupos: Equipo Terminal de Datos (DTE) y Equipo de Comunicación de Datos (DCE). Los DTE son los dispositivos que generan o son el destino de los datos, tales como las computadoras personales, las estaciones de trabajo, los servidores de archivos, los servidores de impresión, todos son parte del grupo de estaciones finales. Mientras que los DCE son los dispositivos de red intermediarios que reciben y retransmiten las tramas dentro de la red, y pueden ser ruteadores, conmutadores (switch), concentradores (hub), repetidores, o interfaces de comunicación, como un módem o una tarjeta de interfase por ejemplo.

Token Ring

Esta arquitectura de red fue creada por IBM en octubre de 1985, aunque anteriormente había comercializado dos tipos de redes locales: una red de

banda base a 375 kilobaudios y para un máximo de 64 computadores y una red de banda ancha a 2 megabaudios para un máximo de 72 computadores.
Emplea una topología de anillo con protocolo de paso de testigo y se puede utilizar cable de par trenzado, cable coaxial y fibra óptica.
La red Token Ring cumple el estándar IEEE 902.5. La conexión física de esta red es en forma de estrella, pero lógicamente se comporta y funciona como una red en anillo, siendo el protocolo de transmisión de paso de testigo de control. Esto quiere decir que los datos pasan de una estación a otra estación de forma secuencial (como si se tratara de una red en anillo), pero siempre pasan por el nodo central (como en una red en estrella).
Hay dos versiones diferentes de red Token Ring, una que funciona a la velocidad de 4 Mbps y otra a 16 Mbps. Puede soportar un máximo de 72 PC usando cable de par trenzado, o 260 PC usando cable coaxial de banda base. La distancia máxima es de 100 metros con cable de par trenzado y de 300 metros con cable coaxial. No obstante se pueden conectar unos anillos a otros con lo que el número de equipos puede ser, en principio, ilimitado. Mediante un adaptador se puede conectar a la línea de ordenadores Personal System/2.
Un token es pasado de computadora en computadora, y cuando una de ellas desea transmitir datos, debe esperar la llegada del token vacío, el cual tomará e introducirá los datos a transmitir, y enviará el token con los datos al destino. Una vez que la computadora destino recibe el token con los datos, lo envía de regreso a la computadora que lo envió con los datos, con el mensaje de que los datos fueron recibidos correctamente, y se libera el token, yendo nuevamente de computadora en computadora hasta que otra máquina desee transmitir, y así se repetirá el proceso.

Hardware utilizado en una red Token Ring

El hardware esta constituido por:

- Una tarjeta PC Network, la cual ha de instalarse en el ordenador que se va a conectar a la red. En esta tarjeta se realiza un intercambio de búfers de datos y bloques de control entre la memoria RAM de la estación de trabajo de la red y la tarjeta, cuya memoria RAM en una copia de la de la estación. Esto reduce el tiempo empleado en el

trasiego de información entre la tarjeta adaptadora y la estación de trabajo.

- La Unidad de Acceso a Múltiples Estaciones (MAU), es un concentrador de conexiones que permite conectar hasta ocho estaciones de trabajo. La MAU es un dispositivo pasivo que contiene circuitos diseñados para detectar la presencia o ausencia de señales de una estación de red. Si la MAU detecta un dispositivo defectuoso o un cable dañado, lo desvía para evitar la perdida de datos y hace que el testigo circule por la red.

Cada unidad de acceso a múltiples estaciones contiene 10 conectores, de los que ocho se utilizan para conectar estaciones de red y los dos restantes para conectar otras unidades de acceso.

Las redes Token Ring son seguras, fiables bajo condiciones de carga muy alta y bastante fáciles de instalar, pero tienen el inconveniente de que el costo total de éstas es considerablemente superior a las redes Ethernet y ARCNET.

FDDI

Tecnología con topología de anillo lógica y una topología física de anillo doble. FDDI (Fiber Distributed Data Interface; Interfaz de Datos Distribuidos por Fibra) es un estándar para transmisión de datos en LAN que opera sobre fibra óptica a velocidades de 100 Mbps, permitiendo extender el tamaño físico de la LAN en rangos de hasta 200 Km y soporta cientos de usuarios. Fue definido en los años 80 por la ANSI (America National Standards Institute; Instituto de Estándares Nacionales de América) ante la necesidad de contar con una tecnología para LAN de gran ancho de banda. Para alcanzar ese objetivo fue necesaria la adopción de la fibra óptica como medio físico, a pesar de que se elevaran demasiado los costos de instalación.

La topología de la red es de anillo, similar a la Token Ring ya que también utiliza el método de transmisión de paso de testigo. El cableado de la FDDI está constituido por dos anillos de fibras, uno transmitiendo en el sentido de las agujas del reloj y el otro en dirección contraria. El primero funciona como anillo principal y el segundo como respaldo de backup. El hecho de poseer dos anillos hace que la red FDDI sea altamente tolerante a fallas. El control de la

red es distribuido, razón por la cual si falla un nodo, el resto recompone la red automáticamente.

Si bien los costos de FDDI son altos, es muy utilizada como red de Backbone (red dorsal), ya que une las diferentes redes de un edificio o planta para conectar estaciones de alto desempeño, sin embargo la aparición de ATM (Asynchronous Transfer Mode; Modo de Transferencia Asíncrona) ha hecho que FDDI se considere la "hermana pequeña" de las redes de comunicación.

2.7 Componentes básicos de una red

Anteriormente se mencionaron aspectos de diseño de las topologías que van de acorde a nuestras necesidades y limitaciones. En este punto, vamos a describir los diferentes elementos de conexión que nos van a permitir implementar nuestra red.

Los componentes básicos necesarios en un sistema de comunicaciones o en una red son los siguientes:

- El emisor, en el cual se genera y del que parte la información.
- El codificador, el cual convierte los datos que se envían en un mensaje, es decir transforma la información para que se pueda enviar.
- El medio de transmisión, el cual proporciona la vía a través de la cual se va a enviar el mensaje. Es decir, el cableado.
- El decodificador, el cual convierte los datos recibidos dejándolos de forma que el receptor pueda entenderlos.
- El receptor, que es el destinatario de la información enviada, y que en definitiva es el que va a utilizarla.

Para que una red funcione correctamente esta ha de disponer de los componentes que se acaban de enumerar.

A continuación, se hará mención de los elementos más importantes que conforman una red:

- *Nodos*. Es un término que se emplea en el ámbito de los grandes ordenadores (mainframes) y que en realidad a lo que se refiere es al principio, al final o a la intersección de un enlace de comunicaciones, no a un dispositivo específico.

- *Las computadoras*, como ya se ha indicado anteriormente, pueden desarrollar dos funciones distintas: de servidores o de estaciones de trabajo.

- *Servidor*. Permite que todos los usuarios en red puedan transferir archivos desde y hacia las demás computadoras y, en caso de ser necesario, utilizar impresoras, escáner, dispositivos de copias de seguridad y cualquier otra máquina conectada a la red. Los recursos que dan servicio a los usuarios se encuentran concentrados en una máquina denominada servidor. Todos los dispositivos en la red (denominados clientes) están conectados al servidor que actúa como un punto central desde el que se gestiona la red. Las instalaciones y actualizaciones de software, la incorporación de nuevos dispositivos de red (por ejemplo, una nueva impresora) y demás tareas pueden realizarse a través del servidor.

2.7.1 Equipos que interconectan redes

Dos o más dispositivos conectados con el objetivo de compartir datos o recursos pueden formar una red. Una red de área local (LAN) puede necesitar cubrir más distancia de la que el medio de transmisión admite. O el número de estaciones puede ser demasiado grande para que la entrega de las tramas o la gestión de red se haga de forma eficiente. En el primer caso, un dispositivo denominado repetidor o regenerador se inserta en la red para incrementar la distancia a cubrir. En el segundo, un dispositivo denominado puente se inserta para gestionar el tráfico.

Cuando dos o más redes diferentes se conectan para intercambiar datos o recursos, se convierten en una red interconectada (o internet). Enlazar varias LAN en una internet requiere dispositivos de interconexión de redes adicionales denominados encamiandores (routers) y pasarelas (gateways). Estos dispositivos están diseñados para solucionar los obstáculos a la interconexión sin interrumpir las funciones independientes de las redes.

Como se mencionó anteriormente, los dispositivos de interconexión de redes y de red se dividen en cuatro categorías: repetidores, puentes, encaminadores y pasarelas (véase figura 2.14).

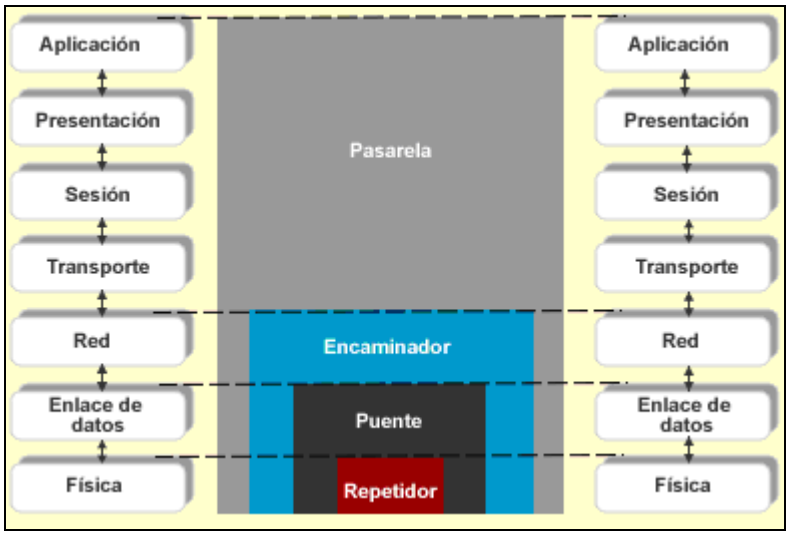


Figura 2.14. Dispositivos de conexión y el modelo OSI

Gatewatys o Pasarelas

Un gateway, que funciona como un traductor, proporciona comunicación entre distintos sistemas operativos y frecuentemente presta sus servicios en Internet. Debe existir un gateway si dos tipos de sistemas operativos, como Windows y Unix, se van a comunicar. Para comunicarse con un nodo de una red distinta en Internet, el dispositivo deberá estar conectado a una LAN o a una conexión de acceso telefónico. El término gateway también hace referencia a un gateway predeterminado, en el que se usa una dirección IP para reenviar paquetes de una subred a otra, siempre que no esté disponible otra información de enrutamiento.

Hubs

Los hubs, que funcionan en la capa física del modelo OSI, constituyen la ubicación central a la que se conecta el cableado en la mayoría de topologías. Hay tres tipos de hubs: pasivos, activos e inteligentes.

Un hub pasivo recibe información a través de uno de sus puertos y luego transmite los datos a través de otro puerto hasta una ubicación de destino.

Carece de potencia eléctrica y no dispone de ninguna capacidad de procesamiento de la señalización. Los hubs pasivos permiten que la comunicación fluya exclusivamente desde una ubicación a otra de la red, y absorben cierta energía de la señal, haciendo que la señal se debilite.

Un hub activo recibe datos a través de uno de sus puertos y luego funciona como un repetidor, regenerando y restableciendo la señal antes de enviarla al destino a través de otro puerto. Los hubs activos también se conocen como repetidores multipuerto.

La mayoría de hubs "comparten" ancho de banda entre los usuarios (más usuarios significa que haya menos ancho de banda por usuario).

Los hubs inteligentes tienen más electrónica que los hubs activos, y permiten la administración de la red (un hub "administrado") o, incluso, la conmutación (un hub "conmutado", o más comúnmente, un "switch").

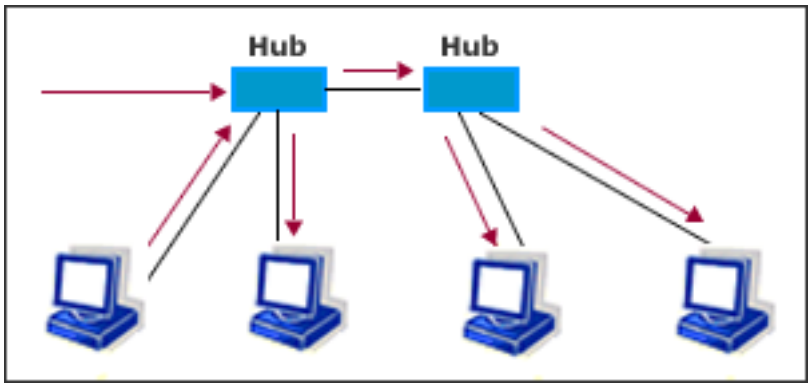


Figura 2.15. Modelo de operación de los hub

Repetidores

Cuando un repetidor recibe datos de un segmento ethernet, decodifica/codifica la información binaria y retransmite la señal al destino. Entre sus ventajas se incluye la capacidad de extender la red más lejos, la capacidad de incrementar el número de dispositivos conectados a la red, en aumento de la tolerancia a fallos, al aislar las interrupciones de una red a únicamente un segmento de cable en concreto, y la capacidad de enlazar distintos tipos de cable (véase figura 2.16). Una desventaja es que los repetidores intercambian dominios de colisión: si dos computadoras envían paquetes a la vez, se producirá una colisión y se aplicará CSMA /CD a la totalidad de la red, haciéndola lenta.

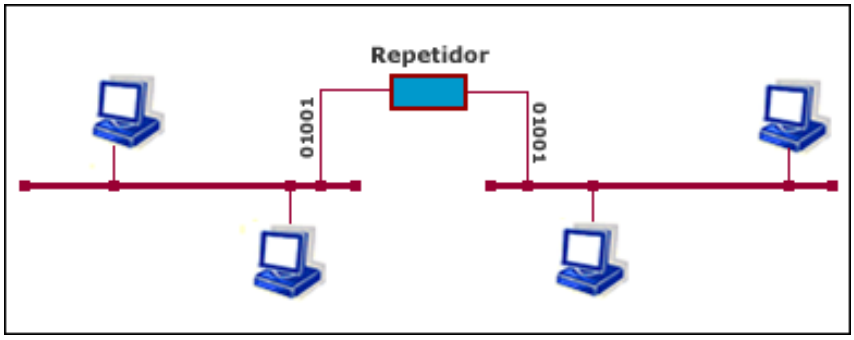


Figura 2.16. Modo de operación del repetidor

Tranceptores

Un tranceptor (transmisor/receptor) es un dispositivo que transmite y recibe datos de la red. Este dispositivo se conecta con la tarjeta de interfaz de red (NIC) de dos formas: como tranceptor incorporado o como transceptor externo.

Un tranceptor incorporado suele estar "en la propia tarjeta", o conectado a la tarjeta adaptadora, como los repetidores RJ-45 y los conectores BNC.

Un tranceptor externo realiza una conexión física con la NIC mediante un pequeño dispositivo, llamado interfaz de unidad de conexión (AUI) o conector Digital-Intel-Xerox (DIX), que está conectado a través de un cable extendido. Un tranceptor externo también puede conectar un extremo a una interfaz AUI y el otro a una interfaz RJ-45.

Router

Un router es un dispositivo de Capa 3 que proporciona la selección de la mejor ruta y la conmutación de paquetes de datos. Para conectar dos redes distintas, hay que usar un router (véase figura 2.17).

Los routers son los encargados de dirigir y traducir las direcciones lógicas a la dirección física de un paquete. Son dispositivos o bien estáticos o bien dinámicos que están normalmente conectados en una topología en malla con otros routers. Los routers configurados estáticamente no se pueden comunicar con otros routers; tienen una ruta fija determinada, que el administrador introduce manualmente. Los routers configurados dinámicamente tienen la capacidad de comunicarse con otros routers para determinar la mejor ruta para

enrutar un paquete con una serie de protocolos, entre los cuales se incluyen RIP, IGRP, EIGRP y OSPF.

Los routers se pueden usar para segmentar las LAN, creando dominios de colisión y dominios de difusión más pequeños. Pero el uso más importante de los routers es como dispositivos de backbone de las WAN. Las redes compuestas de routers, donde todos ellos se pueden comunicar por medio de protocolos de enrutamiento, pueden construirse con el fin de permitir la entrega de datos fiable y flexible. Hacen que Internet sea posible.

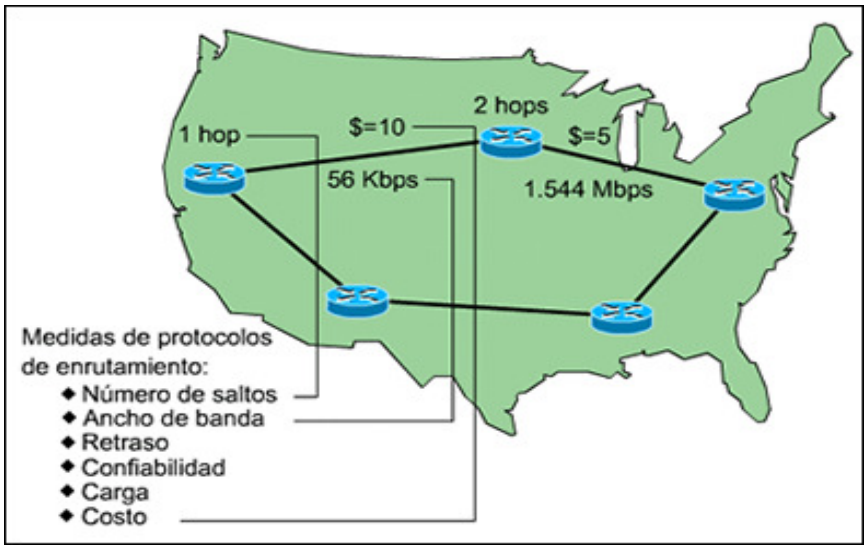


Figura 2.17. Ejemplo de operaciones de router

Brouter

Un brouter es una combinación de un router y un puente. Actúa como un router para los protocolos enrutables y como puente para los no enrutables.

Si un brouter recibe un paquete, deberá determinar la dirección IP. Si la dirección IP no está conectada a ninguno de sus puertos, deberá enrutar el paquete con una dirección IP conectada a uno de sus puertos, el brouter actuará como puente y entregará el paquete a su destino.

Permite que la red resuelva la práctica totalidad de los problemas de conexión utilizando un solo dispositivo; por tanto, es muy económico. No obstante, los brouters están cayendo en desuso, ya que sus funciones están siendo incorporadas en categorías de dispositivos distintas: los routers de Capa 3 y los switches de Capa 2.

NIC (Tarjeta de interfaz de red)

Las tarjetas de interfaz de red (NIC) permiten la comunicación entre una computadora y la red, proporcionando una conexión física (véase figura 2.18). Para que la computadora interactué con la NIC, ésta deberá tener instalados los controladores apropiados. A cada NIC se le asigna una dirección única, llamada dirección MAC. Esta dirección física que esta impresa en la NIC por el fabricante. Ninguna dirección MAC puede ser igual a otra.

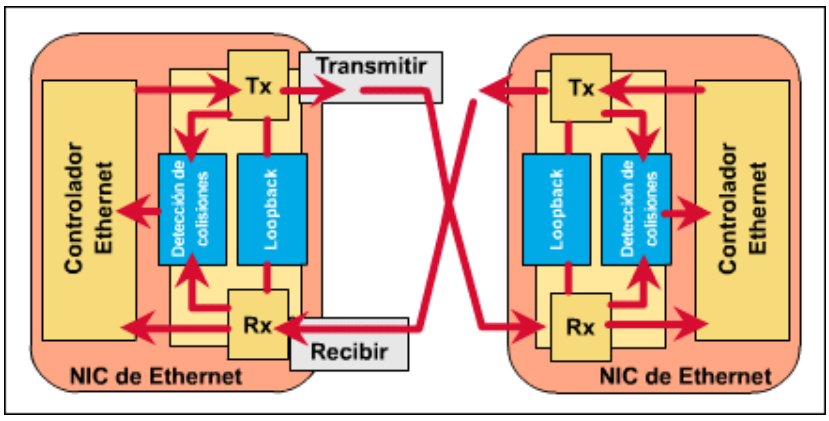


Figura 2.18. Diseño de ethernet semidúplex

2.8 Medios de transmisión

Para efectuar la transmisión de la información se utilizan varias técnicas, pero las más comunes son la banda base y la banda ancha. Donde el ancho de banda es básicamente la diferencia entre la frecuencia más alta y más baja de una determinada onda.

El término ancho de banda hace referencia al medio de transmisión. Cuanto mayor es el ancho de banda, más rápida es la transferencia de datos.

Banda base

La banda base es una técnica de señalización digital que utiliza la totalidad del ancho de banda de un medio para un solo canal a la vez; lo que permite capacidades muy altas (los anchos de banda reales calculados son posibles debido a está técnica de un solo canal a la vez).

Banda base significa que la señal no está modulada y por lo tanto está técnica no es muy adecuada para transmisiones a larga distancia ni para instalaciones

sometidas a un alto nivel de ruidos e interferencias. El empleo de esta técnica permite utilizar dispositivos de interfaz y repetidores que resultan muy económicos.

La técnica de banda base es especialmente adecuada en la transmisión a corta distancia. El medio de transmisión (el cable) ha de poder cambiar de estado con la rapidez que requiera la transmisión de los datos, los dispositivos de interfaz y los repetidores han de ser capaces de leer y transmitir información a esa velocidad.

Un canal que trabaje en modo de banda base se utiliza todo el ancho de banda, por lo que en un determinado momento, sólo puede transmitir una señal.

Banda ancha

La banda ancha es una técnica de señalización analógica que se suele utilizar en la televisión por cable. Este tipo de señalización puede transportar vídeo, voz y datos por un cable. Comparte el ancho de banda del medio por los distintos canales, a menudo utilizando distintas frecuencias de proveedor de servicios. Una de estas técnicas es la multiplexión por división de frecuencias (FDM), y el otro es por multiplexión por división de tiempo (TDM).

Esta técnica consiste básicamente en modular la información sobre ondas portadoras analógicas. Varias portadoras pueden compartir la capacidad del medio de transmisión mediante las técnicas de multiplexado ya se por división de frecuencia o de tiempo.

Aunque todos los usuarios utilizan la misma línea, es como si se estuviesen utilizando varias diferentes. El ancho de banda depende de la velocidad a la que se vayan a transmitir los datos.

Cuando se utiliza el sistema de banda ancha para transmitir datos, es necesario utilizar módems para modular la información. Los módems utilizados en las redes de banda ancha son dispositivos muy complejos, pues han de realizar funciones de modulación/demodulación y de transmisor/receptor.

La transmisión de banda ancha permite que dos o más canales de comunicación compartan el mismo ancho de banda de transmisión.

2.8.1 Tecnología de cables para redes

Par trenzado apantallado (STP)

El cable par trenzado apantallado (STP), al igual que el cable UTP tiene cuatro pares de cables, y cada uno de los cables de cada par ésta trenzado al otro (véase figura 2.19). Sin embargo, la diferencia es que STP ésta rodeado de una pantalla de papel de aluminio y de cobre trenzado en torno a los cables, que permite que haya más protección ante las interferencias electromagnéticas externas.

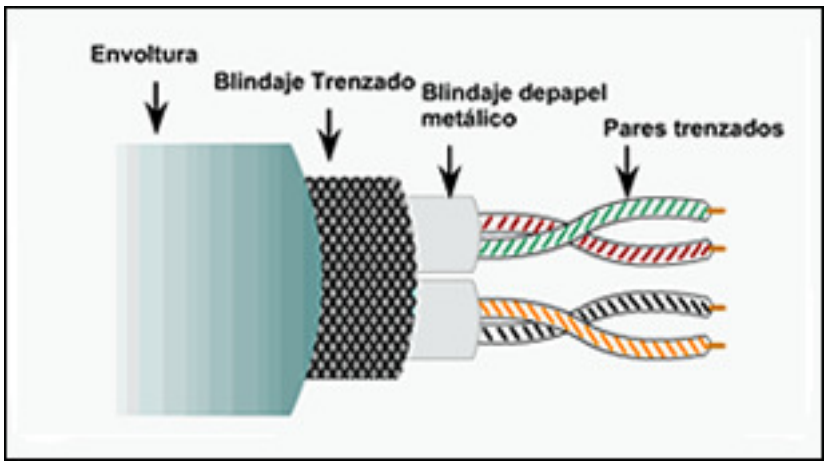


Figura 2.19. Cable par trenzado apantallado

Debido al apantallamiento, el cable es físicamente más grande, más difícil de instalar y terminar, y más caro que el cable UTP. En las aplicaciones en tornos eléctricamente ruidosos, STP utiliza conectores RJ-45, RJ-11, RS-232 y RS- 449.

Al igual que ocurre con UTP, STP también se presenta en grados de Cat 2, 3, 4 ó 5;sin embargo, sólo se recomienda Cat 5 o 6 para cualquier tipo de aplicación de datos. La longitud máxima del cable sin usar un dispositivo de regeneración de señal es de 100 metros, con una velocidad máxima de transferencia de 500 Mbps.

Cable par trenzado sin apantallar (UTP)

El cable de par trenzado sin apantallar (UTP) es un conjunto de tres o cuatro pares de cables, en el que cada uno de los cables de cada par está trenzado al

otro para impedir las interferencias electromagnéticas (véase figura 2.20). El cableado UTP emplea conectores RJ-45, RJ-11, RS-232 y RS-449. Dado que es más barato y fácil de instalar, UTP es más popular que el (STP). Un ejemplo de aplicación UTP son las redes telefónicas, que utilizan conectores RJ-11, y las redes 10BaseT, que utilizan conectores RJ-45. UTP se presenta en forma de grados de Cat 2, 3, 4, 5 y 6; sin embargo ahora se recomienda Cat 5e y 6 para cualquier tipo de aplicación de datos. La longitud máxima es de 100 metros sin usar ningún tipo de dispositivo de regeneración de señal, con una velocidad máxima de transferencia de datos de 1000 Mbps en Gigabit Ethernet.

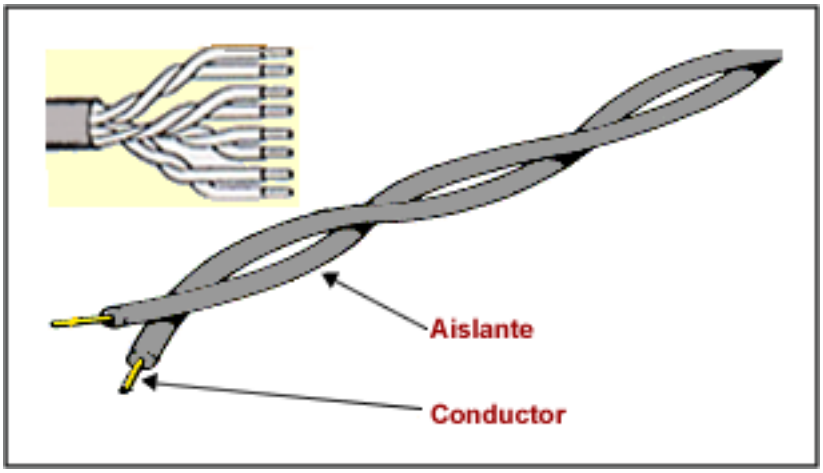


Figura 2.20. Cable UTP

Cable coaxial

El cable coaxial (o coax) transporta señales con rangos de frecuencia más altos que los cables de pares trenzados (véase figura 2.21), en parte debido a que ambos medios están construidos de forma bastante distinta. En lugar de tener dos hilos, el cable coaxial tiene un núcleo conductor central formado por un hilo sólido o enfilado (habitualmente cobre) recubierto por un aislante de material dieléctrico, que está a su vez, recubierto por una hoja exterior de metal conductor, malla o una combinación de ambas (también habitualmente de cobre). La cubierta metálica exterior sirve como blindaje contra el ruido y como un segundo conductor, lo que completa el circuito. Este conductor exterior está

también recubierto por un escudo aislante y todo el cable está protegido por una cubierta de plástico.

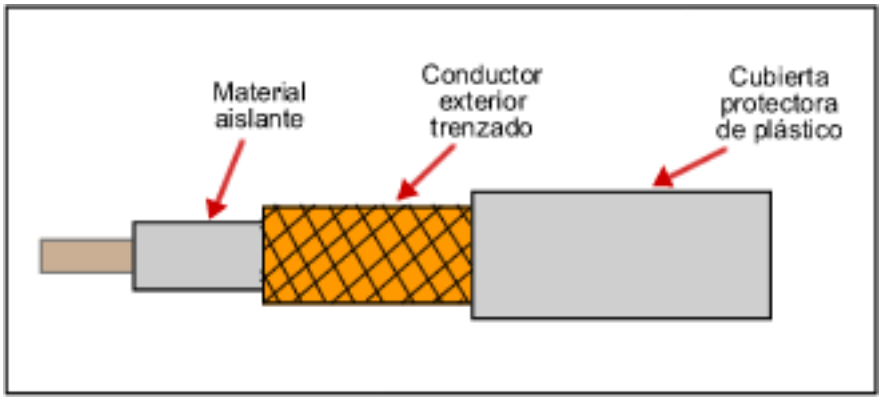


Figura 2.21. Cable coaxial

Fibra óptica

El cableado de fibra óptica transporta señales que han sido convertidas de forma eléctrica a forma óptica (pulsos de luz). Se compone del núcleo, que es, o bien un cilindro extremadamente fino de vidrio, o bien un plástico de calidad óptica que está rodeado de una segunda capa de vidrio o de plástico llamada envoltorio (véase figura 2.22). La interfaz que hay entre el núcleo y el envoltorio puede interceptar señales mediante un proceso llamado reflexión interna total (TIR), donde la fibra óptica actúa como un canal de luz.

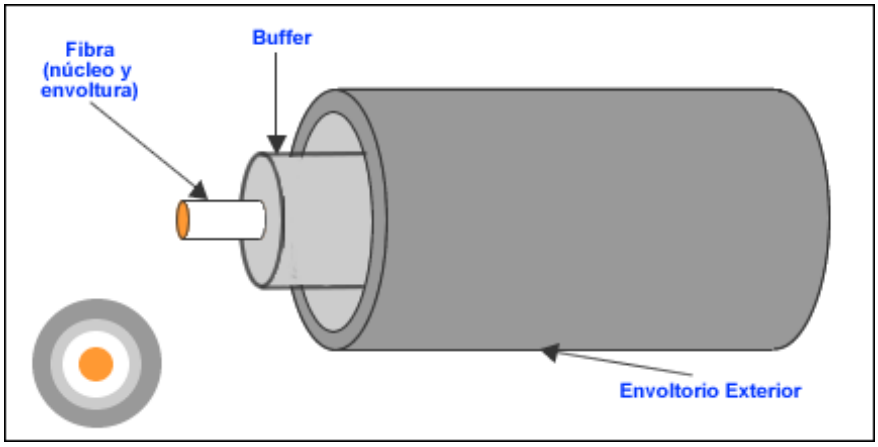


Figura 2.22. Fibra óptica

La tecnología actual de fibra óptica permite velocidades de transmisión de datos que van desde los 100 Mbps a los 2 Gbps. Existen actualmente 2 tipos:

- Fibra monomodo. El diámetro del núcleo o fibra óptica es extremadamente fino. Este tipo de fibra proporciona un alto rendimiento, pero hace que resulte difícil la conexión del cable a transmisores y otros dispositivos.

  El cable de fibra óptica monomodo es el que usan normalmente las compañías telefónicas y el que se usa en las instalaciones de datos como cable backbone. El cable monomodo no se usa como cable horizontal para conectar equipos a concentradores.

  En un cable monomodo la luz viaja en línea recta por la fibra y no va rebotando contra las paredes. La longitud de onda de la fibra monomodo es de 1210 y 1.550 nanómetros.
- Fibra multimodo. El cable multimodo permite utilizar más de un modo de luz para propagarse por el cable. Las longitudes de la luz utilizada en el cable multimodo están comprendidas entre 850 y 1300 nanómetros.

CAPÍTULO III

ENRUTAMIENTO
Y
DIRECCIONAMINETO

3.1 Introducción

Es de gran importancia enfocar en este capítulo, el estudio de cómo y en qué condiciones y con qué reglas viajan los paquetes desde un origen hasta un destino. Llegar al destino puede requerir muchos saltos por enrutadores intermedios. Esta función ciertamente contrasta con la de la capa de enlace de datos, que sólo tiene la meta de mover tramas de un extremo del cable a otro. Por lo tanto la capa de red es la capa más baja que maneja la transmisión de extremo a extremo.

Para lograr el objetivo de transmitir de extremo a extremo, la capa de red debe de conocer la topología de la subred de comunicación (es decir, el grupo de enrutadores) y elegir las rutas adecuadas a través de ella; también debe de tener cuidado al escoger las rutas para no sobrecargar algunas de las líneas de comunicación y los enrutadores y dejar inactivos a otros. Por último, cuando el origen y el destino están en redes diferentes, ocurren nuevos problemas. La capa de red es la encargada de solucionarlos.

