



UNIVERSIDAD NACIONAL AUTÓNOMA DE MÉXICO
FACULTAD DE CONTADURÍA Y ADMINISTRACIÓN



Autores: Armando Moisés Bernal Kaiser y Mireya López Escobar

Informática I		Clave: 1166
Plan: 2005		Créditos: 12
Licenciatura: Informática		Semestre: 1º
Área: Informática (Gestión de la información)		Hrs. Asesoría: 2
Requisitos: Ninguno		Hrs. Por semana: 6
Tipo de asignatura:	Obligatoria (x)	Optativa ()

Objetivo general de la asignatura

Al finalizar el curso, el alumno contará con un panorama amplio y general de lo que es la informática, computación y las ramas en que éstas se dividen.

Temario oficial (horas sugeridas 96)

- 1 Antecedentes históricos de la informática y la computación. (16 hrs.)
- 2 Desarrollo de sistemas. (20 hrs.)
- 3 Introducción a la arquitectura del software. (20 hrs.)
- 4 Fundamentos de redes de computadoras. (20 hrs.)
- 5 Proyección futura de la informática. (20 hrs.)

Introducción

El estudio del desarrollo informático es una mirada al impacto de la tecnología de la información en la humanidad. Las computadoras fueron creadas con el fin de satisfacer la necesidad humana de cuantificar. Si nos remontamos a la época del surgimiento de la vida en sociedad, observaríamos seres humanos cuantificando a





los miembros de su tribu o sus pocas pertenencias; en este tiempo, les bastaba con utilizar sus dedos, piedras o cualquier otro objeto cotidiano, pero a medida que las culturas se fueron haciendo más complejas, también se necesitaron herramientas más complejas para realizar sus conteos. El ábaco es considerado como una de las primeras herramientas mecánicas utilizadas por los babilonios, los chinos y otras culturas para la realización de cálculos aritméticos hace miles de años. El gran aporte del ábaco fue la introducción de la llamada notación posicional, que es la notación que se utiliza hoy en día y que facilita mucho los cálculos aritméticos, y su principal ventaja se hace más evidente cuando se piensa en lo trabajoso que sería realizar una multiplicación usando números romanos.

En la actualidad, uno de los campos de la actividad humana en donde se manifiesta con mayor énfasis el desarrollo tecnológico que se ha presentado en las últimas décadas es el de las telecomunicaciones, que se encuentra en constante evolución y que en su contexto de sistemas moderno ocupa un lugar preponderante en el desarrollo integral de las naciones; por ello, sería ineludible tratar este tema durante este curso; que finalmente, está diseñado para ofrecer al estudiante el conocimiento básico que necesita para sobrevivir y prosperar en este mundo transformado por la tecnología de la información.

El temario de esta asignatura consta de cinco temas relevantes para cumplir con el objetivo general del curso.

En el primer tema de esta asignatura se estudian los antecedentes históricos de la informática y la computación, se analizará cuál fue el punto de partida de la primera computadora programable real y la importancia del software así como la transformación de estos programas para variar la utilidad de la computadora, hasta nuestros días en que es una herramienta de propósito general y no un dispositivo especializado de un único uso.



El segundo tema abarca el desarrollo de sistemas con el propósito de brindar una introducción al proceso de desarrollo de sistemas y definir la terminología básica utilizada en el campo del análisis y diseño de sistemas de información.

En el tema tres se estudiará el conjunto de patrones y abstracciones coherentes que proporcionan el marco de referencia necesario para guiar la construcción del software de un sistema de información, estableciendo los fundamentos para que: analistas, diseñadores, programadores, etc., trabajen en una línea común que permita alcanzar los objetivos del sistema de información, cubriendo todas las necesidades.

El cuarto tema se dedica a estudiar los fundamentos de la conexión de redes; área que, en la actualidad, se ha convertido en la más importante en la industria de las computadoras debido a que la demanda de redes más grandes, rápidas y de mayor capacidad se ha incrementado, conforme en los negocios se ha hecho evidente el valor que tiene conectar en red los sistemas de cómputo.

También se ha dedicado un tema para analizar el futuro de la informática así como el mercado de trabajo para aquellos que quieren hacer de las computadoras su medio de vida, y por último se comentarán las perspectivas de la informática a escala nacional e internacional.



TEMA 1. ANTECEDENTES HISTÓRICOS DE LA INFORMÁTICA Y LA COMPUTACIÓN

Objetivo particular

Después de estudiar este tema, el alumno debe ser capaz de describir qué es y qué hace una computadora y describir las diversas tendencias en la evolución de las mismas.

Temario detallado

- 1.1. ¿Qué es la informática?
- 1.2. Precursores del computador
- 1.3. Desarrollo del computador
- 1.4. Generaciones y tecnologías

Introducción

Esta primera parte comprende cuatro subtemas, el propósito de éstos es brindar una visión histórica de la informática y definir la terminología propia de este campo de estudio desde sus orígenes.

1.1. ¿Qué es la informática?

La informática es una disciplina relativamente nueva que tiene lazos con la ingeniería, las matemáticas y los negocios, pero tiende a centrarse más en el proceso de cálculo que en el *hardware* de la máquina; por ello es que la informática es considerada como la “ciencia de la información” para la mayoría de los autores, pero muchos informáticos prefieren referirse a la informática como la “ciencia de la computación”.

El término es acrónimo de “**información automática**”, que significa: todo aquello que tiene relación con el procesamiento de datos, utilizando las computadoras o los equipos de procesamiento automático de información.



En Estados Unidos no es muy conocido el término, que se toma como sinónimo de *Information Technology* (IT).¹

Como lo ha sido siempre, la información es considerada un valioso recurso empresarial y de poder, por lo que la informática ha ocupado un lugar imprescindible en las actividades humanas y la comunicación, incluyendo las definiciones, usos y distribución de la información.

Ya que hemos utilizado el término “información”, recordemos que este concepto nace cuando un dato (variable que carece de valor) o conjunto de datos tiene un significado que es de utilidad para alguien que debe tomar una decisión. Pongamos algunos ejemplos: El color rojo de un semáforo constituye una información, ya que tiene una interpretación específica y universal para un grupo de personas –los automovilistas- y les sirve como apoyo para tomar una decisión, detener el automóvil. Otro ejemplo de información podría ser la fecha de cumpleaños de algún familiar o allegado, ya que esta fecha tiene, en forma subjetiva, un valor que se traduce a acciones o actividades. En un contexto de negocios, “las utilidades consolidadas del mes anterior después de fletes y comisiones son de \$120,000.00”; es información porque estaría apoyando en el proceso de toma de decisiones.

En el contexto informático, los datos se refieren a los elementos crudos que la computadora puede utilizar (las computadoras trabajan los datos de muchas maneras) y a esta acción se le denomina procesamiento. Los datos pueden consistir en números, letras, sonidos o imágenes que sirven para describir hechos sobre algo y una vez que son procesados por la computadora se convierten en información. A la serie de instrucciones que indica a una computadora cómo realizar las tareas de procesamiento se le llama “programa”.

Relacionando los conceptos anteriores, entendemos que una computadora es un dispositivo electrónico capaz de recibir datos, procesarlos y entregar resultados en la forma deseada. A una computadora se le conoce también como “ordenador”

¹ Gonzalo Ferreyra C., *Informática para cursos de Bachillerato*. México, Alfa omega, 2004, p. 58.



(del francés *ordinateur*) o “computador” (del inglés *computer*) en diferentes regiones del mundo. Más adelante trataremos sobre los componentes que hacen posible el trabajo de esta maravilla del siglo XX.

Mucha gente tiende a usar en forma indistinta el término de informática y el de programación debido a que en la mayoría de los cursos de introducción a la informática ambos conceptos se asocian demasiado; sin embargo para ser más específicos, cabe aclarar que en realidad son procesos distintos y la forma en que se asocian es que la programación resulta ser sólo una herramienta intelectual más del informático; lo mismo que para un escritor de novelas lo sería la gramática.

Ahora bien, siendo la informática un conjunto de conocimientos científicos y de técnicas que hacen posible el tratamiento automático de la información por medio de computadoras. La informática combina los aspectos teóricos y prácticos de la ingeniería, electrónica, teoría de la información, matemáticas, lógica y comportamiento humano. Los aspectos de la informática cubren desde la programación y la arquitectura informática hasta la inteligencia artificial y la robótica.

1.2. Precursores del computador

“En términos generales, una computadora es un dispositivo electrónico usado para procesar datos”² de acuerdo con una serie de instrucciones almacenadas. Entre algunas de las modalidades en que una computadora procesa datos se incluye la realización de cálculos, el ordenamiento de listas de palabras o números, la modificación de documentos e imágenes y el trazado de gráficos. En una computadora la mayor parte de las tareas de procesamiento se realizan en un componente llamado unidad de procesamiento central (CPU, *Central Processing Unit*), que suele ser descrito como el “cerebro” de la computadora y que forma

² Peter Norton, *Introducción a la computación*. 3ª ed., México, Mc Graw Hill, 2006, p. 6.



parte de lo que conocemos como hardware, mientras que los programas conforman el software.

Pero las computadoras no siempre han sido iguales, en realidad existe un abismo enorme entre las primeras computadoras que se crearon y las actuales, que se están volviendo cada vez más pequeñas, más rápidas y más potentes, y son usadas en un número creciente de aplicaciones.

Sería necesario retornar miles de años en la historia de la humanidad para entender de dónde vino la idea de crear un dispositivo que facilitara al hombre hacer los cálculos aritméticos requeridos para solucionar problemas del mundo real a la vez de lograr más exactitud y rapidez. El origen del ábaco y su uso se pierde con el paso del tiempo dando lugar a otros inventos como los “logaritmos” creados por el matemático escocés John Napier (1550-1617); quien basado en su teoría de que todas las cifras numéricas podían expresarse en forma exponencial, creó las primeras tablas de logaritmos y las publicó en su obra *Rabdología*; Napier también inventó unas tablas de multiplicar móviles hechas con varillas de hueso o marfil, conocidas como “huesos de Napier” y se dice representan el antecedente de las reglas de cálculo. La primera máquina mecánica de calcular fue creación del científico alemán Wilhelm Schickard (1592-1635) misma que desafortunadamente se destruyó en un incendio y no pudo ser reconstruida. Fue en 1642 cuando se propuso el concepto de un dispositivo mecánico que fuera capaz de realizar las operaciones matemáticas básicas (suma, resta, producto, cociente); Blaise Pascal diseñó con tal finalidad la “Pascalina” con un mecanismo sencillo basado en piezas de relojería (ruedas giratorias numeradas), esta máquina permitía efectuar sumas y restas simples. Sin embargo, aunque su invento efectivamente funcionaba, no tuvo mucha popularidad debido al alto grado de complejidad que significaba operar la mencionada máquina, por no hablar de su reparación.

El desarrollo de estas herramientas no se detuvo, de forma casi simultánea con la máquina de Pascal, William Oughtret creó una herramienta basada en logaritmos,



con base en los resultados al operar con logaritmos; Oughtred inventó la "*regla de cálculo*"; que adicional a las funciones anteriores, también calculaba exponentes, funciones trigonométricas y otras funciones matemáticas de mayor complejidad, y en 1694 Gottfried Wilhelm von Leibniz (1646-1717) creó una máquina que aparte de sumar y restar podía también multiplicar, dividir y extraer raíz cuadrada.

En el campo de la industria, las fábricas de los siglos XVIII y XIX constituyeron un cambio fundamental en los procesos de manufactura y nuevas necesidades e ideas surgieron en este tiempo, principalmente se pensó en incluir las habilidades de los trabajadores en las máquinas. Un reconocido ejemplo de esta idea es el telar de tejido, inventado en 1801 por el francés Joseph Marie Jacquard (1752-1834), el cuál es usado todavía en la actualidad y se controla por medio de tarjetas perforadas. El telar de Jacquard opera de esta manera: las tarjetas se perforan estratégicamente y se acomodan en cierta secuencia para indicar un diseño de tejido en particular. El telar de Jacquard constituyó una genialidad al lograr que cualquier persona con un adiestramiento mínimo pudiera producir telas con una rapidez mucho mayor que la de los maestros tejedores más expertos con uso de los métodos tradicionales.

La Revolución Industrial plantó la semilla para la revolución de la información debido a que la complejidad creciente de los mundos de la ciencia y el comercio requería cálculos cada vez más complejos, y fue a comienzos del siglo XVIII cuando llegó el momento de crear una nueva máquina calculadora más poderosa.

Charles Babbage (1791-1871), talentoso inventor y matemático británico, dedicó su habilidad y gran parte de su riqueza a crear su "máquina diferencial" no muy diferente en cuanto a sus componentes a las calculadoras mecánicas de la época: los números estaban representados por posiciones de ruedas conectadas mediante ejes, dientes y trinquetes (tecnología de relojería), pero destinada a ser mucho más compleja y autómatas que las existentes; sin embargo, al cabo de 11 años Babbage todavía no terminaba su máquina diferencial y no precisamente por



problemas en sus planos, sino por la tecnología con que tuvo que poner en práctica su diseño.

El concepto de Babbage sufrió lo que hoy se conoce como “problema de escala”: un concepto que funciona perfectamente en un tamaño dado fracasa en forma totalmente impredecible en otro dispositivo cuyo tamaño es decenas o centenas mucho mayor.³

En 1834, aún con su máquina diferencial incompleta, Babbage concibió la idea de aplicar el concepto de las tarjetas perforadas del telar de Jackard en una máquina mucho más compleja que pudiera poner en práctica cualquier conjunto dado de instrucciones matemáticas que se le diera, a la que le llamó “máquina analítica” y en la cuál comenzó a trabajar de inmediato.

La máquina analítica funcionaría con un motor de vapor al igual que los telares de la época y consistiría en varios dispositivos más pequeños: una para recibir la lista de instrucciones (en tarjetas perforadas), otro para poner en práctica las instrucciones codificadas en las tarjetas, uno más para almacenar los resultados de cálculos intermedios y otro para imprimir la información en papel. A este propósito se sumó el esfuerzo de Lady Ada Augusta Lovelace, quien sugirió la idea de que las tarjetas perforadas pudieran adaptarse de manera que propiciaran que el motor de Babbage repitiera ciertas operaciones, por esta razón se le conoce como la primera programadora.

Lovelace mantuvo una correspondencia regular con Babbage y publicó un artículo sobre la máquina analítica que incluía el primer programa para computador. Se hizo socia de Babbage aportando mayores alcances a su idea y corrigiendo los errores en su trabajo.⁴

Los trazos detallados de Babbage describían las características incorporadas ahora en la moderna computadora electrónica (entrada de datos, un procesador central para realizar operaciones aritméticas y lógicas especificadas en un programa, una unidad de memoria que almacene información y una sección de

³ Rick Decker y Stuart Hirshfield, *Máquina analítica. Introducción a las ciencias de la computación con uso de la Internet*. México, Thomson Learning, 2001 p. 7.

⁴ George Beekman, *Computación & Informática Hoy: una mirada a la tecnología del mañana*. Wilmington, Delaware / México. Addison Wesley Longman. 1995 p. 3.



salida de resultados), por ello es que se le conoce como el “padre de la computación”; sin embargo, sus ideas estaban tan adelantadas a su época que no pudo ver su obra terminada y después de su muerte en 1871, su trabajo quedó en el olvido.

Unos años más tarde, en 1887 el gobierno de los Estados Unidos tenía serios problemas para obtener los resultados del censo de 1880 ya que no se habían dado grandes avances en el desarrollo de las calculadoras mecánicas. Afortunadamente, Herman Hollerith, un joven matemático e inventor, combinó la antigua tecnología de las tarjetas perforadas con la nueva tecnología eléctrica para producir una máquina de tabulación y ordenación, con la ayuda de este dispositivo, el censo pudo completarse en seis semanas.

Hollerith fundó la *Tabulating Machine Company* y vendió sus productos en todo el mundo. El primer censo llevado a cabo en Rusia en 1897 se registró con el tabulador de Hollerith. La gran demanda de sus máquinas se debía a que permitía la administración precisa y oportuna de grandes volúmenes de datos. La novedosa empresa de Hollerith varios años después sería conocida con el nombre de *International Business Machines (IBM)*.

1.3. Desarrollo del computador

Tuvieron que pasar más de 30 años (tiempo durante el cual se dieron grandes descubrimientos como el tubo de vacío inventado por Lee De Forest en 1906), para que alguien retomara el interés por eliminar el tedio de los cálculos con una visión más allá de máquinas calculadoras mecánicas. Fue en 1939 cuando el físico John V. Atanasoff comenzó a trabajar en una máquina que utilizaba impulsos eléctricos para representar la información.

La idea de codificar eléctricamente la información tenía funcionando alrededor de cien años desde que se inventó el telégrafo, incluso la máquina de Hollerith utilizaba señales eléctricas; pero Atanasoff y su ayudante Clifford Berry, utilizaron



las ventajas de la tecnología eléctrica para eliminar muchos dispositivos mecánicos en el funcionamiento de su nueva máquina. Utilizaron tubos de vacío (bulbos) de 3 elementos para activar o interrumpir las señales, lo que resultaba más rápido, más fiable y mucho más económico que usar ejes, engranes y palancas para realizar la misma función. Su computadora ABC (de “Atanasoff-Berry Computer”) estaba diseñada para un solo propósito: encontrar la solución a sistemas de ecuaciones lineales y su rapidez de respuesta sería alrededor de cinco veces más rápida que la de una persona utilizando una calculadora de escritorio.

Casi simultáneamente, un ingeniero alemán de nombre Konrad Zuze y su ayudante Helmut Schreyer desarrollaron un proyecto para crear una computadora de propósito general (similar a la máquina analítica de Babbage). Su creación fue la Z1, que tuvo varias versiones, y la diseñaron para efectuar tareas bajo el control de un programa de instrucciones.

Este desarrollo simultáneo e independiente del cálculo electrónico por Atanasoff y Zuze, incluido el redescubrimiento involuntario de las ideas de Babbage por Zuze, es un ejemplo de un tema recurrente en la historia de la tecnología. Con frecuencia, cuando llega el momento justo para una idea, la tecnología y la forma adecuada de analizar un problema son campo fértil para descubrimientos independientes del mismo principio o dispositivo. Sin embargo, tristemente el momento tampoco fue el adecuado para Atanasoff y Zuze. Los gobiernos de sus países estaban enfrascados en un conflicto bélico y solicitaron su apoyo en otras tareas. Ni Atanasoff ni Zuze pudieron concretar sus máquinas, y para cuando terminó la Segunda Guerra mundial, la marcha del progreso los había sobrepasado, alimentada por la misma guerra que interrumpió sus proyectos independientes.⁵

Hubo un tercer proyecto independiente bajo la dirección de Howard Aiken (especialista en física y matemáticas aplicadas de la Universidad de Harvard) que fue presentado a la IBM y posteriormente recibiría el apoyo de la Marina estadounidense: un dispositivo electromecánico controlado por dispositivos que se basaban en relevadores que usaban un imán, controlado por otro circuito eléctrico, para poder moverse. Los relevadores resultaban más lentos que los bulbos, pero

⁵Rick Decker y Stuart Hirshfield, op. cit., p. 12.



tenían otras ventajas respecto a éstos, no se quemaban, usaban muy poca energía y eran baratos.

Hacia el año 1939 la Universidad de Harvard firmó el convenio de trabajo con la IBM, que permitió realizar el proyecto de Aiken y que sería conocido como la primera computadora (al menos americana) funcional: la Harvard Mark I. Ésta procesaba información en cintas perforadas, demorándose un segundo por cada 10 operaciones. Medía más de 15 metros de longitud y 2,5 metros de alto. Posteriormente se construyeron dos versiones más: MARK II y MARK III.

Aunque Harvard e IBM produjeron la primera computadora electromecánica, en otras universidades trabajaban en máquinas similares, éstas llenaron rápidamente su atraso superando tecnológicamente la innovación de Aiken. Así, John Mauchley y J. Presper Eckert construían en 1946, en la Universidad de Pennsylvania, la ENIAC (*Electronic Numerical Integrator And Calculator*), la primera computadora electrónica diseñada con el fin de producir tablas de disparos balísticos para la artillería (calcular las trayectorias de proyectiles), una tarea de cálculo muy compleja, que estaba en funciones al final de la guerra.

El proyecto ENIAC tuvo la fortuna de contar con el talento de Jonh von Neuman, quien fue el primero en proponer la idea de almacenar los datos de entrada y el programa de la computadora en la memoria de la misma usando el código binario (codificación mediante bits; es decir, unos y ceros). Esta fue la idea que llevó finalmente a la aplicación práctica de la nueva tecnología y resolvía el problema de recablear la máquina después de realizar cada operación (debido al calor generado por los tubos de vacío). De estar forma, la ENIAC podía calcular la potencia 5000 de un número de 5 cifras y llegó a ser 1500 veces más potente que la MARK I.

Estos y otros proyectos surgidos posteriormente tenían bases de conocimiento e investigación de personas intelectuales en países distintos. Por ejemplo; en 1937 el norteamericano Claude Shannon, creador del término “bit”, demostró que la



programación de las computadoras era primordialmente una cuestión de lógica, evidenciando la importancia del álgebra de Boole, a la vez que sugirió el uso de sistemas de conmutación (como en centrales telefónicas). En Gran Bretaña, Alan Turing concibió en 1937 un proyecto (teórico) de cerebro artificial. Turing había tenido una participación importante en la construcción de "Colossus", una computadora capaz de descifrar los mensajes nazis generados por su contraparte, la máquina alemana "Enigma" y en 1947, Turing publicó el libro "Maquinaria Inteligente" donde acercándose al tema de la inteligencia artificial comparaba las computadoras con el cerebro de los niños recién nacidos (listos para ser programados). Inventó la prueba de diálogo conocida con su nombre: "si no podemos distinguir entre un interlocutor vivo y una máquina, esta puede ser considerada como inteligente" (prueba de Turing).

Hasta la década de 1950 todas las computadoras eran creadas para propósitos especiales que sólo instituciones enormes como los gobiernos, los ejércitos y las universidades podían costear, pero a partir de 1951, año en que se crea la UNIVAC (*Universal Automatic Computer*), se da el paso hacia la comercialización y masificación de las computadoras.

Las computadoras, como la ENIAC y la UNIVAC, eran tan grandes que abarcaban un gran espacio. Aún con sus enormes "cerebros electrónicos", estas máquinas maravillosas tenían poco más de poder de cómputo que un reloj de pulsera digital o una calculadora de bolsillo modernos.

En la década de 1960, las computadoras modernas empezaron a revolucionar el mundo de los negocios (IBM en auge con su macrocomputadora sistema/360).

En la década de 1970, IBM empezó a fabricar computadoras de distintos tamaños, para satisfacer distintas necesidades y presupuestos. Esto dio a las computadoras un uso más común.

Desde entonces, las computadoras continúan reduciendo su tamaño a la vez que proporcionan más potencia con menos dinero. Hoy en día, el tipo más común de computadora se llama **computadora personal**, o **PC**, porque está



diseñada para ser usada por una persona a la vez. A pesar de su tamaño pequeño, la computadora personal moderna es más potente que cualquiera de las máquinas del tamaño de un cuarto de las décadas de 1950 o 1960.⁶

Se puede decir que a principios de los 80 la mayoría de la gente tenía muy poco que ver con las computadoras, al menos de manera directa. Es decir, a pesar de que la gente llenaba formas, presentaba exámenes y pagaba cuentas computarizados, el trabajo directo y real con las computadoras era cuestión de especialistas: programadores, personal de incorporación de datos y operadores de computadoras.

Pero llegaron las microcomputadoras (PC) y todo cambió. Hoy en día casi cualquier persona debe usar una computadora, ya que es una herramienta común en todas las áreas de la vida (escribir, dibujar, hacer cálculos, etc.), gracias a ésta se han desarrollado nuevas formas de aprendizaje, ya que personas que deben permanecer en su hogar, que viajan constantemente o que trabajan en horarios inusuales, pueden tomar sus cursos por computadora a través de un enlace telefónico. También representa un medio importante para comunicarse y coincidir con personas con intereses similares a los nuestros. Toda clase de personas está haciendo uso de Internet para conocerse y compartir ideas.

Por todo lo anterior, las computadoras son imprescindibles para las personas de negocios. A la gente que hace uso de microcomputadoras o tiene acceso a una computadora más grande, se le conoce como “usuario final”. Para satisfacer sus necesidades particulares de información es necesario que esta gente aprenda a usar programas de computación en paquetes. Cuando una persona aprende a usar la computadora para satisfacer no sólo sus necesidades sino las de una empresa, entonces elevará su productividad y esto lo convertirá en un empleado más valioso.

⁶ P. Norton, op. cit., p. 6.



1.4. Generaciones y tecnologías

Ha sido sorprendente el avance de la tecnología de las computadoras desde los primeros años del siglo XX. Con el descubrimiento de nuevos dispositivos electrónicos, los avances de la programación y el desarrollo de los nuevos sistemas operativos, se han marcado fechas importantes para clasificar a las computadoras de acuerdo con sus componentes y con su capacidad de procesamiento, agrupándolas por “generaciones”, mismas que van aumentando en número a medida que se hacen nuevos descubrimientos y aplicaciones en la ciencia de la computación.

La **primera generación** de computadoras (entre 1951 y 1959 aproximadamente) se caracterizó por el uso de bulbos o de relevadores que propiciaban un consumo excesivo de energía eléctrica y las salas donde se instalaban requerían de costosos sistemas de enfriamiento debido al calor generado. Los operadores ingresaban los datos por medio de tarjetas perforadas y la programación solamente se desarrollaba en “lenguaje de máquina” o “binario”. El almacenamiento interno se lograba con un tambor que giraba rápidamente, sobre el cual un dispositivo de lectura/escritura colocaba marcas magnéticas. Otra característica de las computadoras de esta generación es su limitada capacidad de memoria y procesamiento. Ejecutaban los procesos de una manera netamente secuencial; es decir, toda la información debía ser almacenada en memoria antes de que el programa pudiera ser ejecutado, y no se podía alimentar a la computadora con nueva información hasta que el programa actual terminará de ejecutarse.

Generalmente se considera que empezó con la presencia de la primera computadora electrónica digital comercialmente viable la UNIVAC; esta computadora se instaló en 1951 en la oficina de censos de Estados Unidos y tuvo una gran difusión al predecir la victoria de Dwight Eisenhower sobre Adlai Stevenson en las elecciones presidenciales, con tan sólo el 5% de los votos cuantificados.



Fue hasta que ocurrió el éxito de la UNIVAC I cuando IBM se decidió a desarrollar y comercializar computadoras. El ingreso de esta empresa en este campo se dio con la IBM 701 en 1953. No obstante la IBM 650 introducida en 1954, probablemente sea la razón de que IBM disfrute de una porción considerable del mercado actual de computadoras.⁷

El transistor es inventado en 1948 en los laboratorios Bell por John Bardeen, Walter H. Brattain y William Shockley. Este elemento podía realizar las mismas tareas que los bulbos con muchas más ventajas, por lo que éstos pronto fueron reemplazados.

La **segunda generación**, se inicia cuando aparecen las primeras computadoras a partir de transistores, sustituyendo a las que funcionaban con bulbos. Los laboratorios Bell construyeron en 1954 la primera computadora transistorizada, la TRADIC (*Transistorized Airborne Digital Computer*) cuya estructura interna incluía 800 transistores.

El cambio de tecnología logró un aumento de la confiabilidad de las computadoras en un factor de 10, así como logró disminuir el costo de los mismos, al reducir las necesidades de mecanismos de refrigeración, debido a que los transistores generan muchísimo menos calor que los tubos de vacío (bulbos). Además de estas ventajas la tecnología de los transistores incrementó significativamente la velocidad de procesamiento.

Respecto a la programación, siguen dominando las sistemas de tarjeta perforada o cinta perforada para la entrada de datos. Pero en esta época se desarrolló el primer “lenguaje de alto nivel”, el FORTRAN (FORmula TRANslator) que es muy apropiado para trabajos científicos, matemáticos y de ingeniería. Hubo otras aportaciones importantes en esta rama, como el lenguaje LISP (acrónimo de LISP Processor) que permite el manejo de símbolos y listas; pero sin duda, lo más sorprendente hasta entonces, fue el lenguaje de programación COBOL (COmmon

⁷ Larry E. Long y Nancy Long, *Introducción a las computadoras y a los sistemas de información*. México, Prentice Hall, 1999, p. 36.



Business Oriented Language) uno de los primeros programas que se pueden ejecutar en diversos equipos de cómputo después de un sencillo procesamiento de compilación.

“Grace Murray Hopper (1906-1992), quien en 1952 había inventado el primer compilador, fue una de las principales figuras del CODASYL (COMmittee on DATA SYstems Languages), que se encargó de desarrollar el proyecto COBOL”⁸

Otro cambio notable en las computadoras de esta generación fue su tamaño, gracias a la inclusión de memorias de ferrita (redes de núcleos que contenían pequeños anillos de material magnético, enlazados entre sí, en los cuales podrían almacenarse datos e instrucciones), reduciendo también su consumo de energía eléctrica y el calor generado, y aunque aún necesitaban los sistemas de enfriamiento, este nuevo elemento les permitía trabajar más tiempo sin presentar problemas.

En esta generación, IBM lanza sus primeros modelos de computadoras basadas en el uso de transistores, la 1401 resultó ser una de las más vendidas. Ésta era aproximadamente siete veces más rápida que el modelo 650 de la primera generación. Sin embargo, IBM no era la única empresa dedicada a la construcción y venta de computadoras, HoneyWell se colocó como el primer competidor durante la segunda generación de computadoras y pronto hubo otras empresas en la misma rama.

Las computadoras de la segunda generación se utilizaban para gestionar los sistemas de reservación de aerolíneas, control de tráfico aéreo y simulaciones de uso general. En las empresas comenzaron a usarse en tareas de almacenamiento de registros como manejo de inventarios, nómina y contabilidad. Fueron

⁸ G. Ferreyra, op. cit., p. 33.



computadoras de esta generación las que utilizó la Marina de EU para crear el primer simulador de vuelo (Whirlwind I).

