

UNIVERSIDAD AUTONOMA DE NUEVO LEON

**FACULTAD DE INGENIERIA MECANICA
Y ELECTRICA**

DIVISION DE ESTUDIOS DE POSTGRADO



**LA ENSEÑANZA DE LA PROGRAMACION DE
COMPUTADORAS EN EL INSTITUTO TECNOLOGICO
DE NUEVO LEON AL INICIO DEL
SEGUNDO MILENIO**

**POR
ING. FIDEL GARCIA OCHOA**

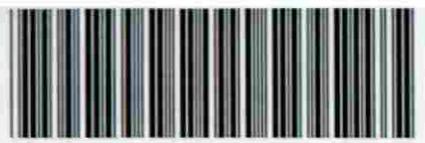
**TESIS
EN OPCION AL GRADO DE
MAESTRO EN CIENCIAS DE LA ADMINISTRACION
CON ESPECIALIDAD EN SISTEMAS**

**SAN NICOLAS DE LOS GARZA, N. L.
ENERO DEL 2002**

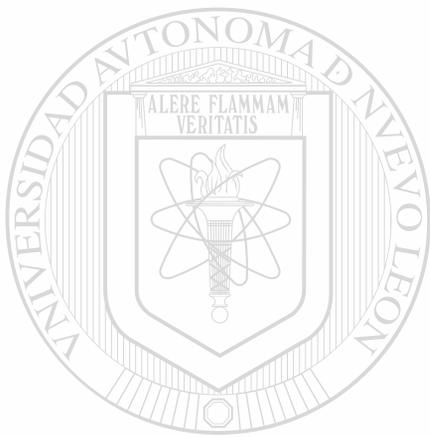
2002
F 93 27 QA76
TM

LA ENSEÑANZA DE LA PROGRAMACION DE
COMPUTADORAS EN EL INSTITUTO TECNOLÓGICO
DE NUEVO LEON AL INICIO DEL
SEGUNDO MILLENIO

F. C. O.
C. C. O.



1080095010



UANL

UNIVERSIDAD AUTÓNOMA DE NUEVO LEÓN

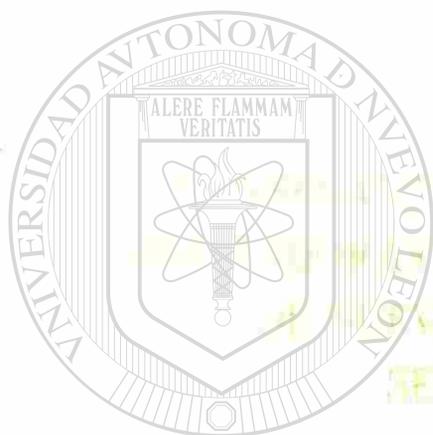


DIRECCIÓN GENERAL DE BIBLIOTECAS

UNIVERSIDAD AUTÓNOMA DE NUEVO LEÓN

FACULTAD DE INGENIERIA MECANICA
Y ELECTRICA

DIVISION DE ESTUDIOS DE POSTGRADO



UANL

UNIVERSIDAD AUTÓNOMA DE NUEVO LEÓN[®]
DIRECCIÓN GENERAL DE BIBLIOTECAS

TESIS

EN OPCIÓN AL GRADO DE

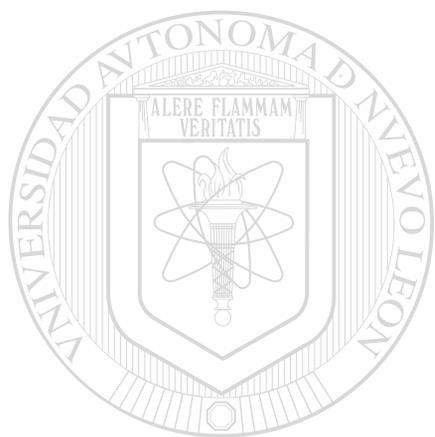
INGENIERO EN CIENCIAS DE LA ADMINISTRACIÓN Y
COMUNICACIÓN EN SISTEMAS

EN NUEVO LEÓN, LOS CARBONEROS, NUEVO LEÓN,

ENERO DEL 2002



T
A76
027
G
e1



UANL

UNIVERSIDAD AUTÓNOMA DE NUEVO LEÓN

®

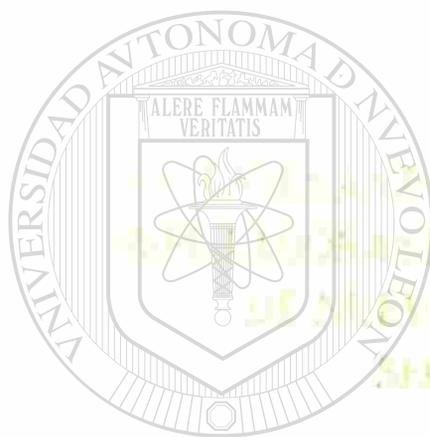
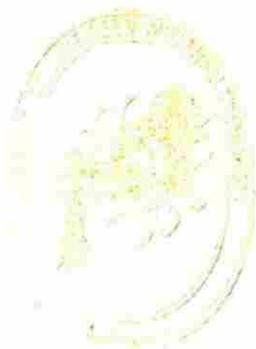
DIRECCIÓN GENERAL DE BIBLIOTECAS



UNIVERSIDAD AUTÓNOMA DE NUEVO LEÓN

FACULTAD DE INGENIERÍA MECÁNICA
Y ELÉCTRICA

DIVISIÓN DE ESTUDIOS DE POSTGRADO



UANL

UNIVERSIDAD AUTÓNOMA DE NUEVO LEÓN

DR. FIDEL GARCÍA CRUZA

®

DIRECCIÓN GENERAL DE BIBLIOTECAS

TESIS

EN OPCIÓN AL GRADO DE

INGENIERO EN CIENCIAS DE LA ADMINISTRACIÓN

CON ESPECIALIDAD EN SISTEMAS

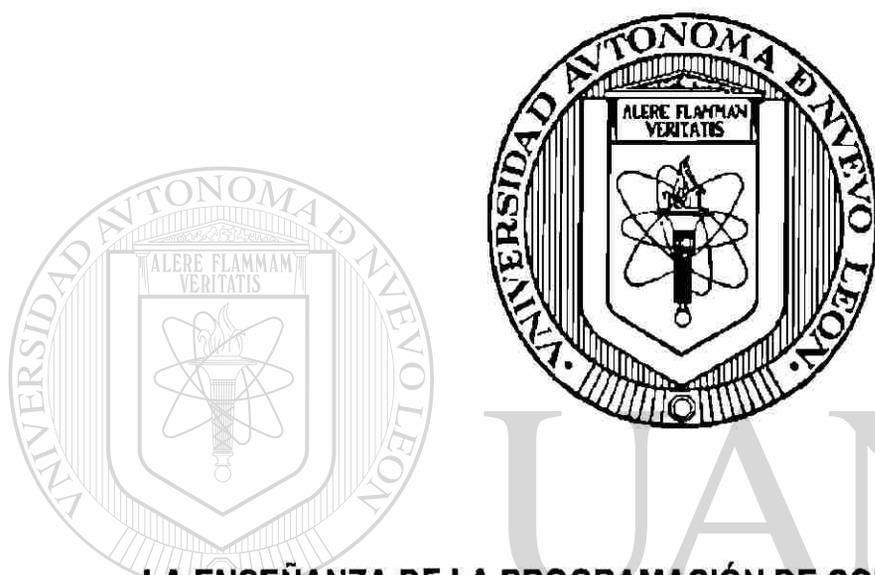
SAN NICOLÁS DE LOS GARZA, T.M.L.

ENERO DEL 2002

UNIVERSIDAD AUTÓNOMA DE NUEVO LEÓN

FACULTAD DE INGENIERIA MECÁNICA Y ELECTRICA

DIVISIÓN DE ESTUDIOS DE POST-GRADO



**LA ENSEÑANZA DE LA PROGRAMACIÓN DE COMPUTADORAS EN EL
INSTITUTO TECNOLÓGICO DE NUEVO LEÓN AL INICIO DEL SEGUNDO
MILENIO**

DIRECCIÓN GENERAL DE BIBLIOTECAS

**POR
ING. FIDEL GARCIA OCHOA**

**T E S I S
EN OPCION AL GRADO DE
MAESTRO EN CIENCIAS DE LA ADMINISTRACIÓN
CON ESPECIALIDAD EN SISTEMAS**

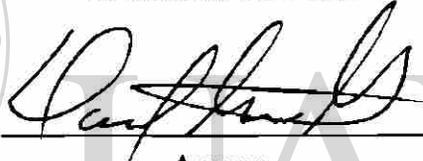
**SAN NICOLAS DE LOS GARZA, NUEVO LEÓN
ENERO DEL 2002**

UNIVERSIDAD AUTÓNOMA DE NUEVO LEÓN
FACULTAD DE INGENIERIA MECÁNICA Y ELÉCTRICA
DIVISIÓN DE ESTUDIOS DE POST-GRADO

Los miembros del comité de tesis recomendamos que la tesis “LA ENSEÑANZA DE LA PROGRAMACIÓN DE COMPUTADORAS EN EL INSTITUTO TECNOLÓGICO DE NUEVO LEON AL INICIO DEL SEGUNDO MILENIO”, realizada por el alumno ING. FIDEL GARCIA OCHOA, matrícula 117924, sea aceptada para su defensa como opción al grado de Maestro en Ciencias de la Administración con especialidad en Sistemas.

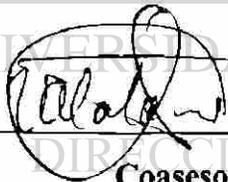


El Comité de Tesis



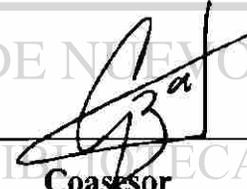
Asesor

M.C. David Garza Garza



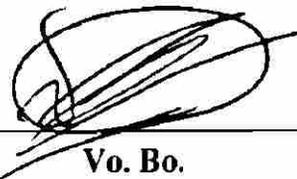
Coasesor

Dr. Victoriano Fco. Alatorre González



Coasesor

M.C. Carlos B. Garza Treviño



Vo. Bo.

M.C. Roberto Villarreal Garza

División de Estudios de Post-grado

San Nicolás de los Garza, N.L., a 12 de Noviembre del 2001

DEDICATORIA

A mi esposa

Por su comprensión y sacrificios al no limitarme en el tiempo necesario para dedicarlo a la culminación de esta meta profesional.



A mis hijos

Por la fuerza interior que han infundido a mi alma para superarme a pesar de mis grandes limitaciones.

UANL

A toda la comunidad del Instituto Tecnológico de Nuevo León

UNIVERSIDAD AUTÓNOMA DE NUEVO LEÓN

Por permitirme ser parte integral de ella, y a través de este proyecto hacer esta pequeña aportación orientada a la superación académica de alumnos y docentes.

DIRECCIÓN GENERAL DE BIBLIOTECAS

AGRADECIMIENTOS

**Bendice, alma mía a Jehová, y bendiga todo mi ser su santo nombre.
Bendice, alma mía a Jehová, y no olvides ninguno de sus beneficios.
Salmo 103:1,2**

Con esta porción de su palabra agradezco a Dios por concederme los beneficios de la fe, así como que por su voluntad tuviese los recursos necesarios para lograr la culminación de esta meta profesional.

Asimismo agradezco al Sistema Nacional de los Institutos tecnológicos por otorgarme las facilidades suficientes para la culminación de este proyecto .

Justo es reconocer también la labor que desarrolla el departamento de Desarrollo académico al realizar con toda buena disposición los trámites requeridos para el desarrollo de proyectos como el presente.

UNIVERSIDAD AUTÓNOMA DE NUEVO LEÓN

No de menor valía es reconocer a la Universidad Autónoma de Nuevo León, específicamente a la Facultad de Ingeniería Mecánica y Eléctrica, que como institución educativa me dio albergue tanto a nivel de licenciatura como en nivel de post-grado, así como también a sus catedráticos por compartir sus conocimientos con este servidor.

PRÓLOGO

La computación es una de las áreas tecnológicas que mas se ha desarrollado en los últimos años, asimismo los lenguajes de programación de computadoras han evolucionado de tal forma que los métodos y técnicas de programación usados actualmente son muy diferentes a los que se usaban hace 8 ó 10 años atrás.

En la mayoría de las instituciones de educación superior aún se realiza la enseñanza de programación de computadoras con lenguajes de tercera generación, esto les permite a los alumnos familiarizarse con herramientas donde en los programas que desarrolle solo aparece el código del alumno. Sin embargo los lenguajes de programación actuales utilizan nuevos métodos de desarrollo, así como también el propio lenguaje inserta en los programas código de instrucciones muy especializadas que el programador no ha puesto en sus programas, y que requiere familiarizarse con él para los casos de correcciones ó mantenimiento de las aplicaciones.

Todos los sectores del ámbito laboral desarrollan sistemas computacionales con las nuevas herramientas de programación, y demandan de las Instituciones de educación la preparación de profesionales que puedan aportar sus esfuerzos en la construcción de soluciones de negocios utilizando la tecnología computacional de vanguardia. Muchas de las veces las empresas empleadoras de profesionistas en Sistemas Computacionales necesitan capacitar a los recién egresados en las nuevas técnicas y métodos de programación, pero esto representa un tiempo improductivo de los nuevos profesionistas para las compañías.

Por esta razón en el Instituto Tecnológico de Nuevo León se hace necesario una obra como la presente que realice una reflexión y análisis de los lenguajes que actualmente son objetos de la enseñanza y la posibilidad de cambiarlos con la intención de que los alumnos de la carrera de Ingeniería en Sistemas Computacionales aprendan en la misma escuela las nuevas técnicas y lenguajes de programación.

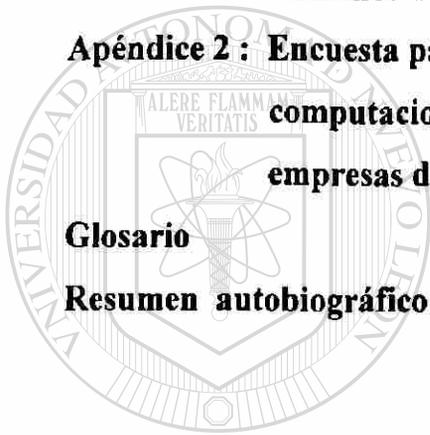
ÍNDICE

Síntesis	1	
1	Introducción	2
	1.1 Descripción del Problema	3
	1.2 Objetivo	4
	1.3 Hipótesis	5
	1.4 Límites del estudio	5
	1.5 Justificación	6
	1.6 Metodología a seguir	7
	1.7 Revisión bibliográfica	8
2	Evolución y clasificación de los Lenguajes de Programación	10
	2.1 Generaciones de las computadoras	10
	2.2 Clasificación de las computadoras	15
	2.3 Clasificación del Software	18
	2.3.1 Software de Sistemas	20
	2.3.2 Software de aplicación	21
	2.4 Evolución y clasificación de los lenguajes de programación	23
3	Características de las generaciones de los Lenguajes de Programación	30
	3.1 Generaciones de los lenguajes de programación	30
	3.1.1 Lenguajes de primera generación	32
	3.1.2 Lenguajes de segunda generación	36
	3.1.3 Lenguajes de tercera generación	38
	3.1.4 Lenguajes de cuarta generación	40
	3.1.5 Lenguajes de quinta generación	42
	3.2 Panorama histórico de los lenguajes de programación	43

4	Importancia de la enseñanza de Técnicas Algorítmicas	48
	4.1 Técnicas Algorítmicas	49
	4.2 Antecedentes Plan reticular ISIC-1993-296	51
	4.3 Información Estadística de Alumnos de Primer Ingreso	54
	4.4 Comparativo del Nivel de Aprovechamiento en programación	57
	4.5 Comparativo de Aprovechamiento entre hombres y mujeres	58
	4.6 Características de Población estudiantil de Primer Ingreso	59
	4.6.1 Edades	60
	4.6.2 Porcentajes de hombres y mujeres	61
	4.6.3 Conocimientos previos en Diseño de Algoritmos	62
	4.6.4 Conocimientos previos en Programación	63
	4.6.5 Alumnos que estudian y trabajan	64
	4.6.6 Tipos de empleos de los alumnos que trabajan	65
	4.6.7 Porqué elegir la profesión de I.S.C.	66
	4.7 Conclusiones de la encuesta a alumnos de Primer Ingreso	67
	4.8 Factores que inciden en el Índice de Reprobación en la Materia de Diseño Estructurado de Algoritmos.	68
	4.9 Propuesta para reducir el índice de reprobación en la materia De Diseño Estructurado de Algoritmos	69
5	El primer Lenguaje de Programación	70
	5.1 Ambientes de Programación	71
	5.2 El Modelo Cliente/Servidor	77
	5.3 Metodologías de la Programación	79
	5.4 Lenguajes Visuales (4GL's) de propósito general mas usados actualmente	82
	5.5 Requerimientos de Recursos y Plataformas de Operación	98
	5.6 Criterios de Selección para el primer Lenguaje de Programación	100
	5.7 Selección del primer lenguaje de programación	105

6	El segundo Lenguaje de Programación y las Bases de datos	109
6.1	Antecedentes de las Bases de datos	110
6.2	Ventajas y desventajas del uso de las Bases de datos	113
6.3	Arquitectura de un DBMS	115
6.4	DBMS para PC's	123
6.5	Perspectivas de las Bases de datos	124
6.6	El Administrador de Bases de datos	128
6.7	El mercado de los SGBD en la actualidad	129
6.8	El segundo lenguaje de Programación	138
7	Diseño y programación de Aplicaciones en Internet	141
7.1	Antecedentes históricos de la red Internet	142
7.2	Herramientas para el diseño y programación en Internet	145
7.2.1	Comentarios generales sobre los lenguajes para Internet	147
7.3	Intranets	155
8	Procesos Administrativos sugeridos a incluir en el plan reticular de los Ingenieros en Sistemas Computacionales	159
8.1	Objetivos de la propuesta	162
8.2	Problemática a resolver	163
8.3	Temas sugeridos	164
9	Conclusiones y recomendaciones	166
9.1	Conclusiones	167
9.2	Recomendaciones	168
9.2.1	Propuesta académica del primer lenguaje de programación	169
9.2.2	Propuesta académica para el segundo lenguaje y Las Bases de datos	174
9.2.3	Recomendación para incluir la enseñanza del Lenguaje Java	175

9.2.4	Recomendación para incluir una materia de especialidad como “lenguaje de programación emergentes”	177
9.2.5	Recomendación para incluir una materia de humanidades como “Comunicación efectiva ISC/usuario”	179
	Bibliografía	181
	Listado de Tablas	182
	Listado de Gráficas	183
	Apéndice 1 : Encuesta sobre identificación de rasgos característicos de los Alumnos de primer ingreso en la carrera de I.S.C.	185
	Apéndice 2 : Encuesta para la identificación de Plataformas de operación computacional y lenguajes de desarrollo mas usados en las empresas del área metropolitana de Monterrey Nuevo León.	189
	Glosario	192
	Resumen autobiográfico	193



UANL

UNIVERSIDAD AUTÓNOMA DE NUEVO LEÓN



DIRECCIÓN GENERAL DE BIBLIOTECAS

SÍNTESIS

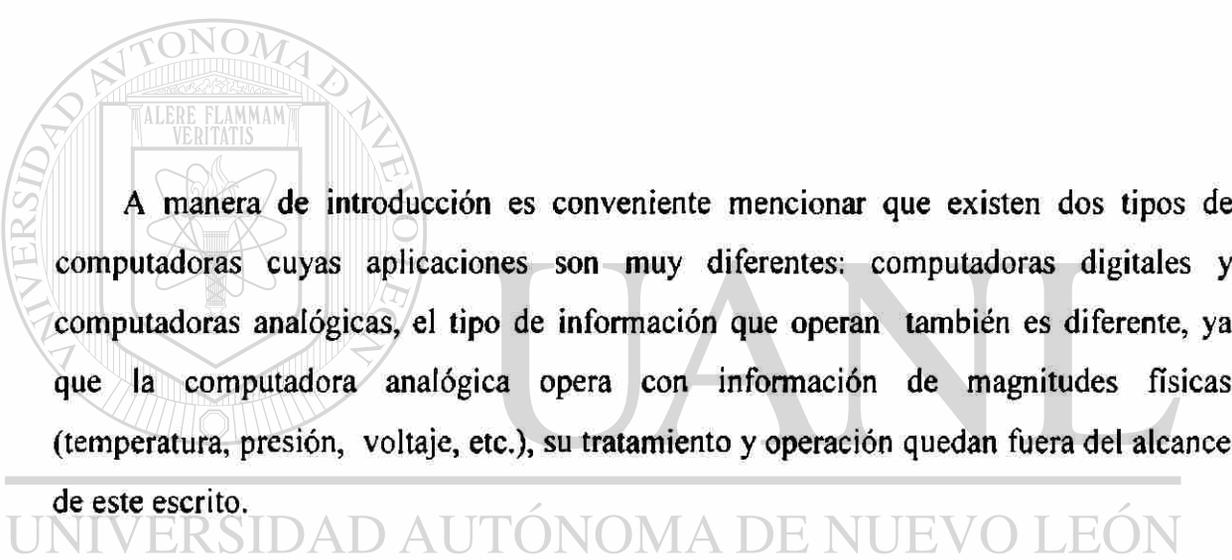
El presente trabajo tiene como objetivo el presentar una alternativa de solución para la actualización de algunas materias contenidas en el plan de estudios de la carrera de Ingeniería en Sistemas Computacionales del Instituto Tecnológico de Nuevo León, y sugerir la inclusión de otras materias nuevas. El énfasis se centra en la línea de la enseñanza de la programación de computadoras, ya que dicha área es la puerta de entrada a la especialidad de la carrera, El presente trabajo incluye los siguientes puntos:

- Mención del marco histórico y actual de los lenguajes y ambientes de desarrollo de programas para computadoras, y la importancia de la enseñanza de métodos algorítmicos para el diseño de soluciones computacionales.
- Analizar los diferentes lenguajes de programación de computadoras existentes en el mercado para seleccionar el mas adecuado en base a ciertos criterios definidos, de forma que los alumnos aprendan a desarrollar aplicaciones con las herramientas, métodos y técnicas de actualidad.
- La determinación de un segundo lenguaje que facilite la construcción y uso de Bases de datos, de forma similar a como se hace en un ambiente laboral.
- Analizar la naciente área de desarrollo de aplicaciones en ambiente Internet/Intranet, los lenguajes mas usados en ese tipo de desarrollos, y sugerir respuesta a esta nueva necesidad.
- Análisis de la situación actual del plan de estudios y la falta de materias que muestren a los alumnos los conceptos y lenguaje de la función de negocios.

En el capítulo de conclusiones y recomendaciones se reflejan las sugerencias de respuesta a los problemas planteados.

1

INTRODUCCIÓN



A manera de introducción es conveniente mencionar que existen dos tipos de computadoras cuyas aplicaciones son muy diferentes: computadoras digitales y computadoras analógicas, el tipo de información que operan también es diferente, ya que la computadora analógica opera con información de magnitudes físicas (temperatura, presión, voltaje, etc.), su tratamiento y operación quedan fuera del alcance de este escrito.

Las computadoras digitales son máquinas que en su funcionamiento usan el sistema numérico binario para representar las instrucciones de los programas y de los datos que se van a procesar. A este respecto la electrónica que se usa para su construcción es la electrónica de flip-flop que facilita la construcción de dispositivos que distinguen la situación entre 2 estados posibles: apagado y encendido (ó “0 y 1”), el presente trabajo se relaciona con la programación de las computadoras digitales.

La programación de computadores digitales es una especialidad profesional muy importante en el área de los Sistemas Computacionales, y quien desea dedicar su tiempo a esta actividad debe obligatoriamente estar actualizado en cuanto a métodos y técnicas modernas de desarrollo para crear aplicaciones ó programas usando la mas moderna tecnología.

1.1 Descripción del problema

La tecnología computacional evoluciona rápidamente y en esa evolución se diseñan nuevos dispositivos, métodos y técnicas para el correcto aprovechamiento de las características de las innovaciones tecnológicas generadas.

La utilización de las innovaciones tecnológicas de parte de los diferentes sectores del ámbito laboral es casi inmediata por la ventaja competitiva que les presenta a sus negocios.

Ante este panorama el Instituto Tecnológico de Nuevo León como todas las Instituciones de educación necesita adecuar constantemente los planes de estudio de las asignaturas de sus carreras profesionales, incorporando la enseñanza de las nuevas técnicas de computación que van surgiendo, con el objetivo de que sus alumnos dispongan de conocimiento actualizado a la par con los requerimientos de los diferentes sectores del ámbito laboral.

DIRECCIÓN GENERAL DE BIBLIOTECAS

De no hacer lo anterior se tendría el peligro que sus alumnos no dispongan del conocimiento de los dispositivos, métodos y técnicas de desarrollo actualizadas y no podrían dar respuesta a las necesidades del sector laboral en la utilización de las mas modernas tecnologías.

1.2 Objetivo.

El objetivo de esta Tesis es presentar una propuesta al Instituto Tecnológico de Nuevo León en cuanto a la enseñanza de la programación de computadoras digitales a los alumnos de la carrera de Ingeniería en Sistemas Computacionales, este aprendizaje les es dado en los primeros semestres de la carrera y representa la inducción básica del alumno en el ambiente propio de la profesión que ha elegido, y su ejercicio será la base para el desarrollo de su vida profesional.

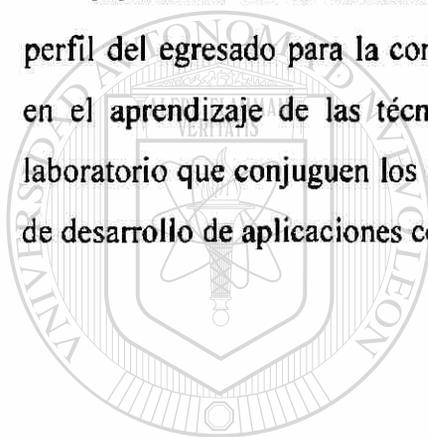
En el transcurso de los últimos 10 años (1990-1999) se han desarrollado y puestos en el mercado de la computación un gran numero de nuevos lenguajes de programación, así como nuevas versiones de lenguajes que ya se usaban, éstos nuevos lenguajes han incorporado nuevas técnicas de desarrollo de programas tales como : programación orientada a objetos, programación orientada a eventos, también a los lenguajes les han implementado herramientas de ambiente visual para el Desarrollo Rápido de Aplicaciones (RAD).

Actualmente en el Instituto tecnológico de Nuevo León, en las materias cuya función es la de la enseñanza de la programación de computadoras, se enseña el Lenguaje C++, en su versión como lenguaje de tercera generación.

1.3 Hipótesis.

Ante la evolución de los lenguajes de programación, y el surgimiento de nuevos métodos y técnicas para la gestión de la información, se pretende demostrar la urgente necesidad de modificar el plan de estudios de la carrera de Ingeniería en Sistemas Computacionales, para adecuarla a los nuevos tiempos de retos y competencia.

Asimismo de la necesidad de incluir asignaturas ó temas que complementen el perfil del egresado para la competencia administrativa, esto además puede facilitar que en el aprendizaje de las técnicas de programación se realicen talleres ó trabajos de laboratorio que conjuguen los métodos administrativos y de producción con las técnicas de desarrollo de aplicaciones computacionales.



UANL

UNIVERSIDAD AUTÓNOMA DE NUEVO LEÓN

1.4 Límites del estudio.

DIRECCIÓN GENERAL DE BIBLIOTECAS



Las fronteras del presente estudio serán el marco de referencia de las metas y objetivos del Instituto tecnológico de Nuevo León.

1.5 Justificación.

El ámbito de mercados globales al cual ha ingresado nuestro país, ha generado que el sector laboral demande profesionistas en el área de Sistemas Computacionales capacitados para desarrollarse en estas nuevas situaciones, donde se requiere que el nuevo profesionista tenga:

- Conocimiento de las funciones básicas de los negocios.
- Conocimientos actualizados de técnicas del desarrollo de aplicaciones usando la mas moderna tecnología computacional.
- Valores humanos tales como la honestidad, responsabilidad, respeto, trato amable, etc...
- Dominio de un segundo idioma.

Las características mencionadas son referidas como rasgos del perfil de los nuevos profesionistas, y han sido expresadas por expertos conocedores de estos temas, tanto en foros y conferencias como también en la prensa escrita. En el presente trabajo de Tesis solo se tiene alcance respecto a los dos primeros puntos.

1.6 Metodología a seguir.

La metodología a utilizar será la recolección de los datos para la sustentación de la Tesis, y se define mediante los siguientes puntos:

- Análisis de herramientas y ambientes de desarrollo usados actualmente en los diferentes sectores del ámbito laboral de la localidad.
- Análisis de las características técnicas de los lenguajes de programación y sus requerimientos de operación.
- Análisis de plataformas de operación.
- Selección de lenguajes Tipo
- Definición de la secuencia de los temas para la enseñanza de los lenguajes de programación y las materias donde se incluirán.
- Definición de los procesos administrativos básicos a incluir en la enseñanza, para la capacitación de los alumnos en ellos.
- Capacitación de la planta docente de la especialidad de Sistemas Computacionales.

1.7 Revisión Bibliográfica.

Para los temas que refieren los antecedentes cronológicos de las generaciones de la computadora se consideró el texto **Elementos de Computación** de Guillermo Levine, de la Editorial Mc Graw Hill.

En lo referente a la Clasificación de los Lenguajes de Programación se consideraron tres textos a saber: **Curso de Programación** de cuyos autores son Jorge Castro, Felipe Cucker, Xavier Messenguer, Albert Rubio y Luis Solano de la editorial Mc Graw Hill, **Lenguajes de Programación** de Allen B. Tucker también de la editorial Mc Graw Hill, **Intelligent Software Systems Development an IS Manager's Guide** de Paul Harmon & Curtis Hall de la editorial Jhon Wiley & Sons.

Del tema Panorama histórico de los Lenguajes de Programación se consultó al libro **Programación Orientada a Objetos, Conceptos, Modelado, Diseño y Codificación en C++** del autor Luis Joyanes Aguilar, asimismo al texto **Intelligent Software Systems Development an IS Manager's Guide**, y el fascículo No. 18 de la serie **Curso IBM de Programación**.

DIRECCIÓN GENERAL DE BIBLIOTECAS

Para el tema del Primer lenguaje de Programación en lo referente a las características de los nuevos lenguajes de programación así como sus métodos y técnicas de programación, se consultaron los **fascículos 1 y 2** de la serie **Curso IBM de Programación**, así como algunas páginas de Internet por ejemplo :

<http://members.es.tripod.de/adm/popup/roadmap-search.html>

<http://www.recurso-as400.com>

<http://www.audisoft.com>

<http://www.programacion.net/cursos/vbeliserv>

<http://www.concytec.gob.pe/institucional/servidor.htm>

<http://www.angel.perez.net/MODINET/desarr.htm>

<http://www.terabyte.cl/cobolwow.htm>

2

Evolución y clasificación de los lenguajes de programación.

2.1 Generaciones de la Computadora.

Históricamente las computadoras digitales han sufrido una evolución que puede relacionarse en términos de “generaciones” de las cuales algunos autores de textos las definen en tres y media (ó casi cuatro), a saber:

Primera generación : Aparece a fines de la década de 1940 y termina a inicios de 1960 la característica que la distingue es su tecnología, ya que fueron construidas a base de bulbos ó tubos de vacío, debido a ello ocupan grandes espacios de los edificios, y su programación representaba una gran dificultad ya que solo se podían programar en lenguaje de máquina. Algunos de los modelos de esta primera generación son:

ENIAC. (1947) Fue la primera computadora digital en la historia, fué un prototipo experimental , no era programable, fue construida en el sótano de la universidad de Pennsylvania de los Estados Unidos de America por los doctores Jhon W. Mauchly y J. Presper Eckert.

EDVAC. (1949) También fue solo un prototipo de laboratorio pero incluía el concepto de “Programa almacenado” tal como las computadoras actuales, el concepto fue acuñado por el doctor Jhonn Von Neumann.

UNIVAC I. (1951) Desarrollada por los doctores Mauchly y Eckert, que fundaron la empresa UNIVAC, fue la primera computadora comercializada, el primer cliente en adquirir este modelo fue la oficina del censo de los Estados Unidos de América.

IBM 701.(1953) La compañía IBM fundada por Thomas Watson, se dedicaba a comercializar equipo de registro unitario, el cual usaba tarjetas perforadas como el medio de alimentación y almacenamiento de datos, y su reconocimiento se conseguía mediante la interpretación magnética del código desarrollado por Herman Hollerit en 1890. Fue el primer modelo comercializado por esta compañía . que llegaría a una posición dominante en años posteriores.

1954 – 1960. Fue un periodo donde IBM presentó varios modelos que utilizaban un medio de almacenamiento llamado “Tambor magnético” con lo que marcó el liderazgo en el ámbito computacional, posteriormente implementó el uso de discos magnéticos llamados disk pack. En este tiempo también aparece la compañía Remington Rand y la UNIVAC continuó presentando nuevos modelos.

Segunda generación : (1960-1964) A principios de la década de 1960 la industria japonesa desarrolla aplicaciones del transistor en la electrónica, y su

primera aplicación es en la producción de radios de transistores, los fabricantes de computadoras no tardan en darse cuenta que el transistor puede también emplearse con las computadoras.

Asimismo también aparecen los primeros lenguajes de “alto nivel” y con ello el “oficio” de programador de computadoras.

De esta manera las características de las computadoras de la segunda generación son las siguientes:

- Construidas en base a electrónica de transistores.
- Inicia la programación simbólica en base a lenguajes de alto nivel(Cobol, Fortran).
- Aparecen nuevas compañías fabricantes de computadoras

Algunos de los nuevos modelos que podemos citar son:

CDC 3000. Aparece la compañía CDC(Control data Company), el modelo se caracterizaba por ser muy una computadora grande, potente y veloz en sus operaciones.

BURROUGHS 5000. Nace la compañía Burroughs, y su serie 5000 es considerada la mas avanzada de la época.

UNIVAC siguió ofreciendo nuevos modelos de computadoras.

IBM. siguió ofreciendo nuevos modelos mejorados cada vez .

Tercera generación : (1964-1970) Esta generación fue marcada por la empresa IBM al presentar en el año de 1964 un nuevo concepto de atención al mercado, anunciando la llamada “Familia 360” que ofrecía la herramienta para la solución de problemas de negocios ya fueran pequeños, medianos ó grandes, pues ofrecía modelos al tamaño del cliente. Asimismo la tecnología de transistores evolucionó a la tecnología de circuitos integrados mediante la miniaturización de los transistores, logrando que las computadoras disminuyeran significativamente su tamaño, pero aún seguían ocupando grandes áreas que debían estar acondicionadas especialmente en cuanto a humedad, aire acondicionado, grandes unidades de respaldo de energía, y equipo de regulación de voltaje. Respecto del software se tiene también un avance ya que aparecen los primeros sistemas operativos que permiten una comunicación entre la computadora y el programador.

Sus características que la distinguen son las siguientes:

- Construidas en base a electrónica de circuitos integrados
- Aparecen los primeros Sistemas Operativos

Las compañías fabricantes continúan generando nuevos modelos:

IBM. Ofrece su “Familia” de computadoras “Serie 360”, a inicios de la década de 1970 evoluciona a Familia de la “Serie 370”, y también ofrece mejoras con un nuevo Sistema Operativo.

CDC Ofrece el modelo CDC serie 6000, destacándose como el fabricante de las computadoras mas grandes y veloces, a inicios de la década de 1970 ofrece el modelo CDC serie 7000, que después fueron substituidos por la serie CDC Cyber.

BURROUGHS Ofrece el modelo Burroughs serie 7000, y un Sistema Operativo muy robusto llamado MCP.

UNIVAC Ofrece la Familia I100.



A principios de 1970 aparecieron computadoras que ya no eran grandes aparatos ocupando grandes áreas especialmente acondicionadas, sino que eran computadoras de mediano tamaño que podían ser instaladas en lugares reducidos y sin acondicionamiento especial. A estas computadoras se les llamó “Minicomputadoras” y fueron desarrolladas por compañías que recién ingresaban al mercado tales como la Digital, Hewlett Packard, Data General, Wang, y otras mas. Un ejemplo de ello fueron:

DIGITAL Su primer minicomputadora de éxito comercial fue la PDP-8. La compañía fue llamada Digital Equipment Corporation quien después presentó otro modelo denominado PDP-11, siguiéndole después con un nuevo modelo mejorado de gran aceptación, el modelo VAX.

HEWLETT PACKARD empresa que entró al mercado de las minicomputadoras con el modelo de la serie HP-3000, posteriormente introdujo un nuevo mode-

lo de la serie HP-9000.

IBM Reforzó su presencia en el mercado de las mini-computadoras con los modelos IBM-34, IBM-36, IBM-38.

Cuasi Cuarta generación. Esta generación fue marcada con la aparición de las microcomputadoras que utilizando tecnología VLSI, integración de circuitos electrónicos a gran escala revolucionó grandemente el mercado de las computadoras, encontrándole aplicaciones en todos los ámbitos: en el hogar, personal, y de negocios .

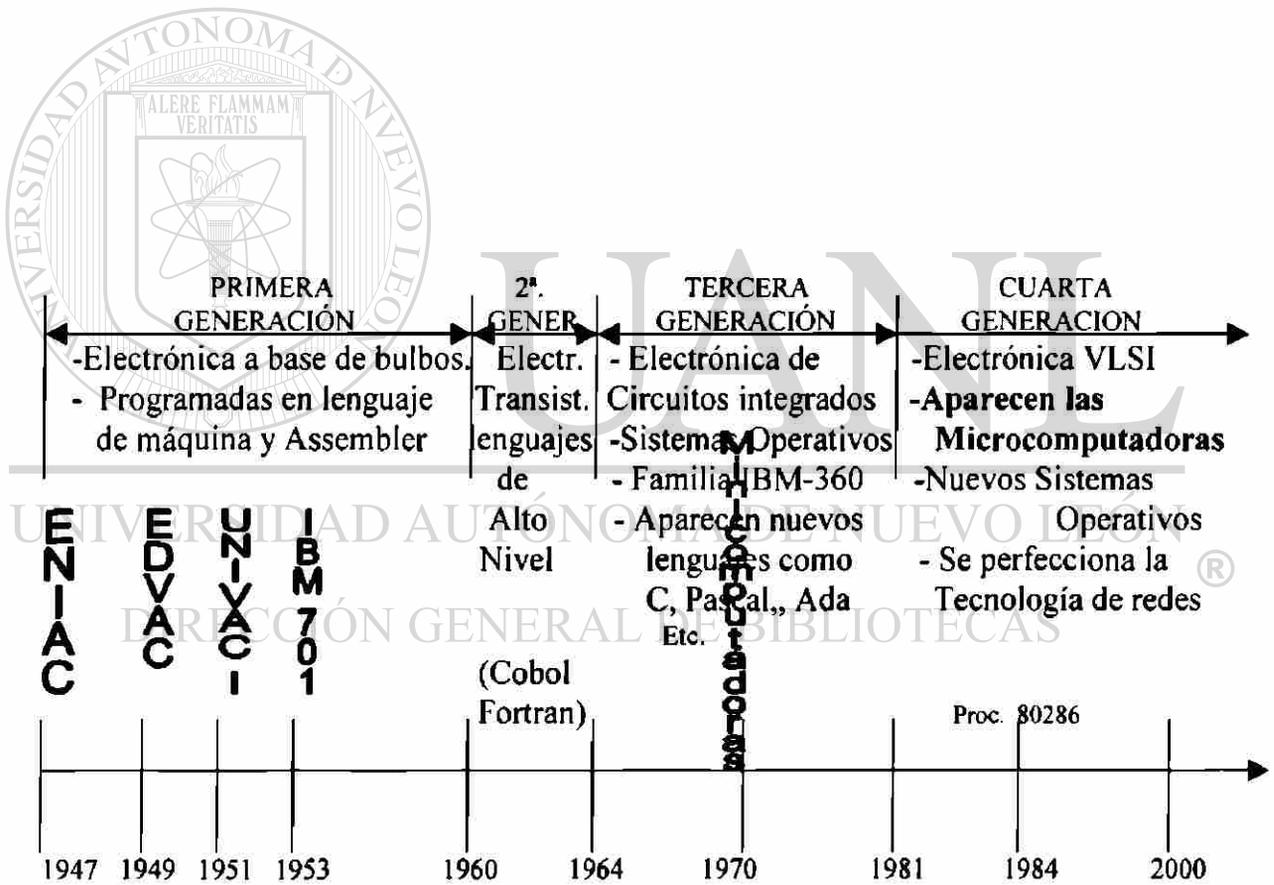


Fig. 1 Cronograma de las generaciones de las computadoras

2.2 Clasificación de las Computadoras.

De acuerdo a su tamaño las computadoras se clasifican en:

- Supercomputadoras
- Macrocomputadoras
- Minicomputadoras
- Microcomputadoras(ó PC's)

Supercomputadoras

La Supercomputadora es el tipo de computadora mas potente y mas rápida en la ejecución de operaciones que puede existir en un tiempo dado, son equipos de gran tamaño y ocupan espacios grandes. Estas máquinas son diseñadas para procesar grandes cantidades de información en poco tiempo, y son dedicadas a tareas específicas. Asimismo son el tipo de computadoras mas caras ya que son valuadas a precios de millones de dólares, una de sus características es que deben estar instaladas en lugares debidamente acondicionados para control especial de temperaturas, para disipar el calor que se produce en algunos de sus componentes. Algunas de sus aplicaciones son:

- 1 Búsqueda y estudio de energías y armas nucleares.
- 2 Búsqueda de yacimientos petrolíferos con grandes bases de datos.
- 3 Estudio y predicción de tornados.
- 4 Estudio y predicción de climas de cualquier parte del mundo.
- 5 Elaboración de maquetas y proyectos de creación de aviones, simuladores de vuelo, etc..

debido a que son computadoras muy caras se construyen pocas de ellas al año.