Así mismo un punto muy importante a destacar es el estudio de los algoritmos de enrutamiento, que son aquella parte del software de la capa de red encargada de decidir la línea de salida por la que se transmitirán los paquetes, y deberá de ser capaz de manejar los cambios de topología y tráfico sin requerir el aborto de todas las actividades en todos los host y el reinicio de la red con cada caída de un enrutador.

3.2 Direccionamiento IP

Cada host TCP/IP está identificado por una dirección IP lógica. Esta dirección es única para cada host que se comunica mediante TCP/IP. Cada dirección IP de 32 bits identifica la ubicación de un sistema host en la red de la misma manera que una dirección identifica un domicilio en una ciudad.

Al igual que una dirección tiene un formato de dos partes estándar (el nombre de la calle y el número del domicilio), cada dirección IP está dividida internamente en dos partes, un Id. de red y un Id. de host:

- El Id. de red, también conocido como dirección de red, identifica un único segmento de red dentro de un conjunto de redes (una red de redes) TCP/IP más grande. Todos los sistemas que están conectados y comparten el acceso a la misma red tienen un Id. de red común en su dirección IP completa. Este Id. también se utiliza para identificar de forma exclusiva cada red en un conjunto de redes más grande.
- El Id. de host, también conocido como dirección de host, identifica un nodo TCP/IP (estación de trabajo, servidor, enrutador u otro dispositivo TCP/IP) dentro de cada red. El Id. de host de cada dispositivo identifica de forma exclusiva un único sistema en su propia red.

A continuación, se muestra un ejemplo de una dirección IP de 32 bits:

10000011 01101011 00010000 11001000

Para facilitar el direccionamiento IP, las direcciones IP se expresan en notación decimal con puntos. La dirección IP de 32 bits está segmentada en cuatro octetos de 8 bits. Estos octetos se convierten a formato decimal (sistema numérico de base 10) y se separan con puntos. Por tanto, la dirección IP del ejemplo anterior es 131.107.16.200 cuando se convierte a la notación decimal con puntos.

La figura 3.1 muestra un ejemplo de dirección IP (131.107.16.200) tal como está dividida en las secciones de Id de red y host. La parte de Id de red (131.107) está indicada por los dos primeros números de la dirección IP. La

parte de Id de host (16.200) está indicada por los dos últimos números de la dirección IP.

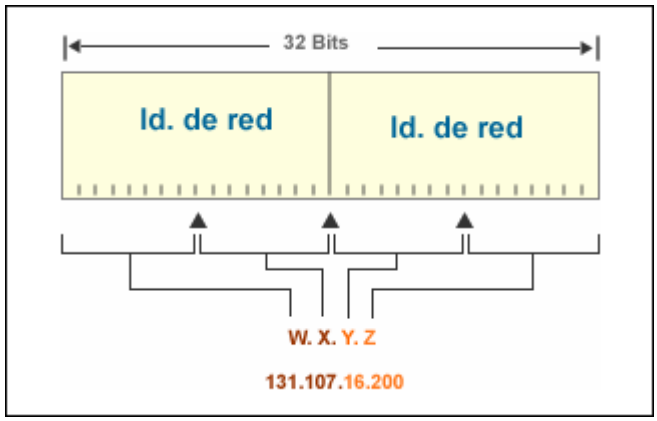


Figura 3.1. División de una dirección IP

Puesto que las direcciones IP identifican los dispositivos de una red, debe asignarse una dirección IP única a cada dispositivo de la red.

En general, la mayor parte de los equipos tienen únicamente un adaptador de red instalado y, por tanto, necesitan sólo una dirección IP. Si un equipo tiene varios adaptadores de red instalados, cada uno necesita su propia dirección IP.

3.2.1 Clases de direcciones IP

La comunidad de Internet ha definido cinco clases de direcciones. Las direcciones de las clases A, B y C se utilizan para la asignación a nodos TCP/IP.

La clase de dirección define los bits que se utilizan para las partes de Id de red e Id de host de cada dirección. La clase de dirección también define el número de redes y hosts que se pueden admitir por cada red.

La figura 3.2 designa los valores de cualquier dirección IP dada. La tabla siguiente sirve para mostrar:

- Cómo el valor del primer octeto de una dirección IP dada indica la clase de dirección.
- Cómo están divididos los octetos de una dirección en el Id de red y el Id de host.

- El número de redes y hosts posibles por cada red que hay disponibles para cada clase.

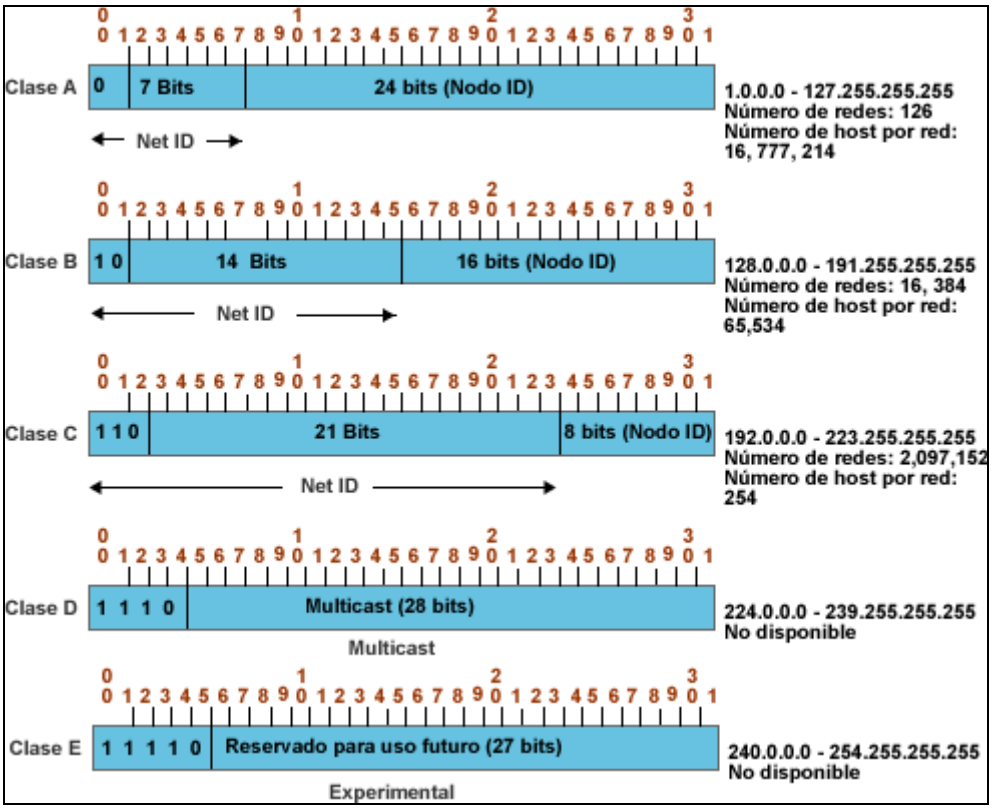


Figura 3.2. Designación de valores en octetos

3.2.2 Máscaras de subred

Los Id. de red y de host en una dirección IP se distinguen mediante una máscara de subred. Cada máscara de subred es un número de 32 bits que utiliza grupos de bits consecutivos de todo unos (1) para identificar la parte de Id. de red y todo ceros (0) para identificar la parte de Id. de host en una dirección IP.

Por ejemplo, la máscara de subred que se utiliza normalmente con la dirección IP 131.107.16.200 es el siguiente número binario de 32 bits:

11111111 11111111 00000000 00000000

Este número de máscara de subred está formado por 16 bits uno seguidos de 16 bits cero, lo que indica que las secciones de Id. de red e Id. de host de esta dirección IP tienen una longitud de 16 bits. Normalmente, esta máscara de subred se muestra en notación decimal con puntos como 255.255.0.0.

La tabla 3.1 muestra las máscaras de subred para las clases de direcciones Internet.

Clase de dirección	Bits para la máscara de subred	Máscara de subred
Clase A	11111111 00000000 00000000 00000000	255.0.0.0
Clase B	11111111 11111111 00000000 00000000	255.255.0.0
Clase C	11111111 11111111 11111111 00000000	255.255.255.0

Tabla 3.1. División de subredes por clase

Normalmente, los valores predeterminados de máscara de subred (como se muestra en la tabla anterior) son aceptables para la mayor parte de las redes sin requisitos especiales en las que cada segmento de red IP corresponde a una única red física.

En algunos casos, puede utilizar máscaras de subred personalizadas para implementar la creación de subredes IP. Con la creación de subredes IP, se puede subdividir la parte de Id. de host predeterminada en una dirección IP para especificar subredes, que son subdivisiones del Id. de red basado en la clase original.

Al personalizar la longitud de la máscara de subred, puede reducir el número de bits que se utilizan para el Id. de host actual.

Cabe aclarar en forma general que al introducir subredes, se cambian las tablas de enrutamiento, agregando entradas con forma de (red, subred,0) y (red, subred, host). Por lo tanto, un enrutador de la subred k sabe cómo llegar a todas las demás subredes y a todos los host de la subred k; no tiene que saber los detalles sobre los host de otras subredes. De echo, todo lo que se necesita es hacer que cada enrutador haga un AND booleano con la máscara de subred de la red para deshacerse del número de host y buscar la dirección resultante en sus tablas (tras determinar de qué clase de red se trata).

Por tanto, la división de redes reduce espacio en la tabla de enrutamiento creando una jerarquía de tres niveles, que consiste en red, subred y host.

3.2.3 NAT (Traducción de dirección de red)

La traducción de dirección de red, aportó una solución a los problemas de escasez de direcciones IP debido a las pocas direcciones con las que pudiera disponer un ISP. Como principal objetivo, asignar una sola dirección IP a cada compañía. Dentro de la compañía, cada computadora tiene una dirección IP única que se usa para enrutar tráfico interno. Sin embargo, cuando un paquete sale de la compañía y va al ISP, se presenta la traducción de la dirección (véase figura 3.3). Para hacer posible este esquema los tres rangos de direcciones IP se han declarado como privados. Las compañías pueden usarlos internamente cuando los deseen. La única regla es que ningún paquete que contiene estas direcciones puede aparecer en la propia Internet. Los tres rangos reservados son:

10.0.0.0	10.255.255.255/8	(16,777,216 host)
172.16.0.0	172.31.255.255/12	(1,048,576 host)
192.168.0.0	192.168.255.255/16	(65,536 host)

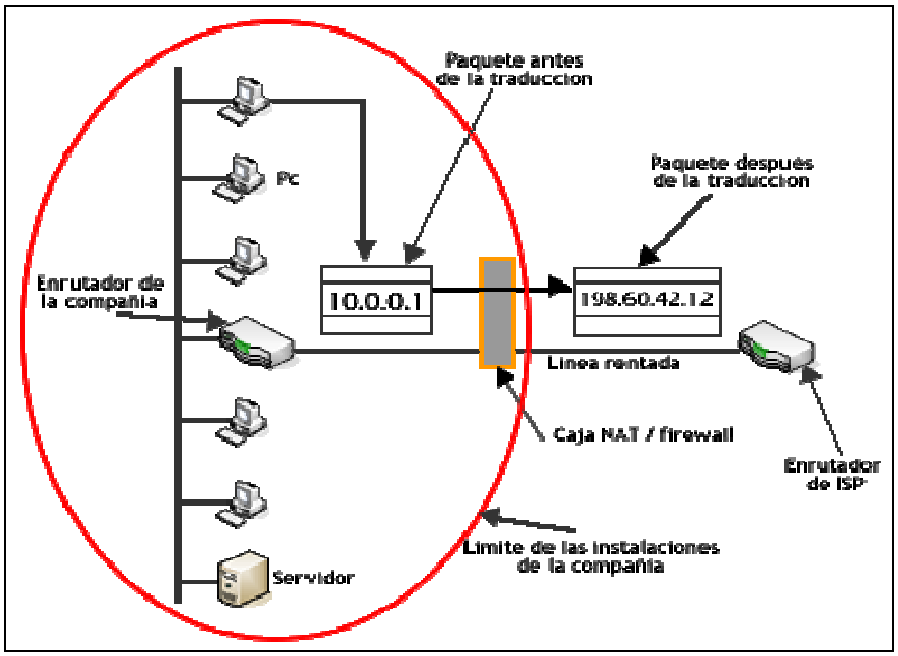


Figura 3.3. Colocación y funcionamiento de una caja NAT

Los diseñadores de esta caja NAT, tuvieron que usar los encabezados de TCP o UDP, para poder saber hacia qué host de la empresa se dirige cuando un

paquete cruza por el NAT. Esto lo resolvieron con ayuda del puerto origen, puerto destino, recalculando las sumas de verificación en los encabezados.

3.3 Novedades del direccionamiento IPv6

La característica distintiva más evidente de IPv6 es la utilización de direcciones de mucho mayor tamaño. El tamaño de una dirección en IPv6 es de 128 bits, que es cuatro veces mayor que el de una dirección de IPv4. Por lo tanto existe un espacio de direcciones de 128 bits que permite 2128 ó 340.282.366.920.938.463.463.374.607.431.768.211.456 (3.4x1038) direcciones posibles.

Con IPv6, es aún más difícil concebir el agotamiento del espacio de direcciones. Para ver el número con perspectiva, un espacio de direcciones de 128 bits proporciona 655.570.793.348.866.943.898.599 (6.5x1023) direcciones por cada metro cuadrado de la superficie terrestre.

Es importante destacar que la decisión de crear la dirección IPv6 con un tamaño de 128 bits no tenía como objetivo asignar 6.5 x 1023 direcciones a cada metro cuadrado de la Tierra. En su lugar, el tamaño relativamente grande de la dirección IPv6 está diseñado para subdividirse en dominios de enrutamiento jerárquicos que reflejen la topología de Internet en la actualidad. La utilización de 128 bits proporciona múltiples niveles de jerarquía y flexibilidad en el diseño del direccionamiento y enrutamiento jerárquicos, que son los elementos de los que carece actualmente la red Internet basada en IPv4.

Asignación actual

De forma similar a como está dividido el espacio de direcciones IPv4, el espacio de direcciones IPv6 está dividido en función del valor de los bits de orden superior de la dirección. Los bits de orden superior y sus valores fijos se denominan Prefijo de Formato (FP,Format Prefix).

En la tabla 3.2 se muestra la asignación del espacio de direcciones IPv6 por FP.

Asignación	Prefijo de formato (FP)	Fracción del espacio de direcciones
Reservado	0000 0000	1/256
Reservado para asignación NSAP	0000 001	1/128
Direcciones globales agregables de unidifusión	001	1/8
Direcciones de unidifusión locales del vínculo	1111 1110 10	1/1024
Direcciones de unidifusión locales del sitio	1111 1110 11	1/1024
Direcciones de multidifusión	1111 1111	1/256

Tabla 3.2. Asignación del espacio de direcciones IPV6, por prefijo de formato

El resto del espacio de direcciones IPv6 no está asignado.

El conjunto actual de direcciones de unidifusión que se pueden utilizar con los nodos IPv6 consta de direcciones globales agregables de unidifusión, direcciones de unidifusión locales del vínculo y direcciones de unidifusión locales del sitio. Estas direcciones sólo representan el 15 por ciento del espacio completo de direcciones IPv6.

Las direcciones IPv4 se representan en formato decimal con puntos. Esta dirección de 32 bits se divide en límites de 8 bits. Cada grupo de 8 bits se convierte a su equivalente decimal y se separa con puntos. En IPv6, la dirección de 128 bits se divide en límites de 16 bits y cada bloque de 16 bits se convierte a un número hexadecimal de 4 dígitos separado por un signo de dos puntos. La representación resultante se denomina hexadecimal con dos puntos.

La siguiente es una dirección IPv6 en formato binario:

0010000111011010 0000000011010011 0000000000000000 0010111100111011
0000001010101010 0000000011111111 1111111000101000 1001110001011010

Cada bloque de 16 bits se convierte a formato hexadecimal y se delimita mediante un signo de dos puntos. El resultado es:

21DA:00D3:0000:2F3B:02AA:00FF:FE28:9C5A

La representación IPv6 se puede simplificar aún más si se quitan los ceros iniciales de cada bloque de 16 bits. Sin embargo, cada bloque debe tener al menos un dígito. Al suprimir los ceros iniciales, la representación de la dirección se convierte en:

21DA:D3:0:2F3B:2AA:FF:FE28:9C5A

Configuración automática de direcciones IPv6

Un aspecto muy útil de IPv6 es su capacidad para configurarse automáticamente sin utilizar un protocolo de configuración con estado, como el Protocolo de Configuración Dinámica de Host para IPv6 (DHCPv6, Dynamic Host Configuration Protocol for IPv6).

Las direcciones configuradas automáticamente tienen uno o varios de los estados siguientes:

- Tentativo. Estado de una dirección cuya exclusividad está en proceso de comprobación. La comprobación se produce mediante la detección de direcciones duplicadas.
- Preferido. Estado de una dirección cuya exclusividad se ha comprobado. Un nodo puede enviar y recibir tráfico de unidifusión hacia y desde una dirección preferida. El período de tiempo que una dirección puede permanecer en estado tentativo o preferido se incluye en el mensaje de anuncio de enrutador.
- Desaconsejado. Estado de una dirección que sigue siendo válida, pero no se recomienda su uso para nuevas comunicaciones. Las sesiones de comunicación existentes pueden continuar utilizando una dirección desaconsejada. Un nodo puede enviar y recibir tráfico de unidifusión hacia y desde una dirección desaconsejada.
- Válido. Estado de una dirección desde la que se puede enviar y recibir tráfico de unidifusión. El estado válido abarca los estados preferido y desaconsejado. La cantidad de tiempo que una dirección puede permanecer en estado tentativo o válido se incluye en el mensaje de anuncio de enrutador. La duración válida debe ser mayor o igual que la duración preferida.

- No válido. Estado de una dirección con la que un nodo ya no puede enviar o recibir tráfico de unidifusión. Una dirección pasa al estado no válido cuando caduca la duración válida.

La figura 3.4 muestra la relación entre los estados de una dirección configurada automáticamente, la duración preferida y la duración válida.

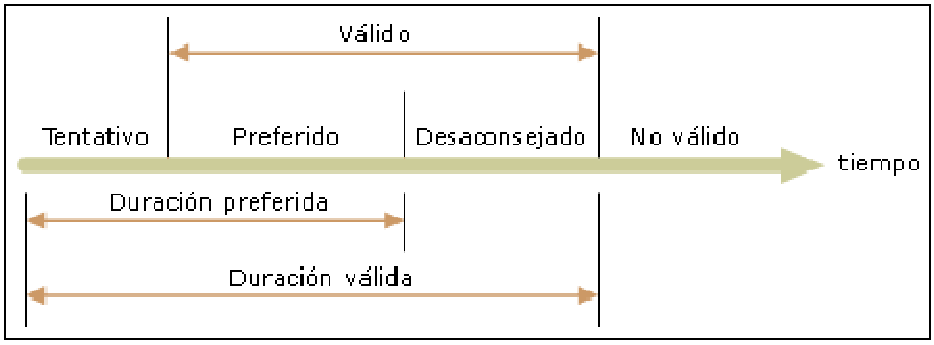


Figura 3.4. Estado de una dirección configurada automáticamente

A excepción de las direcciones locales del vínculo, la configuración automática de direcciones sólo se especifica para los hosts. Los enrutadores deben obtener los parámetros de dirección y configuración por otros medios (por ejemplo, mediante configuración manual).

Proceso de configuración automática

El proceso de configuración automática de direcciones en un nodo IPv6 se produce de la manera siguiente:

1) Se deriva una dirección local del vínculo tentativa, basada en el prefijo local del vínculo de FE80::/64 y el identificador de interfaz de 64 bits.
2) Se lleva a cabo la detección de direcciones duplicadas para comprobar la exclusividad de la dirección local del vínculo tentativa.
3) Si se produce un error en la detección de direcciones duplicadas, se debe realizar la configuración manual en el nodo.
4) Si la detección de direcciones duplicadas tiene éxito, la dirección local del vínculo tentativa se considera única y válida. La dirección local del vínculo se inicializa para la interfaz. La dirección correspondiente de

nivel de vínculo de multidifusión para el nodo solicitado se registra en el adaptador de red.

3.4 Enrutamiento

En términos generales, el enrutamiento es el proceso de reenviar paquetes entre dos redes conectadas. En cuanto a las redes basadas en TCP/IP, el enrutamiento forma parte del Protocolo Internet (IP) y se utiliza junto con otros servicios de protocolo de red para proporcionar capacidades de reenvío entre hosts que se encuentran en segmentos de red distintos dentro de una red basada en un TCP/IP más grande.

IP es la "oficina de correos" del protocolo TCP/IP, donde se ordenan y entregan los datos IP. Cada paquete entrante o saliente se denomina datagrama IP. Un datagrama IP contiene dos direcciones IP: la dirección de origen del host que realiza el envío y la dirección de destino del host receptor. A diferencia de las direcciones de hardware, las direcciones IP de un datagrama siguen siendo las mismas durante su transmisión a través de una red TCP/IP.

3.4.1 Enrutadores IP

Los segmentos de red TCP/IP están conectados entre sí mediante enrutadores IP, que son los dispositivos que transmiten los datagramas IP desde un segmento de red a otro. Este proceso se conoce como enrutamiento IP (véase figura 3.5).

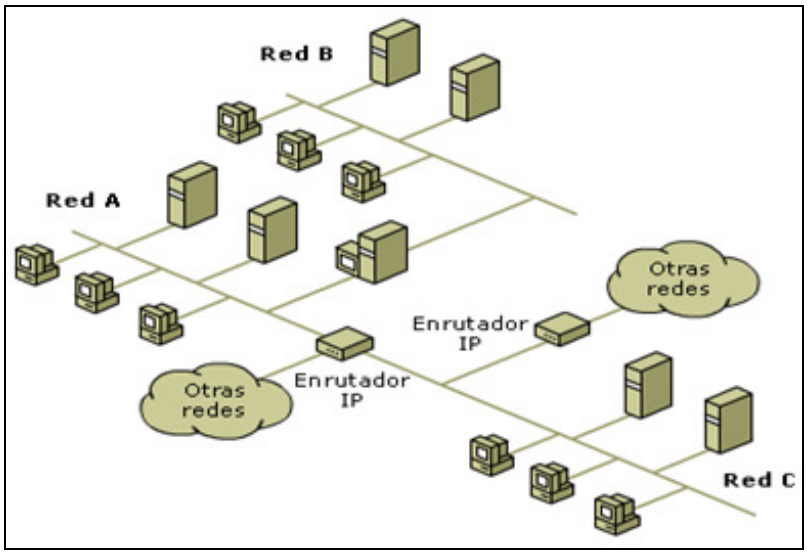


Figura 3.5. Enrutamiento IP

Los enrutadores IP proporcionan el medio principal para unir dos o más segmentos de red IP separados físicamente. Todos los enrutadores IP comparten dos características fundamentales:

- Los enrutadores IP son de hosts múltiples. Un equipo de hosts múltiples es un host de red que utiliza dos o más interfaces de conexión de red para conectarse a cada segmento de red separado físicamente.
- Los enrutadores IP permiten el reenvío de paquetes a otros hosts TCP/IP. Los enrutadores IP se diferencian de otros hosts multitarjeta en una característica importante: un enrutador IP debe ser capaz de reenviar la comunicación basada en IP entre redes para otros hosts de la red IP.

Los enrutadores IP se pueden implementar mediante varios productos de hardware y software posibles. Comúnmente se utilizan enrutadores basados en hardware (dispositivos de hardware dedicados que ejecutan software especializado). Además, se pueden utilizar soluciones de enrutamiento basadas en software, como los servicios de enrutamiento y acceso remoto.

Independientemente del tipo de enrutadores IP que utilice, todo el enrutamiento IP está basado en el uso de una tabla de enrutamiento para la comunicación entre los segmentos de red.

Los enrutadores tienen direcciones y enlaces a dos o más redes al mismo tiempo. En su función más simple, los enrutadores reciben paquete de una red y la pasan a una segunda red conectada. Sin embargo, si un paquete recibido se dirige a un nodo de una red de la cual el enrutador no es miembro, el enrutador es capaz de determinar cuál de las redes a las que esta conectado es la mejor para retransmitir el paquete. Una vez que un enrutador ha identificado la mejor ruta para el paquete, lo pasa a otro enrutador de la red apropiada. El enrutador comprueba la dirección destino, busca la que considera mejor ruta para el paquete y lo pasa a la dirección destino (si esa red es una red vecina) o a través de una red vecina al siguiente enrutador situado en el camino elegido.

3.4.2 Tablas de enrutamiento

Los hosts TCP/IP utilizan una tabla de enrutamiento para mantener información acerca de otras redes IP y hosts IP. Las redes y los hosts se identifican mediante una dirección IP y una máscara de subred. Además, las tablas de enrutamiento son importantes ya que proporcionan la información necesaria a cada host local respecto a cómo comunicarse con redes y hosts remotos.

En cada equipo de una red IP, puede mantener una tabla de enrutamiento con una entrada para cada equipo o red que se comunica con el equipo local. En general, esto no es práctico y se utiliza una puerta de enlace predeterminada (enrutador IP) en su lugar.

Cuando un equipo se prepara para enviar un datagrama IP, inserta su propia dirección IP de origen y la dirección IP de destino del destinatario en el encabezado IP. A continuación, el equipo examina la dirección IP de destino, la compara con una tabla de enrutamiento IP mantenida localmente y realiza la acción adecuada según la información que encuentra. El equipo realiza una de las tres acciones siguientes:

- Pasa el datagrama a un nivel de protocolo superior a IP en el host local.
- Reenvía el datagrama a través de una de las interfaces de red conectadas.
- Descarta el datagrama.

IP busca en la tabla de enrutamiento la ruta que más se parezca a la dirección IP de destino. La ruta, en orden de más a menos específica, se localiza de la manera siguiente:

- Una ruta que coincida con la dirección IP de destino (ruta de host).
- Una ruta que coincida con el Id. de red de la dirección IP de destino (ruta de red).
- La ruta predeterminada.

Si no se encuentra una ruta coincidente, IP descarta el datagrama.

3.5 Algoritmos de enrutamiento

La capa de red proporciona la dirección lógica que permite que dos sistemas dispares que se encuentran en redes lógicas diferentes determinen una posible ruta para comunicarse. Así mismo en esta capa residen los algoritmos que implementan los protocolos de enrutamiento.

Los algoritmos de enrutamiento son parte del software de la capa de red encargada de decidir la línea de salida por la que se transmitirá un paquete de entrada. Si la subred usa datagramas entonces esta decisión debe hacerse cada vez que llega un paquete de datos de entrada, debido a que la mejor ruta podría haber cambiado desde la última vez.

Si la subred utiliza circuitos virtuales internamente, las decisiones de enrutamiento se tomarán sólo al establecerse el circuito y los paquetes seguirán la ruta previamente establecida.

3.5.1 Clasificación de los algoritmos

- *Algoritmos no adaptables:* No basan sus decisiones de enrutamiento en mediciones o estimaciones del tráfico ni en la topología. La decisión de qué ruta tomar de **I** a **J** se calcula por adelantado, fuera de línea y se cargan en los routers al iniciar la red. Éste procedimiento se llama ***enrutamiento estático***. La desventaja de este tipo de algoritmos es que no es posible responder a situaciones cambiantes como por ejemplo saturación, exceso de tráfico o fallo en una línea.

En un conjunto de redes complejas, se necesita cierto grado de cooperación "dinámica" entre los dispositivos de encaminamiento. En particular se deben evitar aquellas porciones de red que sufren congestión, entendiéndose esto como aquella situación donde hay demasiados paquetes en alguna parte de la subred, y como consecuencia el rendimiento de ésta baja.

Para poder tomar estas decisiones de encaminamiento dinámicas, los dispositivos involucrados en el ruteo deben intercambiar información usando algoritmos de encaminamiento especiales para este propósito. La información que se necesita sobre el estado del conjunto de redes tiene que venir expresada en términos de qué redes son accesibles a través de qué dispositivos y en términos de las características de retardo de varias rutas.

- *Algoritmos adaptables:* En contraste con los algoritmos no adaptables, éstos cambian sus decisiones de enrutamiento para reflejar los cambios de topología y de tráfico. Difieren de los algoritmos estáticos en el lugar de obtención de su información (ej. localmente, en los routers adyacentes o de todos), el momento del cambio de sus rutas (ej. cada determinados seg., o cuando cambia la carga) y la métrica usada para la optimización (ej. distancia, nº de escalas, tiempo estimado del tránsito). Este tipo de algoritmos no pueden ser demasiado complejos ya que son implementados en los routers y deben ejecutarse en tiempo real con recursos de CPU y la memoria con que el router dispone.

3.5.2 Algoritmos estáticos

Las rutas estáticas se definen administrativamente y establecen rutas específicas que han de seguir los paquetes para pasar de un puerto de origen hasta un puerto de destino. Si los routers pueden aprender automáticamente cuál es la información de la ruta, puede parecer inútil ingresar información en la tabla de enrutamiento del router de forma manual.

Las rutas estáticas se utilizan habitualmente en enrutamiento desde una red hasta una red de conexión única, ya que no existe más que una ruta de entrada y salida en una red de conexión única, evitando de este modo la sobrecarga de tráfico que genera un protocolo de enrutamiento.

La ruta estática se configura para conseguir conectividad con un enlace de datos que no esté directamente conectado al router. Para conectividad de extremo a extremo, es necesario configurar la ruta en ambas direcciones.
Las rutas estáticas permiten la reconstrucción manual de la tabla de enrutamiento. Con las rutas estáticas, el administrador a de volver a configurar todos los routers para ajustarlos cuando se produce un cambio en la red.

Enrutamiento por trayectoria más corta

Esta es una técnica de amplio uso en muchas formas. Por ejemplo armar un grafo de la subred en el que cada nodo representa un enrutador y cada arco del grafo una línea de comunicación (enlace). Para seleccionar la ruta entre un par dado de enrutadores, el algoritmo simplemente encuentra en el grafo la trayectoria más corta entre ellos.
El concepto de ***trayectoria más corta*** se debe a que la forma de medir la longitud de la ruta es usando alguna métrica (número de saltos, la distancia física, el retraso de transmisión por un paquete de prueba, el ancho de banda, el tráfico promedio, el costo de comunicación, etc.).
Se conocen varios algoritmos de cálculo de la trayectoria más corta entre dos nodos de un grafo. Cada nodo se etiqueta (entre paréntesis) con su distancia al nodo de origen a través de la mejor trayectoria conocida. Inicialmente no se conocen trayectorias, por lo que todos los nodos tienen la etiqueta infinito. A medida que avanza el algoritmo y se encuentran trayectorias, pueden cambiar las etiquetas, reflejando mejores trayectorias. Una etiqueta puede ser tentativa o permanente.
Inicialmente todas las etiquetas son tentativas. Al descubrirse que una etiqueta representa la trayectoria más corta posible del origen a ese nodo, se vuelve permanente y no cambia más.

Inundación

Otro algoritmo estático es la inundación, en la que cada paquete de entrada se envía por cada una de las líneas de salida, excepto aquella por la que llegó. La inundación evidentemente genera grandes cantidades de paquetes duplicados, de hecho, una cantidad infinita a menos que se tomen algunas medidas para limitar ese proceso. Una de tales medidas puede ser un contador de escalas contenido en la cabecera de cada paquete, el cual disminuye en cada escala, descartándose al llegar el contador a cero. Idealmente el contador debe inicializarse a la longitud de la trayectoria; puede inicializar el contador en el peor de los casos, es decir, el diámetro de la subred.

Una variación de la inundación, un poco más práctica es la *inundación selectiva.* En este algoritmo, los enrutadores no envían cada paquete de entrada por todas las líneas, sino sólo por aquellas que van aproximadamente en la dirección correcta.

La inundación no es práctica en la mayoría de las aplicaciones, pero tiene algunos usos. Por ejemplo, en aplicaciones militares y en las aplicaciones de bases de datos distribuidas a veces es necesario actualizar concurrentemente todas las bases de datos, en cuyo caso puede ser útil la inundación.

3.5.3 Algoritmos dinámicos

Los algoritmos adaptativos (dinámicos), cambian sus decisiones de enrutamiento para reflejar los cambios de topología y tráfico. Los algoritmos adaptativos difieren en el lugar de donde obtienen su información (por ejemplo, localmente, de los enrutadores adyacentes o de todos los enrutadores), el momento de cambio de sus rutas (cuando cambia la carga o cuando cambia la topología) y la métrica usada para la optimización (por ejemplo, distancia, número de saltos o tiempo estimado de tránsito).