Con los progresos de la electrónica y los avances de comunicación con las computadoras en la década de los 1960, surge la **tercera generación** de las computadoras. El principal descubrimiento fue el primer “circuito integrado” que consistió en empaquetar cientos de transistores en un delgado chip de silicio.

En lo que respecta a la programación en esta generación proliferan los lenguajes de alto nivel (más cercanos al lenguaje humano), los cuales llegaron a clasificarse en tres tipos: los comerciales (de los cuales el COBOL y el RPG eran los que habían tenido mayor aceptación), los científicos (el FORTRAN era el de mayor uso y el PASCAL el favorito en los principales centros de enseñanza), y los de uso general, entre los cuales destacaban el PL/1, el BASIC y el C. También se destaca la introducción de programas para facilitar el control y la comunicación entre el usuario y la computadora (sistemas operativos).

Se empiezan a utilizar los medios magnéticos de almacenamiento (cintas magnéticas de 9 canales o discos rígidos). Aunque algunos sistemas aún usan las tarjetas perforadas para la entrada de datos, las lectoras de tarjetas ya alcanzan una velocidad respetable.

Las computadoras nuevamente se hicieron más pequeñas, más rápidas, desprendían menos calor y eran energéticamente más eficientes. La IBM 360 se lanzó al mercado en abril de 1964; y tuvo tal impacto, que se fabricaron más de 30 000. Fue entonces cuando IBM se conoció como sinónimo de computación. En ese mismo año, *Control Data Corporation* presenta la supercomputadora CDC 6600, considerada la computadora más poderosa de la época ya que tenía la capacidad de ejecutar unos tres “millones de instrucciones por segundo” (mips). Los circuitos integrados permitieron a los fabricantes de computadoras incrementar la flexibilidad de los programas, y estandarizar sus modelos.



En esta generación se logró una disminución importante en el tiempo ocioso de la unidad central de procesamiento, introduciendo el modelo de procesamiento concurrente. Bajo este esquema varios programas pueden residir simultáneamente en la memoria, pero uno sólo de ellos utiliza el procesador central en un momento dado, cuando el programa se detiene para esperar una operación de entrada o salida de datos, otro programa toma su lugar evitando así tiempos muertos de la unidad central. Este modelo permite el mayor aprovechamiento de los recursos de la computadora.

La fecha que marca el límite entre la tercera y la **cuarta generación** de las computadoras es 1971, cuando *Intel Corporation*, que era una pequeña compañía fabricante de semiconductores ubicada en Silicon Valley, presenta el primer “microprocesador” (chip) de 4 bits. Este chip contenía 2 250 transistores en aproximadamente 4 x 5 mm y fue bautizado como el 4004.

Silicon Valley (Valle del Silicio) era una región agrícola al sur de la bahía de San Francisco, que por su gran producción de silicio, a partir de 1960 se convierte en una zona totalmente industrializada donde se asienta una gran cantidad de empresas fabricantes de semiconductores y microprocesadores. Actualmente es conocida en todo el mundo como la región más importante para las industrias relativas a la computación: creación de programas y fabricación de componentes.⁹

Los microprocesadores representaron un gran adelanto de la microelectrónica, son circuitos integrados de alta densidad y con una velocidad impresionante. Las microcomputadoras fabricadas con base en estos circuitos son extremadamente más pequeñas y baratas, por lo que su uso se extendió al mercado industrial. Las PC (computadoras personales), nombre con el que se conocen a las microcomputadoras, son en la actualidad elementos comunes en oficinas, empresas, domicilios particulares, escuelas, etc., que se caracterizan principalmente por su bajo costo y su simplicidad de uso, mismas que han adquirido proporciones enormes y que han influido en la sociedad en general.

⁹ Ibid., p. 35.



Otra de las áreas que logró un gran avance en esta generación, fue la referente a la interacción hombre–máquina. Se diseñaron interfaces gráficas más “amigables” (iconos para representar funciones, ventanas para visualizar información, etc.) mismas que facilitaban la comunicación con el usuario. También se crean otras aplicaciones, tales como los procesadores de palabra, las hojas electrónicas de cálculo, paquetes gráficos, etc. Las industrias del software de las computadoras personales crecen con gran rapidez, Gary Kildall y William Gates (creadores de CP/M y de los productos de Microsoft) se dedicaron durante años a la creación de sistemas operativos y métodos para lograr una utilización sencilla de las microcomputadoras.

Debido al gran desarrollo logrado en la miniaturización de los circuitos integrados, fue posible llevar el poder de la computación al escritorio de una oficina y hasta un maletín, incluso actualmente en los Pocket PC (computadoras de bolsillo). Sin embargo, sería equivocado suponer que las grandes computadoras han desaparecido; por el contrario, su presencia era ya ineludible en prácticamente todas las esferas de control gubernamental, militar y de la gran industria. Las enormes computadoras de las series CDC, CRAY, Hitachi o IBM por ejemplo, fueron desarrolladas para realizar varios cientos de millones de operaciones por segundo.

La creación en 1982 de la primera supercomputadora con capacidad de proceso paralelo, diseñada por Seymour Cray, y el anuncio por parte del gobierno japonés de un proyecto que tenía por objetivo aplicar la “inteligencia artificial” (AI, *Artificial Intelligence*) sirven como parámetro para identificar una **quinta generación** de computadoras.

El proceso en paralelo es aquél que se lleva a cabo en computadoras que tienen la capacidad de trabajar simultáneamente con varios microprocesadores. Aunque en teoría el trabajo con varios microprocesadores debería ser mucho más rápido, es necesario llevar a cabo una programación especial que permita asignar diferentes tareas de un mismo proceso a los diversos microprocesadores que intervienen. También se debe adecuar la memoria para que pueda atender los requerimientos de los procesadores al mismo tiempo. Para solucionar este problema se



tuvieron que diseñar módulos de memoria compartida capaces de asignar áreas de caché para cada procesador.¹⁰

En particular, el concepto de procesamiento paralelo está despertando bastante interés. El concepto de paralelismo aprovecha el hecho de la existencia de varios procesos que pueden ser divididos en varios más simples y pueden ser ejecutados independientemente. La mejora en el tiempo de ejecución es una de las ventajas evidentes del procesamiento en paralelo. Sin embargo, muchas veces se hace difícil paralelizar un proceso, ya que la mente humana está acostumbrada a pensar de forma secuencial.

El llamado "programa de la quinta generación de computadoras" tenía el firme objetivo de producir máquinas con innovaciones reales que dieran al usuario la capacidad de comunicarse con la computadora en un lenguaje más cotidiano y no a través de códigos o lenguajes de control especializados; además de estar compuestos por una gran cantidad de microprocesadores trabajando en paralelo, también podrían reconocer voz e imágenes y se esperaba, que aprendieran la habilidad para tomar decisiones con base en procesos de aprendizaje fundamentados en "sistemas expertos" e "inteligencia artificial". Este proyecto japonés supuestamente debería haberse concluido en 1992; sin embargo, aún no se han visto cumplidas todas las expectativas.

Pero en otros aspectos el avance ha sido significativo, por ejemplo: el almacenamiento de información ha crecido de manera exponencial al utilizar dispositivos magneto ópticos con capacidades de decenas de Gigabytes; el DVD (Digital Video Disk o Digital Versatile Disk) se convirtió en el dispositivo estándar para el almacenamiento de datos, video y sonido por esta misma razón. Otra expectativa realizada sin interrupciones durante esta generación es la conectividad entre computadoras, que con el advenimiento de la red Internet y del World Wide Web (www), a partir de 1994 ha adquirido una importancia vital en las empresas en general y, entre los usuarios particulares de computadoras. De ahí que se haya

¹⁰ Ibid., p. 39.



generado una competencia internacional por el dominio del mercado de la computación, y en un futuro se espera que la ciencia de la computación siga siendo objeto de atención prioritaria de gobiernos y de la sociedad en conjunto.

Se dice que la **sexta generación** se ha venido desarrollando desde principios de la década de los 90 por lo que estaríamos hablando de los más recientes avances tecnológicos, muchos de ellos aún en desarrollo, como la “inteligencia artificial distribuida”, “teoría de caos”, “sistemas difusos”, holografía”, transistores ópticos”, etc. Las sofisticadas computadoras de este tiempo se caracterizan por contar con arquitecturas combinadas Paralelo/Vectorial, con cientos de microprocesadores vectoriales trabajando al mismo tiempo. Se han creado computadoras capaces de realizar más de un millón de millones de “operaciones aritméticas de punto flotante por segundo” (teraflops).

Hoy en día, las redes de área mundial siguen creciendo utilizando medios de comunicación a través de fibras ópticas y satélites, con anchos de banda impresionantes eliminando barreras de tiempo y distancia.

Bibliografía del tema 1

Beekman, George *Computación & Informática hoy, una mirada a la tecnología del mañana*. Wilmington, Delaware/México: Addison Wesley Longman. 1995.

Beekman, George *Introducción a la Informática*, 6ª ed., Madrid, Pearson. 2007.

Decker, Rick y Stuart Hirshfield *Máquina analítica, Introducción a las ciencias de la Computación con uso de la Internet*. México, Thomson. 2001.

Ferreya Cortés, Gonzalo *Informática para cursos de bachillerato*, México, Alfaomega. 2001.



Jamrich Parsons, June y Dan Oja *Conceptos de computación: nuevas perspectivas*. México, Thomson Learning. 2004.

Long, Larry E y Nancy Long *Introducción a las computadoras y a los sistemas de información*, México, Prentice Hall. 1999.

Norton, Peter *Introducción a la computación*, 3ª ed., México, Mc Graw Hill, 2006.

Actividades de aprendizaje

A.1.1 Elabora un glosario de términos relacionado con la informática y la computación, utilizados en el desarrollo del tema 1. Se sugiere incluya la traducción literal y su abreviatura (en dado caso) y una o dos de las más claras descripciones del término.

Ejemplo:

Computadora Personal *Personal Computer* (PC). 1. Microcomputadora destinada a trabajo individual o de escritorio. 2. Sistema individual que consta de monitor, unidad central de procesamiento y teclado.

A.1.2 Elabora un mapa de contenido con los nombres de los personajes mencionados en este tema con su correspondiente aportación en la historia y desarrollo de las computadoras.



A.1.3 Para entender las diferencias entre cada una de las generaciones completa la información del siguiente cuadro:

Generación / Características	Estructura interna	Dispositivo de entrada o almacenamiento	Programación	Ejemplo
Primera			lenguaje binario	
Segunda	transistores			
Tercera		Tarjeta o cinta perforada. Medios magnéticos		
Cuarta				PC
Quinta				Cray
Sexta	microprocesadores vectoriales		Paralelo/vectorial	

Cuestionario de autoevaluación

1. ¿Cuál fue el primer instrumento de cálculo descubierto en la antigüedad y que se sigue utilizando con fines educativos?
2. ¿A qué científico se debe el descubrimiento de los logaritmos y de las primeras tablas de multiplicar movibles?
3. ¿A quién se debe realmente la construcción de la primera máquina de calcular?
4. ¿Cuál es el nombre de la primera máquina de calcular reconocida?



5. La primera máquina de calcular reconocida, fabricada por Blaise Pascal ¿podía realizar las operaciones de multiplicación y división?
6. ¿Quién diseñó la primera calculadora que podía realizar las 4 operaciones básicas y extraer raíz cuadrada?
7. ¿Cuál fue el primer uso que se dio a las tarjetas perforadas?
8. Augusta Ada (1815-1853), Condesa de Lovelace está considerada como la primera _____
9. ¿Qué máquina es el antecedente de las computadoras digitales actuales?
10. ¿A quién se conoce como el padre de la computación y por qué?
11. ¿Quién utilizó el sistema de tarjetas perforadas para fines de cálculos censales?
12. ¿Qué compañía es el antecedente de la internacional IBM?
13. ¿Quién es el creador de la moderna teoría de la información y del termino *Bit*?
14. ¿De la integración de qué palabras nació el término *informática* y cómo se define?
15. ¿En qué siglo se dan los avances más grandes de la tecnología de las computadoras?
16. ¿Cuál es el nombre de la primera computadora electromecánica, desarrollada en Harvard?



17. ¿Cómo se llamó la primera computadora electrónica y quién contribuyó en su perfeccionamiento?
18. ¿Cuáles fueron los componentes básicos de las computadoras de la llamada *primera generación*?
19. ¿Cuál fue el invento más significativo que influyó en la construcción de las computadoras de la *segunda generación*, y en qué famosos laboratorios de los Estados Unidos fue descubierto?
20. La programadora Grace Murray Hopper de la Marina de los Estados Unidos participó activamente en la creación de un importante lenguaje de programación para aplicaciones de negocios. ¿Cuál fue el nombre del lenguaje?
21. La *tercera generación* de computadoras está señalada por la aparición del primer _____
22. ¿Cuál es la característica principal de las computadoras de la *cuarta generación*?
23. ¿En qué región de los Estados Unidos se concentraron las principales empresas de computación fabricantes de microprocesadores?
24. ¿Cuál tecnología es la base de la *quinta generación* de computadoras?
25. Describe brevemente en qué consiste el procesamiento en paralelo.
26. Mencione algunas de las tecnologías básicas representativas de la *sexta generación* de computadoras.



Examen de autoevaluación

Encierra en un círculo el inciso que responda a la pregunta o complete correctamente el párrafo:

1. Es una definición de Informática:
 - a) Ciencia que estudia la información procesada en una PC.
 - b) Programa que nos permiten editar textos e imágenes.
 - c) Ciencia que se encarga de estudiar todo aquello que tiene relación con el procesamiento de datos utilizando equipos de procesamiento automático de información (computadoras).

2. Fue la primera *máquina diferencial* realizadora de operaciones matemáticas (calculadora)
 - a) ENIAC
 - b) Máquina de Babbage
 - c) Mark I

3. ¿En qué país resurgió la idea de Babbage de crear una computadora de propósito general similar a la máquina analítica?
 - a) Alemania
 - b) Inglaterra
 - c) Estados Unidos

4. ¿Qué suceso histórico motivó a las naciones a patrocinar los proyectos computacionales?
 - a) La segunda guerra mundial
 - b) El renacimiento
 - c) La revolución industrial



5. ¿En qué universidad se construyó la primera computadora electromecánica con apoyo de la Marina estadounidense?
 - a) Universidad de Pennsylvania
 - b) Universidad de Harvard
 - c) Universidad de Cambridge

6. Es la primer computadora electrónica creada durante la guerra:
 - a) La máquina de Hollerith (1887)
 - b) La máquina de Babbage (1832)
 - c) ENIAC (1946)

7. Al hablar de “lenguaje binario”, nos referimos a que:
 - a) el ordenador (computadora) hace cálculos aritméticos y lógicos
 - b) toda la información que maneja el ordenador se traduce a 0 y 1
 - c) el ordenador lee los bits en pares

8. Son las características de la 1^a, 2^a y 3^a generación respectivamente:
 - a) Las máquinas utilizaban bulbos, se inicia el uso de tarjetas perforadas, se perfeccionan los microprocesadores
 - b) Uso de calculadoras, uso de discos flexibles, uso de multimedia
 - c) Uso de bulbos, los transistores sustituyen a los bulbos, uso de circuitos integrados

9. Cuando decimos que un lenguaje de programación es de “alto nivel” nos referimos a que dicho lenguaje:
 - a) Solo es manejado por altos ejecutivos o personal capacitado
 - b) Es más parecido al lenguaje binario o lenguaje máquina
 - c) Es más parecido al lenguaje humano



10. Una computadora es un dispositivo electrónico que permite la entrada de datos, los procesa, y entrega resultados por medio de algún dispositivo. Una vez que los datos son procesados por la computadora, éstos se convierten en información para el usuario.
- a) Todo es cierto
 - b) Sólo el primer enunciado es cierto
 - c) Todo es falso



TEMA 2. DESARROLLO DE SISTEMAS

Objetivo particular

Involucrar al alumno con la teoría de desarrollo de sistemas a partir del método de ciclo de vida de los sistemas de información.

Temario detallado

2.1 Ciclo de vida de los Sistemas de Información

2.1.1. Análisis

2.1.2. Diseño

2.1.3. Programación

2.1.4. Prueba y depuración

2.1.5. Conversión-Implementación

2.1.6. Documentación

2.1.7. Mantenimiento

Introducción

Cuando una organización no cuenta con la tecnología para automatizar sus procesos, ésta no puede estar en una posición competitiva, y pronto queda fuera del mercado. Para evitar esto, las empresas deben hacerse de herramientas que le ayuden a competir no sólo en la fabricación, sino también en la administración y en la necesidad de alinear la planeación de informática con los objetivos del negocio, por ello es importante contar con un plan adecuado para lograr las mayores ventajas del uso de la tecnología.



Es tiempo de fijar un poco la mirada en la forma en que la computadora se ha vuelto un instrumento de apoyo imprescindible en las actividades de las empresas. Gracias al desarrollo de sistemas de información las empresas han automatizado los procesos operativos, también pueden obtener información que sirve de apoyo a la toma de decisiones y, sobre todo, han logrado ventajas competitivas a través de su implantación y uso.

Pero, ¿qué es un sistema de información? Básicamente un sistema de información (SI) es un conjunto de elementos que interactúan entre sí con el fin de apoyar las actividades de una empresa o negocio. Los elementos que conforman un SI son de naturaleza diversa y normalmente incluyen: hardware, el recurso humano que interactúa con el sistema, los datos o información fuente que se introducirán en el sistema y los programas que son procesados.

Un sistema de información realiza tres actividades básicas: entrada de información (el SI toma los datos que requiere para procesar la información); almacenamiento de información (a través de esta propiedad de la computadora, el sistema puede recordar la información guardada en la sesión o proceso anterior); procesamiento de información (se efectúan los cálculos de acuerdo con una secuencia de operaciones preestablecida); y, finalmente, salida de información (el sistema saca la información procesada al exterior mediante alguna de las unidades de salida).

Existen tres estrategias para el desarrollo de sistemas: el método clásico del ciclo de vida de desarrollo de sistemas, el método de desarrollo por análisis estructurado y el método de construcción de prototipos de sistemas. Las tres estrategias de desarrollo tienen un uso amplio en organizaciones de todo tipo y tamaño; cada estrategia es efectiva cuando se emplea adecuadamente.¹¹

El desarrollo de sistemas de información adecuados requiere una mezcla de conocimientos de sistemas de cómputo, de sistemas de información, de cómo

¹¹ James A. Sen, *Análisis y Diseño de Sistemas de Información*. México, Mc Graw Hill, 1992 p. 50.



diseñar e implantar un sistema de información y cómo adquirir el sistema de cómputo necesario.

A continuación analizaremos el método del “ciclo de vida de los sistemas de información”, el cual nos muestra paso a paso el conjunto de actividades que se deben emprender para desarrollar e implantar un sistema de información.

2.1. Ciclo de vida de los sistemas de información

El concepto del ciclo de vida de un sistema de información es medular en un proyecto de desarrollo de sistemas, éste se apoya en un cuerpo extenso de una metodología y un conjunto bien definido de herramientas y técnicas para poder analizar, diseñar e implantar un sistema de información o una porción de un sistema de información.

Hay que entender que, durante su desarrollo, cada sistema se mueve a través de varias fases de un ciclo de vida, después del cuál solo funciona por varios años con un mínimo mantenimiento. El sistema se deteriora gradualmente hasta el punto en que cesa de funcionar por completo y se comienza un nuevo ciclo de vida con el desarrollo de un nuevo sistema.

El número de fases puede variar de autor a autor, pero según Scott, a las primeras cuatro fases del ciclo de vida de los sistemas (investigación preliminar, análisis, diseño, implantación) se les puede llamar “fases de investigación de sistemas” y dice que la preocupación más crítica en el trabajo de sistemas es el problema de obtener la cooperación completa de la gente, ya que los usuarios, los administradores y analistas de sistemas pudieran mostrar un comportamiento que inhiba el proyecto de sistemas.

Si se siguen varios principios de desarrollo de sistemas, pueden aliviarse los problemas con personas, proporcionar la estructura necesaria para una



investigación de sistemas y aumentar en otras formas la probabilidad de que el resultado de una investigación de sistemas sea un sistema con éxito.¹²

Los principios que detalla Scott constituyen una parte de la teoría de desarrollo de sistemas y los considera de suma importancia para la terminación exitosa de un proyecto. Entre estos principios están los siguientes: Involucrar a los administradores y usuarios en el proyecto; planear el proyecto dentro del contexto de planeación a largo plazo; establecer los objetivos y alcances del sistema desde el inicio; identificar el verdadero problema; establecer una programación razonable de terminación del proyecto y establecer las prioridades de los proyectos.

El objetivo del análisis de sistemas es comprender situaciones, no resolver problemas. Por tanto, los buenos analistas hacen hincapié en la investigación y el cuestionamiento para conocer cómo opera el sistema e identificar los requerimientos que tienen los usuarios para modificarlo o proponer uno nuevo. Sólo después de comprender en su totalidad el sistema, los analistas están en posición de analizarlo y generar recomendaciones para el diseño del sistema.

La forma en que se lleva a cabo la investigación de sistemas es la que determina si se reúne la información apropiada. A su vez, tener la información correcta en la calidad de la aplicación. En otras palabras, el buen diseño de sistemas, ya sea que se desarrollen con el método SDLC o algún otro, comienza con la documentación del sistema actual y el diagnóstico apropiado de los requerimientos.

2.1.1. Análisis

El primer paso del análisis del sistema es la identificación de necesidades, en este proceso el analista se reúne con el cliente y/o usuario (un representante institucional, departamental o cliente particular), e identifican las metas globales,

¹² George M. Scott y Daniel Cohen, *Sistemas de Información*. México, Mc Graw Hill, 1996, p. 207.



se analizan las perspectivas del cliente, sus necesidades, requerimientos y otros puntos que puedan ayudar a la identificación y desarrollo del proyecto.

Un requerimiento es una característica que debe incluirse en un nuevo sistema. Ésta puede ser la inclusión de determinada forma para capturar o procesar datos, producir información, controlar una actividad de la empresa o brindar soporte a la gerencia. Es así como la determinación de requerimientos vincula el estudio de un sistema existente con la recopilación de detalles relacionados con él.¹³

A menudo los analistas, cuando emprenden por primera vez la investigación de un sistema, se encuentran en desventaja porque es probable que tengan poco conocimiento del área bajo estudio para la que más tarde deberán hacer recomendaciones importantes. Por consiguiente, es esencial para ellos adquirir en forma rápida y exacta hechos importantes relacionados con los requerimientos del sistema para lo cual pueden utilizar métodos como entrevistas, cuestionarios y revisión de registros, estos métodos son de gran ayuda si son utilizados en forma apropiada ya que cada método tiene sus ventajas y desventajas (ninguno es el más adecuado por sí mismo). Además, durante la fase de hallazgo de hechos, es muy importante efectuar la comprobación cruzada de la información obtenida.

El analista recopilará hechos como: el porqué y cómo se realizan ciertas actividades, qué datos se emplean en el trabajo, el tiempo que tardan, la frecuencia y el volumen de las actividades. El estudio de los controles de los sistemas permite al analista observar la forma en que es posible mantener en marcha, de manera aceptable, las funciones de la empresa. Los requerimientos básicos son aplicables tanto a sistemas de transacciones como a los de decisiones y, en general, forman parte de las necesidades de toda la organización.

El siguiente paso para el analista consiste en organizar los detalles obtenidos y determinar dónde la información es incompleta o inconsistente. A esta actividad suele denominársele “documentar los procesos y decisiones de la organización”,

¹³ J. A. Senn, op. cit., p. 122.



en esta fase es necesario identificar “condiciones” y “acciones” y saber qué información está disponible para sugerir las acciones que deben emprenderse cuando aparezcan combinaciones específicas de condiciones.

Las tres herramientas para documentar procedimientos y toma de decisiones son: “árboles de decisión”, “tablas de decisión” y “español estructurado”.

Los *árboles de decisión* son representaciones gráficas y secuenciales de las variables de decisión que indican qué condiciones considerar en primera instancia, cuáles en segunda y así sucesivamente. La raíz del árbol de decisión es el punto donde se inicia el análisis de una situación específica; las ramas indican la secuencia de decisión que conducen a la acción apropiada que es necesario emprender.

Las *tablas de decisión* relacionan condiciones y acciones por medio de “reglas de decisión”. Una regla de decisión establece las condiciones que deben satisfacerse para emprender un grupo particular de acciones, la regla de decisión incorpora todas las condiciones que al mismo tiempo deben ser verdaderas y no sólo una condición.

Existen cuatro formas de tablas de decisión: de entrada limitada, de entrada extendida, de entrada mixta y formas ELSE. Las tablas múltiples se utilizan en situaciones donde las decisiones son muy amplias o es deseable separar los procedimientos para seguir una condición particular cuando ésta se presente. Sin embargo, todas las formas se deben desarrollar sin redundancia ni contradicción.¹⁴

El *español estructurado* se emplea para declarar reglas de decisión. Los tres tipos de declaraciones utilizadas son: estructuras de secuencia, estructuras de decisión y estructuras de iteración. Estas declaraciones muestran acciones incondicionales, acciones repetitivas y acciones que ocurren sólo cuando se presentan ciertas condiciones. El español estructurado ofrece un camino conciso para resumir un procedimiento donde se deben tomar decisiones y emprender acciones.

¹⁴ Ibid., p. 152.



Por otra parte el resultado puede ser revisado con facilidad por otras personas para detectar y corregir tanto errores como equivocaciones. Los errores que permanezcan en el análisis después de realizar la investigación de sistemas, aparecerán durante el diseño y la implantación, donde será mucho más costoso corregirlos.

Algunas actividades complementarias que también deben documentarse y que para el cliente resulta importante tratar durante la etapa de análisis, consisten en realizar un análisis de riesgos, así como un análisis económico y técnico.

Muchas veces cuando se emprende el desarrollo de un proyecto de sistemas los recursos y el tiempo no son realistas para su materialización, lo cual podría desembocar en pérdidas económicas y frustración profesional.

La viabilidad y el análisis de riesgos están relacionados de muchas maneras, si el riesgo del proyecto es alto, la viabilidad de producir software de calidad se reduce, sin embargo se deben tomar en cuenta cuatro áreas principales de interés:

Viabilidad económica. Una evaluación de los costos de desarrollo, comparados con los ingresos netos o beneficios obtenidos del sistema desarrollado.

Viabilidad Técnica. Un estudio de funciones, rendimiento y restricciones que puedan afectar la realización de un sistema aceptable.

Viabilidad Legal. Es determinar cualquier posibilidad de infracción, violación o responsabilidad legal en que se podría incurrir al desarrollar el sistema.

Alternativas. Una evaluación de los enfoques alternativos del desarrollo del sistema.

El estudio de la viabilidad puede documentarse como un informe aparte para la alta gerencia.



El análisis económico incluye lo que llamamos el análisis de costo–beneficio; significa una valoración de la inversión económica comparado con los beneficios que se obtendrán en la comercialización y utilidad del sistema.

Muchas veces en el desarrollo de sistemas de computación estos son intangibles y resulta un poco dificultoso evaluarlo, esto varía de acuerdo a las características del sistema. El análisis de costo–beneficio es una fase muy importante, de ella depende la posibilidad de desarrollo del proyecto.

En el análisis técnico, el analista evalúa los principios técnicos del sistema y al mismo tiempo recoge información adicional sobre el rendimiento, fiabilidad, características de mantenimiento y productividad.

Los resultados obtenidos del análisis técnico son la base para determinar si se puede continuar con el proyecto, si hay riesgos de que no funcione o no tenga el rendimiento deseado.

Finalmente, toda la información obtenida debe plasmarse en un modelo de lo que será el software, representando todas las funciones y subfunciones de un sistema. El modelo se concentra en lo que debe hacer el sistema no en como lo hace, pero puede incluir notación gráfica, información y comportamiento del sistema.

Todos los sistemas basados en computadoras pueden modelarse como transformación de la información empleando una arquitectura de tipo entrada y salida.

En conclusión, durante la etapa del análisis, se traducen las necesidades del cliente en un modelo de sistema que utiliza uno o más de los siguientes componentes: software, hardware, personas, base de datos, documentación y procedimientos.



2.1.2. Diseño

El diseño de sistemas se define como el proceso de aplicar ciertas técnicas y principios con el propósito de definir un dispositivo, un proceso o un sistema, con suficientes detalles como para permitir su interpretación y realización física.

Uno de los objetivos del diseño de sistemas de información es proporcionar la especificación del bosquejo del sistema, esto es, las “características” del sistema que serán trasladadas en software para su uso por la organización. Estas especificaciones, denominadas “diseño lógico” del sistema, incluyen detalles sobre salidas, entradas, archivos, interacciones con bases de datos, controles y procedimientos. La “construcción física”, que es la etapa que sigue al diseño lógico (según Senn), produce el software, los archivos y un sistema que trabaja.

En otras palabras, cuando los analistas especifican las “características” de un sistema diseñan flujos de datos, almacenes de datos, procesos, procedimientos y controles; pero también describen los papeles que deben asignarse a todas las personas que tienen que ver con el nuevo sistema, como los usuarios, operadores de computadora y el personal de apoyo. Con frecuencia, los procedimientos explican el papel de los individuos. A continuación explicaremos un poco más a detalle el diseño lógico:

Diseño de la Salida. En este caso “salida” se refiere a los resultados e informaciones generadas por el sistema. Para la mayoría de los usuarios la salida es la única razón para el desarrollo de un sistema y la base de evaluación de su utilidad. Y aunque muchas veces los usuarios no operan el sistema de información y tampoco ingresan datos en él, sí utilizan la salida generada por el sistema.