Macrocomputadoras.

Estas computadoras también son llamadas “Mainframe”, y también como las supercomputadoras ocupan grandes espacios, realizan operaciones ejecutables también a grandes velocidades, su diseño va dirigido a tener conexión y controlar a cientos de usuarios simultáneamente, así como con cientos de dispositivos de entrada y salida de datos. Los Mainframes son computadoras algunas veces mas caras que las supercomputadoras, y algunas veces también con mayor capacidad de servicio, pero las supercomputadoras siempre son mas rápidas en cuanto a ejecución de programas.

En el pasado los Mainframes ocupaban salas de computo completas y algunas veces hasta pisos completos de edificios, pero actualmente los Mainframes son parecidos a una fila de archiveros en un cuarto con piso falso, ya que esto así se diseña para ocultar bajo piso los cientos de cables de conexión a periféricos. La temperatura del Site debe ser controlada.

Minicomputadoras.

En 1970 aparece la Minicomputadora, una versión mas pequeña de la Macrocomputadora, desde su aparición fue dirigida a tareas específicas y solo se conectaba solo a los periféricos mas elementales, lo cual se vió reflejado en reducción de precio y costos de mantenimiento. En cuanto a su capacidad de cómputo las Minicomputadoras se encuentran entre los Mainframe y las estaciones de trabajo.

En general una Minicomputadora es un sistema de multiproceso (con capacidad de ejecutar procesos paralelos), puede soportar de 10 hasta 200 usuarios simultáneamente, actualmente se usan para almacenar grandes bases de datos, en la automatización industrial y aplicaciones multiusuario.

Microcomputadoras.

Las Microcomputadoras tuvieron su origen con el surgimiento de los microprocesadores, un microprocesador es “una computadora en un chip”, ó sea en un circuito integrado independiente. Las Minicomputadoras ó PC’s son computadoras de uso personal y relativamente baratas, actualmente son muy usadas en oficinas, escuelas y los hogares.

El término PC fue originado en el año de 1981 cuando IBM sacó al mercado su modelo “IBM PC”, el cual se convirtió en un tipo de computadora ideal para uso “personal”, de ahí ese término cobró popularidad. Posteriormente otras empresas entraron al mercado de las PC’s usando procesadores del mismo tipo que las de IBM y las llamaron “PC compatibles” con IBM. Existe otro tipo de microcomputadoras que no son compatibles con IBM, la cual es la marca Macintosh pero que también se les llama “PC” por ser de uso personal.

UNIVERSIDAD AUTÓNOMA DE NUEVO LEÓN

®

DIRECCIÓN GENERAL DE BIBLIOTECAS

2.3 Clasificación del software.

De todos los inmersos en el ámbito de los Sistemas Computacionales (también llamado ámbito de la Informática, y de la Computación), es sabido que una computadora digital se concibe en forma general bajo dos conceptos a saber: Hardware y Software.

El hardware se refiere a todos los componentes físicos que forman la máquina que llamamos computadora (Gabinete, Monitor, Tarjetas de circuitos impresos, cables, etc.), en suma son “todas aquellas partes que podemos ver y tocar”.

El software se refiere a los elementos intangibles ó sea aquellas partes que no podemos tocar (los programas), pero que al ser “integradas” al hardware se logra que la maquinaria inerte “adquiera vida artificial” y pueda realizar funciones de procesamiento de datos. Metafóricamente el software para el hardware es como el combustible para el automóvil, un elemento que no es parte integral de él, pero que le es indispensable para su funcionamiento.

El software ha sido un pilar para la utilización efectiva del hardware, y los programadores son los artífices “creadores”, sus herramientas principales son :

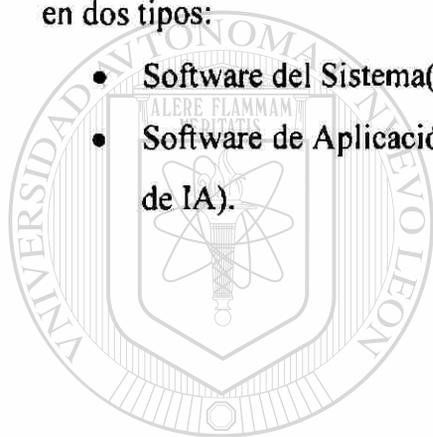
- Los lenguajes de programación (creados por ellos mismos), donde se definen los métodos y técnicas de la programación.
- Su propia imaginación.

para lograr el desarrollo de aplicaciones que sean soluciones a los diferentes tipos de problemas, a través de darle “órdenes” a la computadora para que ejecute las tareas deseadas.

Al igual que el hardware el software ha evolucionado a través del tiempo, y por él se ha logrado que el ser humano “se comunique” con la computadora y le pueda indicar que realice actividades ó tareas definiéndole claramente qué hacer, como hacerlo y cuando hacerlo.

Por medio de los lenguajes de programación se hace uso de las características del hardware y mediante el acceso a la memoria de la máquina se deja (almacena) en ella los datos con los que trabajará y las instrucciones que le indiquen el procedimiento a seguir para la realización de la tarea deseada. En forma general el software se clasifica en dos tipos:

- Software del Sistema(de sistemas, de tiempo real).
- Software de Aplicación(de gestión, de ingeniería y científico, empotrado, de PC, de IA).



UANL

UNIVERSIDAD AUTÓNOMA DE NUEVO LEÓN



DIRECCIÓN GENERAL DE BIBLIOTECAS

2.3.1 Software de Sistemas.

El software de sistemas es un conjunto de programas diseñados para facilitar la operación de otros programas (compiladores, editores, utilerías de gestión de archivos, etc.), los cuales tienen capacidad de operar con estructuras de información.

Otras aplicaciones tales como Sistemas Operativos, utilerías para la operación de periféricos y de procesadores de telecomunicaciones permiten procesar grandes volúmenes de datos.

El software de sistemas se caracteriza por operar en gran manera con el hardware de la computadora, permitir el acceso a múltiples usuarios, operación de procesos concurrentes, planificación de uso y comparación de recursos, óptima planeación de procesos, y conexión a múltiples interfaces externas.

Como software de tiempo real es conocido aquel software que permite medir, analizar y controlar sucesos del mundo real conforme estos ocurren, los elementos de software de tiempo real incluyen un componente de acumulación de datos que recolecta y formatea la información en forma externa, también un componente de análisis que transforma la información según lo requiere la aplicación. El software de tiempo real también tiene un componente de monitorización que coordina a todos los demás componentes, de forma que pueda mantenerse la respuesta en tiempo real (generalmente en el rango de 1 milisegundo a un minuto).

Es conveniente precisar que el término “tiempo real” es diferente del término “tiempo compartido” ya que el un sistema de tiempo compartido puede sobrepasarse el tiempo sin que se ocasione un contratiempo.

2.3.2 Software de Aplicación.

Este tipo de software implementa los procedimientos requeridos para realizar las funciones del procesamiento de la información.

Software de gestión.

El procesamiento de información comercial constituye la mayor de las áreas de aplicación del software, los sistemas administrativos(Nóminas, Contabilidad, Control de inventarios, etc..) son considerados dentro del software de gestión, ya que opera con una ó mas Bases de datos donde se almacenan los datos de la información comercial.

Las aplicaciones de esta área reestructuran los datos existentes para facilitar las operaciones comerciales y la toma de decisiones. Además de las tareas convencionales de procesamiento de datos también realizan operaciones interactivas, por ejemplo transacciones en puntos de venta.

Software de ingeniería y científico.

El software de ingeniería y científico se caracteriza por los algoritmos de “Métodos Numéricos”, estas aplicaciones van desde la astronomía a la vulcanología, desde el análisis de la presión de automotores a la dinámica orbital de las lanzaderas espaciales y desde la biología molecular a la robótica. Sin embargo las nuevas aplicaciones de esta área se han alejado de los métodos convencionales. El diseño asistido por computadora(CAD), la simulación de sistemas y otras aplicaciones interactivas han comenzado a tomar características de software de tiempo real.

Software empotrado.

Este software es definido de esta manera porque reside en “Memoria de solo lectura(ROM)”, y se utiliza para controlar productos y sistemas de los mercados industriales y de consumidores. Este software puede ejecutar funciones muy limitadas(por ejemplo el control de las teclas de un horno de microondas) para facilitar una función significativa y capacidad de control, tales como: control digital de la gasolina de un automotor, sistemas de frenado, etc..

Software de Computadoras Personales (PC).

Este software ha desarrollado grandemente su potencial desde la década pasada, y algunas de sus aplicaciones son: Procesamiento de textos, hojas de cálculo, gráficos por computadora, entretenimiento, gestión de bases de datos, aplicaciones financieras, comerciales y personales, de redes externas y acceso a bases de datos. Este tipo de software representa una de las áreas de mayor innovación.

Software de IA(Inteligencia Artificial).

Este software hace uso de algoritmos heurísticos(no numéricos), y se usa para el diseño de soluciones a problemas complejos que no se pueden resolver directamente con cálculo numérico ó análisis directo. Esta área es también llamada “de los Sistemas Expertos” ó “Sistemas basados en conocimiento”.

2.4 Evolución y clasificación de los lenguajes de programación.

Para la solución de problemas el hombre ha diseñado métodos de acuerdo a las herramientas de que dispone, con la aparición de la computadora y de su adopción como herramienta de solución a cierto tipo de problemas, el hombre ha diseñado también métodos acordes al uso de esta herramienta, y los ha adaptado a la codificación de los lenguajes de programación.

A muchos problemas que son tratados mediante la computadora el medio externo les proporciona cierto dinamismo, y en esos casos los métodos también necesitan modificarse como adecuación a los medios externos, de la misma forma los lenguajes de programación han evolucionado para adecuarse a las nuevas circunstancias.

Los métodos y lenguajes de programación han evolucionado rápidamente desde los primeros lenguajes de alto nivel que aparecieron en la década de 1950. En seguida se enuncian los factores más importantes que han influido en esta evolución.

El hardware y los Sistemas Operativos .

La evolución en sus diseños ó en su arquitectura de los computadores, que han significado un aumento en su velocidad de procesamiento así como aumentando su capacidad de almacenamiento, además del surgimiento y mejoras de los Sistemas Operativos, han influido grandemente en la generación de nuevas versiones de los lenguajes de programación así como el surgimiento de nuevos lenguajes.

Necesidad de nuevas aplicaciones.

Debido a los bajos costos de productos computacionales se ha incrementado la necesidad de nuevas aplicaciones en poco tiempo. Los requisitos ó especificaciones de las nuevas necesidades han sido tomados en cuenta para el diseño de los nuevos lenguajes de programación, y revisión de los ya existentes.

Metodologías de programación.

Las consideraciones respecto a características de los problemas donde las metodologías actuales de programación no han cubierto, respecto a escribir programas complejos se han visto reflejadas en el diseño de los nuevos lenguajes de programación.

Métodos de implementación.

El desarrollo de mejores métodos para la implementación de lenguajes ha permitido el diseño de mejoras al proceso de la implementación. Esto ha reducido notablemente los costos.

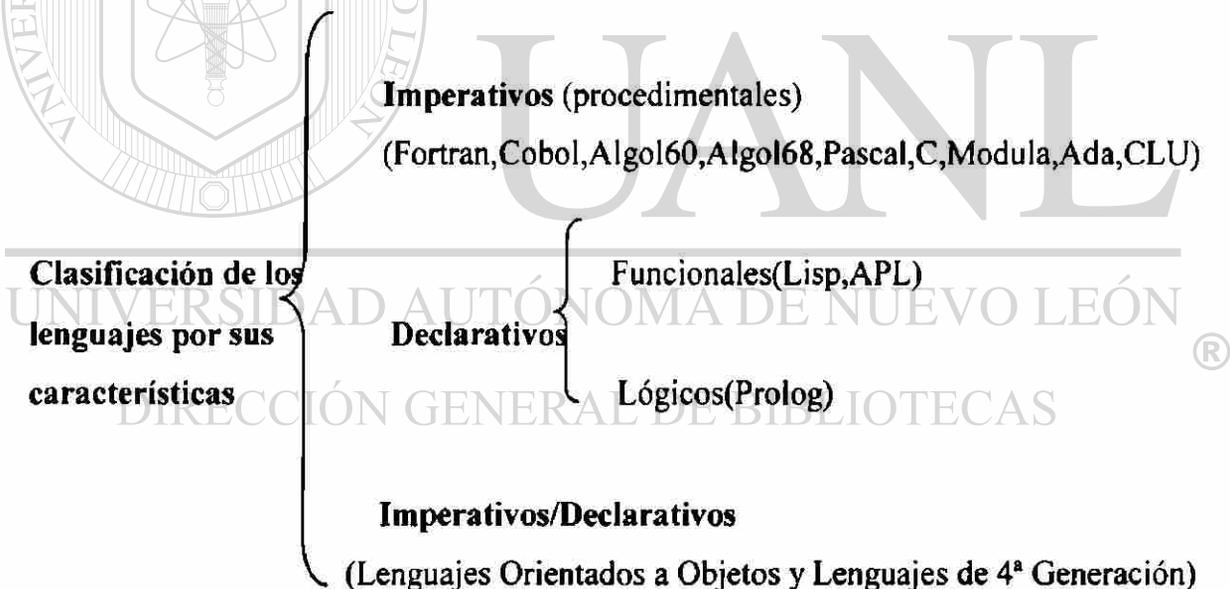
Estudios teóricos.

La investigación en áreas de la computación ha generado mejoras en el diseño de nuevos métodos de programación, asimismo de una visión clara de las debilidades de los métodos actuales. Esto ha abierto nuevos campos de trabajo en áreas de laboratorio probando aplicaciones de las nuevas técnicas de desarrollo, lo cual ha provocado la aparición de nuevos lenguajes de programación con nuevas metodologías de programación.

Estandarización.

Este factor de procurar estandarizar métodos mediante las consideraciones de las organizaciones internacionales ha influido grandemente, tratando que la transportabilidad de los programas entre diferentes computadores sea una realidad, ha sido un aspecto importante en la evolución de los lenguajes de programación.

A ciencia cierta existe un gran numero de lenguajes de programación para las computadoras, y algunos de los conocedores los agrupan bajo características de funcionamiento, aunque algunos lenguajes no encajan directamente en algún grupo en particular. Otros de los conocedores los clasifican por generaciones en similitud de la clasificación del hardware.



Los lenguajes imperativos son aquellos que usan principalmente la instrucción de asignación como el constructor básico y lo complementan las estructuras de control secuencial, estructuras alternativas e iterativas, su diseño se basa en la construcción de algoritmos de solución que son “seguidos” como una receta.

Una crítica generalizada a estos lenguajes es que idealmente el programador debe concentrarse en describir aquello que desea controlar en el programa sin preocuparse en como administrar el proceso de ejecución del programa, sobre todo con respecto al control de la utilización de la memoria.

Los lenguajes declarativos también son considerados declarativos/descriptivos e incluyen significados alternativos que describen valores de los datos y los cálculos. Los lenguajes declarativos tienen tres características las cuales son: **Expresivos** por la manera que describen los problemas, situaciones, métodos ó soluciones. **Fiables** ya que tratan de proteger al programador tanto como sea posible en cuanto a cometer errores. **Elegantes** ya que son dirigidos al uso de un soporte matemático en el desarrollo de las aplicaciones.

Los lenguajes declarativos se clasifican en dos clases: Funcionales y lógicos.

Lenguajes Funcionales. Estos lenguajes son aquellos en que todas las aplicaciones son funciones en el sentido matemático, ó sea que no hay instrucciones, de forma que un programa funcional es una función que se define por descomposición de funciones mas simples. Estos lenguajes no usan la instrucción de asignación, la inexistencia de esta instrucción permite que el programador no se ocupe en el control de la memoria que está usando, esto le permite al programador concentrar sus esfuerzos en la descripción del problema.

Mientras que para los lenguajes orientados a instrucciones la forma de construir los programas es en base a organizar la ejecución en forma secuencial, así como de la de ejecución repetida de instrucciones(ciclos), todo ello a través del uso de estructuras de control adecuadas, en vez de ello la programación funcional hace combinación de funciones para generar otras mas potentes.

Otro punto mas respecto a los lenguajes funcionales es que tratan las funciones de la misma que cualquier otro valor, es decir pueden ser pasadas como parámetros, pueden ser el valor de una expresión ó pueden ser almacenadas en una estructura de datos.

Algunas veces el uso del término “funcional” es aplicado solamente para definir aquellos lenguajes que no usan en absoluto las características procedimentales, a éstos lenguajes se les considera “Lenguajes Funcionales puros” tales como el lenguaje Standard, el lenguaje ML, el lenguaje APL. Pero existen lenguajes funcionales que “si” usan la instrucción de asignación tales como el lenguaje LISP.

Lenguajes lógicos. La idea básica de la programación lógica se traduce a la expresión de la frase: “Algoritmos = Lógica + Control”, esta frase muestra la forma de operar con estos lenguajes, y es opuesta a la forma de operar en los lenguajes tradicionales. El lenguaje lógico mas conocido es **Prolog**. Los lenguajes lógicos tratan con relaciones (predicados) entre objetos(datos), en lugar de hacerlo con funciones. Este principio se basa en la premisa que programar con relaciones permite mas flexibilidad en los programas que programar con funciones, ya que las relaciones tratan con argumentos y otorgan resultados uniformemente. Informalmente el uso de las relaciones permite ignorar direcciones de tal forma que no existe distinción entre los datos que se calculan y a partir de cuales.

Las relaciones se especifican con “reglas” y con “hechos”, un hecho sería una “frase” como : “Rosa es mujer” en Prolog sería `mujer(rosa)`

“Los Padres de Rosa son Carlos y Pilar” en Prolog sería `padres(Rosa, Carlos, Pilar)`

una “regla” se usa para expresar frases condicionales como:

“Dos personas son hermanas si son mujeres y tienen los mismos padres”

en Prolog sería : `hermanas(x,y):- mujer(x), mujer(y), padres(x,p,m), padres(y,p,m).`

Por tanto las reglas no sirven para deducir relaciones entre objetos que no se han definido explícitamente mediante hechos.

Imperativos / declarativos. Bajo esta clase se considera a los lenguajes orientados a objetos y también a los considerados lenguajes de cuarta generación. Los lenguajes orientados a objetos proponen una nueva metodología de programación caracterizado por una nueva concepción de los problemas a resolver mediante programas de

computadoras, esta nueva manera de programar se basa en tres conceptos nuevos: “clase”, “objeto” y “herencia”.

Una clase representa a un tipo de datos y es considerada una “clase de objetos”, los objetos son entes dinámicos ya que pueden ser creados y borrados al momento de la ejecución del programa, los objetos de una misma clase mantienen características y procesos similares.

La herencia permite definir nuevas “clases de objetos” a partir de una clase previamente definida, la nueva clase heredará las características y propiedades de la clase previa.

Además de los conceptos mencionados(objetos,clases,herencia), se utilizan otros dos conceptos que son: “mensaje” y “método”. El “mensaje” es usado para comunicar a un objeto la tarea que se espera que realice, es parecido a un procedimiento donde se especifica el objeto a quien va dirigido, el nombre de la operación a realizar, y los parámetros necesarios para realizarla.

El lenguaje orientado a objetos mas conocido es el Smalltalk el cual fue dado a conocer en 1976, a éste se ha añadido el lenguaje C++ a partir de 1986. Debido a que C++ es una extensión del lenguaje C los mensajes y métodos son directamente los usados para operación de funciones usuales en C, éste lenguaje paulatinamente ha incrementado su popularidad ya que su metodología es adecuada para tratar problemas de gran complejidad, así como el uso de la nueva tecnología emergente.

Los **lenguajes de cuarta generación** son aquellos que pretenden superar los problemas surgidos en los lenguajes de tercera generación tales como:

- Reducir el tiempo de construcción de las aplicaciones.
- Minimizar los problemas en la depuración de aplicaciones.
- Facilitar a los usuarios finales la forma de resolver sus propios problemas, usando la misma herramienta de programación.

- Generación automática de código ejecutable sin errores, a partir de definir los requerimientos deseados, y dándolos mediante expresiones de alto nivel.
- Reducir los costos de mantenimiento haciendo aplicaciones fáciles de hacer cambios.

Generalmente los lenguajes de cuarta generación están orientados al tratamiento de grandes cantidades de información, mediante el uso de grandes Bases de datos.

Algunos de los lenguajes de esta clase son considerados como de propósito general tales como: IDEAL, MANTIS, NATURAL, otros lenguajes de esta misma clase pero considerados como especialmente al tratamiento de datos son: SQL, QBE ó ADRS.

Otro punto de vista en la clasificación es considerando que los lenguajes de programación han sufrido transformaciones que se engloban bajo el concepto de generaciones.

- Lenguajes de primera generación.
- Lenguajes de segunda generación
- Lenguajes de tercera generación
- Lenguajes de cuarta generación
- Lenguajes de quinta generación

Las cuales se comentarán en el siguiente capítulo.

3

Características de las generaciones de los lenguajes de programación

3.1 Generaciones de los lenguajes de programación.

De todos los inmersos en el ámbito de los Sistemas Computacionales (también llamado ámbito de la Informática, y de la Computación), es sabido que una computadora digital se concibe en forma general bajo dos conceptos a saber: Hardware y Software.

El hardware se refiere a todos los componentes físicos que forman la máquina que llamamos computadora (Gabinete, Monitor, Tarjetas de circuitos impresos, cables, etc.), en suma son “todas aquellas partes que podemos ver y tocar”.

El software se refiere a los elementos intangibles ó sea aquellas partes que no podemos tocar (los programas), pero que al ser “integradas” al hardware se logra que la maquinaria inerte “adquiera vida artificial” y pueda realizar funciones de procesamiento de datos. Metafóricamente el software para el hardware es como el combustible para el automóvil, un elemento que no es parte integral de él, pero que le es indispensable para su funcionamiento.

El software ha sido un pilar para la utilización efectiva del hardware, y los programadores son los artífices “creadores”, sus herramientas principales son :

- Los lenguajes de programación(creados por ellos mismos), donde se definen los métodos y técnicas de la programación.
- Su propia imaginación.

mediante los cuales logran el desarrollo de aplicaciones que sean soluciones a los diferentes tipos de problemas, a través de darle “órdenes” a la computadora para que ejecute las tareas deseadas.

Al igual que el hardware el software ha evolucionado a través del tiempo, y a través de él se ha logrado que el ser humano “se comunique” con la computadora y le pueda indicar que realice actividades ó tareas definiéndole claramente qué hacer, como hacerlo y cuando hacerlo.

Por medio del software(Lenguajes de programación) se hace uso de las características del hardware y mediante el acceso a la memoria de la máquina se deja (almacena) en ella los datos con los que trabajará y las instrucciones que le indiquen el procedimiento a seguir para la realización de la tarea deseada.

3.1.1 Lenguajes de primera generación.

En esta generación se considera a los **lenguajes de máquina** y los **lenguajes ensambladores**. Cada arquitectura de computadora tiene un solo lenguaje de máquina que “entiende”, por lo que existen lenguajes de máquina tantos como arquitecturas de computadoras existan. Los **lenguajes de máquina** tienen un conjunto reducido de símbolos numéricos mediante los cuales se indican las acciones a realizar y los datos que intervienen en las operaciones. Ya que los símbolos que utiliza son números sus instrucciones son una secuencia de números, que para otras personas fuera del programador (de lenguaje de máquina) son “algo in entendible”. El programar en este lenguaje se tienen probabilidades de un alto grado de error.

Ejemplo.

```
41 3 0C1A4
3A 2 0C1A8
1A 3 0C1A0
50 3 0C1A4
```

esta secuencia representa la instrucción:

$$z = w + x + y$$

un programa en lenguaje de máquina, es el medio mediante el cual el programador “directamente” le indica a la computadora las acciones de la tarea a realizar.

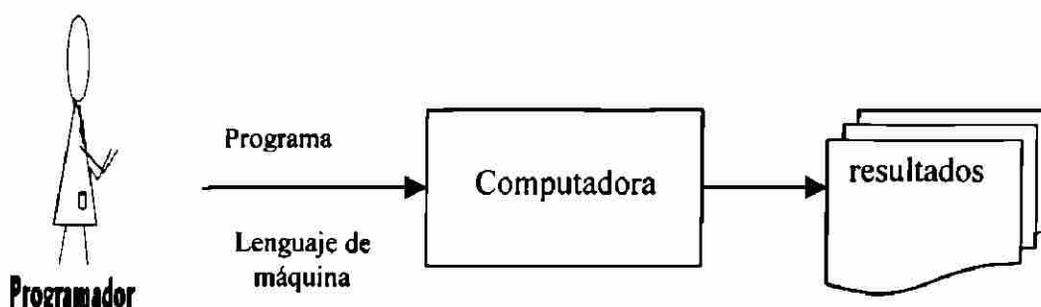


Fig. 2 Esquema del Proceso de la programación en lenguaje de máquina

El “**lenguaje de máquina**” es también llamado “**lenguaje binario**”, ya que sus instrucciones son dadas directamente mediante mecanismos de tipo switch(apagado=0, encendido=1), fue el primer lenguaje de programación ya que nació junto con la primera computadora, y de algunas de sus características se pueden mencionar:

- Dependiente de la máquina a la que se dirige el programa.
- Los programas No tienen transportabilidad.
- La programación es sumamente difícil.
- El período de enseñanza-aprendizaje es muy largo

La programación en lenguaje de máquina por lo difícil de definir sus instrucciones es sumamente tediosa, ya que el programador debe “bajarse” al nivel de entendimiento de la máquina para indicarle paso a paso y en forma numérica las acciones a realizar. Esta tarea es muy propensa a errores y la productividad del programador era muy limitada, pero también hay que reconocer que en ese tiempo era la única alternativa existente.

Debido a la problemática del lenguaje de máquina se diseñó un nuevo lenguaje que permitiera reducirla y surge el **lenguaje ensamblador**, el cual es la representación simbólica del lenguaje de máquina, y significó un avance al intentar reducir las dificultades de la programación, ya que contiene un conjunto de símbolos alfabéticos como equivalentes a los símbolos numéricos de los lenguajes de máquina, solo en el direccionamiento de los operandos se usan símbolos numéricos. Esto redujo la propensión a errores de parte del programador, sin embargo la programación en lenguaje ensamblador requiere de conocer perfectamente la arquitectura del computador para el cual se construye el programa. Cada lenguaje de máquina tiene su lenguaje ensamblador asociado.

Tanto los lenguajes de máquina como los lenguajes ensambladores son considerados “dependientes de la máquina”.

El lenguaje ensamblador requiere de una interfaz de conversión al lenguaje de máquina para que pueda ser ejecutado, esa interfaz es llamada precisamente “programa ensamblador”. El programador codifica el programa(las instrucciones) de acuerdo a los símbolos del lenguaje ensamblador y esa codificación considerada el “programa fuente” y generalmente es almacenada como un archivo. Con el “programa fuente” se realiza el proceso de “traducción” al lenguaje de máquina, y el “traductor” es precisamente el lenguaje “ensamblador” que genera un archivo en lenguaje de máquina para ser ejecutado en la computadora.

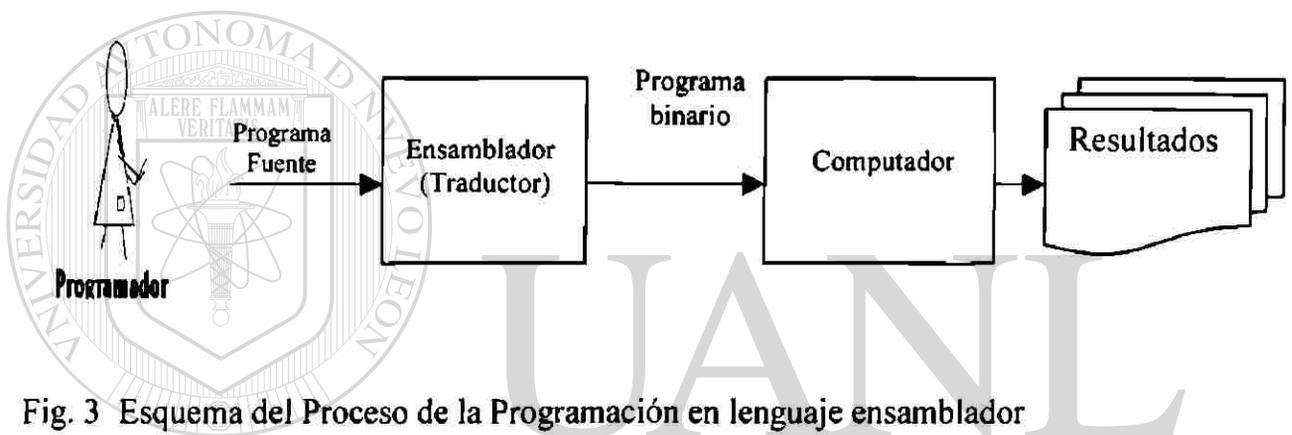


Fig. 3 Esquema del Proceso de la Programación en lenguaje ensamblador

El lenguaje ensamblador redujo los errores triviales, como puede ser el número que corresponde a una operación, los cuales son difíciles de detectar pero fáciles de cometer, sin embargo aún en este lenguaje es fácil para el programador perderse y cometer errores de lógica, ya que debe de “pensar” ó “bajarse” al nivel de la forma en como trabaja el CPU, y entender todo lo que sucede dentro del mismo CPU.

Sus características son parecidas al lenguaje de máquina :

- Dependiente de la máquina a la que se dirige el programa.
- Los programas No tienen transportabilidad.
- La programación sigue siendo tediosa y difícil.
- El período de enseñanza-aprendizaje se reduce comparado con el lenguaje de máquina.

Un ejemplo de los símbolos que se usan en instrucciones del lenguaje ensamblador son los siguientes, los cuales son para el microprocesador 80386/80286:

Símbolo	Significado	Ejemplo
ADD	Suma	ADD AX,AX
MOV	Copia una fuente a un destino	MOVE AX,TABLA_1
MUL	Realiza la multiplicación	MULT VAL_X
NOP	NO OPERACIÓN	NOP
PUSH	Carga palabra en pila	PUSH EBX
INC	Incrementa en 1	INC AX
IN	Entra Byte ó palabra	IN AX W_P_ADR
JA	Bifurca si anterior	JA INST_LABEL
JMP	Bifurcación incondicional	JMP FAR_LABEL
LAR	Carga byte acceso correcto, Selector	LAR ARB,SELECTR
LEA	Carga dirección efectiva de Desplazamiento	LEA AX,[BP][DI]
LOCK	Activa la señal BUS LOCK	LOCK MEM_WORD,AX
LOOP	Control de lazo con Contador	LOOP AGAIN
FIDIV	División entera	FIDIV WORD_INT
FINCSTP	Incrementa puntero de pila	FINCSTP

Respecto a la clase de lenguajes procedurales se considera a los lenguajes Fortran I y Algol58 como los lenguajes procedurales de primera generación, los cuales fueron creados entre 1954 y 1958, los cuales se basaban en expresiones matemáticas.

3.1.2 Lenguajes de segunda generación.

En las décadas de 1950 y 1960 inició el desarrollo de algoritmos de alto nivel, y científicos de otras ramas del conocimiento tales como la física y la química desearon ingresar al uso de la herramienta de la computadora para aplicaciones según su especialidad, por lo que se generó para ellos un nuevo “traductor” al cual se le dio el nombre de “compilador”, con él nacieron los lenguajes de alto nivel y el primer compilador fue para el lenguaje FORTRAN. Esto significó un nuevo esfuerzo en la búsqueda de una mayor abstracción del lenguaje de máquina, facilitando la tarea del programador en la construcción de sus programas y aumentando su productividad.

La aparición de los lenguajes procedurales FORTRAN II, COBOL y ALGOL60(1959-1961), marcaron el surgimiento de la segunda generación de lenguajes de programación de computadoras, estos lenguajes realizaban el proceso de “traducción”, mediante el “traductor” llamado “Compilador”.

El Compilador además del proceso de “traducción” incluye otro proceso donde utiliza una nueva herramienta llamada “Linker”, que utiliza un nuevo concepto llamado “funciones de biblioteca”, los cuales son subprogramas que realizan tareas específicas tales como :

- Calcular una raíz cuadrada
- Calcular el valor absoluto de un valor numérico
- Redondeo de un valor numérico

etc..., y estos subprogramas son incluidos en los archivos del compilador, almacenados en una biblioteca ó librería de donde son extraídos al momento del proceso de “compilación” e insertado su código de máquina en los lugares donde son enunciadas en las instrucciones del programa fuente. La utilización de “Compiladores” es un esfuerzo por lograr que los programas de aplicación sean independientes de la máquina donde son realizados y puedan ser ejecutados en cualquier computadora digital.

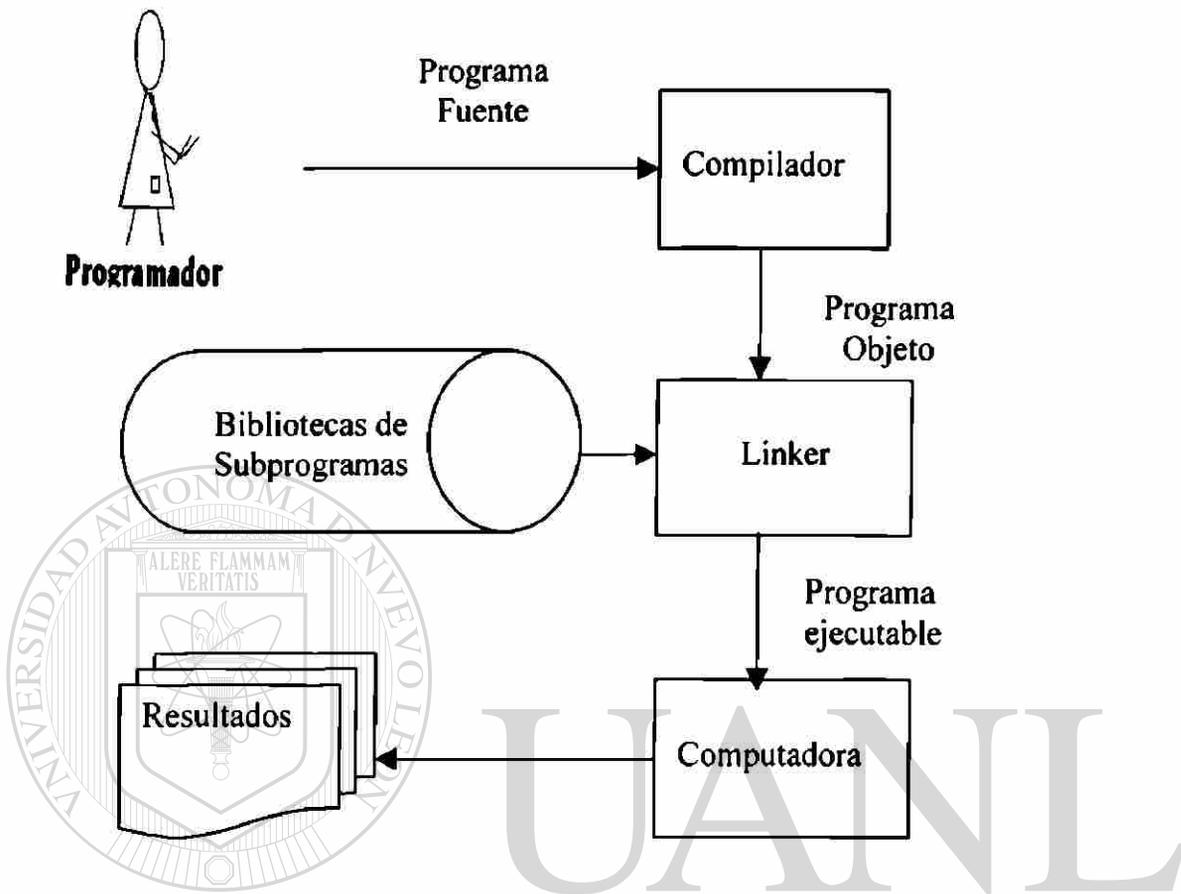


Fig. 4 Esquema del Proceso general de compilación y ejecución de un programa

DIRECCIÓN GENERAL DE BIBLIOTECAS

En este punto es importante señalar que los programadores no seguían ninguna técnica estandarizada en el desarrollo de programas, cada programador “creaba” sus aplicaciones bajo su muy particular punto de vista(lógica), y no se acostumbraba generar una documentación de la aplicaciones, por lo que el desarrollo, depuración y mantenimiento eran sumamente lentos y caros. Un problema fuerte para las instituciones publicas y privadas era la rotación de especialistas computacionales, ya que esto dificultaba aún mas los mantenimientos de las aplicaciones.

3.1.3 Lenguajes de tercera generación.

La tercera generación en los lenguajes de programación emerge con la aparición de lenguajes como: ALGOL68, PL/I, y PASCAL (1962 a 1970), así como una nueva versión del lenguaje Cobol estructurado, los cuales continúan con el uso de compiladores para la generación de los programas en lenguaje de máquina, con los lenguajes de ésta generación surgen las técnicas de “Programación Modular” y la “Programación Estructurada”, con lo que se intenta crear escuela de programadores con alta eficiencia en el desarrollo de aplicaciones, surge también el concepto de “grupos de desarrollo”, donde con las nuevas técnicas se hace posible que un programa pueda ser desarrollado por varios programadores, a través del uso de descomposición modular del problema general.

La técnica de programación estructurada de origen fue diseñada para facilitar el desarrollo de aplicaciones grandes y complejas. Al final de la década de 1970 emergieron numerosos lenguajes de diferentes tipos, en ese tiempo aproximadamente el 85% de los programadores desarrollaban sus aplicaciones en lenguaje Cobol.

En esta generación surge una gran cantidad de lenguajes y algunos los clasifican como **procedimentales** y **declarativos** así como al tipo de aplicaciones a que son dirigidas:

- Científicos(Fortran y APL).
- De gestión de datos(Cobol y RPG).
- De Aplicaciones de propósito general(Basic, Pascal,PL/I).
- Educativos(Logo,Pilot).
- Programación de Sistemas(C,ADA).
- De Inteligencia Artificial(Lisp,Prolog).

También surgen las teorías de **Lenguajes Orientados a Objetos**: (Smalltalk).