Este tipo de enrutamiento, es usado por prácticamente en todas las redes de conmutación de paquetes. Las decisiones de enrutamiento cambian conforme cambia el estado de la red, debido a fallos y desconexiones de enlaces o a la congestión. Además de que necesita información de la red constantemente.

Enrutamiento basado en el vector distancia

En el encaminamiento basado en el vector distancia, cada enrutador periódicamente comparte su conocimiento sobre la red entera con sus vecinos. Hay tres claves para comprender el funcionamiento de este algoritmo:

1) Conocimiento de toda la red. Cada enrutador comparte su conocimiento sobre la red entera. Envía todo el conocimiento que tiene sobre la red a sus vecinos.
2) Enrutamiento sólo a los vecinos. Cada enrutador envía periódicamente su conocimiento sobre la red sólo a los enrutadores sobre los que tiene enlaces directos. Envía el conocimiento que tenga sobre la red completa a través de todos sus puertos. Esta información es recibida y almacenada en cada enrutador vecino, y utilizada para actualizar la propia información que tiene el enrutador sobre la red.
3) Se comparte información a intervalos regulares. Por ejemplo, cada 30 segundos, cada enrutador envía su información sobre la red completa a sus vecinos. Este envío ocurre haya cambiado o no la red desde el último intercambio de información.

En el enrutamiento basado en el vector distancia, cada enrutador comparte su conocimiento sobre la red entera con sus vecinos.

Para comprender el funcionamiento del enrutamiento basado en el vector distancia, se va a examinar la Internet mostrada en la figura 3.6. En donde las nubes representan redes de área local (LAN). El número situado dentro de cada nube representa el identificador de la red de área local. Estas LAN pueden ser de cualquier tipo (Ethernet, de anillo con paso de testigo, FDDI, etc.). Las LAN se conectan entre sí por medio de enrutadores (o pasarelas), representadas por las cajas etiquetadas como A, B, C, D, E y F. El encaminamiento basado en vector distancia, el coste se basa en contar los saltos.

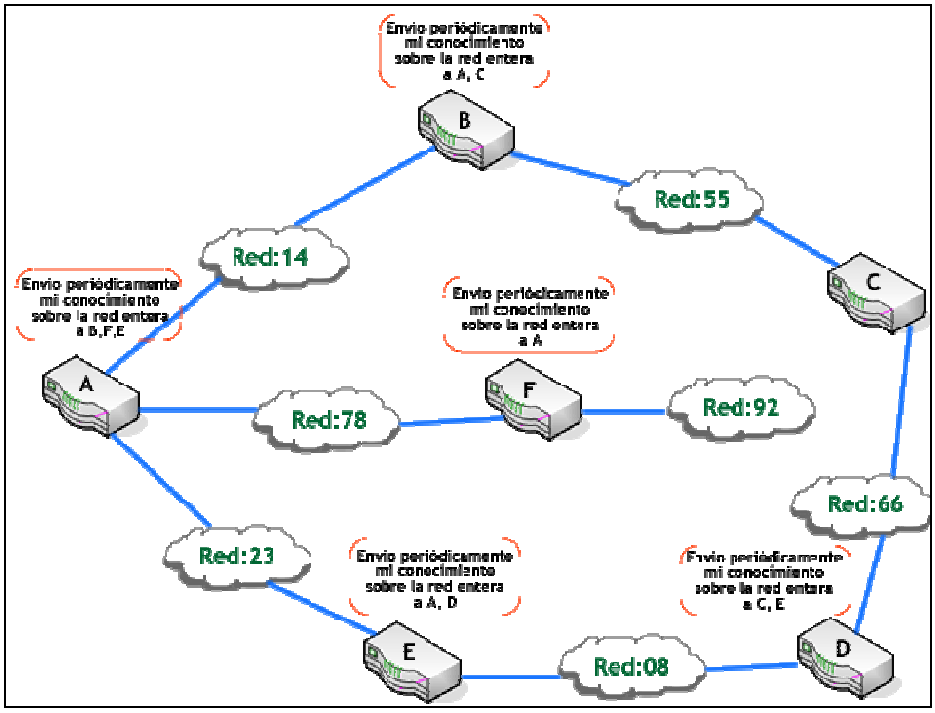


Figura 3.6. El concepto de enrutamiento basado en vector distancia

Como se puede observar en la figura anterior, cada enrutador envía su información sobre la red sólo a sus vecinos inmediatos. Es decir los vecinos añaden su conocimiento a su propio conocimiento y envían la tabla completa a sus propios vecinos. De esta forma, el primer enrutador obtiene su propia información devuelta más nueva información sobre los vecinos de sus vecinos. Cada uno de estos vecinos añade su conocimiento y envía la tabla actualizada a sus propios vecinos (a los vecinos de los vecinos de los vecinos del enrutador original), y así de forma sucesiva. En algún momento, cada enrutador conoce todo acerca de los enrutadores de la red entera.

Enrutamiento basado en el estado enlace

En el enrutamiento estado enlace, cada enrutador comparte el conocimiento que tiene de sus vecinos con el resto de los encaminadores de la red. Se cumplen las siguientes afirmaciones para el encaminamiento basado en el estado del enlace:

1) Conocimiento sobre sus vecinos. En lugar de enviar su tabla de encaminamiento entera, un enrutador sólo envía información sobre su vecindad.

2) A todos los enrutadores. Cada enrutador envía esta información a todos los enrutadores de la red, no sólo a sus vecinos. Esto se hace mediante un proceso denominado **inundación**. La inundación significa que un enrutador envía su información a todos sus vecinos (a través de todos sus puertos de salida). Cada vecino envía el paquete a todos sus vecinos y así sucesivamente. Cada enrutador que recibe el paquete envía copias a todos sus vecinos. Finalmente cada enrutador (sin excepción) recibe una copia de la misma información.
3) Compartir información cuando hay cambios. Cada enrutador envía la información sobre sus vecinos cuando hay algún cambio.

El enrutador basado en el estado enlace, cada enrutador envía el conocimiento que tiene sobre sus vecinos a todos los enrutadores de la red.

Compartir información

La primera etapa en el encaminamiento basado en el estado del enlace es compartir información (véase figura 3.7). Cada enrutador envía el conocimiento que tiene sobre sus vecinos a todos los enrutadores de la red.

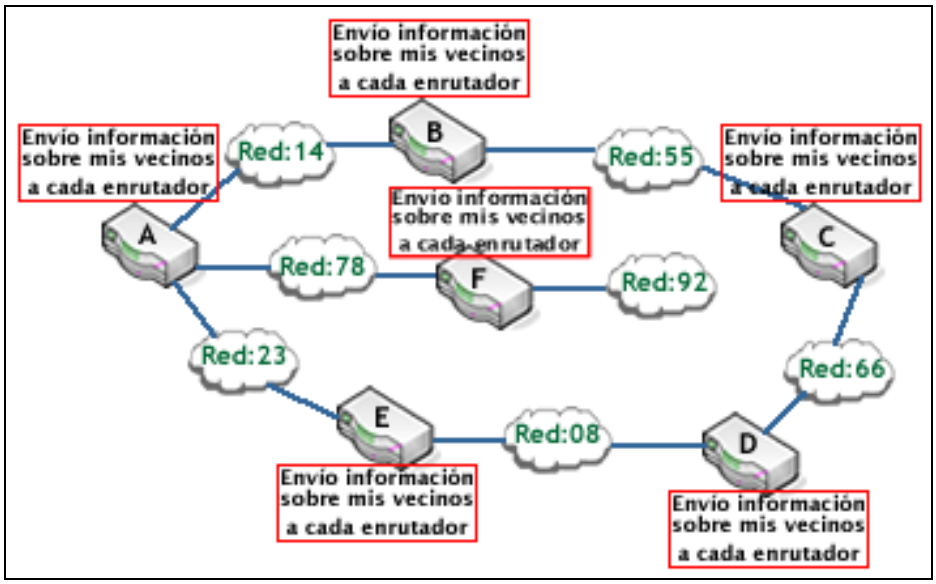


Figura 3.7. Enrutamiento basado en el estado del enlace

Costo del paquete

El enrutamiento basado en el vector distancia, el costo se refiere al contador de saltos. En el enrutamiento basado en el estado del enlace, el costo es un valor con peso basado en una variedad de factores como los niveles de seguridad, el tráfico o el estado del enlace. El costo se aplica cuando un paquete deja el enrutador. Dos factores gobiernan la forma en la que el costo se aplica a los paquetes en la determinación de una ruta (véase figura 3.8):

- El costo es aplicado sólo por los enrutadores y no por el resto de estaciones de red. El enlace de un enrutador al siguiente es una red, no un cable punto a punto. En muchas topologías (como un anillo o un bus), cada estación de la red examina la cabecera de cada paquete que pasa. Si el costo fuera añadido a cada estación, se acumularía de forma impredecible (el número de estaciones en una red puede cambiar por varios motivos, muchos de ellos impredecibles).
- El costo se aplica cuando el paquete deja el enrutador, no cuando entra. La mayoría de las redes son redes de difusión. Cuando un paquete se encuentra en la red, cada estación, incluido el enrutador puede capturarlo. Por tanto, no se puede asignar ningún costo a un paquete cuando va de una red a un enrutador.

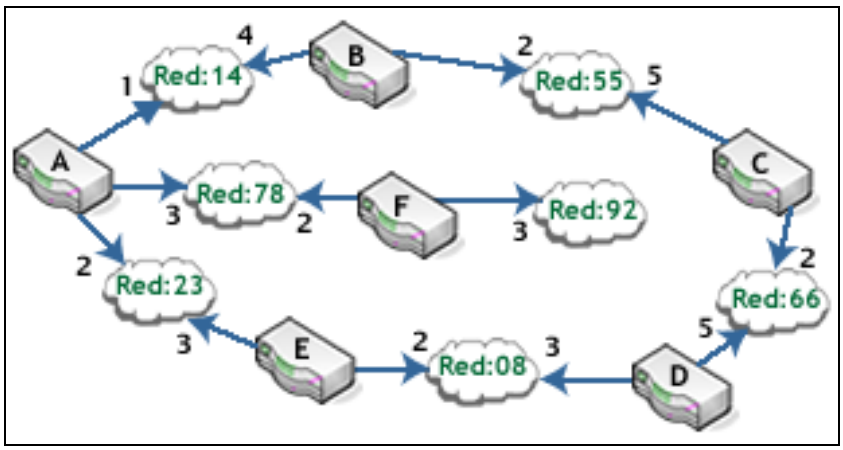


Figura 3.8. Costo en el enrutamiento basado en el estado del enlace

Paquete con el estado del enlace

Cuando un enrutador inunda la red con la información sobre sus vecinos, se dice que va a publicar. La base de está publicación es un paquete pequeño denominado Paquete de Estado del Enlace (LSP); (véase figura 3.9). Un LSP normalmente contiene 4 campos: el identificador del que realiza la publicación, el identificador de la red destino, el coste y el identificador del enrutador vecino.

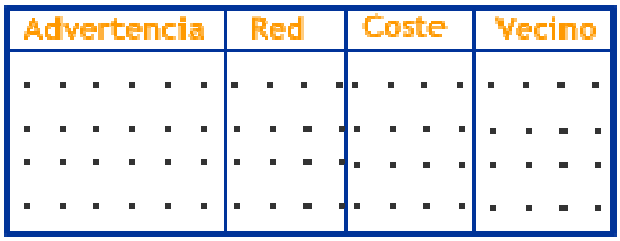


Figura 3.9. Paquete de estado del enlace.

Obtención de información sobre los vecinos

Un enrutador obtiene información sobre sus vecinos de forma periódica enviándoles un pequeño paquete de saludo. Si los vecinos responden a este saludo, como es de esperar, se asume que están funcionando. Si no lo hacen, se asume que ha ocurrido un cambio y el enrutador que envía el paquete de saludo alerta al resto de la red en su siguiente LSP. Estos saludos, son lo bastante pequeños para que no utilicen de forma significativa recursos de la red.

Inicialización

Imaginemos que todos los enrutadores de nuestra red de ejemplo comienzan a funcionar al mismo tiempo. Cada enrutador envía un paquete de saludo a sus vecinos para determinar el estado de cada enlace. Luego se prepara el paquete LSP basado en el resultado de estos mensajes de saludo e inunda la red con él. La figura 3.10, muestra este proceso para el enrutador A. Las mismas etapas son realizadas por cada enrutador de la red cuando se inicializan.

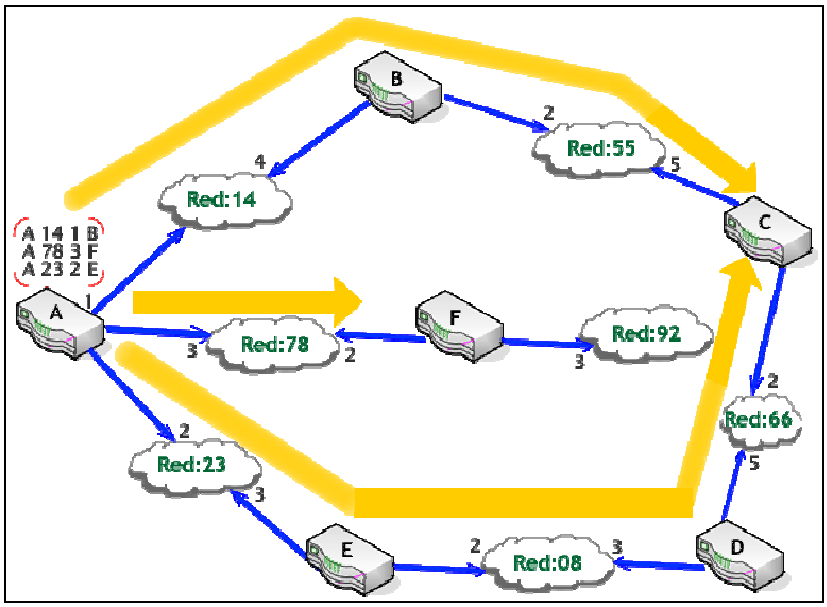


Figura 3.10. Inundación del LSP de A

Base de datos de enlaces

Cada enrutador recibe lada LSP y coloca la información en una base de datos de estados de enlaces. La tabla 3.3, muestra la base de datos para nuestra red de ejemplo.

Anunciante	Red	Coste	Vecino
A	14	1	B
A	78	3	F
A	23	2	E
B	14	4	A
B	55	2	C
C	55	5	B
C	66	2	D
D	66	5	C
D	8	3	E
E	23	3	A
E	8	2	D
F	78	2	A
F	92	3	

Tabla 3.3. Base de datos de estados de enlaces

Debido a que cada enrutador recibe los mismos LSP, cada enrutador construye la misma base de datos. Almacena la misma base de datos en disco y la utiliza para calcular su tabla de enrutamiento. Si un enrutador se añade o se elimina

del sistema, la base de datos completa debe ser compartida para una rápida actualización.

Algoritmo de Dijskstra

Para calcular la tabla de enrutamiento, cada enrutador aplica un algoritmo denominado algoritmo de Dijkstra a su base de datos de estados de enlaces.
El algoritmo de Dijkstra calcula el camino más corto entre dos puntos de una red utilizando un grafo de nodos y arcos. Los nodos son de dos tipos: *redes y enrutadores*. Los arcos son las conexiones entre un enrutador y una red (enrutador a red y red a enrutador). El coste sólo se aplica al arco situado entre un enrutador y la red. El costo del arco de una red a un enrutador siempre es cero (véase figura 3.11).

Árbol del camino más corto

El algoritmo de dijkstra sigue cuatro pasos para descubrir lo que se denomina árbol del camino más corto (tabla de enrutamiento) para cada enrutador:

- El algoritmo empieza a construir un árbol identificando su raíz. La raíz de cada árbol en cada enrutador es el propio enrutador. El algoritmo, a continuación, añade todos los nodos que pueden ser alcanzados desde esa raíz, en otras palabras, todos los nodos vecinos. Los nodos y los arcos son temporales en esta etapa.
- El algoritmo compara los arcos temporales del árbol e identifica el arco con el coste acumulado más bajo. Este arco y el nodo al que se conecta se hacen permanentes en el árbol del camino mas corto.
- El algoritmo examina la base de datos e identifica a cada nodo que puede ser alcanzado desde su nodo elegido. Estos nodos y sus arcos se añaden de forma temporal al árbol.
- Las dos últimas etapas se repiten hasta que cada nodo de la red se ha convertido en parte permanente del árbol. Los únicos arcos permanentes son aquellos que representan la ruta (de menor costo) más corta a cada nodo.

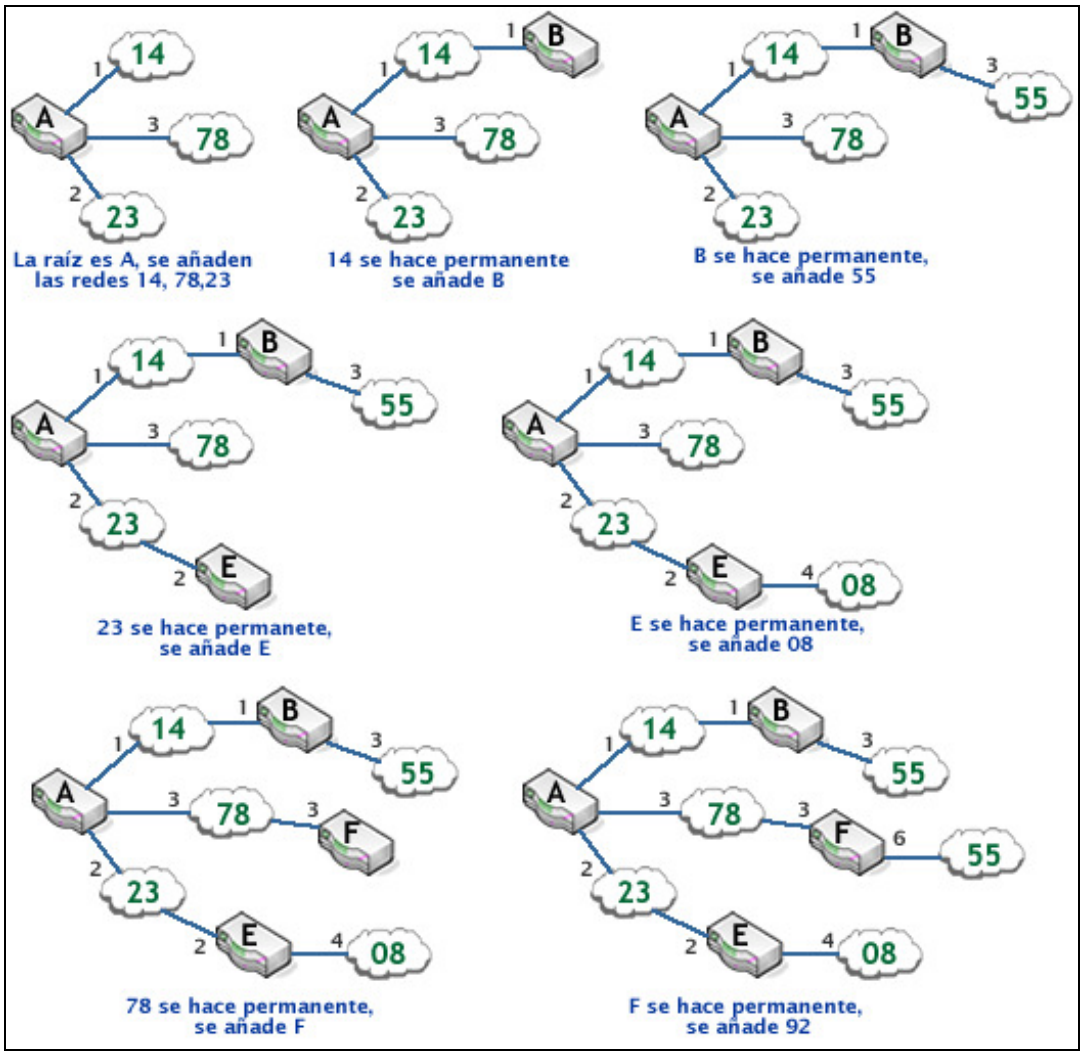


Figura 3.11. Cálculo del camino más corto, parte 1

La figura 3.11 muestra las etapas del algoritmo de Dijkstra aplicadas por el nodo A de nuestra internet de ejemplo. El número con el coste al lado de cada nodo representa el coste acumulado desde el nodo raíz, no el coste del arco individual. El segundo y el tercer paso se repiten hasta que cuatro nodos más se hacen permanentes.

La figura 3.12, muestra la obtención del árbol del camino más corto para el enrutador A.

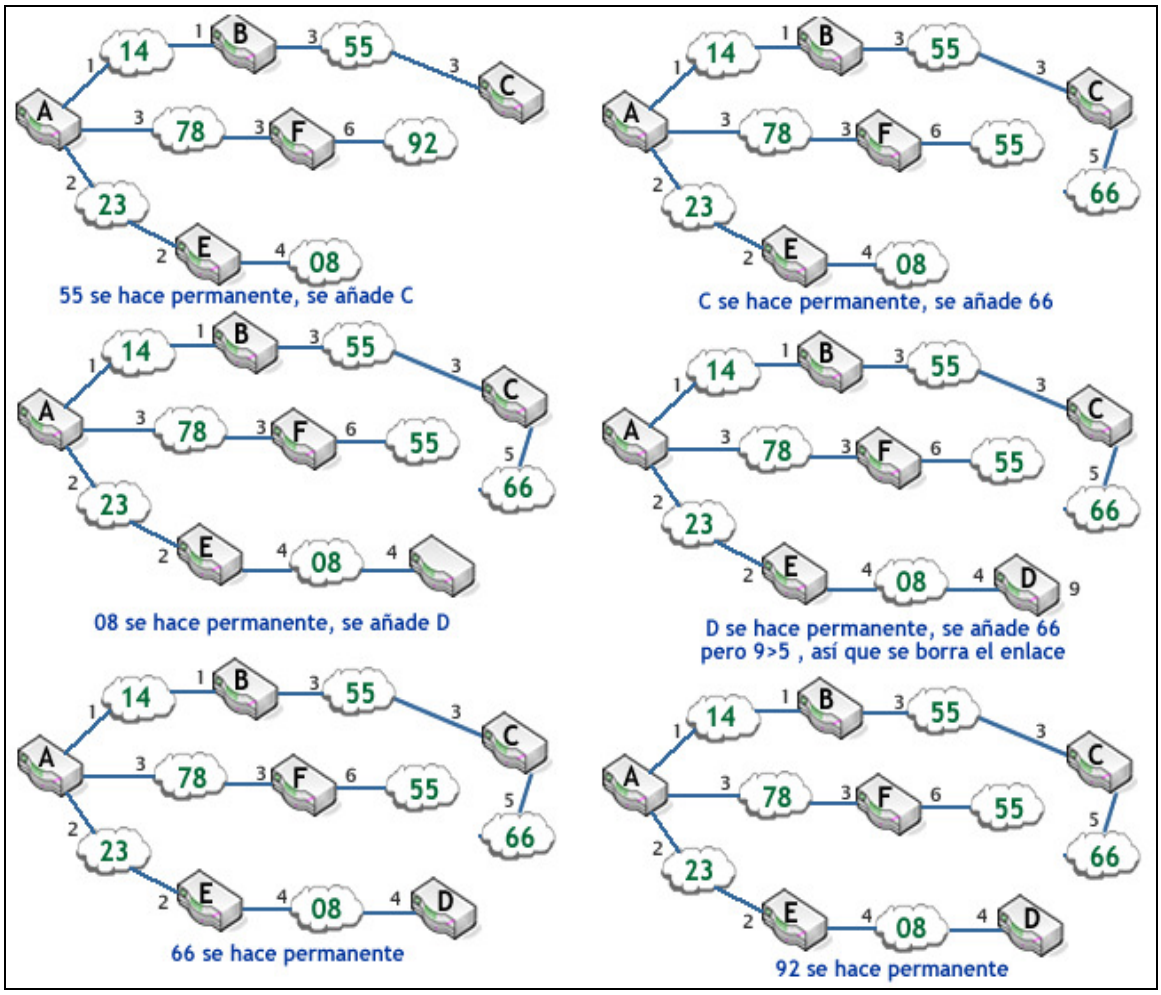


Figura 3.12. Cálculo del camino más corto, parte 2

Tabas de enrutamiento

Cada enrutador ahora utiliza el árbol del camino más corto para construir su tabla de enrutamiento. Cada enrutador utiliza el mismo algoritmo y la misma base de datos de estados de enlaces para calcular su propio árbol de camino más corto y su tabla de enrutamiento: estas son diferentes para cada enrutador. La tabla 3.4, muestra los datos obtenidos por el enrutador A.

Red	Coste	Siguiente encaminador
08	4	E
14	1	-
23	2	-
55	3	B
66	5	B
78	3	-
92	6	F

Tabla 3.4. Tabla de enrutamiento con los estados de los enlaces para el enrutador A

Enrutamiento por difusión

En algunas aplicaciones, los host necesitan enviar mensajes a varios otros host o a todos los demás. El envío simultáneo de un paquete a todos los destinos se llama **difusión**; se han propuesto varios métodos para llevarla a cabo.

Un método de difusión que no requiere características especiales de la subred es el origen simplemente envíe un paquete distinto a todos los destinos. El método no sólo desperdicia ancho de banda, sino que también requiere que el origen tenga una lista completa de todos los destinos, este método es el menos deseable.

La **inundación** es otro candidato obvio. Aunque ésta es poco adecuada para la comunicación punto a punto ordinaria. Este algoritmo genera demasiados paquetes y consume demasiado ancho de banda.

Un tercer algoritmo es el **enrutamiento multidestino**. Con este método, cada paquete contiene una lista de destinos o un mapa de bits que indica los destinos para determinar el grupo de líneas de salida que necesitará (se necesita una línea de salida si es la mejor ruta a cuando menos uno de los destinos). El enrutador genera una copia nueva del paquete para cada línea de salida que se utilizará, e incluye en cada paquete sólo aquellos destinos que utilizarán la línea. Este tipo de enrutamiento es como los paquetes con direccionamiento individual, excepto que, cuando varios paquetes deben seguir la misma ruta, uno de ellos paga la tarifa completa y los demás viajan gratis.

Un cuarto algoritmo de difusión esa explícitamente el árbol del sumidero para el enrutador que inicia difusión, o cualquier otro árbol de expansión adecuado. El **árbol de expansión** es un subgrupo de la subred que incluye todos los enrutadores pero no contiene ciclos. Cada enrutador puede copiar un paquete de entrada difundido en todas las líneas del árbol de expansión, excepto en aquella por la que llegó. Este método utiliza de manera óptima el ancho de banda, generando la cantidad mínima de paquetes necesarios para llevar a cabo el trabajo. Cada enrutador debe tener conocimiento de algún árbol de expansión para que este método pueda funcionar.

Por último existe un algoritmo que se usa cuando los enrutadores no saben nada en absoluto sobre árboles de expansión. La idea, llamada **Reenvío por**

Ruta Invertida (Reverse Path Forwarding). Cuando llega un paquete difundido a un enrutador, este lo revisa para ver si llegó por la línea normalmente usada para enviar paquetes al origen de la difusión. De ser así, hay excelentes posibilidades de que el paquete difundido haya seguido la mejor ruta desde el enrutador y, por lo tanto, sea la primera copia en llegar al enrutador. Si este es el caso, el enrutador reenvía copias del paquete a todas las líneas, excepto a aquella por la que llegó. Sin embargo si el paquete difundido llegó por una línea diferente de la preferida, el paquete se descarta como probable duplicado.

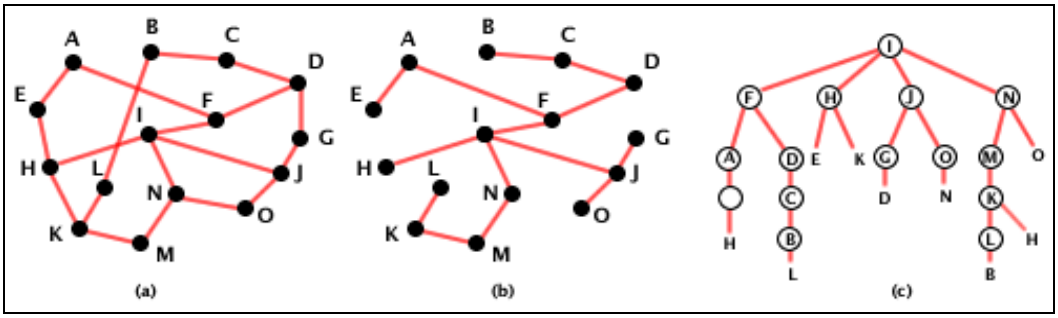


Figura 3.13. Reenvió por ruta invertida. (a) Subred. (b) Árbol sumidero.
(c) Árbol construido mediante reenvío por ruta invertida

La figura 3.13, muestra un ejemplo del reenvío por ruta invertida. En la parte (a) se muestra una subred, en la parte (b) se muestra un árbol sumidero para el enrutador ***I*** de la subred, y en la parte (c) se muestra el funcionamiento del algoritmo de ruta invertida. En el primer salto, ***I*** envía paquetes a F, H, J y N, como lo indica la segunda fila del árbol. Cada uno de estos paquetes llega a ***I*** por la ruta preferida (suponiendo que la ruta preferida pase a través del árbol sumidero). En el segundo salto, se generan ocho paquetes, dos por cada uno de los enrutadores que recibieron un paquete en el primer salto. Como resultado, los ocho llegan a enrutadores no visitados previamente, y cinco llegan a través de la línea preferida. De los seis paquetes generados en el tercer salto, sólo tres llegan por la ruta preferida (C, E y K); los otros son duplicados. Después de cinco saltos y 24 paquetes, termina la difusión en comparación con 4 saltos y 14 paquetes si se hubiera seguido exactamente el árbol sumidero.

Una de las ventajas principales del **reenvío por ruta invertida** es que es razonablemente eficiente y fácil de implementar. No requiere que los enrutadores conozcan los árboles de expansión ni tiene la sobrecarga de una lista de destinos o de un mapa de bits en cada paquete de difusión, como los tiene el direccionamiento multidestino.

Enrutamiento por multidifusión

Existen algunas aplicaciones que requieren que procesos muy separados trabajen juntos en grupo. En estos casos, con frecuencia es necesario que un proceso envíe un mensaje a todos los demás miembros del grupo. Si el grupo es pequeño, simplemente se puede transmitir a cada uno de los miembros un mensaje punto a punto. Si el grupo es grande, esta estrategia es costosa. Por lo tanto, necesitamos una manera de enviar mensajes a grupos bien definidos de tamaño numéricamente grande, pero pequeños en comparación con la totalidad de la red.

El envío de un mensaje a uno de tales grupos se llama multidifusión, y su algoritmo de enrutamiento es el **enrutamiento por multidifusión**.

Para la multidifusión se requiere administración de grupo. Se necesita una manera de crear y destruir grupos, y un proceso para que los procesos se unan a los grupos y salgan de ellos. Es importante que los enrutadores sepan cuáles de sus host pertenecen a qué grupos. Los host deben informar a sus enrutadores de los cambios en los miembros del grupo, o los enrutadores deben enviar de manera periódica la lista de sus host. De cualquier manera, los enrutadores aprenden qué hosts pertenecen a cuáles grupos. Los enrutadores les dicen a sus vecinos, de manera que la información se propaga a través de la subred.

Para realizar enrutamiento de multidifusión, cada enrutador calcula un árbol de expansión que cubre a todos los demás enrutadores de la subred. Por ejemplo, en la figura 3.14(a) tenemos una subred con dos grupos, 1 y 2. Algunos enrutadores están conectados a host que pertenecen a uno o ambos grupos, como se indica en la figura. En la figura 3.14(b) se muestra un árbol de expansión para el enrutador de la izquierda.

Cuando un proceso envía un paquete de multidifusión a un grupo, el primer enrutador examina su árbol de expansión y lo recorta, eliminado todas las líneas que conduzcan a hosts que no sean miembros del grupo. En el ejemplo de la figura 3.14(c) se muestra el árbol de expansión recortado del grupo 1. De la misma manera, en la figura 3.14(d) se presenta el árbol de expansión recortado del grupo 2. Los paquetes de multidifusión se reenvían solo a través del árbol de expansión apropiado.