Al diseñar la salida, el analista debe realizar lo siguiente: determinar qué información presentar; decidir si la información será presentada en forma visual, verbal o impresa y seleccionar el medio de salida; disponer la presentación de la información en un formato aceptable; y, decidir cómo distribuir la salida entre los posibles destinatarios (la “distribución” se refiere a la disposición de la información en una pantalla o documento impreso).



Diseño de la entrada. El analista debe decidir sobre qué datos ingresan al sistema, qué medios utilizar, la forma en que se deben disponer o codificar los datos, el diálogo que servirá de guía a los usuarios para dar entrada a los datos. También debe asignar una validación de datos y transacciones para detectar errores, así como los métodos para llevar a cabo esta validación y los pasos a seguir cuando se presente algún error.

Las decisiones de diseño para el manejo de las entradas especifican la forma en que los datos serán aceptados por la computadora para su procesamiento y también especifican los medios por los que los usuarios finales y los operadores darán instrucciones al sistema sobre las acciones que debe emprender.

También forman parte del diseño de entradas: la disposición de mensajes y comentarios en las “conversaciones en línea”; así como la ubicación de los datos, encabezados y títulos sobre las pantallas o documentos fuentes.

Los sistemas en línea incluyen un diálogo o conversación entre el usuario y el sistema. Por medio del diálogo, el usuario solicita servicios al sistema y le indica cuándo realizar cierta función. A menudo la naturaleza de la conversación en línea hace la diferencia entre un diseño exitoso y otro inaceptable.¹⁵

Diseño de archivos. El diseño de archivos incluye decisiones con respecto a la naturaleza y contenido del propio archivo. Durante el diseño de archivos se debe decidir qué datos deberán incluirse en el formato de los registros contenidos en el archivo; también la longitud de cada registro (con base en las características de los datos que contiene); así como la secuencia a disposición de los registros dentro del archivo (la estructura de almacenamiento que puede ser secuencial, indexada o relativa).

Es posible que el nuevo sistema no requiera el diseño de todos los archivos utilizados por la aplicación, esto sucede en caso de que existan “archivos maestros” que ya estén en uso por otras aplicaciones. En este caso puede ser que la nueva aplicación únicamente haga referencia al archivo maestro y los detalles

¹⁵ Ibid., p. 390.



del archivo queden incluidos en las especificaciones del diseño de la aplicación. Esto evita que el archivo vuelva a diseñarse.

Diseño de Interacciones con la Base de Datos. La mayoría de los sistemas de información, ya sea implantado en sistemas de cómputo grande o pequeño, interactúan con las bases de datos que abarcan varias aplicaciones. Por la importancia de las bases de datos en los sistemas, su diseño es establecido y vigilado por un administrador de base de datos (persona responsable del desarrollo y mantenimiento de la base de datos), en este caso el diseñador no construye la base de datos sino que consulta a su administrador para ponerse de acuerdo en el uso de ésta en el sistema.

Diseño de controles. El analista debe anticiparse a los errores que pudieran cometerse al ingresar los datos en el sistema o al solicitar la ejecución de ciertas funciones. Puede ser que algunos errores no tengan graves consecuencias, pero puede ser que otros ocasionen el borrado de datos o el uso inapropiado del sistema. Aunque exista la más mínima probabilidad de cometer un error serio, un buen diseño de sistema de información ofrecerá los medios para detectar y manejar el error.

Se pueden diseñar controles de entrada que proporcionen medios para que sólo los usuarios autorizados tengan acceso al sistema; medios para garantizar que las transacciones sean aceptables; medios para validar los datos (para comprobar su exactitud) y, medios para determinar si se han omitido datos que sean necesarios.

Diseño de procedimientos. Los procedimientos especifican qué tareas deben efectuarse al utilizar el sistema y quiénes son los responsables de llevarlas a cabo. Algunos ejemplos de procedimientos son: procedimientos para entrada de datos (métodos para la captura de datos de las transacciones y su ingreso en el sistema de información); procedimientos durante la ejecución (pasos y acciones emprendidos por los operadores del sistema y, en ciertos casos, por los usuarios



finales que interactúan con el sistema para alcanzar los resultados deseados); procedimientos para el manejo de errores (acciones a seguir cuando se presentan resultados inesperados); procedimientos de seguridad y respaldo (acciones para proteger al sistema y sus recursos contra posibles daños). Estos procedimientos deben formularse por escrito y formar parte de la documentación del sistema.

Otros objetivos de diseño incluyen: proporcionar especificaciones detalladas para el software y asegurar que el sistema proporcione apoyo a las actividades de la empresa y cumpla con los requerimientos de los usuarios finales.

El *diseño del software* es un proceso y un modelado a la vez. El proceso de diseño es un conjunto de pasos repetitivos que permiten al diseñador describir todos los aspectos del sistema a construir. A lo largo del diseño se evalúa la calidad del desarrollo del proyecto con un conjunto de revisiones técnicas.

El diseño debe implementar todos los requisitos explícitos contenidos en el modelo de análisis y debe acumular todos los requisitos implícitos que desea el cliente. Además, se considera una guía que puedan leer y entender los que construyan el código y los que prueban y mantienen el software.

La importancia del diseño del software se puede definir en una sola palabra: “calidad”, pues es dentro del diseño donde se fomenta la calidad del proyecto. El diseño es la única manera de materializar con precisión los requerimientos del cliente.

Existen criterios que se han establecido para evaluar la calidad del diseño, en ellos se establece que el diseño: presente una organización jerárquica que haga un uso inteligente del control entre los componentes del software; sea modular, es decir, se debe hacer una partición lógica del software en elementos que realicen funciones y subfunciones específicas; contenga abstracciones de datos y procedimientos; produzca módulos que presenten características de funcionamiento independiente y; que además conduzca a interfaces que reduzcan la complejidad de las conexiones entre los módulos y el entorno exterior.



El proceso de diseño del software exige buena calidad a través de la aplicación de principios fundamentales de diseño, metodología sistemática y una revisión exhaustiva.

La información detallada del diseño se proporciona a los programadores para comenzar la fase de desarrollo de software.

2.1.3. Programación

En algunas organizaciones existe una separación entre las responsabilidades del programador y las del analista, pero por lo general se prefiere que ambos compartan responsabilidades en el proyecto y exista una adecuada comunicación entre ellos ya que una vez comenzada la fase de programación, el diseñador contesta preguntas, aclara dudas y maneja los problemas que enfrenta el programador cuando utiliza las especificaciones de diseño.

El encargado de desarrollar el software puede instalar (o modificar y después instalar) software comprado a terceros o escribir programas diseñados a la medida del solicitante. La elección muchas veces depende del costo de cada alternativa, en otras ocasiones depende del tiempo disponible para escribir el software y de la disponibilidad de los programadores ya que no siempre el programador (o el analista programador) pertenece al personal de la empresa. Las empresas pequeñas, que no cuentan con programadores, por lo general contratan servicios externos de programación.

La programación es parte importante de la investigación de sistemas. Las especificaciones del diseño de los programas deben desarrollarse con cuidado usando diagramas de flujo y otras técnicas, y las actividades de programación deben ser monitoreadas. Son frecuentes los casos en que el esfuerzo de programación requerido es mucho mayor que lo que se anticipó, y los programas terminados no satisfacen la calidad esperada. Si se implanta un sistema mal desarrollado, puede ocasionar pérdidas de recursos o causar errores en los archivos de la computadora ocasionando facturas incorrectas.



Los procedimientos utilizados para controlar el desarrollo de programas son similares tanto al desarrollar los nuevos programas como al modificar los ya existentes (un procedimiento de control enlista las actividades desde el diseño del programa hasta su implantación sustituyendo al anterior).

Es recomendable que se destine una computadora (que no sea usada en los procesos de rutina) para realizar la programación, esto facilita la escritura de programas en línea lo cual aumenta la eficiencia del programador, y elimina la posibilidad de que éste altere o interfiera con el software usado para el procesamiento normal de datos mientras desarrolla sus propios programas.

Los esfuerzos de programación deben conducirse con eficiencia, la que se promueve con un buen “sistema de administración de proyectos”. La eficiencia también se mejora con las técnicas de “programación estructurada”. Esta última reduce un programa (que de otra forma sería monolítico) a varios módulos pequeños unidos para el procesamiento. Cada módulo sólo tiene un punto de entrada y uno de salida (una interfase con un módulo precedente y una con uno siguiente), sin necesidad de usar ramificaciones, como lo ejemplifican las instrucciones GO TO de los programas. La lógica de los programas estructurados es mucho más simple y sencilla de entender que la de otros programas, lo que permite su escritura y revisión fácilmente.

El diseño, la documentación y los estándares de definiciones de datos también son ingredientes importantes en la eficiencia de programación. Los estándares de diseño tratan de los asuntos tales como los formatos de archivos y las convenciones de diagramas de flujo. Los estándares de documentación ponen en papel los requerimientos para las descripciones narrativas, de diagramas de flujo y otras descripciones de programas. Un diccionario de datos reúne los estándares de datos que eliminan definiciones múltiples de los datos de los mismos artículos, que de otra forma podrían estar presentes en diferentes archivos y programas.¹⁶

¹⁶ Véase, George M. Scott y Daniel Cohen. *Sistemas de Información*. México, Mc Graw Hill, 1996, p. 278.



Siempre va a ser necesario un programa de estándares, sobre todo en pequeñas instalaciones de procesamiento de datos donde puede ser que sólo una persona esté relacionada con un sistema; en caso de que esa persona se vaya, la documentación estandarizada puede permitir, a otra persona, entender el sistema con mayor rapidez. Sin embargo, mantener los estándares se convierte en algo molesto bajo la presión de las fechas límites de terminación de sistemas. Para asegurar la continuidad de un programa de estándares, se recomienda preparar un manual que detalle los estándares de la instalación. Los analistas y programadores deben entenderlos y asegurarse de su cumplimiento. Por supuesto, estos estándares requieren revisiones periódicas.

Otra forma de mejorar la eficiencia del desarrollo de programas es mediante el uso de “terminales en línea” para realizar la programación. Se supone que las eficiencias de la programación en línea duplican la productividad de los programadores.

La capacidad de desarrollar programas de procesamiento con sutileza y hacer cambios poco a poco y en su momento, en general es la base principal sobre la cual la administración evalúa el desempeño de procesamiento de datos ya que desafortunadamente es muy difícil lograr, en forma global y eficiente, el desarrollo y el mantenimiento de los programas.

Una vez concluido el desarrollo del programa, el programador es responsable de documentar adecuadamente los programas (debe proporcionar una explicación de cómo y por qué ciertos procedimientos se codifican en determinada forma). La documentación es esencial para probar el programa y llevar a cabo el mantenimiento una vez que la aplicación se encuentra instalada.

2.1.4. Prueba y depuración

Dependiendo del tamaño de la empresa que usará el sistema y el riesgo asociado a su uso, puede hacerse la elección de comenzar la operación del sistema sólo en un área de la empresa (como una “prueba piloto”), que puede llevarse a cabo en



un departamento o con una o dos personas. Cuando se implanta un nuevo sistema lo aconsejable es que el viejo y el nuevo funcionen de manera simultánea o paralela con la finalidad de comparar los resultados que ambos ofrecen en su operación, además dar tiempo al personal para su entrenamiento y adaptación al nuevo sistema.

Durante el proceso prueba se deben implementar todas las estrategias posibles para garantizar que en el uso inicial del sistema este se encuentre libre de problemas lo cual se puede descubrir durante este proceso y llevar a cabo las correcciones de lugar para su buen funcionamiento.

2.1.5. Conversión - Implementación

Esta fase se refiere al proceso de instalación del sistema diseñado, incluyendo cualquier equipo y software comprados. Las actividades de esta fase varían mucho de autor a autor, ya que algunos consideran que dentro de la fase de implementación (también llamada implantación) se encuentra la programación, las pruebas y el uso del nuevo sistema; sin embargo, siguiendo con la metodología del ciclo de vida, solo incluiremos las actividades que no hemos tratado en las fases anteriores, como son: el entrenamiento del personal, la preparación del lugar para la instalación del equipo, la conversión de archivos y del sistema, la aceptación del nuevo sistema y su evaluación.

Educación y entrenamiento de personal. Es común que el personal (operadores de entrada) requiera de entrenamiento para utilizar adecuadamente el nuevo sistema, los programadores también pueden necesitar de entrenamiento para completar la implementación del sistema o para mantenerlo, y los usuarios deben educarse también acerca de las capacidades del sistema, así como para saber cómo reunir y preparar los datos para el mismo.

Es recomendable que la labor de entrenamiento y educación se comiencen tan pronto se conozcan las características generales del sistema, para que en cuanto el sistema esté listo, también las personas involucradas estén preparadas



para usarlo. De esta manera se trata de asegurar que el personal coopere y utilice el nuevo sistema. Durante las sesiones de enseñanza se describe el nuevo sistema, se introduce la nueva tecnología incorporada al nuevo sistema y se analizan las ventajas esperadas en relación con el sistema antiguo.

Preparación del lugar para la instalación del equipo. En el caso de que el sistema haya requerido equipo nuevo, debe prepararse un lugar para instalar la computadora. La seguridad física requiere cuidado de almacenamiento a prueba de fuego para archivos y programas críticos, cerraduras de seguridad en las puertas del lugar e instalaciones de detección y prevención de incendios. La seguridad física debe considerarse como parte de un programa general de seguridad que incluya seguridad con palabras clave y otras precauciones.

Los sistemas de computadoras grandes requieren un ambiente controlado cuidadosamente, incluyendo control de temperatura (sistemas de aire acondicionado), control de humedad y control de polvo. Algo de suma importancia es una fuente de poder continua y estable (corriente regulada). En general, es mínima la preparación que se requiere para instalar las microcomputadoras, pero las variaciones en el suministro de energía eléctrica pueden dañar sus dispositivos (aunque sean mínimas) lo cual se puede prevenir instalando dispositivos de regulación de variaciones. Generalmente, se prohíbe fumar cerca de las microcomputadoras ya que sus componentes electrónicos son sensibles al humo.

Aunque con frecuencia se pone muy poca atención a los factores de comodidad humana (ergonomía) al diseñar sistemas de información, cabe mencionar que éstos hacen menos fatigantes y más eficientes las operaciones en las estaciones de trabajo personales.

La instalación y verificación de un equipo nuevo y de los programas más complejos normalmente están supervisadas por los proveedores, de igual manera los ingenieros de la compañía vendedora instalan el equipo de hardware y realizan pruebas para verificar que funcione correctamente.



Conversión de archivos. Si se diseñan nuevos archivos para reemplazar los actuales, los datos de estos últimos deben convertirse para que sean compatibles con el nuevo formato y estructura del archivo, y después deben capturarse en los nuevos. Esta conversión se puede hacer manualmente o con ayuda de programas de conversión; la decisión de hacerlo de una u otra manera es muy importante y muchas veces depende del tamaño del proyecto.

Una conversión realizada manualmente requiere imprimir los contenidos de los archivos de datos actuales y utilizar estas impresiones como la base para recuperar los datos en forma alterada según lo requieran los nuevos estándares del archivo, al realizar esta tarea es común que se cometan errores de captura. De otro modo tendrían que escribirse programas para convertir los datos de los archivos existentes al formato de los nuevos, crearse otros programas para transferir electrónicamente datos, de los archivos viejos a los nuevos; la desventaja es que estos programas quedarían sin valor una vez terminada la conversión.

Conversión de sistemas. Cuando el nuevo sistema (o una parte de él) está listo para su operación. Se usa uno de cuatro métodos de arranque: operaciones paralelas, arranque directo, arranque en fases o, estudio piloto.

La menos riesgosa y más frecuentemente usada en casos de reemplazo de un sistema es la de operaciones en paralelo. El sistema se corre “paralelamente” con el antiguo por un periodo de días, semanas o hasta meses si fuera necesario; es decir, tanto el nuevo sistema como el antiguo procesan las mismas transacciones. Un beneficio de las operaciones paralelas es que proporciona la seguridad de que el nuevo sistema funciona correctamente y de que logrará realizar el trabajo porque si se llegara a descubrir una falla o un defecto serio en el sistema cuando el antiguo ya se ha desmantelado, el resultado sería un retraso en el procesamiento de datos o un gasto adicional.

Un nuevo sistema puede tener varios errores, aunque esté muy bien diseñado e implantado, así que correr los mismos datos a través de ambos



sistemas en paralelo permite comparar los resultados del procesamiento entre los dos sistemas y detectar pequeñas fallas en el nuevo.

El arranque directo es la forma más riesgosa porque significa reemplazar el antiguo sistema con el nuevo, aún sin estar completamente seguros de que éste funcione correctamente. El arranque directo se usa cuando la conversión debe realizarse rápidamente, cuando no hay suficiente espacio o personal para llevar el sistema en paralelo, o bien, cuando existe un alto grado de confianza de que el nuevo sistema funcione satisfactoriamente.

Un arranque en fases comprende la secuencia de las actividades de investigación de sistemas para que se diseñen e instalen módulos separados en momentos diferentes; entonces cada uno puede comenzar la operación tan pronto como esté listo. Debido a que cada módulo sólo es una parte de todo el proyecto, cualquier problema de operación con un módulo específico es poco probable que interrumpa las operaciones con gravedad, y, cuando llega a pasar, en general puede corregirse antes de que el siguiente módulo comience a operar.

El arranque por estudio piloto se usa cuando eventualmente se implantan muchos sistemas idénticos o similares en varios departamentos, sucursales o divisiones. El sistema se diseña, pero inicialmente se implanta sólo en una de las unidades de la organización para probarlo. Cuando se encuentran y corrigen todos los problemas, el sistema se implanta en las otras unidades.

Aceptación del nuevo sistema. Cuando ya se ha usado cada una de las funciones de un nuevo sistema o programa y está libre de errores en su procesamiento, debe hacerse una revisión detallada en la que se determine sistemáticamente que los problemas anteriores han sido resueltos con el nuevo sistema. Después de esta revisión, el sistema se acepta con toda formalidad; ya no está “a prueba” y se usa en forma rutinaria para las operaciones. Finalmente todas las partes involucradas en el proyecto firman un documento en el que se especifica que el sistema está completo y les es aceptable. En el caso de haber requerido hardware o software de un proveedor externo, es recomendable realizar una evaluación del



desempeño del nuevo sistema (auditoría posterior) antes de firmar el documento que da por terminadas las obligaciones.

Evaluación del sistema o auditoría posterior. Se lleva a cabo para identificar puntos débiles y fuertes del sistema implantado. La evaluación ocurre a lo largo de cualquiera de las siguientes dimensiones: evaluación operacional, impacto organizacional o desempeño del desarrollo.

En *la evaluación operacional* se califica la manera en que funciona el sistema, esto incluye su facilidad de uso, tiempo de respuesta ante una necesidad o proceso, la manera en que se adecuan los formatos, en que se presenta la información, y su nivel de utilidad.

El impacto organizacional se mide a través de los beneficios operacionales para la empresa en el área de finanzas (costos, ingresos y ganancias), también con la eficiencia, el desempeño laboral e impacto competitivo obtenidos, así como la rapidez y organización en el flujo de información interna y externa.

El desempeño del desarrollo. Es la evaluación del proceso de desarrollo tomando en cuenta ciertos criterios como que el tiempo y esfuerzo en el desarrollo concuerden con el presupuesto y los estándares u otros criterios de administración de proyectos. Además se incluyen la valoración de los métodos y herramientas utilizados durante el desarrollo del sistema.

Los objetivos de realizar una auditoría posterior son los siguientes: determinar si el sistema resuelve el problema o problemas que se tratan de resolver; determinar hasta qué punto el sistema satisface las necesidades que se especificaron en el informe de requerimientos del sistema; determinar, después de la experiencia con el sistema, si existen nuevas oportunidades para mejorarlo; establecer si el sistema se terminó a tiempo y con el presupuesto asignado y, si no, por qué; y finalmente, evaluar el desempeño de los miembros del equipo de investigación de sistemas.



2.1.6. Documentación

Como ya se había mencionado, el programador es responsable de documentar adecuadamente los programas explicando cómo y por qué ciertos procedimientos se codifican en determinada forma. Una vez probado y aceptado el nuevo sistema, el programa debe quedar correctamente documentado, para posteriormente poder realizar el mantenimiento del mismo.

Toda la información que presentemos en los informes puede presentarse de diversas maneras, por ejemplo: diagramas de flujo, diccionario de datos, o cualquier otra técnica disponible.

Existen dos tipos de documentación: interna y externa.

La documentación interna se puede realizar utilizando terminología técnica para describir la estructura del sistema: datos y funciones. En ella se presenta información dirigida hacia el área técnica o de desarrollo de sistemas, incluyendo el posible origen de los datos, la descripción de algoritmos, estructuras de datos, flujo de control, etc.

Mientras que la documentación interna es concisa y está escrita en un nivel apropiado para un técnico o programador, la documentación externa se realiza para ser leída principalmente por quienes no conocen un lenguaje técnico.

La documentación externa se encarga de elaborar un informe del diseño del sistema en forma más amplia y detallada, ya que esta deberá estar dirigida a los usuarios y a los clientes en un lenguaje natural para describirles que es lo que el sistema hace.

2.1.7. Mantenimiento

Después de la aceptación del nuevo sistema comienza la parte de mantenimiento del ciclo de vida. Durante este periodo, que por lo común dura de 3 a 8 años, el



sistema se “mantiene” mediante cambios menores o mayores según sea necesario. En algún punto, se comenzará una nueva investigación de sistemas que tendrá como resultado el reemplazo del sistema. La necesidad de una nueva investigación es el resultado de una combinación de lo siguiente: el medio ha cambiado, por tanto, las principales funciones que realiza el sistema ya no son necesarias o no son suficientes; los cambios al sistema lo han hecho un parche de cambios, lo cual causa ineficiencia en el procesamiento, errores o fallas frecuentes; la tecnología ha avanzado y aunque el sistema aún funcione con eficiencia un nuevo sistema con tecnología nueva tendrá una mejor proporción costo-beneficio.



Bibliografía del tema 2

O' Leary, Timothy J. y Linda I. O'Leary *Computación básica*, México, Mac Graw Hill. 1997.

Senn, James A. *Análisis y diseño de sistemas de información*, 2ª ed., México, 1992.

Scott, George M. y Daniel Cohen *Sistemas de información*, México, Mc Graw Hill. 1996.

Actividades de aprendizaje

A.2.1 Elabora un cuadro sinóptico que muestre las fases del ciclo de vida de los sistemas con sus conceptos clave.

A.2.2 Investiga en una organización qué método se utiliza para el desarrollo de los sistemas y marca las diferencias que encuentres con el método estudiado en este tema (ciclo de vida del desarrollo de sistemas).

A.2.3 Elabora un resumen del tema: desarrollo de un sistema.

Cuestionario de autoevaluación

- 1) ¿Según Scott, a las primeras cuatro fases del ciclo de vida de los sistemas (investigación preliminar, análisis, diseño, implantación) se les puede llamar?
- 2) El primer paso del análisis del sistema es la identificación:_____
- 3) La viabilidad económica es:
- 4) Viabilidad técnica es:
- 5) Viabilidad legal es:
- 6) ¿Cómo se define el diseño de sistemas?



- 7) Una vez concluido el desarrollo del programa, ¿de qué es responsable el programador?
- 8) ¿Qué es lo que se aconseja cuando se va a implantar un nuevo sistema?
- 9) ¿A qué se refiere la fase de conversión – implementación?
- 10) ¿Cuántos tipos de documentación existen?
- 11) ¿Comúnmente cuánto dura el periodo de mantenimiento?

Examen de autoevaluación

1. Es la etapa en el ciclo de vida del desarrollo de un sistema en el que pueden utilizarse diversos instrumentos (muestreo, estudio de datos y formas usadas por la organización, entrevista, cuestionarios, etc.), para determinar los requerimientos de información de la empresa:
 - a) Determinación de los requerimientos de información
 - b) Identificación de problemas, oportunidades y objetivos
 - c) Análisis de las necesidades del sistema

2. Durante esta fase, el analista de sistemas analiza las decisiones estructuradas por realizar, que son decisiones donde las condiciones, alternativas, acciones y reglas de acción podrían determinarse:
 - a) Implantación y evaluación del sistema
 - b) Identificación de problemas, oportunidades y objetivos
 - c) Análisis de necesidades del sistema

3. Actividad básica de un sistema de información, mediante la cual el sistema toma datos que requiere para su procesamiento:
 - a) Almacenamiento
 - b) Entrada
 - c) Salida



4. Capacidad del sistema de información para efectuar cálculos de acuerdo con una secuencia de operaciones establecida:
 - a) Procesamiento
 - b) Entrada
 - c) Salida

5. La función básica de un sistema informático consiste en:
 - a) Recibir datos, procesarlos y arrojar información
 - b) Imprimir la información de la manera más nítida y rápida posible
 - c) Ordenar a la computadora que procese los datos

6. Es el conjunto de hardware, software, personas, procedimiento y datos:
 - a) Red
 - b) Sistema
 - c) Análisis y diseño

7. El inicio del diseño de un sistema es el desarrollo de la estructura del mismo, se debe considerar como estructura de un sistema el agrupamiento ordenado de todos los elementos componentes de éste. ¿Cuál es el objetivo primordial que marcará la pauta de todo el sistema?
 - a) Objetivo particular
 - b) Objetivo general
 - c) Políticas

8. El ciclo de vida del desarrollo de sistemas en una organización es una forma de:
 - a) Diseñar reportes según las necesidades.
 - b) Desarrollar un sistema de información.
 - c) Crear formatos de captura.



9. Dos tareas en la fase de _____ es que el equipo de desarrollo se dedique a: definir el problema y la forma de proceder para el desarrollo de un sistema de información y seleccionar la mejor solución al problema y definir su función.

- a) Análisis de las necesidades.
- b) Implementación.
- c) Desarrollo.

10. Para determinar la aplicación de un sistema de información se consideran dos factores importantes:

- a) El orden y el tiempo de llegada de los datos.
- b) El significado y la forma de la información.
- c) El contenido y el determinismo de la información.



TEMA 3. INTRODUCCIÓN A LA ARQUITECTURA DEL SOFTWARE

Objetivo particular

Después de estudiar este tema, el alumno identificará y asociará el conjunto de patrones y abstracciones coherentes que proporcionan el marco de referencia necesario para dirigir la construcción del software hacia los objetivos de un sistema de información.

Temario detallado

- 3.1. Introducción a la arquitectura de sistemas de software
- 3.2. Requisitos de calidad
- 3.3. Documento de diseño
- 3.4. RTFS- Método de control del diseño
- 3.5. Introducción al diseño arquitectónico de la interfaz humano-computador
- 3.6. Fundamentos de la arquitectura de software
- 3.7. Patrones de arquitectura de software
- 3.8. Arquitectura de software para sistemas distribuidos

Introducción

Una vez que ha sido instalado un sistema de información, éste debe ser administrado generalmente por un analista de sistemas con la ayuda de un equipo de programadores y otros profesionales informáticos (dependiendo de la magnitud de la organización y del sistema). Estos profesionales utilizan una enorme variedad de herramientas y técnicas para desarrollar y administrar los sistemas.



Los informáticos son responsables de las herramientas software y los conceptos que hacen posible la existencia de otros programas. Para ello, se especializan en distintas áreas como la teoría, los algoritmos, las estructuras de datos, los conceptos y lenguajes de programación, la arquitectura de las computadoras, la inteligencia artificial, la ingeniería de software, etc.

Sin embargo; a pesar de los avances producidos en la informática, el estado actual del desarrollo de software está muy lejos de ser el ideal. Tanto sus desarrolladores como sus usuarios se enfrentan a dos grandes problemas: el costo y la falta de fiabilidad.

A medida que las computadoras han evolucionado, el costo de su hardware ha ido en descenso, cada año surgen máquinas más potentes y fiables a un precio cada vez menor. En comparación; los costos derivados del desarrollo de los programas han aumentado. Con frecuencia se escuchan historias de sistemas informáticos que costaron mucho más de lo que se presupuestó para ellos y que llevaron mucho más tiempo del estimado en desarrollarlos; en otros casos, los desarrolladores terminan por abandonar el proyecto debido al alto costo. Estos aumentos en los precios no significan que el nivel de fiabilidad del software incremente también puesto que los errores de programación siguen presentes y las técnicas actuales de desarrollo no ofrecen una seguridad plena de que el sistema funcione sin fallo en cualquier circunstancia. A medida que crece el número de instituciones humanas que dependen de las computadoras, se hace cada vez más importante y necesario que los informáticos encuentren formas de desarrollar software en el que las personas puedan confiar.

Por todo lo anterior, el punto tres de este temario está destinado a la Arquitectura del Software y debido a la imperante relación con el desarrollo de sistemas se analizarán detalladamente algunos de los puntos que revisamos durante el ciclo de vida de los sistemas de información: la calidad del sistema y la documentación del mismo. Los demás puntos enriquecerán el marco teórico correspondiente a las metodologías y técnicas para el diseño lógico.



3.1. Introducción a la arquitectura de sistemas de software

El concepto de arquitectura se usa de forma amplia y en campos muy distintos, pero en general, tiene mucho que ver con el diseño de estructuras.

En el campo del software, la arquitectura nos identifica los elementos más importantes de un sistema así como sus relaciones. Es decir nos da una visión global del sistema. Su importancia radica en que necesitamos arquitectura para entender el sistema, organizar su desarrollo, plantear la reutilización del software y hacerlo evolucionar.

Cómo determinar los elementos que definen una arquitectura es una tarea muy importante para los informáticos; generalmente las metodologías de desarrollo indican principios para identificar y diseñar una arquitectura, aunque por ahora la ayuda real que ofrecen es muy limitada al basarse en principios muy genéricos. Más adelante describiremos algunas técnicas que nos pueden resultar muy útiles para la construcción de arquitecturas software.