La “Programación Modular” y la “Programación Estructurada” son técnicas muy usadas en la actualidad, pero la aparición de nueva tecnología ha aumentado la complejidad de las aplicaciones por lo que ésta ha rebasado a las técnicas mencionadas.

Los lenguajes de tercera generación (ó lenguajes de “alto nivel”) son lenguajes que permiten expresar los algoritmos en forma fácil, legible y comprensible para otros programadores, además el uso de compiladores le otorga las siguientes características:

- Independiente de la máquina de la que se dirige el programa.
- Los programas Si tienen transportabilidad.
- La programación se facilita siguiendo una metodología de desarrollo estándar.
- Se reduce considerablemente el período de enseñanza-aprendizaje

Algunos lenguajes de ésta generación usan otro tipo de “traductor” llamado “interprete”, el cual tiene como característica que no genera código ejecutable sino que genera un código intermedio llamado pseudocódigo, que se ejecuta al momento de “interpretar” la acción a realizar, pero que al terminar la ejecución del programa el pseudocódigo se pierde. Esta forma de operar obliga a que en cada ejecución de un programa se realice la “traducción” directamente del programa fuente. ®

3.1.4 Lenguajes de cuarta generación.

Cuando los programadores empezaron a usar metodologías estandarizadas inició también la documentación realizando sus algoritmos bajo las técnicas de: diagramas de flujo, Pseudocódigo, Nassi-Schneiderman, y otras mas. En la década de 1980 cuando estuvieron disponibles las PC's y las interfaces gráficas algunas compañías usaron herramientas gráficas que les permitieran analizar y crear sus diagramas estructurados directamente en la computadora y guardadas en archivos.

Por el mismo tiempo otras personas crearon herramientas que generaban código directamente a partir de diagramas estructurados, ya que cada símbolo gráfico de una estructura gráfica "significaba" ciertas palabras que eran traducidas a una ó mas instrucciones de algún lenguaje procedural. Al conjuntar las herramientas de gráficas de diagramas estructurados, y la generación automática de código se creó una nueva herramienta llamada CASE(Computer-Aided Software Engineering). Algunos proveedores de Software CASE promocionan este producto como una herramienta de desarrollo de cuarta generación.

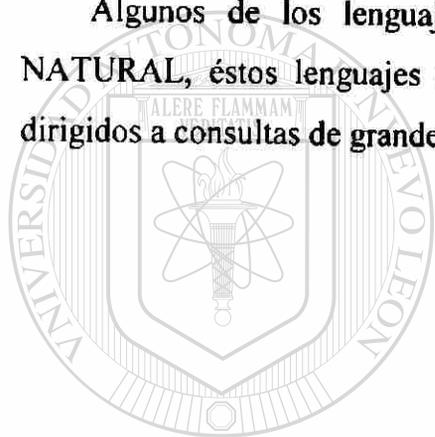
Los lenguajes de esta generación intentan superar los problemas de los lenguajes de la tercera generación, haciendo manejable la complejidad de las aplicaciones usando la nueva tecnología computacional. Con los lenguajes de cuarta generación el programador hace uso de un conjunto de instrucciones secuenciales así como una gran variedad de mecanismos como formularios para la interacción en pantalla.

El uso de Bases de datos relacionales generó la posibilidad de almacenar una gran cantidad de datos que fueran independientes de cualquier aplicación, de forma que pudiesen generarse aplicaciones cortas y fáciles que accasaran y explotaran la información contenida en las Bases de datos. A los lenguajes que permiten realizar lo anterior son considerados los lenguajes de cuarta generación(4GLs).

Los lenguajes de cuarta generación (4GL's) permiten acceder Bases de datos relacionales.

Algunos lenguajes de esta generación están basados en cuestionarios, otros son generadores de documentos ó de gráficos, con ellos el usuario estipula lo que desea realizar y el lenguaje se encarga él solo de definir “cómo” hacerlo.

Algunos de los lenguajes de la cuarta generación son: MANTIS, IDEAL, NATURAL, éstos lenguajes son de propósito general. SQL, y QBE son lenguajes dirigidos a consultas de grandes Bases de datos.



UANL

UNIVERSIDAD AUTÓNOMA DE NUEVO LEÓN



DIRECCIÓN GENERAL DE BIBLIOTECAS

3.1.5 Lenguajes de quinta generación.

Los lenguajes de esta generación permiten al programador el ingreso de “hechos” y consultas, sin especificar ningún algoritmo para obtener la respuesta deseada.

Estos lenguajes permiten el diseño de programas que emulan un comportamiento inteligente, sus aplicaciones son consideradas campos de la Inteligencia Artificial(IA).

LISP y PROLOG son los lenguajes prototipo de ésta generación de lenguajes ya que facilitan aplicaciones como : juegos de ajedrez, aplicaciones para diagnósticos médicos, diagnósticos mecánicos etc.

Algunas de las aplicaciones ó áreas de la Inteligencia Artificial son:

- Sistemas Expertos
- Robótica
- Procesamiento de lenguajes naturales
- Visión Artificial
- Modelos del conocimiento

3.2 Panorama histórico de los lenguajes de programación.

Una visión retrospectiva de los lenguajes de programación nos permite visualizar su evolución histórica y de este modo apreciar sus características diferentes en el tiempo, los lenguajes mas modernos actualmente nos aconsejan no usar en lo posible la instrucción: goto, para acciones de bifurcación. Esto es correcto de acuerdo a la metodología de la programación estructurada y a la ingeniería de software.

A este respecto conviene recordar que hubo un tiempo en que la instrucción goto en combinación con el if eran la única herramienta disponible para el programador, ya que no existían instrucciones como el for,while ó el if-then-else.

Algo importante que nos permite la historia es ver la evolución de familias de lenguajes, así como la influencia que ejercen las arquitecturas de las computadoras en el diseño de los lenguajes de programación, y evitar futuros defectos de diseño aprendiendo de los diseños anteriores.

En seguida se muestra una gráfica donde muestran algunos de los lenguajes de programación existentes hasta la década de 1980, hubo algunos lenguajes que no se hicieron populares y que tenían nombres muy diversos tales como: MAD, AMBIT, BASEBALL. Los lenguajes que se mencionan se eligieron por su gran aceptación y uso entre los programadores así como por sus características.

Las líneas continuas que hacen relaciones entre los lenguajes indican ascendencia directa, mientras que las líneas discontinuas indican solo influencia. Los lenguajes prefijados con las letras ANS indican que este lenguaje ha sido adoptado por la American National Standards Institute(ANSI), como un estándar nacional., reforzando de esta forma la “Transportabilidad” de programas de una máquina a otra, ya que todos los proveedores de computadoras deberán implementar este lenguaje en sus productos.

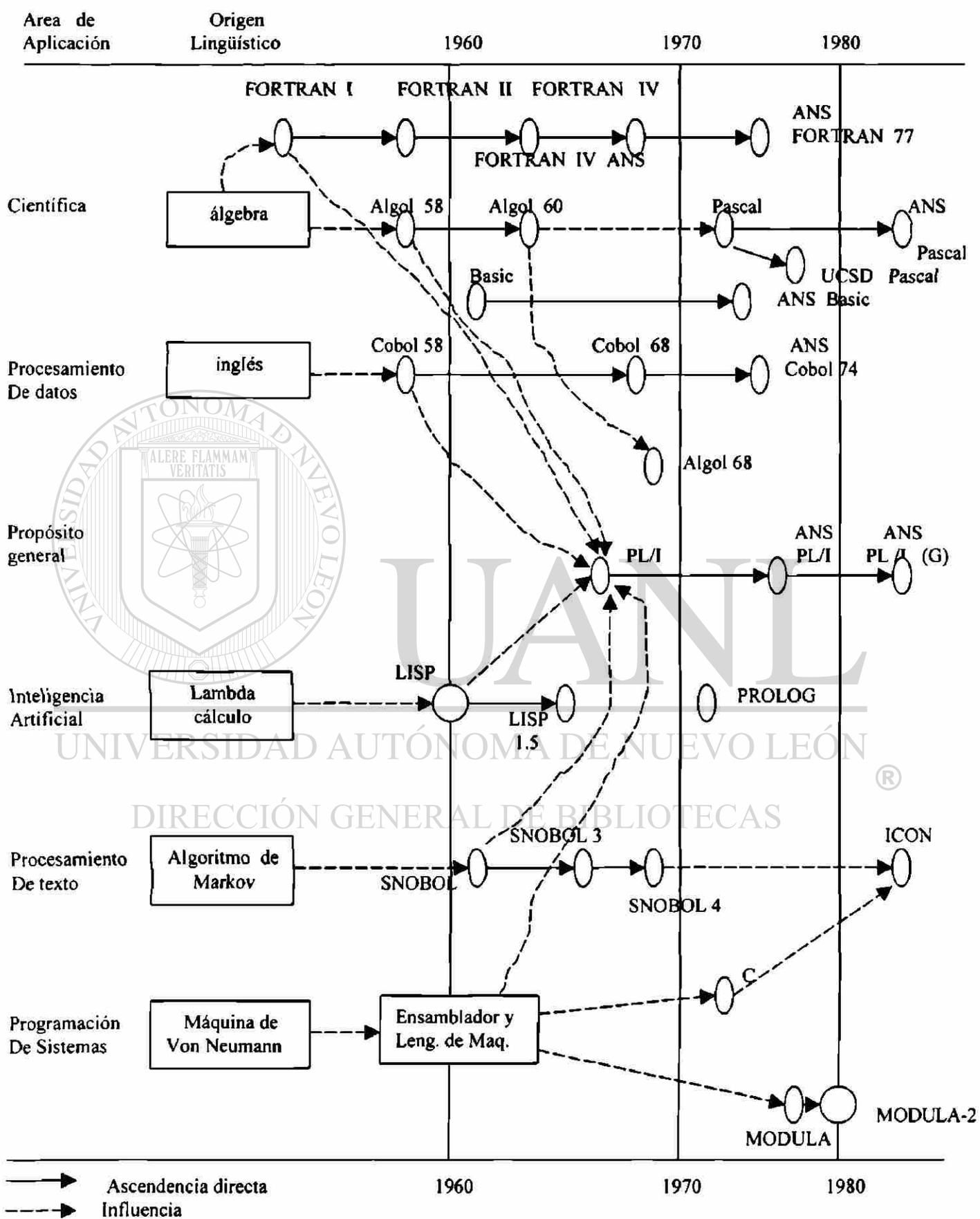


Fig. 5 Perspectiva histórica de algunos lenguajes de programación

Simula-67 fue el lenguaje que introdujo los conceptos de “clase” y “herencia” los cuales fueron adoptados por el lenguaje Smaltalk, éste lenguaje aparece comercialmente en 1976 y tuvo una gran aceptación con su versión Smaltalk-80, pero su versión mas reciente la Smaltalk/V de la empresa Digital ha tenido una buena aceptación ya que opera bajo Windows, con un ambiente de desarrollo integrado y operado mediante menús.

La mayoría de los lenguajes orientados a objetos que se han desarrollado han sido en base a lenguajes ya tradicionales como Basic, C++, Objective-C Modula-2, Object Pascal y recientemente Object Cobol.

De los lenguajes orientados a objetos mas recientes y mas populares por su versatilidad y su gran poder de cómputo es el lenguaje Java de Sun Microsystems.

UNIVERSIDAD AUTÓNOMA DE NUEVO LEÓN

Existen lenguajes orientados a objetos que no se han popularizado tales como el lenguaje Eiffel que soporta todas las propiedades y fundamentos del ambiente de objetos, y solo es usado en ambientes universitarios y de investigación, aunque su versión mas reciente ya opera en ambiente Windows y se espera que aumente su difusión. Ada es otro lenguaje que con su versión mas reciente Ada-95 ya soporta los conceptos de herencia y polimorfismo.

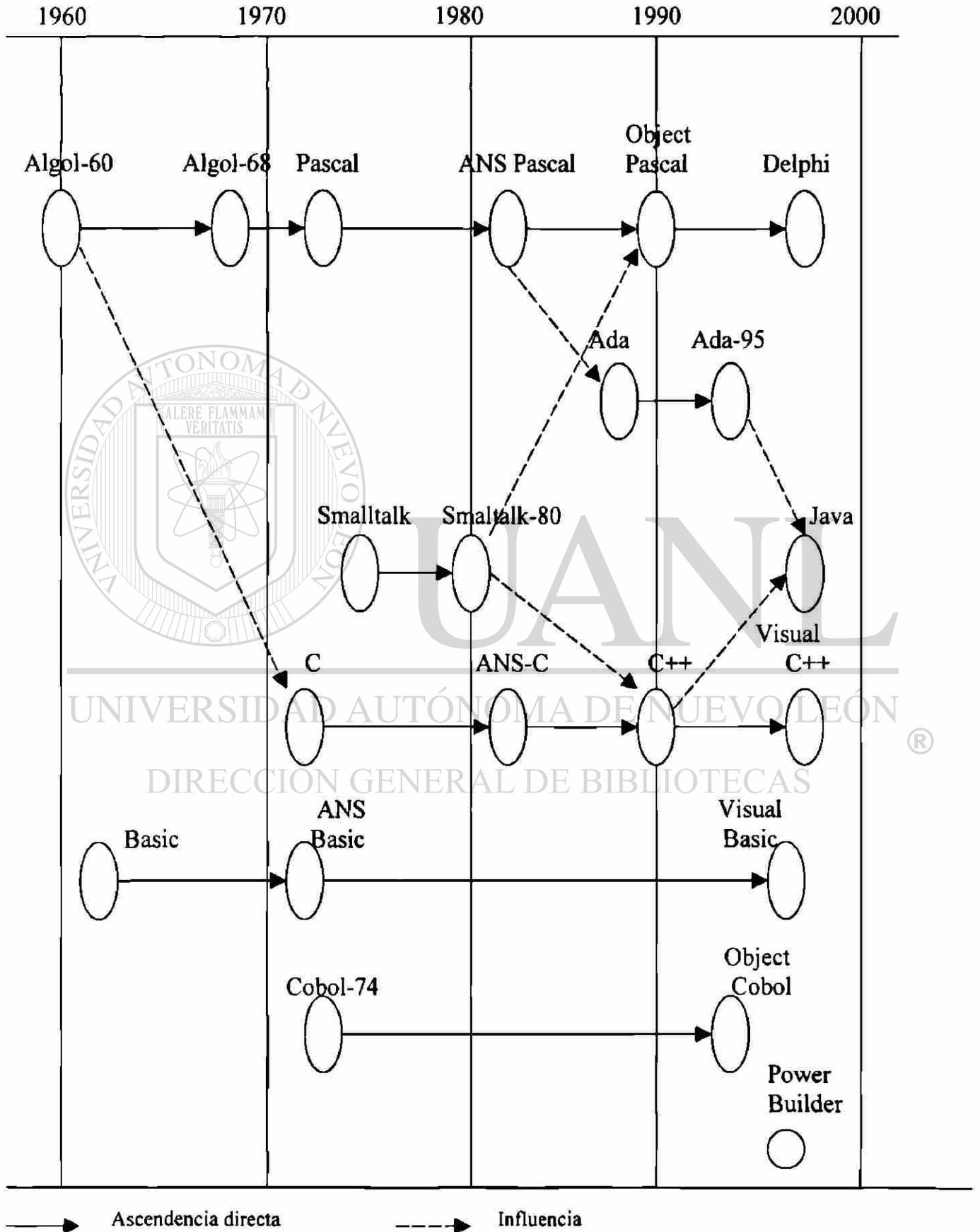


Fig. 6 Perspectiva histórica de los Lenguajes Orientados a Objetos

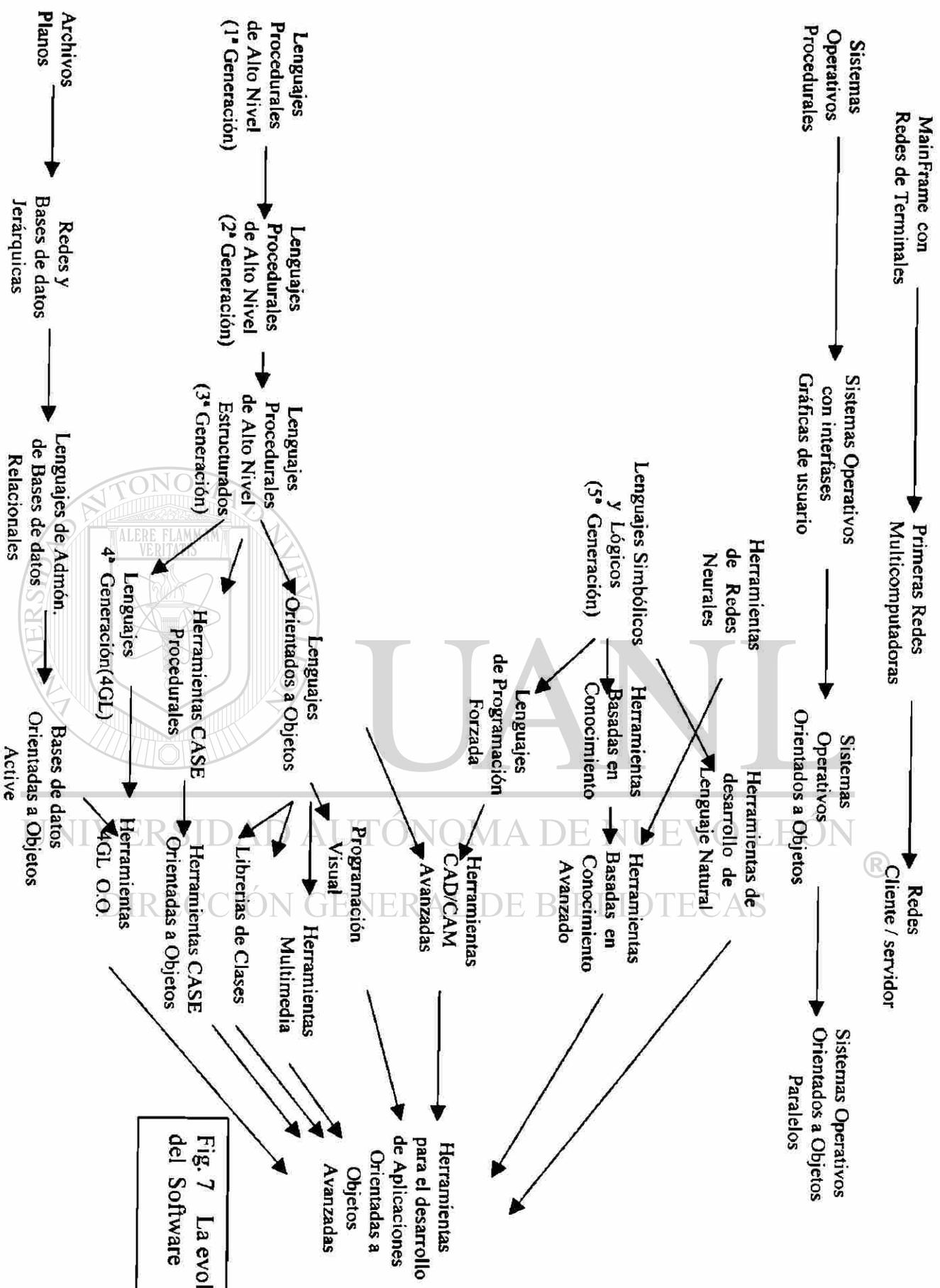


Fig. 7 La evolución del Software

4

IMPORTANCIA DE LA ENSEÑANZA DE TÉCNICAS ALGORITMICAS

Uno de los rasgos del perfil del profesionalista del área de Sistemas Computacionales es la capacidad de razonamiento analítico de los problemas, ya que ésta será la forma en que abordará los diversos problemas que se le plantearán en el ámbito de sus actividades. El término analítico implica la descomposición de un todo haciendo distinción de los elementos constitutivos(ó partes), haciendo de esta actividad un método de solución de problemas. Los alumnos de Ingeniería en Sistemas Computacionales ejercitan esta cualidad mediante el aprendizaje del desarrollo de programas de computadoras para la solución de diversos tipos de problemas. En el aprendizaje de la programación de computadoras los alumnos se enfrentan a dos problemas:

- Aprender a analizar el problema a resolver y formular el procedimiento de solución, éste debe quedar definido dentro del “Marco Computacional” ó sea en acciones que sean fáciles de traducir a una computadora.
- Debe aprender a comunicar a la computadora el procedimiento de solución que ha desarrollado, mediante “órdenes” que entienda y ejecute la computadora.

El primer problema requiere de entusiasmo, disciplina, desarrollo de una lógica para la aplicación del método analítico, y temple para no darse por vencido ante los intentos fallidos, el segundo problema requiere del conocimiento de los “comandos” de operación de un lenguaje computacional específico.

4.1 Técnicas Algorítmicas.

La solución de problemas mediante el uso de computadores es un rasgo que caracteriza a los profesionistas de la carrera de Ingeniería en Sistemas Computacionales, esto lo realizan a través de generar aplicaciones ó programas orientados a problemas específicos.

Para la generación de aplicaciones ó programas se sigue un procedimiento de Programación que incluye las siguientes fases:

- Análisis del Problema a resolver
- Diseño del Algoritmo de solución
- Codificación del Algoritmo en un Lenguaje Computacional específico
- Ejecución y validación de resultados
- Liberación ó puesta en producción
- Mantenimiento

La observancia estricta del orden descrito de las fases es requisito fundamental para que las aplicaciones generadas logren los resultados deseados.

Las fases 1 y 2 son para la Programación de computadoras lo mismo que los cimientos para un edificio: la base que sostiene toda la estructura, si la base es sólida soportará todas las pruebas a que sea sometida, en caso contrario todo lo que se construya sobre ella no podrá sostenerse.

La fase del Análisis del problema tiene como objetivo discernir alternativas ó procedimientos de solución al problema planteado, y seleccionar la alternativa mas factible.

La fase de Diseño del Algoritmo tiene como objetivo representar en forma gráfica las acciones a realizar para lograr la solución deseada.

Definición del término Algoritmo.

Un Algoritmo es una secuencia de instrucciones a seguir para resolver un problema específico.

Para los aprendices de la programación de computadoras(alumnos), es obligatorio “aprender” y “dominar” primeramente las fases:

- Análisis del problema a resolver
- Diseño del Algoritmo de solución

del procedimiento para la programación de computadoras, estas fases son independientes de lenguaje computacional.

Se disponen de tres Técnicas Algorítmicas para la representación de los algoritmos a saber:

- Diagramas de Flujo

- Diagramas Nassi-Schneiderman
- Pseudocódigo

Las Técnicas Algorítmicas son herramientas que tienen por objetivo facilitar el diseño de soluciones de problemas orientadas al uso de la computadora.

4.2 Antecedentes Plan reticular ISIC-1993-296

El plan reticular ISIC-1993-296 de Ingeniería en Sistemas Computacionales tiene definida la enseñanza del Procedimiento de la Programación en forma secuenciada, ya que define primeramente el aprendizaje del método analítico en la solución de problemas en el primer periodo escolar a cursar con la materia “Diseño estructurado de Algoritmos”, y posteriormente en el siguiente periodo escolar el aprendizaje de un lenguaje de programación para el desarrollo de soluciones mediante el uso de la computadora.

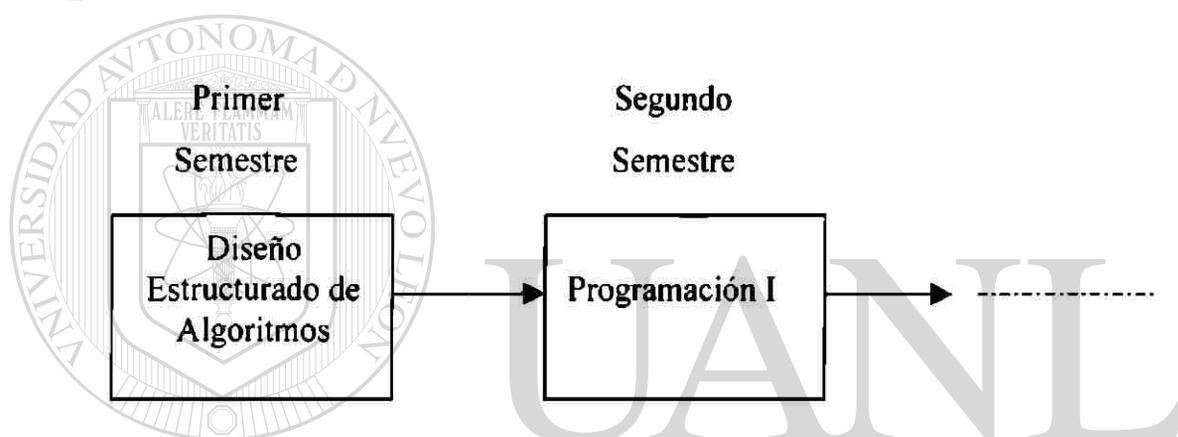


Fig. 8 Secuencia de la enseñanza de la programación a alumnos de I.S.C. en los 2

Primeros semestres.

Nota. La línea continua en flecha significa que la materia precedente es requisito aprobar para poder cursar la siguiente materia en secuencia.

Este diseño ha sido definido tomando como base que en los planteles escolares de niveles precedentes al universitario, no han ejercitado a los alumnos en el método analítico para la solución de problemas mediante el uso de la computadora.

La materia de “Diseño Estructurado de Algoritmos” tiene el objetivo de lograr que el alumno sea capaz de plantear metodológicamente la solución a problemas susceptibles de ser computarizados, mediante la aplicación del Análisis, así como la enseñanza de diversas técnicas para representar el diseño de la solución .

La materia en si no es una materia fácil, ya que en ella los alumnos se enfrentan al problema de describir en forma lógica paso a paso las operaciones que se deben realizar para dar solución a un problema específico, sin hacer referencia a algún lenguaje computacional, sino solo en el planteamiento del problema y el diseño Algorítmico de la solución.

Esta actividad los alumnos no están acostumbrados a realizarla, y por lo tanto los ejercicios que se plantean “se les hacen muy complicados”.

El tomar costumbre de realizar la actividad del análisis del problema como el inicio para obtener el diseño de la solución es un proceso lento, la parte práctica de este proceso es la representación gráfica del algoritmo de solución mediante alguna de las técnicas mas conocidas (Diagrama de flujo, Pseudocódigo, Diagramas N-S), así de esta forma cada alumno desarrolla esta habilidad en forma individual, facilitando que cada uno de ellos puede desarrollar “su propia lógica” en el diseño y representación de la solución del problema planteado.

Sin embargo cuando por alguna situación los alumnos no le dedican el tiempo suficiente a ejercitar esta habilidad, y no se logra el objetivo del desarrollo de la “lógica” individual en el diseño de soluciones a los problemas, como resultado de esta problemática se tiene un alto nivel de reprobación en los alumnos que cursan por primera vez esta materia.

La materia “Programación I” tiene el objetivo de que el alumno continúe con el aprendizaje de las Fases del Procedimiento para la Programación, ejercitándolo en todas las fases del mismo, en esta materia se asume que el alumno “ya” aprendió a desarrollar algoritmos de solución a los problemas planteados, por lo que ahora aprenderá código y sintaxis de un lenguaje computacional.

La materia de “Diseño Estructurado de Algoritmos” es muy importante para que los alumnos logren un buen aprovechamiento del contenido de la siguiente materia en secuencia ó sea “Programación I”.

Si la materia “Diseño Estructurado de Algoritmos” no se cursa adecuadamente podría ser causa de un alto índice de reprobación en “Programación I”, y aún mas si no se incluyera en el plan reticular de la carrera.

Se podría hacer un comparativo de los alumnos de otras carreras de Ingenierías que también cursan la materia de Programación, similar a la materia de Programación I de la carrera de Ingeniería en Sistemas Computacionales.



UANL

UNIVERSIDAD AUTÓNOMA DE NUEVO LEÓN



DIRECCIÓN GENERAL DE BIBLIOTECAS

4.3 Información Estadística de Alumnos de Primer Ingreso.

Para este trabajo de tesis se realizó una pequeña investigación sobre los 6 semestres inmediatos pasados (de feb-Jun'1998 a Ago-Dic'2000), sus resultados se muestran en la siguiente gráfica.

Materia: Diseño Estructurado de Algoritmos

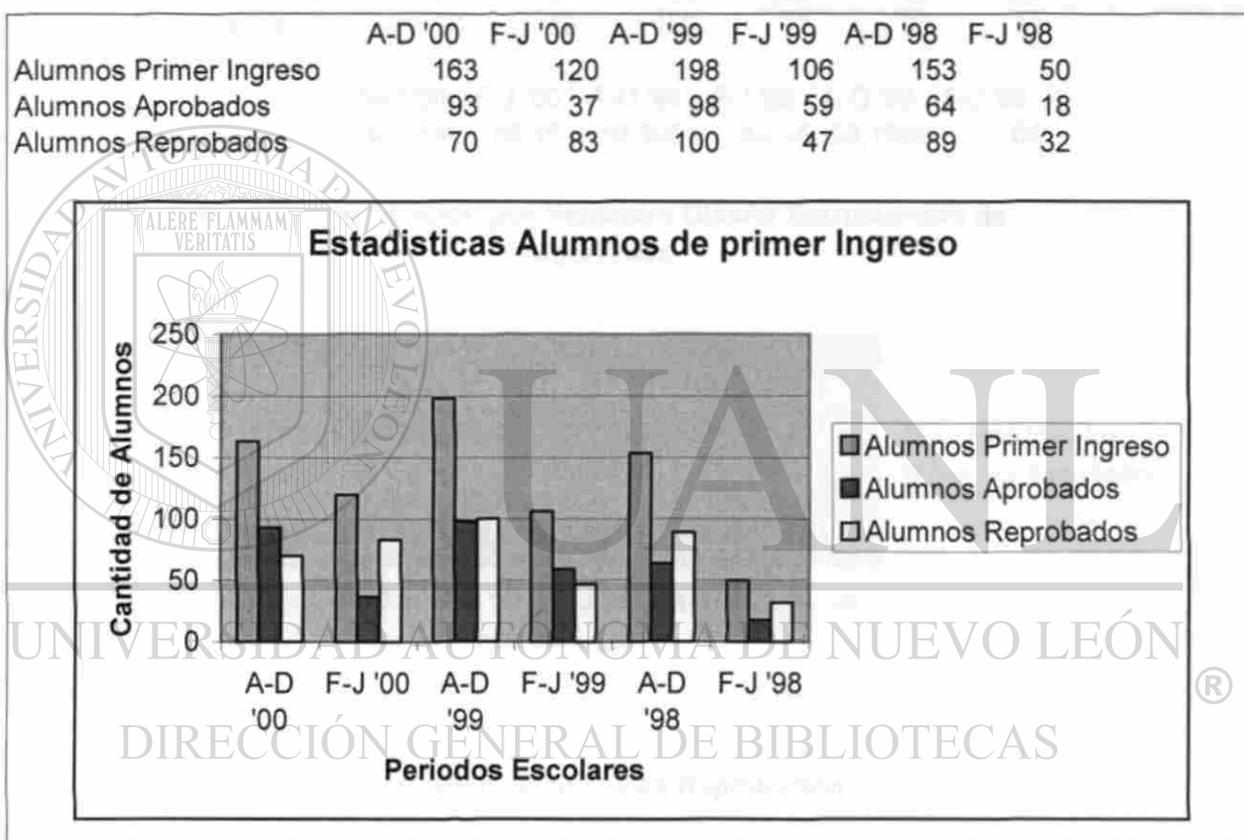


Fig. 9 Estadísticas del Nivel de reprobación en Diseño Estructurado de Algoritmos

El significado de las claves son:

- A-D Agosto-Diciembre
- F-J Febrero-Junio
- '00 Año 2000
- '99 Año 1999
- '98 Año 1998

Como puede observarse en la gráfica el nivel de reprobación en la materia mencionada es muy alto, si calculamos el índice de reprobación que le corresponde a cada Periodo Escolar quedaría de la siguiente manera:

	A-D '00	F-J '00	A-D '99	F-J '99	A-D '98	F-J '98
Alumnos Inscritos	163	120	198	106	153	50
Alumnos Reprobados	70	83	100	47	89	32

	A-D '00	F-J '00	A-D '99	F-J '99	A-D '98	F-J '98
Indice de Reprobacion	42,9448	69,167	50,5051	44,34	58,1699	64

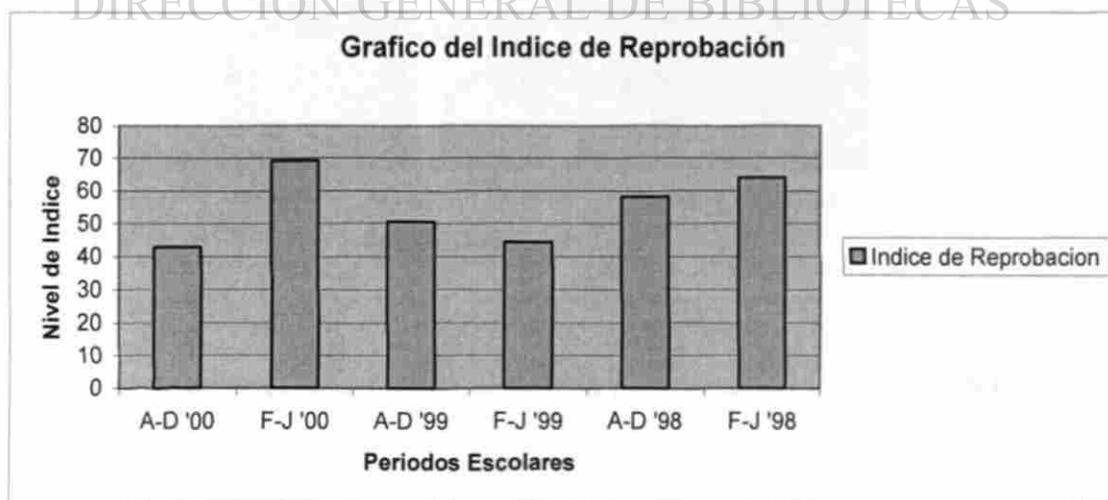
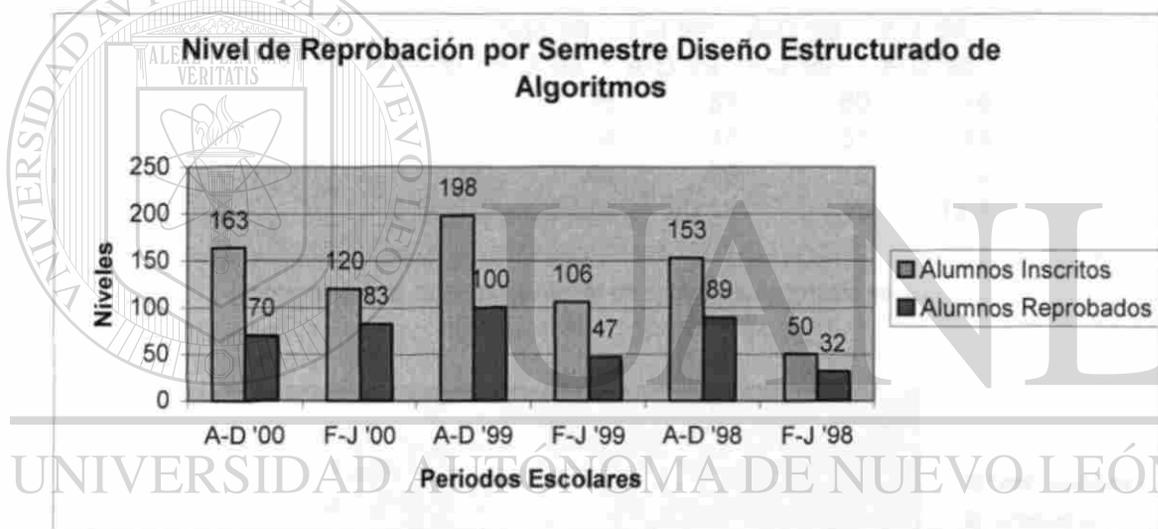


Fig. 10 Índice de reprobación en materia Diseño Estructurado de algoritmos

Se concluye que el índice de reprobación es alto en “Diseño Estructurado de Algoritmos” como primer materia relacionada con la profesión de Ingeniero en Sistemas Computacionales, sin embargo para los alumnos que logran aprobar esta materia los resultados del aprendizaje en la programación de computadoras se refleja en la siguiente materia en secuencia: “Programación I”.

En la materia “Programación I” el alumno aprenderá código y sintaxis de un lenguaje computacional, y con ello ejercitará **“Todas”** las fases del Proceso de programación desarrollando programas para diversos tipos de problemas.

	A-D '00	F-J '00	A-D '99	F-J '99	A-D '98	F-J '98
	F-J '01	A-D '00	F-J '00	A-D '99	F-J '99	A-D '98
Total Alumnos		32	98	53	60	16
Aprobados		15	54	37	51	14
Reprobados		17	44	16	9	2
Indice Reprob.		53,12	44,89	30,18	15	12,5

Seguimiento al Aprovechamiento en la Materia de Programacion I

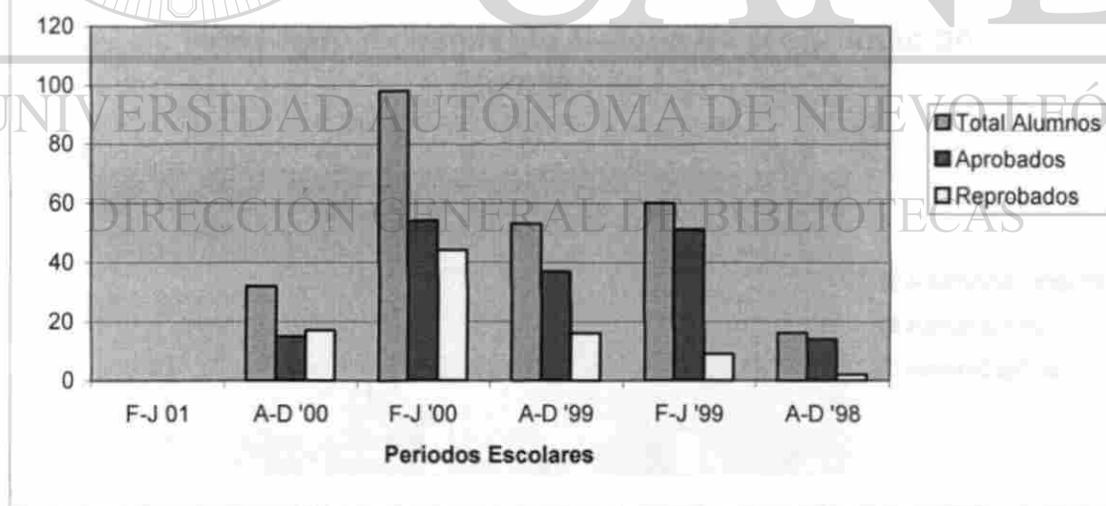


Fig. 11 Nivel de aprovechamiento en la materia programación I carrera I.S.C.

Esta gráfica nos muestra que la mayoría de los alumnos tienen un buen nivel de aprovechamiento en Programación I, y esto es debido a que al cursar esta materia ya traen un background importante en el Análisis de problemas y Diseño algorítmico de su solución.

4.4 Comparativo del Nivel de Aprovechamiento en Programación.

Otras Carreras Profesionales también incluyen en su retícula de estudios la materia de Programación, pero no incluyen el precedente de “Diseño de Algoritmos”, sino que directamente tratan el aprendizaje de código y sintaxis de un lenguaje computacional, dando poco énfasis en el Análisis del problema y el diseño algorítmico de la solución.

Para este trabajo de tesis se hizo una pequeña investigación sobre el nivel de aprovechamiento de la materia de “Programación” en los alumnos de la carrera de Ingeniería Electrónica, y su comparativo con el nivel de aprovechamiento de su equivalente “Programación I” de la carrera de Ingeniería en Sistemas Computacionales.