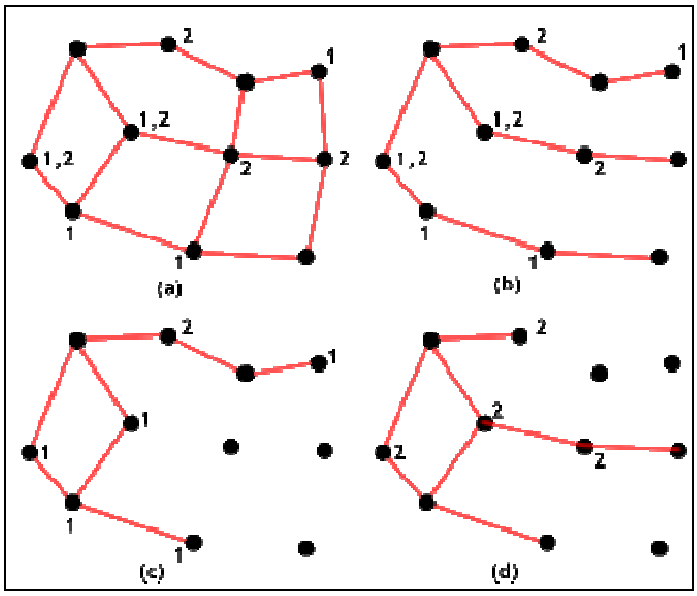


Figura 3.14. Enrutamiento de multidifusión

Hay varias maneras de recortar el árbol de expansión. Se puede empezar por el final de cada ruta y trabajando hacia la raíz, eliminando todos los enrutadores que no pertenezcan al grupo en cuestión. Una desventaja potencial de este algoritmo es que no escala bien en redes grandes. Suponga que una red tiene ***n*** grupos, cada uno con un promedio de ***m*** miembros. Por cada grupo se deben almacenar **m** árboles de expansión recortados, lo que da un total de **mn** árboles. Cuando hay muchos grupos grandes, se necesita bastante espacio para almacenar todos los árboles.

Un diseño alterno utiliza **árboles de núcleo** (core-based trees). Aquí se calcula un solo árbol de expansión por grupo, con la raíz (el núcleo) cerca de la mitad del grupo. Para enviar un mensaje de multidifusión, un host lo envía al núcleo,

que entonces hace la multidifusión a través del árbol de expansión. Aunque este árbol no será optimo para todos los orígenes, la reducción de costos de almacenamiento de **m** árboles a un árbol por grupo representa un ahorro

3.6 Políticas de control de flujo y de congestión

Se denomina congestión a la circunstancia en la que el rendimiento de la red (o una parte de ella) se degrada debido a la presencia de demasiados paquetes. La congestión es un problema global, que se da en el nivel de red como consecuencia del tráfico agregado de varias fuentes sobre un enlace o router de baja capacidad. a diferencia de la congestión, el control de flujo es una circunstancia que sólo puede darse en conexiones punto a punto (es decir, en el nivel de enlace o en el nivel de transporte).

Hay varias causas de congestión. Las más importantes son:

- La memoria insuficiente de los conmutadores. Si de manera repentina comienzan a llegar cadenas de paquetes por tres o cuatro líneas de entrada y todas necesitan la misma línea de salida, se generará una cola. Si no hay suficiente memoria para almacenar todos los paquetes, algunos de ellos se perderán.
- Insuficiente CPU en los nodos. Puede que el nodo sea incapaz de procesar toda la información que le llega, con lo que hará que se saturen las colas (búferes de encolamiento, actualización de tablas, etc).

Algunos de los parámetros que permiten detectar la presencia de congestión pueden ser los siguientes:

- Porcentaje de paquetes descartados.
- Longitud media de las colas en las interfaces de los routers.
- Número de paquetes que dan timeout y se retransmiten (no debidos a errores).
- Retardo medio por paquete.
- Desviación media del retardo por paquete.

El **control de congestión** se ocupa de asegurar que la subred sea capaz de transportar el tráfico ofrecido. Es un asunto global, en el que interviene el comportamiento de todos los hosts, todos los enrutadores, el proceso de

almacenamiento y reenvío dentro de los enrutadores y todos los demás factores que tienden a disminuir la capacidad de transporte de la subred.

En contraste, el **control de flujo** se relaciona con el tráfico punto a punto entre un emisor dado y un receptor dado. Su tarea es asegurar que un emisor rápido no pueda transmitir datos de manera continua a una velocidad mayor que la que puede absorber el receptor. El control de congestión siempre implica una retroalimentación directa del receptor al emisor, para indicar el emisor cómo van las cosas en el otro lado.

3.6.1 Principios generales del control de congestión

Este método conduce a dividir en dos grupos todas las soluciones: de ciclo abierto y de ciclo cerrado. El primero intenta resolver el problema mediante un buen diseño, para asegurarse en primer lugar de que no ocurra. Una vez que el sistema está en funcionamiento, no se hacen correcciones a medio camino.

Las herramientas de control de ciclo abierto incluyen decidir cuándo aceptar tráfico nuevo, decidir cuándo descartar paquetes, cuáles, y tomar decisiones de calendarización en varios puntos de la red. Todas tienen en común el hecho de que toman decisiones independientemente del estado actual de la red.

En contraste, las soluciones de ciclo cerrado se basan en el concepto de un ciclo de retroalimentación. Este método tiene tres partes cuando se aplica al control de congestión:

1) Monitorear el sistema para detectar cuándo y dónde ocurren congestiones.

2) Pasar esta información a lugares en los que puedan llevarse a cabo acciones.

3) Ajustar la operación del sistema para corregir el problema.

Las principales métricas para monitorear la subred en busca de congestiones son: el porcentaje de paquetes descartados debido a falta de espacio de búfer, la longitud promedio de las colas, la cantidad de paquetes para los cuales termina el temporizador y se transmiten de nueva cuenta, el retardo promedio de los paquetes y la desviación estándar del retardo de paquete. En todos los casos, un aumento en las cifras indica un aumento en la congestión.

El segundo paso del ciclo de retroalimentación es la transferencia de información relativa a la congestión desde el punto en que se detecta hasta el punto en que se puede hacer algo al respecto. La manera más obvia es que el enrutador que detecta la congestión envíe un paquete al origen u orígenes del tráfico, anunciando el problema. La desventaja es que estos paquetes aumentan la carga en los puntos en los que hay saturación.

Otra estrategia es hacer que los hosts o enrutadores envíen de manera periódica paquetes de sondeo para preguntar explícitamente sobre la congestión. Esta información puede usarse para enrutar tráfico fuera de áreas con problemas.

3.6.2 Políticas de prevención de congestión

Estos sistemas están diseñados para reducir al mínimo la congestión desde el inicio, en lugar de permitir que ocurra y reaccionar después del hecho. Tratan de lograr su objetivo usando políticas adecuadas en varios niveles. En la Tabla 3.5 vemos diferentes políticas para las capas de enlace de datos, red y transporte que pueden afectar a la congestión.

Capa	Políticas
Transporte	* Política de retransmisión * Política de almacenamiento en caché * Política de confirmaciones de recepción * Política de control de flujo * Política de terminaciones de temporizador
Red	* Circuitos virtuales * Datagramas en la subred * Política de encolamiento y servicio de paquetes * Política de descarte de paquetes * Algoritmo de enrutamiento * Administración de tiempo de vida del paquete
Enlace de datos	* Política de retransmisiones * Política de almacenamiento en caché de paquetes fuera de orden * Política de confirmación de recepción * Política de control de flujo

Tabla 3.5. Políticas relacionadas con la congestión

De acuerdo con la figura anterior, la capa de enlace de datos se encarga de la política de retransmisiones, la cual tiene que ver con la rapidez con la que un

emisor termina de temporizar y con lo que transmite al ocurrir una terminación de temporizador. Un emisor nervioso que a veces termina de temporizar demasiado pronto y retransmite todos los paquetes pendientes usando el protocolo de retroceso impondrá una carga más pesada al sistema que un emisor calmado que usa repetición selectiva. La política de almacenamiento en caché está muy relacionada con esto. Si los receptores descartan de manera rutinaria todos los paquetes que llegan fuera de orden, posteriormente se tendrán que enviar otra vez, lo que creará una carga extra.

La política de confirmación de recepción también afecta a la congestión. Si la recepción de cada paquete se confirma de inmediato, los paquetes de confirmación de recepción generan tráfico extra.

En la capa de red, la decisión entre circuitos virtuales y datagramas afecta la congestión, ya que muchos algoritmos de control de congestión sólo funcionan con subredes de circuitos virtuales. La política de encolamiento y servicio de paquetes se refiere a que los enrutadores tengan una cola por la línea de entrada, y una o varias colas por línea de salida. También se relaciona con el orden en que se procesan los paquetes (por ejemplo, el round robin o con base en prioridades). La política de descarte es la regla que indica qué paquete se descarta cuando no hay espacio. Una buena política puede ayudar a aliviar la congestión y una mala puede hacerlo peor.

En la capa de transporte surgen los mismos problemas que en la capa de enlace de datos, pero además es más difícil la determinación del intervalo de expiración, por que el tiempo de transito a través de la red es menos predecible que el tiempo de tránsito por un cable entre dos enrutadores. Si el intervalo es demasiado corto, se enviarán paquetes extra de manera innecesaria. Si es muy largo, se reducirá la congestión, pero el tiempo de respuesta se verá afectado cada vez que se pierda un paquete.

CAPÍTULO IV

FUNDAMENTOS DE SISTEMAS, APLICACIONES WEB Y CONCEPTOS DE DISEÑO

4. Introducción

Esta es una de las etapas del software muy importante, ya que se deben tener bien definidos los requisitos, es decir establecer un entendimiento común entre el cliente y el proyecto de software respecto a los requisitos del cliente a abordar en el proyecto de software.

La gestión de requisitos implica el establecimiento y mantenimiento de un acuerdo con el cliente sobre los requisitos del proyecto de software. A este acuerdo se hace referencia como los “requisitos del sistema asignados al software”.

Es de suma importancia estudiar los principios y metodologías para el desarrollo y mantenimiento de sistemas de software. En este caso, empezaremos describiendo algunos de los principios y métodos de la ingeniería de software, a fin de obtener un software rentable, que sea fiable y fácil de utilizar.

En este capítulo encontraremos algunos de los procesos y metodologías, que por décadas han investigado los mejores esquemas, para mejorar la productividad en el desarrollo y la calidad del producto software.

Así mismo, explicaremos las herramientas más comunes para el desarrollo de nuestro proyecto. También se explicarán algunas terminologías y comparaciones entre programas, ventajas, desventajas y rentabilidad.

4.1 ¿Qué es software?

Palabra proveniente del inglés (literalmente: partes blandas o suaves), es el conjunto de programas de cómputo, procedimientos, reglas, documentación y datos asociados que forman parte de las operaciones de un sistema de computación.

Bajo esta definición, el concepto de software va más allá de los programas de cómputo en sus distintos estados: código fuente, binario o ejecutable; también su documentación, datos a procesar e información de usuario es parte del software: es decir, abarca todo lo intangible, todo lo "no físico" relacionado.

Clasificación del software

Software de sistema: Es aquel que permite que el hardware funcione. Su objetivo es desvincular adecuadamente al programador de los detalles del computador en particular que se use, aislándolo especialmente del procesamiento referido a las características internas de: memoria, discos, puertos y dispositivos de comunicaciones, impresoras, pantallas, teclados, etc. El software de sistema le procura al usuario y programador adecuadas interfases de alto nivel y utilidades de apoyo que permiten su mantenimiento. Incluye entre otros:

- Sistemas operativos.
- Controladores de dispositivo.
- Herramientas de diagnóstico.
- Herramientas de corrección y optimización.
- Servidores.
- Utilidades.

Software de programación: Es el conjunto de herramientas que permiten al programador desarrollar programas informáticos, usando diferentes alternativas y lenguajes de programación, de una manera práctica. Incluye entre otros:

- Editores de texto.
- Compiladores.
- Intérpretes.
- Enlazadores.
- Depuradores.

Entornos de Desarrollo Integrados (IDE): Agrupan las anteriores herramientas, usualmente en un entorno visual, de forma que el programador no necesite introducir múltiples comandos para compilar, interpretar, depurar, etc. Habitualmente cuentan con una avanzada interfaz gráfica de usuario (GUI).

Software de aplicación: Aquel que permite a los usuarios llevar a cabo una o varias tareas específicas, en cualquier campo de actividad susceptible de ser automatizado o asistido, con especial énfasis en los negocios. Incluye entre otros:

- Aplicaciones de control y automatización industrial.
- Aplicaciones ofimáticas.
- Software educativo.
- Software médico.
- Software de cálculo numérico.
- Software de diseño asistido (CAD).
- Software de control numérico (CAM).

4.2 ¿Qué es un proceso de software?

Un proceso de software es un conjunto de actividades y resultados asociados que producen un producto de software. Existen cuatro actividades fundamentales de procesos que son comunes para todos los procesos del software. Estos son:

1. Especificación del software donde los clientes e ingenieros definen el software a producir y las restricciones sobre su operación.
2. Desarrollo del software donde el software se diseña y programa.
3. Validación del software donde el software se valida para asegurar que es lo que el cliente requiere.
4. Evolución del software donde el software se modifica para adaptarlo a los cambios requeridos por el cliente y el mercado.

Diferentes tipos de sistemas necesitan diferentes procesos de desarrollo. Por ejemplo, el software de tiempo real en un avión tiene que ser completamente especificado antes de que empiece el desarrollo, mientras que en un sistema de comercio electrónico, la especificación y el programa normalmente son desarrollados juntos. Por o tanto, estas actividades genéricas pueden

organizarse de diferentes formas y describirse en niveles diferentes de detalle para diferentes tipos de software.

4.2.1 Modelo de procesos del software

El modelo del proceso de software es una descripción simplificada de un proceso de software que presenta una visión de ese proceso. Estos modelos pueden incluir actividades que son parte de los procesos y productos de software y el papel de las personas involucradas en la ingeniería de software. Algunos de estos tipos de modelos son:

1. **Un modelo de flujo de trabajo**. Muestra la secuencia de actividades en el proceso junto con sus entradas, salidas y dependencias, las actividades en este modelo representan acciones humanas.
2. **Un modelo de flujo de datos o de actividad**. Representa el proceso como un conjunto de actividades, cada una de las cuales realiza alguna transformación en los datos. Muestra cómo la entrada en el proceso, tal como una especificación se transforma en una salida, tal como un diseño. Pueden representar transformaciones llevadas a cabo por las personas o por las computadoras.
3. **Un modelo de rollación**. Representa los roles de las personas involucradas en el proceso de software y las actividades de las que son responsables.

La mayor parte de los modelos de procesos del software se basan en uno de los tres modelos generales o paradigmas de desarrollo de software:

1. **El enfoque en cascada.** Considera las actividades anteriores y las representa como fases de procesos separados, tales como la especificación de requerimientos, el diseño del software, la implementación, las pruebas, etc. Después de que cada etapa queda definida “se firma” y el desarrollo continúa con la siguiente etapa.
2. **Desarrollo iterativo.** En este enfoque entrelaza las actividades de especificación, desarrollo y validación. Un sistema inicial se desarrolla rápidamente a partir de especificaciones muy abstractas. Éste se refina basándose en las peticiones del cliente para producir un sistema que satisfaga las necesidades de dicho cliente. El sistema puede ser

entonces entregado. De forma alternativa, se puede reimplementar utilizando un enfoque más estructurado para producir un sistema más sólido y mantenible.

3. **Ingeniería del software basado en componentes (CBSE).** Esta técnica supone que las partes del sistema existen. El proceso de desarrollo del sistema se enfoca en la integración de estas partes más que desarrollarlas desde el principio.

4.2.2 Costos de la ingeniería de software

Los costos del proceso de software, están relacionados con diferentes actividades, como el proceso utilizado y el tipo de software que se vaya a desarrollar. Por ejemplo, el software de tiempo real requiere una validación y pruebas más extensas que los sistemas basados en web. Si se considera que el costo total del desarrollo de un sistema de software complejo es de 100 unidades de costo, la figura 4.1 muestra cómo se gastan éstas en las diferentes actividades del proceso.

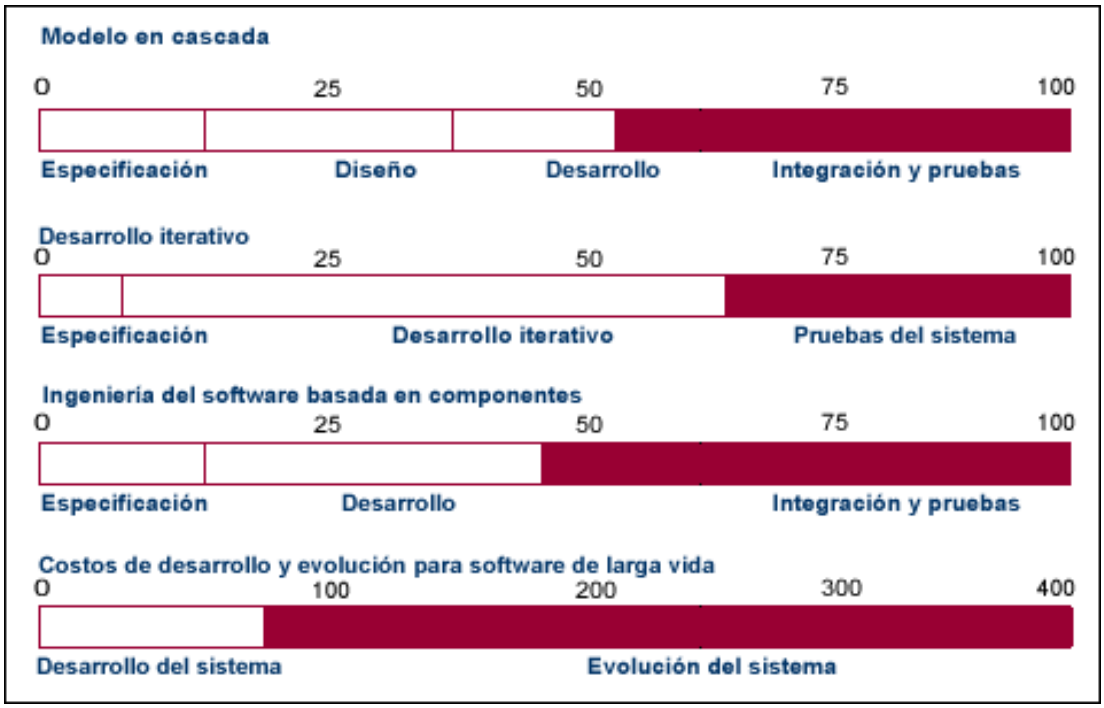


Figura 4.1. Costos del proceso de software

En el enfoque en cascada, los costos de especificación, diseño, implementación e integración se miden de forma separada. Observe que la

integración y pruebas del sistema son las actividades de desarrollo más caras. Normalmente, éste supone alrededor del 40% del costo del desarrollo total, pero para algunos sistemas críticos es probable que sea al menos el 50% de los costos de desarrollo del sistema.

Si el software se desarrolla utilizando un enfoque iterativo, no existe división entre la especificación, el diseño y el desarrollo. En este enfoque, los costos de especificación se reducen debido a que sólo se produce la especificación de alto nivel antes que el desarrollo. La especificación, el diseño, la implementación, la integración y las pruebas se llevan a cabo en paralelo dentro de una actividad de desarrollo.

La ingeniería de software basada en componentes sólo ha sido ampliamente utilizada durante un corto periodo de tiempo. En este enfoque, no hay figuras exactas para los costos de las diferentes actividades del desarrollo de software. Sin embargo, sabemos que los costos de desarrollo se reducen en relación a los costos de integración y pruebas. Los costos de integración y pruebas se incrementan por que tenemos que asegurarnos de que los componentes que utilizamos cumplen realmente su especificación y funcionan como se espera con otros componentes.

Para fines de este proyecto, se aplicó el modelo en cascada, ya que conjugó los cuatro patrones que componen este modelo. Realmente es complicado medir los costos en cuando no haya una dedicación determinada en tiempo al día, pero si podemos decir que la mayor parte de horas invertidas fue a partir del desarrollo, la integración y las pruebas.

4.2.3 Retos fundamentales que afronta la ingeniería del software

En el siglo XXI, la ingeniería del software afronta tres grandes retos fundamentales:

1) **El reto de la heterogeneidad**. Cada vez más, se requiere que los sistemas operen como sistemas distribuidos en redes que incluyen diferentes tipos de computadoras y con diferentes clases de sistemas de soporte. El reto es desarrollar técnicas para construir software confiable que sea lo suficientemente flexible para adecuarse a esta heterogeneidad.

2) **El reto de la entrega**. El reto de la entrega es reducir los tiempos de entrega para los sistemas grandes y complejos sin comprometer la calidad del sistema.
3) **El reto de la confianza**. El reto de la confianza es desarrollar técnicas que demuestren que los usuarios pueden confiar en el software.

Por supuesto, éstos no son independientes. Para llevar acabo esos cambios es necesario de nuevas herramientas y técnicas, así como formas innovadoras de combinación y uso de métodos de ingeniería del software existentes.

4.3 Arquitectura de software

La arquitectura de software de un sistema de programa o computación es la estructura de las estructuras del sistema, la cual comprende los componentes del software, las propiedades de esos componentes visibles externamente, y las relaciones entre ellos.

La arquitectura no es el software operacional. Más bien, es la representación que capacita al ingeniero del software para:

- Analizar la efectividad del diseño para la consecución de los requisitos fijados.
- Considerar las alternativas arquitectónicas en una etapa en la cual hacer cambios en el diseño es relativamente fácil.
- Reducir los riesgos asociados a la construcción del software.

Importancia de la arquitectura del software

Las razones por las que la arquitectura de software es importante son:

- Las representaciones de la arquitectura de software facilitan la comunicación entre todas las partes (partícipes) interesadas en el desarrollo de un sistema basado en computadora.
- La arquitectura destaca decisiones tempranas de diseño que tendrán un profundo impacto en todo el trabajo de ingeniería del software que sigue, y es tan importante en el éxito final del sistema como una entidad operacional.

- La arquitectura constituye un modelo relativamente pequeño e intelectualmente comprensible de cómo está estructurado el sistema y de cómo trabajan juntos sus componentes.

Al igual que otras actividades de la ingeniería del software, el diseño de datos (a veces llamado arquitectura de datos) crea un modelo de datos y/o información que se representa con un alto nivel de abstracción (la visión de datos del cliente/usuario).

4.4 Estilos y patrones arquitectónicos

El software construido para sistemas basados en computadoras también cuenta con diversos estilos arquitectónicos. Cada estilo describe una categoría del sistema que contiene:

1) Un conjunto de componentes (por ejemplo, una base de datos, módulos computacionales) que realizan una función requerida por el sistema.
2) Un conjunto de conectores que posibilitan la comunicación, la coordinación y la cooperación, entre otros componentes.
3) Restricciones que definen cómo se pueden integrar los componentes que forman el sistema.
4) Modelos semánticos que permiten al diseñador entender las propiedades globales de un sistema para analizar las propiedades globales de un sistema para analizar las propiedades conocidas de sus partes constituyentes.

Aunque durante los pasados 50 años se han creado cientos de miles de sistemas basados en computadora la gran mayoría pueden ser clasificados dentro de uno de los estilos arquitectónicos:

Arquitecturas centradas de datos. En el centro de esta arquitectura se encuentra un almacén de datos (por ejemplo, un documento o una base de datos) al que otros componentes acceder con frecuencia para actualizar, añadir, borrar o bien modificar los datos del almacén. La figura 4.3 representa un estilo típico basado en los datos. El software de cliente accede a un almacén central. En algunos casos el almacén de datos es pasivo. Esto significa que el software de cliente accede a los datos independientemente de cualquier cambio en los datos o de las acciones de otro software de cliente.

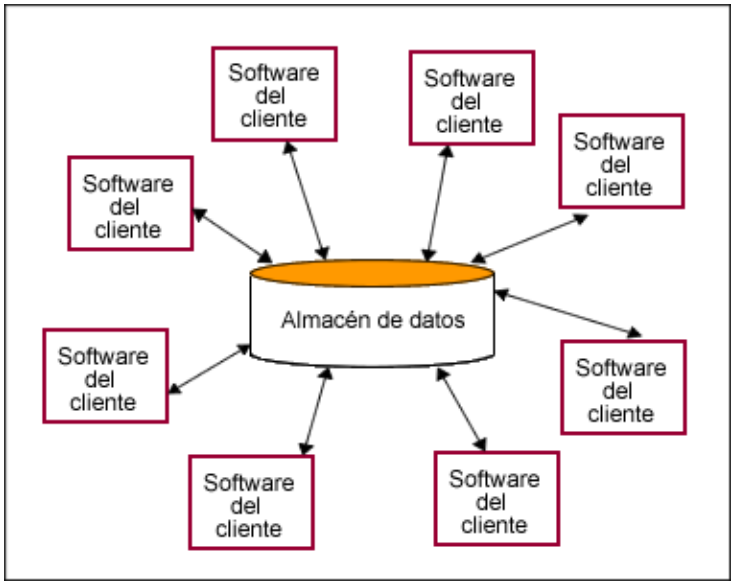


Figura 4.3 Arquitectura basada en los datos.

Arquitectura de flujo de datos. Esta arquitectura se aplica cuando los datos de entrada son transformados a través de una serie de componentes computacionales o manipulativos en los datos de salida. Un patrón tubería y filtro tiene un grupo de componentes, llamados filtros, conectados por tuberías que transmiten datos de un componente al siguiente. Cada filtro trabaja independientemente de aquellos componentes que se encuentran en el flujo de entrada o de salida; está diseñado para recibir la entrada de datos de una cierta forma y producir una salida de datos (hacia el siguiente filtro) de una forma específica. Sin embargo, el filtro no necesita conocer el trabajo de los filtros vecinos.

Si el flujo de datos degenera en una simple línea de transformadores figura 4.4 (b) se le denomina secuencial por lotes. Este patrón aplica una serie de componentes secuenciales (filtros) para transformarlos.

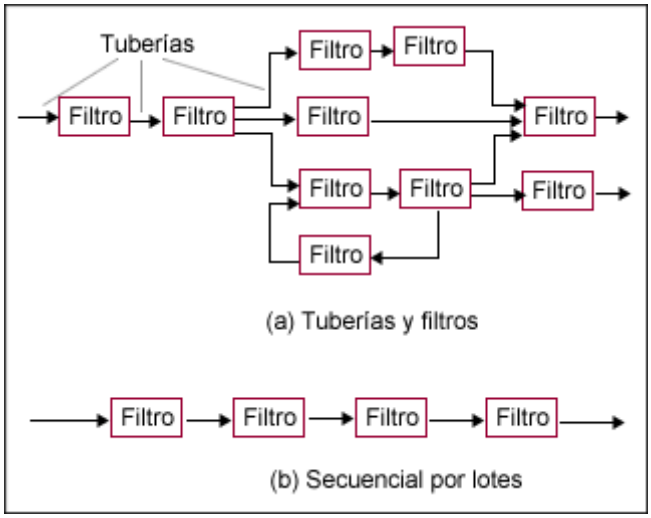


Figura 4.4 Arquitecturas de flujo de datos

Arquitecturas de llamada y de retorno. Este estilo arquitectónico permite al diseñador del software construir una estructura de programa relativamente fácil de modificar y ajustar a escala. Existen dos subestilos dentro de esta categoría:

- **Arquitecturas de programa**. Esta estructura clásica de programación descompone las funciones en una jerarquía de control donde un programa "principal" llama a un número de componentes del programa, los cuales en respuesta, pueden también llamar a otros componentes. La figura 4.5 representa una arquitectura de este tipo.

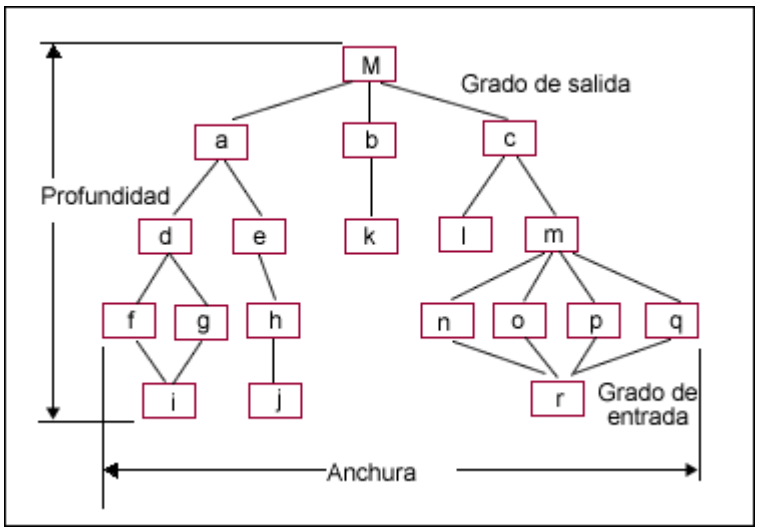
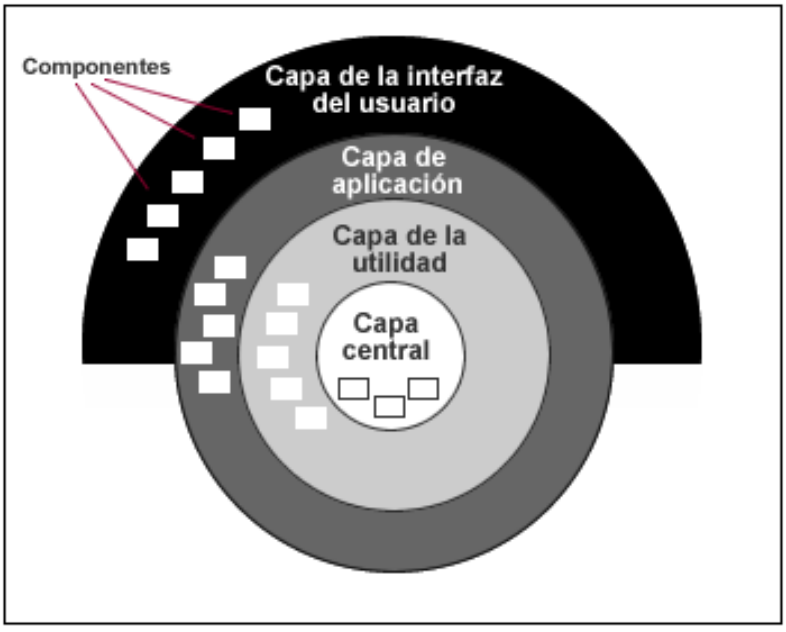


Figura 4.5. Terminologías de estructura para un estilo arquitectónico de llamada y retorno.

- **Arquitecturas de llamada de procedimiento remoto**. Los componentes de una arquitectura de programa principal/subprograma, están distribuidos entre varias computadoras en una red.

Arquitecturas orientadas a objetos. Los componentes de un sistema encapsulan los datos y las operaciones que se deben realizar para manipular los datos. La comunicación y la coordinación entre componentes se consiguen a través del paso de mensajes.

Arquitecturas estratificadas. La estructura básica de una arquitectura estratificada se representa en la figura 4.6. Se crean diferentes capas y cada una realiza operaciones que progresivamente se aproximan más al cuadro de instrucciones de la máquina. En la capa externa, los componentes sirven a las operaciones de interfaz de usuario. En la capa interna, los componentes realizan operaciones de interfaz del sistema. Las capas intermedias proporcionan servicios de utilidad y funciones del software de aplicaciones.



4.6. Arquitectura estratificada

Los estilos arquitectónicos citados anteriormente son sólo una pequeña parte de lo que dispone el diseñador de software. Una vez que la ingeniería de requisitos define las características y las restricciones del sistema que ha de

ser construido, se escoge el patrón arquitectónico (estilo) o la combinación de patrones (estilos) que mejor encajan con las características y restricciones.

4.5 Diseño de la interfaz de usuario

El diseño de la interfaz se centra en tres áreas de interés:

1) El diseño de la interfaz entre otros componentes del software.
2) El diseño de las interfaces entre el software y los otros productores y consumidores de información no humanos.
3) El diseño de la interfaz entre el hombre (esto es, el usuario) y la computadora.

Ben Shneiderman habla sobre esta categoría de diseño en el prólogo de su libro y afirma lo siguiente:

"Para muchos usuarios de sistemas de información computarizados la frustración y la ansiedad forman parte de su vida diaria. Luchan por aprender el lenguaje de órdenes y los sistemas de selección de menús que supuestamente les ayudan a realizar su trabajo. Algunas personas se encuentran con casos tan serios de shocks informáticos, terror en el Terminal o neurosis en la red, que evitan utilizar sistemas computarizados".

Theo Mandel en su libro crea tres reglas de oro para el diseño de la interfaz:

1) Dar el control al usuario
2) Reducir la carga de memoria del usuario
3) Consumir una interfaz consecuente

Estas reglas de oro forman en realidad la base para los principios del diseño de la interfaz de usuario que servirán de guía para esta actividad de diseño de software tan importante.

4.5.1 Dar el control al usuario

Mandel define una serie de principios de diseño que permiten dar control al usuario:

- Definir los modos de interacción de manera que no obligue a que el usuario realice acciones innecesarias y no deseadas.