Las arquitecturas software no responden únicamente a requisitos estructurales, sino que están relacionadas con aspectos de rendimiento, uso, reutilización, restricciones económicas y tecnológicas, e incluso cuestiones estéticas.

Actualmente existen muchas metodologías de desarrollo de software, desde métodos muy “pesados” y burocráticos, métodos ajustables al proyecto y a las condiciones de desarrollo, hasta métodos “ligeros” que surgen como respuesta a los excesos “formales” de otros métodos.

Partiendo de los principios de tantas y diversas metodologías, resulta algo complicado sacar una visión unificada sobre el diseño arquitectónico; sin embargo; destacan una serie de elementos comunes en aquellas que más se centran en este tema. Uno de estos principios comunes es la existencia de una fase en la que se establece o diseña una arquitectura base; otro sería la altísima dependencia



que existe entre los casos de uso (interacciones entre el usuario y el sistema) y la arquitectura.

Desde un punto de vista arquitectónico, no todos los casos de uso tienen la misma importancia, destacando aquellos que nos ayudan a mitigar los riesgos más importantes y sobre todo aquellos que representan la funcionalidad básica del sistema a construir. Esta arquitectura base estará especificada por diagramas que muestren subsistemas, interfaces entre los mismos, diagramas de componentes, clases, descripciones diversas y, por el conjunto de casos de uso básicos. Este conjunto de especificaciones nos permiten validar la arquitectura con los clientes y los desarrolladores, y asegurarnos que es adecuada para implementar la funcionalidad básica deseada.

A partir de aquí, lo común sería desarrollar el sistema de forma iterativa hasta tenerlo funcionalmente completo.

Una visión alternativa sería identificar el tipo de sistema que se quiere construir; tal vez no haya dos aplicaciones iguales, pero sí existen claros paralelismos entre las aplicaciones construidas para resolver problemas similares. Estudiar aplicaciones del mismo tipo tiene muchas ventajas ya que nos ayuda a entender las necesidades del cliente y las soluciones ya encontradas por otros. Este método podría minimizar las desventajas de utilizar una metodología tradicional de desarrollo.

Gracias a la experiencia en otras ramas informáticas, desde el inicio del desarrollo de una aplicación, es posible hallar componentes que implementen tecnologías o ciertas funcionalidades e integrarlos dentro del proceso de desarrollo de software.

Por otra parte; identificar el tipo de sistema a construir nos permite examinar la arquitectura de sistemas ya construidos, comprender los requisitos a los que se enfrentan, y compararlos con los de nuestros clientes. Si tenemos en cuenta que en cualquier tipo de sistema (por ejemplo: portales web) existen necesidades similares, muchos de los componentes que se usan en su desarrollo suelen ser los



mismos (por ejemplo: *parsers Xml*, motores de transformación *Xslt*, componentes de acceso a datos, buscadores, carritos de la compra, gestores de contenidos, etc.).

Las metodologías que gestionen de forma directa las cuestiones arquitectónicas y estructurales, podrán obtener no sólo productos de mayor calidad, sino a un menor costo y en menos tiempo. Esto se debe a que los riesgos arquitectónicos del proyecto son menores y están mucho más controlados, y que al poder integrar una visión orientada a componentes, las posibilidades de reutilizar software ya desarrollado son mucho mayores, con las ventajas que ello implica.

Construir una arquitectura es una actividad que permite desarrollar ideas nuevas así como una oportunidad de usar la experiencia acumulada, siendo casi siempre responsabilidad del desarrollador crear un producto de calidad y por tanto conocer el tipo de sistema a construir. Afortunadamente para esto último, los lenguajes de patrones nos pueden proporcionar una inestimable ayuda.

En los inicios de la informática, la programación se consideraba un arte, debido a la dificultad que entrañaba para la mayoría de los mortales, pero con el tiempo se han ido desarrollando metodologías que permiten establecer los fundamentos para que analistas, diseñadores, programadores, etc. trabajen en una línea común que logre alcanzar los objetivos del sistema de información, cubriendo todas las necesidades y requerimientos.

3.2. Requisitos de Calidad

La calidad del software es uno de los principales objetivos estratégicos de las organizaciones, que dependen de los sistemas informáticos para un buen funcionamiento. Consiste en desarrollar productos lógicos que satisfagan las necesidades del usuario y que tiendan a cero defectos.

El *aseguramiento de la calidad* es la revisión de los productos y documentación relacionada con el software para verificar su cobertura, corrección, confiabilidad y facilidad de mantenimiento. Y por supuesto,



incluye la garantía de que un sistema cumple las especificaciones y los requerimientos para su uso y desempeño deseados.¹⁷

Los analistas usan cuatro niveles de aseguramiento de la calidad: prueba, verificación, validación y certificación. A continuación explicaremos brevemente en qué consiste cada uno de ellos.

Prueba. Es un proceso que puede resultar caro, pero crítico; que se lleva a cabo para demostrar que no hay errores en un programa (un error aparece cuando el sistema no produce los resultados esperados, no solamente son de sintaxis de programación). Sin embargo; es imposible que los analistas demuestren que un sistema está 100% limpio de errores; por lo que se tiende a cambiar el significado de la prueba, considerándola entonces como el proceso de ejecutar un programa con la intención de hallar errores, es decir, hacer que el programa falle, se entiende que la prueba es exitosa cuando se detecta un error.

Se usan dos estrategias de prueba: la “prueba de código”, en la que el analista desarrolla casos de prueba para ejecutar cada instrucción o ruta dentro de un programa; y la “prueba de especificación”, en la que el analista examina las especificaciones del programa y escribe los datos de prueba para determinar cómo opera el sistema bajo condiciones específicas.

Por otro lado, las pruebas pueden realizarse en dos niveles: el primero se trata de una “prueba parcial” en la que el analista, usando alguna de las estrategias anteriores, examina los programas que conforman un sistema; el segundo nivel sería la “prueba de sistemas” en la que se pretende hallar cualquier discrepancia entre el sistema y sus objetivos originales.

Existen también otras pruebas de tipo especial que se centran en hallar los defectos de operación en el sistema para prevenir su falla, ellas son: la prueba de carga máxima, almacenamiento, tiempo de ejecución, recuperación, procedimiento, y de factores humanos.

¹⁷ J. A. Senn, op. cit., p. 793.



A pesar de todo lo anterior, los analistas saben que un programa de prueba efectivo no garantiza la confiabilidad del sistema. La confiabilidad debe diseñarse en el sistema, por lo que no se puede hacer una prueba para ella; sin embargo, existe un segundo nivel de confiabilidad que tiene que ver con los resultados reales que el sistema entrega al usuario. En este nivel la confiabilidad del sistema se entrelaza con la ingeniería del software y su desarrollo.

Se dice que un sistema tiene confiabilidad si no produce fallas costosas o peligrosas al usarse de manera razonable, es decir, de tal forma que un usuario típico espera que sea normal.¹⁸

Verificación y validación. Continuando con la intención de hallar errores, la verificación se lleva a cabo ejecutando un programa en un ambiente simulado. La validación se refiere al proceso del uso del software en un ambiente no simulado para hallar sus errores.

La retroalimentación de la fase de validación generalmente produce cambios en el software para resolver los errores y fallas que se descubren. La dinámica es la siguiente: se elige un conjunto de instalaciones usuarias que ponen a trabajar el sistema en las actividades cotidianas procesando transacciones y produciendo las salidas normales del sistema; es decir, el sistema se pone a prueba, pero los usuarios están advertidos de que están usando un sistema que puede fallar. Vale la pena recalcar el hecho de que las transacciones y los usuarios son reales. Esta prueba puede extenderse varios meses, tiempo durante el cual pueden hacerse modificaciones en el software de acuerdo a las fallas encontradas.

La verificación también es conocida como “prueba alfa” y la validación como “prueba beta”.

Certificación. Es una garantía de lo correcto en un programa y cada vez es más importante para las aplicaciones de sistemas de información. Actualmente se tiende a comprar o rentar software comercial en vez de desarrollarlo dentro de la organización; sin embargo, antes de decidir adquirirlo de esa manera es necesario

¹⁸ Ibid., p. 765.



que dicho software requiera una certificación por parte del fabricante para garantizar, que realmente hace lo que el vendedor afirma. Las empresas involucradas en la certificación de software deben contar con personal altamente experimentado en los procesos correspondientes a las aplicaciones que examinarán. La dinámica consiste en que los especialistas examinen cuidadosamente la documentación del sistema para verificar lo que afirma el vendedor que el sistema hace y cómo lo lleva a cabo. Entonces ellos prueban el sistema contra esas afirmaciones. Si no se encuentran serias discrepancias o fallas, certifican que el software hace lo que la documentación afirma. Esta certificación no garantiza que el software sea apropiado para una u otra organización, esa responsabilidad es de la misma organización y su grupo de analistas.

3.3. Documento de Diseño

La documentación consiste en material que explica las características técnicas y la operación de un sistema. En el tema dos mencionamos la importancia que la documentación tiene para proporcionar entendimiento de un sistema a quien lo vaya a usar para mantenerlo como una fase del ciclo de vida de los sistemas de información. Otras utilidades de la documentación residen en permitir auditorías del sistema y para enseñar a los usuarios cómo interactuar con el sistema y a los operadores cómo hacerlo funcionar.

La documentación de programas explica las razones y la lógica de un programa e incluye descripciones narrativas, diagramas de flujo, listados de programas y otros documentos. La del usuario, en general es menos técnica y explica a los usuarios en forma general la naturaleza y capacidades del sistema y cómo usarlo; esta documentación casi siempre está en forma de manual.¹⁹

El software modular bien diseñado tiene mayor probabilidad de cumplir los requerimientos de facilidad de mantenimiento, confiabilidad y prueba (que es el

¹⁹ Geoge. M. Scott y Daniel Cohen, op. cit., p. 281.



objetivo del diseño) por lo que es de suma importancia que todo quede debidamente documentado.

La documentación del diseño debe contener una parte denominada justificación racional del diseño, donde se indiquen las cuestiones críticas y los compromisos que fueron considerados en la generación del diseño. Toda esta información va orientada a los clientes, a los usuarios y a otros desarrolladores a comprender cómo y por qué encajan ciertas partes del sistema.

Algunas de las interrogantes que debe responder la documentación del diseño son: ¿Este diseño es la solución al problema?, ¿Es un diseño bien estructurado y fácil de comprender?, ¿Puede hacerse algo para mejorar la estructura y la compresión del sistema?, ¿El diseño es transportable a otras plataformas? y, ¿Es reutilizable?

La documentación debe contener también la descripción de los componentes del sistema. Una de las secciones deberá indicar la forma en que interactuarán los usuarios con el sistema, para ello se debe incluir: Menús y otros formatos de presentaciones en pantalla; Interfaces hombre-máquina: teclas de fusión, descripciones en pantalla, teclados, uso de ratón o de los dispositivos requeridos; y, formatos de los informes. También especificaciones sobre la procedencia de los datos (entrada), indicar cómo se les dará forma y por qué medio serán almacenados; así como de la forma en que se presentarán los datos (salida).

Además, debe contener información acerca de las características de las funcionales generales, exigencias del performance, y tratamiento de defectos.

Otro de los aspectos importantes que debe contener la documentación tiene que ver con circunstancias como: el entorno de red (en caso de que el sistema lo requiera); en tal caso, se debe presentar la topología de la red y la forma en que los nodos de la red deben tener acceso a otros nodos o al servidor y la filiación de servicios incluyendo información respecto a restricciones de los nodos hacia el



servidor y recomendaciones sobre la integridad de la red, asegurándose de que los datos sean exactos y que puedan recuperarse después de una falla.

Otras secciones de la documentación de diseño pueden tratar aspectos especiales como: Localización y aislamiento de errores; reconfiguración del sistema y medidas especiales de seguridad.

3.4. RTFS - Método de control del diseño

Las revisiones técnicas formales (RTF) son parte de las actividades que garantizan la calidad del software. Principalmente se hacen por dos razones bien justificadas: porque el trabajo técnico necesita ser revisado y porque hay errores que son percibidos más fácilmente por otras personas que por los creadores, por lo que las revisiones se centran solamente en el producto (no en los desarrolladores) para saber si hace lo que debe hacer.

En realidad, las RTF son una actividad colectiva que permite ampliar la mirada sobre lo que se revisa (situación que se profundiza al ser aplicada por profesionales de distintos niveles y especialidades). Lo que se revisa son los distintos elementos que componen el software con el fin de señalar la necesidad de mejoras o señalar qué no hay que mejorar y conseguir un trabajo técnico más homogéneo.

A lo largo del proyecto se realizan revisiones informales que se llevan a cabo constantemente, sin tales revisiones la programación y comprensión de un proyecto serían imposibles. Sin embargo; es imprescindible realizar revisiones “formales” que agrupen diversos mecanismos de revisión y permitan establecer un marco común para la definición de distintas etapas de revisión en el ciclo de vida del software, esto no sólo está pensado para las etapas tempranas del ciclo de vida sino que también puede -y debe- ser utilizado en etapas como la de prueba de software y el mantenimiento.



Las revisiones formales tienen tres elementos: el informe escrito del estado del producto revisado; la participación activa y abierta de todos los del grupo de revisión; y la total responsabilidad de todos los participantes en la calidad de la revisión.

Las RTF deben estar específicamente orientadas; es decir; cada RTF se debe centrar en una parte muy bien delimitada del software total (en lugar de intentar revisar un diseño completo), de esta manera se hacen inspecciones para cada módulo o pequeño grupo de módulos. La razón de hacerlo de esta manera es porque al limitar el centro de atención de la RTF, la probabilidad de descubrir errores es mayor.

Se ha encontrado que llevar a cabo las RTF objetivamente tiene ciertas ventajas como: reducir sustancialmente el costo del software; tienen gran valor educativo para los participantes; y además, sirven para comunicar la información técnica y fomentar tanto la seguridad como la continuidad.

Esencialmente, las RTF se realizan con la finalidad de cumplir ciertos objetivos, éstos son: descubrir errores en la función, la lógica o la implementación; comprobar que el software bajo revisión cumple los requisitos de software; garantizar que el software está bajo los estándares; lograr uniformidad en el software y facilitar el manejo de proyectos.

Las RTF pueden llevarse a cabo durante las siguientes fases: después del desarrollo de las especificaciones; después del análisis y diseño del programa o después de las pruebas.

Una RTF se realiza bajo el siguiente procedimiento:

El productor informa al jefe de proyecto la terminación del producto.

El jefe de producto contacta al jefe de revisión y programan la reunión de revisión.



Se reúnen los revisores y el jefe de revisión para revisar el producto.

La reunión revisión es llevada a cabo por el jefe de revisión, los revisores y el productor.

Al final, todos los participantes deben decidir si: aceptan el producto sin posteriores modificaciones; rechazan el producto por errores; aceptan el producto provisionalmente.

Una vez tomada la decisión, todos firman.

3.5. Introducción al Diseño Arquitectónico de la Interfaz Humano-Computador

Una interfaz (también llamada “interfase”), es la frontera entre el usuario y la aplicación del sistema de cómputo (el punto donde la computadora y el individuo interactúan). Sus características influyen en la eficiencia del usuario, al igual que en la frecuencia de errores cuando se introducen datos o instrucciones.²⁰

La interfaz diseñada debe cumplir con los siguientes objetivos:

- Decir al sistema las acciones a realizar;
- facilitar el uso del sistema y
- evitar los errores del usuario.

Las características de la interfaz en los sistemas en línea (ya se había explicado este término en el tema 2.1.2) incluyen los dispositivos utilizados para introducir y recibir datos (teclado, ratón, pluma óptica, pantalla sensible al tacto, voz, etc); el diálogo (mensajes, pasajes, inducciones, respuestas) que incita y guía a los usuarios; y los métodos y patrones que se siguen al mostrar la información.

En un sistema en línea, es muy importante la manera en que se organiza la información para ser mostrada ya que los usuarios también reaccionan ante esto.

²⁰ J. A. Senn, op. cit., p.518



Normalmente, la forma en que se estructura el área física de un monitor, así como los métodos particulares para destacar y señalar datos, mejoran la lectura de la información mostrada.

La calidad de la interfaz determina, entre otras cosas: si el usuario acepta o no el sistema; si los diseñadores del sistema son elogiados o reprobados; y si un sistema tiene éxito o fracasa en el mercado o la empresa. En lo anterior está implícita la importancia del diseño arquitectónico de toda interfaz. Al diseñar la interfaz de un sistema o aplicación, se debe tener en cuenta el deseo del usuario de enfrentarse a algo *fácil, pero a la vez poderoso*.

Existen modelos diferentes para el desarrollo de interfaces:

Modelo de diseño. Consiste en representar el software de acuerdo a los datos, arquitectura, interfaz y procedimiento (especificación de requisitos).

Modelo de Usuario: Representa el perfil del usuario final (edad, cultura, etnia, educación, etc.).

Estos modelos permiten al diseñador de interfaces satisfacer un elemento clave del principio más importante del diseño de interfaces de usuario: conocer al usuario y conocer las tareas que se automatizarán.

Existen tres tipos de usuario: principiantes, esporádicos y frecuentes.

La percepción del sistema (modelo de usuario): es la idea que tienen los usuarios sobre la posible interfaz del sistema.

La imagen del sistema es un modelo que intenta mezclar lo que es la estructura del sistema con analogías de la vida real.

Por otro lado, el diseño visual está involucrado en la selección cuidadosa de elementos formales que sobresalgan en el diseño. Las soluciones “elegantes”, generalmente las más simples, revelan un entendimiento íntimo del problema y



que su esencia ha sido comprendida y representada. La simplicidad juega un papel central en los diseños “a tiempo”, por esto las soluciones que resuelven problemas de forma clara y económica traen consigo beneficios como: accesibilidad, reconocimiento, rapidez de percepción y uso continuo.

Aunque los aspectos que determinan el diseño de interfases gráficas de usuario son múltiples, los podemos agrupar en tres grandes grupos:

Interacción general: sobre la consistencia de comandos, protección del sistema y facilidades de ayuda y asistencia. Las directrices de diseño para una interacción general son: ser consistente; ofrecer una realimentación significativa; verificar cualquier acción destructiva no trivial, permitir una vuelta atrás fácil en la ejecución de la mayoría de las acciones; reducir la cantidad de información que debe ser memorizada entre acciones; buscar la eficiencia en el diálogo, el movimiento y el pensamiento; perdonar los errores; categorizar las actividades en base a su función y organizar la geografía de la pantalla convenientemente; proporcionar facilidades de ayuda sensibles al contexto; utilizar verbos de acción simples o frases verbales cortas para nombrar las ordenes.

Visualización de información: cómo el sistema presenta resultados intermedios o finales al usuario, o en su defecto alguna situación que requiera su intervención. Sus directrices son: Mostrar sólo aquella información que sea relevante en el contexto actual; no abrumar al usuario con datos, utilizar un formato de presentación que permita una asimilación rápida de la información; utilizar etiquetas consistentes, abreviaciones estándar y colores predecibles; permitir al usuario mantener el contexto visual; producir mensajes de error significativos; utilizar mayúsculas y minúsculas, tabulaciones y agrupaciones de texto para ayudar a la comprensión; utilizar ventanas (si están disponibles) para modularizar los diferentes tipos de información; utilizar representaciones analógicas para mostrar la información que es más fácil de asimilar bajo este tipo de representación; considerar la geografía disponible en la pantalla y utilizarla eficientemente



Entrada de datos: cómo el usuario se comunica con el sistema para proporcionar datos y establecer las condiciones de funcionamiento del sistema. Las directrices generales para el diseño de la entrada de datos son: minimizar el número de acciones de entrada de datos que debe realizar el usuario; mantener la consistencia entre la información visualizada y los datos de entrada; permitir al usuario personalizar la entrada de datos; la interacción también debe ser flexible y estar ajustada al modelo de entrada preferido por el usuario; desactivar ordenes que sean inapropiadas en el contexto actual; permitir al usuario controlar el flujo interactivo; proporcionar ayuda en todas las acciones de entrada de datos; eliminar las entradas innecesarias.

Las fases del proceso del desarrollo de interfaces de usuario son:

- Análisis de usuarios, tareas y entornos.
- Diseño de la interfaz.
- Implementación de la interfaz.
- Validación de la interfaz.

Se deben seguir las siguientes recomendaciones: establecer los objetivos e intenciones de cada tarea; hacer correspondencia entre cada objetivo con una secuencia de interacción; especificar la secuencia de acciones de tareas y subtareas; indicar el estado del sistema; definir mecanismos de control; mostrar la forma en como los mecanismos de control afectan el estado del sistema; indicar la forma en que el usuario interpreta el estado del sistema a partir de la información presente en la interfaz.

A medida que el software evoluciona, también lo hace la interfaz del usuario (el aspecto y el comportamiento de una computadora desde el punto de vista de un humano). En la actualidad, los usuarios emplean la mayoría de su tiempo en el trabajo con aplicaciones ya programadas, como los procesadores de texto, que simulan y aumentan las posibilidades de herramientas reales.



3.6. Fundamentos de la Arquitectura de Software

No podemos hablar de arquitectura del software sin relacionarnos con la ingeniería de sistemas, generalmente llamada “Ingeniería de Software”, ya que el diseño del software se encuentra en el núcleo técnico de la Ingeniería de sistemas cuyo objetivo es producir software de alta calidad. En la ingeniería de sistemas se ve al diseño como el “plano del software” y se destaca que las áreas en las que se aplica son: los datos, la arquitectura, las interfaces y los componentes.

Mientras que el diseño de datos transforma el dominio de información en estructuras de datos, el diseño arquitectónico define la relación entre elementos estructurales del software.

Se le denomina “diseño de interfaz” a la manera en que el software se comunicará dentro de sí mismo, con otros sistemas y con las personas.

El diseño de los componentes transforma elementos estructurales en descripciones procedimentales.

El proceso de diseño es un proceso iterativo que está basado en principios fundamentales para lograr su eficacia. Inicialmente se presenta en un nivel de abstracción alto y cada vez se refina más. La calidad de su evolución se evalúa mediante Revisiones Técnicas Formales (RTF).

Dentro de las directrices para la calidad del diseño se menciona que éste deberá presentar una estructura arquitectónica que sea creada con patrones de diseño reconocibles; que esté formada por componentes de calidad y funcionalmente independientes; que contenga interfaces que faciliten las conexiones y que se pueda implementar evolutivamente. Otra de las directrices indica que el diseño debe ser modular; además de que debe contener diferentes representaciones y derivarse del análisis de forma controlada.



Profundizando un poco en el proceso de diseño, explicaremos a qué se refieren los principios de: abstracción, refinación, ocultamiento de información y modulación que son básicos para lograr un buen diseño.

Abstracción: Es una descripción simplificada o especificación que enfatiza algunos de los detalles o propiedades del sistema, mientras suprime otros. Se refiere a los niveles de resolución de problema, los cuales pueden ser altos si se especifica en lenguaje del entorno del problema; o bajo nivel de abstracción (o inferior) si tiene una orientación procedimental. La abstracción puede ser de datos, procedimientos y control.

La abstracción es procedimental cuando se trata de una secuencia nombrada de instrucciones que tiene una función específica y limitada.

La abstracción es de datos cuando se hace referencia a una colección nombrada de datos que describe un objeto de datos.

La abstracción es de control si explicita el mecanismo de control de programa sin especificar los datos internos.

Refinamiento: Es la base del diseño. Es un proceso de elaboración que comienza con un nivel de abstracción alto y va descendiendo sucesivamente de nivel de abstracción hasta llegar a un nivel bajo. Durante el proceso de refinamiento se van obteniendo detalles tanto de procedimientos como de datos obteniendo mejores soluciones más fáciles de implementar. El refinamiento tiene como objetivo encontrar detalles de bajo nivel que podrían ser difíciles de plasmar.

El refinamiento hace que el diseñador trabaje sobre la sentencia original proporcionando cada vez más detalles a medida que van teniendo lugar sucesivamente todos y cada uno de los refinamientos.

Modularidad: Un método de diseño software se dice que es modular si ayuda a los diseñadores a construir sistemas software formados por elementos autónomos y



organizados en arquitecturas sencillas. Se trata de dividir el software en componentes nombrados y abordados por separado llamados “módulos”, que se integran para resolver los requisitos del problema.

La modularidad es el atributo de software que permite que un programa sea manejable intelectualmente, por lo que se deduce que los módulos son los componentes básicos de todo sistema y tienden a satisfacer a uno o más requerimientos.²¹

Un software monolítico (programa grande formado por un solo módulo) no puede ser entendido fácilmente por el lector.

Aunque la modularidad sea benéfica para un sistema, tampoco debe exagerarse en el número de módulos, estas guías ayudarán a hacer más efectiva la modularidad: *diseñar pocas interfaces* (en un sistema formado por N módulos, el número de conexiones entre ellos debe acercarse al número mínimo que al máximo); *procurar interfaces estrechas* (si dos módulos se comunican deben de intercambiar el mínimo de información posible); *lograr interfaces explícitas* (la comunicación entre dos módulos debe poder deducirse a partir del texto de ambos); *cumplir el principio de ocultamiento de información*: Los módulos de un sistema deben diseñarse de modo que la información contenida en ellos sea inaccesible a todos aquellos módulos que no necesiten tal información.

Uno de los conceptos fundamentales del diseño lo marca la arquitectura del software, que -como ya lo habíamos mencionado- hace referencia a la estructura global del software y a las formas en que ésta proporciona la integridad conceptual de un sistema, dicha estructura es “jerárquica” en forma de módulos.

La arquitectura de software debe ayudar a definir como interactúan los componentes de software entre sí y las estructuras de los datos.

²¹ Roger S. Pressman. *Ingeniería de Software. Un enfoque práctico*. 5ª ed., México, Mc Graw Hill, 2002 p. 153.



La jerarquía de control representa dos características, visibilidad y conectividad. La *visibilidad* es el conjunto de componentes de un programa que pueden ser invocados o utilizados sus datos por un componente aun de manera directa. La *conectividad* indica el conjunto de componentes que son accedidos de manera directa por otros componentes.

La partición estructural de una arquitectura de software puede ser horizontal (de acuerdo a las áreas de diseño) datos, procesos y control o bien vertical definiendo una jerarquía de módulos (los módulos se programan de tal forma que los datos no estén accesibles por otros módulos).

Se dice que un diseño modular es efectivo cuando éste ayuda a reducir la complejidad, facilita los cambios y ayuda a producir soluciones más sencillas.

Los tres tipos de módulos existentes son: secuencial, incremental y paralelo.

Por todo lo anteriormente expuesto, se llega a la conclusión de que la “independencia funcional” del diseño se adquiere cuando se desarrollan los conceptos de modularidad, abstracción y ocultamiento de información.

La independencia funcional se mide con base en dos criterios: cohesión y acoplamiento.

Cohesión: es una extensión del principio de ocultación de información, es deseable tener una alta cohesión. Esta se obtiene cuando el contenido de un módulo está diseñado para realizar una tarea sencilla sin depender de otros módulos; o bien que un módulo lleve a cabo sólo una función de procesamiento.

El acoplamiento se refiere a la fuerza de la relación entre módulos de un sistema. En general, los buenos diseñadores buscan desarrollar la estructura de un sistema de tal forma que un módulo tenga poca dependencia de cualquier otro módulo. Es necesario tener un bajo acoplamiento. El acoplamiento se mide en las relaciones



que guardan los módulos con sus interfaces de entrada y salida. Hay tres tipos de acoplamiento: común, de datos y control.

Finalmente, vale la pena recordar algunos consejos prácticos para el diseño de datos: es recomendable definir todas las posibles operaciones a realizar sobre los datos, posteriormente se deben refinar las estructuras de datos hasta tener representaciones de bajo nivel; se deben desarrollar bibliotecas útiles para la manipulación de datos; el lenguaje de implementación debe soportar tipos de datos abstractos; y, tener cuidado de no convertir el diccionario de datos, en “basurero de datos”.

A continuación, trataremos lo relativo a los patrones de arquitectura software y estilos arquitectónicos.

3.7. Patrones de arquitectura de software

El concepto de *Arquitectura de Software* tiene mucho tiempo de antigüedad, pero no fue hasta la década de los 90 que comenzó a utilizarse de manera formal; del mismo modo, analizando los sistemas se observó que existen patrones que se repiten conformando lo que se conoce como estilos arquitectónicos (un patrón es un modelo que podemos seguir para realizar algo, surgen de la experiencia de seres humanos de tratar de lograr ciertos objetivos).

Un patrón de arquitectura de software describe un problema particular y recurrente del diseño, que surge en un contexto específico, y presenta un esquema genérico y probado de su solución.²²

Un estilo arquitectónico define un conjunto de familias de patrones de software con una determinada estructura y restricciones.

²² Hanna Oktaba, *Introducción a Patrones* Facultad de Ciencias, UNAM, consultado el 30 de mayo en <http://www.mcc.unam.mx/~cursos/Algoritmos/javaDC99-2/patrones.html>



Por otro lado, con frecuencia se dice que la función define a la forma, es decir, que la estructura o la arquitectura de cualquier sistema está muy relacionada con lo que dicho sistema tiene que hacer.

Esta es la razón por la que los sistemas con objetivos similares comparten también una arquitectura común, unos procesos bien definidos, y un conjunto de elementos similares (patrones de diseño). Similar funcionalidad y servicio, similar estructura.

Al desarrollar un sistema que se encuadra dentro de cierto tipo, es muy útil consultar lenguajes de patrones que traten el dominio en el que estamos. Un lenguaje de patrones nos sirve como referencia conceptual del dominio del problema, ya que éstos parten como solución a un conjunto de casos de uso, e interacciones con actores específicos. Además constituyen también un marco conceptual en el diseño de la arquitectura de nuestros sistemas, ya que como la función define a la forma, sintetizan por lo general soluciones arquitectónicas y estructurales bien probadas y muy útiles dentro del tipo de problemas que modelan.