	A-D '00	F-J '00	A-D '99	F-J '99	A-D '98	F-J '98
Alumnos inscritos	83	10	34	16	28	13
Aprobados	44	5	19	14	13	7
Reprobados	39	5	17	2	15	6
Índice de Reprobacion	46,988	50	50	12,5	53,5714	46,154
Índice Reprob. I.S.C.	53,12	44,89	30,18	30,18	15	12,5

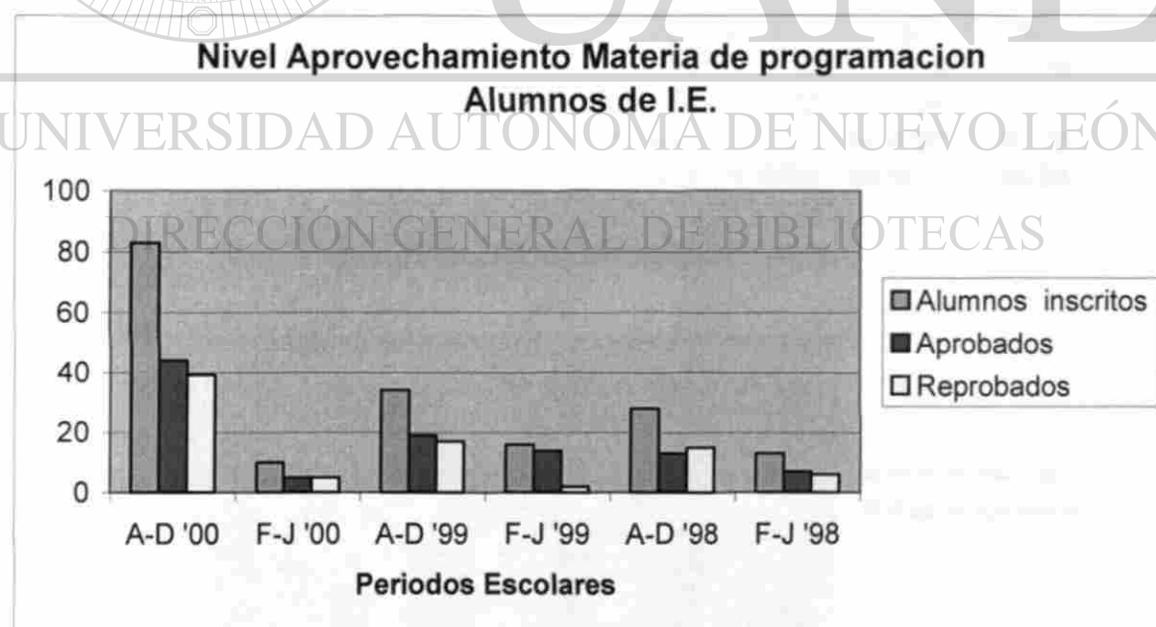


Fig. 12 Comparativo de nivel de aprovechamiento en programación I.S.C. y I.E.

Calculando un Índice de Reprobación Promedio para ambas carreras se tiene:

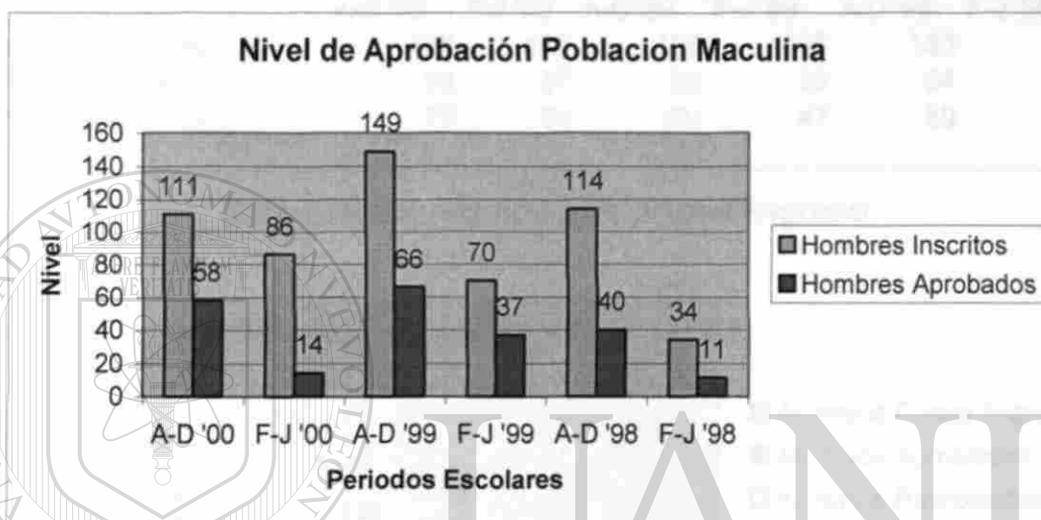
Ingeniería Electrónica 43.20

Ingeniería en Sistemas Computacionales 30.97

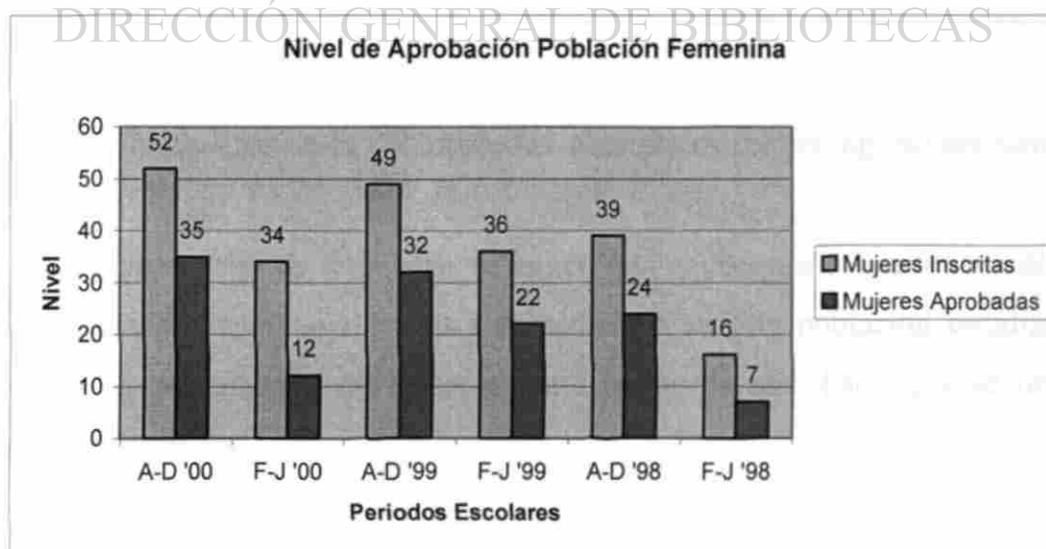
4.5 Comparativo de Aprovechamiento entre hombres y mujeres.

Información adicional es un análisis comparativo de niveles de aprovechamiento entre hombres y mujeres, el análisis nos refleja lo siguiente:

	A-D '00	F-J '00	A-D '99	F-J '99	A-D '98	F-J '98
Hombres Inscritos	111	86	149	70	114	34
Hombres Aprobados	58	14	66	37	40	11
% Hombres Aprobados	52,2523	16,2791	44,2953	52,8571	35,0877	32,3529
% Hombres Reprobados	47,7477	83,7209	55,7047	47,1429	64,9123	67,6471



	A-D '00	F-J '00	A-D '99	F-J '99	A-D '98	F-J '98
Mujeres Inscritas	52	34	49	36	39	16
Mujeres Aprobadas	35	12	32	22	24	7
% Mujeres Aprobadas	67,3077	35,2941	65,3061	61,1111	61,5385	43,75
% Mujeres Reprobadas	32,6923	64,7059	34,6939	38,8889	38,4615	56,25



Proporcionalmente aprueban la materia mas mujeres que hombres.

Fig. 13 Comparativo de aprovechamiento entre hombres y mujeres

4.6 Características de Población estudiantil de Primer Ingreso

Tema de preocupación en el Instituto Tecnológico de Nuevo León es Índice de Reprobación de los alumnos en la materia “Diseño Estructurado de Algoritmos”, lo cual se ha mencionado en el punto 4.3, donde se muestra la siguiente gráfica:

	A-D '00	F-J '00	A-D '99	F-J '99	A-D '98	F-J '98
Alumnos Primer Ingreso	163	120	198	106	153	50
Alumnos Aprobados	93	37	98	59	64	18
Alumnos Reprobados	70	83	100	47	89	32

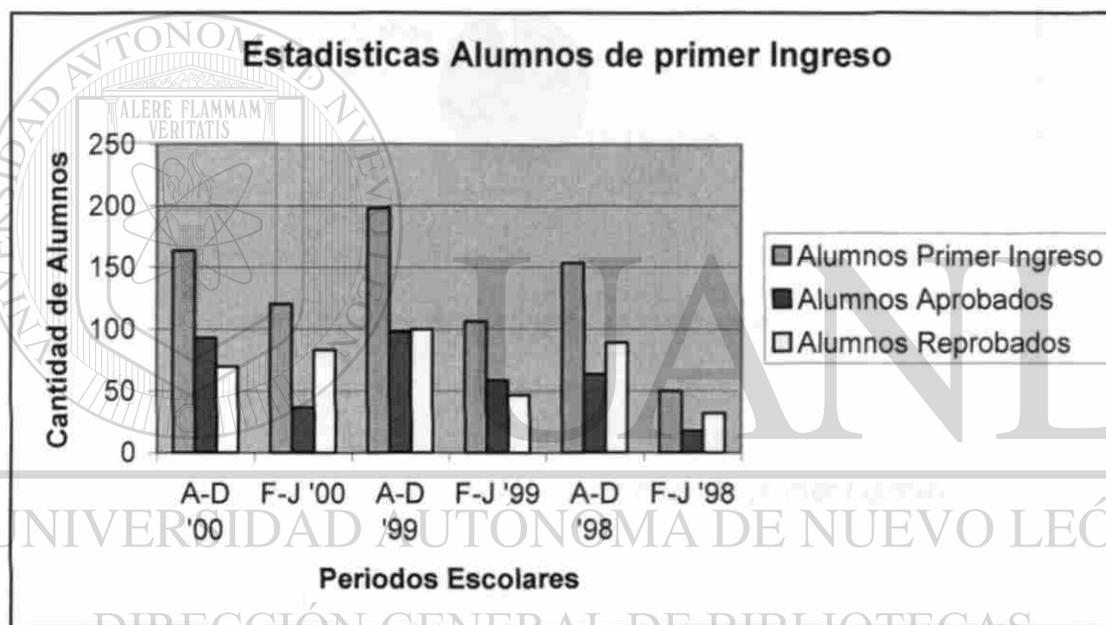


Fig. 14 Estadística Aprobados y Reprobados alumnos de primer ingreso por semestres

Para el desarrollo de ésta Tesis se aplicó una encuesta a los alumnos de primer ingreso, buscando determinar rasgos característicos de ésta población estudiantil que incidan en la presentación del problema del índice de reprobación, y se obtuvo la siguiente información.

4.6.1 Edades. de los hombres y mujeres.

17-19 años	79
20-25 años	30
25-30 años	5

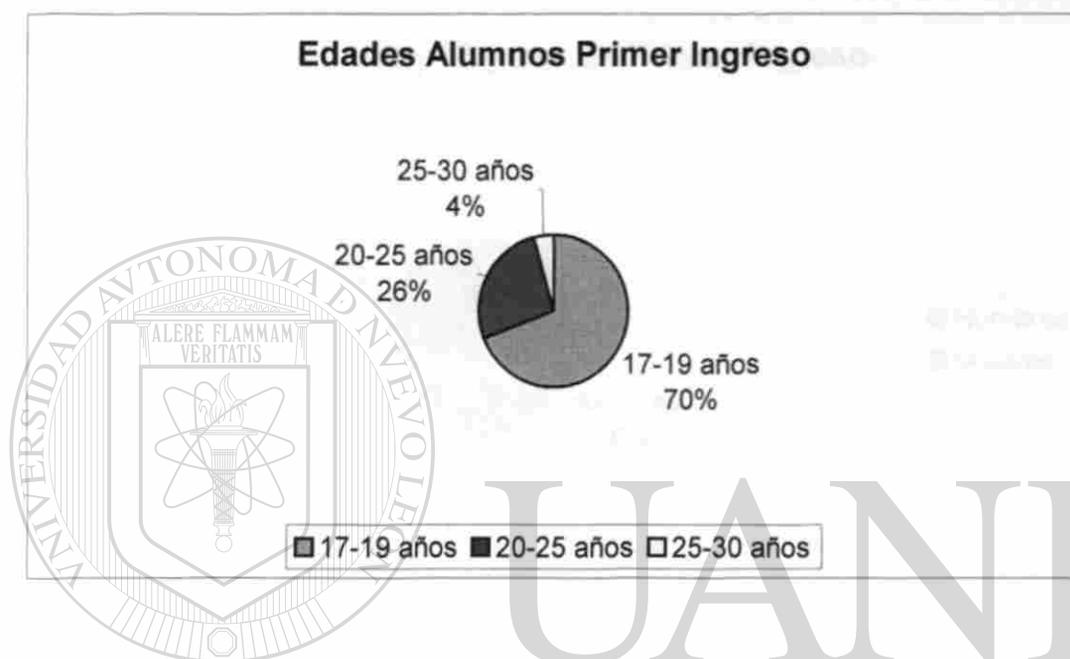


Fig. 15 Porcentajes en rangos de edades de alumnos de primer ingreso

Estos datos nos indican

- Que la mayor parte de alumnos que ingresan al I.T.N.L. son estudiantes muy jóvenes, recién egresados de la Preparatoria (ó Bachillerato).
- También ingresan alumnos que por alguna razón abandonaron sus estudios por un tiempo, y después regresan a darles continuidad .

4.6.2 Porcentajes de hombres y mujeres. Algoritmos.

Hombres	78
Mujeres	36

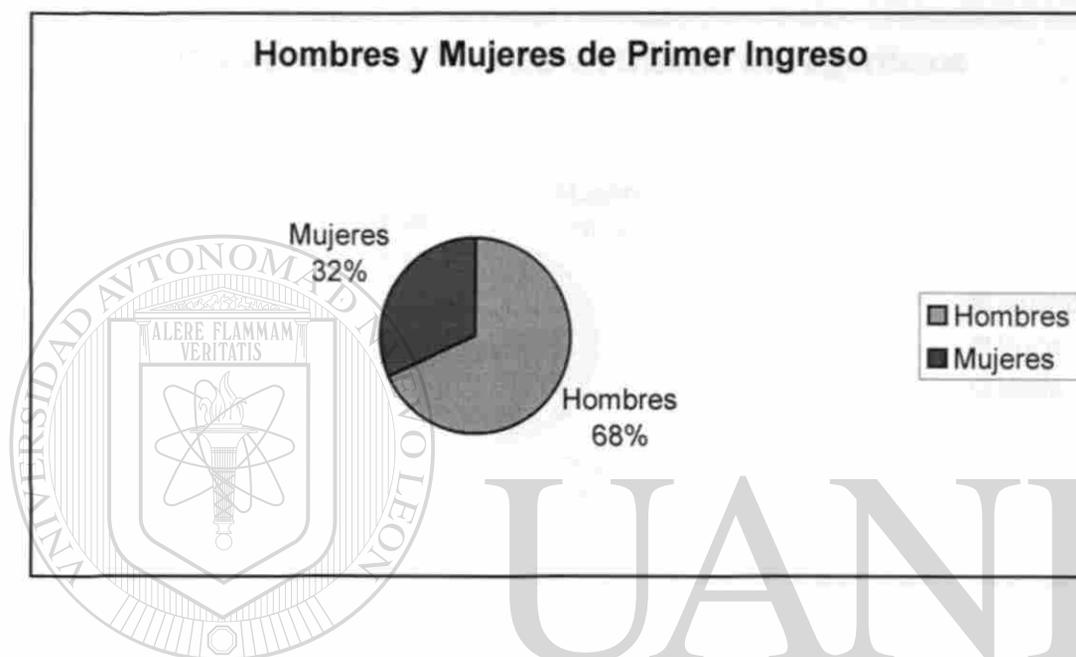


Fig. 16 Porcentajes de hombres y mujeres en alumnos de primer ingreso

DIRECCIÓN GENERAL DE BIBLIOTECAS

Estos datos nos muestran:

- Que la carrera de Ingeniería en Sistemas Computacionales por de rama ingenieril tiene mas aceptación entre los alumnos de sexo masculino.

4.6.3 Conocimientos previos en Diseño de Algoritmos.

Mucho	18
Poco	66
Nada	30

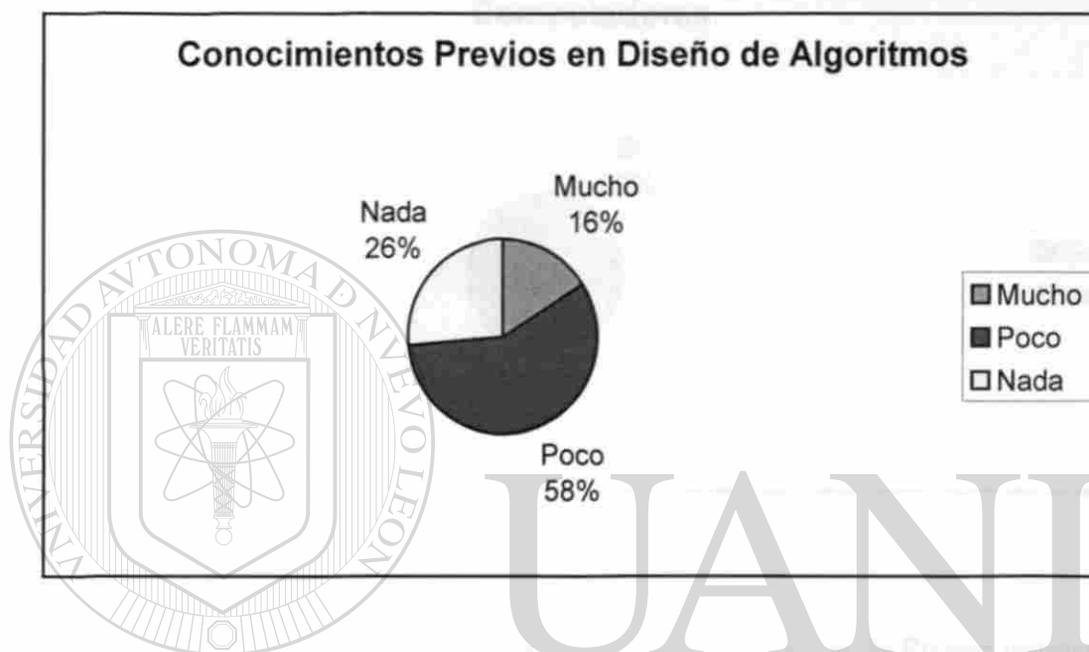


Fig. 17 Conocimientos previos en Diseño de Algoritmos en alumnos de primer ingreso

Estos datos de la encuesta nos indican:

- Que algunas Instituciones de nivel educativo precedente ya han incluido programas de estudio que realizan importante aportación en el estudiante sobre conocimientos de técnicas para el diseño de algoritmos de solución a problemas específicos.
- Sin embargo el mayor porcentaje de los alumnos reconocen que sus conocimientos al respecto son “poco” y “nada”, esto nos indica que ingresan al I.T.N.L. con cierta “debilidad” en el diseño de Algoritmos.

4.6.4 Conocimientos previos en Programación.

Si	25
No	89



Fig. 18 Conocimientos previos en programación de alumnos de Primer ingreso

Estos datos nos muestran:

- Que algunos alumnos al momento de ingresar al I.T.N.L, ya conocen ó ya tienen alguna experiencia en Programación de computadoras, pero son una mínima población de primer ingreso.
- La mayor parte de la población de primer ingreso no han tenido un acercamiento al conocimiento de algún lenguaje de programación.

La encuesta también incluía una pregunta donde se les pedía a estos alumnos que mencionaran los lenguajes de programación en los cuales ya tenían alguna experiencia, y mencionaron algunos como:

- Power Builder, Turbo C, C++, Turbo Pascal, Visual Fox Pro, Dbase III, Dbase IV, Qbasic, Visual Basic, Access, HTML.

4.6.5 Alumnos que estudian y trabajan.

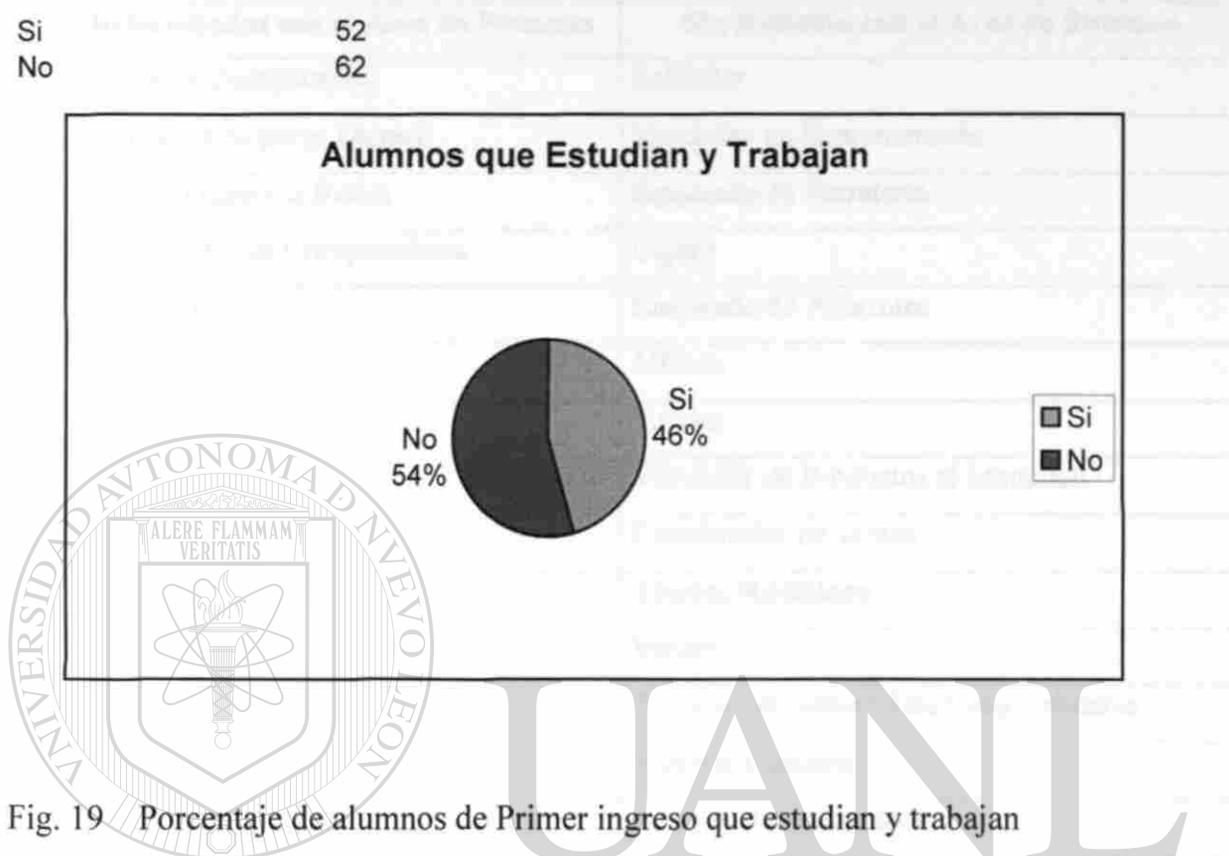


Fig. 19 Porcentaje de alumnos de Primer ingreso que estudian y trabajan

Estos datos nos muestran que:

- Casi la mitad de la población estudiantil de primer ingreso son alumnos que “ya” tienen un empleo y desean combinar las actividades de “Estudio y Trabajo”, pero esta combinación de actividades les genera un grave problema que es la administración del tiempo; ya que esto les reduce grandemente su tiempo disponible para la realización de tareas así como de tiempo de estudio fuera de las aulas de clase.

4.6.6 Tipos de empleos de los Alumnos que trabajan.

Relacionados con el Área de Sistemas	Sin Relación con el Área de Sistemas
Asesor de Computación	Soldador
Auxiliar de Soporte Técnico	Vendedor en Supermercado
Soporte Técnico a Redes	Empleado de Ferretería
Programador de Computadoras	Cajero
Capturista	Empleado de Abarrotes
	Músico
	Taxista
	Vendedor de Productos al Menudeo
	Coordinador de ventas
	Técnico Radiólogo
	Mesero
	Promotor de ventas Teléfonos Celulares
	Auxiliar Contable
	Auxiliar Administrativo
	Dueño de Negocio Propio
	Supervisor de Control de Calidad
	Empleado Empresa de Seguros
	Empleado de Carnicería
	Empleado en una Imprenta
	Instalador de Ductos de Aire
	Maestro de Primaria
	Chofer de Camión
	Electricista

Tabla 1 Tipos de empleos de los alumnos de primer ingreso que trabajan

Estos datos nos muestran que :

- La mayoría de los empleos de los alumnos de primer ingreso no tienen relación con el área de Sistemas, pero ellos tienen el deseo de a través de sus estudios relacionarse con ésta área.

4.6.7 Porqué elegir la profesión de I.S.C.

La encuesta realizada incluyó también que el alumno contestara a la siguiente pregunta ¿Porqué elegiste estudiar la carrera de Ingeniería en Sistemas Computacionales?, y sus respuestas se englobaron en las siguientes:

- Soy Técnico en Computación y deseo mayor preparación.
- Porque el área computacional tiene mucha demanda y quiero superarme.
- Porque me gusta todo lo relacionado con la computadora.
- Porque me gusta mucho hacer programas para la computadora.
- Porque es un reto para mi.
- Porque deseo conocer el funcionamiento interno de la computadora.
- Porque esta carrera tiene mucho futuro.
- Porque esta profesión es muy bien pagada.
- Porque deseo poner mi propio negocio de computación.

4.7 Conclusiones de la encuesta a alumnos de Primer Ingreso.

En la revisión de los datos de la encuesta realizada se obtuvo la siguiente información:

- La mayoría de los alumnos son jóvenes de entre 17 y 19 años, recién egresados de la preparatoria.
- La gran mayoría son alumnos de sexo masculino, solo un pequeño porcentaje son de sexo femenino.
- La mayoría de los alumnos al ingresar tienen gran dificultad al aplicar el método analítico en la solución de problemas, orientado a la programación.
- Una minoría de alumnos “ya” conocen código y sintaxis de uno ó varios lenguajes de programación, algunos de ellos ya conocen y aplican lenguajes de programación de tipo visual (RAD), pero algunos de ellos son solo “codificadores” ya que no han ejercitado el Análisis del problema y su solución algorítmica.
- ~~Casi la mitad de los alumnos “ya” tienen un empleo de jornada de trabajo completa(8 horas/día).~~
- La mayoría de los alumnos que trabajan tienen empleos NO relacionados con el área de Sistemas.

4.8 Factores que inciden en el Índice de Reprobación en la Materia de Diseño Estructurado de Algoritmos.

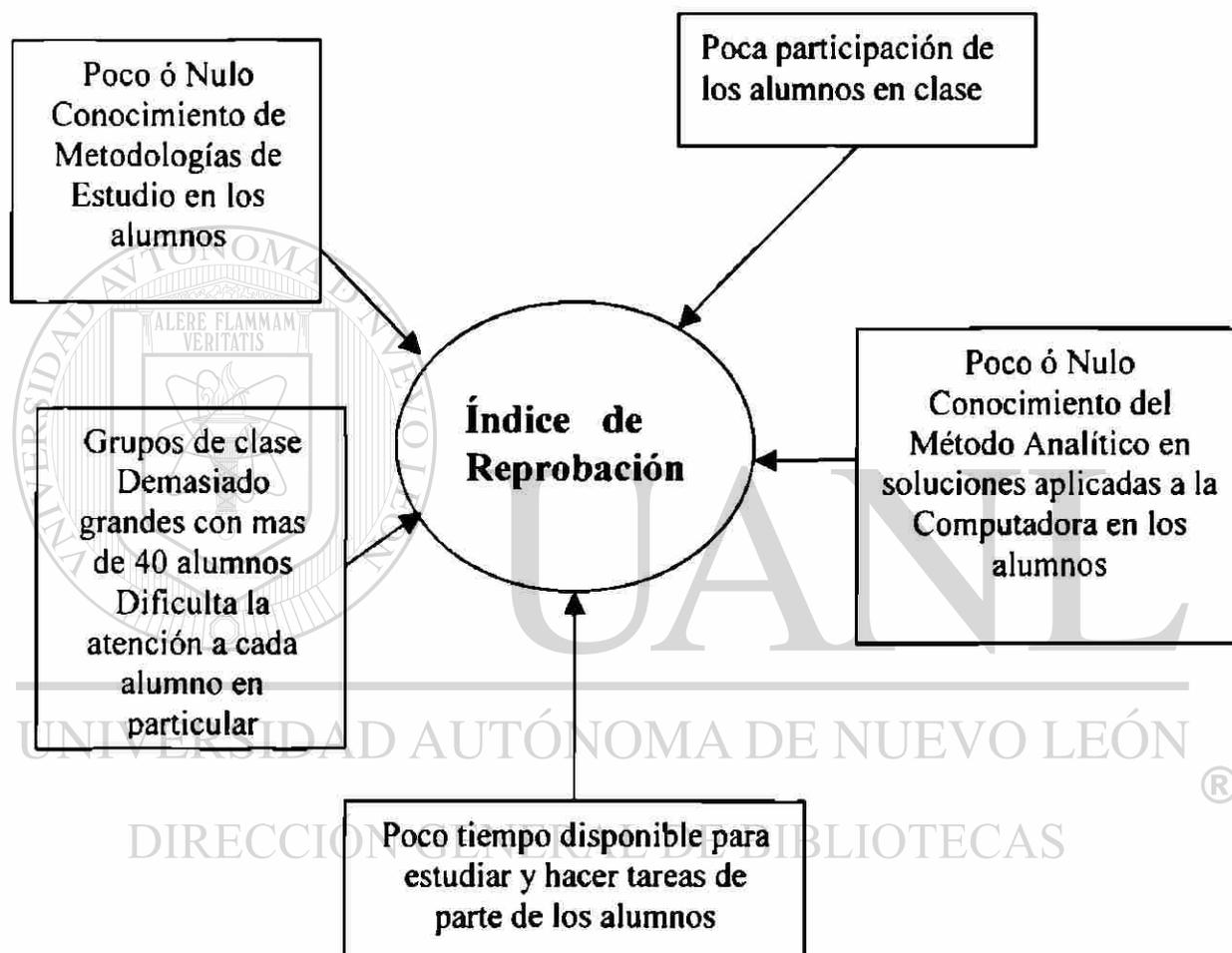
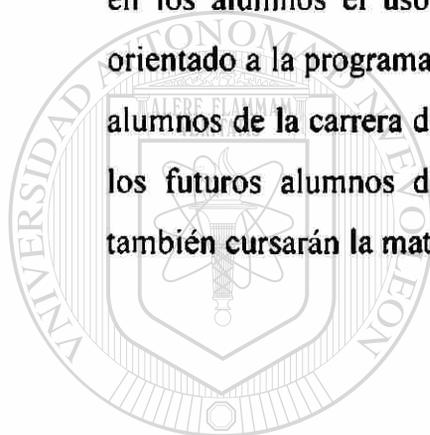


Fig. 20 Factores que inciden en el índice de reprobación de la materia de Diseño Estructurado de Algoritmos.

4.9 Propuesta para reducir el índice de reprobación en la materia de Diseño Estructurado de Algoritmos

- En los cursos propedéuticos impartir como primer tema “Metodologías de estudio”, el cual debe concluir al inicio de los demás temas a saber: matemáticas, Química, etc., de forma que los alumnos puedan aplicar un método de estudio desde el inicio de estos cursos.
- En los cursos propedéuticos incluir el tema “Lógica de programación” que inicie en los alumnos el uso del método analítico para la resolución de problemas, orientado a la programación de computadoras. Esto ayudará no solo a los futuros alumnos de la carrera de Ingeniería en Sistemas Computacionales sino también a los futuros alumnos de las demás carreras de corte ingenieril, ya que ellos también cursarán la materia de Programación.



UANL

UNIVERSIDAD AUTÓNOMA DE NUEVO LEÓN



DIRECCIÓN GENERAL DE BIBLIOTECAS

5

EL PRIMER LENGUAJE DE PROGRAMACION

La programación de computadoras nació con la misma computadora, con las primeras computadoras la programación fue sumamente difícil ya que se realizaba en lenguaje de máquina absoluto a través de conexiones de cables específicos para controlar las funciones elementales de la máquina. En la década de 1950 el proceso mejoró con la utilización de tarjetas perforadas ya que con ello se logró transferir la escritura de programas a las tarjetas y su lectura automática en vez de insertar cables en el hardware, aunque los procesos de compilación y ejecución seguían requiriendo esperar que no se fundiese algún bulbo que abortara el proceso, por ese tiempo no se disponían de sistemas operativos y normalmente el mismo personal construía, programaba y daba mantenimiento a la computadora.

La aparición del transistor y su aplicación en las computadoras logró hacerlas más confiables así como su comercialización, esto coadyuvó a una separación y especialización de funciones, ya que el mismo equipo de personas que construía la computadora no podría “ir” a programar y dar mantenimiento a todas las computadoras vendidas, de forma que las funciones “programar” y dar “mantenimiento” debían de hacerlas otros equipos de personas.

El “ambiente de programación” tiene que ver con las herramientas que faciliten al programador realizar su función, a este respecto se puede afirmar que la programación de computadoras ha ido de la mano con la evolución de los sistemas operativos.

5.1 Ambientes de Programación

El ambiente de desarrollo “batch” también es conocido como “Sistema de procesamiento por lotes” fue la manera de operar de los computadores de la primera y segunda generación que comprende de inicio 1951 hasta 1964, en estas generaciones no habían sistemas operativos, y las empresas fabricantes de computadoras producían también el software a usarse en ellas, estas computadoras solo podían ejecutar un proceso a la vez, y las características del “ambiente de programación” fueron las siguientes:

- a) El programador realizaba la codificación del programa en hojas de papel graficadas con el formato correspondiente al lenguaje en uso (Fortran, Cobol, Assembler, etc..)
- b) Pasaba la codificación a tarjetas perforadas
- c) Llevaba las tarjetas perforadas a la sala de lectura y las dejaba con el operador de computadora para la lectura y compilación del programa
- d) Esperaba a que el operador le entregara un listado emitido por la computadora dando los resultados de la compilación
- e) Si había errores se hacían las correcciones y se realizaba de nuevo el proceso a partir del paso b)

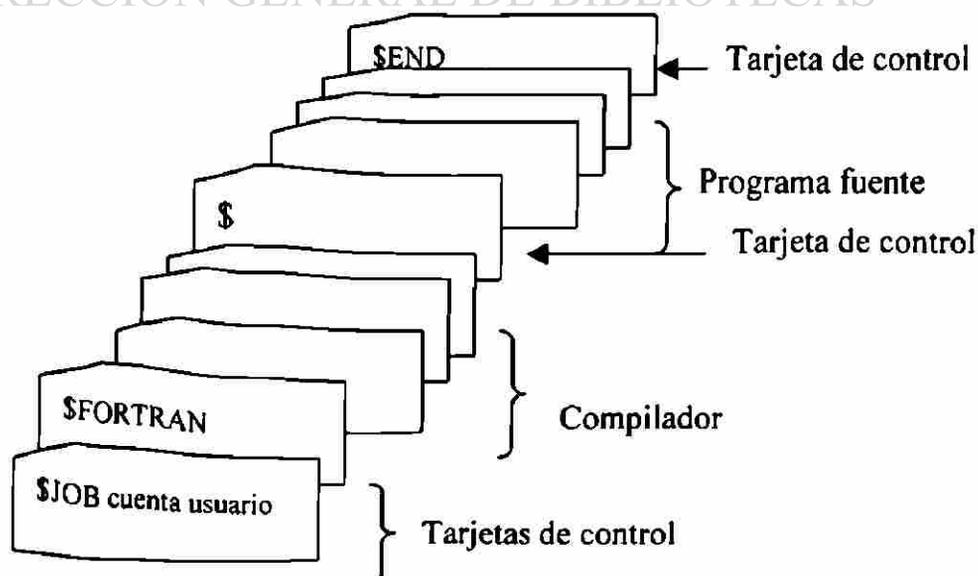


Fig. 21 Estructura del lote de un proceso de compilación

Las computadoras de la tercera generación tuvieron la característica de ser construidas en base a circuitos integrados(a baja escala), tecnología que les proporcionó ventajas en precio y desempeño además de gran poder de computo, ya que podían realizar procesos comerciales y científicos.

En esta generación (1964 – 1981) aparecen los primeros sistemas operativos y con ellos las técnicas de partición de memoria y multiprogramación, que dieron lugar a llamarlas “computadoras multiusuario”, esto les permitieron atender a varios usuarios en forma simultánea, cada usuario operaba una terminal conformada por una consola y un teclado conectados con la computadora en forma local ó remota, el uso de la consola por parte del usuario hizo posible el surgimiento del término “interactivo”.

Con los sistemas operativos aparecieron también los editores de texto que permitieron a los programadores la captura de sus programas y grabarlos como archivos en el disco magnético, donde también residían los compiladores, este “ambiente de programación y Procesos de datos” dejó obsoleto el uso de las tarjetas perforadas.

Como “ambiente de programación” la característica principal fue que tanto el editor como también el compilador y sus Librerías eran herramientas que no estaban integradas, y el programador operaba con ellas en forma separada. Varios programadores podían trabajar al mismo tiempo, operando cada uno su propia terminal, y también compilando sus programas al mismo tiempo.

Los procesos de compilación y de datos seguían siendo “batch”, y su ejecución se “ordenaba” mediante un archivo “Job Control” también capturado en el editor de texto, que contenía las órdenes mediante comandos dirigidos al sistema operativo(similar a los archivos .bat operados en MS-DOS), que le indicaban al computador las actividades a realizar , en detalle paso a paso.

En las computadoras de esta generación seguía persistiendo el hecho de que las empresas que fabricaban el hardware también producían el software que se usaría en ellas.

Las actividades del proceso de compilación eran las siguientes:

- a) El programador capturaba la codificación del programa en el editor de texto dando al archivo el formato requerido de acuerdo al lenguaje a usar (Cobol, Fortran, Assembler, etc...)
- b) El programador capturaba el archivo Job Control en el editor de texto
- c) El programador enviaba a ejecución el archivo Job Control que entraba a una cola de espera
- d) El programador esperaba el resultado que arrojaba la compilación mediante un listado de la compilación que indicaba si había o no errores, el listado de la compilación podía ser visto en pantalla a través del editor ó enviarlo a la impresora.
- e) Si había errores se hacían las correcciones en el archivo correspondiente y se repetía el proceso a partir del paso c)

EJEMPLO DE UN ARCHIVO "JOB CONTROL"

```
$JOB cuenta usuario
$lenguaje a usar
$CALL compilador
$CALL nombre del programa fuente
$END
```

Fig. 22 Segmento de un archivo Job Control

La aparición de la microcomputadora marcó el inicio de las computadoras de cuarta generación y también un parte aguas en la generación de software, ahora además del software de los diseñadores y fabricantes de hardware (IBM, Hewlett Packard, Digital Equipment, etc.) aparecieron nuevas empresas desarrolladoras de software (Microsoft, Borland Inc., Lotus Development Corp., etc.) cuyos productos pueden ejecutarse en cualquier microcomputadora que sea de arquitectura abierta (IBM y compatibles).

En el ámbito de las microcomputadoras se generó una rápida evolución de Sistemas Operativos (y con ello el “ambiente de programación”), renglón en el cual ha dominado le empresa Microsoft quien con diversas versiones del Sistema Operativo MS-DOS predominó en el mercado de las microcomputadoras.

La empresa Apple Computer fue la primera en fabricar hardware y software para microcomputadoras, y su política de mercadeo fue apartarse de la arquitectura abierta impulsada por IBM y Microsoft Co., de tal forma que hasta la actualidad su arquitectura sigue siendo “NO compatible”, sin embargo hasta la actualidad se ha mantenido en el mercado con su singular computadora modelo Macintosh.