- Tener en consideración una interacción flexible. Dado que diferentes usuarios tienen preferencias de interacción diferentes, se deberán proporcionar diferentes selecciones.
- Permitir que la interacción del usuario se pueda interrumpir y deshacer. Cuando un usuario se ve involucrado en una secuencia de acciones, deberá poder interrumpir la secuencia para hacer cualquier cosa (sin perder el trabajo que se hubiera hecho anteriormente).
- Aligerar la interacción a medida que avanza el nivel de conocimiento y permitir personalizar la interacción. El usuario a menudo se encuentra haciendo la misma secuencia de interacciones de manera repetida. Merece la pena señalar un mecanismo de "macros" que posibilite al usuario la interfaz y así facilitar la interacción.
- Ocultar al usuario ocasional los entresijos técnicos. La interfaz de usuario deberá introducir al usuario en el mundo virtual de la aplicación. El usuario no tiene que conocer el sistema operativo, las funciones de gestión de archivos, o cualquier otro secreto de la tecnología informática. Esencialmente, la interfaz no deberá requerir nunca que el usuario interactúe a un nivel interno de la máquina.
- Diseñar la interacción directa con los objetos que aparecen en la pantalla. El usuario tiene un sentimiento de control cuando manipula los objetos necesarios para llevar a cabo una tarea de forma similar a lo que ocurriría si el objeto fuera algo físico.

4.5.2 Modelos de diseño de la interfaz

El modelo de diseño de un sistema completo incorpora las representaciones del software en función de los datos, arquitectura, interfaz y procedimiento. Para construir una interfaz de usuario efectiva, todo diseño deberá comenzar por conocer los usuarios destino, así como los perfiles de edad, sexo, habilidades físicas, educación, antecedentes culturales o étnicos, motivación, objetivos y personalidad. Además de esto se pueden establecer las siguientes categorías de usuarios:

- Principiantes. En general no tienen conocimiento sintáctico, ni conocimientos semánticos de la utilización de la aplicación o del sistema.
- Usuarios esporádicos y con conocimiento. Poseen un conocimiento semántico razonable. Pero una retención baja de la información necesaria para utilizar la interfaz.
- Usuarios frecuentes y con conocimientos. Poseen el conocimiento sintáctico y semántico suficiente como para llegar al síndrome de usuario avanzado, esto es, individuos que buscan interrupciones breves y modos abreviados de interacción.

4.5.3 El proceso de diseño de la interfaz de usuario

El proceso de diseño de interfaces de usuario es iterativo y se puede representar mediante un modelo espiral. En la figura 4.7 se puede observar que el proceso de diseño de la interfaz de usuario acompaña cuatro actividades distintas del marco de trabajo:

1) Análisis y modelado de usuarios, tareas y entornos.
2) Diseño de la interfaz.
3) Implementación de la interfaz.
4) Validación de la interfaz.

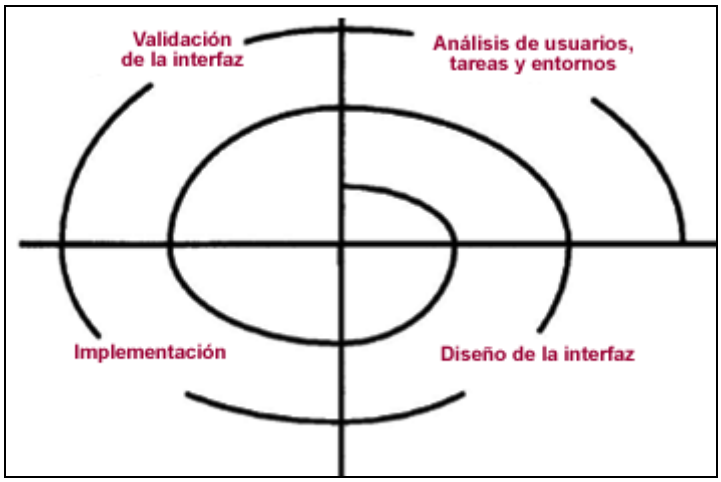


Figura 4.7 El proceso de diseño de la interfaz de usuario.

La espiral que se muestra en la figura 4.7 implica que cada una de las tareas anteriores aparecerán más de una vez, en donde a medida que se avanza por la espiral se representará la elaboración adicional de los requisitos y el diseño resultante.

El objetivo del diseño de la interfaz es definir un conjunto de objetos y acciones de interfaz (y sus representaciones en pantalla) que posibiliten al usuario llevar acabo todas las tareas definidas de forma que cumplan todos los objetivos de usabilidad definidos por el sistema.

La actividad de implementación comienza normalmente con la creación de un prototipo que permita evaluar los escenarios de utilización, a medida que avanza el proceso de diseño iterativo, y para completar la construcción de la interfaz, se puede utilizar un kit de herramientas de usuario.

La validación se centra en: (1) la habilidad de la interfaz para implementar correctamente todas las tareas del usuario, para acoplar todas las variaciones de tareas, y para archivar todos los requisitos generales del usuario; (2) el grado de facilidad de utilización de la interfaz y de aprendizaje, y (3) la aceptación de la interfaz por parte del usuario como una herramienta útil en su trabajo.

4.5.4 Herramientas de implementación

Una vez creado el modelo de diseño, se implementa como un prototipo, que los usuarios han examinado. Para acoplar este enfoque de diseño iterativo se ha desarrollado una clase extensa de herramientas diseño de interfaz y de generación de prototipos. Estas herramientas así llamadas, juego de herramientas de la interfaz de usuario o sistemas de desarrollo de la interfaz de usuario (SDIU), proporcionan componentes u objetos que facilitan la creación de ventanas, menús, interacción de dispositivos, mensajes de error, órdenes y muchos otros elementos de un entorno interactivo.

Mediante los componentes de software preestablecidos que se pueden utilizar para crear una interfaz de usuario, un SDIU proporcionará los mecanismos para:

- Gestionar los dispositivos de salida (tales como el ratón o el teclado).
- Validar la entrada del usuario.

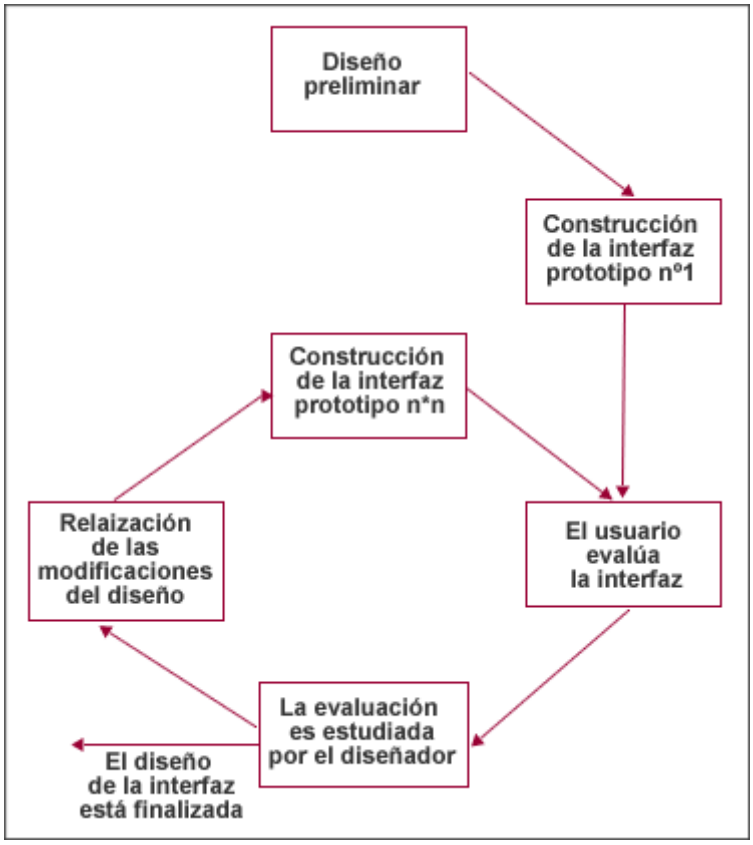
- Manipular los errores y visualizar mensajes de error.
- Proporcionar una respuesta (por ejemplo, un eco automático de entrada).
- Proporcionar ayuda e indicaciones de solicitud de entrada de órdenes.
- Manipular ventanas y campos, desplazarse por las ventanas.
- Establecer conexiones entre el software de la aplicación y la interfaz.
- Aislar la aplicación de las funciones de gestión de la interfaz.
- Permitir que el usuario personalice la interfaz.

Estas funciones se pueden implementar mediante un enfoque gráfico o basado en lenguajes.

4.5.5 Evaluación del diseño

El ciclo de evaluación continúa hasta que ya no sean necesarias más modificaciones del diseño de la interfaz (véase figura 4.8). Si se ha creado un modelo de diseño de la interfaz, durante las primeras revisiones del diseño se podrán aplicar una serie de criterios de evaluación:

1) La duración y la complejidad de la especificación que se haya escrito del sistema y de su interfaz proporcionan una indicación de la cantidad de aprendizaje que requieren los usuarios del sistema.
2) La cantidad de tareas especificadas y la cantidad medida de acciones por tarea proporcionan una indicación del tiempo y la eficacia global del sistema.
3) La cantidad de acciones, tareas y estados de sistemas indicados con el modelo de diseño indican la carga de memoria que tienen los usuarios del sistema.
4) El estilo de la interfaz, las funciones de ayuda y el protocolo de solución de errores proporcionan una indicación general de la complejidad de la interfaz y el grado de aceptación por parte del usuario.



4.8 El ciclo de evaluación de diseño de la interfaz

Existen tres principios importantes que dirigen al diseño de interfaces de usuario eficaces: (1) poner el control en manos del usuario; (2) reducir la carga de la memoria del usuario; (3) construir una interfaz consecuente. Para lograr que una interfaz se atenga a estos principios, se deberá desarrollar un proceso de diseño organizado.

4.6 Sistemas distribuidos

Los sistemas distribuidos se caracterizan por su rendimiento, compartición de recursos y tolerancia a fallos:

Rendimiento. El rendimiento de muchos tipos de sistemas distribuidos se puede incrementar añadiendo simplemente más computadoras. Los sistemas típicos en donde se puede lograr este incremento en el rendimiento son aquellos en donde las computadoras distribuidas llevan a cabo mucho proceso, y en donde la relación de comunicaciones y proceso es bajo.

Compartición de recursos. Un sistema distribuido permite a sus usuarios acceder a grandes cantidades de datos que contienen las computadoras que

componen el sistema. En lugar de tener que reproducir los datos en todas las computadoras se pueden distribuir por un pequeño número de computadoras. Un sistema distribuido también proporciona acceso a servicios especializados que quizás no requieran muy frecuentemente, y que se puedan centralizar en una computadora del sistema.

Tolerancia a fallos. Un sistema distribuido se puede diseñar de forma que tolere los fallos tanto del hardware como del software.

Los bloques básicos de construcción de un sistema distribuido son el cliente y el servidor.

4.6.1 Cliente/Servidor

La arquitectura cliente/servidor forma parte de los sistemas distribuidos. Dichos sistemas consisten en un número de computadoras que están conectadas y que llevan a cabo diferentes funciones.

En la arquitectura cliente/servidor, el software que reside en una computadora cliente –solicita servicios y/o datos de otra computadora, - servidor -.

Un servidor es una computadora que lleva a cabo un servicio que normalmente requiere mucha potencia de procesamiento.

Un cliente es una computadora que solicita los servicios que proporciona uno o más servidores y que también lleva a cabo algún tipo de procesamiento por sí mismo.

4.6.2 Servidores

Servidores de bases de datos. Los servidores de bases de datos son computadoras que almacenan grandes colecciones de datos estructurados. El servidor de bases de datos lee el código SQL, lo interpreta, y a continuación, lo visualiza en algún objeto de la interfaz hombre-máquina tal como una caja de texto. El servidor de bases de datos lleva a cabo todo el procesamiento, donde el cliente lleva a cabo todos los procesos de extraer alguna consulta de algún objeto de entrada, tal como un campo de texto, enviar la consulta y visualizar la respuesta del servidor de bases de datos en algún objeto de salida, tal como un cuadro de desplazamiento.

Servidores web. Un servidor web es un programa que se ejecuta en un servidor (máquina que almacena y administra los sitios web) con el propósito de atender y responder a las diferentes peticiones de los navegadores de los usuarios, proporcionando los recursos que soliciten usando el protocolo HTTP (Hiper Text Transfer Protocol, Protocolo de Transferencia de Hipertexto) o el protocolo HTTPS (Secure Hyper Text Transfer Protocol, Protocolo Seguro de Transferencia de Hipertexto).

Un servidor web básico cuenta con un esquema de funcionamiento muy simple basado en ejecutar infinitamente el siguiente ciclo:

1) Espera peticiones en el puerto TCP indicado (el estándar por defecto para HTTP es el 80).
2) Recibe una petición.
3) Busca el recurso.
4) Envía el recurso utilizando la misma conexión por la que recibió petición.
5) Vuelve al segundo punto.

Un servidor Web que siga el esquema anterior cumplirá todos los requisitos básicos de los servidores HTTP, aunque sólo podrá mostrar archivos estáticos. A partir del anterior esquema se han diseñado y desarrollado todos los servidores de HTTP que existen, variando sólo el tipo de peticiones (páginas estáticas, CGIs, Servlets, etc.) que pueden atender peticiones, en función de que sean o no sean multi-proceso o multi-hilo, etc.

Todos los servidores web deben incluir, al menos, la capacidad para montar los archivos estáticos que se encuentren en alguna parte del disco. Un requisito básico es la capacidad de especificar qué parte del disco se servirá. No resulta recomendable que el programa servidor obligue a usar un directorio concreto, aunque sí puede tener uno por defecto. Algunos servidores web permiten también especificar directivas de seguridad (quién puede acceder a los recursos), mientras que otros hacen posible la especificación de los archivos que se deben considerar como índice del directorio.

Uno de los aspectos fundamentales del servidor web que se elija es el nivel de soporte que ofrece para mostrar contenido dinámico. La mayor parte de los servidores web ofrecen soporte para CGI (se debe recordar que los CGI son el método más antiguo y sencillo para generar contenido dinámico). Otros

muchos ofrecen soporte para algunos lenguajes de programación (normalmente lenguajes interpretados) como PHP, JSP, ASP, etc. Es muy recomendable que el servidor web que vayamos a utilizar proporcione soporte para algunos de estos lenguajes, especialmente PHP, sin tener en cuenta JSP, que normalmente requerirá un software externo para funcionar (como un contenedor de Servlets). La oferta es muy amplia, pero antes de elegir un lenguaje de programación de servidor se debe plantear si se desea un lenguaje muy estándar para que la aplicación no dependa de un servidor web o una arquitectura concreta o si, al contrario, la portabilidad no es prioritaria y sí lo es alguna otra prestación concreta que pueda ofrecer algún lenguaje de programación concreto.

4.6.3 Aplicaciones Multinivel

Al hablar de desarrollo de aplicaciones web resulta adecuado presentarlas dentro de aplicaciones multinivel (véase figura 4.9). Los sistemas típicos cliente/servidor pertenecen a la categoría de las aplicaciones de dos niveles. La aplicación reside en el cliente mientras que la base de datos se encuentra en el servidor. En este tipo de aplicaciones el peso del cálculo recae en el cliente, mientras que el servidor hace la parte menos pesada. Además, está el problema de la actualización y el mantenimiento de las aplicaciones, ya que las modificaciones a la misma han de ser trasladada a todos los clientes.

Para solucionar estos problemas se ha desarrollado el concepto de arquitecturas de tres niveles: interfaz de presentación, lógica de la aplicación y los datos. La capa intermedia es el código que el usuario invoca para recuperar los datos deseados. La capa de presentación recibe los datos y los formatea para mostrarlos adecuadamente. Esta división entre la capa de presentación y la de la lógica permite una gran flexibilidad a la hora de construir aplicaciones, ya que se pueden tener múltiples interfaces

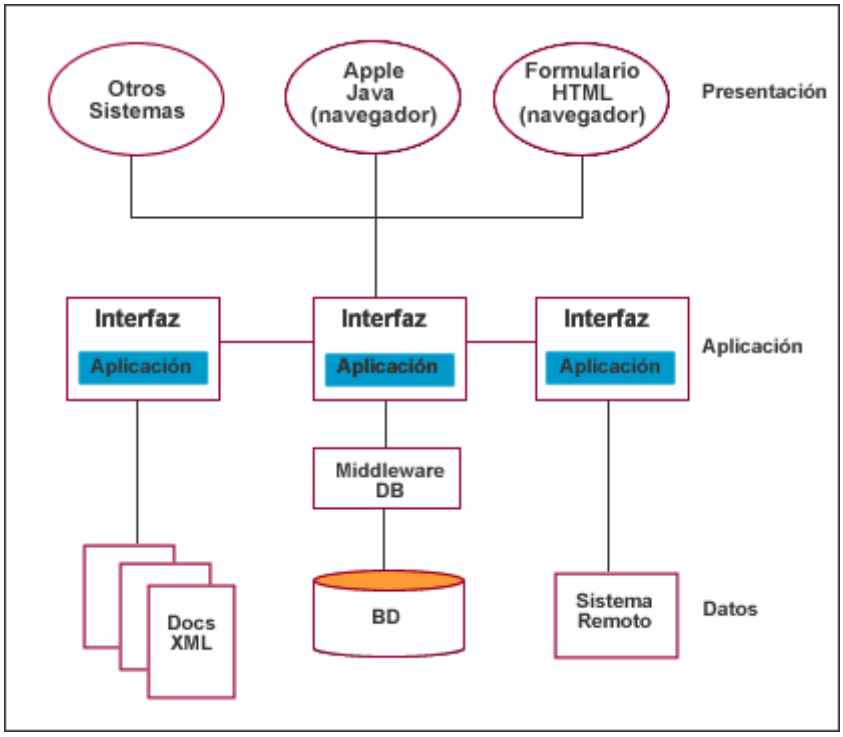


Figura 4.9 Arquitectura de aplicaciones Web representada en esquema multinivel

La arquitectura de las aplicaciones web suele presentar un esquema de tres niveles (véase figura 4.10). El primer nivel consiste en la capa de presentación que incluye no solo el navegador, sino también el servidor web que es el responsable de dar a los datos un formato adecuado. El segundo nivel está referido habitualmente a algún tipo de programa o script. Finalmente, el tercer nivel proporciona al segundo los datos necesarios para su ejecución.

Una aplicación web típica recogerá datos del usuario (primer nivel), los enviará al servidor, que ejecutará un programa (segundo y tercer nivel) y cuyo resultado será formateado y presentado al usuario en el navegador (primer nivel otra vez).

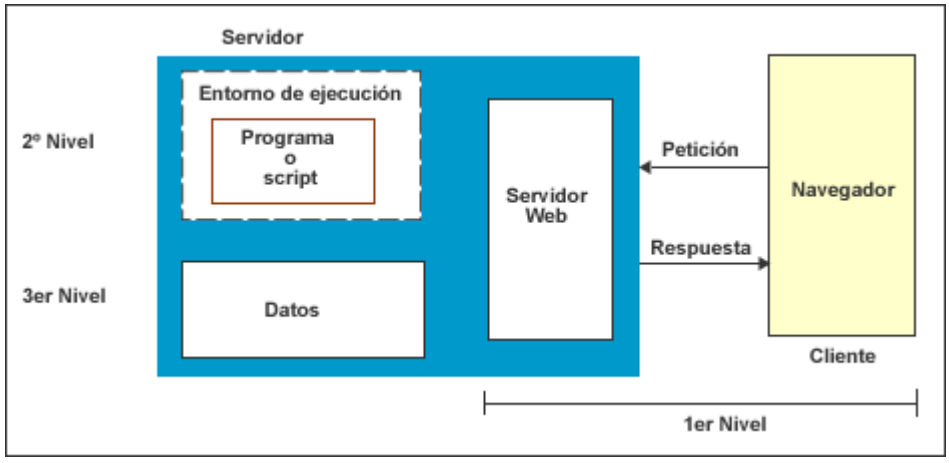


Figura 4.10 Arquitectura de aplicaciones Web representada en esquema de tres.

4.7 Fundamentos de Diseño Web

4.7.1 Elementos que componen una página Web

Una página web o documento HTML puede contener texto, imágenes, animaciones, audio, video, mapas de imagen, JavaScript, frames o marcos, enlaces, formularios, tablas, listas, etc. Pero los elementos más comunes que aparecen en las páginas Web, son el texto y las imágenes.

Texto

Cualquier palabra en la pantalla, desde los encabezados, etiquetas, títulos, y documentos enteros. El texto siempre va a ser un contenido y parte fundamental en una página o sitio Web.

Imágenes

Los archivos de imágenes pueden ser de una gran variedad de tipos; sin embargo, para poder visualizarlos en los navegadores, hay que convertirlos a formatos estándar.

Los formatos estándar que soportan los navegadores son: jpg, gif y png. Las imágenes pueden ser ilustraciones, gráficos, botones, y fotografías, que servirán de referencia y atracción visual en las páginas Web.

Animaciones

La animación en una página web permite dar vida y movimiento a los elementos que la conforman como texto, botones, ventanas, etc.

Las animaciones funcionan con la sucesión de fotogramas que se muestra uno tras otro, dando la sensación de movimiento.

Los formatos de archivos en animaciones que se pueden utilizar en el WWW son gif y swf.

Enlaces

Los enlaces (link, liga, hipervínculo), funciona como conexiones a otros documentos HTML dentro del mismo sitio, a direcciones electrónicas URL, a archivos multimedia y a enlaces dentro del mismo documento.

Mapa de imagen

Un mapa de imagen es un archivo normal de imagen, en el que se definen zonas particulares que tienen un enlace con un documento. Estas zonas vienen definidas por unas coordenadas y pueden tener forma rectangular, elíptica o poligonal, con las dimensiones que se especifiquen. Los mapas de imagen también pueden enlazar a otros documentos.

Separadores horizontales

Sirven para organizar de manera gráfica y delimitar el texto en la página web, esto con la finalidad de separar el contenido de ciertos temas dentro de la organización de la página.

Tablas

La elaboración de tablas en la creación de páginas web, además de frecuente, es muy importante ya que las tablas permiten distribuir y diagramar la información textual y gráfica, logrando con ello una composición coherente de la página. Una tabla en HTML se compone de renglones, columnas y celdas.

Frames o Marcos

Los frames o marcos sirven para dividir la pantalla en renglones y columnas. Según son las necesidades del diseño y de función que se requieran cada uno de los marcos o frames sirven para abrir en ellos otras páginas web, dentro de la misma pantalla.

JavaScript

Es el lenguaje orientado a objetos para el desarrollo de aplicaciones cliente – servidor en Internet. El código fuente de un programa escrito en JavaScript, se incluye en el mismo documento HTML. Con JavaScript se pueden hacer programas que se ejecuten en el propio cliente. De esta forma es posible, por ejemplo, realizar programas que comprueben la información de un formulario escrito en HTML sin necesidad de alguna conexión de red, además el empleo de JavaScript ofrece mayor interactividad en las páginas web.

4.8 Herramientas de programación

Dada la importancia y popularidad que adquiría la Web con el paso del tiempo por encima de otros medios de comunicación, su interacción con los usuarios también aumentaba. Debido a la constante necesidad de desarrollar páginas web que no solo se limitaran a mostrar información, fue necesario buscar alternativas a las tecnologías del lado del cliente (como HTML y JavaScript). Así fue como nacieron tecnologías como PHP,ASP,JSP, etc. A continuación se explican las características esenciales de las tecnologías que se utilizarán en el desarrollo del sistema.

4.8.1 Tecnología PHP

PHP es un acrónimo recursivo que significa PHP Hypertext Pre-processor (inicialmente PHP Tools, o, Personal Home Page Tools). Fue creado originalmente por Rasmus Lerdof en 1994.

PHP es un lenguaje de programación interpretado, diseñado originalmente para la creación de páginas web dinámicas. Es usado principalmente en interpretación del lado del servidor (server-side scripting) pero actualmente puede ser utilizado desde una interfaz de línea de comandos o en la creación

de otros tipos de programas incluyendo aplicaciones con interfaz gráfica usando las bibliotecas Qt o GTK+.

PHP es un lenguaje interpretado de propósito general ampliamente usado y que está diseñado especialmente para desarrollo web y puede ser embebido dentro de código HTML. Generalmente se ejecuta en un servidor web, tomando el código en PHP como su entrada y creando páginas web como salida. Puede ser desplegado en la mayoría de los servidores web y en casi todos los sistemas operativos y plataformas sin costo alguno. PHP se encuentra instalado en más de 20 millones de sitios web y en un millón de servidores.

Los principales usos del PHP son los siguientes:

- Programación de páginas web dinámicas, habitualmente en combinación con el motor de base de datos MySQL, aunque cuenta con soporte nativo para otros motores, incluyendo el estándar ODBC, lo que amplía en gran medida sus posibilidades de conexión.
- Programación en consola, al estilo de Perl o Shell scripting.
- Creación de aplicaciones gráficas independientes del navegador, por medio de la combinación de PHP y Qt/GTK+, lo que permite desarrollar aplicaciones de escritorio en los sistemas operativos en los que está soportado.

Ventajas

- Es un lenguaje multiplataforma.
- Capacidad de conexión con la mayoría de los manejadores de base de datos que se utilizan en la actualidad, destaca su conectividad con MySQL.
- Capacidad de expandir su potencial utilizando la enorme cantidad de módulos (llamadas extensiones).
- Posee una amplia documentación en su página oficial, entre la cual se destaca que todas las funciones del sistema están explicadas y ejemplificadas en un único archivo de ayuda.
- Es libre, por lo que se presenta como una alternativa de fácil acceso para todos.
- Permite las técnicas de programación orientada a objetos.

- Biblioteca nativa de funciones sumamente amplia e incluida.
- No requiere definición de tipos de variables.
- Tiene manejo de excepciones.

Desventajas

- No posee una abstracción de base de datos estándar, sino bibliotecas especializadas para cada motor (a veces más de una para el mismo motor).
- No posee adecuado manejo de internacionalización, unicode, etc.
- Por su diseño dinámico no puede ser compilado y es muy difícil de optimizar.
- Por sus características promueve la creación de código desordenado y complejo de mantener.
- Está diseñado especialmente para un modo de hacer aplicaciones web que es ampliamente considerado problemático y obsoleto (mezclar el código con la creación de la página web).

4.8.2 Hojas de estilo en cascada (Cascading Style Sheets, CSS).

Las hojas de estilo en cascada (Cascading Style Sheets, CSS) son un lenguaje formal usado para definir la presentación de un documento estructurado escrito en HTML o XML (y por extensión en XHTML). El W3C (World Wide Web Consortium) es el encargado de formular la especificación de las hojas de estilo que servirán de estándar para los agentes de usuario o navegadores.

La idea que se encuentra detrás del desarrollo de CSS es separar la estructura de un documento de su presentación.

Los tres tipos de estilos

CSS proporciona tres caminos diferentes para aplicar las reglas de estilo a una página Web:

1. **Una hoja de estilo externa**, que es una hoja de estilo que está almacenada en un archivo diferente al archivo donde se almacena el código HTML de la página Web. Esta es la manera de programar más potente, porque separa completamente las reglas de formateo para la página HTML de la estructura básica de la página.

2. Una hoja de estilo interna, que es una hoja de estilo que está incrustada dentro de un documento HTML. (Va a la derecha dentro del elemento <head>). De esta manera se obtiene el beneficio de separar la información del estilo, del código HTML propiamente dicho. Se puede optar por copiar la hoja de estilo incrustada de una página a otra, (esta posibilidad es difícil de ejecutar si se desea para guardar las copias sincronizadas). En general, la única vez que se usa una hoja de estilo interna, es cuando se quiere proporcionar alguna característica a una página Web en un simple fichero, por ejemplo, si se está enviando algo a la página web.

3. Un estilo en línea, que es un método para insertar el lenguaje de estilo de página, directamente, dentro de una etiqueta HTML. Esta manera de proceder no es excesivamente adecuada. Al incrustar el formateo dentro del documento de la página Web la descripción de la página, a nivel de código se convierte en una tarea larga, tediosa y poco elegante de resolver el problema de la programación de la página. Este modo de trabajo se podría usar de manera ocasional si se pretende aplicar un formateo con prisa, al vuelo. No es todo lo claro, o estructurado, que debería ser, pero funciona.

Las ventajas de utilizar CSS (u otro lenguaje de estilo) son:

- Control centralizado de la presentación de un sitio web completo con lo que se agiliza de forma considerable la actualización del mismo.
- Los navegadores permiten a los usuarios especificar su propia hoja de estilo local que será aplicada a un sitio web, con lo que aumenta considerablemente la accesibilidad. Por ejemplo, personas con deficiencias visuales pueden configurar su propia hoja de estilo para aumentar el tamaño del texto o remarcar más los enlaces.
- Una página puede disponer de diferentes hojas de estilo según el dispositivo que la muestre o incluso a elección del usuario. Por ejemplo, para ser impresa, mostrada en un dispositivo móvil, o ser "leída" por un sintetizador de voz.
- El documento HTML en sí mismo es más claro de entender y se consigue reducir considerablemente su tamaño (siempre y cuando no se utilice estilo en línea).

4.8.3 Lenguaje HTML

El lenguaje HTML HyperText Markup Language (Lenguaje de Marcado de Hipertexto) es el lenguaje de marcado predominante para la construcción de páginas web. Es usado para describir la estructura y el contenido en forma de texto, así como para complementar el texto con objetos tales como imágenes. HTML se escribe en forma de "etiquetas", rodeadas por corchetes angulares (<,>). HTML también puede describir, hasta un cierto punto, la apariencia de un documento, y puede incluir un script (por ejemplo Javascript), el cual puede afectar el comportamiento de navegadores web y otros procesadores de HTML. Los elementos son la estructura básica de HTML. Los elementos tienen dos propiedades básicas: atributos y contenido. Cada atributo y contenido tiene ciertas restricciones para que se considere válido al documento HTML.

4.8.4 Fireworks

Es una aplicación en forma de estudio (basada por supuesto en la forma de estudio de Adobe Flash®) pero con más parecido a un taller destinado para el manejo híbrido de gráficos vectoriales con Gráficos en mapa de bits y que ofrece un ambiente eficiente para la creación rápida de prototipos de sitios Web e interfaces de usuario como para la creación y Optimización de Imágenes para web. Originalmente fue desarrollado por Macromedia, compañía que fue comprada en 2005 por Adobe Systems. Fireworks está enfocado en la creación y edición de gráficos para internet. Está diseñado para integrarse con otros productos de Adobe, como Dreamweaver y Flash.

Las ventajas de utilizar la nueva versión de Fireworks CS3.

- Compatibilidad con los nuevos SO Mac y Windows®
- Integración con Adobe Photoshop® e Illustrator®
- Organización jerárquica de las capas
- Escala inteligente
- Compatibilidad con varias páginas
- Prototipos de diseños de RIA
- Activos personalizables
- Modos de fusión Photoshop
- Integración con Adobe Flash® y Dreamweaver®

4.8.5 Flash

Flash es la tecnología más comúnmente utilizada en el web que permite la creación de animaciones vectoriales. El interés en el uso de gráficos vectoriales es que éstos permiten llevar a cabo animaciones de poco peso, es decir, que tardan poco tiempo en ser cargadas por el navegador.

Existen dos tipos de gráficos:

Los gráficos vectoriales, en los cuales una imagen es representada a partir de líneas (o vectores) que poseen determinadas propiedades (color, grosor, etc.). La calidad de este tipo de gráficos no depende del zoom o del tipo de resolución con el cual se esté mirando el gráfico. Por mucho que nos acerquemos, el gráfico no se pixeliza, ya que el ordenador traza automáticamente las líneas para ese nivel de acercamiento.

Las imágenes en mapa de bits. Este tipo de gráficos se asemejan a una especie de cuadrícula en la cual cada uno de los cuadrados (píxeles) muestra un color determinado. La información de estos gráficos es guardada individualmente para cada píxel y es definida por las coordenadas y color de dicho píxel. Este tipo de gráficos son dependientes de la variación del tamaño y resolución, pudiendo perder calidad al modificar sucesivamente sus dimensiones.

Así, Flash se sirve de las posibilidades que ofrece el trabajar con gráficos vectoriales, fácilmente redimensionables y alterables por medio de funciones, así que de un almacenamiento inteligente de las imágenes y sonidos empleados en sus animaciones por medio de bibliotecas, para optimizar el tamaño de los archivos que contienen las animaciones.

Esta optimización del espacio que ocupan las animaciones, combinada con la posibilidad de cargar la animación al mismo tiempo que ésta se muestra en el navegador (técnica denominada streaming), permite aportar elementos visuales que dan vida a una web sin que para ello el tiempo de carga de la página se prolongue hasta límites insoportables por el visitante.