De alguna forma, los patrones permiten identificar y completar los casos de uso básicos expuestos por el cliente, comprender la arquitectura del sistema a construir así como su problemática, y buscar componentes ya desarrollados que cumplan con los requisitos del tipo de sistema a construir (es decir, permiten obtener de una forma sencilla la arquitectura base que se busca durante la fase de diseño arquitectónico).

Sin embargo, los lenguajes de patrones tampoco son la panacea y presentan muchas lagunas. Sobre todo, hay que recordar que todo este movimiento de documentación de diseño se origina a mediados de los noventa y que aun siendo mucho el trabajo realizado, no existe todavía ninguna estandarización sobre cómo abordar el desarrollo de estos lenguajes.



Estos lenguajes deberían, además, tener una visión orientada a la construcción de software y a constituirse como elementos integrables en el proceso de desarrollo de las aplicaciones.

Generalmente los patrones de diseño y arquitectura definen soluciones para medios repetitivos.

En este momento, podemos ver la arquitectura de software como una abstracción del sistema que nos permite observar su estructura y sus relaciones. Para el desarrollo del diseño arquitectónico se recomiendan seguir los siguientes pasos:

- a) Estructuración del sistema
- b) Modelado de control
- c) Descomposición modular

Por otro lado, existen diferentes estilos arquitectónicos, a continuación mencionaremos algunos de ellos.

La Arquitectura de Flujo de Datos parte del DFD para obtener una arquitectura del sistema:

- Se establece el tipo de flujo de información
- Se indican los límites del flujo
- Se convierte el DFD en una estructura del programa
- Se define la jerarquía de control mediante particionamiento.
- Se refina la estructura resultante utilizando heurísticas de diseño.

La arquitectura centrada en datos tiene como componente principal un repositorio, del cual surgen los demás componentes.

Las arquitecturas estratificadas son de las más utilizadas en la actualidad, dado que dividen las actividades y responsabilidades de sistemas por capas. El software más elaborado como los sistemas operativos, software de base, sistemas distribuidos y otros maneja variantes de esta arquitectura.

El diseño se debe refinar realizando cada uno de los siguientes pasos:



- a) Desarrollar una descripción del procedimiento para cada módulo.
- b) Desarrollar una descripción de la interfaz para cada módulo.
- c) Se definen las estructuras de datos generales y globales.
- d) Se anotan todas las limitaciones/restricciones del sistema.

Se debe refinar el diseño hasta que esté completo. Se recomienda completar la arquitectura con el diseño de interfaces.

Los patrones de diseño, también nos ayudarán a especificar las interfaces, identificando los elementos claves en las interfaces y las relaciones existentes entre distintas interfaces. El diseño de interfaces se refiere al estudio de las relaciones entre los usuarios y las computadoras para que un sistema se pueda ejecutar. El diseño de una interfaz puede definir el éxito de cualquier proyecto, ya que la utilización de cualquier interfaz de usuario depende de factores humanos. Existen algunas reglas de oro para el buen diseño de interfaces de usuario, éstas son: dar el control al usuario; reducir la carga de memoria del usuario; y, construir una interfaz consecuente. Los modelos de desarrollo de interfaces para el usuario las hemos tratado en el punto 3.5. de este tema.

3.8. Arquitectura de software para sistemas distribuidos

La capacidad de conectar a los usuarios con el recurso computacional por medio de comunicaciones es esencial para las organizaciones, ya que una inversión razonable en hardware y software hace que el acceso a las computadoras esté disponible a todas las personas que lo necesitan. Sin embargo, hay casos en que el usuario desea procesar datos en una instalación específica y después enviar datos en forma periódica a un sistema de cómputo en la instalación central (a menudo a lugares distintos y separados entre sí); además, los costos de comunicación están aumentando mientras que el de las computadoras decrece (por lo que es común que las empresas utilicen legiones de computadoras pequeñas). Éstas son algunas razones para que los profesionales de la informática tomen en cuenta los “sistemas distribuidos” en sus diseños de sistemas de información.



Un sistema distribuido interconecta los lugares que tienen recursos computacionales para: capturar y almacenar datos, procesarlos y enviar datos e información a otros sistemas, tales como un sistema central.²³

En general, se podría ver a un sistema distribuido como la unión de una tecnología de red y el uso de potentes máquinas multiprocesadores. Un sistema distribuido es una colección de computadoras independientes o autónomas que aparecen ante los usuarios del sistema como una única computadora.

Los sistemas distribuidos se caracterizan por contar con múltiples elementos de procesamiento y mecanismos de intercomunicación. En estos sistemas existe una independencia a fallos en los nodos de procesamiento, esto debido a que trabajan en un estado de compartición y un esquema de protección en diversas plataformas (heterogéneas).

Utilizar los sistemas distribuidos tiene objetivos como: compartir información y otros recursos entre más de un usuario; economizar el rendimiento (procesamiento y almacenamiento); mantener alta disponibilidad; obtener capacidad de crecimiento incremental y confiabilidad (si un nodo de procesamiento falla, el sistema sigue funcionando).

Una de las ventajas de los sistemas distribuidos, respecto a los centralizados, es la economía, pues es mucho más barato añadir servidores y clientes cuando se requiere aumentar la potencia de procesamiento.

La capacidad de crecimiento incremental se refiere a que se puede añadir procesadores al sistema incrementando su potencia en forma gradual según sus necesidades.

Se dice que tienen una mayor confiabilidad porque al estar distribuida la carga de trabajo en muchas máquinas la falla de una de ellas no afecta a las demás, el sistema sobrevive como un todo.

²³ James A. Senn, *op. cit.* p.723.



Además, cuando se opte por una arquitectura distribuida, se deben considerar distintos aspectos, entre otros:

- Escalabilidad
- *Performance*
- Activación
- Manejo de estados
- *Garbage Collection*
- Seguridad

Los distintos protocolos que trabajan en sistemas distribuidos cubren algunos de estos aspectos. Los modelos de programación tradicional han sido extendidos para aplicarlos en la programación distribuida.

El llamado a procedimiento convencional local se extendió al remoto (RPC). Al surgir programación orientada a objetos (POO), fue necesaria la invocación remota de métodos (RMI). RMI permite que un objeto que está dentro de un proceso, invoque los métodos de un objeto que está en otro proceso.

El modelo de programación basado en eventos para progresar a la versión distribuida necesitó extender la capacidad de sus objetos de recibir notificaciones de eventos que ocurren en otros objetos a los que quieren acceder.

El uso de interfases para la comunicación entre distintos módulos permite “esconder” implementación y ofrecer solamente una forma de comunicación con el módulo.

La implementación puede cambiar pero la interfase permite que los otros módulos sigan comunicándose con el “modificado” de igual forma, si esto es deseable.

En el modelo RPC, hablamos de interfases de servicio. En el modelo RMI, de interfases remotas.

Las de servicio, usadas en el modelo cliente servidor (C/S), ofrecen la forma de acceder a los servicios que brinda un servidor. La interfaz de servicio es la



especificación de los procedimientos que ofrece un servidor; define los argumentos de entrada y salida, etc.

La interfaz remota especifica los métodos de un objeto factibles de ser invocados por objetos de otros procesos. Define argumentos de entrada y de salida.

La diferencia entre ambas interfases es que en la remota, los métodos pueden transmitir objetos como argumentos y como resultados. También se pueden usar referencias de objetos remotos.

Cuando un lenguaje quiere usar RMI, debe considerarse cómo interpretar la definición de las interfases que indiquen cómo invocar los argumentos de entrada y salida. Un ejemplo de esto es Java RMI. Pero esta solución exige que los objetos que se quieren comunicar estén en Java.

Para que objetos implementados en lenguajes diferentes puedan invocarse, existen los lenguajes de definición de interfases (IDL). ORB (*Object Request Broker*) es el *middleware* que establece una relación cliente servidor entre objetos. A través del ORB un cliente invoca de forma transparente un método de un objeto servidor ya sea en la misma máquina o a través de la Red.

El ORB provee interoperabilidad entre objetos en ambientes distribuidos. Su tarea comprende: interceptar la invocación al objeto; encontrar el objeto invocado; transmitir los parámetros; invocar al método; y retornar los resultados.

El cliente no tiene que conocer la ubicación del objeto ni en qué lenguaje está escrito. Sólo necesita saber cómo invocarlo.

En las aplicaciones típicas cliente/servidor el desarrollador tiene dos alternativas: diseñar él mismo un protocolo o usar un estándar que depende del lenguaje, el transporte, etc. En ORB el protocolo es definido por el IDL.



Para poder implementar la reusabilidad, una característica clave de la tecnología de objetos, la capa de *middleware* es imprescindible para soportar la comunicación entre objetos. Así, la interfase del objeto sirve para especificar los métodos.

Quizás el usuario común, no desarrollador o que no trabaja a partir de datos que ofrecen aplicaciones en la red, no llegue a vislumbrar los cambios o los beneficios, pues la transparencia es uno de los conceptos más importantes de este modelo distribuido.

Pero desde el punto de vista del desarrollador, este modelo le ofrece la posibilidad de: escribir aplicaciones que interactuarán con distintos dispositivos sin tener que modificar su aplicación, permitiendo que su aplicación *e-commerce* sea accedida sin restricciones de este tipo; reutilizar software escrito por otros desarrolladores; y, lo más importante, reutilizar ese software desinteresándose por cómo y en qué lenguaje está escrito.

Si hablamos de interacción, no podemos obviar que la solución que se presente debe considerar la heterogeneidad antes mencionada. Esta solución debe prever su implementación transparente sobre ambientes multiplataforma.

Analicemos, por ejemplo, el tema de los dispositivos.

La innovación frecuente (consideremos el progreso de la última década en cuanto a teléfonos inalámbricos, teléfonos inteligentes, nuevos modelos de laptops, etc.) exige que el modelo actual tanto de aplicación como de comunicación, no sea suficiente.

Hasta ahora, con cada nuevo dispositivo, era necesaria la adaptación de la aplicación para integrarlo. Las soluciones que se brinden desde el servidor deben tener en cuenta a estos nuevos dispositivos del cliente.



Una alternativa es desarrollar una solución para cada dispositivo, lo cual, obviamente, es ineficiente, considerando que la falta de escalabilidad no permitirá una rápida adaptación de la aplicación a nuevos dispositivos que puedan surgir (y surgirán) en el futuro.

Los dispositivos tienen, por ejemplo, diferentes resoluciones de pantalla. Además, en la mayoría de los casos, deben integrar datos provenientes de diferentes servidores.

El desafío es, entonces, desarrollar una aplicación que deba tener en cuenta esta variedad, y que sea reusable para nuevos dispositivos que surjan (lo cual es lo esperado).

Este objetivo está dentro de lo que se llama *ubiquitous connectivity* (algo así, como ‘conectividad para todos’, ‘entre todos’). En GLA000 se hace una referencia interesante de los *web services* como evolución y revolución. Por ejemplo, en un futuro, podemos pensar que una máquina expendedora de gaseosa, *wireless*, pueda contactar al proveedor para ordenar un pedido sobre una marca de gaseosa faltante. El “diálogo” se establecería entre dispositivos.

Tratando de explotar la ley de Moore y el abaratamiento del ancho de banda, pensar en un modelo distribuido basado en Internet, es hoy una posibilidad cierta.

El implacable progreso de la industria de la computación está mostrado en la Ley de Moore. En 1965, Gordon Moore, el presidente del fabricante de chips Intel, predijo que la potencia de un chip de silicio del mismo precio podría doblarse cada 18 meses durante al menos dos décadas. En la actualidad, su predicción se ha mostrado totalmente acertada.²⁴

Este nuevo modelo incluye componentes de software que usan protocolos abiertos de Internet, incluyendo HTTP para transporte y XML para la representación de la información. Es un modelo orientado a servicios WEB. Cualquier dispositivo y plataforma hoy puede trabajar con estos protocolos. De allí que la adaptación al nuevo modelo sea factible a corto plazo.

²⁴ Geoge Beekman. *Introducción a la informática*, 6ª ed., Madrid, Pearson. 2007 p.36



Y en cuanto a la seguridad, al usar HTTP siempre se puede usar SSL, para obtener un nivel básico, que puede complementarse con servicios de autenticación.

Para adaptarse a este modelo, la nueva generación de lenguajes de computadoras debe integrar Internet de una manera transparente.

Las plataformas de lenguajes tipo Java usan interpretadores en el momento de la ejecución para soportar las diferencias en el hardware cliente. Sus características de orientación a objetos no logran la independencia esperada del hardware ni los niveles de performance, ni la interoperabilidad.

La Web ha evolucionado desde contenidos estáticos, como documentos, a contenidos dinámicos, usando servidores de aplicaciones usando *business logic*, como programas CGI o JAVA y transacciones.

También usan nuevos protocolos como WAP (*Wireless Access Control*) y WML (*Wireless Markup Language*).

El popular modelo P2P en una comunicación de igual a igual es utilizado con frecuencia con algunas variaciones; por ejemplo, el modelo cliente-cliente (*peer to peer*) trata sobre la comunicación entre clientes, sin usar, imprescindiblemente, un servidor.

Como ocurre con los modelos, no es aplicable a todas las situaciones. Si el cliente debe ser autenticado o si los clientes quieren trabajar asincrónicamente, se hace necesario el uso de un servidor. También se debe tener en cuenta la capacidad de proceso del cliente: si no es suficiente, deberá recurrir a otras máquinas para el procesamiento de la información.

Para este modelo se debe garantizar que se cumplen las exigencias de un modelo integrado de distribución sobre Internet, se debe analizar:

- Manejo de identidad y autenticación del usuario y los servicios asociados



- Capacidad de trabajar en línea y fuera de línea
- Optimizar la forma de mostrar la información
- Libre uso de dispositivos
- Interacción con otros usuarios
- Servicios disponibles para el usuario de acuerdo a su personalización

La interfaz de un proceso es la especificación del conjunto de funciones que pueden invocarse sobre él. Esto es válido tanto para C/S como para comunicación entre iguales.

En cliente-servidor, cada proceso servidor es una entidad con una interfase definida.

Cuando tratamos con lenguajes OO, un proceso puede encapsular varios objetos. En el modelo OO los nuevos tipos de objetos deben ser instanciados y estar disponibles rápidamente para su invocación.

Cuando el lenguaje lo permite, los procesos distribuidos se organizan como objetos. Se pueden encapsular muchos objetos y las referencias a estos objetos se van pasando a otros procesos para que puedan invocar sus métodos. JAVA, con su RMI (invocación remota de objetos) usa este mecanismo.

Es muy importante en cuanto al diseño, la manera en que se distribuyen las responsabilidades en el modelo distribuido. Esta distribución puede ser estática (por ejemplo, un servidor de archivos se responsabiliza de los archivos no de proxys, mails, o páginas WEB) o más dinámica como es el modelo orientado a objetos donde se pueden instanciar nuevos objetos y nuevos servicios inmediatamente disponibles para ser invocados.

Las referencias a esos procesos deben armarse de tal forma que puedan realizarse invocaciones remotas (o lo que es equivalente, invocaciones a objetos remotos); es decir, invocarse sus métodos desde otros procesos. Cabe aclarar que estas invocaciones deberán ocurrir entre objetos en diferentes procesos sin importar que estén éstos en la misma computadora o en computadoras interconectadas. En consecuencia, a los objetos que pueden recibir invocaciones



remotas se les denomina objetos remotos. Por lo contrario, cuando se invocan métodos entre objetos que están en el mismo proceso, son invocaciones locales.

Analizando los objetos remotos comenzaremos por mencionar que para poder invocar un objeto, se debe tener una referencia a él. Y el objeto remoto debe tener una interfaz remota que diga cuál de sus métodos puede invocarse remotamente.

Cuando se trabaja con este tipo de objetos, se debe tener en cuenta que, por ejemplo, una vez que el objeto fue borrado, no deben resolverse futuras referencias, ni derivarlas a otros objetos.

En el ambiente C/S se habla de interfaz de servicio pues los servidores proporcionan servicios y para invocarlos debe ofrecer una especificación de los procedimientos disponibles. Define allí también los argumentos de E y de S.

En el modelo de objetos distribuidos, la interfaz remota especifica los métodos de un objeto disponible a ser invocadas por otros procesos. Define también argumentos de entrada y de salida.

Pero lo que hay que tener en cuenta, y es lo que lo diferencia del otro modelo, es que los métodos de interfases remotas pueden tener objetos como argumento y como resultado. También se pueden pasar referencias a objetos remotos.

Cuando un proceso cliente quiere invocar un método de un objeto remoto, debe asegurarse de estar invocando el método deseado. El proceso envía un mensaje de invocación al servidor donde está el proceso que alberga el objeto remoto. El mensaje especifica el objeto remoto de una manera que debe ser válida para el sistema distribuido. En un ambiente así, hay distintos procesos que alojan objetos factibles de ser invocados remotamente. La forma de referenciarlos debe asegurar la unicidad.

Una referencia a un objeto remoto es un identificador que es válido dentro del sistema distribuido.



Las referencias pueden transmitirse como argumentos y pueden devolverse como resultado de la invocación a métodos remotos. Incluso las referencias pueden compararse para saber si se refieren al mismo objeto. No sólo la referencia debe asegurar la unicidad con respecto a la ubicación: también es importante considerar el tiempo, es decir, la posibilidad de estar invocando una versión que no sea la más reciente. Una forma estática de construir una referencia sería incluir la dirección de Internet de la computadora donde reside, número de puerto del proceso que lo creó y momento de creación y un número de objeto local. Este último número es actualizado cada vez que el proceso crea un nuevo objeto.

El problema es la falta de flexibilidad de este esquema. ¿Qué pasa si los objetos son “reubicados”? ¿Qué pasa con la deseada transparencia de ubicación o de migración? Se puede utilizar un servicio de localización, que utiliza una base de datos que relaciona objetos remotos con sus direcciones actuales, o por lo menos, las últimas direcciones reportadas.

Se llama ‘acción’ a una cadena de invocaciones de métodos relacionados, cada uno de los cuales retornará en algún momento. Se dice que un objeto remoto está activo cuando está disponible para su invocación dentro de un proceso en ejecución.

Está en estado pasivo o inactivo, si no está activo en ese momento, pero que se puede llegar a activar. Un objeto pasivo contiene la implementación de sus métodos y su estado.

Cuando un objeto pasivo se “activa”, crea una nueva instancia de su clase y se inicializan las variables de instancia desde el estado que está almacenado. Pueden activarse bajo demanda como cuando son invocados por otros procesos. Los objetos invocados remotamente, cuando no están siendo utilizados pero tienen información, gastan recursos. Por lo tanto, en los servidores hay actividades que se arrancan bajo demanda para activar los procesos.



Por otra parte, serializar un objeto significa “aplanarlo”, es decir, obtener una forma lineal adecuada para ser almacenado en disco o para transmitirlo en un mensaje. La deserialización consiste en volver a llevar esa forma lineal a la forma original del objeto.

El problema de distribuir los procesos a lo largo de la red implica un análisis profundo de cómo se relacionan. Se debe pensar, por ejemplo, en los cambios necesarios y recurrentes en la lógica de negocio (*business logic*), imposible de realizar rápidamente si las componentes que la implementan están repartidas en varias máquinas pequeñas. Además, los cambios que se realicen en la interfase de usuario no deben incidir en las componentes que implementan la lógica de negocio. Una solución es separar el cliente y la instancia de la componente que se invoca en dos procesos diferentes, donde el proceso cliente trabaja en computadoras orientadas a la interacción con el usuario, y los procesos componentes en computadoras especializadas (y preparadas) para compartir y administrar la lógica de negocio.

Estando en diferentes computadoras, la comunicación entre el proceso cliente y el proceso componente se implementaría de la siguiente forma: en el proceso cliente, se agrega una representación del componente (*instance surrogate*). En el proceso componente, se crea una representación del cliente (*client surrogate*).

Veamos cómo funciona: el cliente hace un requerimiento a la componente, a través de su interfase. Ese requerimiento llega al *instance surrogate* que está en el mismo proceso cliente. Esta representación del componente es quien se comunica con el *client surrogate*, empaquetando el requerimiento y mandándolo a través de la red, a la máquina donde reside el proceso componente. Una vez allí lo recibe el *client surrogate* quien le hace el requerimiento a la instancia real del componente, quien contesta al *client surrogate*. La respuesta es empaquetada por el *client surrogate*, llegando a la máquina del proceso cliente, donde la recibe el *instance surrogate*, quien es quien se la manda al proceso cliente.



En conclusión, el proceso cliente se abstrae de la ubicación del componente, implementándose los que en sistemas distribuidos se llama transparencia de acceso, que es la abstracción por parte del solicitante sobre la ubicación del recurso, local o remoto.

Las componentes residirán en una computadora preparada para este fin, permitiendo una fácil administración (por ejemplo, en caso de ser necesaria la modificación) y seguridad. Esta arquitectura de distribución es la que usan los sistemas con componentes (CORBA, DCOM/COM/MTS, EJB de Sun).

Una ventaja de la tecnología de componentes es la separación entre lógica de ejecución y lógica de negocio.

Pero aún quedaría la duda de ¿cómo asegurar que la componente que responde es realmente la que se invocó? Y, además, ¿cómo asegurar la no-intercepción de la respuesta?

En cuanto al acceso a bases de datos, el modelo distribuido de componentes exige interactuar con múltiples bases, con procesos cooperantes. Las transacciones deben adaptarse a esta nueva forma de trabajo, acostumbrados como están a la relación de requerimiento proceso-base de datos. El sistema de transacciones tradicional trabaja adecuadamente en ambientes “cerrados”, con transacción de corta vida. En el nuevo modelo a plantear, la lógica de negocio distribuida es implementada con lenguajes como el BPEL4WS.

Con respecto a la *escalabilidad*, en el modelo distribuido de componentes, cada cliente es representado en el proceso cliente por el *surrogate* y la instancia de la componente. Esas instancias seguramente se conectarán con distintas bases generando varios mensajes, y además, procesarán gran cantidad de datos. Si cada PC es un cliente potencial, estaremos generando una actividad que no se puede sostener. Hay que lograr una escalabilidad que permita un aumento



importante en el número de clientes. Para Roger Sessions algunas soluciones las da TPM, *Transaction Processing Monitors*, de Microsoft²⁵.

Haciendo referencia a una de las aplicaciones de estos modelos, las aplicaciones *Three-Tier* tienen importantes ventajas:

- Separación entre la interfase de usuario y la lógica de negocio.

La interfase de usuario se ubica en máquinas a tal efecto, diferentes de las máquinas donde reside la lógica de negocio. Física y lógicamente están separadas.

Esta característica brinda flexibilidad tanto para el diseño de la interfase, como para la reusabilidad de la lógica de negocio. Por ejemplo, una componente que calcula el precio de un producto de acuerdo a diferentes pautas, puede ser accedida por páginas Web, PCs clientes de negocios mayoristas, servicios autorizados, etc.

- Rapidez en el desarrollo y actualización

Si la lógica de negocio ha sido implementada y probada exitosamente, podrán agregarse nuevas interfases rápidamente. También modificar las componentes, sin modificar la interfase de usuario. O construir componentes nuevas e incluirlas posteriormente, una vez que se hayan probado.

- Facilidad de administración

Otro aspecto importante en las estructuras distribuidas son los lenguajes de definición de interfases (IDL), los cuales fueron desarrollados para que los objetos en lenguajes diferentes puedan invocarse entre sí. Corba usa CORBA IDL, Sun propone XDR para su RPC, DCE usa su IDL para RPC, Microsoft usa DCOM IDL.

²⁵ Consultado el 13/06/08, disponible en:
<http://www.sdtimes.com/content/article.aspx?ArticleID=25420>



Un punto interesante es si estos IDLs exponen las interfases de manera tal que sean comprendidos por cualquier objeto invocante.

En estos lenguajes, se denomina “representación externa de datos” al estándar que acuerdan dos o más computadoras para intercambiar datos. De esta forma se superan problemas como el ordenamiento de enteros, diferentes formas de representar flotantes o la cantidad de caracteres que toman los distintos formatos.

En RMI o RPC se debe llevar cualquier estructura que desea pasarse como argumento o devuelto como resultado a una forma plana, que viaje y sea interpretada correctamente en el destino (serialización de objetos).

Por lo anteriormente expuesto podemos concluir que el modelo distribuido sobre la Red es hoy una necesidad. Pero para implementar ese modelo distribuido es imprescindible acordar cómo trabajar con distinto hardware, distintas plataformas; integrando servicios, pero sin perder la transparencia para el usuario.

Bibliografía del tema 3

Pressman, Roger S *Ingeniería de software. Un enfoque práctico*, 5ª ed., México, Mc Graw Hill. 2002.

Senn, James A. *Análisis y diseño de sistemas de información*, 2ª ed., México, 1992.

Scott, George M. y Daniel Cohen *Sistemas de información*, México, Mc Graw Hill. 1996.

Referencias de Internet:

http://trevinca.ei.uvigo.es/~gbernard/PPI/temas/PPI-t4_1.ppt

<http://siona.udea.edu.co/~aoviedo/Arquitectura%20de%20Software/RTF.htm>

http://antares.itmorelia.edu.mx/~jcolivar/cursos/is208a/is2_u2.ppt



<http://www.cs.cinvestav.mx/CursoVis/prinvisual.html>

http://www.sedici.unlp.edu.ar/search/downloadp.php?id_document=ARG-UNLP-TPG-0000000068&id_parte=1851

http://www.sedici.unlp.edu.ar/search/downloadp.php?id_document=ARG-UNLP-TPG-0000000068&id_parte=1852

http://antares.itmorelia.edu.mx/~jcolivar/cursos/sd107v/sd1_u1.pdf

<http://www.sdtimes.com/content/article.aspx?ArticleID=25420>

Actividades de aprendizaje.

A.3.1 A partir de la revisión general de tema 3, elabora un mapa conceptual con el fin de tener un panorama general del mismo.

A.3.2 Con el fin de tener un glosario de términos amplio, elabora una lista de palabras relacionadas con este tema y su definición.

A.3.3 Define el perfil profesional de la persona o personas que se desempeñan en la rama de la arquitectura del software.

Cuestionario de autoevaluación

1. ¿En qué radica la importancia de la arquitectura de sistemas de software?
2. Las arquitecturas software no responden únicamente a requisitos estructurales, ¿con qué más se relacionan?
3. ¿Qué es el aseguramiento de la calidad?
4. ¿Qué es la Certificación?



5. ¿Para qué sirve la documentación de programas?
6. Las revisiones técnicas formales (RTF) son parte de las actividades que garantizan la calidad del software pero, ¿qué son en realidad las RTF?
7. Se ha encontrado que llevar a cabo las RTF objetivamente tiene ciertas ventajas ¿como cuáles?
8. ¿Bajo qué procedimiento se realiza una RTF?:
9. ¿Qué es lo que describe un patrón de arquitectura de software?
10. ¿Qué es lo que caracteriza a los sistemas distribuidos?
11. En general, ¿cómo se podría ver a un sistema distribuido?

Examen de autoevaluación

1. Los analistas usan cuatro niveles de aseguramiento de la calidad ¿Cuáles son?:
 - a) Prueba, verificación, validación y certificación.
 - b) Normatividad, certificación, análisis, diseño
 - c) Análisis, prueba, diseño, validación.
2. ¿Qué es una interfaz?
 - a) La comunicación del sistema integral de los programas
 - b) Es la frontera entre el usuario y la aplicación del sistema de cómputo
 - c) Es una rama del Hardware
3. Aunque los aspectos que determinan el diseño de interfases gráficas de usuario son múltiples, los podemos agrupar en tres grandes grupos, ¿cuáles son?
 - a) Interacción general, comunicación total, determinación de la información.



- b) Visualización de información, entrada de datos, comunicación general.
 - c) Entrada de datos, visualización de la información, interacción general.
4. Las fases del proceso del desarrollo de interfaces de usuario son:
- a) Análisis de usuarios, tareas y entornos. Implementación y validación de la interfaz
 - b) Análisis de usuarios, tareas y entornos Diseño de la interfaz. Implementación de la interfaz. Validación de la interfaz
 - c) Análisis de usuarios, tareas y entornos Diseño de la interfaz. Implementación de la interfaz. Validación del sistema
5. Dentro de las directrices para la calidad del diseño se menciona que éste deberá presentar una estructura arquitectónica que sea creada con patrones de diseño reconocibles que son básicos para lograr un buen diseño, estos principios son:
- a) Abstracción, refinación, ocultamiento de información y modulación
 - b) Visualización de información, entrada de datos, comunicación general.
 - c) Visualización de información, entrada de datos, Abstracción, refinación.
6. ¿Qué pasos se recomiendan seguir para el desarrollo del diseño arquitectónico?
- a) Estructuración del sistema, modelado de control, análisis
 - b) Normatividad, certificación, análisis, diseño
 - c) Estructuración del sistema, modelado de control, descomposición modular, desarrollo.



7. Es una ventaja de la tecnología de componentes:
 - a) La separación entre lógica de ejecución y lógica de negocio.
 - b) La utilización de la unidad aritmética lógica
 - c) La compactación de la información.

8. ¿A qué se refiere el término *capacidad de crecimiento incremental*?
 - a) A la reducción del tiempo del recorrido de los datos en el BUS
 - b) A la posibilidad de aumentar el número de dispositivos de entrada y salida.
 - c) A que se puede añadir procesadores al sistema incrementando su potencia en forma gradual según sus necesidades.

9. Las soluciones que resuelven problemas de forma clara y económica traen consigo beneficios como:
 - a) Toma de decisiones acertadas, naturalidad, menos esfuerzo.
 - b) Accesibilidad, reconocimiento, rapidez de percepción y uso continuo.
 - c) Gran cantidad y calidad en la información.

10. Estudiar aplicaciones del mismo tipo tiene muchas ventajas ya que nos ayuda a entender:
 - a) Las necesidades del cliente y las soluciones.
 - b) Las expectativas del programador
 - c) El diseño de los sistemas.