El “ambiente de programación” imperante durante el dominio del Sistema Operativo MS-DOS fue similar al “ambiente de consola” de las computadoras de la tercera generación, solo que las microcomputadoras solo pueden atender a un usuario a la vez. Para atender a varios usuarios a la vez usando microcomputadoras varias empresas desarrollaron tecnología para comunicaciones entre computadoras (entre ellas Novell Inc.), dando lugar al ambiente de Redes LAN.

En 1983 Borland Internacional Inc. Puso a disposición del mercado computacional un compilador del lenguaje Pascal y lo llamó “Turbo Pascal”, el cual tenía las siguientes características:

- Eficiente y rápido
- Sumamente económico (\$49.95 dls.)

- Ambiente de programación “Turbo” (Editor, Compilador, Librerías y Depurador integrados como un solo programa)

este producto tuvo una gran aceptación no solo en ambientes académicos sino también los demás sectores de la computación y dio inicio al “ambiente de programación” llamado “Turbo”, para el cual posteriormente también se dispuso de otros productos como “Turbo Basic”, “Turbo C”, etc..

La programación en ambiente MS-DOS también se le conoció como “programación de 16 bits” por la capacidad de direccionamiento en su arquitectura.

En 1984 la empresa Apple Computer anunció comercialmente su modelo Macintosh que ofrecía el primer sistema operativo con ambiente gráfico operado mediante “ventanas” y de un dispositivo denominado “mouse”, a este ambiente se le denominó “Ambiente GUI” (Graphical User Interface).

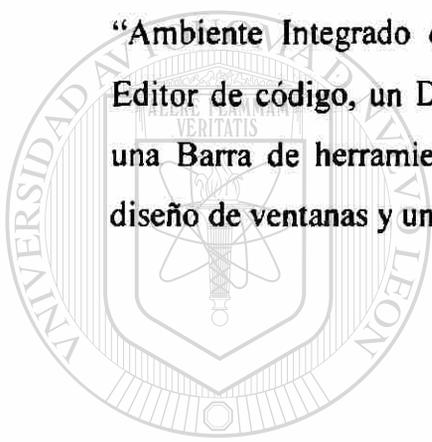
Esto representó un reto para la empresa Microsoft de manera que en marzo de 1987 anunció su primera versión de su sistema operativo “Windows 1.0”, pero no tuvo el éxito que se esperaba, y los usuarios preferían seguir usando el MS-DOS. Microsoft siguió insistiendo hasta que en 1990 anuncia la versión “Windows 3.0” con la cual logró un éxito sin precedentes, y con esta versión de “Windows” la mayoría de los usuarios tanto “caseros” como de “negocios” migraron del ambiente de MS-DOS al “Ambiente Windows”.

En estas versiones del sistema operativo aparecieron las interfases de programación API (Application Programming Interface) para versiones Windows 3.x.

En 1995 Microsoft promueve en el mercado su nueva versión de Windows denominado “Windows 95” con cambios muy significativos, y posteriormente en 1998 se reafirmó con una nueva versión llamada “Windows 98”.

Los cambios sustanciales fueron :

- Programación a 32 bits
- Programación RAD (Rapid Applications Development)
- Dispone de una gran cantidad de API y DLL's que facilitan el trabajo de la programación
- Permite la programación tipo "Visual"
- Ambiente de programación IDE (Intergrated Development Environment), ó "Ambiente Integrado de Desarrollo" cuya característica es que contiene: un Editor de código, un Depurador de errores, una barra de herramientas visuales, una Barra de herramientas de manejo de Bases de datos, un Formulario para diseño de ventanas y un Inspector de Objetos.



UANL

UNIVERSIDAD AUTÓNOMA DE NUEVO LEÓN

®

DIRECCIÓN GENERAL DE BIBLIOTECAS

5.2 El Modelo Cliente/Servidor

Los avances en la tecnología de comunicaciones entre computadoras ha permitido el amplio uso del Redes de computadoras permitiendo la distribución de los recursos de cómputo, este ambiente de redes ha desplazado poco a poco al “Modelo Host” ya que éste tiene como característica principal la centralización de procesos, en cambio el ambiente de Red permite la descentralización de los mismos.

La aparición de la microcomputadora ha dispuesto capacidad de procesamiento de datos fuera de los Centros de Cómputo, esta nueva relación cambia el paradigma de Cómputo centralizado y genera un nuevo paradigma llamado “Cliente/Servidor”, el cual tiene implicaciones tanto de hardware como de software.

En cuanto a software el término “Cliente/Servidor” define una relación entre dos programas de computación, donde uno de ellos (el Cliente) solicita un servicio, y el otro (el Servidor) satisface la solicitud. El Modelo “Cliente/Servidor” brinda una forma conveniente y simple para interconectar programas que están distribuidos en locaciones distantes. En este Modelo un “Servidor” que a veces se denomina “daemon” ó “demonio” es activado, y permanecerá en espera de alguna petición ó solicitud de algún “Cliente”, normalmente muchos programas “Cliente” podrán compartir los servicios potenciales de un “Servidor” común.

El Modelo “Cliente/Servidor” requiere disponer de varios elementos como:

Arquitectura completa de Telecomunicaciones, Administración de archivos de respaldo, Control de Procesos distribuidos, Mensajería, Comunicación entre Aplicaciones, Bases de datos Distribuidas, Lenguajes de consulta, Bases de datos independientes de los Lenguajes de consulta, Aplicaciones independientes de las Bases de datos y de los tipos de terminales.

Los componentes principales de una aplicación “Cliente/Servidor” son consideradas tres, que son:

- Interfase al usuario (presentación)
- Administración de los datos
- Procesamiento de los datos

estos componentes pueden estar distribuidos a lo largo de toda la red, éste Modelo implica soporte de ambientes complejos de múltiples plataformas y protocolos, pero reteniendo la simplicidad de un sistema individual.

En el acceso remoto de datos la Administración de los mismos se lleva a cabo en el “servidor” y el procesamiento y presentación de los datos se lleva a cabo en el “Cliente”.

El uso del Modelo “Cliente/Servidor” se ha convertido en una de las características principales de los Lenguajes de programación, ya que la gran mayoría de las aplicaciones de negocios utilizan este modelo. Los programadores en ambiente “Cliente/Servidor” deben asumir responsabilidades mayores a las que se tenían en el pasado, facilitando en sus aplicaciones la toma de decisiones operativas, tácticas y estratégicas.

5.3 Metodologías de la Programación

Los tiempos de la programación a través de interruptores eléctricos, de tiras de papel perforada y listas de bytes en hexadecimal han pasado, primero se avanzó hacia los lenguajes de alto nivel como Cobol, Fortran, Algol entre otros con la Programación Estructurada y la Programación Modular, el siguiente paso parecen ser los ahora populares lenguajes de programación visual.

Los lenguajes visuales ostentan este nombre porque proporcionan al programador de herramientas gráficas que le permiten mejorar la velocidad en el desarrollo de un proyecto, pues simplifican en gran medida la construcción de la interfaz con el usuario, lo cual representa generalmente mas del 50% del trabajo que requiere un proyecto típico. No se puede concebir la programación actual con los lenguajes de hace una década, sino con los lenguajes de hace 3 ó 4 años, muchas herramientas de desarrollo en la actualidad se han incorporado al nuevo paradigma de la programación visual.

La mayor ventaja del desarrollo visual es que permite generar aplicaciones funcionales, sencillas y a la vez elegantes en forma rápida.

Programar computadoras en la actualidad no solo es generar código para procesarlo a las mas altas velocidades posibles, ahora se requiere programar la parte medular de cualquier software(el procesamiento de la información) cuidando la interfaz con el usuario, y la generación de ayuda en línea(mediante el paradigma del hipertexto para moverse con facilidad sobre información), sin dejar de lado la obligatoriedad de un código rápido y eficiente entre otras cosas.

** Notas extraídas del artículo “El universo de los lenguajes visuales”, de los autores Alejandro Arcos, David Garza, Manuel López, Marco A. Pérez.

Article A20504689 1997-Servicios Editoriales Sayrols S.A. de C.V.

El medio ambiente computacional y las metodologías de la programación han cambiado y se han dirigido hacia el desarrollo de aplicaciones Cliente / servidor y de interfase gráfica(GUI), esto representa un cambio en el paradigma de la programación ya que las aplicaciones son “Orientadas al usuario(Visuales)” en vez de “Orientadas al Programador” y al los Sistemas Gerenciales Integrados(MIS).

Los 4GL's están orientados a incrementar la velocidad del desarrollo de las aplicaciones y a satisfacer las necesidades de los usuarios de los Sistemas denominados “desk System” ó sistemas de escritorio con calidad y consistencia, así como facilidades en el diseño y modelado de datos integrados con herramientas CASE, de forma que el Programador pueda dedicarse al diseño y optimización de la solución.

La Programación Visual tiene implícita el uso de 2 Metodologías a saber:

- Programación Orientada a Objetos
- Programación Orientada a Eventos

y a los lenguaje de Programación Visual se les considera como “Lenguajes de cuarta generación(4GL's)

La Programación Orientada a eventos tiene como característica el uso de componentes(objetos) que le dan al usuario la posibilidad de construir aplicaciones por medio de interfaces gráficas en base a la ocurrencia de “Eventos”. Los programas “Orientados a Eventos” son aplicaciones típicas del manejo del ambiente Windows que una vez que han iniciado su ejecución quedan en espera de las acciones de los usuarios que en este caso son llamadas “Eventos”.

En la Programación Orientada a Eventos cada componente u objeto tiene definidas “Propiedades” y es manipulado por “Métodos”, los eventos típicos son por ejemplo “un click sobre un botón”, y se activa un “Método” que puede ser para Abrir un Archivo, ó escribir sobre una caja de Texto ó activar una nueva ventana, etc..

Por definición los Lenguajes de cuarta generación(4GL's) son los lenguajes de "Programación Asistida" por medio de ayudantes ó "Wizard", estos lenguajes se han diseñado para facilitar la realización de muy variadas tareas, tanto de tipo científico, estadístico y de negocios.

A muchos de los Lenguajes 3GL les han sido implementadas estas nuevas características y han evolucionado como 4GL's, entre ellos podemos encontrar : Visual Basic, Visual C++, Visual J++, INFORMIX 4GL, Delphi(Pascal), etc...

Los 4GL's contienen facilidades para el acceso a Bases de datos tanto de tipo Relacional como también a Bases de datos Orientados a Objetos.



UANL

UNIVERSIDAD AUTÓNOMA DE NUEVO LEÓN



DIRECCIÓN GENERAL DE BIBLIOTECAS

5.4 Lenguajes Visuales (4GL's) de propósito general mas usados actualmente

Lenguaje Visual Basic

Es un producto de la casa de software Microsoft, es una herramienta RAD(Rapid Application Development), éste lenguaje fue el iniciador del paradigma del desarrollo visual para Windows y desde su aparición en 1991 ha demostrado facilidad de uso y de aprendizaje, sus características de desarrollo han mejorado a través de sus múltiples versiones incrementando de esta forma su poderío y eficiencia.

Su ambiente de programación es de tipo IDE y permite generar aplicaciones de negocios, con Bases de datos e Internet. Tiene capacidad para creación de controles ActiveX y un compilador de código nativo, pero no puede crear ejecutables por sí solo ya que requiere de por lo menos un archivo DLL para que funcione su ejecutable. Puede generar aplicaciones Cliente / servidor y cuenta con asistentes de ayuda Intellisense. Soporta la Programación Orientada a Objetos y puede ser usado como lenguaje de para otros productos de Microsoft Office(Word, Excell, Access).

Es compatible con proveedores nativos de OLE DB para operar con MS SQL Server y Oracle 7.3.3+. La versión mas actualizada a la fecha es Visual Basic 6.0 y se comercializa en tres presentaciones: Empresarial, Profesional y Para la Enseñanza, la versión para la enseñanza es la mas austera mientras que la Empresarial es la mas sofisticada.

Este lenguaje es una buena opción para el desarrollo de aplicaciones en Windows 95, Windows 98 , Windows NT, y Windows 2000.

5.4 Lenguajes Visuales (4GL's) de propósito general mas usados actualmente

Lenguaje Visual Basic

Es un producto de la casa de software **Microsoft**, es una herramienta RAD(Rapid Application Development), éste lenguaje fue el iniciador del paradigma del desarrollo visual para Windows y desde su aparición en 1991 ha demostrado facilidad de uso y de aprendizaje, sus características de desarrollo han mejorado a través de sus múltiples versiones incrementando de esta forma su poderío y eficiencia.

Su ambiente de programación es de tipo IDE y permite generar aplicaciones de negocios, con Bases de datos e Internet. Tiene capacidad para creación de controles ActiveX y un compilador de código nativo, pero no puede crear ejecutables por sí solo ya que requiere de por lo menos un archivo DLL para que funcione su ejecutable. Puede generar aplicaciones Cliente / servidor y cuenta con asistentes de ayuda Intellisense. Soporta la Programación Orientada a Objetos y puede ser usado como lenguaje de para otros productos de Microsoft Office(Word, Excell, Access).

Es compatible con proveedores nativos de OLE DB para operar con MS SQL Server y Oracle 7.3.3+. La versión mas actualizada a la fecha es Visual Basic 6.0 y se comercializa en tres presentaciones: Empresarial, Profesional y Para la Enseñanza, la versión para la enseñanza es la mas austera mientras que la Empresarial es la mas sofisticada.

Este lenguaje es una buena opción para el desarrollo de aplicaciones en Windows 95, Windows 98 , Windows NT, y Windows 2000.

Algunas características propias de Visual Basic 6.0

- **Compilador de código nativo de alto rendimiento construyendo aplicaciones Cliente/servidor optimizado con el compilador optimizado de Visual C++**
- **Acceso a todas las fuentes de datos corporativas mediante ODBC, OLE DB y Microsoft ActiveX Data Objects(ADO), mediante controladores incorporados para SQL Server, Oracle, Microsoft Access, ODBC y SNA Server**
- **Herramientas de Bases de datos Integradas**
- **Enlace de datos automático ajustando propiedades en la ventana de propiedades y lo conecta a cualquier fuente de datos**
- **Facilidad en la creación de componentes COM**
- **Diseñador Data Environment para generar en forma visual objetos reutilizables de tipo recordset enlazando múltiples fuentes de datos para adición y manipulación**
- **Diseñador Web Class para permitir la creación de componentes de Servidor accesibles desde cualquier navegador sobre cualquier plataforma**
- **Aplicaciones de 32 bits**

UNIVERSIDAD AUTÓNOMA DE NUEVO LEÓN

DIRECCIÓN GENERAL DE BIBLIOTECAS

Tabla 2 Algunas características entre versiones de Visual Basic 6.0

Características	Empresarial	Profesional	Enseñanza
Uso de Visual Basic en Aplicaciones de Microsoft Office(Word,Excell,Access)	si	si	si
Reducción en tiempo de desarrollo usando controles explorador Web, Check Box, etc..	si	si	no
Construcción de componentes en código nativo	si	si	no
Generación de Aplicaciones de Internet en cualquier plataforma	si	si	no
ADO(ActiveX Data Objetc) en Redes LAN	si	si	no
Microsoft Visual SourceSafe sistema para control de versiones para compartir y asegurar código en desarrollo para Windows y la Web	si	no	no
Microsoft Data Engine(MSDE) para Visual estudio 6.0 para soluciones compartidas compatibles con SQL 7.0 y migración a SQL Server	si	no	no
Data Objetc Wizard para simplificar la creación de componentes COM	si	no	no
Creación de fuentes de datos personalizadas	si	si	no
Creación de formularios e informes con drag & drop	si	si	no
Soporte nativo de procesadores Alpha para plataformas DEC Alpha	si	no	no
Depurador de sentencias SQL(TSQL) para depuración interactiva de procedimientos almacenados y Trigger's	si	no	no
Desarrollo de Aplicaciones de 32 bits	si	si	si
Manejo de punteros en forma natural	no	no	no
Soporte a aplicaciones de consola	no	no	no

Lenguaje Delphi

Es un producto de la casa de software Borland, es una herramienta RAD(Rapid Application Development) adecuada para crear aplicaciones distribuidas de Internet y de Bases de datos, la versión mas reciente es Delphi 5 y a ella le han sido añadidas una larga lista de funciones nuevas para desarrolladores individuales y para equipo de desarrollo.

Delphi permite generar aplicaciones de 32 bits y gestionar todo tipo de datos tanto multimedia como gráficos, sonidos y animaciones. Asimismo es posible manejar Bases de datos de múltiples formatos, así como descender al mínimo detalle del computador hasta acceder directamente a periféricos, dispositivos de entrada y de salida é incluso puntos de pantalla.

Delphi está basado en el lenguaje Object Pascal lo cual le permite operar en forma natural la programación Orientada a Objetos y también de punteros, éste lenguaje se distribuye en 3 presentaciones:

- Delphi Enterprise
- Delphi Professional
- Delphi Standard

Delphi Standard es una versión austera propia para la enseñanza de este lenguaje, mientras que la versión Enterprise es la versión mas sofisticada con la totalidad de características propias de Dephi 5.

Algunas características propias de Delphi 5

- Aplicaciones compiladas en código nativo, no requiere intérprete Run Time
- Robustez con manejador de excepciones
- Drives de Bases de datos nativas de alta velocidad, libres de regalías para : Paradox, Access, Fox Pro, Dbase, AS/400, DB2, MS SQL Sever, Oracle, Informix e InterBase
- Herramientas Cliente / servidor como: Monitor SQL, Explorador SQL, Asistente de migración a Bases de datos
- Cubo de decisión para soporte de decisiones y análisis de negocios
- Reporteador integrado, análisis gráficos y multidimensionales
- Implementación nativa de COM en estándares industriales
- Implementación nativa de CORBA
- Capacidad en Bases de datos escalables
- Diccionario de datos escalable para consistencia visual e integridad de datos
- Tecnología de compilación de paquetes para ejecutables mas pequeños
- Arquitectura Orientada a Objetos
- Salidas Internet HTML

Tabla 3 Algunas características entre versiones de Delphi 5

Característica	Enterprise	Professional	Standard
Desarrollo a 32 bits con soporte a Win API	si	si	si
Alto desempeño a 32 bits, con compilador de código nativo	si	si	si
Acceso completo a Win 32 API, ActiveX, Multihlo, OLE, OLE DB, DCOM,	si	si	si
Manuales con Documentación impresa	si	si	no
Documentación en línea	si	si	si
Diseñador con Modulo de datos con árbol y visor de diagrama de datos	si	si	no
Aplicaciones ejemplo para facilitar Aprendizaje	si	si	si
Asistente para Aplicaciones de consola	si	si	si
Aplicaciones de servicio a Windows NT	si	si	no
Editor de código AppBrowser con símbolos de hipervínculos	si	si	no
Depurador entre Procesos para pasar de un Proceso a otro	si	si	no
Depurador de procesos remotos para desarrollo distribuido	si	no	no
Browser Internet Explorer	si	si	si
Soporte a JPEG	si	si	no
Asistente Active Server Object para desarrollo en Servidores ASP	si	no	no
Drives para Access, Fox Pro, Paradox, Dbase y Redes LAN	si	si	no
Conectividad con ODBC	si	si	no

Lenguaje C++ Builder

Este lenguaje es producto de la casa de software Borland, es una herramienta RAD(Rapid Application Development), y permite a los desarrolladores la creación de aplicaciones C++ de alta velocidad, desde aplicaciones científicas, comerciales, de gestión de Bases de datos y de Internet, hasta controladores de dispositivos de entrada y salida de datos así como aplicaciones de consola(Ambiente MS DOS).

El lenguaje C++ Builder cubre las necesidades de los desarrolladores de Software base cuyas preocupaciones son la velocidad del compilador y el soporte a estándares establecidas. C++ Builder por naturaleza soporta la Programación Orientada a Objetos y la programación Estructurada, así como las características de la programación Cliente/servidor.

C++ Builder soporta los estándares de la industria de software como: Oracle 8i Data Base Server Oracle Corporation, Microsoft Foundation Classes(MFC) de Microsoft, Microsoft SQL Server 7 y MTS(Microsoft Transaction Server), las Bibliotecas de Inprise Corporation, Librerías de Objetos de Windows (OWL) y Librería de Componentes Visuales(VCL).

La versión mas actualizada a este momento es la versión: C++ Builder 5, y se distribuye en tres presentaciones a saber:

- C++ Builder 5 Enterprise
- C++ Builder 5 Professional
- C++ Builder 5 Standard

la presentación C++ Builder 5 Standard es la versión mas austera, mientras que la versión C++ Builder 5 Enterprise es la versión mas sofisticada.

Algunas Características propias de C++ Builder 5

- Programación para PC, Cliente / servidor y Multi-Nivel
- Tecnología de desarrollo Visual Bidireccional
- Aplicaciones para gestión de Bases de datos
- Desarrollo rápido de aplicaciones distribuidas con objetos estándar CORBA y COM, para plataformas Unix, Java y Windows
- Soporte a especificaciones estándar de C++ ANSI/ISO
- Entorno Integrado de Desarrollo(IDE), con herramientas visuales bidireccionales y su arquitectura de componentes, facilita todo tipo de programación C++ desde sistemas Back-End hasta desarrollos visuales Front-End, todos en una sola herramienta
- Herramientas Internet que incluyen ActiveForm para construir aplicaciones Web en C++ y WebBroker, así como también para desarrollar aplicaciones CGI, WinCGI, ISAPI y NASPI, y soporte a protocolos HTTP, FTP, SMTP, POP, NNTP, HTML y TCP/IP
- Kit de Desarrollo de servicios de Bases de datos Multi-Nivel(MIDAS) con MIDAS-2, para simplificar el desarrollo, integración y despliegue de clientes de Bases de datos distribuidas
- EZ-COM simplifica el desarrollo en C++ de Clientes COM y creación de controladores ActiveX en un paso para soporte a Data Binding

Tabla 4 Algunas características entre versiones de C++ Builder 5

Características	Enterprise	Professional	Standard
AppBrowser con hipervínculos de símbolo e historia de navegación	si	si	si
Class Explorer para el mapa de la fuente de clases y asistente de creación de miembros	si	si	no
Creación Visual de componentes	si	si	si
Compatible con expresiones regulares de PERL	si	si	si
Herencia y encadenamiento Visual de formas	si	si	si
Repositorio de Objetos para almacenar y reutilizar formas, módulos de datos y asistentes	si	si	si
Visor de dependencias de archivos del Project Manager	si	si	si
Exportar archivos Project Make	si	si	si
Asistente para aplicaciones Cliente en Win2000	si	si	si
Asistente para desarrollo de Applets	si	si	no
Asistente para Objetos COM	si	si	no
Asistente para Aplicaciones Windows NT	si	si	no
Asistente para Aplicaciones con MFC	si	si	no
Acceso en Aplicaciones de escritorio para Access, Fox Pro, Paradox y dBase	si	si	no
Conectividad completa con ODBC	si	si	no
Enlaces con SQL, InterBase, Oracle, SyBase, Informix, MS SQL Server y DB2	si	no	no
Herramientas de Desarrollo Integrado con Bases de datos	si	si	no
MIDAS Kit de desarrollo	si	no	no
Soporte a Oracle 8i avanzado	si	no	no
Soporte a aplicaciones de consola	si	si	si

Lenguaje Visual C++

Acerca del Lenguaje C y de C++ desde siempre se ha dicho que es difícil, críptico y que no es posible dominarlo del todo, el manejo de variables puntero siempre ha sido molesto y en ello es muy fácil equivocarse, pero el manejo de estas variables es quizá la herramienta más poderosa en el desarrollo de aplicaciones realmente de corte profesional. Desde su aparición el Lenguaje C++ ha sido dirigido hacia la aplicación de la Programación Orientada a Objetos, y al agregarle las técnicas de la Programación Orientada a Eventos se ha generado el conocido Visual C++, este ha sido utilizado para el desarrollo de programas como Word, Excel, PowerPoint, Fox Pro, Netscape navigator, Internet Explorer, Visual Basic, Paint Shop Pro...y otras más. Asimismo es usado cuando un lenguaje por sus características no puede con el desarrollo de algo en particular (ejemplo DLL's).

Visual C++ es un producto de la casa de software Microsoft, y para su aprendizaje es necesario también aprender las funciones MFC (Microsoft Foundation Class), el cual es un producto "empacado" junto con Visual C++ y aunque esto incrementa el número de cosas por aprender al final permiten la creación de aplicaciones empleando "objetos" que ya están probados en su funcionamiento. Se puede decir que Visual C++ es un lenguaje tan difícil como cualquier otro aunque requiere de más cuidado de parte del programador, pues el manejo de memoria no es automático y esta es la fuente de la mayoría de los errores que se cometen, asimismo es preciso que el programador adquiera mucha práctica para el aprendizaje de los detalles de la MFC.

La versión más actual a este momento es Visual C++ 6.0, y es distribuido bajo tres presentaciones a saber:

- Visual C++ Enterprise Edition
- Visual C++ Professional Edition
- Visual C++ Standard Edition

Algunas Características propias de Visual C++ 6.0

- Aumenta la velocidad del desarrollo de aplicaciones y simplifica la codificación con la tecnología IntelliSense, el uso de wizard's permiten eliminar la memorización de sintaxis complejas, así como parámetros y propiedades de objeto.
- Los desarrolladores pueden editar código al tiempo que realizan la depuración, sin tener que salir de la sesión, asimismo reestructurar, reiniciar el depurador y regresar a la aplicación donde se generó un problema.
- Integra fácilmente y sin problemas la funcionalidad de Microsoft Word, Microsoft Excel, así como otras aplicaciones con Active Document Containment (Contenido de Documento Activo), los usuarios obtienen las funciones y la capacidad total a que están acostumbrados, mientras que el desarrollador escribe menos código mediante el uso de wizard's.
- Hace que la recompilación sea mas rápida, el tiempo de recompilación se ha reducido en un 30%.

- Dispone de mas de 150 Clases escritas por profesionales, incluidas en la biblioteca (Library) Microsoft Foundation Classes and Templates(MFC&T).
- Soporta las especificaciones ANSI/ISO.
- Permite generar aplicaciones Web basadas en Multimedia altamente interactivas con Dynamic HTML.
- Permite crear controles ActiveX y objetos COM.
- Permite depurar interactivamente procedimientos almacenados en SQL Server.
- Simplifica grandemente el acceso a cualquier puente relacional ó no relacional, correo electrónico y sistemas de archivo, texto y gráficos, objetos de negocio del cliente y mas, con los wizard OLE DB.
- Soporte a aplicaciones de consola

Lenguaje Cobol

El Cobol es un Lenguaje de programación que ha sido capaz de permanecer activo desde el inicio de la informática hasta nuestros días, y por lo visto no tiene la menor intención de abandonarnos. Hablar de Cobol es remontarnos más de 40 años atrás, ya que por esa época cada computadora era construida, programada en su propio Sistema Operativo y sus propios Lenguajes de programación, era la época de la guerra fría entre los diversos fabricantes de computadoras y no existía la menor compatibilidad entre computadoras de diferentes marcas fabricantes.

Cobol nació por el deseo de desarrollar un lenguaje de programación que pudiera “correr” en cualquier computadora de cualquier marca, y fue en Mayo de 1959 cuando en los Estados Unidos de América organizó una comisión fabricantes de computadoras, empresas privadas y representantes del gobierno, y la llamó CODASYL (Conference On Data System Languages). Las reuniones de esta comisión dieron origen al Lenguaje Cobol y a su primera versión de le denominó Cobol-60, el lenguaje siguió evolucionando con una nueva versión en 1968 y otra en 1974 las cuales fueron conocidas como Cobol-Ansi.

DIRECCIÓN GENERAL DE BIBLIOTECAS

Cobol significa “Lenguaje Común Orientado a los Negocios” por lo que se comprende que su misión es ser utilizado en la gestión y administración de los negocios, en contraste con otros lenguajes Cobol no fue concebido para cálculos matemáticos o científicos ya que de hecho solo dispone operadores para realizar los cálculos más elementales, su empleo es más apropiado para el proceso de datos en aplicaciones comerciales y la utilización de grandes Bases de datos.

Cobol es un lenguaje Multiplataforma y puede operar en Windows, Unix, MS-DOS, Linux, OS/400, S36, S34, VMS, Netware, Solaris etc., Actualmente Cobol puede ser usado en ambiente gráfico y también en aplicaciones con Internet, integrar las aplicaciones de Cobol en el mundo Web es una realidad.

Algunas de las características del Lenguaje Cobol

- Cobol tiene futuro. La ANSI pondrá en marcha una versión actualizada de Cobol adecuada a los grandes adelantos de en tecnología.
- Cobol representa la experiencia. El 80% de los sistemas desarrollados en el mundo operan bajo este lenguaje.
- Cobol es sólido. Un gran numero de desarrolladores de software han desarrollado aplicaciones para ayudas en comprobación, análisis de aplicaciones, asistencia en la producción y reutilización de código.
- Cobol puede abarcar un amplio abanico de necesidades de procesamiento de datos. Las aplicaciones Cobol son ricas en posibilidades y funcionan tan bien al realizar las funciones vitales del negocio que los usuarios se resisten a prescindir de ellas.
- Cobol cumple su función de Soporte a aplicaciones empresariales.
- Cobol es fácil de aprender. Aún personas con pocos conocimientos técnicos han aprendido Cobol en muy poco tiempo, y se enseña en todo el mundo.
- Cobol es fácil de mantener. Los programas Cobol son fáciles de desarrollar y por su estructura de sintaxis es fácil de dar mantenimiento a sus aplicaciones.
- Cobol puede operar en múltiples plataformas.

Actualmente varias casas de software han desarrollado nuevas versiones de Cobol adecuándolo a las Metodologías actuales de la Programación Visual, entre las que podemos mencionar son:

- COBOL-WOW(RM/Cobol)
- ACUCOBOL-GT
- NET EXPRESS(MICROFOCUS)
- OBJECTIVE COBOL
- FUJITSU POWER-COBOL

Bill Gates dijo al público en la Conferencia de Desarrolladores Profesionales reunida en Orlando Florida el pasado 12 de Julio del 2000 que Microsoft había escogido el Cobol de Fujitsu para apoyar sus planes en el desarrollo de la próxima Tecnología Microsoft.NET, “Hay todavía un nivel muy alto de uso de código Cobol, y por esto nosotros pensamos en el Cobol como una herramienta líder en el mercado”.

Objective Cobol

Es un entorno visual de programación orientada a objetos, destinado al desarrollo rápido (RAD) de aplicaciones de uso general para Windows 95, 98 y NT. Con Objective Cobol se pueden desarrollar aplicaciones muy eficaces para Windows con un mínimo de código manual.

Objective Cobol proporciona una biblioteca de componentes reutilizables y una serie de herramientas de diseño RAD, entre ellas plantillas de aplicación y de fichas, además de Asistentes para la programación. Con estas herramientas y el compilador de código nativo de 32 bits se pueden crear prototipos de gran calidad, convirtiéndolos en aplicaciones robustas para satisfacer las necesidades de las empresas. Sus principales características son:

- Entorno de desarrollo integrado
- Diseño mediante el método de arrastrar y soltar
- Herramientas bidireccionales
- Compilador de código nativo
- Conectividad a las Bases de datos

NETEXPRESS-MICROFOCUS

Cobol NetExpress es la evolución a 32 bits del Cobol Microfocus para MS-DOS, Windows, y OS/2, Tiene un entorno de desarrollo IDE que ofrece un completo conjunto de herramientas y asistentes para desarrollar aplicaciones.

Las eficaces funciones de gestión de proyectos simplifican el ciclo de edición, compilación y depuración, mientras que el editor que ha sido adaptado a las necesidades del programador le simplifica el mantenimiento necesario del código.

NetExpress incluye un compilador de última generación que potencia los tradicionales puntos fuertes de Cobol, algunos de ellos son:

- Generación de código nativo de alto rendimiento
- Operar con múltiples dialectos Cobol, facilitando la reutilización de código
- Obtiene escalabilidad desde PC's independientes hasta Sistemas Unix
- Tiene soporte de desarrollo y depuración en Object Cobol y e-Cobol
- Soporte a bibliotecas básicas de clases y con una extensa colección de clases, incluyendo la biblioteca completa de clases Win32 GUI
- Depuración remota de aplicaciones Cobol ejecutándose en otras plataformas
- Soporte Animación JIT(Just-in-Time) si la aplicación falla durante la ejecución
- Diseñador de formularios(Form Designer)
- Edición de páginas y formularios HTML
- Asistente para Scripts
- Aplicaciones Web de gran rendimiento
- Servidor Web local para el desarrollo
- Soporte a Objetos COM

ACUCOBOL-GT

Con Acucobol-GT se ha logrado combinar la tecnología de interfaces gráficas con la portabilidad de los sistemas abiertos, en este lenguaje solo se compilan los programas una sola vez para después ser ejecutados en cualquier plataforma. Algunas de sus características son:

- Interfaz grafica nativa permitiendo programar pantallas gráficas incluyendo ventanas flotantes y controles gráficos, las cuales pueden ser definidas en la Screen Section
- Formatos de Tipos de Archivos, mediante Vision el cual es el formato de archivos indexados propios de Acucobol-GT, mediante el cual le es permitido moverse entre diferentes plataformas sin necesidad de conversión de datos
- Administración de Transacciones, obteniendo capacidad de trabajo en aplicaciones que así lo requieran
- Ambiente de desarrollo IDE haciendo que el desarrollo sea fácil y productivo
- Un administrador de proyectos para organizar, desarrollar y mantener los archivos de cada proyecto siendo éste el centro del control organizativo
- Screen Painter, como herramienta visual para crear y diseñar pantallas gráficas interactivamente(100% windows)
- Un generador de código para facilitar al programador lo necesario para generar y manejar el código fuente

5.5 Requerimientos de Recursos y Plataformas de Operación

Tabla 5 Requerimientos de Recursos para el Lenguaje Visual Basic

Requerimientos	Empresarial	Profesional	Aprendizaje
Sistema Operativo	NT, W95 ó +	NT, W95 ó +	NT, W95 ó +
Procesador	Pentium /90 ó +	Pentium/90 ó +	486DX/66 ó +
Memoria RAM	32 MB recom.	32 MB recom.	24 MB recom.
Tipo Monitor	SVGA recom.	SVGA recom.	SVGA recom.
Mouse	MS ó compatible	MS ó compatible	MS ó compatible
Área en D.D.	707 MB aprox.	666 MB aprox.	537 MB aprox.
Internet Explorer	completa	completa	Típica
SQL Server, SNA Server	195 MB adición	180 MB adición	-....-
CD-ROM	si	si	si

Tabla 6 Requerimientos de Recursos para el Lenguaje C++ Builder

Requerimientos	Enterprise	Professional	Standard
Sistema Operativo	NT, W95 ó + Linux	NT, W95 ó + Linux	NT, W95 ó + Linux
Procesador	Pentium/90 ó +	Pentium/90 ó +	Pentium/90 ó +
Memoria RAM	32 MB	32 MB	32 MB
Tipo Monitor	VGA/SVGA	VGA/SVGA	VGA/SVGA
Mouse	compatible	compatible	Compatible
Area en D.D.	388 MB	360 MB	185 MB
Internet Explorer	completa	completa	típica
SQL Server, SNA Server	opcional	opcional	-....-
CD-ROM	si	si	si

Tabla 7 Requerimientos de Recursos para el Lenguaje Delphi

Requerimientos	Enterprise	Professional	Standard
Sistema Operativo	NT, W95 ó + Linux	NT, W95 ó + Linux	NT, W95 ó + Linux
Procesador	Pentium/90 ó +	Pentium/90 ó +	Pentium/90 ó +
Memoria RAM	32 MB	32 MB	32 MB
Tipo Monitor	VGA/SVGA	VGA/SVGA	VGA/SVGA
Mouse	Compatible	compatible	Compatible
Area en D.D.	230 MB	185 MB	120 MB
Internet Explorer	Completa	completa	tipica
SQL Server, SNA Server	Opcional	opcional	-....-
CD-ROM	Si	si	si

Tabla 8 Requerimientos de Recursos para el Lenguaje Visual C++

Requerimientos	Enterprise	Professional	Standard
Sistema Operativo	NT, W95 ó +	NT, W95 ó +	NT, W95 ó +
Procesador	Pentium/90 ó +	Pentium/90 ó +	Pentium/90 ó +
Memoria RAM	32 MB recom.	32 MB recom.	32 MB recom.
Tipo Monitor	VGA/SVGA	VGA/SVGA	VGA/SVGA
Mouse	MS ó compatible	MS ó compatible	MS ó compatible
Area en D.D.	977 MB	947 MB	857 MB
Internet Explorer	completa	completa	tipica
SQL Server, SNA Server	100 MB adicion	100 MB adicion	-....-
CD-ROM	si	si	si

5.6 Criterios de selección para el primer Lenguaje de Programación

La decisión acerca de elegir un lenguaje de programación implica considerar una serie de factores tanto de aspecto técnico como también de los objetivos cualitativos que se persiguen en la utilización de esta herramienta. Las consideraciones de aspecto técnico deben tomarse en cuenta para implantar la plataforma y recursos computacionales necesarios para la operación adecuada de la misma.

Las consideraciones de aspecto técnico de los lenguajes candidatos a elección ya han sido expuestas en la sección anterior, por lo que ahora nos centraremos en los objetivos cualitativos. Para la elección de un Lenguaje que soporte la enseñanza de las técnicas actuales de programación de computadoras en el Instituto Tecnológico de Nuevo León, en la carrera de Ingeniero en Sistemas Computacionales, se han considerado que incluya los siguientes aspectos:

- Facilidad para su Aprendizaje
- Soporte multiplataforma
- Programación Visual
- Soporte a Internet/Web
- Uso en la Industria
- Soporte a diferentes asignaturas académicas
- Soporte Aplicaciones Cliente/Servidor

cada una de las cuales se describen a continuación.

Facilidad para su Aprendizaje

Tomando en cuenta que con el lenguaje elegido la mayoría de los alumnos de la carrera de Ingeniería en Sistemas Computacionales tendrán su primera experiencia en la programación de computadoras, es conveniente que incluya en si mismo todos los elementos necesarios para el desarrollo de las aplicaciones.

Debido a que el código básico de un 4GL tiene su origen en su versión 3GL, es conveniente que para su enseñanza proporcione la opción de funcionamiento como 3GL en ambiente MS-DOS, precisamente para la enseñanza del código básico y posteriormente en temas subsecuentes tratar lo relativo a la Programación Visual.

Para el desarrollo de aplicaciones visuales básicas y avanzadas es importante lo completo de su ambiente IDE para facilitar la enseñanza de los componentes visuales.

Soporte Multiplataforma

Esta característica es importante porque puede dar la facilidad de mantener la inversión en los lenguajes y la operación de las aplicaciones realizadas en dado el caso de que se presentara la necesidad de hacer un cambio de plataforma(Sistema Operativo).

Programación Visual

Esta característica es requerida para introducir a los alumnos en las practicas de las Metodologías actuales de la programación de computadoras.

Soporte a Internet / Web

Dado que Internet es de amplio uso tanto documental como también de negocios es de uso corriente el desarrollo de aplicaciones(programas) que utilizan las facilidades de Internet para el proceso de transacciones comerciales bajo los conceptos de Internet/Intranet; Por lo que es necesario que el lenguaje electo proporcione dichas facilidades.