Además de este aspecto meramente estético, Flash introduce en su entorno la posibilidad de interaccionar con el usuario. Para ello, Flash invoca un lenguaje de programación llamado Action Script. Orientado a objetos, este lenguaje tiene claras influencias del Javascript y permite, entre otras muchas cosas, gestionar

el relleno de formularios, ejecutar distintas partes de una animación en función de eventos producidos por el usuario, saltar a otras páginas, etc.

CAPÍTULO V

DISEÑO Y DESARROLLO DEL SISTEMA

5.1 Panorama general del proyecto planteado

El proyecto en forma general está compuesto de tres componentes (véase figura 5.1) de los cuales describo a cada uno en forma particular.

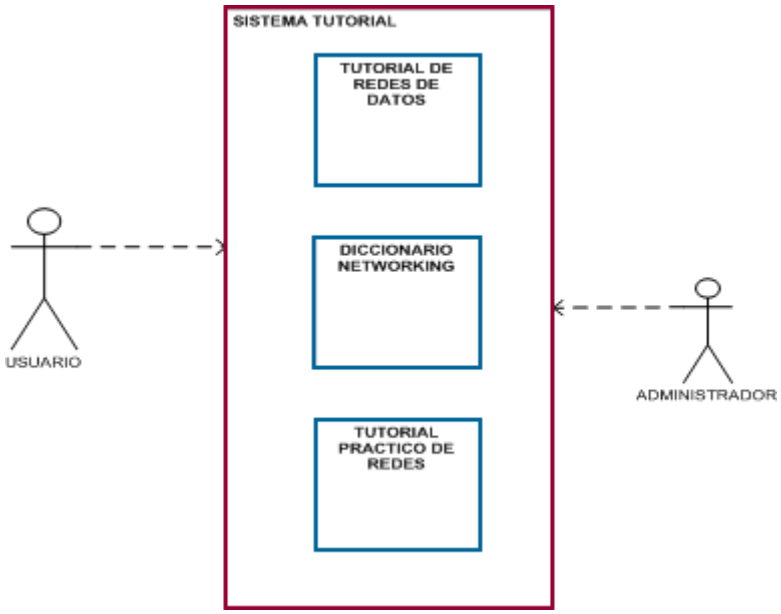


Figura 5.1. Estructura general del sistema tutorial.

Es importante tomar en cuenta el factor de la administración, ya que el sistema deberá de estar en constante mantenimiento y actualización, por el factor académico.

La idea principal del sistema tutorial de redes de datos, es desarrollarlo con hojas de estilo en cascada (CSS), además de enriquecerlo con imágenes y animaciones con movimiento. Como se menciono en el objetivo, este sistema contendrá la mayor parte de material académico acerca de la materia de redes de datos.

El diccionario networking deberá cumplir con el objetivo de contar con una base de datos que contenga el significado de los términos más usuales en la materia. Por tal motivo es necesario hacer uso de un programa que pueda ser usado a la par con un manejador de base de datos. En este caso, el servidor del laboratorio de redes, cuenta con el programa PHP y Mysql en plataforma Linux Enterprise , lo cual es una excelente noticia, para iniciar con el desarrollo.

Por último el desarrollo del tutorial práctico de redes estará conformado por ejemplos prácticos en la materia. Se ha decidido desarrollarlo con hojas de estilo en cascada (CSS), figuras y animaciones.

5.2 Diseño del tutorial de redes de datos

Como primer paso para elaborar el diseño de este tutorial, fue necesario recabar información de fuentes confiables de libros, revistas y sitios de Internet. La información recabada corresponde en su totalidad al temario de redes de datos que imparte actualmente la Facultad de Ingeniería.

Posteriormente, se analizaron los requerimientos del tutorial y se hizo un prototipo de la página Web, para ello se consideró el tipo de navegación, la distribución de la información, el color, la tipografía, el fondo, la ubicación de las imágenes, el tipo de animaciones de acuerdo al tema (o solo informativas), la ubicación de los botones, el índice general y el índice por capitulado.

Respetando el objetivo que nos planteamos al inicio del proyecto, en donde enfatizamos la posibilidad de que el sistema tutorial de redes, fuera totalmente portable, por eso se tomo la decisión de trabajar con hojas de estilo en cascada (CSS- Casscading Style Sheets), lo cual tiene muchas ventajas, entre las cuales están:

- El control centralizado de la presentación de un sitio web completo con lo que se agiliza de forma considerable la actualización del mismo.
- Hay un incremento en la accesibilidad, por ejemplo, las personas con deficiencias visuales pueden configurar su propia hoja de estilo para aumentar el tamaño del texto o remarcar los enlaces.
- Una página puede disponer de diferentes hojas de estilo según el dispositivo que la muestre o incluso a elección del usuario (impresora, dispositivo móvil, etc.).
- El documento HTML es más claro de entender y se consigue reducir considerablemente su tamaño (siempre y cuando no se utilice estilo en línea).

5.2.1 Distribución de Paquetes

El diagrama de paquetes permite analizar y organizar los elementos en grupos. Los elementos estructurales y los elementos de comportamiento pueden agruparse en paquetes. Se visualizan como carpetas (véase figura 5.2).

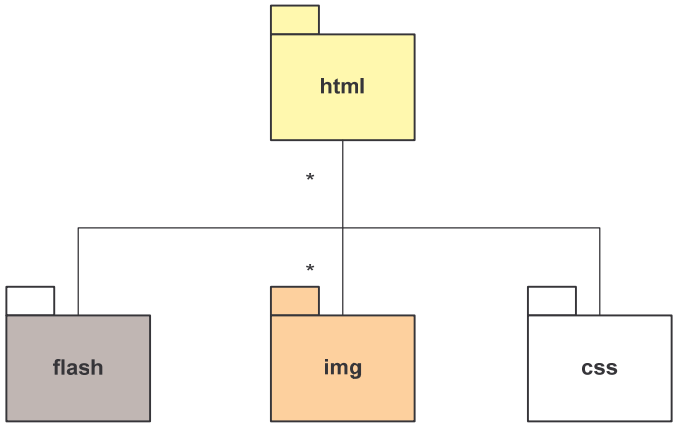


Figura 5.2. Diagrama de paquetes

El archivo CSS, es un archivo único que se generalizó para todo el tutorial. Así que lo vamos a encontrar en la raíz junto con el index. Por tal motivo, cualquier modificación al código del archivo (CSS) afectará a todas las páginas del tutorial.

La siguiente figura 5.3 muestra el orden de las carpetas, para su mejor administración y localización de las páginas web, animaciones e imágenes.

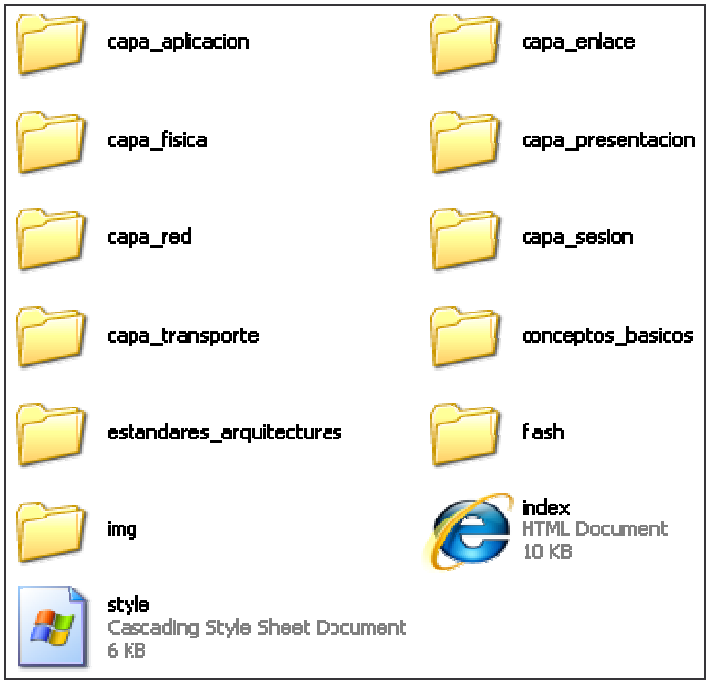


Figura 5.3. Distribución de carpetas

Como podemos observar, en la figura anterior, la hoja de estilo (style) es la que afecta directamente a todas las carpetas que contienen páginas web, esto con el fin de mantener el mismo orden.

El archivo index, es aquella página principal que encabeza a todo el grupo de carpetas, desde este punto podemos enlazarnos a cualquier título, subtítulo o a cualquiera de los capítulos que se desarrollaron.

La carpeta de img, contiene las imágenes generales del tutorial, como botones (de siguiente y atrás), imágenes del header, footer y algunas imágenes que complementan la presentación del tutorial.

La carpeta de flash, sólo contendrá la animación del index, ya que cada carpeta de cada tema tendrá su carpeta única de animaciones.

La siguiente figura 5.4 muestra la estructura interna de cada carpeta (capítulo), desglosada de cada subcapítulo que se muestra en el índice. La carpeta de flash incluye una animación por cada página. La carpeta de img incluye aquellas imágenes con las que se enriqueció de información al tutorial.

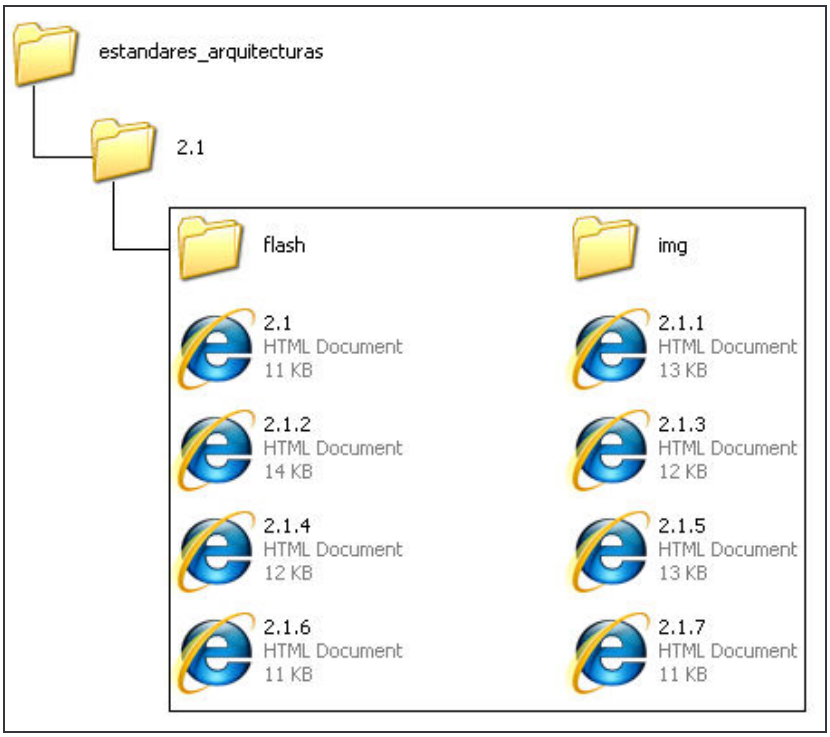


Figura 5.4. Estructura interna de los capítulos.

5.2.2 Pantalla principal. Descripción general.

Una de las partes importantes del sistema, es la distribución de todos los elementos que contendrá la pantalla, es decir deberá de ser amigable y fácil de utilizar.

A continuación se muestra un panorama general del proyecto que se desea desarrollar (véase figura 5.5).

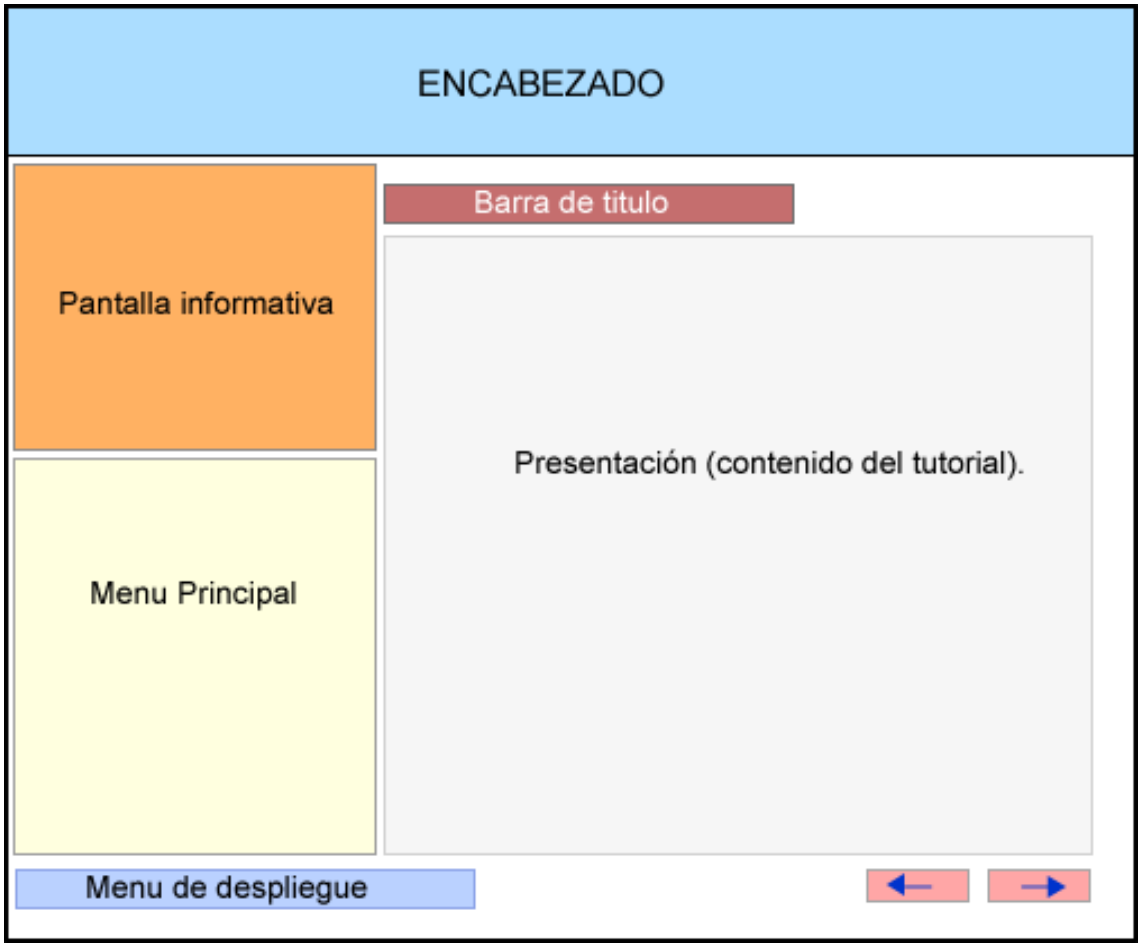


Figura 5.5. Prediseño del tutorial de redes de datos.

1. Encabezado. Se muestra el logo de la Universidad Nacional Autónoma de México en color blanco y fondo azul, en la parte superior derecha. En la parte izquierda solo se indica el nombre de la asignatura de la cual se desarrolló el tutorial.
2. Menú principal. Este menú está representado con letras de color rojo ubicado en la parte inferior izquierda. Este menú nos llevará a cada uno de los temas o subtemas que se muestran ahí por medio de ligas (links).
3. Menú de despliegue. Este menú ubicado en la parte del footer (inferior izquierda), solo muestra los nueve capítulos que abarca el tutorial. Con esta opción podemos desplazarnos al capítulo que deseemos ir.
4. Pantalla informativa. Esta pantalla ubicada en la parte superior izquierda, muestra información general con respecto a las redes o en otras ocasiones muestra algún ejemplo con movimiento (flash) pudiendo

detener cuando se desee y continuar con el botón que tiene esta pantalla.

5. Presentación. En está área se muestra el contenido del tutorial, así como imágenes planas, para no alterar la lectura del usuario. Así mismo cuenta con ligas (links) que nos llevan a las fuentes originales para ampliar la información.
6. Botones. Los botones (de atrás o siguiente) están ubicados en la parte inferior, derecha. Este tipo de botones tienen la peculiaridad de hacer el enlace desde el index, hasta el último capítulo, además de pasar por cada uno de los temas, subtemas e incisos en su caso.
7. Barra de título. En esta barra de color rojo, con letras del nombre del título en blanco, describe el número y título del capítulo. Está ubicada por encima del área de presentación.

A continuación se muestra la pantalla definida y terminada con las herramientas propuestas anteriormente. Es muy agradable la combinación de colores y tonos en las distintas partes del tutorial.

Para llegar a este desarrollo final fue necesario la opinión de varios alumnos y maestros, lo cual fue muy útil para la combinación de colores y la distribución de la información (véase figura 5.6).

Este sistema contará con un script que llenara la pantalla completamente con el explorador, dejando a la vista del usuario sólo el tutorial. Para obtener una calidad de imagen es necesario tener una resolución de pantalla de 1024 x 728 píxeles.

Figura 5.6. Pantalla definitiva del tutorial de redes de datos

5.3 Diseño de la parte lógica del Diccionario Networking

La idea fundamental del diccionario networking, es la de resolver dudas sobre conceptos o siglas que comúnmente encontramos en el área de redes.

El primer paso fue, la concentración de definiciones y términos más usuales relacionados a redes y algunos conceptos básicos de uso diario sobre seguridad informática.

Para el diccionario Networking es necesario contar con una base de datos y por lo tanto con un lenguaje de programación amigable en su caso PHP (Personal Home Page Tools). Esto con el fin de mantener actualizado el diccionario con una mayor rapidez, así como para hacer modificaciones al sistema. La base de datos a utilizar será Mysql.

A continuación se muestra el diagrama que se utilizó para la realización del diccionario (véase figura 5.7). En él se puede observar la parte lógica y la parte física del sistema.

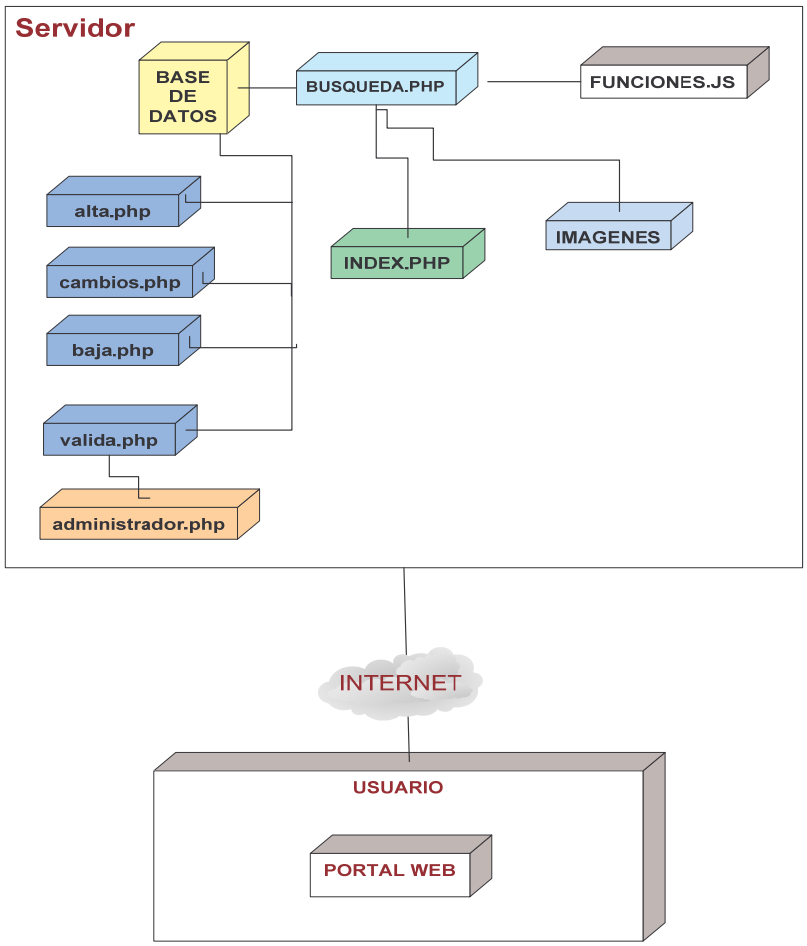


Figura 5.7. Diagrama de distribución

5.3.1 Diseño de la Base de datos

Para este proyecto, decidí utilizar como base de datos Mysql, debido a sus características que guarda a la par con PHP. La base de datos se llamará artículo y contendrá dos tablas llamadas diccionario y usuarios. A continuación se muestran los campos que conforman cada una de las tablas (véase figura 5.8):

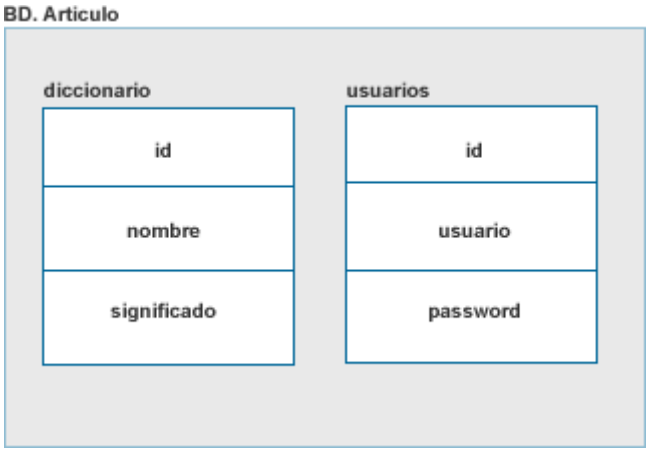


Figura 5.8. Diagrama de Base de Datos

La tabla diccionario se conforma de tres campos, id, nombre y significado. El campo id es numérico, es decir sólo aceptará números en su campo de no más de diez dígitos (véase figura 5.10).

El campo nombre, es de tipo text, es decir aceptara todo tipo de caracteres y números. Por último, el campo significado es de tipo text, al igual que el anterior.

Por otra parte, la tabla usuarios está conformada por 3 campos, el id, usuario y password; es decir aquí es donde se van a validar los usuarios para la administración del sistema (véase figura 5.11).

En el campo id, va a servir para llevar un control de cuantos administradores hay en el sistema, por eso solo lo limitaremos a 2 dígitos. Los campos nombre y usuario son de tipo text, así que aceptara todo tipo de caracteres.

A continuación se muestra la base de datos artículo, con un manejador de base de datos llamado phpmyadmin (véase figura 5.9).

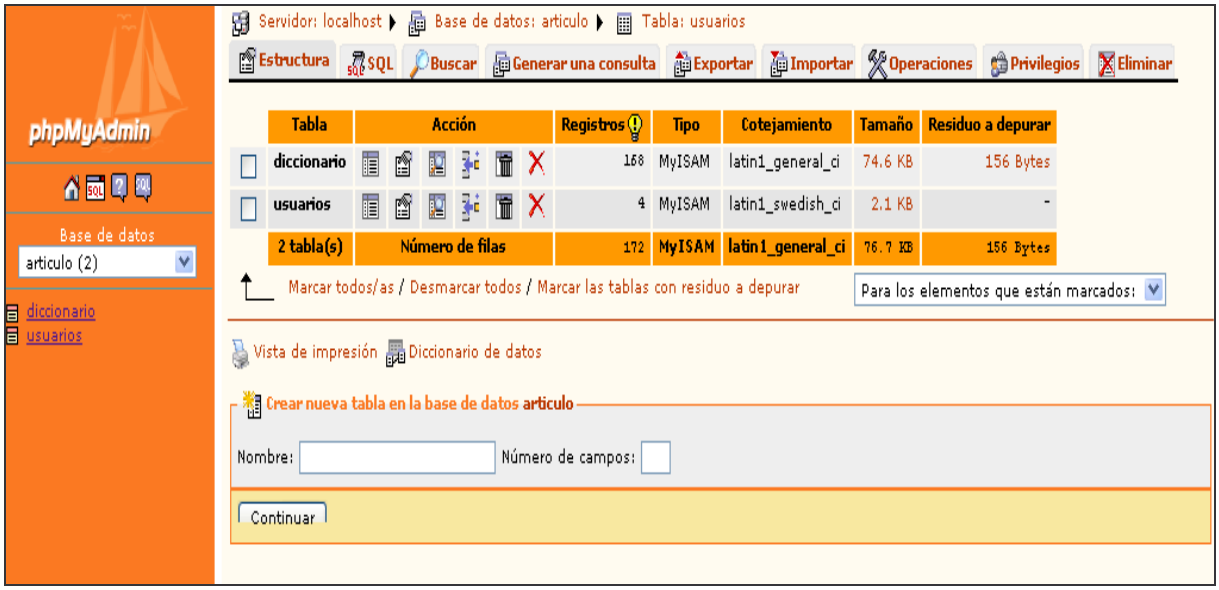


Figura 5.9. Vista de la base de datos articulo

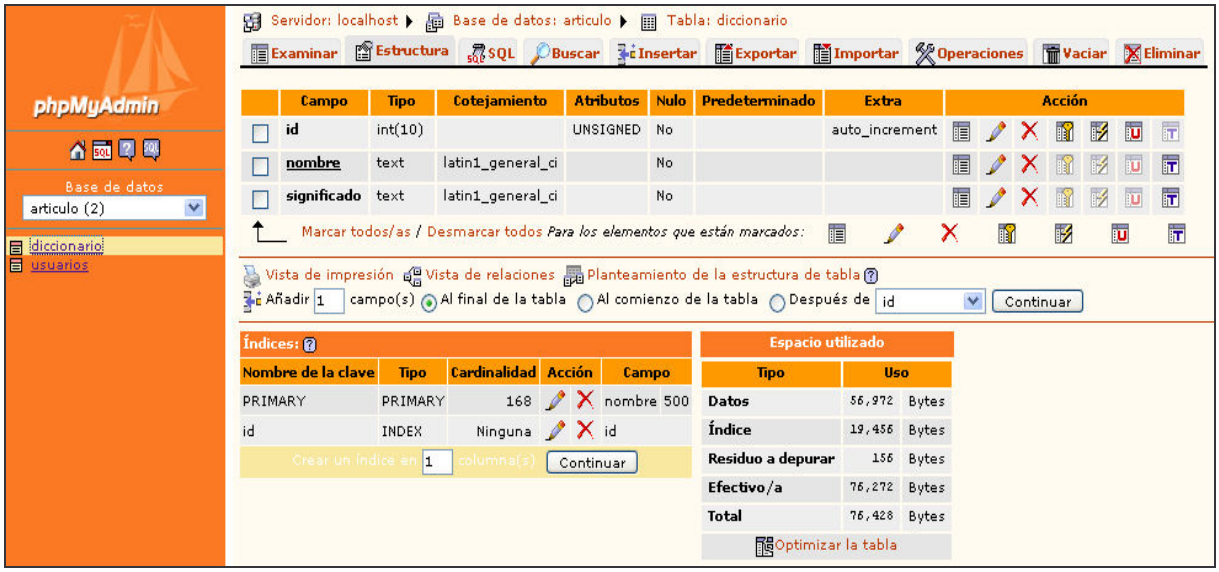


Figura 5.10. Vista de la tabla llamada diccionario

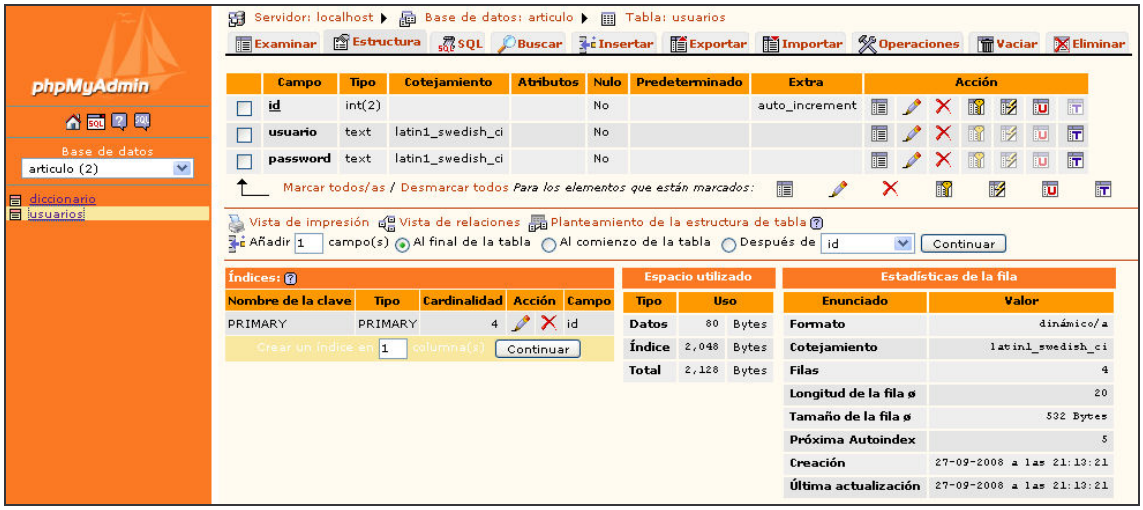


Figura 5.11. Vista de la tabla llamada usuarios

5.3.2 Diseño y desarrollo del diccionario networking.

La idea fundamental, que se planteo para el diseño de este sistema, fue tener sólo como punto principal dos factores muy importantes, por un lado la opción de buscar (buscador en la base de datos) y por otro la opción de presentar los datos en pantalla (véase figura 5.12).

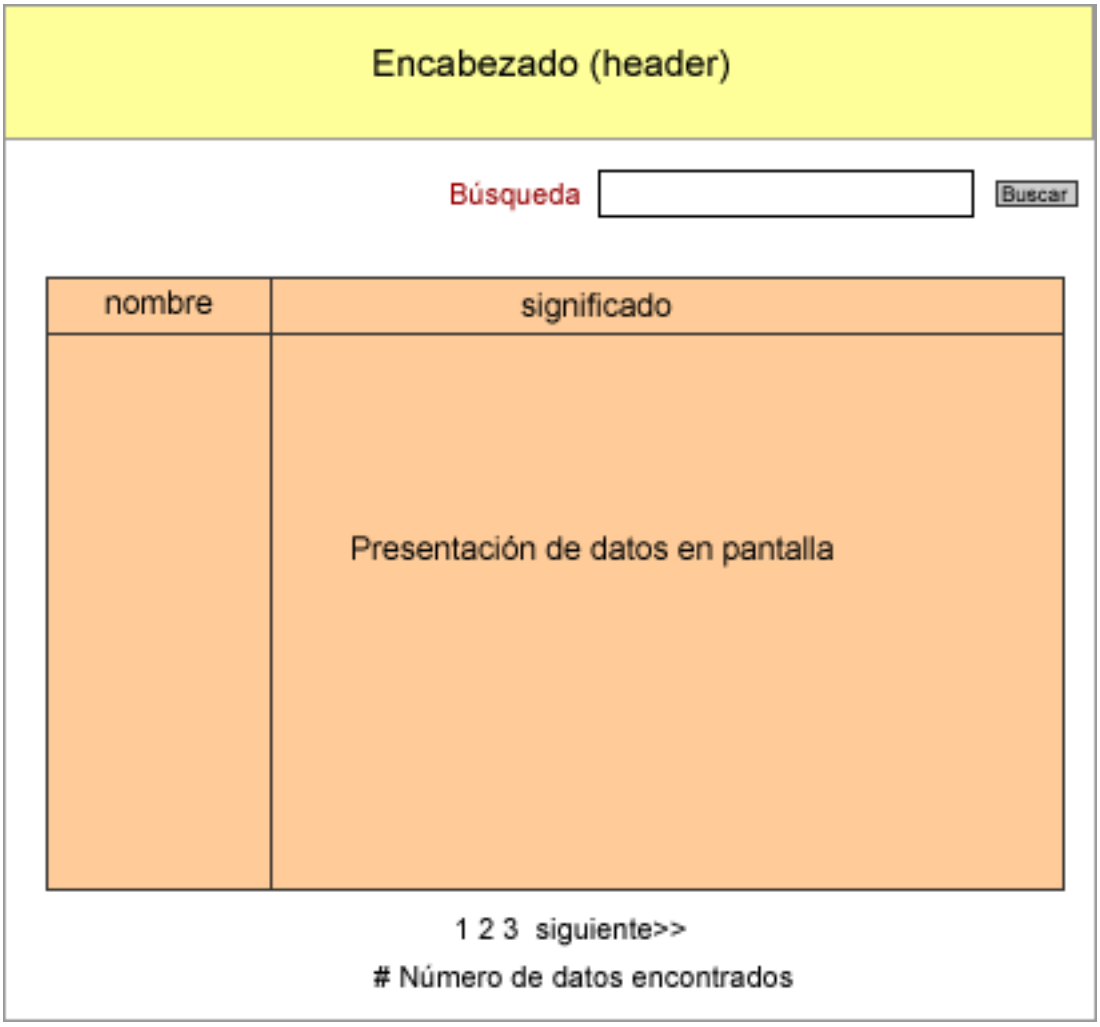


Figura 5.12. Esquema de diccionario networking

1. Encabezado. Se pretende tener un logo relacionado con la seguridad y las redes de datos. En la parte derecha, se agregará una imagen de la Facultad de Ingeniería (UNAM).