TEMA 4. FUNDAMENTOS DE REDES DE COMPUTADORAS

Objetivo particular

Brindar al alumno las bases teóricas que le permitan describir la arquitectura de redes y analizar los beneficios derivados de conectar computadoras para formar una red funcional y segura.

Temario detallado

- 4.1. Introducción a las redes de datos
- 4.2. Topología de redes
- 4.3. Protocolos de comunicación
- 4.4. Tipos de redes
- 4.5. Seguridad en red

Introducción

Este tema lo dedicaremos a estudiar con detalle el envío de datos entre sistemas de cómputo, las redes que se necesitan y los llamados “protocolos” que se requieren para asegurar que los sistemas se puedan comunicar apropiadamente.

4.1. Introducción a las redes de datos

Conforme las PC comenzaron a difundirse en los negocios y aparecieron los complejos multiusuarios de software, conectar entre sí las PC se convirtió en una meta para la generalidad de las organizaciones debido a la necesidad de compartir información o datos mediante algún método más seguro y adecuado que los disquetes blandos. Posteriormente, esta demanda se amplió no sólo entre departamentos de la misma organización sino fuera de ella, con más rapidez y en mayor volumen cada vez. Surgieron así las redes de cómputo de procesamiento distribuido avanzadas. La comunicación de datos, es decir, la transmisión



electrónica de información entre computadoras, se convirtió en punto esencial para la industria de las computadoras.

Aunque el término “red” tiene muchas definiciones, en el campo de la informática debemos definirla como un modo de conectar computadoras para que se puedan comunicar, intercambiar información y compartir recursos.²⁶

El rápido crecimiento de la red mundial de computadoras llamada Internet (red mundial a la cabeza de la supercarretera de la información) hizo que la difusión de comunicación de datos se apresurara aún más.

Técnicamente, Internet es una red que conecta a miles de otras redes y millones de computadoras; sin embargo, representa mucho más. Es el medio actual para tener acceso a/y compartir conocimientos. Es también el cimiento para la comunidad global de usuarios del mañana.²⁷

Las computadoras se comunican de dos formas: por medio de módems y de redes. Los módems habilitan a las computadoras para usar líneas telefónicas (fijas), conexiones celulares (telefonía móvil) o incluso vínculos satelitales para conectarse a otras computadoras y de esta forma intercambiar información, más adelante explicaremos cómo es que lo hacen, pero cabe mencionar que para que exista tal comunicación, es imprescindible que las computadoras involucradas dispongan de los mismos elementos (módem, conexión a línea telefónica o a la red). Las redes conectan computadoras directamente (conexión directa) a velocidades altas, ya sea por medio de cableado especial -si la distancia es corta- o mediante alguna forma de transmisión inalámbrica si la distancia entre sí es grande (normalmente más de 20 metros). El proceso de conectar una computadora a una red, ya sea utilizando un módem o por medio de una conexión directa se conoce como “estar en línea” (*on-line*).

Una red de cómputo ligeramente más compleja puede conectar varias terminales de cómputo de edificios lejanos con la computadora principal (anfitriona) de un centro especializado de datos. Otra red puede ser la que se emplea en una oficina

²⁶ P. Norton, op.cit, p. 248.

²⁷ Timothy J. O'Leary y Linda I. O'Leary, *Computación Básica*. México, Mc Graw Hill, 1997 p. 163.



para interconectar varios dispositivos de cómputo de escritorio, impresora láser, dispositivos de almacenamiento de datos, etc. Las redes más complejas de cómputo pueden interconectar -por ejemplo- las diferentes computadoras grandes de los principales centros financieros del mundo y suministrar a los comerciantes información de mercado de último momento.

Los principios básicos de transmisión de datos (ancho de banda, transmisión en serie y en paralelo, dirección del flujo y los modos de transmisión y los protocolos) se aplican de igual manera a los datos que se comunican sobre las redes de computadoras. De modo que las redes de conmutación de circuitos o las simples líneas de punto a punto también se pueden utilizar para la comunicación de datos. Sin embargo, la conmutación de datos impone mayores demandas en su red básica que el servicio de señal analógica o de voz, necesitándose medidas adicionales para codificar los datos como preparación para la transmisión y para controlar el flujo de datos durante la transmisión. Para asegurar que la información que pasa entre computadoras es correcta, completa y se entiende apropiadamente, se emplea “protocolos” especiales en la comunicación de datos, más adelante, en el punto 4.3, retomaremos el tema de los protocolos con detalle.

Por otro lado, se ha encontrado que la transmisión digital es la ideal para el envío de datos binarios. La transmisión digital se está convirtiendo en la actualidad en la estructura fundamental de las redes privadas y públicas, sin embargo, a pesar de la creciente disponibilidad y la conveniencia ideal de la transmisión digital para la comunicación de datos, desafortunadamente no siempre es aprovechable. En estos casos la información de computadora con estructura digital se debe convertir a la forma apropiada para transmitirla a través de una red analógica. Esta conversión se lleva a cabo con el equipo que se conoce como “modulador/demodulador” (módem). El módem transmite los datos mediante la imposición de datos binarios (digitales) sobre una señal portadora de audiofrecuencia. El proceso es muy similar al que se emplea en el MDF de canales de voz, pero el módem debe convertir y reconvertir las señales digitales que reciben de un equipo terminal de datos en señal analógica apropiada para la



transmisión de datos. Al proceso que realiza el módem para convertir señales digitales a analógicas se le denomina “modulación” y la acción de convertir señales analógicas a digitales se le llama “demodulación”. Hay varios tipos de módems, desde el externo que funciona fuera de la cabina y se conecta a ella por cable, hasta el módem inalámbrico que no se conecta a la línea telefónica sino que recibe datos por vía aérea.

Cada día existe más demanda de servicios de telecomunicación entre computadoras, y entre éstas y terminales conectados en lugares alejados de ellas, lo cual abre más el abanico de posibilidades de la conjunción entre las comunicaciones y la computación o informática, conjunción a la que se da el nombre de telemática.²⁸

Finalmente, listamos algunos de los usos más comunes de una red:

Las redes permiten a los usuarios el acceso simultáneo a programas comunes e información.

Las redes también permiten a los usuarios compartir dispositivos periféricos, como impresoras y discos duros, y por lo tanto prolongan la vida útil de muchas máquinas.

Las redes, por lo general incluyen la capacidad para enviar correo electrónico, muchos sistemas de correo electrónico permiten a los usuarios añadir archivos a sus mensajes.

Algunas redes también ayudan a la comunicación al proveer herramientas para el uso de teleconferencia y la videoconferencia.

Conectar computadoras para formar una red hace más fácil respaldar la información en todos los discos duros puestos en la red.

²⁸ Consultado el 08/05/08, desde http://es.wikipedia.org/wiki/Usuario:Javier_betancort



En el siguiente punto analizaremos la arquitectura de red, la cuál describe la forma en que está configurada una red de cómputo y las estrategias que emplea.

4.2. Topología de redes

Una red puede ordenarse o configurarse de varias maneras siendo la forma elegida un factor fundamental que va a determinar el rendimiento y la funcionalidad de la red. A esta disposición se le llama “topología de la red”. La topología idónea para una red concreta va a depender de diferentes factores, como el número de máquinas a interconectar, el tipo de acceso al medio físico que deseemos, etc. Además, es importante considerar los siguientes aspectos referentes a las topologías en general: la topología física, que es la disposición real de las máquinas, dispositivos de red y cableado (los medios) en la red; la topología lógica, que es la forma en que las máquinas se comunican a través del medio físico. Los dos tipos más comunes de topologías lógicas son *broadcast* (Ethernet) y transmisión de *tokens* (*Token Ring*) y por último la topología matemática, que hace referencia a mapas de nodos y enlaces, a menudo formando patrones.

La topología de *broadcast* significa que cada *host* envía sus datos hacia todos los demás *hosts* del medio de red. Las estaciones no siguen ningún orden para utilizar la red, sino que cada máquina accede a la red para transmitir datos en el momento en que lo necesita. Esta es la forma en que funciona Ethernet. En cambio, la transmisión de tokens controla el acceso a la red al transmitir un token eléctrico de forma secuencial a cada *host*. Cuando un *host* recibe el token significa que puede enviar datos a través de la red. Si el *host* no tiene ningún dato para enviar, transmite el token hacia el siguiente *host* y el proceso se vuelve a repetir.²⁹

Las cuatro principales topologías de red son *en estrella*, *de bus*, *en anillo* y *jerárquica*. Aunque en la actualidad, con la influencia de nuevas tecnologías se

²⁹ Consultado el día/mes/años, desde <http://mx.geocities.com/alfonsoaraujocardenas/topologias.html>



advierten nuevas topologías basadas -por ejemplo- en medios inalámbricos (topología *celular*).

En una red *en estrella* varias computadoras o dispositivos periféricos pequeños se enlazan con una unidad central. Esta unidad central puede ser una computadora anfitriona o un servidor de archivos. Todas las comunicaciones pasan por esta unidad central. El control se ejerce mediante “sondeo”. Esto es, a cada dispositivo conectado en red se le pregunta (“sondea”) si tiene un mensaje que enviar, tras de lo cual se les permite enviar sus mensajes por turnos.

Una de las ventajas de esta modalidad de red es que puede servir para ofrecer un sistema de tiempo compartido. Es decir, varios usuarios pueden compartir recursos (“tiempo”) con una computadora central. La red en estrella es comúnmente utilizada para enlazar a varias microcomputadoras con una macrocomputadora que dé acceso a la base de datos de una organización.

Algunas ventajas de utilizar una red en estrella es que esta topología reduce la posibilidad de fallo de red conectando todos los nodos a un nodo central ya que este concentrador central reenvía todas las transmisiones recibidas de cualquier nodo periférico a todos los nodos periféricos de la red, algunas veces incluso al nodo que lo envió. Todos los nodos periféricos se pueden comunicar con los demás transmitiendo o recibiendo del nodo central solamente. De esta manera, un fallo en la línea de conexión de cualquier nodo con el nodo central provocaría el aislamiento de ese nodo únicamente y el resto del sistema permanecería intacto.

La desventaja radica en la carga que recae sobre el nodo central. Pues la cantidad de tráfico que deberá soportar es grande y en aumento conforme se agreguen más nodos periféricos, lo que la hace poco recomendable para redes de gran tamaño. Además, un fallo en el nodo central puede dejar inoperante a toda la red. Esto último conlleva también una mayor vulnerabilidad de la red, en su conjunto, ante ataques.



En una *red de bus*, cada dispositivo en red se encarga de su propio control de comunicaciones. En este caso no existe una computadora anfitriona. Todas las comunicaciones viajan por una línea común de conexión (bus). A esta topología algunos autores la llaman “red de turno en línea” por la forma de operar que la distingue: una red de bus controla la transmisión de datos sobre la trayectoria de transmisión mediante el empleo de un solo turno. Solamente la terminal con el “turno” puede transmitir paquetes sobre la línea (bus). El turno se puede poner disponible para cualquier terminal que desee transmitir datos. Cuando una terminal tiene el turno, envía cualquier estructura de datos que tenga lista y después pasa el turno a la siguiente terminal. Para verificar que su sucesor ha recibido el turno correctamente, la terminal se asegura que el sucesor esté transmitiendo datos. Si no es así, se supone que el sucesor está en una parte defectuosa de la red y, para evitar que la red se “trabe”, la terminal original crea otro sucesor generando un nuevo turno. Las fallas de transmisión en el bus pueden, por lo tanto, ser evadidas en algún grado. Sin embargo, aquellas partes de la red que están aisladas del turno, permanecen fuera.

Las redes de bus normalmente no se usan en ambientes de oficina, son más comunes en instalaciones de fábricas, a menudo operando como redes de banda ancha (gran ancho de banda) para el montaje y control de máquinas robot complejas.

En una *red en anillo*, cada dispositivo se conecta a otros dos. Formándose así un anillo. En esta topología tampoco hay una computadora o servidor de archivos central. Los mensajes recorren el anillo hasta llegar a su destino.

Una red en anillo es similar en operación al turno en línea, empleando el turno para pasar “el derecho a transmitir datos” sobre cada terminal en el anillo. La secuencia de paso del turno es diferente: el turno mismo se emplea para portar el paquete de datos. La terminal que transmite pone la bandera del turno, colocando la dirección de destino en el encabezado para indicar que el turno está completo. Después, el turno se pasa alrededor del anillo desde una terminal a la siguiente.



Cada terminal verifica si los datos van dirigidos a ella y si no es así, los pasa; tarde o temprano llegan a la terminal de destino en donde los datos se leen. La recepción de datos se confirma al transmisor cambiando el valor de un *bit* en la bandera de turno. Cuando el turno regresa a la terminal de transmisión, la terminal está obligada a vaciar el turno y pasarlo a la siguiente terminal del anillo.

Estas redes son comunes en los ámbitos de oficinas, enlazando computadoras personales para el propósito de transferencia de datos de archivos, mensajería electrónica, interacción de computadora principal o compartición de archivos. Aunque se dice que funcionan mejor que las Ethernet (tecnología de red muy comúnmente usada) a capacidad casi completa o durante sobre carga resultan ser más difíciles y costosas de instalar, especialmente cuando sólo se incluye un pequeño número de agregados.

Las redes en anillo son útiles en organizaciones descentralizadas, porque hacen posible la formación de un sistema de procesamiento de datos distribuido. Es decir, las computadoras pueden llevar a cabo tareas de procesamiento en su propia ubicación, a pesar de que se hallen dispersas. Sin embargo, también pueden compartir entre sí programas, datos y otros recursos.

La *red jerárquica* se compone de varias computadoras enlazadas a una computadora anfitriona central, tal como en una de estrella con la diferencia de que aquellas otras computadoras son a su vez anfitrionas de otras más pequeñas o de dispositivos periféricos. Como su nombre lo indica, en el punto más alto de la jerarquía se ubica a la anfitriona que podría ser una macrocomputadora. Las computadoras jerárquicamente inferiores a ella podrían ser minicomputadoras y las inferiores a éstas, microcomputadoras. Este tipo de red permite que varias computadoras compartan bases de datos, capacidad de procesamiento y diferentes dispositivos de salida.

A este tipo de red a veces se le llama “de árbol” y asemeja las jerarquías como las ramas y las hojas, también se dice que son como un conjunto de redes estrellas



enlazadas. Considerando esto, podemos suponer que, como en las redes en estrella convencionales los nodos individuales pueden quedar aislados de la red por un fallo puntual en la ruta de conexión del nodo. Si falla un enlace que conecta con un nodo hoja, ese nodo hoja queda aislado; si falla un enlace con un nodo que no sea hoja, la sección entera queda aislada del resto.

Para resolver el conflicto creado por la cantidad de tráfico de red que se necesita para retransmitir todo a todos los nodos, se desarrollaron nodos centrales más avanzados que permiten mantener un listado de las identidades de los diferentes sistemas conectados a la red. Se dice que estos *switches* de red “aprenderían” cómo es la estructura de la red transmitiendo paquetes de datos a todos los nodos y luego observando de dónde vienen los paquetes respuesta.

Existen algunas otras topologías más complejas como la de *en malla*, en la cuál hay al menos dos nodos con dos o más caminos entre ellos. El número de caminos arbitrarios en las redes en malla las hace más difíciles de diseñar e implementar, pero su naturaleza descentralizada las hace muy útiles.

De esa manera se puede llegar a crear una *red totalmente conectada o completa*, en la que hay un enlace directo entre cada pareja de nodos. En una red totalmente conexa con n nodos, hay $\frac{n \times (n - 1)}{2}$ enlaces directos. Aunque esta red resultaría cara de instalar, es muy fiable gracias a los múltiples caminos por los que los datos pueden viajar. Se ve principalmente en aplicaciones militares.

No importa cuál sea su estructura, todas las redes dependen de los medios para enlazar sus nodos y/o servidores. Los medios más comunes para la comunicación de datos son el cable de par trenzado, el cable coaxial, el cable de fibra óptica y los enlaces inalámbricos. Hoy en día la comunicación inalámbrica compite con el cable de par trenzado, el coaxial y la fibra óptica. La ventaja de la comunicación inalámbrica es la flexibilidad que ofrece en términos del acomodo de la red. Para transmitir la información, la comunicación inalámbrica depende de señales de radio o de señales infrarrojas. Es importante recordar que las frecuencias de radio



forman parte del espectro electromagnético, que incluye los rayos X, la luz ultra violeta, los rayos infrarrojos, las microondas y las ondas de mayor longitud usadas por las estaciones de radio comerciales.

En cuanto al método de coordinación del compartimiento de información y recursos, las estrategias de red más comunes son el sistema terminal, de igual a igual y de cliente/servidor.

En un *sistema de red terminal* la capacidad de procesamiento se centraliza en una computadora de gran tamaño, por lo general una macrocomputadora. Los nodos conectados a esta computadora anfitriona son terminales con escasa o nula capacidad de procesamiento. Las redes en estrella y jerárquica son configuraciones que habitualmente utilizan este sistema con UNIX como sistema operativo.

En un *sistema de red de igual a igual* los nodos puedan actuar lo mismo como servidores que como clientes. Por ejemplo, una microcomputadora puede obtener archivos localizados en otra microcomputadora y al mismo tiempo proporcionar archivos a otras microcomputadoras. Una de las configuraciones más usuales para los sistemas de igual a igual es la red de bus.

Los *sistemas de red de cliente/servidor* emplean una potente computadora para coordinar y suministrar servicios a todos los demás nodos en red. El servidor da acceso a recursos centralizados como bases de datos, software de aplicación y hardware. Se puede decir que este sistema se basa en la especialización ya que los nodos servidores coordinan y prestan servicios especializados, mientras que los nodos clientes solicitan estos servicios. Son ideales para manejar redes de gran tamaño.



4.3. Protocolos de comunicación

Para que la transmisión de datos sea exitosa, emisor y receptor deben seguir una serie de reglas de comunicación para el intercambio de información. Estas reglas para el intercambio de datos entre computadoras se llaman “protocolos” de la línea.

Cuando en una red son conectados diferentes tipos de microcomputadoras, el protocolo puede volverse sumamente complejo y para que las conexiones funcionen, los protocolos de red deben obedecer a ciertos estándares.

Originalmente, los protocolos fueron relativamente sencillos como aquellos sobre los que se apoyaron las redes simples computadora-terminal y que estuvieron contenidos en otros programas de aplicación a computadoras, de tal manera que además de su función principal de procesamiento, la computadora estaría controlando la transmisión de línea entre ella y las terminales asociadas y otro equipo periférico. IBM puso en circulación el primer conjunto de estándares comerciales al que le llamó *Systems Network Architecture* (arquitectura de redes de sistemas, SNA) pero éstos estándares sólo operaban con el equipo propio de IBM. Conforme las redes se sofisticaron, muchos accesorios de computadora (equipos de distintos fabricantes) resultaban incompatibles. Para frenar esta situación, se desarrolló el concepto de *protocolos de capas* con el objeto de separar todas las funciones de telecomunicaciones para formar un conjunto de subfunciones por capas. En poco tiempo, la *International Standards Organization* (Organización de Normas Internacionales. ISO) definió una serie de protocolos de comunicaciones llamada *Open Systems Interconnection* (interconexión de sistemas abiertos, OSI) cuyo propósito es identificar las funciones provistas por cualquier red retomando el concepto de trabajar en capas con la idea de establecer estándares mundiales de diseño para todos los protocolos de datos de telecomunicaciones, para que todos los equipos que se produzcan sean compatibles.



Bajo este esquema de protocolo, cada capa desarrollaría una tarea distinta y autosuficiente pero sería dependiente de las subcapas. Así las tareas complejas comprenderían varias capas, mientras que las sencillas requerirían sólo algunas. La función simple de cada capa implicaría realización simple de circuitería y logística y sería independiente de las funciones de otras capas, de tal manera que se podría cambiar ya sea las funciones o la realización de una capa funcional con mínimo impacto sobre la logística y la circuitería de las otras capas.

Actualmente, la mayoría de los protocolos de transferencia de datos de uso común emplean un “arreglo” de protocolos de capas, es importante estudiar este arreglo para obtener una buena idea de todo el rango de funciones que se necesitan para la transferencia exitosa de datos. Para esto tenemos que considerar las funciones de cada capa de protocolo establecidas en el modelo OSI (en español, ISA). Este modelo no es en sí mismo un conjunto de protocolos, sino que define cuidadosamente la división de las capas funcionales con la cual se espera que se conformen todos los protocolos modernos.

El principio del modelo de interconexión de sistemas abiertos es que en tanto que las capas interactúen de una manera “aparejada” y toda vez que la interfaz entre la función de una capa y su capa inmediata superior e inferior no se afecten, entonces no es importante la forma como se lleva a cabo la función de esa capa individual. Este modelo subdivide la comunicación de datos en siete capas “aparejadas” que, en orden ascendente, son las siguientes:

Capa de aplicación (capa 7). Suministra servicios de comunicación para satisfacer todos los tipos de transferencia de datos entre computadoras “co-operantes”.

Capa de presentación (capa 6). Su tarea es negociar una técnica mutuamente acorde para la codificación y puntuación de datos (sintaxis de datos) y encargarse de cualquier conversación necesaria entre diferentes formatos de código o arreglo de datos, para que la capa de aplicación reciba el tipo que reconoce.



Capa de sesión (capa 5). El protocolo de sesión incluye comandos tales como arranque, interrumpir, reanudar y terminación, para gestionar una sesión de comunicación (conversación) entre dispositivos de forma apropiada y en orden.

Capa de transporte (capa 4). El servicio de transporte se encarga del relevo de datos de extremo a extremo que se necesita en la sesión de comunicación. Establece la conexión de red que más se adapte a los requerimientos de sesión en términos de la calidad de servicio, tamaño de la unidad de datos, control de flujo y necesidades de correo de datos. También debe suministrar las direcciones de red que necesita la capa de red para la entrega correcta del mensaje.

Capa de red (capa 3). Establece la conexión extremo a extremo a través de una red real y determina qué permutación de enlaces individuales se emplea (funciones de enrutamiento).

Capa de enlace de datos (capa 2). La capa de enlace de datos opera solamente dentro de los enlaces individuales de una conexión, manejando la transmisión de datos, para que los bits individuales se envíen sobre aquellos enlaces sin error.

Capa física (capa 1). El protocolo de capa física envía los datos sobre el medio; es una combinación de material y logística que convierte los bits de datos (que necesita la capa de enlace de datos) en pulsos eléctricos, tonos de módem. Señales ópticas o cualquier otra entidad que va a transmitir los datos. Se asegura que los datos se envíen sobre el enlace y se presenten en ambos extremos de la capa de enlace de datos en la forma estándar.

En cuanto al formato que deben tener los datos para ser manejados por los protocolos, la clave está en utilizar encabezados. Cada capa de protocolo agrega un encabezado que contiene información para su propio uso, de tal modo que todo el mensaje es más largo que el que se recibe desde la capa más alta (capa 7). Los encabezados portan la información que el protocolo necesita para hacer su trabajo, son eliminados del mensaje al ser recibidos y después se pasan a la siguiente capa superior del protocolo (normalmente en forma síncrona).



En realidad, la mayoría de las capas de protocolo ISA existen solamente en software y no pueden ser identificadas como elementos físicos. Sin embargo, no todas las diferentes capas de protocolo necesitan ser instrumentadas dentro del mismo programa de computadora o llevadas a cabo por la misma parte del equipo.

Otro aspecto importante del modelo ISA es que proporciona grandes posibilidades, permitiendo el desarrollo de redes muy sofisticadas. Se puede dar el caso de que no se necesiten las funciones muy complejas, en este caso el modelo permite el empleo de los *protocolos nulos*. Por ejemplo, en una red que emplea dispositivos terminales similares, las posibilidades de conversión de sintaxis de la capa de presentación son innecesarias. De esta manera se evita instrumentar funciones que pudieran elevar el costo y volumen de la administración.

Hoy en día, la red que conecta a miles de redes y millones de usuarios alrededor del mundo es Internet, la cual se puede ver como una comunidad cooperativa enorme sin una propiedad central. En sí misma Internet es el conducto para transportar datos entre computadoras. Cualquiera que cuente con acceso a Internet puede intercambiar texto, archivos de datos y programas con cualquier otro usuario. Pero esto no sería posible si cada computadora conectada a Internet no utilizara el mismo conjunto de reglas y procedimientos (protocolos) para controlar la sincronización y el formato de los datos. Al conjunto de comandos y las especificaciones de sincronización utilizados por Internet se le llama *Protocolo de Control de Transmisión / protocolo Internet*, universalmente conocido como *TCP/IP*. Este protocolo hace posible enlazar cualquier tipo de computadoras, sin importar el sistema operativo que se use ni el fabricante, y el sistema de IP permite a las redes enviar correo electrónico, transferencia de archivos y tener una interacción con otras computadoras sin importar dónde estén localizadas, siempre y cuando tengan acceso a Internet.

Los protocolos TCP/IP incluyen las especificaciones que identifican a las computadoras individuales e intercambian datos entre computadoras. También incluyen reglas para varias categorías de programas de aplicación, es así que los



programas que se ejecutan en diferentes tipos de computadoras pueden comunicarse unos con otros.

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 al conectarse, sin tomar en cuenta el tamaño de ellas, ya sean locales o de cobertura amplia. Igualmente, aunque el software TCP/IP parezca distinto en diferentes tipos de computadoras, para la red siempre presenta la misma apariencia. Sin embargo; define que todas las redes que intercambiarán información deben estar conectadas a una misma computadora o equipo de procesamiento (dotados con dispositivos de comunicación); denominadas enrutadores o puentes. Por lo tanto, la actividad de Internet puede definirse como la de computadoras que se comunican con otras computadoras mediante el uso de TCP/IP.

Además, para que en una red dos computadoras puedan comunicarse entre sí ambas deben estar identificadas con precisión porque la computadora que origina una transacción debe identificar con una dirección única el destino al que se dirige. Por lo que en Internet cada computadora tiene una dirección numérica que consta de cuatro partes y que se conoce como dirección de protocolo Internet o dirección IP, ésta identifica tanto a la red a la que pertenece una computadora como a ella misma dentro de dicha red porque contiene información de enrutamiento.

Los servicios más importantes de TCP/IP son:

Transferencia de Archivos FTP (File Transfer Protocol). Este protocolo permite a los usuarios obtener o enviar archivos a otras computadoras.

Acceso Remoto (Telnet): El acceso remoto (*Telnet*) es un protocolo que permite el acceso directo de un usuario a otra computadora en la red. Para establecer un *Telnet*, se debe establecer la dirección o nombre de la computadora a la cual se desea conectar. Cuando se accede por este tipo de protocolos, generalmente la computadora remota pregunta por un nombre de usuario (*user name, login, etc.*) y



por una clave (*password*). Cuando ya se desea terminar con la sesión, basta con terminar este protocolo, para salir generalmente con los comandos: *logout*, *logoff*, *exit*, etc.

Correo en las computadoras (e-mail): Este protocolo permite enviar o recibir mensajes a diferentes usuarios en otras computadoras.

Sistemas de archivo en red (NFS): Esto permite a un sistema acceder archivos en otra computadora de una manera más apropiada que mediante un FTP. El NFS da la impresión de que los discos duros de la computadora remota están directamente conectados a la computadora local. De esta manera, se crea un disco virtual en el sistema local. Esto aparte de los beneficios económicos, también permite trabajar a los usuarios en varias computadoras y compartir archivos comunes.

Impresión Remota: Esto permite acceder impresoras conectadas en la red, para lo cual se crean colas de impresión y el uso de dichas impresoras se puede restringir, ya sea mediante alguna contraseña o a ciertos usuarios. El beneficio es el poder compartir estos recursos.

Ejecución remota: Permite correr algún programa en particular en alguna computadora. Es útil cuando se tiene un trabajo grande que no es posible correr en un sistema pequeño, siendo necesario ejecutarlo en uno grande.

La mayoría de las computadoras en Internet (excepto aquellas que se utilizan exclusivamente para conmutación y enrutamiento interno) también tiene una dirección llamada "dirección de Sistema de Nombres de Dominio (DNS), la cual emplea palabras en vez de números para facilitar el manejo de estas direcciones a los humanos. Las direcciones DNS constan de dos partes: un nombre individual, seguido por un "dominio" el cual generalmente identifica el tipo de institución que utiliza la dirección; por ejemplo, *.com* se refiere a negocios comerciales. A veces este dominio puede estar dividido en "subdominios" para especificar más la



dirección, inclusive un domino también puede identificar el país en el cual se localiza el sistema; por ejemplo, *.mx* se refiere a México.

Cuando una computadora está al servicio de muchos usuarios, cada uno de ellos también debe identificarse adquiriendo una cuenta única dentro del dominio, el formato estándar es el nombre de usuario, separado de la dirección DNS por el símbolo @ (arroba) que significa “en”; por ejemplo, *mireyale@yahoo.com.mx*

Desde la creación de la World Wide Web (la Web o WWW) en 1989 y de los examinadores Web que se desarrollaron a partir del mismo, se ha abierto un mundo de posibilidades para que la gente realice actividades a través de una PC desde su hogar u oficina gracias a Internet.

La estructura interna del *World Wide Web* está construida sobre un conjunto de reglas llamado protocolo de transferencia de hipertexto (*Hypertext Transfer Protocol:HTTP*) y un lenguaje de descripción de página llamado lenguaje para marcación de hipertexto (*Hypertext Markup language: HTML*). HTTP utiliza direcciones Internet en un formato especial llamado localizador uniforme de recursos (*Uniform Resouse Locator: URL*). Los URL son semejantes a lo siguiente: *tipo://dirección/ruta*, “tipo” especifica el tipo del servidor en el cual se localiza el archivo, “dirección” es la dirección del servidor y “ruta” es la ubicación dentro de la estructura de archivos del servidor. Así que el URL para la Universidad de Illinois es: *http://www.uiuc.edu*.³⁰

4.4. Tipos de Redes

A continuación revisaremos los diferentes tipos de redes reales que se emplean para el envío de datos, comenzando con la tecnología simple de punto a punto hasta las WAN.