Uso en la Industria

De los lenguajes de desarrollo que usan en las empresas y que se amoldan al campo de la enseñanza son los lenguajes 4GL's de propósito general. Para definir esta característica se hizo un pequeño estudio con algunas empresas de la localidad, tanto a empresas grandes como empresas pequeñas y a empresas prestadoras de servicios computacionales, por confidencialidad de la información no me es posible mencionarlas, pero los datos recabados dieron lugar a los siguientes resultados:

COBOL Lo usan el 50% de las empresas entrevistadas

C/C++ Lo usan el 20% de las empresas entrevistadas

Visual C++ Lo usan el 48% de las empresas entrevistadas

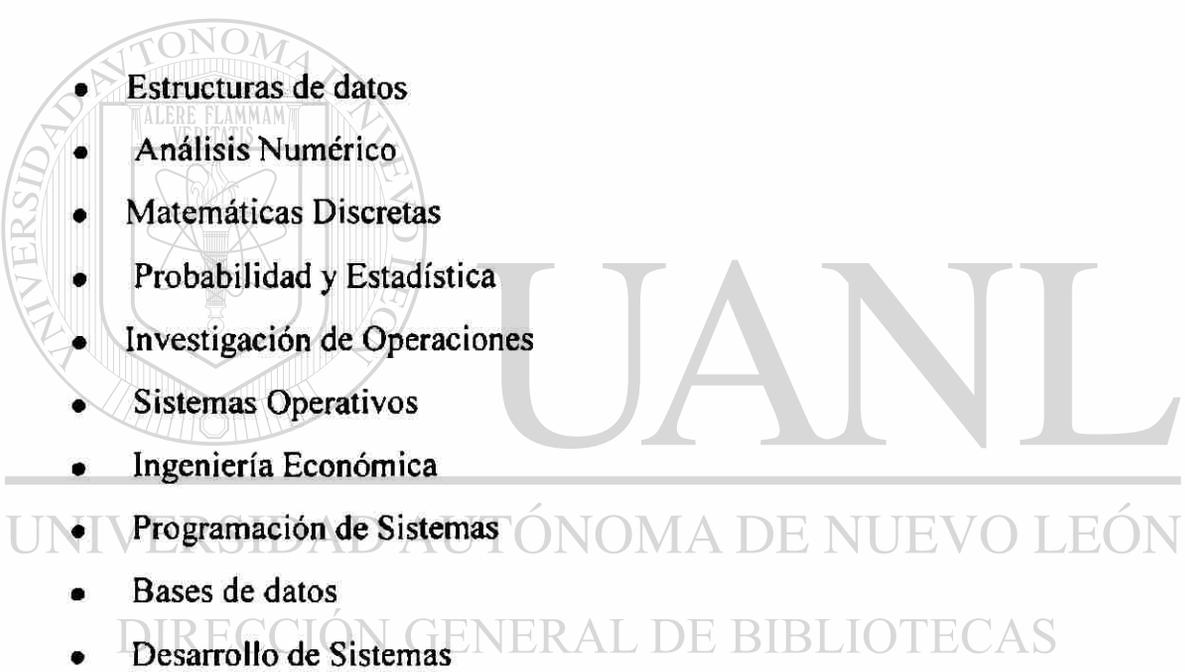
Visual Basic Lo usan el 80% de las empresas entrevistadas

Delphi Lo usan el 15 % de las empresas entrevistadas

C++ Builder Lenguaje Visual de reciente Introducción al mercado computacional

Soporte a diferentes asignaturas académicas

El plan de estudios de la carrera de Ingeniería en Sistemas Computacionales incluye varias asignaturas en las cuales los conceptos en ellas tratados son susceptibles de aplicar en ellos prácticas por medio de la programación de computadoras para reafirmar cada uno de los conceptos, y por tanto el primer lenguaje de programación aprendido por los alumnos debe proporcionarles las herramientas para el desarrollo de esas prácticas. Algunas de las asignaturas a soportar son:

- 
- Estructuras de datos
 - Análisis Numérico
 - Matemáticas Discretas
 - Probabilidad y Estadística
 - Investigación de Operaciones
 - Sistemas Operativos
 - Ingeniería Económica
-
- Programación de Sistemas
 - Bases de datos
 - Desarrollo de Sistemas

Soporte al Modelo Cliente/Servidor

El soporte al modelo Cliente/Servidor es una característica de los lenguajes de la actualidad, ya que en el desarrollo de aplicaciones administrativas y de negocios se requiere del uso de ésta técnica de programación.

Tabla 9 Cuadro comparativo de cumplimiento a los Criterios de Selección

Criterio de Selección	Cobol	Visual Basic	Visual C++	Dephi	C++ Builder
Facilidad de aprendizaje	si	si	No	si	si
Soporte en aplicaciones de Consola	no	no	Si	si	si
Soporte multiplataforma	si	no	Si	si	si
Programación Visual	si	si	Si	si	si
Soporte Internet/ Web	si	si	Si	si	si
Uso en la Industria	si	si	Si	si	no
Soporte al Modelo Cliente/Servidor	si	si	Si	si	si
Soporte a asignaturas académicas	---	---	---	---	---
Estructuras de datos	no	no	Si	si	si
Análisis Numérico	no	si	Si	si	si
Matemáticas Discretas	no	no	Si	si	si
Probabilidad y Estadística	no	si	Si	si	si
Investigación de Operaciones	no	si	Si	si	si
Sistemas Operativos	no	no	Si	si	si
Ingeniería Económica	si	si	Si	si	si
Programación de Sistemas	no	no	Si	si	si
Bases de datos	si	si	Si	si	si
Desarrollo de Sistemas	si	si	Si	Si	si

De acuerdo a los datos que se muestran en la tabla los lenguajes que pueden ser considerados como candidatos a ser el primer lenguaje de programación a impartir a los alumnos de Ingeniería en Sistemas Computacionales son: Visual C++, Delphi y C++ Builder; y los criterios que marcan la diferencia son : Facilidad de aprendizaje y uso en la industria.

5.7 Selección del primer lenguaje de programación

La orientación de Visual C++ (de la empresa Microsoft) es hacia el desarrollo de software de base, así como también para la generación de rutinas muy especializadas en programas que así lo requieran, como el desarrollo de aplicaciones comerciales ó de negocios en otros lenguajes como Visual Basic(también de la empresa Microsoft) en el cual no pueden realizarse las rutinas muy especializadas, de tal forma que Visual C++ es usado también como un complemento a las tareas que Visual Basic no puede realizar, como componentes ActiveX y DLL's, aplica la técnica de programación visual y la programación orientada a objetos.

El lenguaje Visual C++ aunque es una herramienta muy poderosa su programación tiene varias complejidades, entre ellas podemos mencionar el uso del SDK que es un complejo conjunto de funciones que añade numerosas definiciones de tipos de datos nuevos aún para programadores expertos que ya conocen el lenguaje C/C++, asimismo el uso la MFC(Microsoft Foundation Classes) le añade complejidad al desarrollo de aplicaciones al imponer una serie de normas de programación con las que debe regirse el programador, el programador además de controlar el código de C/C++ también debe conocer perfectamente la manera de usar las clases de MFC para poder utilizar su potencia en el lenguaje.

Dado lo anterior podemos concluir que el lenguaje Visual C++ no cumple el requisito de facilidad de aprendizaje, por lo que no es factible usarlo como el primer lenguaje de programación en alumnos que recién inician su andar por este camino y no tienen conocimientos previos de estos temas.

El lenguaje Delphi (de la empresa Borland), es también una poderosa herramienta de programación, tiene las características de la programación RAD(Rapid Application Development), aplica la programación visual y la programación orientada a objetos, es clasificado como un lenguaje de cuarta generación(4GL), y mediante este lenguaje se pueden desarrollar aplicaciones de todo tipo: científica, de ingeniería, comercial, de negocios, etc., permite el desarrollo de componentes ActiveX y DLL's, permite la interacción de sus aplicaciones en ambiente Internet/Intranet, en su mas reciente versión(Delphi 6) se pueden desarrollar aplicaciones para e-business & e-commerce.

Delphi está basado en un lenguaje de gran trayectoria como es el lenguaje Pascal, en su versión actualizada de Object Pascal, opera en un ambiente de programación denominado IDE(Integrated Development Environment), en el cual se incluyen herramientas que facilitan en gran manera la enseñanza del lenguaje, tanto desde las bases de Object Pascal con aplicaciones de consola similar al ambiente D.O.S. como en aplicaciones en ambiente Windows.

En el uso de los conceptos de la programación orientada a objetos Delphi facilita en gran manera su enseñanza y utilización a través de una biblioteca de componentes visuales denominada VCL(Visual Component Library), la cual contiene en un carpetero debidamente organizado un gran numero de objetos y clases en modo gráfico, disponibles para ser usados en las aplicaciones, el programador solo se concreta a visualizar los objetivos de su aplicación y coloca dicho código en el programa donde Delphi le indica que debe colocarlo, mediante esta forma de trabajo es extremadamente fácil el diseño y construcción de aplicaciones en ambiente visual ó gráfico.

Delphi cumple con todos los criterios que fueron definidos para la selección del primer lenguaje de programación: facilidad de aprendizaje, soporte en aplicaciones de consola, soporte multiplataforma, programación visual, soporte Internet/Web, uso en la industria, soporte al modelo cliente/servidor, soporte a diversas asignaturas académicas, por lo que es un fuerte candidato a ser elegido como el primer lenguaje de programación en la carrera de Ingeniero en Sistemas Computacionales.

El lenguaje C++ Builder(también de la empresa Borland), en una versión del lenguaje C/C++ con la misma potencia de desarrollo que Visual C++ y con su misma orientación hacia el desarrollo de software de base, en su operación es muy semejante a Delphi, ya que también tiene las características de la programación RAD(Rapid Application Development), y opera también en el mismo ambiente IDE(Integrated Development Environment), es también considerado un lenguaje de cuarta generación(4GL), permite el desarrollo de componentes ActiveX y DLL's, permite generar aplicaciones para ambiente Internet/Intranet, aplica la técnica de programación visual y la programación orientada a objetos.

Al igual que Delphi también utiliza la VCL(Biblioteca de componentes visuales) para la generación de aplicaciones en ambiente grafico(la VCL que utiliza C++ Builder es la misma que utiliza Delphi, y está desarrollada en Object Pascal), lo cual facilita su enseñanza y uso en las aplicaciones, comparativamente la diferencia entre Delphi y C++ Builder es solo la sintaxis del código, ya que Delphi genera código en Object Pascal, y C++ Builder genera código de C/C++, otra de sus diferencias es que Delphi produce programas ejecutable mas pequeños que C++ Builder. Aprender Delphi ó C++ Builder es como aprender 1.5 lenguajes, ya que solo con aprender la sintaxis del otro lenguaje se aprenderán dos lenguajes, esto puede ser una ventaja competitiva para los alumnos. ®

DIRECCIÓN GENERAL DE BIBLIOTECAS

La gran semejanza entre Delphi y C++ Builder a nuestro juicio empata entre ellos la selección del primer lenguaje de programación, entre ellos solo existe la diferencia del criterio "uso en la industria" ya que C++ Builder como la mas reciente herramienta desarrollada por Borland está en proceso de tener presencia en el área de desarrollo computacional en los diferentes sectores productivos y de servicios.

Digno de mención es indicar que en el ámbito educativo como herramienta para la enseñanza de la programación de computadoras C++ Builder ya ha logrado tener presencia, puesto que existe un gran numero de planteles educativos que lo usan para tal fin.

Un aspecto adicional a considerar es que el lenguaje C/C++ en forma práctica ha servido de guía para la construcción de varios lenguajes usados para el desarrollo de aplicaciones Web, tales como: Java, JavaScript, Perl y PHP, los cuales son lenguajes ampliamente usados en ambiente Internet, y las sintaxis de sus códigos tienen rasgos muy marcados con respecto al lenguaje C/C++ por lo que conviene primero aprender lo concerniente a C/C++ para facilitar posteriormente el aprendizaje de la programación de aplicaciones en ambiente Internet.

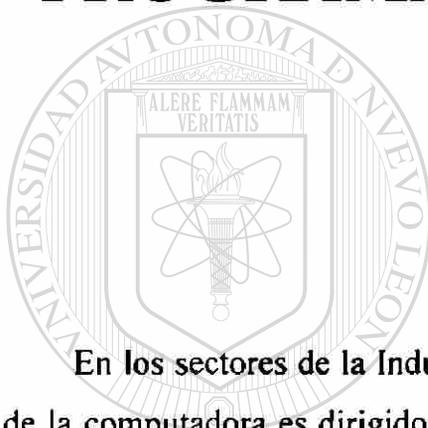
Dado lo anterior se presenta el aspecto que puede marcar la diferencia entre Delphi y C++ Builder en cuanto a la selección del primer lenguaje de programación, y el criterio aconseja el uso de C++ Builder, ya que el aprender esta herramienta se cumple:

- El objetivo del Instituto Tecnológico de Nuevo León de que sus alumnos dispongan del conocimiento de técnicas y herramientas de actualidad, para ser factores de respuesta ante las necesidades de los diferentes sectores productivos y de servicios de nuestro país
- El aprendizaje de dichas herramientas sea soporte del aprendizaje de futuras herramientas y técnicas

de esta forma después del análisis expuesto podemos recomendar al lenguaje C++ Builder como el primer lenguaje de programación para los alumnos de la carrera de Ingeniería en Sistemas Computacionales del Instituto Tecnológico de Nuevo León.

6

EL SEGUNDO LENGUAJE DE PROGRAMACION Y LAS BASES DE DATOS



En los sectores de la Industria, el Comercio, de Gobierno, y de los Servicios el uso de la computadora es dirigido especialmente al almacenamiento masivo de datos y a su manipulación, ya que los datos almacenados pueden generar (a través de un correcto Procesamiento) información útil para el control administrativo, así como para la toma de decisiones.

DIRECCIÓN GENERAL DE BIBLIOTECAS

Para lograr estas facilidades se han diseñado diversos dispositivos de Hardware y mecanismos de Software que han apoyado esta necesidad, en ello los Ingenieros de Hardware y los Ingenieros de Software han unido sus esfuerzos para lograrlo.

Actualmente la mayoría de los negocios(pequeños, medianos y grandes) se apoyan en la tecnología computacional de las Bases de datos para generar información que les ayuden a afianzar sus oportunidades de negocio, así como facilitar el control de las operaciones de los mismos.

6.1 Antecedentes de las Bases de datos

El procesamiento de datos por computadora ha pasado por dos etapas importantes que son:

- Sistema de Procesamiento de Archivos
- Sistema de Procesamiento de Bases de Datos

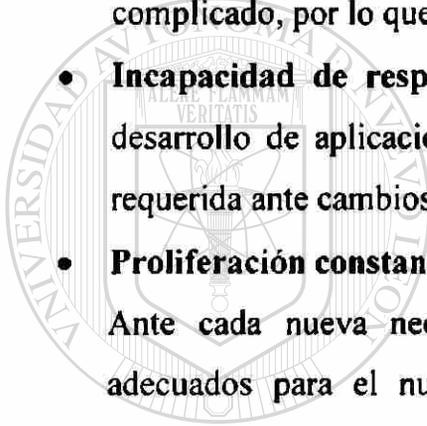
la etapa del Sistema de Procesamiento de Archivos inició con la primera generación de computadoras (cuando aparecen: la UNIVAC I, la IBM 701 y la IBM 1401), usándose con ellas las tarjetas perforadas como dispositivos de almacenamiento de los datos.

Con ello se inició el soporte a los negocios que necesitaban tener controles para sus procesos administrativos tales como : Costeo de sus productos, Pagos a sus trabajadores, la Contabilidad, Control de compras, Control de ventas, etc., ya que este método ofrecía una sistematización de los procesos manuales.

Aún y cuando esta forma de procesamiento mejoró respecto a la forma manual, computacionalmente tenía grandes deficiencias, tales como:

- **No integración de los datos.** Los datos se distribuían en diferentes archivos, de forma que al procesarlos era necesario disponer de todos esos archivos, esto complicaba su actualización y su mantenimiento.
- **Duplicidad de datos.** Las claves que referencian a los archivos deben estar presentes en todos ellos, por lo que esto genera redundancia (duplicidad) de datos.
- **Programas dependientes de los formatos de los archivos.** Cuando era necesario modificar la estructura de un archivo, era obligatorio “recompilar” todos los programas que accedían al archivo, de lo contrario los procesos “abortaban”.

- **Incompatibilidad entre lenguajes de programación.** El acceso a los datos de los archivos solo era posible con programas del mismo lenguaje en que el archivo fue creado.
- **El usuario no tiene la visión del almacenamiento de los datos que opera.** Le era difícil al usuario prever posibles errores en los datos, ya que no tenía acceso a su almacenamiento.
- **Desarrollo de programas lento y costoso.** La generación de programas de aplicación requería de expertos programadores con altas remuneraciones, como los lenguajes de desarrollo eran de tercera generación el manejo de archivos era complicado, por lo que el desarrollo de los programas se volvía lento.
- **Incapacidad de responder rápidamente a los cambios.** La lentitud del desarrollo de aplicaciones complicaba la generación de información oportuna, requerida ante cambios imprevistos en las transacciones del negocio.
- **Proliferación constante de nuevos archivos.** Ante cada nueva necesidad se generaba un nuevo archivo con los datos adecuados para el nuevo requerimiento, generándose de esta forma nuevas necesidades de dispositivos de almacenamiento con mayor capacidad.



UANL

UNIVERSIDAD AUTÓNOMA DE NUEVO LEÓN



DIRECCIÓN GENERAL DE BIBLIOTECAS



Cientes

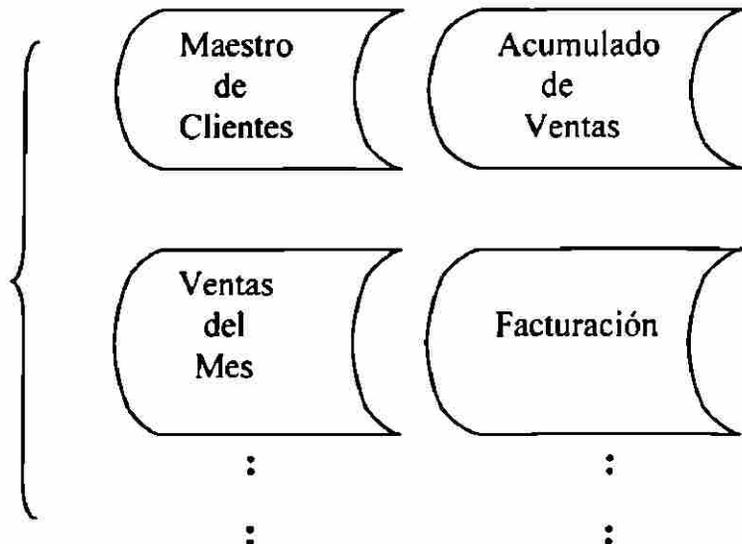


Fig. 23 Datos distribuidos en un Sistema de Procesamiento de Archivos

Las limitaciones del Sistema de Procesamiento de Archivos hicieron que los Ingenieros de Software y los Ingenieros de Hardware tomaran en reto de diseñar sistemas lo suficientemente fuertes para atender varios usuarios a la vez y con una rapidez lo suficiente para soportar el nivel de operaciones (ó transacciones), y además que fuera transparente al usuario.

Se diseñaron nuevas formas de almacenar, controlar, respaldar, y proteger a los datos, generándose así la tecnología de las Bases de datos, en la década de 1970 había nacido esta nueva tecnología computacional que resolvía la problemática del Sistema de Procesamiento de Archivos.

Los Administradores de Bases de datos se dieron cuenta que los usuarios debían tener acceso a los datos para actualizarlos al momento de una operación, sin embargo debido a que la explotación de la Base de datos aún se programaba con lenguajes de tercera generación no fue posible tener esa facilidad.

En 1970 E. F. Codd publicó su memorable artículo "A Relational Model of Data for Large Shared Databanks", donde aplicó conceptos del Algebra Relacional a la problemática del almacenamiento masivo de datos, dicho artículo generó un movimiento en la comunidad diseñadora de Bases de datos, y generó los conceptos de la Base de datos Relacional.

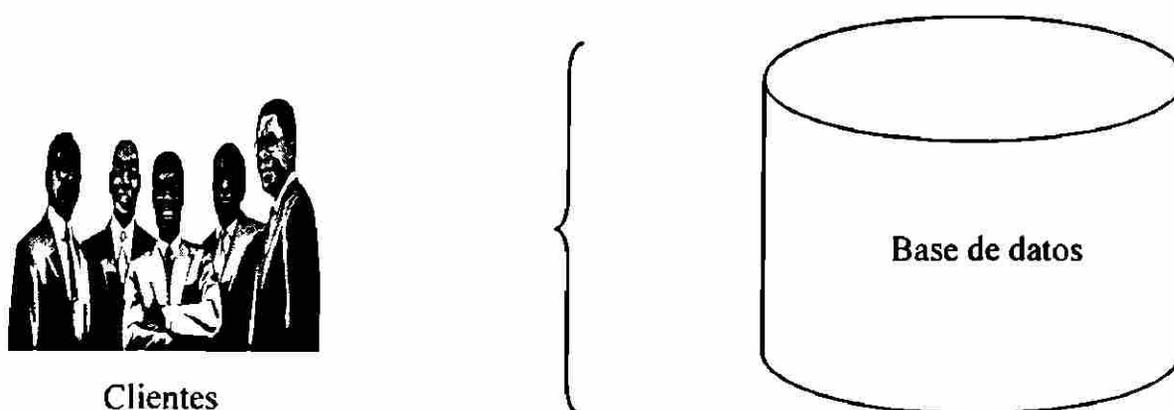


Fig. 24 Datos integrados en un Sistema de Base de datos

6.2 Ventajas y desventajas de las Bases de datos

El uso de la tecnología y métodos de las Bases de datos ha permitido un avance en el almacenamiento y tratamiento de los datos, dejando en el pasado el Sistema de Procesamiento de Archivos y su problemática, ahora las Bases de datos proporcionan las siguientes ventajas:

- **Integridad de los datos.** Con las Bases de datos se elimina el aislamiento de los datos, los cuales se almacenan en un solo gran esquema organizado en subesquemas, y el DBMS(Data Base Managment System) opera los accesos necesarios.
- **Se minimiza la duplicación de los datos.** La duplicidad no se elimina completamente, aunque si se reduce notablemente, lo cual fortalece la integridad de los datos.
- **Independencia Programas-datos.** Los formatos de los registros se almacenan en la misma Base de datos, y son accedidos por el DBMS no por los programas de aplicación, esto permite que los programas no necesiten el formato de los registros de los archivos que procesan, los programas solo contienen la longitud y el tipo de dato del dato que necesitan, el propio DBMS localiza el dato requerido. De esta forma si existiesen cambios de formato esto sería transparente para los Programas de aplicación .
- **Control del usuario.** El usuario tiene a la vista la representación de los datos que normalmente opera.
- **Preparación para futuras aplicaciones.** El diseño del esquema y los subesquemas de la Base de datos permiten una visión de presente y futuro del uso de los datos, teniendo de esta manera una preparación a las necesidades futuras.
- **Datos compartidos e información oportuna.**
Los datos almacenados en la Base de datos pueden ser accedidos por múltiples usuarios a la vez, sus transacciones son controladas por la misma Base de datos, de esta forma asegura que los datos en ella son verídicos.

- **Seguridad de los datos.**

Las Bases de Bases de datos incorporan en su funcionamiento mecanismos que permiten el acceso a datos solo a usuarios calificados mediante identificación, restringiendo el acceso a usuarios que no puedan identificarse plenamente, y aún así cada usuario tiene restricciones en cuanto al tipo de operaciones que puede hacer con los datos(consulta, modificación, borrar registros,etc.). Lo anterior se basa en el diseño de las “Reglas de autorización” para el acceso a los datos.

El término Sistema de Base de datos surge como traducción al término anglosajón Data Base Managment System(DBMS), por lo que ambos términos son usados en los textos como equivalentes.

Con la tecnología de las Bases de datos aún se tienen algunos problemas que se resumen en las siguientes desventajas:

- No se tiene en el mercado actual un ambiente de desarrollo basado en un DBMS que cumpla con todos los objetivos y ventajas esperadas de ellos.
- Los ambientes de desarrollo basados en un DBMS consumen mas recursos computacionales que los Sistemas de Archivos tradicionales. ®
- Existe una falta de personal capacitado para aplicar y explotar adecuadamente la filosofía de las Bases de datos.
- La Base de datos centraliza todos los datos, y si por alguna razón llega a fallar, todas las transacciones no podrán efectuarse.

6.3 Arquitectura de un DBMS (una Base de datos)

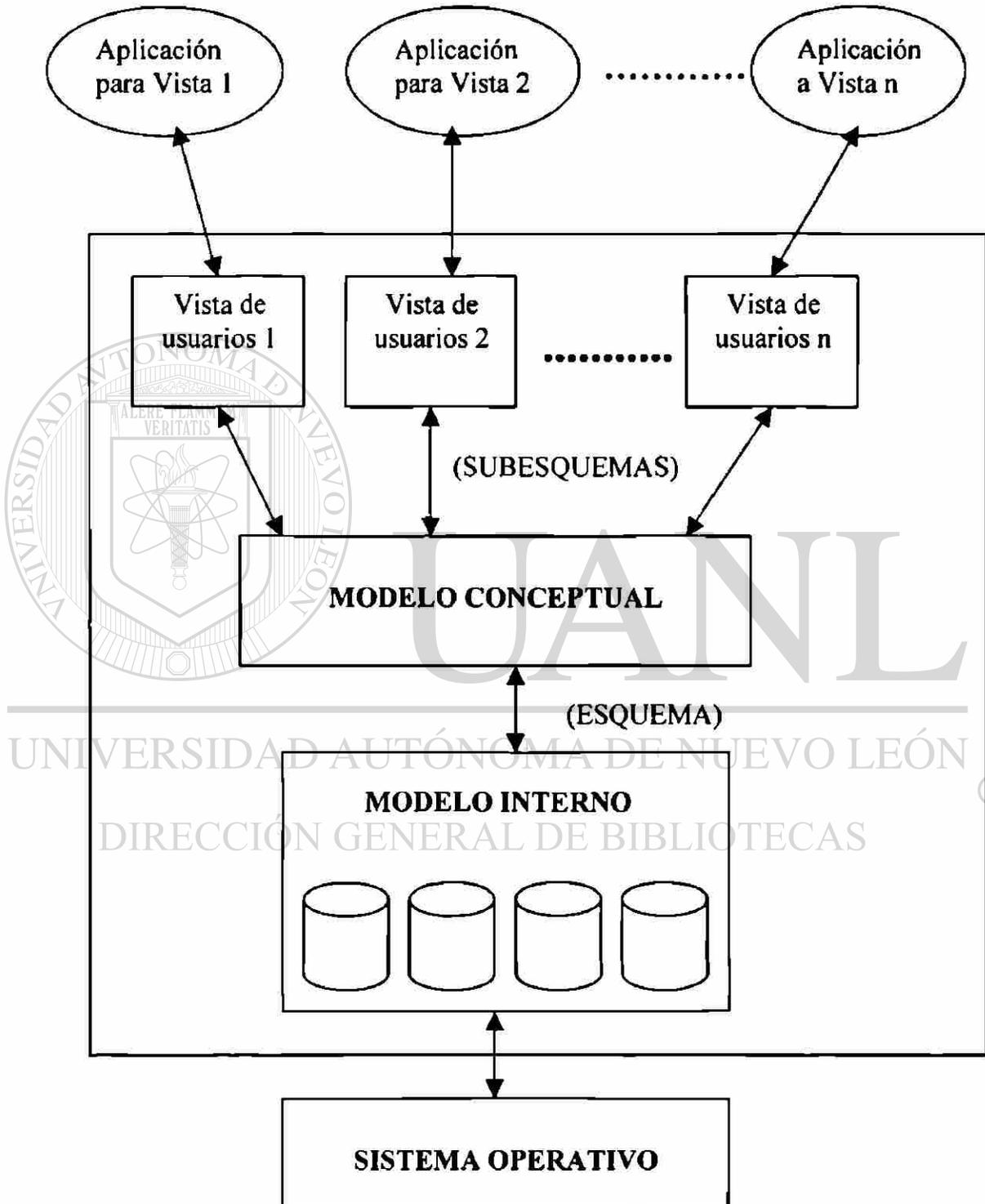


Fig. 25 Arquitectura de un DBMS

La arquitectura de una Base de datos(DBMS) tiene varios componentes entre los cuales podemos destacar:

- **El Modelo interno.** También conocido como el “Nivel físico”, es el nivel de abstracción mas bajo, en él se describen la manera a nivel de bits de cómo se almacenan los datos así como las estructuras de los mismos.
- **Modelo Conceptual.** También conocido como “Nivel conceptual”, es el siguiente nivel de abstracción(mas alto que el Nivel físico), aquí se describen los datos que son almacenados, así como las relaciones existentes entre ellos.
- **Esquema.** Es la definición de la estructura de la Base de datos(DBMS), con sus archivos(tablas), relaciones, dominios, reglas de acceso,etc.
- **Subesquema.** Define las estructura de datos y los archivos(tablas) disponibles para un tipo de usuario, ejemplo: Recursos humanos, Abastecimientos, Mercadotecnia, Producción, Administración, etc...
- **Vista del usuario.** Se refiere al área de control del usuario el cual puede acceder los datos que son inherentes a su trabajo.
- **Aplicaciones.** Se refiere a los programas de aplicación que opera el usuario.

Todo el funcionamiento de la Base de datos es soportada por el Sistema Operativo instalado en el computador.

Desde el punto de vista del Software la parte mas importante de la Base de datos es el Modelo Conceptual, ya que éste soporta diferentes estructuras de datos para lograr las ventajas propias del uso las Bases de datos en forma eficientemente. La definición del Modelo Conceptual se basa en las estructuras que maneja, y puede ajustarse a uno de los siguientes modelos:

Modelo Relacional

Modelo de Redes

Modelo Jerárquico

Modelo Relacional

Este modelo establece los datos y las relaciones entre ellos mediante un conjunto de tablas, los datos se referencian por su ubicación en cuanto a fila y columna, cada columna representa un dato(campo) y cada fila representa un registro, de esta forma una tabla representa un archivo.

Ejemplo:

Matricula	Nombre	Direccion
109020	Juan Pérez Martínez	Nogal 113 Nte.
109121	Ismael García Robles	Rio Nazas 240 Ote.
109240	Rebeca Morales Torres	Benito Juarez 1020 Sur
119470	Pedro Rodríguez Mata	Nogal 113 Nte.

No. Teléfono	Recibo Pago
607060	120.00
806090	170.00
102030	250.00
301020	120.00

Fig. 26 Tablas del Modelo Relacional

En 1974 la Compañía IBM inició un proyecto denominado “Sistema R” cuyo objetivo era encontrar una aplicación práctica del Modelo Relacional que por esa época se había propuesto en forma teórica, uno de los problemas era diseñar interfases de usuario apropiadas así como métodos para implementar las teorías del Modelo Relacional.

Entre las contribuciones mas importantes del “Sistema R” se tienen:

- El desarrollo del lenguaje de consultas SQL
- El concepto de compilación y optimización de consultas
- La integración de un lenguaje Relacional y un lenguaje de programación convencional
- La serializabilidad y el bloqueo de dos fases

En la misma época en que IBM atendía el proyecto del “Sistema R” en la Universidad Berkeley de California varios investigadores desarrollaban experimentos con una Base de datos llamada “Ingres”, y también la base de este proyecto eran las teorías del Modelo Relacional, aunque diferían del “Sistema R” en los diseños de las interfases de usuario.

El proyecto “Ingres” fructificó en un producto comercializable que lleva el mismo nombre del proyecto: “Ingres”.

A través de los años han surgido mas productos DBMS basados en las teorías del Modelo Relacional, entre los cuales se pueden mencionar:

- Oracle
- Progress
- Informix
- Sybase
- Unify
- SQL/DS
- DB2
- Ingres

Modelo de Red

En este modelo los datos se representan mediante conjuntos de registros, y las relaciones entre ellos se representan mediante enlaces, los cuales pueden tratarse como punteros, los registros se organizan como conjunto de grafos arbitrarios.

Ejemplo:

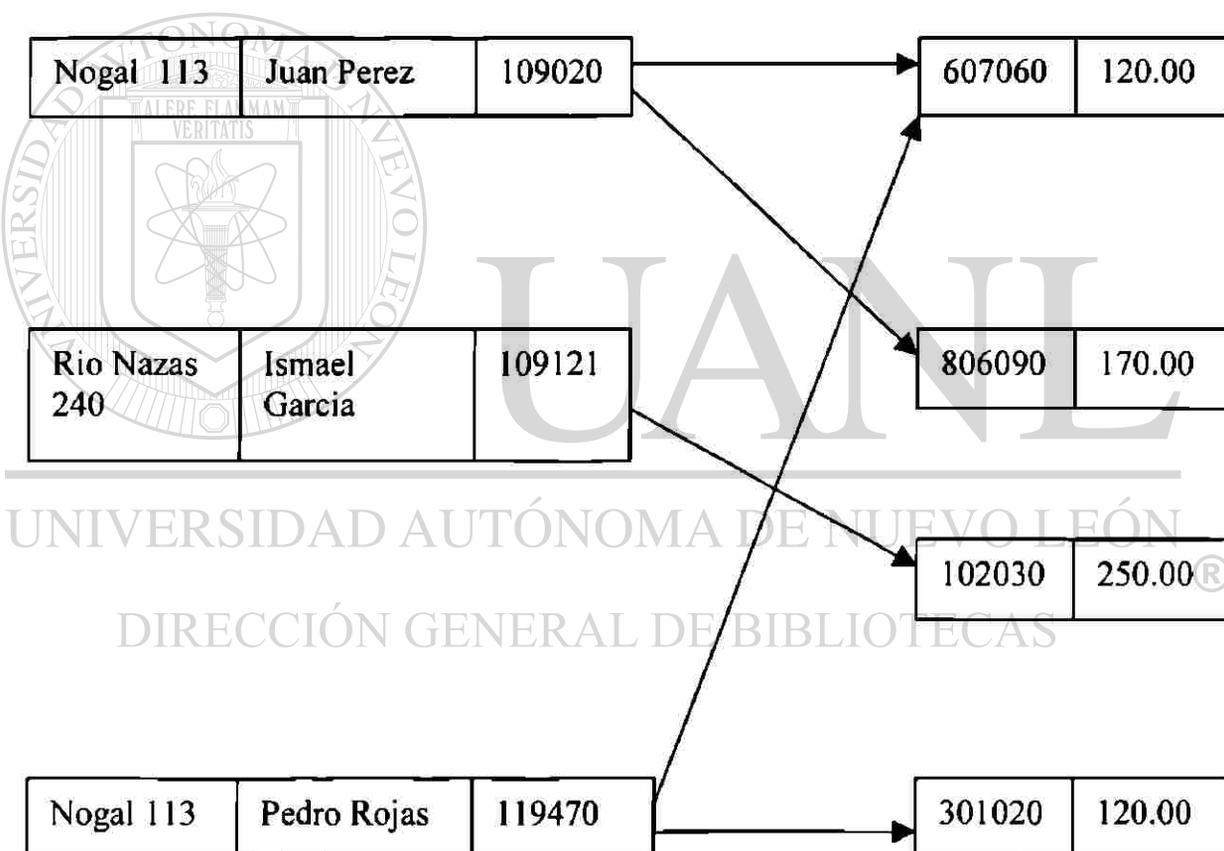


Fig. 27 Esquema del Modelo de Red

A fines de la década de 1960 se diseñaron varios sistemas comerciales de Bases de datos basados en el Modelo de Red, uno de los proyectos mas significativos de este Modelo fue el proyecto DBMS CODASYL DBGT.

En 1971 se emitió la primera especificación estandar del Modelo de Red, y se llamó CODASYL DBGT 1971. El proyecto CODASYL DBGT fue desarrollado por un grupo de investigadores llamado DATA BASE TASK GROUP, y CODASYL significa Conference on Data System Languages.

El Modelo de Red así como los productos basados en ella están dejando de tener demanda, y algunas de las razones para que esto suceda son:

- El Modelo es complejo y no cohesivo, por lo que al definir un esquema es posible que suceda que se combine con una operación aparentemente “No relacionada” con algún programa de aplicación, generándose con ello un conjunto de errores imprevisibles.
- La definición de sus esquemas “forzan” al uso del lenguaje Cobol, pero algunos usuarios no desean usar este lenguaje

Algunos de los productos existentes de este Modelo son:

- IDMS/R
- DMS-170

Modelo Jerárquico

Este modelo es similar al modelo de red, dado que los datos y las relaciones entre ellos se representan mediante registros y enlaces respectivamente. En este Modelo los registros están organizados como colecciones de árboles en lugar de grafos.

Ejemplo:

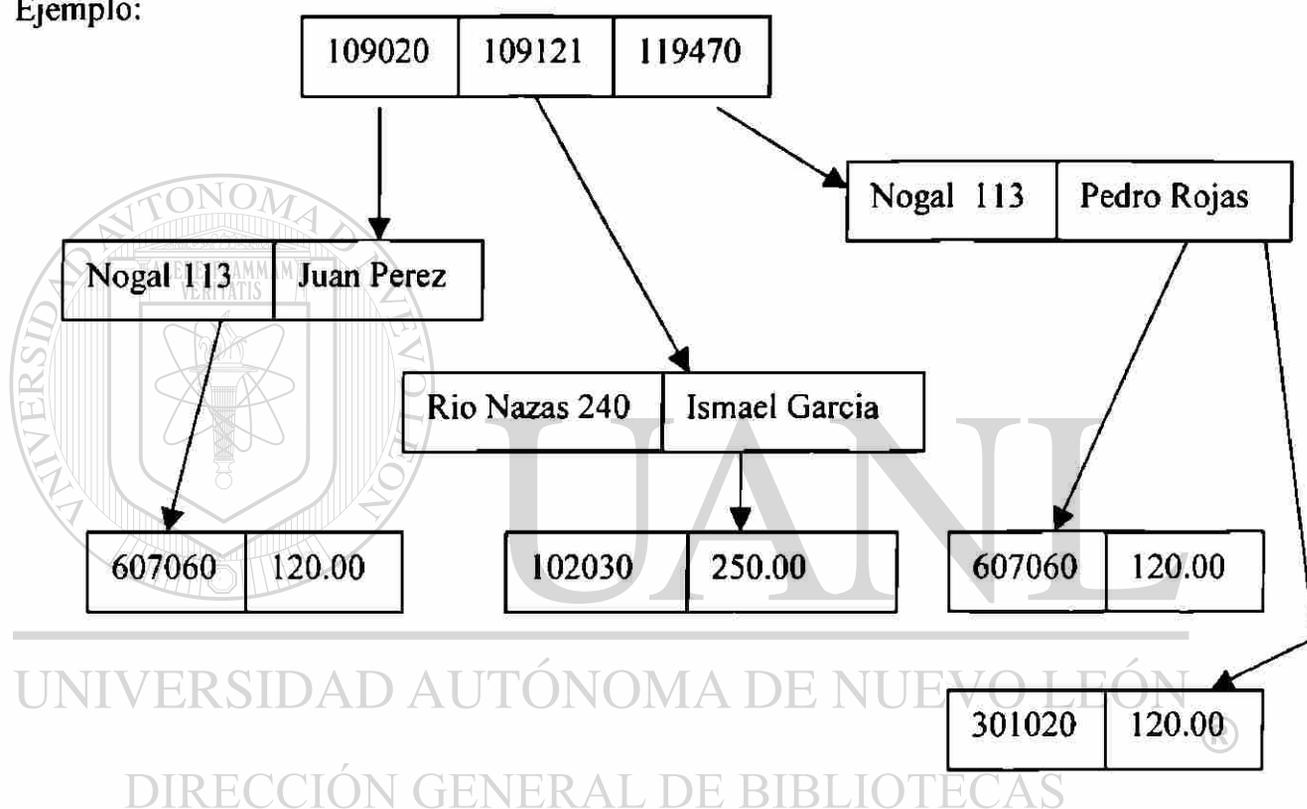
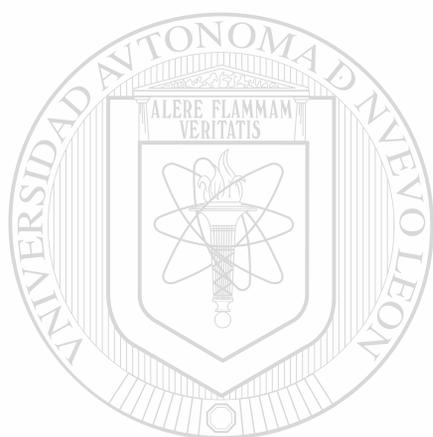


Fig. 28 Esquema del Modelo Jerárquico

Los Bases de datos basadas en el Modelo Jerárquico fueron creados antes de que surgiesen los conceptos del Modelo Relacional, muchos DBMS se desarrollaron en el Modelo Jerárquico y aún se siguen desarrollando. De acuerdo a éste modelo las relaciones entre los datos deben transformarse en Jerarquías ó árboles.