2. Campo de búsqueda. En este campo se indicará la palabra que se desee saber el significado, así como para conocer varios tipos de abreviaturas.
3. Pantalla informativa. La presentación de datos, está dividida en dos columnas, llamadas nombre y significado. El buscador trae a la pantalla datos con las palabras completas encontradas, tanto del lado de nombre o de significado.
4. Paginación. Este elemento es importante que este a la vista del usuario, ya que muestra la cantidad de pantallas disponibles con información buscada, en donde el usuario podrá navegar. Así mismo podrá el usuario recorrer por cada una de las pantallas.
5. Número de registros. Muestra el número total de registros encontrados.

A continuación se muestra la pantalla definida y terminada con las herramientas propuestas anteriormente. Cabe aclarar que la terminología del diccionario está conformada tanto en idioma español como en inglés, para su comprensión (véase figura 5.13).

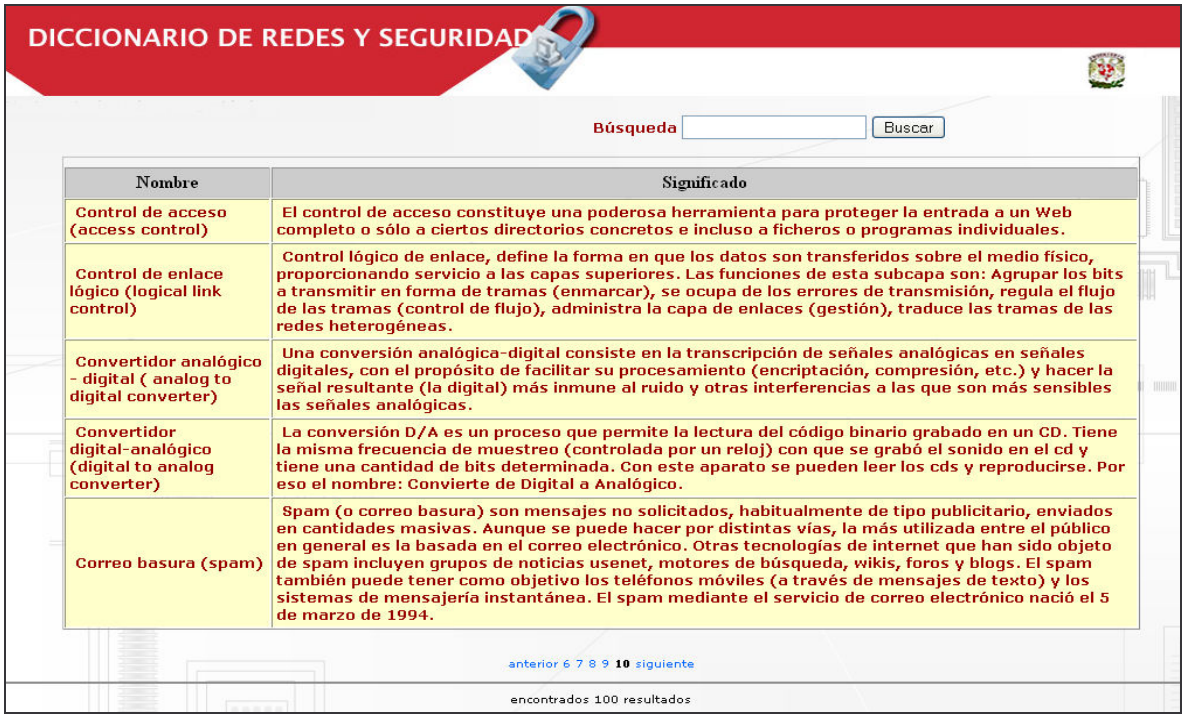


Figura 5.13. Pantalla definitiva del diccionario networking

5.3.3 Acceso al sistema

El acceso al sistema estará conformado por una clave de administrador, el cual tendrá los privilegios de poder modificar, agregar o borrar registros de la base de datos. En la figura 5.14 se muestra un diagrama de casos de uso de la administración del sistema:

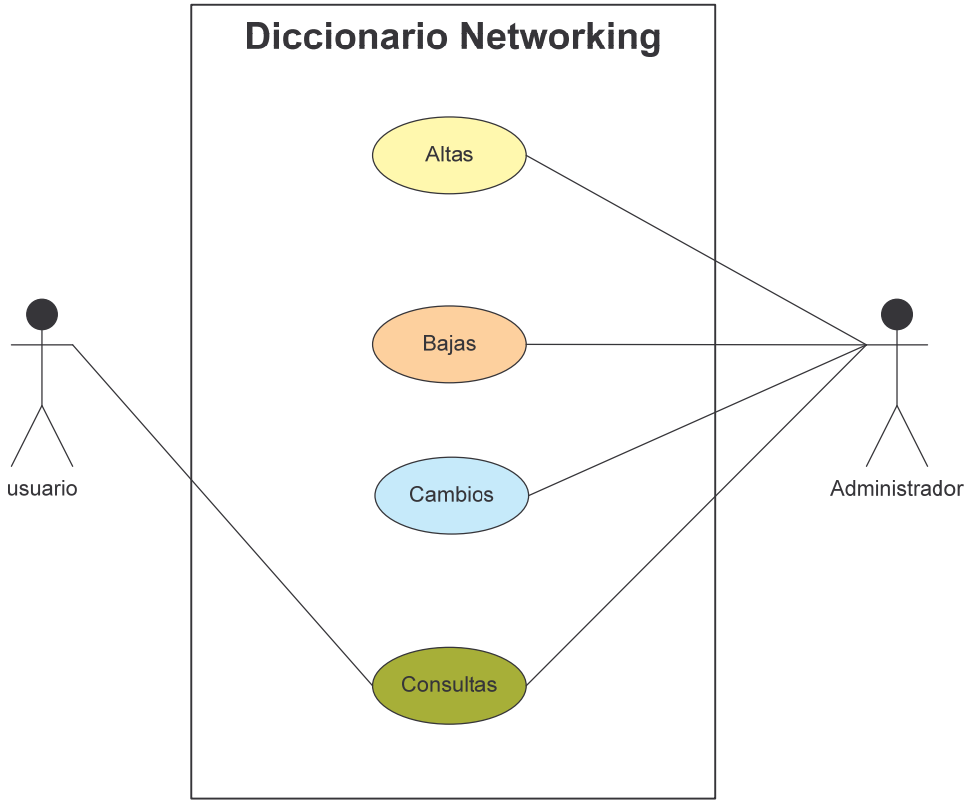


Figura 5.14. Administración del diccionario networking

Como se puede observar, el administrador será la única persona autorizada para poder ingresar o modificar información evaluada por un responsable del área. En la figura 6.4 se puede observar el proceso de evaluación de usuario.
Pasos que deberá cumplir el administrador:

1) Deberá introducir su nombre de usuario.
2) Introducir contraseña.
3) Elegir la operación a realizar (alta, baja, cambios).

5.4 Diseño y desarrollo para la administración del sistema

Para la administración del sistema son dos puntos muy importantes, tanto la validación como los formularios de opción del menú.

A continuación se muestra un esquema de administración del diccionario networking (véase figura 5.15).

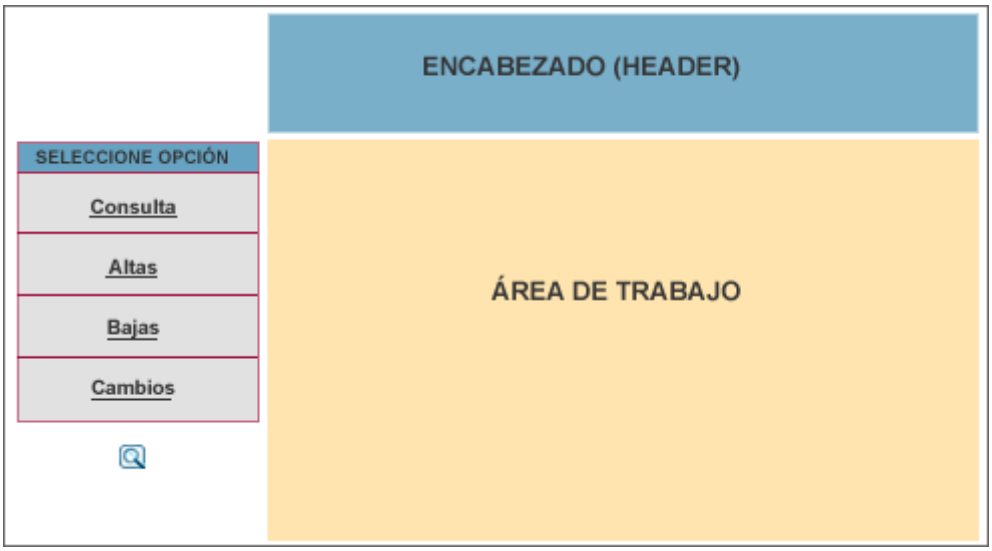


Figura 5.15. Esquema para la administración del diccionario Networking

1. Encabezado. Se pretende tener un logo relacionado con la administración del sistema.
2. Área de trabajo. En el área de trabajo, aparecerán los formularios de acuerdo a la opción seleccionada en el menú.
3. Menú. En esta parte se podrá seleccionar cualquiera de las opciones que se desee trabajar.
4. Icono de lupa. Este icono mostrará el diccionario networking, para verificar los cambios realizados en el mismo.

Con lo planeado anteriormente, llegamos al diseño final de la página del administrador, utilizando herramientas de programación como PHP, un manejador de base de datos para Mysql, JavaScript y algunas imágenes editadas en Fireworks. De acuerdo a la figura 6.7, se puede observar la relación entre la validación y las páginas de administración (véase figura 6.16).

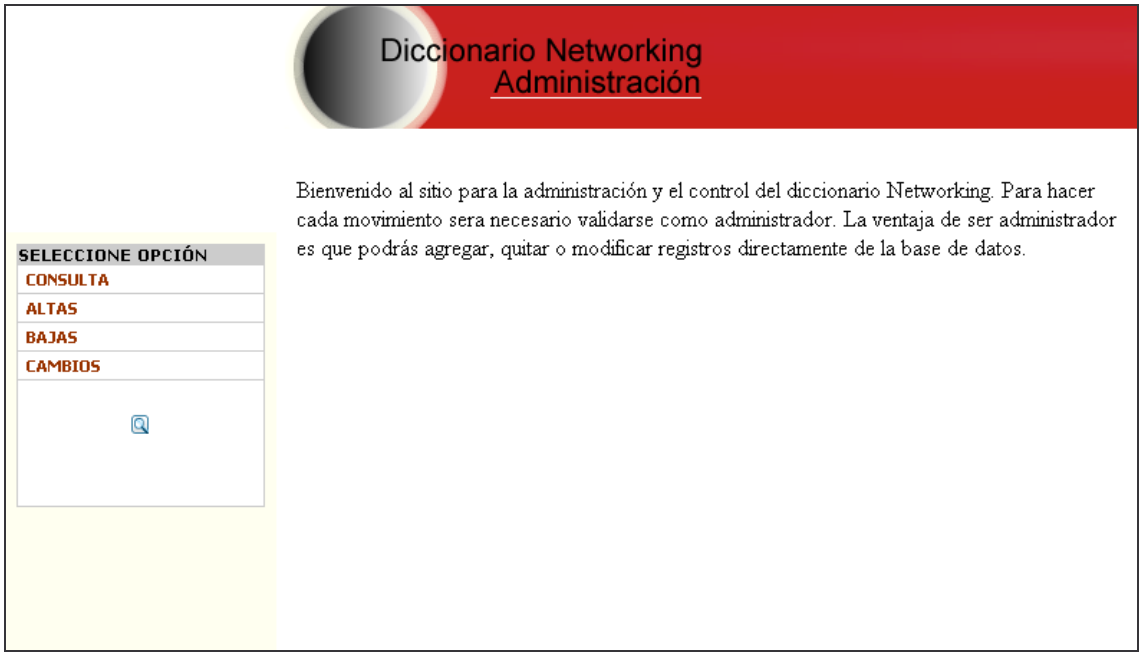


Figura 5.16. Pantalla final de administración del diccionario networking

Para la validación de usuarios, es una pantalla que sólo comprenderá dos campos donde escribir el nombre de usuario y el password, los cuales deberán de estar almacenados en la base de datos para su autenticación (véase figura 5.17).

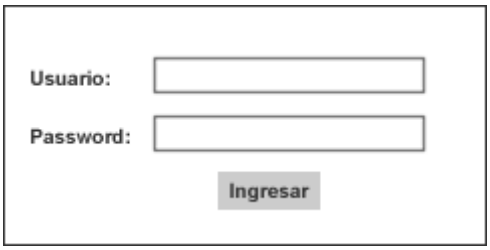


Figura 5.17. Validación de administrador

En la siguiente figura 5.18 podemos apreciar las cuatro opciones del menú que se utilizarán para consultar y modificar datos de la base de datos.

Figura 5.18. Selección de opción

El prediseño de la opción de consulta

En este diseño se pretende mostrar las tres columnas que componen a la base de datos. A continuación se muestra un esquema de administración de la consulta (véase figura 5.19).

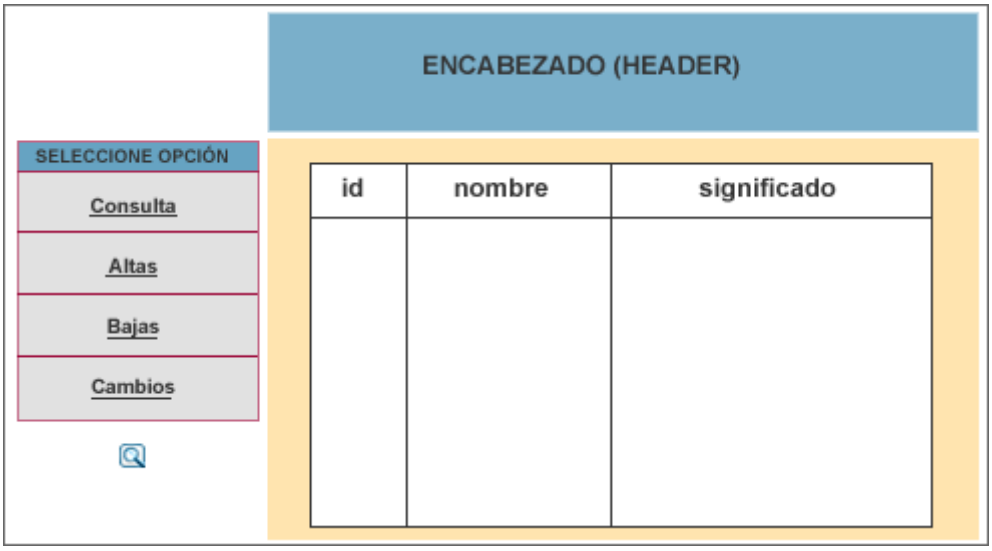


Figura 5.19. Prediseño de la opción de consulta

1. id. Es un identificador único para cada uno de los nombres y definiciones.
2. nombre. Esta columna muestra el nombre o concepto que se va a definir.
3. significado. Muestra el significado en relación con la columna nombre. Cabe aclarar que el diccionario puede contener varias definiciones de un mismo concepto.

De acuerdo a lo planeado con anterioridad, podemos observar que la pantalla de administración muestra la lista general que contiene la base de datos, previamente identificados con su id (véase figura 5.20).

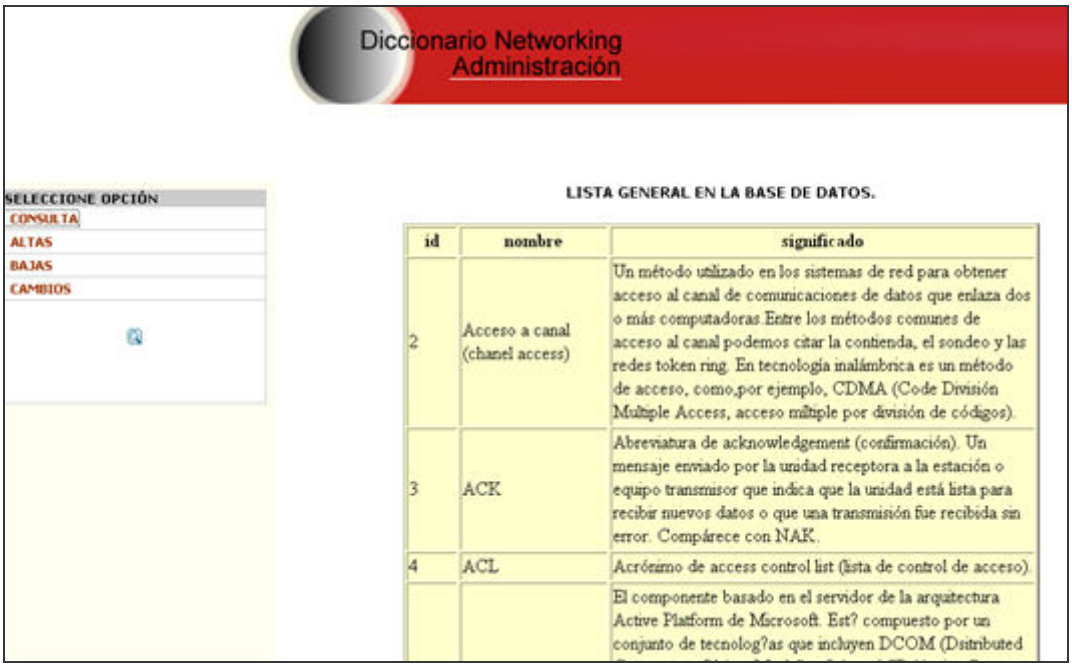


Figura 5.20. Pantalla de consulta del diccionario Networking

El prediseño de la opción de alta

Para agregar registros en la base de datos, será necesario contar en esta opción con un formulario con dos opciones, tales como nombre y significado.

El id no se muestra, por que se incrementa automáticamente cuando agregamos un registro (véase figura 5.21).

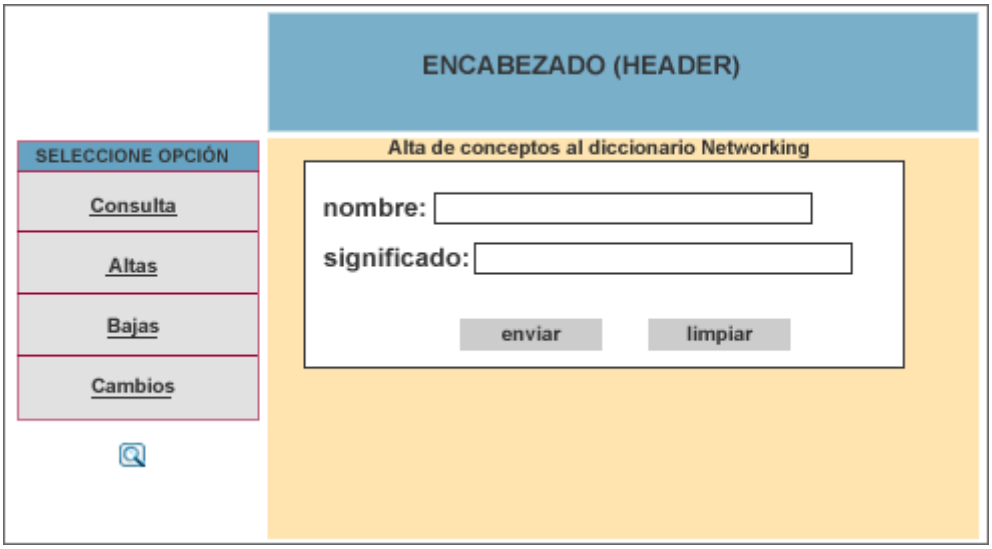


Figura 5.21. Prediseño de la pantalla de alta de registros

1. Nombre. En esta opción se capturará el nombre o concepto a definir.
2. Significado. Aquí se deberá definir el concepto de la opción anterior.

A continuación se muestra la pantalla final según el diseño anterior (véase figura 5.22).

Figura 5.22. Pantalla de alta de registros

El prediseño de la opción baja

Es importante en ocasiones rectificar registros en la base de datos, y la mejor manera es darlo de baja. Para ello sólo se pide el id a dar de baja, por ese motivo es importante consultarlo con anterioridad (véase figura 5.23).

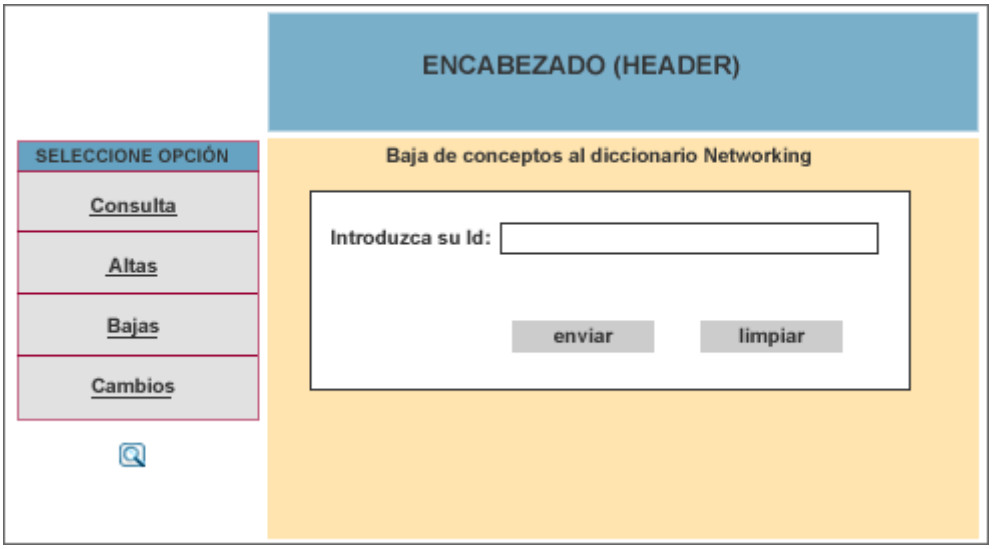


Figura 5.23. Prediseño de la pantalla de baja de registros

1. Introduzca su id. En esta opción deberá de capturar el id a dar de baja, después de ello viene una pantalla de confirmación, para asegurarse que realmente se va a dar de baja.

A continuación se aprecia la pantalla de baja, en donde acorde con el diseño anterior, el Id es el único rubro que se debe capturar (véase figura 5.24). Después de darle la opción de aceptar vendrá una pantalla de confirmación.

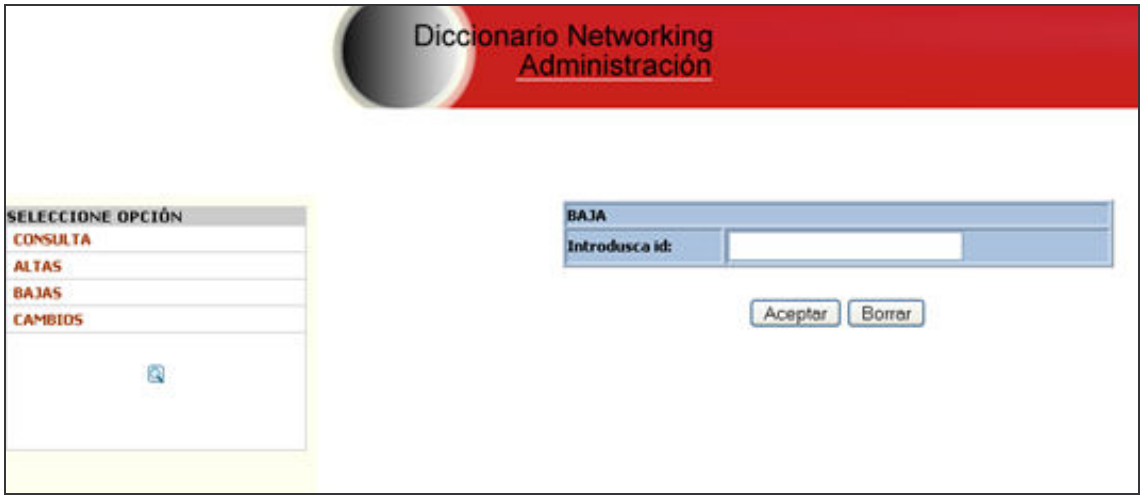


Figura 5.24. Pantalla de baja de registros

El prediseño de la opción de cambios

Para realizar cambios en los registros en la base de datos es necesario seleccionar el id que se desea cambiar, sólo basta con llenar cualquiera de las dos opciones a cambiar, ya sea el nombre o el significado (véase figura 5.25).

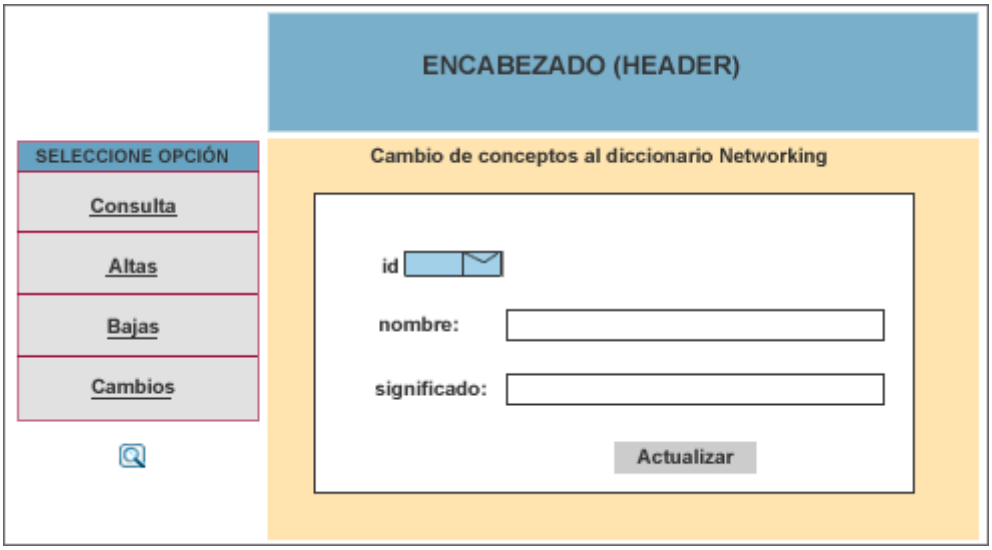


Figura 5.25. Prediseño de la pantalla de cambios en los registros

A continuación se muestra la pantalla de actualización de registros, en donde sólo es necesario seleccionar el id, para hacer las actualizaciones pertinentes (véase figura 5.26).

Figura 5.26. Pantalla de actualización de registros

Por último, cabe mencionar que el icono de la lupa muestra la pantalla del diccionario, para verificar las actualizaciones o modificaciones realizadas en la misma página del administrador (véase figura 5.27).

Figura 5.27. Pantalla del diccionario vista desde el icono de la lupa

5.5 Diseño y Desarrollo del Tutorial Práctico de Redes en Linux

Como primer paso para elaborar el diseño de este tutorial, fue necesario recabar información de fuentes confiables, como libros, revistas y sitios de Internet y con asesoría de profesores de la Facultad de Ingeniería.

Posteriormente, se analizaron los requerimientos del tutorial y se hizo un prototipo de la página web, para ello se consideró el tipo de navegación, la distribución de la información, el color, la tipografía, el fondo, la ubicación de las imágenes, el tipo de animaciones de acuerdo al ejemplo.

Respetando el objetivo que nos planteamos al inicio del proyecto, en donde enfatizamos la posibilidad de que el sistema tutorial, fuera totalmente portable, por eso se tomo la decisión de trabajar con hojas de estilo en cascada (CSS-Casscading Style Sheets), lo cual tiene muchas ventajas, entre las cuales están:

- El control centralizado de la presentación de un sitio web completo con lo que se agiliza de forma considerable la actualización del mismo.
- Hay un incremento en la accesibilidad, por ejemplo, las personas con deficiencias visuales pueden configurar su propia hoja de estilo para aumentar el tamaño del texto o remarcar los enlaces.
- Una página puede disponer de diferentes hojas de estilo según el dispositivo que la muestre o incluso a elección del usuario (impresora, dispositivo móvil, etc.).
- El documento HTML es sí mismo es más claro de entender y se consigue reducir considerablemente su tamaño (siempre y cuando no se utilice estilo en línea).

5.5.1 Pantalla principal. Descripción general

Una de las partes importantes del sistema, es la distribución de todos los elementos que contendrá la pantalla, es decir deberá de ser amigable y fácil de utilizar (véase figura 5.28).

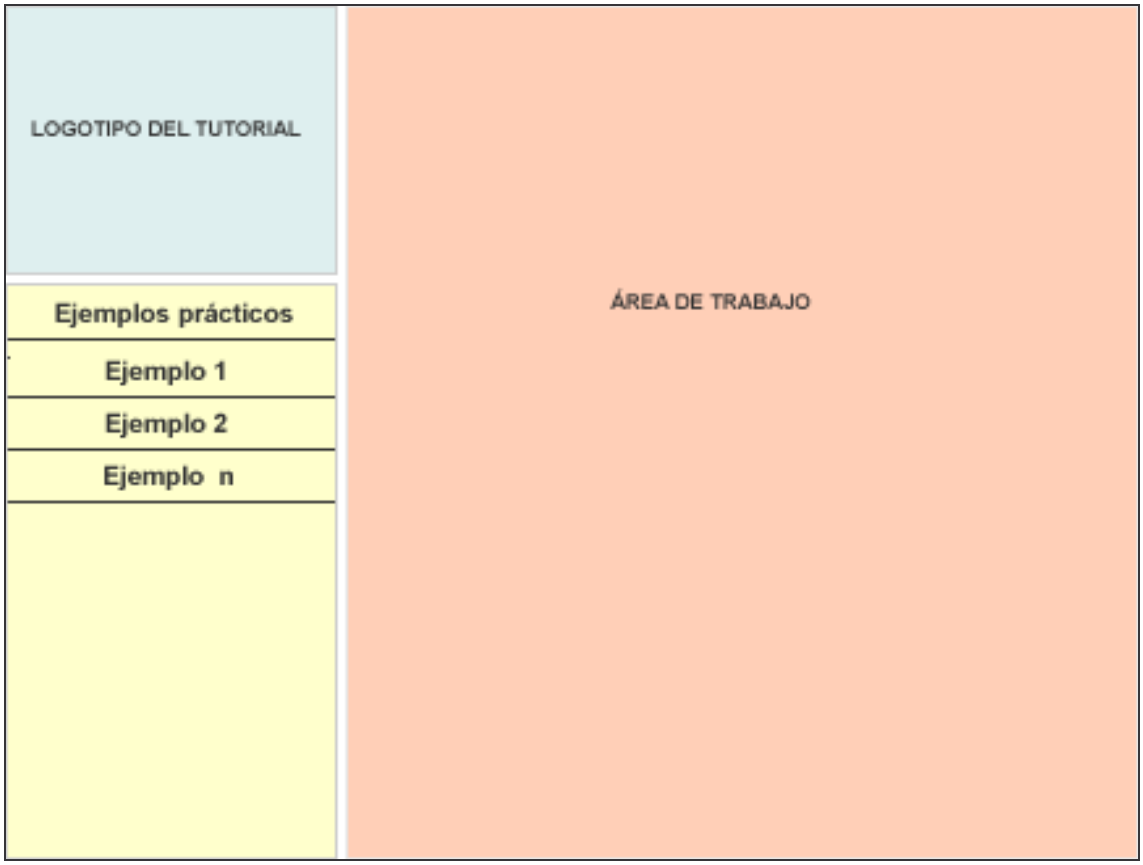


Figura 5.28. Prediseño de la pantalla de tutorial práctico de redes en linux

1. Ejemplos prácticos. Este menú contendrá los ejemplos a desarrollar en este tutorial.
2. Área de trabajo. El área de trabajo estará enriquecida con animaciones en flash, figuras y texto.
3. Logotipo del tutorial. Deberá ser una imagen representativa del tutorial.

A continuación se muestra la pantalla final del tutorial práctico (véase figura 5.29). Para este tutorial se seleccionaron los temas más utilizados en la administración de redes con Linux.