Las *redes de punto a punto* que implican la simple interconexión de dos equipos son relativamente simples de establecer, pueden emplear ya sea líneas digitales, líneas analógicas o módems. Siempre que los protocolos en ambos extremos del enlace concuerden, los equipos terminales de datos (ETD) dialogan fácilmente. En su forma más sencilla, una red de punto a punto se puede trabajar en el modo

³⁰ P. Norton, op.cit., p. 302.



asíncrono, carácter por carácter. Éste es un método común de conectar terminales remotas a una computadora. El empleo de esta sencilla técnica reduce considerablemente la complejidad y el costo del material y logística que se necesitan en las terminales (remotas) de computadora. Este tipo de conexión no concuerda con el ideal ISA, ya que sólo las terminales de computadora de este tipo de unos cuantos fabricantes se pueden usar con las computadoras de otros fabricantes; pero una de las desventajas del modelo ISA es el volumen de equipo y logística que se necesita en cada dispositivo de transmisión y recepción.

En una *red de datos con conmutación de circuitos* se establece una conexión sobre demanda entre dos clientes de una red telefónica, dicha conexión se establece durante el tiempo de la llamada y al finalizarla, se libera la conexión. El beneficio de estas redes radica en el gran rango de destinos disponible sobre diferentes llamadas y esto les da una utilidad muy importante en el envío de datos. Un ejemplo es la red de telex. Una vez que se establece una conexión telex, se transmiten caracteres entre las dos terminales *telex* a la velocidad de línea de 50 *baud*, empleando el alfabeto IA2. Las conexiones en una red de datos con conmutación de circuitos se establecen de la misma manera que las conexiones telefónicas. Una vez establecida la conexión, los protocolos de datos en las dos terminales extremas controlan el flujo de datos a través de lo que es una conexión de punto a punto. También es posible la conmutación a cierto rango de destinos mediante el empleo de una red telefónica ordinaria, un teléfono y un dispositivo conocido como acoplador acústico (que haría la función de módem, pero con tonos de sonido) y aunque pudiera ser más confiable el uso de un módem, el acoplador acústico puede ser más apropiado para ciertos clientes (móviles) con terminal portátil.

Una limitación de las redes con conmutación de circuitos (al ser empleadas para datos) es su inhabilidad para proporcionar conexiones con ancho de banda variable. Lo cual lleva al uso ineficiente de recursos cuando sólo se requiere un ancho de banda estrecho (o velocidad baja de bits). Inversamente, cuando se



necesitan ráfagas cortas de ancho de banda grande pueden existir retardos en la transmisión de datos.

A principios de los años 70 apareció un método más eficiente de envío de datos, éste es la conmutación de paquetes. Se llama así porque todo el mensaje del usuario se descompone en cierto número de paquetes más pequeños, cada uno de los cuales se envía por separado. Cada paquete se rotula con la dirección de su destino, agregándose varios campos de control antes de ser enviado. El extremo receptor re-ensambla los paquetes en el orden apropiado, con ayuda de los números de secuencia. Con una *red de datos con conmutación de paquetes*, el ancho de banda variable se hace posible porque las centrales (computadoras con gran capacidad de almacenamiento de datos y logística de gran riqueza) pueden operar en la modalidad de almacén y envío, enrutando cada paquete a través de la red en la forma más eficiente disponible en el momento; es decir que cada paquete, puede tomar diferente trayectoria a través de la red, inclusive, se mezclan con paquetes de otros mensajes formando lo que se conoce como “canal lógico o virtual”. Los paquetes se insertan a través de las trayectorias individuales dentro de la red, de acuerdo con las condiciones prevalecientes de tráfico, la confiabilidad de error y la trayectoria más corta hacia el destino. Las rutas que se eligen se controlan mediante la logística (capa 3) del conmutador de paquetes, junto con información de enrutamiento preestablecida por el administrador de la red.

La conmutación de paquetes es confiable y eficiente en el empleo de enlaces y recursos de la red, por lo que su uso se ha incrementado y ha contribuido en gran manera a la eficiencia y flexibilidad de las redes de datos de área extensa; sin embargo; la conmutación de paquetes no resulta muy eficiente para redes de escala pequeña, por ejemplo, aquellas limitadas a enlazar computadoras personales casi idénticas dentro de una sola oficina; éste sería el dominio de un tipo de red alternativo conocido como *red de área local*, más conocida por sus siglas en inglés como LAN (*Local Area Network*), éstas son las redes que en los últimos años se han reconocido como el medio más importante de envío de datos



entre diferentes máquinas dentro de una misma oficina o edificio porque están idealmente adaptadas para el transporte de datos a corta distancia. Debido a sus limitadas dimensiones, son redes muy rápidas en las cuales cada estación se puede comunicar con el resto. Están restringidas en tamaño, lo cual significa que el tiempo de transmisión es mínimo. Además, simplifica la administración de la red. Suelen emplear tecnología de difusión mediante un cable sencillo (coaxial o UTP) al que están conectadas todas las máquinas. Operan a velocidades entre 10 y 100 Mbps.

Una LAN de alta velocidad puede transportar altos volúmenes de datos con tiempos rápidos de respuesta. Tal funcionamiento es importante para la mayoría de las aplicaciones de oficina y ha sido el fundamento ideal para la nueva generación de “oficinas electrónicas” que incluyen estaciones electrónicas de trabajo, procesadores de palabras, impresoras compartidas, gabinetes de archivo electrónico, sistemas de correo electrónico, etc.³¹

Una LAN puede consistir de sólo dos o tres PC conectadas para compartir recursos o puede incluir varios cientos de computadoras diferentes que se encuentren en el mismo edificio o en un grupo de edificios adyacentes. Esta red permite a todas las computadoras conectadas a ella compartir hardware, software e información. Los recursos más comúnmente compartidos son los dispositivos de almacenamiento de disco y las impresoras.

Las LAN pueden tener distintas topologías. Todas ellas comprenden una trayectoria única de transmisión que interconecta a todos los dispositivos terminales de datos, con una velocidad de bits típica entre 1 y 30 Mbit/s, junto con los protocolos que hacen posible la transferencia de datos. Las tres topologías comunes en las LAN son la de estrella, de anillo y de bus (línea). Los estándares de protocolo suelen variar ligeramente dependiendo de la topología usada.

El siguiente paso después de la LAN podría ser la MAN, *Metropolitan Area Network* o red de área metropolitana. Estas redes comprenden una ubicación geográfica determinada por lo que sirven de enlace entre edificios de oficinas

³¹ Enrique Herrera Pérez. *Introducción a las telecomunicaciones modernas*. México, Limusa, 2001, p. 211.



dentro de una misma ciudad. Su arquitectura comprende dos buses unidireccionales, cada uno de ellos es independiente del otro en cuanto a la transferencia de los datos. Básicamente se considera una versión a escala mayor de una LAN ya que usa una tecnología similar.

Los sistemas telefónicos celulares amplían la flexibilidad de las MAN, ya que permiten el establecimiento de enlaces con teléfonos de automóviles y portátiles.

Las redes de área extensa o WAN (*Wide Area Network*) son redes a escala nacional y mundial ya que interconectan países y continentes, las líneas utilizadas para esta interconexión suelen ser parte de redes públicas de transmisión de datos; entre otros tipos de canales, utilizan relevadores de microondas y satélites para llegar a usuarios a lo largo de grandes distancias. Al tener que recorrer una gran distancia sus velocidades son menores que en las LAN aunque son capaces de transportar una mayor cantidad de datos. Una WAN está formada por una vasta cantidad de computadoras interconectadas (llamadas *hosts*), por medio de subredes de comunicación o subredes pequeñas, con el fin de ejecutar aplicaciones, programas, etc. Es común que las redes LAN se conecten a las redes WAN para tener acceso a mejores servicios, como por ejemplo a Internet; esto obliga a las WAN a enrutar correctamente toda la información proveniente de las redes conectadas a ellas.

Entre las principales razones de la popularización y el éxito de Internet está el hecho de que es una *red abierta*. Debido a que el protocolo utilizado por las computadoras que se conectan a Internet (TCP-IP) es gratuito, cualquier red y cualquier computadora pueden conectarse sin más costes que los de la conexión. Como ya lo habíamos mencionado antes, no hay ningún propietario de Internet, tampoco una autoridad central que pueda imponer un precio o unas condiciones diferentes de las estrictamente técnicas. Pero esta situación implica la existencia de la impunidad en delitos computacionales y la importancia de la seguridad en computación, tema que trataremos a continuación.



4.5. Seguridad en red

Son amenazas a la seguridad de computación delitos computacionales como virus, irrupciones electrónicas y otros riesgos naturales y de otra especie. Las medidas de seguridad consisten en codificación, acceso restringido, anticipación de desastres y realización de copias de respaldo.³²

Para que la información pueda ser efectivamente privada, es necesario proteger a los sistemas de cómputo contra delincuentes, riesgos naturales y otras amenazas. Un delito computacional es un acto ilegal cometido por una persona provista de conocimientos especiales sobre la tecnología de la computación. En repetidas ocasiones, la categoría más grande de delincuentes computacionales la componen los mismos empleados de las empresas (por el fácil acceso a los sistemas y al equipo). Es frecuente que el empleado no pretenda otra cosa que hurtarle algo al jefe, como: equipo, software, fondos electrónicos, información propietaria o tiempo de cómputo; pero muchas veces lo hace pensando en “vengarse” de la compañía por algún resentimiento. Pero no sólo los empleados, sino también proveedores o clientes pueden tener acceso al sistema de cómputo de una compañía para obtener contraseñas confidenciales o encontrar alguna otra manera de cometer algún delito de cómputo.

Existen personas que, valiéndose de sus habilidades técnicas, obtienen acceso no autorizado a un sistema de cómputo con la intención de divertirse y causar complicaciones; a estos piratas se les reconoce como “hackers”. Pero a las personas que hacen esto con intenciones más maliciosas como hurtar información técnica o introducir en los sistemas software tipo “bomba” (programas destructivos) se les llama “crackers”.

También el crimen organizado ha descubierto que las computadoras pueden usarse tal como lo hace la gente de negocios real, pero con propósitos ilegales; por ejemplo, son útiles para controlar bienes robados o deudas de juego ilegales, inclusive para falsificar documentos, dinero, cheques, etc.

³² T. J. O’Leary, op. cit., p. 277.



El delito computacional puede adoptar varias formas: desde los daños causados por los empleados disgustados al destruir computadoras, introducir instrucciones para destruir o modificar software (como el programa caballo de Troya), o eliminar archivos, hasta la producción con fines de lucro de software pirata, lo cual es ilegal y repercute directamente en un daño económico al desarrollador del programa original. Incluso colarse a una red de cómputo y dejar un mensaje en broma puede parecer divertido (según los piratas), pero también es ilegal y castigado por la ley aunque la manipulación realizada sea aparentemente inofensiva.

En los últimos años los virus informáticos han ganado amplia notoriedad. Los virus son programas que “migran” por redes y sistemas operativos y se insertan en programas y bases de datos. Una variante de virus es el llamado “gusano” (worm). Este programa destructivo invade sistemas de cómputo con información duplicada por sí sola, lo que los obstruye al punto de retardar o detener las operaciones, por ejemplo; en 1988 un gusano conocido como “gusano de Internet” recorrió buena parte de América del Norte, interrumpiendo a su paso el funcionamiento de miles de computadoras. Los virus comenzaron a introducirse en las microcomputadoras a través de discos flexibles copiados o programas descargados de tableros electrónicos, posteriormente con el uso del correo electrónico, los virus encontraron nuevas rutas de propagación. Dada su peligrosidad, es recomendable que los usuarios de computadoras sean cautos al aceptar nuevos programas y datos procedentes de otras fuentes para reducir el riesgo de contraer un virus informático y atenuar sus potenciales daños. No obstante que existen programas detectores de virus, cada día se desarrollan otros nuevos de modo que no todos ellos pueden ser detectados, por esto los usuarios lidian una constante batalla para mantener limpios sus sistemas. El objetivo de los programas antivirus es buscar virus y troyanos en el ordenador (disco duro y memoria) y eliminarlos; en la actualidad, muchos de ellos también revisan los mensajes que llegan por correo y los archivos que el usuario “baja” de Internet. Sin embargo, se debe considerar que la eficacia de un antivirus depende de su actualización, por lo cual es importantísimo mantenerlo actualizado. No obstante, esta recomendación no



siempre es considerada, por lo que existe un constante riesgo y depende totalmente del usuario tratar de prevenirlo, por ejemplo cuando aparece un archivo ejecutable (que el usuario no haya solicitado) en el buzón de correo, es importante no ejecutarlo incluso cuando el remitente sea alguien conocido. Algunos virus usan la agenda del equipo infectado para mandar mail con el virus contenido en un gracioso “attach” o un archivo adjunto. Para evitar la infección, lo más recomendable es borrar este tipo de mensajes sin abrirlos.

Por otro lado, cada vez que se envía información por una red, existe la posibilidad de acceso no autorizado. Entre mayor sea la distancia que el mensaje tiene que recorrer, mayores son también los riesgos a su seguridad. Por ejemplo, un mensaje de correo electrónico en una LAN involucra a un número limitado de usuarios en operación de medios controlados, como oficinas. Pero un mensaje de correo electrónico que tiene que recorrer todo el país por la carretera nacional de la información es mucho más vulnerable a interceptaciones. Una de las soluciones a este problema es el uso de la **codificación** de mensajes, práctica adoptada por las empresas desde hace ya muchos años y que se ha ido perfeccionando pero a la vez que ha sido necesario controlar mediante el empleo de un procedimiento estándar de codificación especial para las empresas usuarias de Internet, con el fin de que los organismos de vigilancia no se vean impedidos a descifrar mensajes de “sospechosos”.

Los expertos en seguridad no dejan de idear maneras de proteger a los sistemas de cómputo contra el acceso por parte de personas no autorizadas. En ocasiones, para garantizar la seguridad basta con colocar vigilantes en las salas de cómputo de las compañías y en requerir identificación de los usuarios. Otras veces, es necesario asignar cuidadosamente contraseñas a las personas y modificarlas cuando un empleado abandona la compañía.

Las **contraseñas** son palabras o números secretos que deben introducirse en un sistema de cómputo para tener acceso a él. En sistemas de cómputo de “devolución de llamada”, el usuario telefona a la computadora, marca la



contraseña correspondiente y cuelga. La computadora devuelve la llamada a cierto número preautorizado.³³

En algunos sistemas de seguridad más sofisticados se hace uso de la *biométrica*, ciencia que estudia la medición de características físicas individuales. Así, las máquinas son capaces de identificar las huellas digitales, firma, voz o hasta fotografías de una persona.

Casi todas las grandes organizaciones cuentan con un *plan de recuperación de desastres*, en el que se describen los medios para mantener las operaciones mientras se restaura el funcionamiento normal de las computadoras. Otra acción que han emprendido estas organizaciones es establecer convenios entre sí para compartir sus recursos (equipo) en caso de desastre y algunas otras cuentan con sus propias instalaciones de emergencia que pueden ser centros de cómputo totalmente equipados (sitios calientes) o salas vacías en espera de que se instale hardware (sitios fríos).

Por todo lo anteriormente expuesto, podemos considerar la seguridad como la protección de la información así como del hardware y software contra uso no autorizado, así como contra daños por intrusiones y sabotaje.

En cuanto a otros peligros que amenazan un sistema de cómputo podemos citar los desastres naturales (causados por incendios, inundaciones, ventiscas, huracanes, temblores), las fallas tecnológicas e incluso errores humanos (que son inevitables). Este tipo de riesgos sólo tiene una solución preventiva: guardar copias de respaldo de la información en lugares protegidos (contra incendios o tormentas) y contar con el equipo de protección contra descargas eléctricas.

³³ T. J. O'Lear, op. cit., p. 281.



Bibliografía del tema 4

Herrera Pérez, Enrique *Introducción a las telecomunicaciones modernas*, México, Limusa. 2001.

Norton, Peter *Introducción a la computación*, 3ª ed., México, Mc Graw Hill, 2006.

O' Leary, Timothy J. y Linda I. O`Leary *Computación básica*, México, Mac Graw Hill. 1997.

Referencias de Internet:

<http://mx.geocities.com/alfonsoaraujocardenas/topologias.html>

http://es.wikipedia.org/wiki/Usuario:Javier_betancort

http://es.wikipedia.org/wiki/Arquitectura_de_red

<http://www.forest.ula.ve/~mana/cursos/redes/protocolos.html>

Actividades de aprendizaje.

A.4.1 Elabora una definición propia del término “red” y ejemplifica mediante una ilustración cada una de las topologías descritas en este tema, explicando brevemente en qué consiste cada una.

A.4.2 Esquematiza el modelo de interconexión de sistemas abiertos ISA.

A.4.3 Elabora una tabla donde se muestren las características de cada tipo de red con el fin de tener claras las diferencias entre ellos.



Cuestionario de autoevaluación

1. Una red puede ordenarse o configurarse de varias maneras siendo la forma elegida un factor fundamental que va a determinar el rendimiento y la funcionalidad de la red. ¿Cómo se le llama a esta disposición?
2. ¿Cuáles son las cuatro principales topologías de red?
3. ¿Qué sucede en un sistema de red terminal?
4. Da una definición del término “red”
5. ¿Cuáles son los principios básicos de transmisión de datos?
6. Para que la transmisión de datos sea exitosa, emisor y receptor deben seguir una serie de reglas de comunicación para el intercambio de información. ¿Cómo se llaman estas reglas para el intercambio de datos entre computadoras?
7. La *International Standards Organization* definió una serie de protocolos de comunicaciones llamada *Open Systems Interconnection* ¿cuál fue el propósito?
8. ¿Cuál es el principio del modelo de interconexión de sistemas abiertos?
9. ¿Cómo se le llama al conjunto de comandos y las especificaciones de sincronización utilizados por Internet?
10. Los servicios más importantes de TCP/IP son:
11. La estructura interna del *World Wide Web* está construida sobre un conjunto de reglas, mejor conocidas como _____
12. ¿Cómo se llama al lenguaje de descripción de página?



13. ¿Cómo se llama al formato especial de direcciones Internet que utiliza HTTP?
14. ¿Que son los URL?
15. Menciona los tipos de redes.
16. ¿En que consisten las medidas de seguridad?
17. ¿Qué son las contraseñas?

Examen de autoevaluación

1. Un nodo se define como:
 - a) Dispositivo E/E conectados a la red
 - b) Cualquier dispositivo conectado a la red
 - c) Un paquete de información discreta viajando por la red

2. Es el que se encarga de solicitar un servicio en un esquema cliente/servidor:
 - a) Servidor
 - b) Ruteador
 - c) Cliente.

3. Son elementos básicos para realizar comunicación entre dos dispositivos
 - a) Cable, protocolo y red
 - b) Emisor, identificador y receptor
 - c) Emisor, medio y receptor.

4. Un medio de comunicación:



- a) Interconecta redes y computadoras y transporta datos.
 - b) Es un medio virtual para enviar información.
 - c) Sólo sirve para el transporte de datos.
5. Es un medio de transmisión inalámbrico:
- a) Microondas
 - b) Fibra óptica
 - c) Multiplexor
6. Un medio de transmisión es:
- a) Un teclado
 - b) Una PC
 - c) La atmósfera
7. Método de transmisión que tiene distintos niveles para representar unos y ceros binarios ejemplo: on y off
- a) Banda base
 - b) Analógico
 - c) Digital
8. La función del control de flujo es sincronizar:
- a) Acceso a la red.
 - b) Emisor- receptor.
 - c) Relojes internos.
9. Las señales _____ se representan por ondas continuas mientras que las _____ sólo pueden tomar un número finito de valores en ciertos instantes de tiempo.



- a) Digitales, analógicas.
- b) Magnéticas, analógicas
- c) De radio, conmutadas.

10. Es la topología de red en la que existe un nodo central que realiza todas las tareas de computación en la red, mediante un enlace punto a punto generalmente semi-duplex, entre los nodos restantes llamados nodos periféricos:

- a) Bus
- b) Estrella
- c) Anillo



TEMA 5. PROYECCIÓN FUTURA DE LA INFORMÁTICA

Objetivo particular

Después de leer este tema, el alumno debe ser capaz de describir el papel que jugarán las computadoras en nuestras casas, trabajo y formación profesional a medida que se evoluciona hacia una conectividad global.

Temario detallado

- 5.1. El futuro de la Informática.
- 5.2. Mercado de trabajo y áreas de especialización.
- 5.3. Perspectivas de la Informática a escala internacional.

Introducción

Podríamos asegurar que la implementación de redes y la comunicación son el futuro de la computación. Actualmente estamos presenciando una carrera hacia la conectividad global, con avances que se están alcanzando en casi todos los aspectos: La tecnología de conexión en red se está volviendo más compleja y las velocidades de transmisión son cada vez más rápidas; todas las industrias de telecomunicaciones están trabajando para ofrecer grandes anchos de banda a hogares y negocios. Por estas y otras razones dedicaremos este último tema a analizar lo que se espera de la ciencia de la información en un futuro de acuerdo con los avances que se dan en nuestros días y cómo influyen éstos en la creación de nuevas carreras informáticas.

5.1. El futuro de la Informática

La era de la información ha redefinido nuestro entorno; es como si la especie humana hubiera sido trasplantada en otro mundo. Aún cuando el cambio se ha producido en un lapso de tiempo muy corto, muchos de nosotros no podemos



imaginarnos un mundo sin computadoras y se podría asegurar que cada día dependemos más de ellas, a tal grado que hoy en día tanto los consumidores de información como la industria de las telecomunicaciones se están aplicando hacia la meta común de la conectividad masiva. Cada grupo se ve en un futuro en donde el ancho de banda sea barato para que las personas puedan mantenerse en contacto sin importar dónde estén, y que la variedad de información disponible sea virtualmente ilimitada.

Por otra parte, el interés en multimedia ciertamente se incrementará en el futuro cercano, pero ya no aparecerá como una categoría de software por separado. Aún más, los elementos de multimedia (sonido, video, interacción) serán integrados en toda clase de documentos y programas. Lo que hoy son los nuevos medios (así se le conoce a la multimedia), serán los medios estándar en un futuro no muy lejano.

En cuanto a los sistemas de información empresariales, se espera que se vuelvan más inteligentes para procesar el amplio rango de información que la compañía acumule. Los científicos de computación están examinando formas para cambiar de bases de datos pasivas a “activamente inteligentes”. En este escenario, los depósitos de datos agruparían la información de una compañía, la cual podría ser explotada por la base de datos. En otras palabras, un agente de software o algún otro mecanismo podrían usarse para encontrar tendencias en la información antes de que el administrador comience a buscarlas.

Conforme las compañías continúen reduciéndose, importarán más funciones de sistemas de información y la interconexión de redes distribuida se volverá aún más predominante. Como resultado, el software para comunicaciones y mensajes es ahora un mercado en expansión pues las compañías están buscando una mejor compatibilidad y desempeño en estos productos de software.

Algunos expertos ya han pronosticado la aparición de la *oficina sin papeles*, una oficina del futuro en la que los archivos magnéticos y ópticos sustituirán a los libros y a los archivadores, la comunicación electrónica a las cartas y los informes, y las



publicaciones web a los periódicos y otras publicaciones. En estas oficinas, la gente leerá pantallas de una computadora y no documentos en papel. Esta tendencia ya es una realidad en algunos aspectos: los dispositivos de almacenamiento digital están sustituyendo a muchos depósitos de papel, las computadoras actuales llevan más mensajes de correo electrónico que los carteros y la Web ha acelerado la tendencia hacia la publicación *online*. Pero a pesar de estos avances, los ejecutivos o los empleados no han sido capaces de reducir en forma significativa el flujo de información basada en papel, lo que sí han cambiado es el modo en que lo utilizan, ya que de ser un dispositivo de almacenamiento ha pasado a ser una interfaz, incluso un medio de visualización volátil y temporal que poco a poco sea erradicado; por lo pronto, se espera ver que en las oficinas se siga usando menos papel gracias a tecnologías como HTML y PDF (formato de documento portátil) de Adobe que facilitan la transmisión de documentos y su almacenamiento sin perder el formato.

En el aspecto de la programación, la calidad de los programas de aplicación disponibles para los usuarios mejora cada año. Una tendencia en desarrollo dentro de la industria de la programación hoy en día es una dependencia creciente en componentes reusables de software. En el futuro cercano, los programadores o equipos de desarrollo podrán comprar a los proveedores la mayoría de los objetos componentes del programa que necesiten, así como hacen ahora con los componentes de hardware como impresoras y estaciones de trabajo. En este punto, el enfoque de trabajo de los programadores cambiará a la integración de los objetos, no al desarrollo personalizado de programas desde el principio.

Una segunda tendencia en la programación es el resultado de la presencia en expansión de Internet en nuestras vidas personales y de negocios. Pronto, los programadores podrían emplear Internet para mucho más que buscar y enviar archivos o mensajes. Podrían estar usándolo para ejecutar programas. Los programas que se corran desde Internet tendrán acceso a los inmensos recursos mundiales.



Esta situación traerá consigo cambios sorprendentes en el desempeño laboral, ya que esta tecnología permitirá que crezca cada vez más el número de profesionistas modernos que trabajen desde sus hogares. En la actualidad, en el mundo hay cientos de miles de escritores, programadores, contadores, grabadores de datos y otros trabajadores de la información para quienes el *teletrabajo* por módem ha sustituido a las horas de tráfico necesarias para llegar a sus puestos de trabajo.

El futurólogo Alvin Toffler popularizó el término **casa electrónica** para describir una casa donde la tecnología permitiera a una persona trabajar en ella. Toffler y otros predijeron que el número de teletrabajadores se dispararía en las próximas décadas.³⁴

Algunos analistas sugieren que, a medida que se abaraten los sistemas de teleconferencia multimedia, el trabajo se convertirá en una situación más popular entre trabajadores y directivos, ya que entonces será posible que ambas partes tengan una telepresencia en el lugar de trabajo aunque no estén físicamente presentes. Además las PC cada vez más potentes y pequeñas así como los dispositivos de bolsillo inalámbricos permiten que vendedores, ejecutivos, consultores, ingenieros y otras personas lleven sus oficinas consigo a cualquier lado al que vayan, por ello es que muchos expertos predicen que el **m-commerce** (comercio móvil, *mobile commerce*) seguirá expandiéndose entre docenas de profesiones en las próximas décadas.

No obstante; afirma Beekman, nadie sabe a ciencia cierta cómo afectará la informática al empleo en las próximas décadas; los expertos están lejos de ser unánimes en sus predicciones, especialmente desde la caída económica del 2001. La mayoría de ellos están de acuerdo en que la tecnología de la información provocará dolorosos períodos de ajuste para muchos trabajadores de fábricas, oficinistas y otros empleados semicualificados, quienes verán cómo sus trabajos se automatizarán o se desplazarán a países del tercer mundo. Pero muchos otros también piensan que la demanda de profesionales (en especial ingenieros,

³⁴ George. Beekman, *Introducción a la informática*, Madrid, Pearson Educación, 2005, p. 403.



profesores y personal sanitario en todas sus vertientes) aumente como resultado de estos cambios en la economía de la información.

La era de la información no sólo afecta al empleo. Sus influencias también se están dejando sentir en nuestro sistema educativo. Y, por encima de todo, la revolución de la información tendrá un impacto profundo y permanente sobre el modo de enseñar. Es claro que la informática puede tener un impacto positivo en la enseñanza, pero las computadoras por sí solas no pueden garantizar una mejora. La investigación, la planificación, la formación al profesorado, la implicación de la comunidad y la reestructuración de las aulas deben acompañar a la nueva tecnología.

Así mismo; es de esperar que se vayan realizando mejoras en el tipo y anchura de la información disponible enviada desde instituciones para aprendizaje a distancia, lo cual ayudará a formar el futuro de la educación. Un ejemplo claro del compromiso que tienen las instituciones de educación superior al proporcionar educación en línea, lo refleja el hecho de que más de 100 universidades en Estados Unidos están desarrollando una nueva infraestructura para Internet, llamada Internet2, cuyo objetivo es proporcionar conexiones rápidas para que los educadores e investigadores compartan y transmitan información alrededor del mundo. Internet2 está siendo construida sobre una red multimedia de banda amplia (ancho de banda de alta capacidad) con el propósito de proporcionar una arquitectura para la colocación de librerías digitales, teleinmersión (similar a ambientes de realidad virtual) y laboratorios virtuales.

También se espera un crecimiento en el número de familias que usa las computadoras en sus hogares para realizar pequeños negocios, formarse, acceder a la información, comunicarse o entretenerse y que todas estas aplicaciones cambien radicalmente a medida que la tecnología evolucione.



En conclusión, la tecnología de la información está teniendo una profunda influencia en la forma en la que vivimos y trabajamos, y probablemente cambie muchas de nuestras creencias, suposiciones y tradiciones.

5.2. Mercado de trabajo y áreas de especialización

Las computadoras y las redes están transformando el mundo en forma rápida e irreversible. Trabajos que han existido durante cientos de años han desaparecido por culpa de la automatización, mientras que están surgiendo otras nuevas profesiones a la sombra de esta emergente tecnología. Las empresas de las nuevas tecnologías crean mercados de la noche a la mañana mientras que las empresas tradicionales luchan por no perderles el paso.

Más que cualquier otra tecnología, las computadoras son responsables de los profundos cambios sufridos en nuestra sociedad. Como ya hemos mencionado, basta imaginar un mundo sin ellas para reconocer su impacto, cada año aparecen nuevos avances técnicos que permiten que estas innovaciones realicen labores que anteriormente estaban reservadas a los humanos. Por supuesto, los informáticos no son los únicos responsables de toda esta turbulencia tecnológica. Los desarrollos en áreas tan diversas como las telecomunicaciones, la ingeniería genética, la medicina, y la física atómica también contribuyen tanto a aumentar la velocidad del cambio como a globalizar los mercados de trabajo.