En la década de 1960 la Compañía IBM y la North American Aviation unieron sus esfuerzos para desarrollar un lenguaje dirigido al procesamiento de una Base de datos, generándose de esta manera el lenguaje PL/I implementándose en la Base de datos IMS de la IBM. Algunos de los DBMS mas usados en Modelo Jerárquico son:

- IMS de la IBM
- Sistema 2000 de la Compañía Intel



UANL

UNIVERSIDAD AUTÓNOMA DE NUEVO LEÓN

®

DIRECCIÓN GENERAL DE BIBLIOTECAS

6.4 DBMS PARA PC's

En 1979 una pequeña empresa denominada Ashton-Tate generó y promocionó en el mercado computacional un producto llamado Dbase II, lo comercializó como un DBMS Relacional, este producto fue dirigido a las Micro-computadoras y muchas personas adquirieron este producto ya que se promocionó a muy bajo precio.

Dbase II generó confusión respecto a las teorías establecidas para las Bases de datos ya que internamente no era un DBMS ni tampoco Relacional sino que en forma práctica era un mero Sistema de Procesamiento de archivos, pero sus compradores pensaban que estaban operando un DBMS. Inmediatamente después Ashton-Tate lanzó al mercado una nueva versión denominada Dbase III, la cual si cumplía con los atributos de un DBMS y posteriormente promocionó una nueva versión mejorada denominada Dbase IV.

Después de Dbase otros fabricantes de Software diseñaron mas productos DBMS para Micro-computadoras y además los vendedores de DBMS para MainFrames generaron versiones de sus productos para Micro-computadoras, algunos de ellos fueron: Oracle, Focus, Ingres.

Asimismo surgieron nuevos fabricantes de Software con nuevos productos entre los cuales podemos citar: Paradox, Revelation, MDBS, Helix.

Actualmente los productos DBMS para Micro-computadoras se presentan con interfases para el usuario basadas en Windows, y algunas de ellas son: Access, Fox Pro y SQL de Microsoft, además de VDBase y Paradox de Borland. Oracle 8i,9i de Oracle y Progress de Progress.

6.5 Perspectivas de las Bases de datos

Nuevas fronteras en la guerra de las Bases de datos.

Aunque el mercado de software de bases de datos parece estar completamente maduro, tecnologías emergentes como son en este momento el e-commerce, la gestión de contenidos y la inteligencia empresarial, están elevando notablemente el interés mostrado por las organizaciones en estas plataformas, incrementando consecuentemente la demanda de sistemas de gestión de bases de datos.

En este sentido y según informes elaborado por AMR Research entre los directivos de las mas importantes multinacionales de todo el mundo, casi la mitad de los ejecutivos consultados afirman piensan destinar la mayor parte de las inversiones que se realicen en el año 2001 al área de las bases de datos. Igualmente la mayor de los encuestados aseguró que continuará destinando la mayor parte de sus presupuestos de TI en las plataformas durante el año 2002.

Por su parte otra reconocida firma analista, Dataquest, estima que el mercado de bases de datos alcanzará un volumen total de 12.7 billones de dólares en el año 2004; mientras que Data Corp. Asegura que la necesidad de acceder de forma inmediata a los datos on-line crece, a la vez que las compañías incrementan el numero de funcionalidades expuestas en Internet con propósito de potenciar sus actividades de e-business. Esto significa que según se incrementan la práctica del negocio electrónico entre las organizaciones, se incrementa paralelamente la necesidad de instalar paquetes de software que gestionen estas bases de datos.

Otro de los aspectos que merece la pena destacar en cuanto a la relación entre aplicaciones de e-business y software de bases de datos es que los principales fabricantes de este segmento, como Oracle ó IBM están integrando sus bases de datos en infraestructuras de e-business.

En otro orden de cosas, hay que señalar también que según las mas importantes consultoras, con Sybase e Informix perdiendo cuota de mercado y debido a que SQL de Microsoft todavía no está a la altura de IBM y Oracle, la competencia de este segmento está marcada prácticamente por estos dos últimos fabricantes.

Artículo publicado en ComputerWorld del 17 de Noviembre del 2000

Prosperidad económica en el mercado de bases de datos Internet y las aplicaciones de business intelligent sacan a flote el mercado de bases de datos de cara al siglo XXI.

Según se recoge en el estudio realizado por Dataquest, las bases de datos gozarán de buena salud para el año 2004, ya que se estiman unos beneficios de 12.7 billones de dólares. Para Norma Schroder, autora del estudio, la causa de este crecimiento se fundamenta en el próspero mercado de Internet y las aplicaciones de business Intelligent.

Aunque el reparto del mercado sigue perteneciendo a Oracle e IBM, con el 31.1% y el 29.9% del cuota del mercado respectivamente, y Microsoft, con el 13.1%, Norma Schroder abre la posibilidad de nuevos vendedores encaminados a abastecer un mercado de industrias específicas o de áreas de aplicación con un bajo coste y alta ejecución, destinadas básicamente para uso Web.

Artículo publicado por Fernando Muñoz en Dealer Word de Noviembre del 2000

Los artículos anteriormente citados declaran la importancia significativa de las bases de datos para las empresas pequeñas, medianas y grandes en la actualidad, y asimismo su tendencia a futuro.

En todas las empresas la información se ha convertido en un recurso de sumo valor, el cual debe ser gestionado de forma segura(lo mas posible), y que se logre su acceso a todo tipo de usuario(en forma controlada) en el menor tiempo posible independientemente de su situación geográfica.

En el mercado de las bases de datos actualmente existen un gran numero de empresas desarrolladoras de software de gestión de bases de datos, a los cuales también se les conoce como “Sistema gestor de Bases de datos(SGBD)”.

Por definición un SGBD es un Servidor de bases de datos relacionales de arquitectura Cliente/Servidor que proporciona servicios de gestión, administración y protección de los datos a través de conexiones de red gobernadas por protocolos definidos, con el objetivo de operar grandes volúmenes de datos de una manera estable, fiable, coherente y segura en un entorno heterogéneo de trabajo.

Una de las funciones de un SGBD es proporcionar herramientas de apoyo a la toma de decisiones(Datawarehouse) al tiempo que proporciona una plataforma de transacciones on-line(OLTP), que hacen que la información esté siempre actualizada y consistente. Así como mecanismos de comunicación con otras plataformas que actúen como Clientes y como Servidores de datos, dando la posibilidad de “replicación de datos”.

Aunque parece clara la función de un SGBD, en la actualidad cada vez mas filosofías y tecnologías tienden a confluir en un mismo punto, ya se habla de las posibilidades de los nuevos SGBD para poder almacenar contenidos multimedia, objetos, así como documentos complejos.

Puesto que todo tiende a unificarse con Internet, los SGBD también deben proporcionar servicios de datos a la Red, los servicios disponibles incorporan generación y alimentación de páginas Web a partir de consultas prediseñadas en la base de datos, dichas consultas mantendrán alimentadas las páginas Web, las cuales estarán siempre actualizadas con la última información.

Los SGBD mediante el uso de Bases de datos distribuidas facilitan la compartición y distribución de la información a lo largo de los servidores situados en cualquier parte del mundo, esto permitirá a las empresas integrar sus bases de datos con sus proveedores ó clientes, de manera que podrán colaborar con los niveles de servicios y recursos de información, ganando en rapidez y fialidad.

El servicio mas importante que proporciona un SGBD es el acceso a la información que almacena, el como recuperar y actualizar dicha información es un proceso critico del cual depende el éxito de este tipo de sistemas. El lenguaje que actualmente se usa es el SQL bajo el estándar ANSI SQL'92 el cual es la base de todos los SGBD independientemente de la empresa que lo desarrolló.

Asimismo los SGBD contienen herramientas que facilitan las acciones de la administración de los datos en un entorno amigable y sencillo de operar evitándole al administrador las acciones de bajo nivel al mismo tiempo que maximice el nivel de automatización de tareas.

6.6 El Administrador de Bases de datos (DBA)

La funcionalidad adecuada de los procesos de los SGBD es responsabilidad de la persona a la que se le confía el puesto de “Administrador de Bases de datos”, y en particular algunas de sus funciones son :

- Encargado del mantenimiento de la base y de los datos,
- Vigilar por la su integridad de los datos
- Manejo del gestor de datos
- Definición y mantenimientos de los índices (por corrupción o cualquier motivo)
- Definición y mantenimiento de directorios y álias de las bases de datos
- Control de las consultas y sentencias desde fuera de programas
- Vigilar conexiones o transacciones, etc., como del acceso de usuarios
- Autorizaciones y palabras de paso
- Diseñar métodos de seguridad y su implantación
- La operación depende totalmente de la base de datos de que se trate.

UNIVERSIDAD AUTÓNOMA DE NUEVO LEÓN

DIRECCIÓN GENERAL DE BIBLIOTECAS

6.7 El mercado de los SGBD en la actualidad

En el mercado de los Sistemas de Gestión de Bases de Datos existe un gran número de empresas desarrolladoras de software que ofrecen sus productos, cada uno con características propias, pero unos más potentes ó con más facilidades que otros.

EMPRESA	SUS PRODUCTOS
Oracle	Oracle 8i Oracle 9i
IBM	DB2 Universal Database
Microsoft	SQL Server Desktop SQL Server 7.0 SQL Server 7.0 Enterprise Edition
Computer Associates	CA-Datcom CA-IDMS Ingres II Jasmine [®]
ComShare	Essbase
Informix	On-line Dynamic Server
Progress	Progress Personal Database Progress Workgroup Database Progress Enterprise Database
Saga	Adabas v6.2 Adabas v4.1
Sybase	Sybase Adaptive Server Enterprise

Tabla 10 Empresas y sus productos DBMS

Características de los productos.

DB2 Universal Database (de la IBM)

Dirigido a :

Personal Edition : Microempresas y usuarios personales

Workgroup Edition : Empresas pequeñas y medianas

Enterprise Edition : Empresas con necesidad de operar grandes bases de datos y/o configuraciones complejas en plataformas Intel ó Unis

Plataformas de operación : Windows NT, Windows 95, Windows 98, OS/2, SCO, Sun-Solaris, HP-UX, AIX

Soporte de parte de Casas de Software: Baan, SAP, PeopleSoft, JD Edwards

Ventajas competitivas : Escalabilidad, integración total con Internet, multiplataforma, capacidad para manejar grandes volúmenes de datos, manejo de funciones OLAP

CA-Datacom, CA-IDMS (de Computer Associates)

Dirigido a : Empresas pequeñas, medianas y grandes

Plataformas de operación : Unix, Linux, Windows NT, Open VMS, VAX, Novell, Windows 95, 3.1 y 3.1.1

Soporte de parte de : Adobe Software, Altavista Internet Software, BEA Systems, BEA Systems, Business Objects, Centura Software, Soft-Tek Internacional, Verity, Wave Research entre otros.

Ventajas competitivas : permiten utilizar los tres paradigmas en manejo de información: el modelo legacy, el modelo relacional, y el modelo de objetos en forma integrada, permiten que un cliente pueda completar un esquema para el manejo de información sin importar las adquisiciones hechas con anterioridad y manteniendo sus inversiones con vigencia tecnológica.

CA-IDMS es una base de datos para Mainframes y trabaja bajo arquitectura de redes, con opción a funcionar bajo arquitectura relacional.

Ingress II (de Computer Associates)

Dirigido a : Empresas pequeñas, medianas y grandes

Plataformas de operación : Unix, Linux, Windows NT, Open VMS, VAX, Novell, Windows 95, 3.1 y 3.1.1

Soporte de parte de : Adobe Software, Altavista Internet Software, BEA Systems, BEA Systems, Business Objects, Centura Software, Soft-Tek Internacional, Verity, Wave Research entre otros.

Ventajas competitivas : Base de datos relacional diseñada para aplicaciones de misión crítica, permite acceder distintas fuentes de información (Sybase DB2, RDB, VSAM, Datacom, IDMS, Informix, All Base y SQL Server) para integrar accesos de información mediante Gateways diseñados para cada uno de los fabricantes mencionados,

permite que la información se despliegue en sitios Web, herramientas de desarrollo, generadores de reportes, utilerías gráficas de administración, sistemas de respaldo y recuperación de información.

Jasmine (Computer Associates)

Dirigido a : Empresas pequeñas, medianas y grandes

Plataformas de operación : Windows NT, Unix.

Soporte de parte de : Adobe Software, Altavista Internet Software, BEA Systems, BEA Systems, Business Objects, Centura Software, Soft-Tek Internacional, Verity, Wave Research entre otros.

Ventajas competitivas : Base de datos orientada a objetos para aplicaciones multimedia y de comercio electrónico, manejo de datos tipo sonido (Wav), video (AVI, Quicktime), VRTML puede ser accesada vía C++, Visual Basic, Java.

Essbase (de ComShare)

Dirigido a : Grandes Corporativos.

Plataformas de operación : AS/400, Unix, Windows NT, Windows 95, Windows 98.

Soporte de parte de : ComShare.

Ventajas competitivas : Es una base de datos orientada a sistemas OLAP, constituida por dimensiones, provee acceso desde hojas electrónicas como Excel y Lotus hasta visualizadores altamente analíticos; se pueden construir aplicaciones personalizadas con Visual Basic y otras herramientas, permite crear aplicaciones para el Web.

Sobre las aplicaciones es una base de datos centralizada, multi – dimensional, permite construcción de escenarios en línea, está diseñada para la toma de decisiones.

Oracle 8i (de Oracle)

Dirigido a : Oracle 8i Personal Para desarrolladores(PC's y Laptop).

Oracle 8i Server Para empresas pequeñas y medianas.

Oracle 8i Enterprise Para empresas grandes con volúmenes grandes de información y con gran numero de usuarios

UNIVERSIDAD AUTÓNOMA DE NUEVO LEÓN

Plataformas de operación : Unix, Windows NT, NetWare, Linux, UnixWare y otras más.

DIRECCIÓN Se instala en máquinas Sun, HP 9000, RS 6000, Digital Alpha

Soporte de parte de : Los fabricantes de software: CTI, Ingeniería Cóndor, Sice Corporation, Grupo SYC, Anderson Consulting, Antar, ASSIS, SIGSA, JD Edwards, Exactus Software, CINCOM, Arango Software, etc.

Ventajas competitivas : Es una base de datos diseñada para Internet, menor costo de propiedad, alta disponibilidad, particionamiento a nivel tabla, repli – cación avanzada, operaciones distribuidas, manejo de grandes volúmenes de información, alta confiabilidad, concurrencia de datos.

*** En Mayo del 2001 Oracle anuncia la comercialización de Oracle 9i

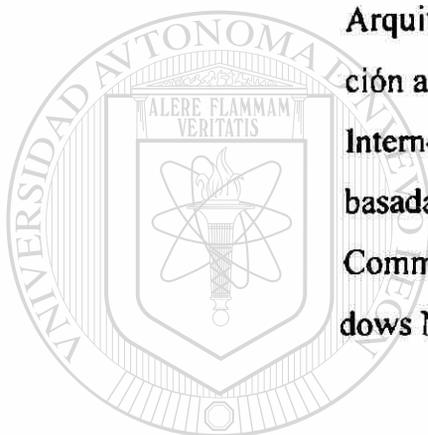
On-line Dynamic Server (de Informix)

Dirigido a : Empresas medianas y grandes.

Plataformas de operación : Unix, Plataformas independientes y de red.

Soporte de parte de : fabricantes de software como SAP, PeopleSoft, Baan, Siebel.

Ventajas competitivas : De configuración multi-hilo, de multiprocesamiento simétrico, Arquitectura dinámica y escalable cliente/servidor, administración a nivel de mainframe, capacidad de multimedia y para Internet, soporta toda la línea de herramientas de desarrollo basadas en SQL, administración gráfica a través de Enterprise Command Center, instalado en un cliente Windows 95 ó Windows NT.



UANL

SQL Server 7.0, SQL Server Desktop (de Microsoft)

Dirigido a : Empresas pequeñas, medianas y grandes.

Plataformas de operación : Windows 95, Windows 98, Windows NT.

Soporte de parte de : Fabricantes de software como ASPEL, SAP, PeopleSoft, Baan, Solomon, entre otros.

Ventajas competitivas : Integración, plataforma sencilla y poderosa, servicios OLAP para análisis de datos en distintos niveles de la organización, facilidad de uso, administración automática de espacio, facilidad para construir modelos OLAP y OLTP a un bajo costo de administración y de propiedad.

Progress (de Progress)

Dirigido a : Personal Database a usuarios individuales, desarrolladores monousuarios y Pequeñas empresas.

Progres Workgroup a empresas pequeñas y medianas con personal de 5 a 500 empleados.

Progress Enterprise a empresas medianas y grandes, con corporativos de 500 a miles de empleados.

Plataformas de operación : Unix, SCO, DG-UX, HP-UX, AIX, Solaris, Digital UX, Windows, OS/400.

Soporte de parte de : Fabricantes de software como QAD, Synix, Datasul, Micros Personales, Grupo SPIC, Tarik.

Ventajas competitivas : Alta flexibilidad, Independencia de bases de datos, conectividad Con DB2/400 en forma nativa, con bases de datos de Informix, Sybase, Visual Basic a través de ODBC. Conectividad con desarrollos de Internet a través de JDBC, multiplataforma, bajo costo de mantenimiento, escalable, cumple con estándares de SQL'92 e interfaces ODBC 3.0, JDBC, ESQL/C y CLI, altos niveles de seguridad.

Adabas V6.2, V4.1, V3.1 (de Saga)

Dirigido a : Empresas medianas y grandes.

Plataformas de operación : Adabas 6.2 en MVS, OS390, VSE, VM.

Adabas 4.1 en Open VMS

Adabas 3.1 en Unix, HP-UX, AIX, Digital, Solaris, SCO y
Windows NT.

Soporte de parte de : Fabricantes de Software como Grupo Fabre, Unión Fenosa, Soft –
ware Tecnology, ABS, entre otros.

Ventajas competitivas : 29 años en el mercado, escalable y portable, alto rendimiento,
manejo de grandes volúmenes de información, control total en
transacciones diarias de misión críticas, protección a la inver-
sión del cliente, relación costo/beneficio.

UNIVERSIDAD AUTÓNOMA DE NUEVO LEÓN
Sybase Adaptive Server Enterprise (de Sybase) ®

DIRECCIÓN GENERAL DE BIBLIOTECAS

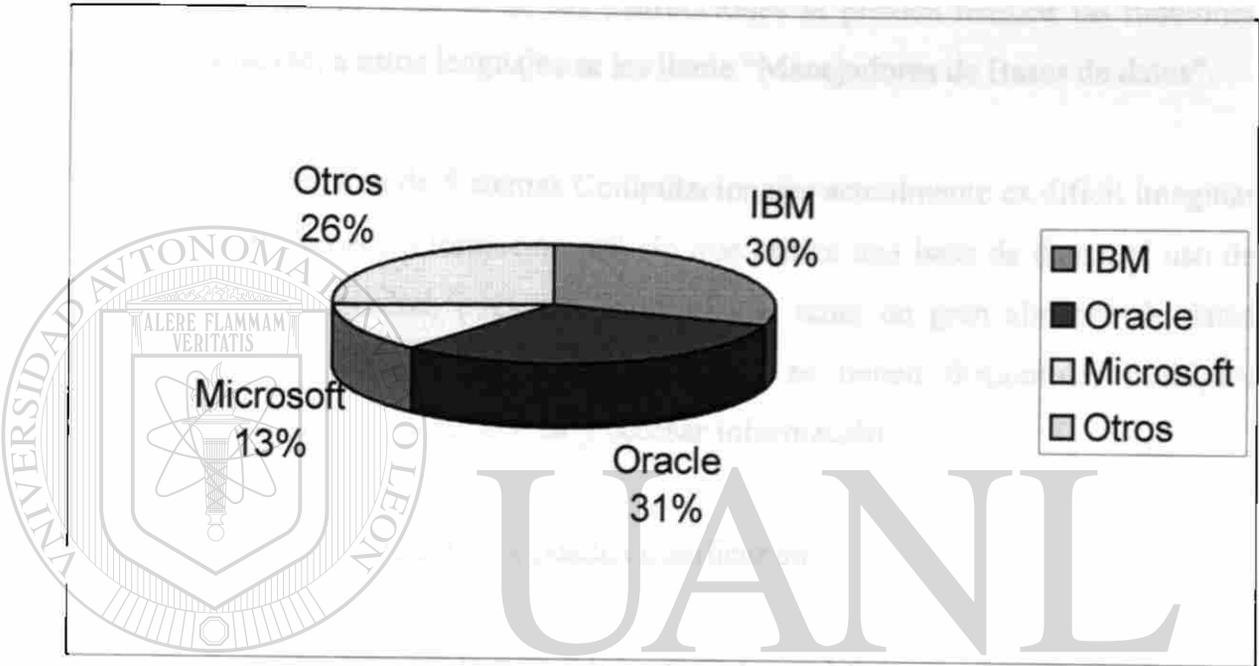
Dirigido a : Empresas pequeñas medianas y grandes.

Plataformas de operación : DOS, Windows 3x, Unix, VMS.

Soporte de parte de : Sybase.

Ventajas competitivas : 100% compatible con el estándar SQL'92, acceso a bases de
bases de datos heterogéneas, cómputo móvil, replicación en
línea, ambiente Java, alto desempeño, menor costo de hardware,
replicación orientada a mensajes.

IBM	299
Oracle	311
Microsoft	131
Otros	259



UNIVERSIDAD AUTÓNOMA DE NUEVO LEÓN
DIRECCIÓN GENERAL DE BIBLIOTECAS

Fig. 29 Distribución del mercado de las Bases de datos

6.8 El segundo lenguaje de Programación.

Las bases de datos propiamente dicho son las tablas y los datos almacenados en ellas, y para su manipulación (ordenamientos, búsquedas, generación de reportes, etc.) se utilizan lenguajes donde a través de sus instrucciones se pueden realizar las funciones para su manipulación, a estos lenguajes se les llama “Manejadores de Bases de datos”.

En el ámbito del área de Sistemas Computacionales actualmente es difícil imaginar una aplicación que genere información útil sin que utilice una base de datos, el uso de bases de datos es primordial, pues resulta en ello el tener un gran almacén de datos disponibles para su proceso, en tales situaciones se tienen disponibles múltiples proveedores de software para almacenar y acceder información.

En general las bases de datos se pueden clasificar en :

- Bases de datos Desktop (ó de escritorio) usadas en Micro-computadoras
- Bases de datos para grandes equipos de cómputo (Servidores)

Bases de datos Desktop

Para facilitar el acceso a los datos los fabricantes de software dotaron a éste de una capa intermedia entre el código de la aplicación y los controladores ODBC a la cual se le denomina “Motor de Base de de datos”, actualmente dos compañías han construido su propio “Motor” y son:

Compañía	Motor de Base de datos
Microsoft	Microsoft Jet
Borland	BDE (Borland Database Engine)

Tabla 11 Motores de Bases de datos

Con la aparición de la metodología RAD en la programación esta actividad se reduce a manejar adecuadamente controles y objetos que definen la operacionabilidad de las nuevas aplicaciones.

Borland Databse Engine y Microsoft Jet ambos soportan la programación tradicional para bases de datos relacionales : Lenguaje SQL.

Bases de datos para grandes equipos de cómputo (Servidores)

Un Servidor de bases de datos debe estar preparado para operar con las siguientes características :

- Alto rendimiento
- Gran escalabilidad
- Replicación de datos
- Bloqueo de datos
- Gestión de transacciones en línea, Asistencia a decisiones
- Portabilidad

Existen actualmente muchos productos de software para bases de datos que ostentan un lugar en el mercado y que cumplen con las características mencionadas, tales como : Interbase de Borland, SQL Server de Microsoft, Adabas D y Adabas C de Software AG, SQL Server de Sybase, Oracle, Progress, etc.. y la característica común de todos ellos es que su programación puede ser orientada al Lenguaje SQL.

Para efectos de este punto de la Tesis, y buscando conocer el producto (ó productos) manejador(es) de Base de datos mas usado(s) en la industria del área Metropolitana de Monterrey Nuevo León, se aplicó una encuesta a un grupo de empresas, y sus resultado fue el siguiente :

- El 42% de las empresas encuestadas usan SQL y Access
- El 17% de las empresas encuestadas usan SQL y Progress
- El 34% de las empresas encuestadas usan SQL y Oracle
- El 7% de las empresas encuestadas no indicaron el uso de Base de datos

Es entonces que lo mas lógico definir a SQL como el segundo lenguaje(para el manejo de bases de datos), en conjunción con otro producto como Oracle, Progress ó Access, para su aprendizaje en la carrera de Ingeniero en Sistemas Computacionales en el Instituto Tecnológico de Nuevo León.

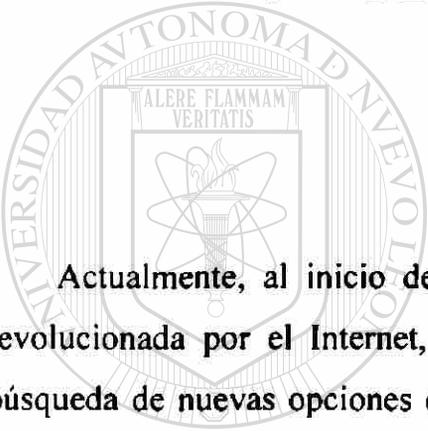
Tomando como modelo al producto SQL Server de Microsoft, los requerimientos para su instalación en red son:

- Sistema Operativo Windows NT Server version 4.0 ó version posterior, tambien puede operar en Windows 2000 Server y Windows 2000 Advanced Server, así como en Windows 2000 datacenter Server.
- Memoria RAM de 64 MB(se recomienda 128)
- Espacio en Disco Duro de 270 MB para una instalación completa
- Unidad de CD-ROM
- Monitor VGA(se recomienda SVGA)
- Navegador de Internet

en red es compatible con Windows 95, Windows 98, Windows Me, Windows NT WorkStation 4.0 y Windows 2000 Professional, Unix, Apple Macintosh, OS/2, el software del cliente requiere de ODBC.

7

DISEÑO Y PROGRAMACIÓN DE APLICACIONES EN INTERNET



Actualmente, al inicio del segundo milenio el área de la computación ha sido revolucionada por el Internet, éste como red de alcance mundial ha despertado la búsqueda de nuevas opciones de aplicación en todas las comunidades tanto científicas como educativas, así como de las empresas y hasta las Instituciones de gobierno.

En el plano educativo ha significado una poderosa herramienta para despertar en los estudiantes el espíritu de investigación y consulta de información relativos a temas tratados en las aulas, en las empresas significa la búsqueda de oportunidades de los negocios electrónicos así como de la optimización de algunos de sus procesos.

Muchas de las empresas así como las Instituciones de gobierno utilizan el Internet para comunicar acerca de sus productos y/o servicios, así como de las disposiciones definidas y cambios efectuados para un servicio ó compra efectiva, y para lograr todo lo anterior es necesario que los alumnos(sobre todo los de las carreras relacionadas con la computación) en las Instituciones de educación aprendan las técnicas y metodologías adecuadas en el diseño y programación de aplicaciones en Internet para apoyar las actividades de los usuarios.

7.1 Antecedentes históricos de la red Internet

En tiempos de la segunda guerra mundial un hombre cuyo nombre fue Vannevar Bush hizo posible que se unieran los esfuerzos del gobierno de los Estados Unidos, los empresarios y la comunidad científica de ese país, con objeto de potencializar tecnologías que coadyuvaran en el desarrollo científico y con aplicaciones militares; Esta alianza generó asociaciones con propósitos específicos tales como la NSF(National Science Foundation) y la ARPA(Advanced Research Projects Agency).

La ARPA fue fundada con objeto de desarrollar tecnologías para la defensa militar, un empleado de la ARPA(J.C.R. Licklider), en 1962 propuso el establecimiento de una red de computadoras que comunicara información entre los nodos en forma eficiente, esta fue la primera idea de comunicación entre computadores. Por esta misma época en el Instituto Tecnológico de Massachussets(MIT) se diseñaron las teorías de la comunicación por paquetes.

En 1965 la DARPA(Defense Advanced Research Projects Agency) de los Estados Unidos promueve un estudio sobre “Redes Cooperativas de computadoras de tiempo compartido” y al siguiente año, en 1966 Larry Roberts del MIT publica un artículo llamado “Hacia una red cooperativa de computadoras de tiempo compartido”.

Entre 1966 y 1969 la IPTO(Information Processing Techniques Office) que era dirigida por Bob Taylor Acogió “antigua” idea puesta por Licklider y contrató a Larry Roberts como director del proyecto, de este proyecto surgió la creación de una conocida como ARPANET, que sería la antecesora de Internet.

A fines de la década de 1960 una preocupación de las Fuerzas Armadas de los Estados Unidos era tener una red de comunicaciones descentralizada, de forma que no hubiera un solo nodo central, que si llegara a ser destruido en un ataque nuclear el país

se quedara paralizado por la in-comunicación, sino que solo los nodos que sufrieran un ataque quedarían paralizados, pero no toda la red.

En 1969 la DARPA en conjunto con la compañía Rand Corporation implementaron una red descentralizada basándose en la técnica de la “Conmutación por paquetes”, utilizando un protocolo de comunicación llamado NCP(Network Control Protocol). En 1971 Ray Tomlinson creó el primer programa para el envío de correo electrónico combinado con otro programa de transferencia de archivos, fueron programas “sencillos” pero fueron las bases para el posterior “File Transmisión Protocol” ó “FTP”.

Las Instituciones académicas lograron conectarse a la ARPANET a través de la NSF(National Science Foundation), en la década de 1970 la ARPANET creció enormemente, y también se crearon otras redes como Telnet, Alohanet y Satnet, de forma que ya había muchas redes alrededor del mundo pero eran incompatibles entre ellas.

En 1974 Vinton Cerf con Bob Kahn publican su artículo “Protocolo para Intercomuni3n de Redes por Paquetes” especificando el dise1o de un nuevo protocolo llamado TCP(Transmisión Control Protocol), el cual al ser desarrollado se convirti3 en el est1andar de las comunicaciones. En 1979 ARPA cre3 la comisi3n de control de la configuraci3n de Internet, quien en 1981 concluye la definici3n del protocolo TCP/IP (Tansfer Control Protocol/Internet Protocol) y se adopta como un est1andar en 1982 substituyendo a NCP. “Internet” es la contracci3n de “Interconecetd Network”.

En 1983 ARPANET se separa de la red militar, quienes al liberar sus nodos dejan abierta la oportunidad para el ingreso de empresas, Instituciones educativas y otras mas, la gran expansi3n de Internet inici3 en 1987 cuando NSFNET permite el ingreso de otras redes de Europa, asimismo en ese a1o se genera la primera aplicaci3n en hipertexto, fue el producto “hypercard” generado para la Macintosh.

En 1990 la conexión desmedida de usuarios provocó que la ARPA se retirara de la conexión, y de sus nodos se hizo cargo la NSF, el Internet comenzó a saturarse y sus responsables se vieron desbordados, ese mismo año Tim Berners-Lee diseñó una nueva manera de operar con el Internet: El World Wide Web, haciendo mas fácil compartir y consultar datos. En 1993 Marc Andresen creó un navegador en la Web llamado "Mosaic", y después participó en el proyecto del desarrollo de un nuevo navegador "Netscape Navigator", ese mismo año nuevamente Berners-Lee creó las bases del protocolo HTTP, el lenguaje HTML y los conceptos de los URL.

En 1995 fue el año del gran despegue del Internet en el plano comercial, superando a Telnet, consolidándose como el servicio de red mas popular en el mundo, a partir de este año el incremento en el desarrollo de tecnologías tendientes al aprovechamiento del Internet se ha incrementado notablemente: se desarrollaron los Motores de búsquedas y se les añaden búsquedas inteligentes en varios idiomas, se generaliza la utilización del lenguaje Java lográndose la creación de entornos virtuales, la conexión telefónica simple permite la conexión con precio de llamada local, se desarrolla el comercio electrónico mediante tiendas virtuales, surgen los virus de HTML, la tecnología de telefonía móvil y el Internet se unen para que éste sea accesado desde teléfonos móviles mediante el uso de protocolos WAP(Wirless Application Protocol).

DIRECCIÓN GENERAL DE BIBLIOTECAS

Consecuencia del uso de la tecnología móvil en el Internet se desarrollaron páginas WML que pueden ser accesadas desde cualquier terminal WAP, las cuales ofrecen servicios de buscadores, juegos de entretenimiento, comercio electrónico y hasta aplicaciones bursátiles.

Como ha sido descrito el uso de la red de Internet es amplio y variado, por lo que en la actualidad las Instituciones de educación superior deben incluir entre sus asignaturas temas que otorguen a sus alumnos rasgos del perfil informático del uso de esta herramienta, y a los alumnos del área de Sistemas y Computación el conocimiento de las herramientas a usar para el desarrollo de aplicaciones en el ambiente del Internet.

****Información obtenida de la página "Enteder Internet" de www.fiestas.com**

7.2 Herramientas para el diseño y la programación en Internet

La evolución de la tecnología de protocolos en Internet ha permitido el amplio uso de la red en el plano de la comunicación entre usuarios y entre computadores, y en el plano del desarrollo de aplicaciones han surgido diversos lenguajes orientados al Diseño y la Programación a través de las llamadas “Páginas” en Internet.

El World Wide Web es un sistema distribuidor de información operado a través de Internet, el cual está fundamentado en el uso de Hipertexto e Hipermedia, desarrollado por Tim Berners-Lee en el Laboratorio Europeo de Física en Partículas(CERN) ubicado en Suiza, en él se incluyen las teorías y conceptos del Protocolo http, el lenguaje HTML u los URL. Que son las bases para la construcción, la localización y el acceso a las “Páginas” en Internet, comúnmente llamadas “Páginas Web”.

Toda aplicación operada a través de Internet se basa en la presentación y operación de una “Página Web” a través de la cual se pueden realizar diversas tipos de operaciones:

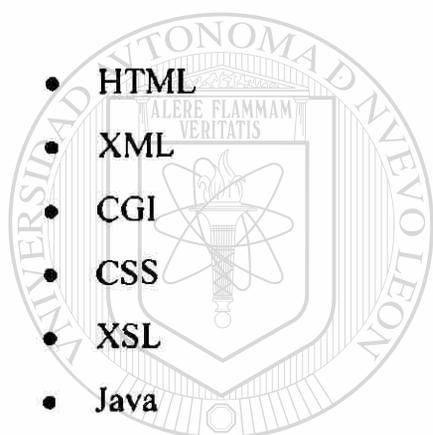
- Promoción de Productos y/o Servicios (Publicidad).
- Proceso de Compra/Venta en línea
- Publicación de información de todo tipo (Cultural, Recreativa, Técnica, etc).
- Consulta de información de diversos tipos
- Etc.

En el transcurso del tiempo del uso de la red Internet poco a poco se han incorporado diversas tecnologías propias, por medio de las cuales se han diversificado los tipos de aplicaciones: Transmisión de datos, sonido, animaciones, video en tiempo real, imágenes en 3D, chats, etc.

Paralelamente a la generación de nuevos tipos de aplicaciones en Internet se han desarrollado diversos lenguajes que permiten el diseño y desarrollo de las aplicaciones que a través de una "Página Web" logren acceder información distante en forma transparente para el usuario.

De los lenguajes de programación en Internet unos son mas sencillos que otros, pero cada uno de ellos tiene características que los definen como lenguajes especializados en tareas determinadas para el World Wide Web, algunos de los que tienen mas presencia en los desarrollos son:

- HTML
- XML
- CGI
- CSS
- XSL
- Java
- JavaScript
- VBScript
- Perl
- PHP
- Flash
- VRML
- ASP



UANL

UNIVERSIDAD AUTÓNOMA DE NUEVO LEÓN
DIRECCIÓN GENERAL DE BIBLIOTECAS



7.2.1 Comentarios generales sobre los Lenguajes para Internet.

HTML. Este lenguaje cuyo significado es Lenguaje de HiperTexto basado en marcas (HiperText Markup Language), que por sus características y sencillez se convirtió en un lenguaje estandar para aplicaciones en Internet, es un derivado directo de un antiguo lenguaje llamado SGML(Standard Generalized Markup Language). Las primeras “Páginas Web” creadas en HTML se creaban manualmente por lo que el programador debía dominar completamente el lenguaje. En la actualidad se tienen Procesadores de Texto que han sido dotados con facilidades de creación de “Paginas Web” (Word, WordPerfect, Ventura, HotMetal, PageMaker, Ami Pro, etc). HTML contiene una serie de normas y reglamentaciones para el manejo de características de los formatos de texto tales como: Tipo y tamaño de Fuentes por líneas ó párrafos, su código es en base al uso de los símbolos “<” y “>” que inician y terminan respectivamente líneas ó párrafos de la aplicación. Las páginas creadas solo con HTML solo manejan el texto y no pueden ser dotadas de imágenes, tablas, 3D, sonido etc. Si se desea dotar de estas características a la página en construcción sse deberá usar otras herramientas además del HTML.

DIRECCIÓN GENERAL DE BIBLIOTECAS

®

XML. Ante las limitaciones del lenguaje HTML subsanadas mediante la incrustación de tecnologías Scripts, ActiveX y HTML Dinámico, se ha generado un nuevo derivado del SGML: El lenguaje XML(eXtensible Markup Language), el cual es un lenguaje de matamarcado que opera bajo un formato facilitador para la descripción de datos estructurados, además éste lenguaje permite que los datos sean uniformizados con independencia del software de aplicaciones, esta interoperabilidad dá oportunidad para una nueva generación de aplicaciones para el comercio electrónico en la Web. XML proporciona un nuevo estándar de codificación de datos permitiendo aplicaciones para :

- Un documento normal
- Registros estructurados (Pedidos, Pagos, etc.)
- Manipular Objetos con datos y métodos, como un formulario permanente de un objeto Java ó un control ActiveX
- Un registro de datos como un conjunto de resultados de una consulta
- Operar Metacontenido sobre un sitio Web
- Representaciones gráficas como interface de usuario

En principio XML y HTML no son rivales entre sí, sino que se complementan uno al otro pudiendo estar HTML dentro de XML anidándose ambas gramáticas, algunas características de XML son:

- Puede ser utilizado directamente sobre Internet
- Soporta una amplia variedad de aplicaciones
- Compatible con SGML
- Los documentos XML operan bajo dos estructuras: Lógica y Física
- Facilidad para el desarrollo de programas para proceso de documentos
- El diseño de documentos con XML es fácil, rápido y eficiente
- El programador puede inventar sus propias etiquetas
- Las facilidades Hipertextuales son mayores en XML que en HTML
- Los archivos XML pueden ser accesados desde SGML
- Otorga mas facilidades de representación a los visualizadores
- Elimina muchas de las dificultades del SGML

CGI. El nombre de esta herramienta es “**Common Gateway Interface**” y es un protocolo operado por la World Wide Web que le otorga la facilidad de tener interactividad con el “cliente(ó navegador Web)”, este protocolo construye una interfase entre la Página Web-Servidor y Web-Cliente de pequeños programas (ó Scripts), operados en tiempo real, permitiendo el envío de datos desde el “cliente”, para obtener una respuesta en forma interactiva a través de formularios, donde el “cliente” captura los datos solicitados para un proceso. Esta forma de operar en la “Páginas Web” permite una recopilación automática de la información dada directamente por el mismo “cliente”, así como operaciones de consulta y/o modificación de datos desde el “cliente”, la interfase de CGI es usado para operar contadores en las “Páginas Web”, el manejo de Bases de datos, Motores de búsquedas, generadores de e-mail, foros de discusión, chats, comercio electrónico, rotadores de imágenes, juegos en línea y otras aplicaciones mas. Los “Scripts” que manejan el CGI pueden ser generados y compilados en diferentes lenguajes tales como Perl, C, C++, Java, Visual Basic, Delphi, etc., pero lo mas común es que sean compilados en lenguaje Perl.