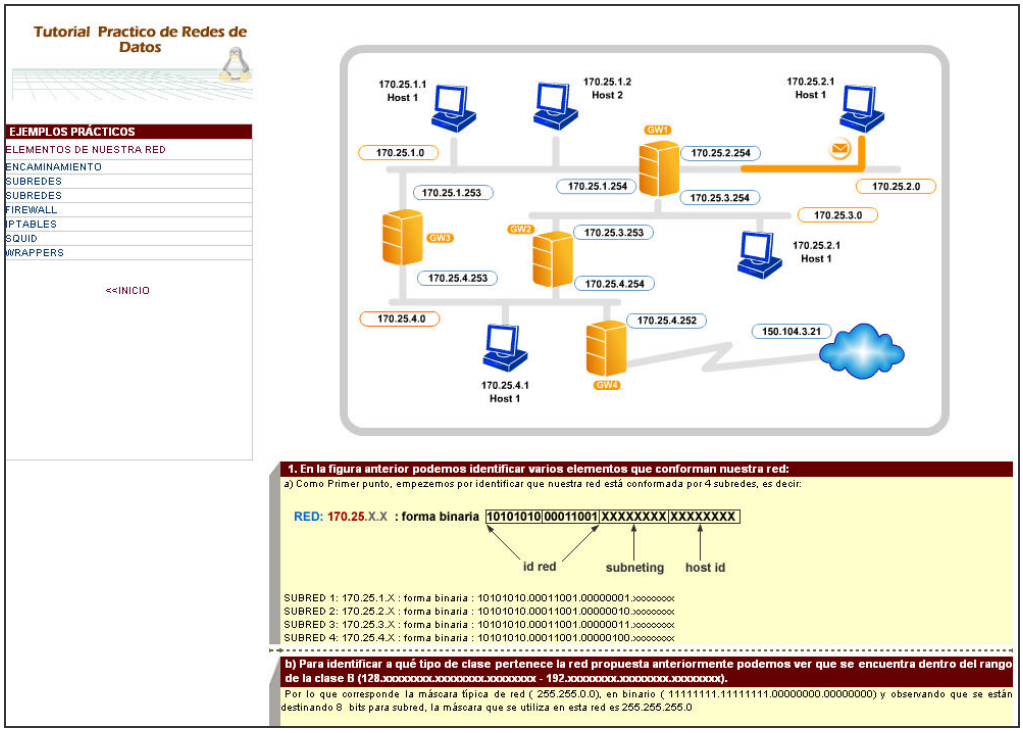


Figura 5.29. Diseño de la pantalla del tutorial práctico de redes en linux

CAPÍTULO VI

IMPLEMENTACIÓN Y PRUEBAS

6.1 Introducción

Una de las partes más importantes en el desarrollo de software, son la implementación y las pruebas, por que es aquí en donde vemos los resultados terminados de nuestro proyecto. La funcionalidad también depende de el lugar en donde se implemente, es decir, dependerá de las condiciones del servidor, la configuración y por su puesto la administración.

En la etapa de Implementación comenzamos con el resultado de la etapa de Diseño e implementamos el sistema en términos de componentes, es decir, ficheros de código fuente, scripts, ficheros de código binario, ejecutables y similares.

El objetivo principal de la etapa de implementación es desarrollar la arquitectura y el sistema como un todo. De forma más específica, los propósitos de la Implementación son:

- Definir la organización del código.
- Planificar las integraciones de sistema necesarias en cada iteración.
- Implementar las clases y subsistemas encontrados durante el diseño.

La prueba no es una actividad sencilla, no es una etapa del proyecto en la cual se asegura la calidad, sino que la prueba debe ocurrir durante todo el ciclo de vida: podemos probar la funcionalidad de los primeros prototipos; probar la estabilidad, cobertura y rendimiento de la arquitectura; probar el producto final, etc.

El objetivo de la etapa de pruebas es garantizar la calidad del producto desarrollado. Además, esta etapa implica:

- Verificar la interacción de componentes.
- Verificar la integración adecuada de los componentes.
- Verificar que todos los requisitos se han implementado correctamente.
- Identificar y asegurar que los defectos encontrados se han corregido antes de entregar el software.

6.2 Servidor PowerEdge 1800

Rendimiento.

El servidor PowerEdge 1800 utiliza los últimos procesadores Intel® XeonTM y el chipset Intel 7520 para proporcionar impresionantes velocidades de 800 MHz en un bus frontal, creando así una vía rápida para mover datos entre la memoria y el procesador. Ofrece también una memoria DDR2 para las transmisiones de datos de alta calidad y una capacidad de expansión en el futuro. El sistema PowerEdge 1800 está diseñado para proporcionar un rendimiento factible. No sólo tiene un precio de nivel básico, sino que también soporta sistemas operativos heredados para reducir el coste total de propiedad.

Capacidad de expansión y flexibilidad

Otra ventaja clave del servidor PowerEdge 1800 es su capacidad de expansión y su flexibilidad de configuración integradas. La capacidad de expansión comienza en el núcleo del sistema, gracias a la tecnología de 64 bits de memoria ampliada Intel en el procesador1. Esto le permite desplegar la tecnología de 32 bits hoy, con la posibilidad de pasar a sistemas operativos y aplicaciones de 64 bits en el futuro. El servidor PowerEdge 1800 le ofrece también un máximo de 12 GB de memoria2 para el rendimiento de aplicaciones ampliables, así como hasta 6 discos duros. Además, la arquitectura PCI ExpressTM del servidor proporciona un rendimiento de E/S de próxima generación que ofrece espacio para acoplar periféricos en el futuro.

El servidor PowerEdge 1800 ofrece una amplia gama de opciones de configuración. El chasis flexible se puede desplegar como una torre o como un servidor montado en rack 5U. De esta forma, se puede mover fácilmente de ubicaciones distribuidas al centro de datos. También presenta un controlador de interfaz de red integrado y un canal simple SCSI para una mayor flexibilidad de configuración, dejando todas las ranuras de E/S abiertas.

Disponibilidad

Además, el sistema PowerEdge 1800 incluye muchas funciones de alta disponibilidad para ayudar a maximizar el tiempo de actividad del servidor. Ofrece fuentes de alimentación y unidades de disco duro SCSI redundantes conectables en caliente, junto con una amplia gama de opciones RAID y un indicador de salud del sistema Active ID.

También ofrece una red de área de almacenamiento, clústeres de alta disponibilidad y soporte de almacenamiento externo para la protección óptima de datos.

Gestionabilidad y capacidad de servicio

El servidor PowerEdge 1800 ofrece una sencilla gestionabilidad y capacidad de servicio, ya que está diseñado para disminuir los costos de mantenimiento y garantizar un tiempo de inactividad menor. Incluye una gestionabilidad remota gracias a un controlador de gestión de placa base estándar que soporta IPMI 1.5 (Intelligent Platform Management Interface, Interfaz de gestión de plataforma inteligente). Esto permite la gestión desde cualquier programa de gestión estándar de la industria que soporte IPMI. Además, tiene la opción de agregar una tarjeta de gestión 4/P DRAC para una gama más amplia de capacidades de gestión remota que incluyan una consola de vídeo continua, integración con Active Directory y acceso a disquete/CD virtual. Gracias al chasis con el que no es necesario utilizar herramientas y al acceso trasero a las fuentes de alimentación, el mantenimiento del sistema PowerEdge resulta sencillo.

El servidor PowerEdge 1800 ofrece un precio de nivel básico con lo último en tecnología de procesadores, rendimiento ampliable, disponibilidad y gestionabilidad; y todo ello en un sistema flexible y rentable.

6.2.1 Características del servidor

CARACTERÍSTICAS	Servidor PowerEdge™ 1800 de Dell™
Factor de forma	Torre o montaje en rack 5U
Procesadores	Hasta dos procesadores Intel® Xeon™ con tecnología de 64 bits de memoria ampliada[1] hasta 3,4 GHz
Bus frontal	800 MHz
Caché	1 MB L2
Chipset	Intel E7250
Memoria	256 MB / 12 GB DDR2 400 SDRAM; 8 GB en–12 GB con disponibilidad de bastidor único de 2 GB DIMM[2]
Canales de E/S	Seis en total: dos ranuras PCI Express™ (1 x 8 vías y 1 x 4 vías); dos ranuras PCI-X® (64 bits / 100 MHz); una PCI (32 bits / 33 MHz, 5v) y una PCI (64 bits / 66 MHz)
Controlador de unidad	Canal simple incorporado Ultra320 SCSI y controlador SATA de dos canales
Controlador RAID	CERC SATA de 6 canales opcional, CERC SATA 2S, PERC4/DC, PERC4/SC y PERC4e/DC[3]
Capacidad de unidades de disco	Seis unidades SCSI conectables en caliente Ultra320 de 1" o seis unidades SATA de conexión por cable
Almacenamiento interno máximo	SCSI: hasta 1,8 TB con soporte para y disponibilidad de disco duro de 300 GB; SATA: hasta 1,5 TB
Unidades de disco duro	36 GB, 73 GB, 146 GB y 300 GB[3] (10.000 rpm) Ultra320 SCSI 18 GB, 36 GB, 73 GB y 146 GB[3] (15.000 rpm) Ultra320 SCSI 40 GB, 80 GB, 160 GB y 250 GB (7.200 rpm) SATA
Almacenamiento interno	Unidades de 10 K / 15 K RPM SCSI; unidades 7.200 RPM SATA
Almacenamiento externo	SCSI de PowerVault™ de Dell y almacenamiento de canal de fibra de Dell/EMC®
Opciones de copia de seguridad en cinta	Interno: PowerVault 100T, 110T; unidades de cinta de altura 1/2 soportadas internamente Externo: PowerVault 114T, 122T y 132T

Tarjeta de interfaz de red	Incorporado simple Intel 10/100/1000 Gigabit NIC; Intel PRO/1000 MT Gigabit NIC (cobre), Intel PRO/1000 MF Gigabit NIC (fibra)
Fuente de alimentación	650 W no redundante o alimentación redundante conectable en caliente a 675 W
Disponibilidad	Memoria ECC, Corrección de datos de dispositivo simple (SDDC); conexión en caliente opcional Unidades de disco duro SCSI; alimentación redundante de conexión en caliente opcional; chasis sin necesidad de herramientas; soporte para canal de fibra de alta disponibilidad y clúster SCSI; hardware o software RAID opcional; Active ID
Vídeo	ATI Radeon 7000-M incorporado con SDRAM de 16 MB
Gestión remota	Controlador de gestión de placa base con soporte IPMI 1.5, accesible mediante red o puerto serie; tarjeta PCI DRAC4/P opcional
Gestión de sistemas	OpenManage™de Dell
Soporte para rack	4 postes (bastidor Dell) y otros fabricantes
Sistemas operativos	Microsoft® Windows® 2000 Server, Microsoft Windows 2000 Advanced Server, Windows Server 2003 Standard Edition, Windows Server 2003 Enterprise Edition, Windows Server 2003 Small Business Premium Edition, Windows Server 2003 Small Business Standard Edition, Red Hat® Linux® Enterprise v2.1 AS y ES y Red Hat Linux Enterprise v3 AS y ES, Novell® NetWare® 5.1 y 6.5

6.3 Sistema Operativo

Red Hat Enterprise Linux también conocido por sus siglas RHEL es una distribución comercial de Linux desarrollada por Red Hat. Es la versión comercial de Fedora Core, y anteriormente lo era de RH Linux, de forma similar a como Novell SUSE Enterprise lo es respecto de OpenSUSE.

Mientras que las nuevas versiones de Fedora salen cada aproximadamente 6 meses, las de RHEL suelen hacerlo cada 18 o 24 meses. Tienen soporte oficial de RedHat, programas de certificación, etc.

A continuación se brinda una breve reseña de las nuevas características que se incluyeron en Red Hat Enterprise Linux v.3:

- **Soporte para arquitecturas múltiples:**
  Intel X86, Intel Itanium2, AMD AMD64 y IBM zSeries, POWER Series, y S/390. (Con Update 2, entregada en mayo de 2004, se agregó soporte para Intel EM64T.)

- **Partición de memoria 4-4:**
  Incremento de Kernel y espacio de dirección del usuario para los sistemas X86, permitiendo el soporte para 64GB de memoria principal y aplicaciones más grandes de usuario.
- **Native Posix Thread Library (NPTL):**
  Una capacidad multi-threading de alto desempeño que brinda desempeño mejorado para aplicaciones multi-threaded.
- **Basado en kernel Linux 2.4.21:**
  Red Hat Enterprise Linux utiliza un kernel Linux 2.4.21 estable con diversas características incluidas en los kernels de Linux 2.5/2.6. Consulte características del kernel Enterprise Linux 3.
- **Escalabilidad mejorada:**
  El soporte para SMP más grande, memoria y sistemas I/O permite a la versión 3 soportar a los servidores aproximadamente dos veces el tamaño de la versión 2.1.
- **Compatibilidad hacia adelante:**
  La versión 3 incluye bibliotecas de compatibilidad de modo que pueda ejecutar las aplicaciones de la versión 2.1 sin modificación.
- **Desktop mejorado:**
  Incluye la interfaz de usuario gráfica basada en GNOME de Red Hat, y un amplio grupo de aplicaciones de productividad personal.
- **Mejor seguridad:**
  Incluye varias nuevas características de seguridad, que incluyen el soporte para el sistema de archivo ACL.
- **Stronghold en paquete:**
  La solución de servidor Web segura de Red Hat, previamente disponible como un producto separado en capas para Enterprise Linux AS, ha sido actualizado a Apache versión 2 e incluido como parte del grupo de productos Red Hat Enterprise Linux básico.
- **Herramientas/compilador mejorado:**
  Incluye GCC 3.2 y herramientas debugging/profiling.
  Administrador de Volumen Lógico

Brinda administración de almacenamiento de nivel empresarial.

- **Networking mejorado:**

  Incluye numerosas características para mejorar la estabilidad y el desempeño.

  Soporte para Samba 3 brinda numerosas nuevas características, que incluyen una mejor interoperabilidad con el soporte de impresión Windows 2000/XP/2003 y Microsoft Active Directory.

- **Documentación y software disponible en 10 idiomas:**

  Inglés, japonés, alemán, portugués de Brasil, coreano, italiano, francés, chino simplificado, chino tradicional, y español.

Además, Red Hat Enterprise Linux v.3 soporta una cartera de más de 1000 aplicaciones ISV (Independent Software Vendor, Distribuidor Independiente de Equipamiento Lógico) y está certificado por los principales proveedores OEM en más de 600 sistemas de hardware.

6.4 Pruebas

En la siguiente figura 6.1 podemos ver la implementación y el buen funcionamiento del tutorial de redes de datos y sus complementos. Como primer paso, debemos de ingresar a la página de redes y seguridad en la siguiente dirección ip (132.248.54.2).

Figura 6.1 Pantalla principal del laboratorio de redes y seguridad

Después de ingresar, al portal de redes de datos, hay una opción del menú (proyectos), en la cual encontraremos la siguiente pantalla (véase figura 6.2):

Figura 6.2 Pantalla de ubicación de los proyectos

Por último, debemos de ingresar a la opción de tutorial de redes de datos, en la cual encontraremos el menú con tres opciones (tutorial de redes de datos, diccionario networking y el tutorial práctico de redes (véase figura 6.3)). Si existen problemas de visibilidad del menú, es necesario instalar un complemento de adobe (flash player, www.adobe.com).

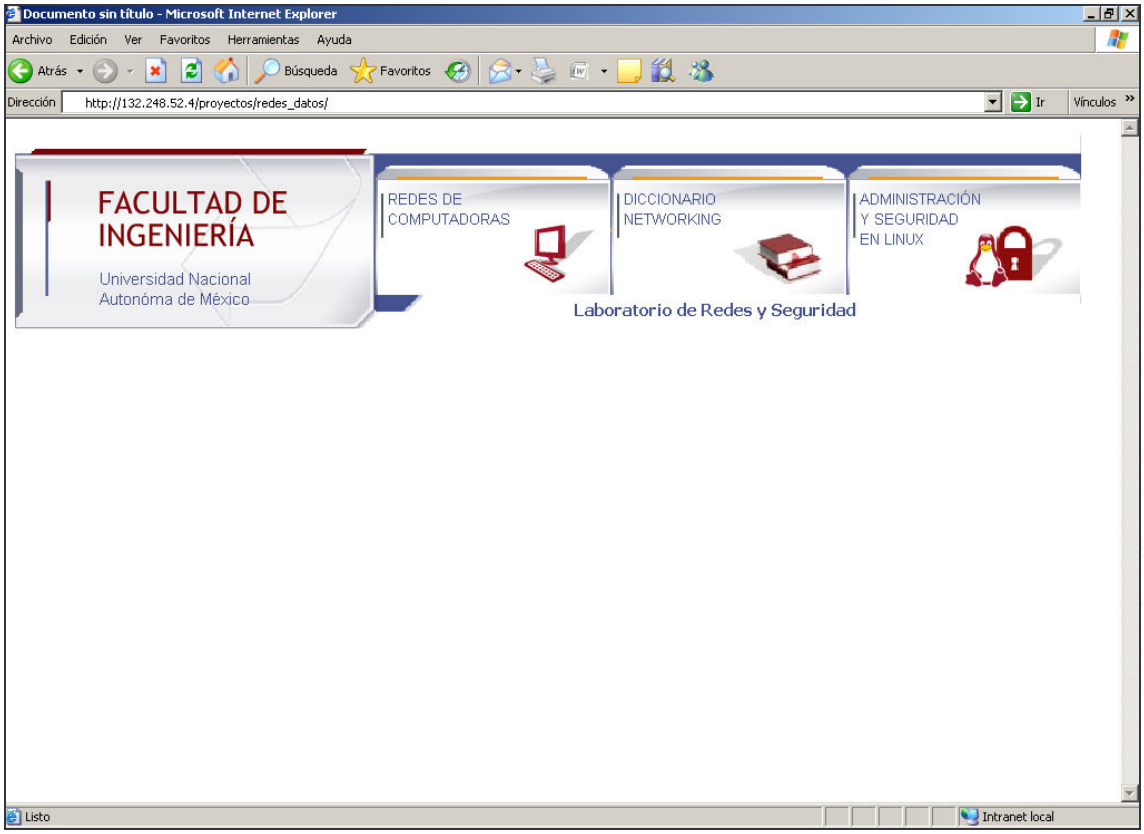


Figura 6.3. Pantalla del menú principal

En la pantalla anterior sólo bastará con seleccionar cualquiera de las tres opciones. En el tutorial de redes de datos, al darle clic, se abre una pantalla que a su vez redirecciona a otra, en donde se despliega como fullscreen (pantalla completa). En ocasiones hay bloqueadores emergentes en los navegadores, por lo cual debemos de deshabilitarlos para poder visualizarla. Esta diseñada para una resolución de 1024 x 780 pixeles (véase figura 6.4).

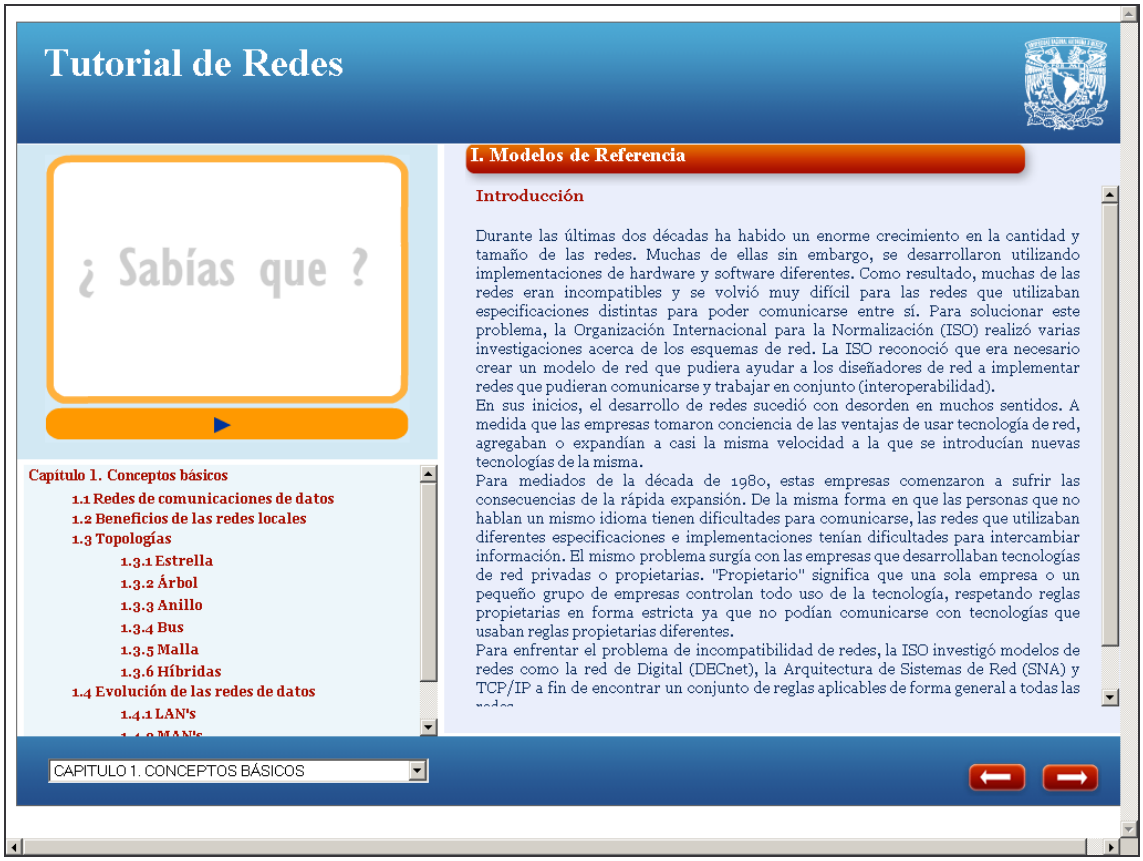


Figura 6.4. Pantalla como fullscreen en el navegador

La siguiente pantalla muestra el funcionamiento del diccionario networking.

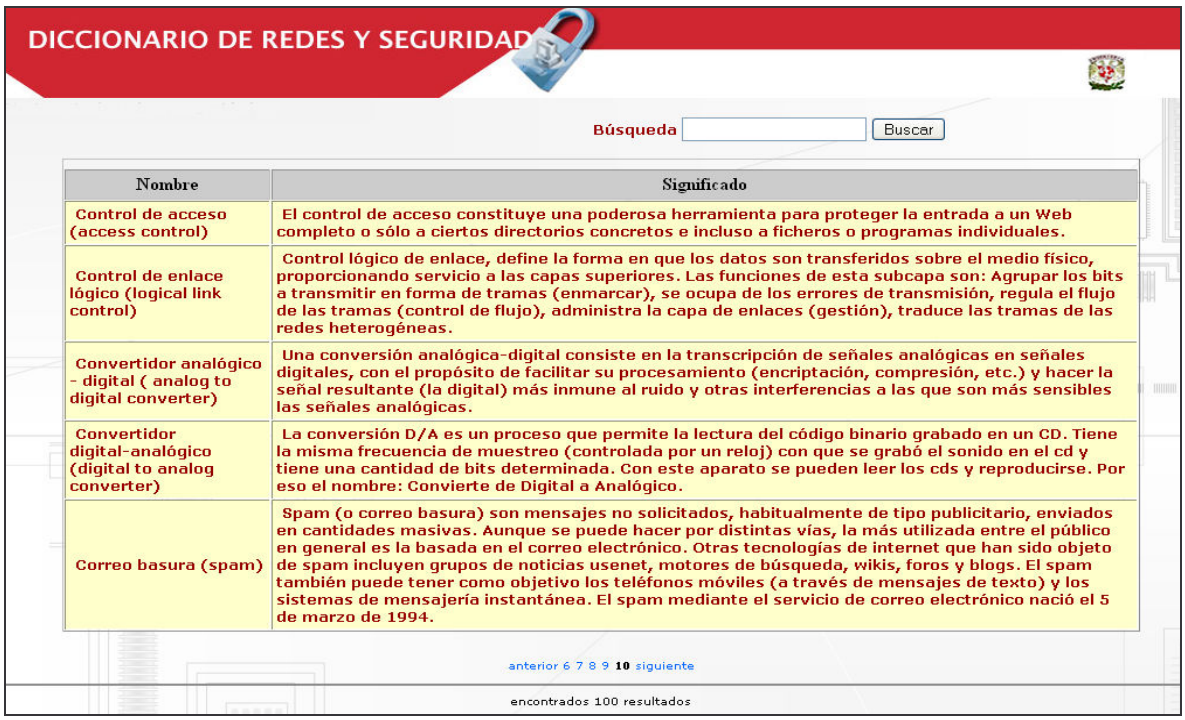


Figura 6.5. Pantalla final del diccionario networking

Y por último, podemos verificar que el tutorial práctico de redes de datos y seguridad, no presenta ningún problema en manejo.

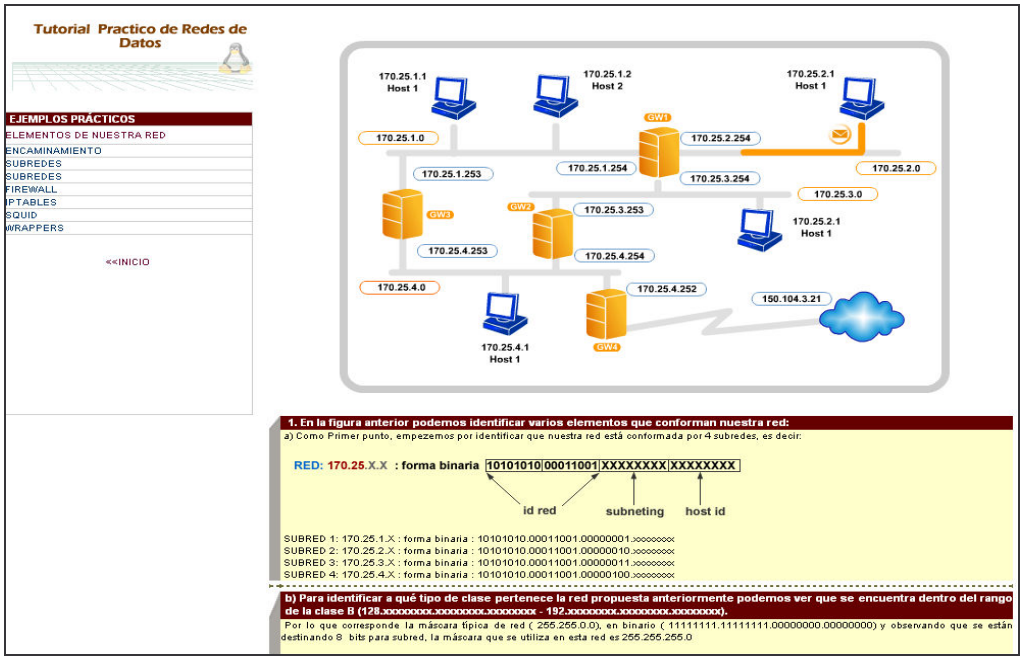


Figura 6.6. Pantalla final del tutorial práctico de redes de datos y seguridad

Conclusión

Es muy satisfactorio el saber que un proyecto como el desarrollado en este trabajo sirva como apoyo a profesores, alumnos y público en general. Además de haber cumplido con los objetivos previstos al inicio del trabajo. Una de las cualidades importantes de este trabajo es la facilidad de uso de los tutoriales, así como la portabilidad que guarda el sistema. Desde sus inicios se planteó la necesidad de hacer el sistema considerando a aquellas personas que no cuentan con internet en su hogar o en su lugar de trabajo, de tal forma que pueda distribuirse entre la comunidad estudiantil de nuestra Facultad de Ingeniería. Así mismo esto será el principio de un nuevo ciclo, en donde los alumnos y maestros puedan aportar más elementos para enriquecer este proyecto.

La elaboración de la presente tesis se fundamentó en bases sólidas de los temas de la asignatura Redes de Datos, que se imparte en la Facultad de Ingeniería; de tal modo, que se trata de un material sumamente valioso para el desarrollo académico tanto de estudiantes como profesores, ya que permite realizar un estudio complementario de los temas que conforman el área de estudio. Además, se implementaron dos módulos de ejercicios prácticos a través de los cuales el estudiante puede poner en práctica sus conocimientos tanto de Redes de Datos (diseño y configuración de redes) como de Administración de Redes donde se hace énfasis en la implementación de la seguridad mediante el uso de software libre. Cabe mencionar que este proyecto incluye varios aspectos que lo hacen todavía más útil, ya que cuenta con su propio Diccionario Networking para la búsqueda de conceptos en el área de Redes y Seguridad.

Ciertamente es importante hacer énfasis en cuanto a que las herramientas empleadas en el presente desarrollo son de última generación y con las que cuenta el Laboratorio de Redes y Seguridad, tanto de software libre como de hardware y que se adaptaron justo a la medida de nuestras necesidades;

asimismo reconocer la valiosa participación y asesoría del personal que ahí labora.

Igualmente es necesario decir que el éxito del material está íntimamente relacionado con el uso que de él se haga y su constante actualización.

Finalmente puedo decir que ha sido un gran esfuerzo la búsqueda y el procesamiento de la información que ha redundado en la creación de un sistema tutorial de vanguardia, con información factible, veraz y al día.

FUENTES DE CONSULTA

BIBLIOGRAFÍA

William Stallings.
Comunicaciones y Redes de Computadores.
Pearson Prentice Hall, 7ª ed. España,2004.

Redes de computadores. Aspectos Técnicos y operacionales
Daniel A. Menascé
Paraninfo;1988

José Luís Raya Cabrera, Víctor Rodrigo Raya
Domine TCP/IP
México D.F; santafe de Bogota;Alfaomega,1998

Adam Engst, Glenn Fleishman
Introducción a las redes inalámbricas
Anaya Multimedia

Guía del Segundo Año,
Academia de Networking de Cisco Systems CCNA 3 y 4.

Douglas E. Comer, David L. Stevens
InterConectividad de Redes TCP/IP
México D.F; Perarson Education 2000

Guía de Segundo Año
Academia Networking Cisco
Shannon Gross
Traducción. Santiago Fraguas Berasain
Pearson Educación
España; Madrid, 2002

Pressman Roger S.
Ingenieria del Software, Un enfoque práctico
Mc Graw Hill
Quinta Edición, España,2002

Tanenbaum Andrew S.
Redes de Computadoras
Prentice Hall
4ª, Edición
México, 2003

Halsall Fred
Comunicaciones de Datos, Redes de Computadores y Sistemas Abiertos
Pearson Education
4ª Edición
Mexico 1998

Comer Douglas E.
Interconectividad de Redes con TCP/IP. Vol I y II
Prentice Hall
3ª, Edición
México, 2000

Behrouz A. Forouzan
Transmisión de datos y redes de comunicaciones
Mc Graw Hill
2ª Edición
México 2002

MESOGRAFÍA

1. http://neo.lcc.uma.es/evirtual/cdd/tutorial/modelos/trasosi.html
2. http://es.wikipedia.org/
3. http://www.cybercursos.net
4. http://www.cenidet.edu.mx/wm-sema
5. http://www.enterate.unam.mx
6. http://www.iso.org/iso/home.htm
7. http://www.itu.int/net/home/index-es.aspx
8. http://www.ieee.org.mx/
9. http://www.mfaforum.org/
10. http://www2.rad.com/networks/2004/atm/main.htm
11. http://www.rediris.es/rediris/boletin/46-47/ponencia10.html
12. http://es.tldp.org/Manuales-Lucas/doc-unixsec/unixsec-html/node336.html
13. http://www.seguridad.unam.mx/
14. http://es.wikipedia.org/wiki/Capa_de_aplicación
15. http://technet2.microsoft.com/windowsserver/es/library/d1e53415-9a93-4407-87d2-3967d62182dc3082.mspx?mfr=true
16. http://www.economia-noms.gob.mx/
17. http://www.lania.mx
18. http://www.monografias.com/trabajos/redes
19. http://www.desarrolloweb.com
20. http://www.webopedia.com
21. http://es.wikibooks.org/wiki/Enrutamiento
22. http://www.monografias.com/trabajos29/direccionamiento-ip/direccionamiento-ip.shtml
23. http://www.criptored.upm.es
24. http://www.desarrolloweb.com/php/
25. http://www.programacion.net/php/
26. http://www.php.com/
27. http://www.w3c.es/divulgacion/guiasbreves/HojasEstilo
28. http://www.webestilo.com/css/

29. http://www.manualdecss.com/
30. http://www.mysql.com/
31. http://dev.mysql.com/doc/refman/5.0/es/index.html
32. http://rhn.redhat.com/errata/rhel3es-errata.html
33. http://www.programatium.com/diseno/fireworks.htm
34. http://www.aulaclic.es/flash8/
35. http://www.dell.com/downloads/global/products/pedge/es/1800_specs.pdf
36. http://www.linuxparatodos.net/portal/
37. http://www.elhacker.net/Linux.htm
38. http://www.redes-linux.com/manuales.php
39. http://www.elrincondelprogramador.com/default.asp?pag=articulos/leer.asp&id=14
40. http://www.linux.org/docs/ldp/howto/Firewall-HOWTO.html