Por supuesto que los investigadores de todos estos campos dependen de las computadoras para llevar a cabo sus trabajos, lo cual quiere decir que la tecnología informática también crea nuevos puestos de trabajo ya que alguien tiene que diseñar, construir, programar, vender, ejecutar y reparar las computadoras, los robots y las redes.

Pero muchos de estos trabajadores externos no tienen la preparación o los conocimientos necesarios como para programar una computadora, diseñar un robot, instalar una red o, incluso, leer un informe. Con frecuencia este tipo de personal es forzado a realizar tareas que precisen de poca preparación técnica, servicios de bajo costo como cajeros o vigilantes. Debido a la



automatización, un trabajador inexperto y sin formación puede encontrar trabajos con muy poca remuneración y, con frecuencia, escasas condiciones de salud. La tecnología puede ayudar a crear una sociedad desequilibrada con dos clases: un creciente grupo de personas pobres y sin educar, y otro de menor tamaño pero con riqueza y educación.³⁵

A pesar de esta situación, es tremendamente excitante considerar las oportunidades surgidas a raíz de los avances en la inteligencia artificial, la multimedia, la robótica y cualesquiera otras tecnologías de la revolución electrónica. Esto puede observarse a escala mundial, aunque en los países en desarrollo los nuevos profesionistas aún sufren dificultades para adoptar la tecnología de punta que les permita desarrollar sus conocimientos y ampliar sus oportunidades. Así mismo, muchas profesiones relacionadas con la informática aún no son debidamente reguladas y reconocidas, lo que puede derivar en abusos o subempleo. Por ello es importante realizar estudios que muestren la demanda del mercado para estas profesiones y los puestos sean debidamente definidos y valuados.

La *Asociación de Ingenieros e Ingenieros Técnicos en Informática (ALI)* en Madrid publicó un estudio del mercado laboral en Informática durante el año 2006. En él se trata de puntualizar qué tipo de titulaciones se requieren para cada tipo de puesto de trabajo, y qué puestos de trabajo se ofertan. Aunque este estudio se realizó en España, es interesante conocer los resultados obtenidos ya que es probable que exista la misma tendencia en otros países, incluyendo México. Una de las conclusiones obtenidas fue que, para un puesto determinado, tal como *Director de Informática*, la titulación requerida puede ser muy variada, aunque generalmente se solicitan licenciados. También se ve que la oferta de puestos de trabajo en 2001 es ligeramente inferior a 1999, y casi la mitad del 2000 lo cuál puede ser consecuencia de que no existe una clara definición de las funciones de cada puesto ofertado y la ausencia de regulación del mercado de trabajo de esta profesión.

³⁵ G. Beekman, *Introducción a la informática*, ed. cit., p. 409.



A continuación se transcribe el contenido del estudio, el cual se obtuvo de la página <http://www.ali.es/modules/miprofesion/item.php?itemid=20> en Internet:

EL MERCADO LABORAL EN INFORMÁTICA 2006

Realizado sobre una muestra de ofertas de empleo (sector privado y público) publicadas en los diarios ABC, EL PAÍS, EXPANSIÓN, EL MUNDO; revistas especializadas COMPUTING, COMPUTERWORLD y recibidas en nuestra Secretaría Técnica.

Denominación	Total de ofertas (%)	Ingeniero en informática (%)	Ing. Técnico en informática (%)	FP en informática (%)	Titulación universitaria de 2º ciclo (%)	Titulación universitaria de 1º ciclo (%)	No especifica titulación universitaria (%)
Dirección de informática	3,0	50,00	---	---	36,36	---	13,64
Analista de sistemas/ funcional	2,3	23,53	23,53	---	17,64	---	35,30
Analista de aplicaciones/ orgánico	8,4	16,12	14,51	1,62	9,67	3,22	54,86
Analista programador	8,5	25,39,	17,46	1,58	14,28	1,58	39,71
Programador	23,3	9,88	8,14	19,19	5,82	4,06	52,91
Jefe de proyecto	8,4	27,42	12,90	---	25,80	3,22	30,66
Sistemas	7,3	27,78	12,96	7,41	25,92	1,85	24,08
Seguridad informática	3,9	24,13	17,24	---	17,24	6,89	34,50
Ingeniero de Software	4,6	44,12	14,71	---	26,47	5,88	8,82
Auditoría Informática	1,9	42,86	35,72	---	14,28	7,14	---
Administrador de redes y/o BD	2,5	11,11	16,66	5,55	5,55	11,11	50,02
Consultor	5,4	20,00	20,00	---	22,50	7,50	30,00
Administración Pública	6,5	12,5	33,33	12,50	29,16	6,26	6,25
Comercial Marketing	1,5	9,09	18,18	---	18,18	27,27	27,28
Ingeniero en Informática	1,8	100	---	---	---	---	---
Ingeniero Técnico en Informática	0,9	100	---	---	---	---	---
Otros	9,8	16,21	21,62	33,78	9,45	6,75	12,19

Todos los datos están expresados en porcentajes.



1. DENOMINACIÓN

En la columna DENOMINACIÓN se ha adoptado los nombres más conocidos que aparecen en la denominación del puesto ofertado.

La columna INGENIERO EN INFORMATICA indica las ofertas que requieren explícitamente titulación universitaria de Ingeniero en Informática o la anterior denominación de licenciado en informática (R.D. 1954/1994 de 30 de septiembre).

La columna INGENIERO TÉCNICO EN INFORMATICA indica las ofertas que requieren explícitamente titulación universitaria de ingeniero técnico en informática de gestión o de sistemas o la anterior denominación de diplomado en informática (R.D. 1954/1994 de 30 de septiembre).

La columna FP EN INFORMATICA indica las ofertas que requieren explícitamente titulación de formación profesional en informática en alguna de sus especialidades de la rama de informática.

La columna TITULACION UNIVERSITARIA DE 2º ó 1º CICLO indica las ofertas que no especifican explícitamente titulaciones universitarias en informática, como por ejemplo titulación universitaria o titulación en matemáticas, titulación en físicas, titulación en ingeniería o ingeniería técnica de telecomunicaciones, etc.

La columna NO ESPECIFICA TITULACION UNIVERSITARIA comprende todas aquellas, que bien indican bachiller superior en el caso de administración pública, o no indican ningún tipo de titulación.

2. En los PERFILES

Detectamos, en algunas solicitudes, ambigüedad en el nombre del perfil y las funciones que se detallan, llegando a confundirse. Por ejemplo, ocurre en las solicitudes de Técnicos de y administradores de sistemas.



En sistemas englobamos: técnico de sistemas, ingeniero de sistemas, administración de sistemas, jefe de sistemas, consultor de sistemas.

En jefe de proyecto englobamos responsable de informática.

En dirección de informática englobamos director de informática, gerente, director de sistemas.

En seguridad englobamos ingeniero seguridad, técnico seguridad, responsable seguridad, consultor seguridad, administrador seguridad, gerente seguridad, mantenimiento seguridad.

En programador englobamos desarrollador:

En las solicitudes de la administración pública existe una gran variedad de perfiles (técnico medio informática, técnico superior informática, técnico soporte, programador, informático, técnico de sistemas, técnico auxiliar informática, analista programador, consultor, ingeniero en informática, analista, ingeniero sistemas, ingeniero desarrollo, técnico, ingeniero técnico en informática, director, administrador de sistemas, formación).

En otros englobamos: mantenimiento, explotación, soporte, formación no universitaria, especialista en redes y comunicaciones, informático (engloba un número de funciones sin especificar, técnico bdd)

CONCLUSIONES

La solicitud de ofertas de perfiles en informática ha disminuido con respecto al año anterior. En 2006 se ha solicitado 739 puestos de trabajo, en 2005 han sido 909 puestos de trabajo, en 2004 han sido 705, 2003 con 743, por debajo del 2002 con 1036, y muy por debajo de los años 1999 con 2348 ofertas, 2000 con 4931 ofertas, y 2001 con 2327 ofertas.



Denominación	2006 Total de Ofertas (%)	2005 Total de Ofertas (%)	2004 Total de Ofertas (%)
Director de informática	3	3,96	4,26
Analista de Sistemas/ Funcional	2,3	1,98	3,12
Analista de aplicaciones/ Orgánico	8,4	6,60	3,69
Analista Programador	8,5	7,15	6,38
Programador	23,3	18,26	14,75
Jefe de proyecto	8,4	5,83	6,24
Sistemas	7,3	15,30	7,52
Seguridad Informática	3,9	5,28	5,67
Ingeniero de Software	4,6	3,52	3,26
Auditoria Informática	1,9	4,07	3,55
Administrador de redes y/o BD	2,5	2,31	3,05
Consultor	5,4	7,04	7,24
Administración pública	6,5	4,49	9,36
Comercial/ Marketing	1,5	2,40	7,94
Ingeniero en informática	1,8	2,50	2,13
Ingeniero Técnico en Informática	0,9	1,50	1,56
Otros	9,8	7,81	10,28

Comparando con el año anterior 2005, podemos observar:

Para el puesto de Director de Informática las organizaciones han solicitado mayoritariamente la titulación de Ingeniero en Informática: 50% de las ofertas para desempeñar este puesto, seguido por otras titulaciones de segundo ciclo (36%). Se puede observar que el importante crecimiento que han tenido otras titulaciones de segundo ciclo para el desempeño de este puesto, que ha pasado del 22% del año anterior al 36% en 2006 se ha hecho en detrimento de la titulación de ingeniero técnico en informática que en el año 2005 fue solicitado en el 16% de las ofertas.

En 2006 se inicia un fuerte impulso en el desarrollo de nuevos proyectos informáticos y todos los puestos de trabajo relacionados con desarrollo experimentan una fuerte demanda:



Los puestos de jefes de proyecto, analistas y programadores totalizan el 51% del total de ofertas recibidas, frente al 40% del año anterior.

La demanda del mercado de puestos de analistas-programadores y programadores representan el 32% del total de ofertas, frente al 25% del año anterior que no han podido ser cubiertas satisfactoriamente por la formación profesional y las empresas han optado por demandar Ingenieros en Informática para cubrir estos puestos o por no solicitar ninguna titulación.

Este dato ha sido ratificado por otras fuentes: “Ingenieros e Ingenieros Técnicos en Informática de las dos últimas promociones que han manifestado a los Colegios y Asociaciones de Ingenieros e Ingenieros Técnicos en Informática que los puestos que les ofrecían eran de “desarrolladores” que es como genéricamente denominan las empresas a estos puestos.³⁶

El puesto de “Jefe de Proyecto” avanza su consolidación en la Ingeniería en Informática: Desde 2005 se están solicitando mayoritariamente las titulaciones de Ingeniero e Ingeniero Técnico en Informática en más del 40% de las ofertas que solicitan este puesto. Aunque resulta alarmante que se haya producido un gran incremento de las ofertas que no exigen ninguna titulación para el desempeño de este puesto que han pasado del 18% en 2005 al 30% en 2006.

Esta alarma ha sido ratificada por otras fuentes: “Ingenieros e Ingenieros Técnicos en Informática de las dos últimas promociones que han manifestado a los Colegios y Asociaciones de Ingenieros e Ingenieros Técnicos en Informática que se están viendo involucrados en Proyectos con grandes deficiencias, cuyos Jefes y Responsables desconocen profundamente la Informática en general y la Ingeniería en Informática en particular”.

El puesto de “Programador” avanza en su consolidación en FP: mientras que en 2005 sólo el 5% de las ofertas especificaban FP para desempeñar este puesto en 2006 ha pasado a ser el 19%, aunque aún está lejos de consolidarse ya que para el 52% de estas ofertas no se especifica ninguna titulación.

³⁶ Consultado el día del mes del año, disponible en <http://www.fi.upm.es/?pagina=587>



Subcontratación de Ingenieros e Ingenieros Técnicos en Informática como “Analistas-Programadores”: Es un dato muy alarmante que para este puesto de FP, en el 43% de las ofertas se haya solicitado la titulación de Ingeniero o Ingeniero Técnico en Informática y tan sólo en 1,5% formación de FP, o que no se especifique ninguna titulación en el 40% de las mismas.

Con carácter general los datos nos dicen con claridad que en 2006 el “Desarrollo de los Proyectos Informáticos ha adolecido de falta de profesionalidad” y las consecuencias futuras en relación con las garantías de calidad y seguridad exigible a estos proyectos son impredecibles. En los mismos han participado, porque así lo han solicitado las empresas que los han llevado a *cabo*:

Denominación	No se especifica Titulación (%)
Jefe de Proyecto	30, 66
Analista de Sistemas/ Funcional	35, 30
Analista de aplicaciones/ orgánico	54, 86
Analista Programador	39, 71
Programador	52, 91

Los datos lo muestran con claridad: “Para una media del 42% del personal contratado profesionalmente para el Desarrollo de Proyectos Informáticos, no sólo no se ha requerido la titulación en informática, sino que las empresas no han requerido ninguna titulación”. ¿Se puede hablar de intrusismo profesional en la Informática a la luz de estos datos? La respuesta es clara.

Auditoria y Seguridad Informática: En 2006 el total de ofertas disminuye considerablemente con respecto a 2005 y a pesar de que este sector en los tres últimos años ha seguido un crecimiento sostenido del 18%. Si tenemos en cuenta que para estos puestos se ha solicitado las titulaciones de Ingeniero o Ingeniero Técnico en Informática: Seguridad Informática en el 41% de las ofertas, Auditoria Informática el 78% de las ofertas. Dadas las características específicas de estos puestos se podría deducir que para cubrirlos, las organizaciones no están



acudiendo a los anuncios en prensa sino a redes de contactos y al reciclaje interno de ingenieros e ingenieros técnicos en informática en estas especialidades.

El puesto de “Ingeniero de Software” avanza su consolidación en la Ingeniería en Informática: se ha pasado del 47% en 2005, al 59% en 2006 en las ofertas que solicitan las titulaciones de Ingeniero o Ingeniero Técnico en Informática para desempeñar este puesto.

Las plazas de Informática en la Administración Pública: es muy positivo el cambio de tendencia, aumentando la contratación, aunque aún lejos de los porcentajes de años precedentes, que se situaba entorno al 10%.

Los puestos de Comercial / Marketing: siguen por tercer año consecutivo en porcentajes insignificantes. Cabría deducir que para cubrir estos puestos las organizaciones no están acudiendo a los anuncios en prensa, sino a redes de contactos. Además para cubrir estos puestos tan solo en el 27% de las ofertas se ha solicitado la titulación de Ingeniero ó Ingeniero Técnico en Informática.

Las profesiones de Ingeniero e Ingeniero Técnico en Informática son completamente desconocidas en la sociedad española: los datos, por tercer año consecutivo, hablan con claridad: se solicita “directamente” un Ingeniero en Informática en el 1,5% de las ofertas y un Ingeniero Técnico en Informática en el 0,9 de las ofertas.

Esta situación, completamente anormal para el resto de Ingenierías, es normal que se produzca en la Informática: La sociedad española “no puede conocer profesiones de Ingeniero” que las leyes no han regulado.

Por otra parte, al analizar las tendencias actuales en cuanto a las profesiones y ocupaciones en los países tecnológicamente más avanzados, se puede llegar a conclusiones prospectivas como las obtenidas por la OCDE (Organización para la



Cooperación y Desarrollo Económico) en estudios recientes y que ratifican que trabajadores del conocimiento ya representan el 80% del empleo en los países avanzados; además se da muestras de que en el futuro el mercado de trabajo y la creación de riqueza se fundamentarán cada vez más en la información y su tratamiento aplicado, es decir, el conocimiento.

Por lo tanto, los expertos apuntan a un nuevo tipo de trabajo más cualificado, complejo y orientado al proceso de datos, en el que la navegación por el ciberespacio será un elemento de exigencia pero que a la vez requiera un menor número de horas que en la actualidad y con una productividad superior.

No cabe duda que el mercado de trabajo del futuro, y las nuevas profesiones y ocupaciones que vayan surgiendo, tendrán que ser cualitativamente distintas de las actuales, pero se empiezan a configurar a partir de buena parte de los avances que ya conocemos y con los que vamos conviviendo.

Por citar algunos ejemplos, los estudios antes citados apuntan a la aparición de dos nuevos ámbitos de empleo, en cuanto al volumen de oferta realizado por las empresas, y que se concreta en los “telecomunicólogos”, profesionales especializados en la conexión extensa de ordenadores y diversos dispositivos electrónicos por medio de redes de telecomunicaciones, vía satélite, cada vez más complejas, rápidas y eficientes que transportarán la información, materia prima de los empleos y trabajos, a cualquier sitio del planeta, por recóndito que sea.

Existirá igualmente un amplio volumen de oferta de empleo vinculada a los denominados “ingenieros moleculares”, especialistas en diseño y manipulación de nuevos materiales, desde las cerámicas y aleaciones especiales, hasta las fibras sintéticas.

Dos ejemplos de lo que se avecina en el futuro, no tan lejano, por cuanto algunas de estas ocupaciones ya se consolidan como realidad en los países tecnológicamente más avanzados en la actualidad. Si se presta atención a las ocupaciones que van a tener una mayor demanda a escala mundial, y se trata de realizar un análisis por grandes sectores y áreas de especialización, las tecnologías de la información, en sentido amplio, van a convertirse en el principal yacimiento de empleo en el futuro y se extenderán a la práctica totalidad de sectores.

Como ya se ha señalado, habrá una oferta masiva vinculada a administradores y analistas de redes y comunicaciones, ingenieros especializados en software, administradores de bases de datos, programadores, especialistas en soporte de programación, digitalización de contenidos, entre otros. Esta área va a crecer de forma espectacular conforme los países en vías de desarrollo vayan incorporándose a los esfuerzos tecnológicos ya impulsados en Occidente, y sobre todo, por la



necesidad de realizar una continua actualización de los equipos y procesos, aplicados tanto a la producción y el consumo, como al ocio y tiempo libre.

Un segundo espacio de nuevas ocupaciones se encuentra en el área prometedora de las ingenierías y sus distintas especialidades, muchas de ellas ya presentes en la actualidad. La biotecnología, combinando disciplinas como la industria química, alimentos, farmacia, va a constituir un yacimiento de empleo destacado, con numerosas profesiones y ocupaciones directamente relacionadas con la investigación y ciencia aplicada.

En tercer lugar, el área de la tecnología geoespacial, vinculada a especialidades profesionales como la ingeniería medioambiental, cartografía digital, fotometría, geocientíficos, ingenierías industriales y eléctricas, aeroespaciales, todo ello combinado con el conocimiento necesario de la informática, va a generar un volumen importante de ocupaciones nuevas vinculadas a los niveles más elevados de desarrollo e innovación³⁷.

Otros sectores, como el de la salud, el industrial e incluso la construcción también podrán tener un importante espacio para nuevas ocupaciones (que exigirán una inversión muy fuerte en conocimiento) con los profesionales especializados que deberán poseer amplios conocimientos informáticos, de esta manera se llevará a cabo una reconversión de los oficios tradicionales al nuevo escenario de la cibernética.

La formación continua, impulsada por la renovación profesional, será la nota característica de un entorno laboral en constante transformación, con clientes y consumidores cada vez más exigentes que cambiarán sus gustos y preferencias con comportamientos hasta ahora desconocidos.

5.3. Perspectivas de la Informática a escala internacional

Ya hemos expuesto lo que los expertos afirman acerca de la informática en un futuro a escala global. Ahora, para observar las implicaciones de una conectividad mundial, comenzaremos por retomar los comentarios de Bill Gates.

³⁷ Consultado el 10/06/08 disponible en: http://www.sve.es/noticias/educaweb_30012006.pdf



El Fundador y accionista mayoritario de Microsoft afirma que en el mundo existen cientos de millones de personas que nunca han utilizado una PC, pero a medida que sus países se modernicen, comenzarán a formar parte de la economía global, entonces las computadoras ayudarán a estas personas a liberar su potencial y conectarse al mundo. Lo cual significa que la industria de la computación aún tiene mucho camino por delante. En una reciente publicación, el magnate del software comenta:

La industria de la tecnología es legendaria por su rápida innovación. Sin embargo, algunos de sus más grandes logros son productos para apostar a largo plazo y mantener ese compromiso año tras año. Por ejemplo, el crecimiento acelerado de Internet a finales de los noventa podría parecer repentino, pero fue el producto de décadas de investigación e innovación.

Muchas de las apuestas más grandes de Microsoft también pueden medirse en años, no en meses. Fuimos de los primeros en creer en la promesa de la "computación con pluma", y después de años de insistencia la Tablet PC ya se está convirtiendo en algo común en oficinas y aulas. La televisión interactiva ha sido otro de nuestros sueños, y nuestras inversiones a largo plazo en esta área están comenzando a ver la luz por medio de la amplia aceptación de IPTV, la cual considero que revolucionará la manera en que concebimos a la TV.

Es por todas estas razones que me siento orgulloso de nuestros logros en Microsoft: de apostar por tecnologías como la interfaz de usuario gráfica o los servicios Web, y verlos crecer y convertirse en algo que la gente utiliza todos los días. La investigación a largo plazo que estamos llevando a cabo actualmente con uno de los retos más difíciles, ayudar a las computadoras a escuchar, hablar, aprender y entender, conducirá a lo que creo que será la próxima ola de crecimiento e innovación para nuestra industria. Claro que existen factores que podrán retrasar el futuro de la informática. Debemos continuar construyendo la confianza en los sistemas informáticos protegiendo a los usuarios de virus, spyware y otras amenazas de seguridad, así como seguir trabajando con la industria y el gobierno para proteger la privacidad de los usuarios y la seguridad de sus hijos en línea.

También debemos innovar para responder a las necesidades de los millones que apenas comienzan a conocer el mundo de la computación, a través de dispositivos que sean adecuados al ambiente donde viven y de software que hable su propio idioma. Desde luego, todos nos beneficiaremos con nuestro trabajo constante en la industria para hacer que las computadoras sean menos complicadas y más económicas.

Anteriormente he dicho que estamos a mitad de la "década digital", un momento donde las computadoras verdaderamente se vuelven un elemento central en la manera en que vivimos y trabajamos y es difícil imaginar vivir sin PC's. Algunos se preguntan si Microsoft está perdiendo el ímpetu



después de 30 años de innovación. Yo diría que apenas estamos empezando.³⁸

Aún para Bill Gates, la súbita popularidad de Internet es increíble, no sólo porque ha permitido expandir la comunicación entre las personas, sino porque ha desencadenado una feroz competencia en la industria informática.

Lo cierto es que Internet ha abierto un mundo de posibilidades para los negocios. El comercio electrónico comparte la información de los negocios, mantiene las relaciones empresariales y dirige las transacciones comerciales mediante el uso de las redes de telecomunicaciones, especialmente Internet. Actualmente, una empresa que realice negocios por Internet debe desarrollar un código de conducta que informe a los usuarios de su sitio web de sus políticas de comercio electrónico, especialmente en lo relacionado con la privacidad de la información, lo cuál ha sido tema preocupante a escala internacional.

Por otra parte, el término “supercarretera de la información” describe el futuro de las redes de comunicación y las computadoras. Los sociólogos han sugerido que los avances más significativos darán pie al desarrollo de lo que se conoce como “aldea global”. Esta aldea global no estará restringida por el espacio geográfico lo está únicamente por el ciberespacio o enlaces de comunicación con computadoras.

Todo indica que los habitantes de la aldea global establecerán entre sí estrechas relaciones sin verse jamás. En efecto, vivirán quizá en distintas partes del mundo, con culturas e historia de lo más diversas.³⁹

³⁸ <http://www.clarin.com/suplementos/economico/2005/10/16/n-03601.htm> consultado el 10/06/08

³⁹ T. J. O’Leary, op.cit., p. G117



Bibliografía del tema 5

Beekman, George *Introducción a la Informática*, 6ª ed., Madrid, Pearson. 2007.

O' Leary, Timothy J. y Linda I. O`Leary *Computación básica*, México, Mac Graw Hill. 1997.

Norton, Peter *Introducción a la computación*, 3ª ed., México, Mc Graw Hill, 2006.

Referencias de Internet:

<http://www.ali.es/modules/miprofesion/item.php?itemid=20>

<http://www.educaweb.com/EducaNews/interface/asp/web/NoticiesMstrar.asp?NoticiaID=982&SeccioID01351>

<http://www.clarin.com/suplementos/economico/2005/20/16/n-03601.htm>

Actividades de aprendizaje.

A.5.1 Elabora un listado de criterios que se utilizan para determinar el futuro de la informática.

A.5.2 Con el fin de captar el nivel de avance en la institución en donde te desenvuelves observa la tecnología que se utiliza y escribe tu opinión acerca de lo que le hace falta.

A.5.3 Redacta un texto donde puedas expresar cómo esperas influir los cambios tecnológicos en tu vida personal y profesional.



Cuestionario de autoevaluación

1. ¿Por qué es importante que el ancho de banda sea cada vez más barato?
2. ¿Qué sucederá con los elementos de multimedia (sonido, video, interacción)?
3. ¿Qué se espera en cuanto a los sistemas de información empresariales?
4. En el futuro cercano, ¿qué sucederá con los programadores o equipos de desarrollo?
5. ¿Cómo afectará la informática al empleo en las próximas décadas?
6. En tu opinión, ¿qué tipos de puestos de trabajo en Informática podrían convertirse en *teletrabajo*?
7. En los países en desarrollo ¿qué pasa con los nuevos profesionistas?
8. ¿Cuáles son los puestos con mayor número de ofertas recibidas en el 2006 según el estudio? ¿qué opinas al respecto?
9. ¿Qué diferencia existe entre el e-commerce y el m-commerce?
10. ¿Crees que las expectativas de los sociólogos respecto a la aldea global se materialicen en un futuro cercano?

Examen de autoevaluación

1. ¿Cuál es la meta común que se están aplicando tanto los consumidores de información como la industria de las telecomunicaciones?
 - a) La información
 - b) La conectividad masiva
 - c) La comunicación



2. El software para comunicaciones y mensajes es ahora un mercado en _____ pues las compañías están buscando una mejor compatibilidad y desempeño en estos productos de software.
 - a) Introducción
 - b) Expansión
 - c) Retracción

3. Los dispositivos de almacenamiento digital ¿a qué están sustituyendo?
 - a) Archiveros
 - b) Plataformas
 - c) Depósitos de papel

4. Una tendencia en desarrollo dentro de la industria de la programación hoy en día, es una dependencia creciente en:
 - a) Componentes reusables de software.
 - b) Componentes de monitoreo
 - c) Sistemas audibles

5. Es claro que la informática puede tener un impacto positivo en la enseñanza, pero ¿qué sucede con las computadoras?
 - a) Se modernizan
 - b) Por sí solas no pueden garantizar una mejora.
 - c) Tienen mejores dispositivos

6. Un segundo espacio de nuevas ocupaciones se encuentra en el área:
 - a) de las ingenierías



- b) geoespacial
 - c) de las ciencias sociales
7. Anteriormente he dicho que estamos a mitad de la " _____ ", un momento donde las computadoras verdaderamente se vuelven un elemento central en la manera en que vivimos y trabajamos y es difícil imaginar vivir sin PC's.
- a) década analógica
 - b) década perdida
 - c) década digital
8. Según Bill Gates; también debemos innovar para responder a las necesidades de los millones que apenas comienzan a conocer el mundo de la computación, a través de dispositivos que sean adecuados al ambiente donde viven y de software que:
- a) escriba en su propio idioma
 - b) lea en su idioma
 - c) hable su propio idioma
9. El término "supercarretera de la información" describe el futuro de:
- a) Las redes de comunicación y las computadoras
 - b) La información actual de las organizaciones
 - c) La investigación de nuevos campos de trabajo



10. Los sociólogos han sugerido que los avances más significativos darán pie al desarrollo de lo que se conoce como:

- a) Aldea global
- b) Comercio global
- c) Educación global



Bibliografía General

- Beekman, George *Computación & Informática hoy, una mirada a la tecnología del mañana*, Wilmington, Delaware, México: Addison Wesley Longman. 1995.
- Beekman, George *Introducción a la Informática*, 6ª ed., Madrid, Pearson. 2007.
- Decker, Rick y Stuart Hirshfield *Máquina analítica, Introducción a las ciencias de la Computación con uso de la Internet*. México, Thomson. 2001.
- Ferreya Cortés, Gonzalo *Informática para cursos de bachillerato*, México, Alfaomega. 2001.
- Herrera Pérez, Enrique *Introducción a las telecomunicaciones modernas*, México, Limusa. 2001.
- Jamrich Parsons, June y Dan Oja *Conceptos de computación: nuevas perspectivas*. México, Thomson Learning. 2004.
- Long, Larry E y Nancy Long *Introducción a las computadoras y a los sistemas de información*, México, Prentice Hall. 1999.
- Norton, Peter *Introducción a la computación*, 3ª ed., México, Mc Graw Hill, 2006.
- O' Leary, Timothy J. y Linda I. O'Leary *Computación básica*, México, Mac Graw Hill. 1997.
- Pressman, Roger S *Ingeniería de software. Un enfoque práctico*, 5ª ed., México, Mc Graw Hill. 2002.
- Scott, George M. y Daniel Cohen *Sistemas de información*, México, Mc Graw Hill. 1996.
- Senn, James A. *Análisis y diseño de sistemas de información*, 2ª ed., México, 1992.



**Respuestas de los exámenes
Informática 1**

	Tema 1	Tema 2	Tema 3	Tema 4	Tema 5
1	c	a	a	b	b
2	b	c	b	c	b
3	a	b	c	c	c
4	a	a	b	a	a
5	b	a	a	a	b
6	c	b	c	c	a
7	b	b	a	b	c
8	c	b	c	b	c
9	c	a	b	a	a
10	a	c	a	b	a