CSS. Esta herramienta son las iniciales de “**Cascade StyleSheet**” y viene a ser una especificación sobre los estilos físicos de un documento HTML, con el fin de dar una separación definitiva de la lógica(estructura) y el físico (presentación) del documento. El estilo lógico se refiere a la estructura del documento como son : los párrafos, las cabeceras, el cuerpo, etc, entendiéndose solo de la estructura del documento(ignorando por completo a la apariencia final) al revés del estilo físico. El estilo físico se refiere a la apariencia final (ignorando la estructura), entendiéndose solo de tipos de fuentes, tamaño de las fuentes, color de las fuentes y del fondo, etc.. Los estilos lógicos y físicos pueden ser modificados desde aplicaciones en JavaScript incrementando con ello la interactividad en las “Páginas Web”, asimismo CSS opera mediante capas que solucionan problemas de colocar elementos en una posición específica, también adecua márgenes y algunas otras propiedades. Un documento CSS debe acompañar a otro documento en HTML.

XSL. Es una especificación desarrollada dentro de la World Wide Web, usada para aplicar formatos a los documentos XML estandarizadamente, XSL es un lenguaje para definir hojas de estilo, y consta de dos partes:

- a) **Un lenguaje de transformación**, con el cual se logra transformar un documento XML a otro documento XML
- b) **Un lenguaje de formateo**, que contiene un vocabulario XML para especificar Objetos de formateo

Durante el desarrollo de XSL se dejó al lenguaje de transformación como un producto independiente y se le llamó XSLT, de forma que XSL hace uso de XSLT, y XSLT pueda ser usado también por otros lenguajes de formateo. Asimismo durante el desarrollo de XSLT el equipo desarrollador decidió que las instrucciones que hacían el acceso y el moverse por los nodos del documento XML también fueran consolidadas como otro producto independiente y se le denominó XPath, el cual es accesada desde XSLT y XPointer. Esta tecnología aún se encuentra en fase de maduración por lo que aún no se ha extendido su uso.

UNIVERSIDAD AUTÓNOMA DE NUEVO LEÓN[®]
DIRECCIÓN GENERAL DE BIBLIOTECAS

Java. Es un lenguaje clasificado como de multi-propósito y orientado a objetos, y que además es tan versátil que puede operar en la World Wide Web, fue desarrollado por la empresa Sun Microsystems, Inc., a fines de 1993 la empresa Sun encaminó uno de sus proyectos de definición de un lenguaje en el apoyo al desarrollo de la World Wide Web, dicho lenguaje debía ser orientado a la programación en diferentes plataformas de computadoras conectadas a Internet. Java permite crear aplicaciones robustas, así como pequeñas aplicaciones llamadas “applets” que combinadas con el lenguaje HTML otorgan a las “Páginas Web” una presencia dinámica y espectacular. Con características interactiva, con sonido y movimiento, Java es un lenguaje parecido a otros como el C++, pero puede generar aplicaciones para la Web y aplicaciones independientes de la Web, las aplicaciones de Java tienen las siguientes ventajas:

- **Seguridad.** Elimina la posibilidad de ser infectadas por virus almacenados en el disco duro
- **Compatibilidad.** Sus aplicaciones se pueden ejecutar en cualquier tipo de plataforma ya que puede emular a una “Maquina Virtual”
- **Rapidez.** El desarrollo se realiza rápida y fácilmente por la sencillez del código fuente
- **Robustez.** Debido a la emulación de “Máquina Virtual” es casi nula la posibilidad de que se “congelen” las aplicaciones
- **Distribuido.** Al poderse ejecutar sus aplicaciones en distintas plataformas se pueden distribuir las tareas

JavaScript. Este lenguaje es diferente totalmente al lenguaje Java, y fue diseñado para añadir funcionalidades al lenguaje HTML (uso de formularios, ventanas, imágenes cambiantes, controles para video, controles para sonido y últimamente opción de manipular HTML Dinámico), por lo que su código es insertado dentro de los documentos de HTML, por medio de JavaScript las “Páginas Web” pueden operar eventos iniciados por el “cliente” tales como : entrar a un formulario ó hacer un enlace, todo ello sin necesidad de algún tipo de transmisión sino que todo ello es validado en la misma aplicación del lado del “cliente” y al finalizar la operación se hace la transmisión al “servidor”. JavaScript puede operar por medio de funciones, las cuales son iniciadas desde el lado del “cliente” y definidas dentro de las marcas de encabezado(<head>), JavaScript está basado en objetos(NO Orientado a Objetos), dentro de HTML es el ejecutor de “applets” de Java, su programación no soporta la concurrencia ni el acceso compartido. Una característica mas es que los navegadores de Internet deben tener la capacidad de soportar esta herramienta.

VBScript. Desarrollado por Microsoft este lenguaje está orientado al desarrollo de Script's, los cuales son una secuencia de instrucciones dentro del documento HTML que serán interpretados para su ejecución en el computador del "cliente", VBScript es un derivado directo del lenguaje Visual Basic, VBScript es considerado una herramienta de la tecnología ASP de Microsoft y solo funciona correctamente con el navegador Internet Explorer 4.0 y superiores. Este lenguaje de Script's al igual que JavaScript está basado en objetos(NO orientado a Objetos).

Perl. Este nombre es el abreviado de Practical Extracción and Report Language y es un lenguaje dirigido a la manipulación de cadenas de datos y de archivos, facilitando el proceso de grandes volúmenes de información sin sacrificar el rendimiento del computador, este lenguaje no se orienta a una filosofía de programación en particular aunque soporta la programación Modular, Estructurada y la Orientada a Objetos, unas de sus características son la facilidad para su aprendizaje y la claridad en el código de sus aplicaciones, en sus diseños se ignora la distribución de la pantalla en cuanto a botones, ventanas, menús, opciones de selección, etc., sino que su función es el uso de modelos de búsquedas, Perl es un intérprete creado por Larry Wall, y liberado al mundo para ser usado por cualquier persona en forma gratuita, pero con la condición de no utilizarlo con fines de lucro. Existen versiones de Perl para casi todas las plataformas de computadoras y es ampliamente usado en ambiente Internet, aunque su origen fue para la plataforma UNIX.

PHP. Es la abreviación de **PHP:Hypertext Preprocessor**, y representa una tecnología para insertar código en HTML además de ser un lenguaje independiente, PHP es un lenguaje interpretado basado en los lenguajes C/C++ y Java, por lo que su semántica es muy parecida a ellos, así como también características del lenguaje Perl, esto hace posible que quienes de antemano conocen los lenguajes mencionados será sumamente fácil el aprendizaje de PHP, éste lenguaje permite toda clase de facilidades para la conexión a gestores de Bases de datos para su acceso desde las “Páginas Web”, y realizar operaciones comerciales y de negocios electrónicos.

Flash. Es un producto de la empresa Macromedia, que complementa al software “**Director**” en la creación de producciones de multimedia enfocadas a aplicaciones en la Web, Flash diseña gráficas definidas en puntos y líneas en vez de pixeles, donde los vectores son un conjunto de instrucciones matemáticas que por medio de valores le dan forma a una imagen, de esta forma un círculo vectorial puede ser ampliado sin perder su uniformidad, algo que no se logra con el manejo de pixeles. Flash permite incluir audio comprimido en diversos formatos, uno de ellos es MP3, puede importar gráficas creadas con otros programas así como formularios. Flash permite dotar a las “Páginas Web” de animación de imágenes así como de diversos tipos de movimientos, incluyendo al Texto, sin embargo sus aplicaciones quedan almacenadas en archivos muy pequeños que se cargan en poco tiempo, cuando se termina el desarrollo de una aplicación es exportada a un formato comprimido llamado “**Shockwave Flash**” cuya extensión es “.swf”. La utilización de Flash requiere de un “Plug-in” instalado el navegador de Internet. Digno de mencionar respecto de este software es la campaña que se ha generado por su utilización de parte de personas y empresas faltas de ética, ya que mediante el uso de las llamadas pantallas splash e Intro grandes animaciones se cargaban aú antes de desplegarse la “Página de inicio” así como elementos móviles y botones con efectos, generando una completa falta de consideración hacia los usuarios de Internet.

VRML. Esta es la abreviación de “Virtual Reality Modeling Language” ó sea Lenguaje de Modelado para realidad Virtual, fue desarrollado por Mark Pesce, este lenguaje realiza manipulaciones de bits logrando la posibilidad de la visualización tridimensional de texto e imágenes. VRML es un lenguaje para describir simulaciones interactivas, mundos virtuales enlazados a través de Internet e hiperlinks con WWW, todos los aspectos que se muestran en el mundo virtual pueden ser especificados en este lenguaje, los mundos virtuales pueden tener objetos que a través de hiperlinks se enlazan con otros mundos también virtuales y el usuario puede interactuar con ellos sin necesidad de tarjetas ni lentes de Realidad Virtual (VR). Esta es una tecnología emergente que se está difundiendo rápidamente.

ASP. Esta es la abreviación de “Active Server Page” ó Página Activa del Servidor, la cual más que un lenguaje está catalogada como una tecnología, fue desarrollada por la empresa Microsoft, su objetivo es ser un recurso para la creación de “Sitios Web” como marco de aplicaciones para Internet, y se basa en la creación de páginas en HTML que incluye uno ó varios Script pequeños incrustados en la página y que se procesan en el “servidor Web” antes que la Página sea enviada al usuario. Asimismo también se le pueden incluir componentes ActiveX, consiguiendo la generación de páginas dinámicas y potentes aplicaciones Web con acceso a Bases de datos en forma rápida y sencilla, soporta código de VBScript y Jscript, todos ellos son tecnología de la empresa Microsoft, esta tecnología ASP es contraparte de la utilización de Perl y CGI. Una de sus desventajas es que solo puede operar en plataforma Windows NT.

7.3 Intranets

La proliferación de los usuarios de Internet pone de manifiesto el auge que éste ha despertado en todos los ámbitos de la sociedad, la misma sociedad últimamente ha visto las potencialidades que Internet le puede otorgar, las empresas e Instituciones han crecido debido al uso de tecnologías que le han facilitado ese crecimiento particularmente en el área computacional.

En la década de 1970 las empresas e Instituciones tenían grandes computadores centrales que soportaron el crecimiento de las mismas, durante la década de 1980 con la aparición de las computadoras personales muchas de las funciones de oficina fueron automatizadas mediante la utilización de PC's, en la década de 1990 fue un tiempo de auge de interconexión de computadoras aplicando los conceptos de "Red de computadoras" transfiriendo a ellas los procesos que se realizaban en los grandes computadores (ó mainfram's), ya que los éstos requerían de sitios especiales para su instalación y mantenimiento (esto es traducido a costos en la empresa ó Institución), en cambio las redes no tenían ese requerimiento, con ello se redujeron los costos y la función computacional no solo se mantenía sino que también se podía incrementar. ®

Fué en 1995 cuando se inició el uso masivo del Internet ante el éxito de la tecnología W W W, y a partir de entonces las empresas e Instituciones han ido incrementado el uso de estas tecnologías, de forma que ha surgido la teoría de un nuevo concepto llamado "Intranet", mediante el cual se aconseja usar las tecnologías y protocolos de Internet en las redes locales de las empresas e Instituciones para ser usadas por los empleados de forma que dispongan de la información local y remota para desempeñar eficientemente su trabajo.

Las Intranets marcan un avance en el uso de la tecnología al servicio del hombre, y las empresas e Instituciones poco a poco han ido implementando estos conceptos en sus organizaciones, teniendo a los datos como un bien estratégico y haciendo que sus empleados trabajen cooperativamente entre departamentos con flujo de información entre ellos, esto supone el logro de mayores índices de calidad y productividad así como el mejoramiento de las relaciones humanas ante el uso de mejores herramientas.

En la actualidad se dice que nos hallamos en la era de la Sociedad en Red, y que se está sentando las bases para el establecimiento de la economía conectada en red, donde las empresas, Instituciones, clientes, proveedores, consumidores, bancos, etc., se intercomunican a través de medios electrónicos.

Según los conocedores de estas teorías una red Intranet permite con el menor costo posible garantizar la perennidad de las aplicaciones existentes, incrementar la productividad gracias a la circulación optimizada de la información y modificando la percepción de la informática de todos los colaboradores en la empresa convirtiéndolos en miembros activos de una red cooperativa. El punto neurálgico de una Intranet descansa sobre varios servidores que simplifican el acceso a los datos.

Un asunto por demás importante es la seguridad de los datos, ya que Internet ha aumentado la capacidad de las empresas e instituciones para lograr que la información esté disponible para empleados, clientes, proveedores, etc., pero esto aumenta el riesgo de uso incorrecto de los datos, por lo que el reto principal es garantizar que las personas puedan acceder solo la información que necesiten y no poder acceder a la información que no les compete y no estén autorizados para su consulta y/o modificación.

Una de las razones por la que las Intranets han sido ampliamente aceptadas es por el entorno distribuido, ya que los servidores Intranet se ubican estratégicamente en un grupo ó nivel departamental para minimizar los retrasos en el envío de los datos, las primeras empresas que han implementado estos conceptos lo hicieron para convertir su sistema tradicional de distribución de información, basada en papel en un potente

sistema de comunicación electrónico, concentrando sus esfuerzos en información de carácter crítico como ventas, recursos humanos, soporte técnico para ayuda va los usuarios, finanzas, administración de proyectos, etc..

Algunos de los servicios que los usuarios pueden disponer a través de una Intranet son:

- Acceso a Bases de datos
 - Publicación de “Paginas Web” del departamento ó de sus empleados
 - Correo electrónico entre empleados y agentes externos
 - Foros de discusión
 - Conferencias en línea
 - Publicación de documentos
 - Avisos en tableros electrónicos
 - Teléfono en Internet(NetMeeting)
 - Oficina virtual
-
- Comercio electrónico
 - Servicios EDI
 - Transferencia de datos
 - Acceso a los Sistemas Administrativos de la Organización
 - Chat
 - Publicación de manuales de procedimientos
 - Enlaces de inventarios empresa-proveedor-clientes
 - Estadísticas

En general los concedores de esta teoría concluyen que algunos de los beneficios para la empresa ó institución que disponga de una Intranet pueden ser los siguientes:

- Centralizar la información
- Organización de la información
- Disminución de costos y aumento de la eficiencia
- Compartir los recursos computacionales
- Minimizar tiempo y costos de distribución de la información
- Mayor rapidez en el desarrollo de aplicaciones y de su distribución
- Minimizar el uso de papel
- Documentación electrónica de la información de la empresa ó institución

Respecto al software para la programación de las Intranets es el mismo que el usado para la programación en Internet: Protocolos http, TCP/IP, los lenguajes HTML, ASP, Perl, Java PHP, CGI, XML, etc., que interactúan con lenguajes como Visual Basic, Delphi, C++, y otros mas, así como con los manejadores de grandes bases de datos como Oracle, DB2, Progress, Sybase, InterBase, y bases de datos de escritorio como Access, DBase, Paradox, etc.

®

DIRECCIÓN GENERAL DE BIBLIOTECAS

En conclusión a lo anterior debe considerarse la importancia de incluir en el plan de estudios de la carrera de Ingeniero en Sistemas Computacionales del Instituto tecnológico de Nuevo León los temas concernientes al desarrollo de aplicaciones para Intranets, ya que éste será el ambiente al que se enfrentarán los alumnos al concluir su carrera profesional.

8

PROCESOS ADMINISTRATIVOS SUGERIDOS A INCLUIR EN EL PLAN RETICULAR DE LOS INGENIEROS EN SISTEMAS COMPUTACIONALES

En el marco de referencia de las metas y objetivos del Instituto Tecnológico de Nuevo León en cuanto a lograr una excelencia en la preparación académica de sus egresados, para que en forma eficiente coadyuven al progreso de los diferentes sectores productivos en nuestro país, este humilde trabajo de tesis propone la conveniencia de analizar la necesidad de reforzar el perfil del Ingeniero en Sistemas Computacionales con la enseñanza (desde el punto de vista computacional), de diversos procesos administrativos y de producción, tratándolos como un conjunto de procesos que generarán reportes con información clasificada como estratégica.

Actualmente uno de los objetivos pretendidos con el plan reticular de la carrera de Ingeniero en Sistemas Computacionales es que sus egresados puedan en sus centros de trabajo **“Apoyar a la administración con el uso de recursos y sistemas computacionales en la toma de decisiones, y soporte estratégico a la organización”**, para lograr lo anterior se requiere que sus estudiantes conozcan y entiendan los conceptos y el lenguaje de los procesos administrativos, así como los de producción.

La enseñanza de dichos procesos hará surgir en el alumno una comprensión de la magnitud e importancia del cuidado y de la correcta utilización de los recursos tangibles e intangibles de una empresa(ó institución), la propuesta de esta tesis es que los procesos sean estudiados desde el punto de vista computacional de la información:

- Datos que se generan
- Formatos adecuados para su registro manual y computacional
- Almacenamiento computacional
- Flujo de la información
- Aportación a información estratégica
- Afectaciones colaterales entre procesos

Buscando opinión del sector productivo de la región para esta propuesta se concensó a un grupo de empresas del área metropolitana de Monterrey Nuevo León, mediante un cuestionario donde se incluyó una pregunta donde se inquiría si se consideraba necesario que un Ingeniero en Sistemas Computacionales conociera de procesos administrativos y de producción, y se mencionaron como ejemplo los siguientes:

- Ventas
- Producción
- Contable
- Costos
- Almacén
- Recursos Humanos

a lo cual el 92% de las empresas encuestadas dio una respuesta afirmativa, y una de ellas del sector de consultorías computacionales agregó lo siguiente : **“Es importante el conocimiento de todas las áreas ya que sistemas puede intervenir en cualquier área para mejorar procesos”**.

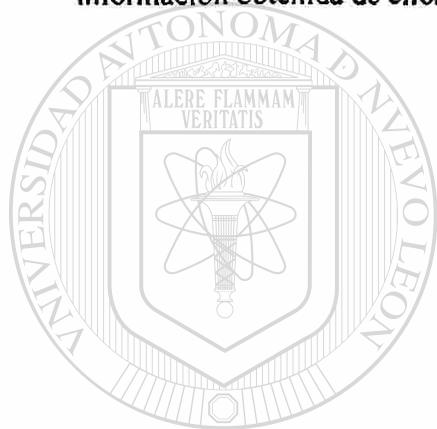
A efecto del análisis de la propuesta es importante considerar en el mismo, que la preparación académica de los alumnos es una capacitación para mejorar el desempeño de sus actividades tanto personales como colectivas en su trabajo, de manera que si hubiese alumnos que paralelamente estudien y trabajen el conocimiento de los procesos les proporcionará herramientas para comprender mejor las actividades que realizan y como coadyuvan a las metas del negocio donde trabajan.

Para el caso de alumnos con carácter de emprendedores les es conveniente el conocimiento de los objetivos y detalles de dichos procesos para visualizar su posible implantación en el arranque de un negocio.

Asimismo para los alumnos que ya trabajen (ó recién se contraten en una empresa) cuyas actividades laborales se desarrollen en el ámbito de un departamento de desarrollo y mantenimiento de sistemas computacionales les es conveniente el conocimiento de dichos procesos para “entender el lenguaje y las necesidades del cliente”.

8.1 Objetivo de la propuesta

Complementar el perfil de los egresados de la carrera de Ingeniería en Sistemas Computacionales del Instituto Tecnológico de Nuevo León, añadiendo en ellos la habilidad de conocer y comprender el lenguaje técnico y los procesos relativos a las áreas administrativas y de producción, de forma que puedan ser soporte para el diseño, desarrollo, mantenimiento y posibles mejoras de sus procesos sistematizados y la información obtenida de ellos.



UANL

UNIVERSIDAD AUTÓNOMA DE NUEVO LEÓN



DIRECCIÓN GENERAL DE BIBLIOTECAS

8.2 Problemática a resolver

Actualmente en la retícula ISIC-1993-296 no se incluye ninguna materia que incluya el tratamiento de los temas relativos a la enseñanza de procesos administrativos y de producción, por lo que al egresar los alumnos de Ingeniería en Sistemas Computacionales estos conceptos los irán aprendiendo en sus centros de trabajo en la medida que se presenten ocasiones de tratamiento computacional de este tipo de datos, pero esto tiene el inconveniente que durante un buen tiempo dependerán de otras personas que puedan y quieran “enseñarles” (sobre la marcha) los procesos y el flujo de los datos que en ellos se generan, así como la visión general de sus aplicaciones.



UANL

UNIVERSIDAD AUTÓNOMA DE NUEVO LEÓN

®

DIRECCIÓN GENERAL DE BIBLIOTECAS

8.3 Temas sugeridos

I Clasificación y codificación de datos

Objetivos: Que los alumnos conozcan y dominen las metodologías para la clasificación de datos, de manera que estos puedan identificar el tipo de transacciones realizadas, asimismo que los datos puedan registrarse en un sistema de cómputo de manera predeterminada, orientada a la solución de necesidades futuras de procesamiento y recuperación.

II Abastecimientos

Objetivo: Que los alumnos conozcan y dominen los conceptos, metodologías y formatos de control, y de datos, orientados a su uso computacional, usados para el asegurar el aprovisionamiento de partes y materiales para la fabricación de productos, procurando que estén disponible al tiempo de su requerimiento.

III Inventarios

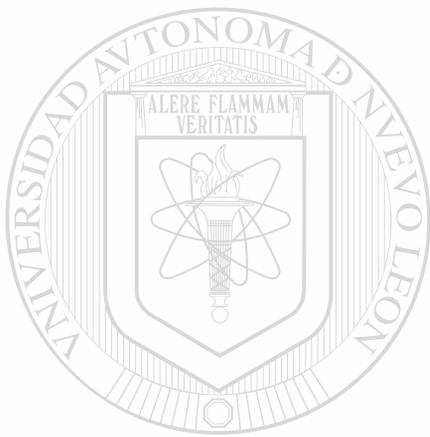
Objetivos: Que los alumnos conozcan y dominen los conceptos, metodologías y formatos de datos, orientados a su uso computacional, usados para controlar y costear los bienes tangibles de una empresa.

IV Administración

Objetivo: Que los alumnos conozcan y dominen los conceptos, metodologías y formatos de datos, orientados a su uso computacional, usados por los administradores para determinar una visión global de los factores financieros de una empresa.

V Remuneración de los Trabajadores

Objetivo: Que los alumnos conozcan y dominen los conceptos, leyes y reglamentos, formatos de datos y métodos de cálculo de los factores básicos que intervienen en el cálculo de un sistema de retribución de los trabajadores de una empresa, de forma que puedan ser usados en los diseños computacionales.



UANL

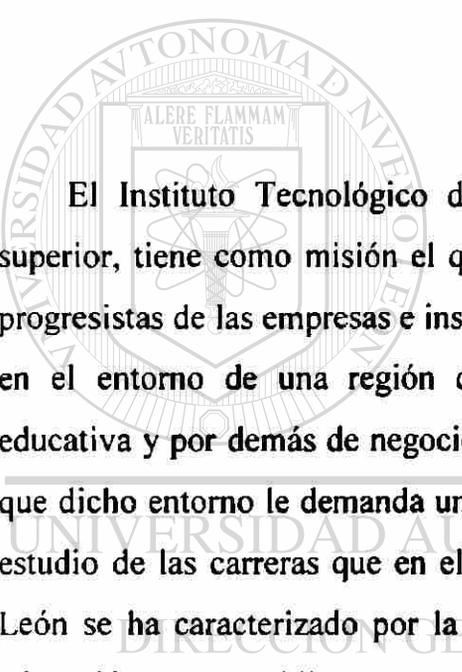
UNIVERSIDAD AUTÓNOMA DE NUEVO LEÓN



DIRECCIÓN GENERAL DE BIBLIOTECAS

9

CONCLUSIONES Y RECOMENDACIONES



El Instituto Tecnológico de Nuevo León siendo una Institución de educación superior, tiene como misión el que sus alumnos sean soporte y apoyo en los esfuerzos progresistas de las empresas e instituciones públicas de nuestro país, asimismo se ubica en el entorno de una región de México con gran actividad industrial, comercial, educativa y por demás de negocios como es el pujante estado de Nuevo León, de forma que dicho entorno le demanda una actualización constante de los planes y programas de estudio de las carreras que en el Instituto se imparten. Asimismo el estado de Nuevo León se ha caracterizado por la existencia de una gran diversidad de instituciones de educación tanto públicas como privadas en todos los niveles de la enseñanza, esto implica que el Instituto Tecnológico de Nuevo León necesita de las ventajas competitivas a ofrecer a la comunidad, de forma que éstas marquen una diferencia con respecto a otras universidades. En múltiples foros y conferencias organizadas por el departamento de Sistemas y Computación del instituto los conferencistas han expresado la conveniencia de que los egresados de las carreras cuya especialidad sean los sistemas computacionales conozcan los conceptos, métodos y flujo de los datos generados en los procesos de un negocio, para que puedan entender las necesidades específicas de los usuarios en cuanto al manejo y generación de información necesaria para las actividades del negocio, por lo que esta propuesta también fue incluida en el presente trabajo.

9.1 Conclusiones

De acuerdo a lo expuesto en la presente Tesis podemos concluir que en los últimos 10 años el campo de la programación de computadoras ha sufrido una evolución de grandes repercusiones, originada por el desarrollo y aprovechamiento de nuevas tecnologías, asimismo han surgido nuevos métodos y técnicas que han revolucionado la forma de realizar aplicaciones para el manejo de la información de todo tipo: científico, educativo y de negocios.

El uso de herramientas tecnológicas de actualidad producen una ventaja competitiva a las instituciones privadas ó públicas que las tienen implementadas y las usan de forma corriente, asimismo debido a la globalización de mercados, y al uso de dichas tecnologías les permite a dichos organismos la asociación y/o cooperación productiva con organismos similares de otros países.

Lo anterior fue el pensamiento que dio origen al proyecto de esta Tesis, ya que en la actualidad los organismos públicos y privados realizan el desarrollo de sus aplicaciones computacionales con manejadores de Bases de datos y lenguajes de cuarta generación(4GL's), por lo que era necesaria una propuesta académica en ese sentido, además el uso de la tecnología Internet en varias empresas de la localidad ha marcado el inicio de los e-business, así como el uso de las Intranet y Extranet.

9.2 Recomendaciones

En los puntos siguientes se enuncian sugerencias y recomendaciones de temarios de materias cuyo contenido representa una actualización respecto a los temas y materias del actual plan académico de la carrera de Ingeniería en Sistemas Computacionales del Instituto Tecnológico de Nuevo León.

Lo anterior representa una pequeña aportación de este trabajo de tesis para fortalecer el perfil de los alumnos que egresen de la especialidad mencionada, sin embargo necesario es hacer mención que en el ambiente computacional los cambios tecnológicos se suceden ó evolucionan con mucha rapidez, por lo que los métodos y técnicas que por ahora son “de actualidad” en un futuro cercano pudieran pasar a ser “obsoletas”, y surgir nuevos métodos ó técnicas nuevas.

Algunas empresas generadoras de tecnologías para software tales como Microsoft , Borland y otras mas actualmente investigan nuevas tecnologías, por lo que es recomendable que proyectos como el presente se realicen con frecuencia, de manera que se analizen los nuevos métodos y técnicas emergentes, así como la posibilidad de integrarlos a los planes de estudio. Lo anterior es una recomendación para los docentes y directivos de los planteles de educación superior.

9.2.1 Propuesta académica del primer lenguaje de programación

Esta propuesta sugiere que la enseñanza de programación de computadoras en los alumnos de la carrera de Ingeniería en Sistemas Computacionales dé inicio con el apoyo del lenguaje C++ Builder, ya que dicho lenguaje incluye en si mismo todas las técnicas y metodologías nuevas de la programación, se propone que la enseñanza de este lenguaje se lleve a cabo en el transcurso de dos periodos escolares, divididos en dos materias secuenciadas:

- **INTRODUCCIÓN A LA PROGRAMACION**

Aportación de la asignatura al perfil del egresado:

Esta asignatura le aportará al alumno los conocimientos básicos y habilidades metodológicas para el diseño de soluciones mediante el uso de los recursos computacionales.

Objetivo general del curso :

El alumno aprenderá a plantear soluciones computacionales mediante el diseño de algoritmos, y aprenderá uso y sintaxis de estatutos básicos de uno de uno de los lenguajes computacionales mas versátiles, que incluye las mas actuales técnicas de programación.

- **PROGRAMACIÓN I**

Aportación de la asignatura al perfil del egresado:

Esta asignatura aportará al alumno los conocimientos y habilidades de las técnicas de programación orientada a objetos, y la programación visual.

Objetivo general del curso :

Continuidad en el aprendizaje del lenguaje C/C++, añadiendo los conceptos de la programación orientada a objetos, así como la forma de usarla para construir aplicaciones en ambiente visual.

TABLA 12 Temario para la Materia Introducción a la Programación

Unidad	Temas	Subtemas
1	Introducción	1.1 Metodologías de la programación 1.2 Proceso del desarrollo de programas 1.3 Técnicas para desarrollo de Algoritmos
2	Elementos de un programa	2.1 Constantes, variables, declaraciones 2.2 Identificadores, comentarios 2.3 Operadores aritméticos, de relación 2.4 Operadores unarios, de ámbito, lógicos 2.5 Expresiones, Tipos de datos 2.6 Cadenas, secuencias de escape 2.7 Esquema general de un programa C/C++
3	Ambientes de desarrollo de programas	3.1 Ambiente IDE 3.2 Ambiente DOS
4	Estatutos de entrada y salida estándar de datos C/C++	4.1 cin, cout, cin.getline, manipuladores 4.2 scanf y printf 4.3 Componentes VCL Standard
5	Estructuras selectivas	5.1 Estatuto if, if-else, if's anidados 5.2 Operador coma 5.3 Estatutos switch, break, continue
6	Manejo de excepciones	6.1 try.....catch
7	Estructuras de control	7.1 Estatutos for, while, do-while 7.2 Bucles anidados
8	Funciones	8.1 De Biblioteca y definidas por el usuario 8.2 Declaración y definición de funciones 8.3 Ámbito de variables 8.4 Transferencia de parámetros por valor 8.5 Transferencia de parámetros por referencia 8.6 Recursividad 8.7 Funciones en línea 8.8 Funciones sobrecargadas

Materia precedente : ninguna

Materia subsecuente : Programación I

Recomendación : Las unidades 3, 4, 5, 6, 7 y 8 deben ser reforzadas con prácticas de Laboratorio, mediante aplicaciones en lenguaje C/C++, en ambiente D.O.S., así como el desarrollo de aplicaciones visuales sencillas en ambiente IDE.



UANL

UNIVERSIDAD AUTÓNOMA DE NUEVO LEÓN

®

DIRECCIÓN GENERAL DE BIBLIOTECAS

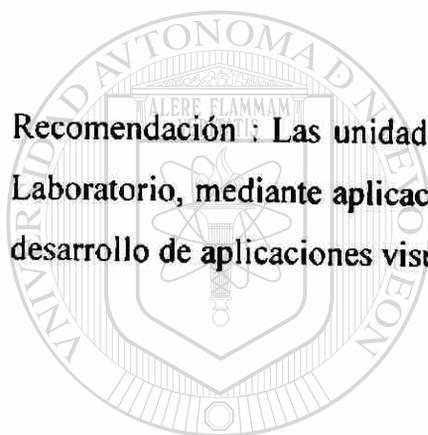
Tabla 13 Temario para la materia Programación I

Unidad	Temas	Subtemas
1	Arreglos	1.1 Unidimensionales, Multidimensionales 1.2 Arreglos entre funciones
2	Punteros	2.1 Definición y declaración 2.2 Operadores dirección, indirección 2.3 Aritmética de punteros 2.4 Punteros, arreglos y funciones
3	Estructuras y uniones	3.1 Definición y declaración de Estructuras 3.2 Procesamiento de Estructuras 3.3 Datos definidos por el usuario (typedef) 3.4 Estructuras y punteros 3.5 Estructuras y funciones 3.6 Estructuras enlazadas
4	Programación orientada a objetos	4.1 Clases y objetos, Abstracción, Herencia 4.2 Encapsulamiento, Polimorfismo 4.3 Constructores y destructores 4.4 Puntero this
5	Programación orientada a eventos	5.1 Componentes VCL Additional 5.2 Diseño de Menús 5.3 Aplicaciones visuales básicas
6	Archivos	6.1 Componentes VCL Dialogs 6.2 Archivos de Texto 6.3 Archivos binarios 6.4 Clases ofstream, ifstream, fstream 6.5 Salida de datos a impresora

Materia precedente : Introducción a la programación

Materia subsecuente : ninguna

Recomendación : Las unidades 1, 2, 3, 4, 5, y 6 deben ser reforzadas con prácticas de Laboratorio, mediante aplicaciones en lenguaje C/C++, en ambiente D.O.S., así como el desarrollo de aplicaciones visuales (propias a los temas) en ambiente IDE.



UANL

UNIVERSIDAD AUTÓNOMA DE NUEVO LEÓN



DIRECCIÓN GENERAL DE BIBLIOTECAS

9.2.2 Propuesta académica para el segundo lenguaje y las Bases de datos

De acuerdo a lo expresado en el capítulo 6 respecto al segundo lenguaje de programación y las Bases de datos se concluye recomendar la enseñanza del lenguaje SQL, el cual es usado en la mayoría de los manejadores de Bases de datos, y además es el más usado en las empresas del área metropolitana de Monterrey Nuevo León, donde está ubicado el Instituto Tecnológico de Nuevo León.

Lo anterior nos sirve de confirmación de los planes académicos que ha llevado a cabo la academia del departamento de Sistemas y Computación, ya que la enseñanza del SQL es llevado a cabo en las materias de Bases de datos I y Bases de datos II, donde se enseña la utilización de instrucciones correspondientes al Lenguaje de descripción de datos(DDL), así como las instrucciones correspondientes al Lenguaje de manipulación de datos(DML).

En este trabajo de Tesis solo podríamos recomendar lo siguiente: ya que el lenguaje SQL es usado ampliamente por la mayoría de los Servidores de Bases de datos en el mercado computacional, asimismo por todos los lenguajes de cuarta generación, es recomendable que se establezcan sesiones de laboratorio donde mediante el uso de algunos lenguajes como :

- Visual Basic
- Delphi
- C++ Builder
- Visual C++

se realicen prácticas de generación y manipulación de Bases de datos usando SQL.

Se recomienda la enseñanza del estándar SQL'92 de la ANSI.

9.2.3 Recomendación para incluir la enseñanza del lenguaje Java

Java es una potente herramienta de programación de computadoras, usa los conceptos de la programación orientada a objetos, así como también los conceptos de la programación orientada a eventos, este lenguaje en su diseño parte de las bases de C++, y con esta herramienta es posible diseñar y desarrollar aplicaciones de cualquier tipo, igual que con otros lenguaje como C/C++.

El lenguaje Java tiene como característica que sus ejecutables son independientes de plataforma computacional, de manera que sus ejecutables pueden ser directamente ejecutables en computadoras de cualquier tipo y marca, su entorno de desarrollo es visual. Java es ampliamente usado en el desarrollo de aplicaciones en ambiente Internet/Intranet.

Por lo anterior, y de acuerdo a lo descrito en el capítulo 7 este trabajo de Tesis recomendamos la enseñanza de este lenguaje de manera que los alumnos del Instituto Tecnológico de Nuevo León sean capacitados con herramientas computacionales de gran actualidad y sean factores de respuesta a necesidades actuales en el campo de la informática.

Aportación de la asignatura al perfil del egresado:

El alumno podrá analizar y desarrollar aplicaciones de multiplataformas, así como aplicaciones en ambiente Web.

Objetivo general del curso :

El alumno aprenderá a usar uno de los lenguajes mas versátiles orientados al análisis y diseño de aplicaciones Web.

9.2.3 Recomendación para incluir la enseñanza del lenguaje Java

Java es una potente herramienta de programación de computadoras, usa los conceptos de la programación orientada a objetos, así como también los conceptos de la programación orientada a eventos, este lenguaje en su diseño parte de las bases de C++, y con esta herramienta es posible diseñar y desarrollar aplicaciones de cualquier tipo, igual que con otros lenguaje como C/C++.

El lenguaje Java tiene como característica que sus ejecutables son independientes de plataforma computacional, de manera que sus ejecutables pueden ser directamente ejecutables en computadoras de cualquier tipo y marca, su entorno de desarrollo es visual. Java es ampliamente usado en el desarrollo de aplicaciones en ambiente Internet/Intranet.

Por lo anterior, y de acuerdo a lo descrito en el capítulo 7 este trabajo de Tesis recomendamos la enseñanza de este lenguaje de manera que los alumnos del Instituto Tecnológico de Nuevo León sean capacitados con herramientas computacionales de gran actualidad y sean factores de respuesta a necesidades actuales en el campo de la informática.

Aportación de la asignatura al perfil del egresado:

El alumno podrá analizar y desarrollar aplicaciones de multiplataformas, así como aplicaciones en ambiente Web.

Objetivo general del curso :

El alumno aprenderá a usar uno de los lenguajes mas versátiles orientados al análisis y diseño de aplicaciones Web.

Tabla 14 Temario para la Materia Programación Avanzada

Unidad	Tema	Subtemas
1	Introducción al lenguaje Java	1.1 Características de Java 1.2 Elementos básicos de Java
2	Programación Orientada a objetos	2.1 Clases ,y Objetos, instancias 2.2 Herencia, Polimorfismo 2.3 Constructores, destructores
3	Paquetes	3.1 Control de acceso
4	Identificadores y clases especiales	4.1 null, this, super 4.2 Math, character, string
5	Manejo de Errores	5.1 try...catch...finally 5.2 Clase Exception 5.3 Captura de excepciones
6	Applets	6.1 Ciclo de vida de un applet 6.2 Etiqueta Applet 6.3 Clase Graphics 6.4 Ejecución multihilos 6.5 Eventos
7	Operación en red	7.1 Conectividad 7.2 Puertos 7.3 Operaciones con URL's 7.4 Sokets 7.5 Datagramas cliente/servidor 7.6 Seguridad

Conocimientos precedentes : se requiere conocer el lenguaje C/C++

Se recomienda programar sesiones de laboratorio para el grupo de clase, para ejemplificar los temas por medio de aplicaciones en el laboratorio de cómputo.

9.2.4 Recomendación para incluir una materia de especialidad como “Lenguajes de programación emergentes”

En el capítulo 7 de este trabajo de Tesis se menciona que en el área computacional de las instituciones públicas y privadas se están gestando cambios sustanciales en la forma de operar los sistemas de información y la conectividad a través de redes, de forma que la información puede ser compartida entre empleados(en forma local) y con agentes externos(en forma remota), implementando en redes locales la tecnología Internet, generando de esta forma el concepto de redes Intranet/Extranet.

Dado lo anterior se recomienda la inclusión de una materia de especialidad cuyo objetivo sea la enseñanza básica e introductoria de tres lenguajes usados para el desarrollo de aplicaciones en ambiente Web, como son:

- Lenguaje HTML
- Lenguaje Perl
- Lenguaje PHP

Aportación de la asignatura al perfil del egresado:

El alumno conocerá diferentes herramientas de programación de gran actualidad para realizar aplicaciones en ambiente Internet/Intranet..

Objetivo general del curso :

El alumno conocerá varios lenguajes orientados al análisis y diseño de aplicaciones Internet/Intranet.

Tabla 15 Temario para la Materia de Lenguajes de Programación emergentes

Unidad	Temas	Subtemas
1	Introducción al lenguaje HTML	1.1 Introducción y Diseño básico 1.2 Manejo de Textos y de gráficos 1.3 Manejo de tablas y Formas 1.4 Frames 1.5 Introducción a JavaScript
2	Introducción al lenguaje Perl	2.1 Filosofía de Perl 2.2 Elementos básicos 2.3 Operaciones básicas 2.4 Estructuras de control 2.5 Operaciones con archivos 2.6 Perl en línea 2.7 Programación CGI en Perl
3	Introducción al lenguaje PHP	3.1 Constantes y variables 3.2 Operadores y expresiones 3.3 Estructuras de control 3.4 Funciones 3.5 Clases y Objetos 3.6 Manejo de errores 3.7 Creación de imágenes GIF 3.8 Autenticación 3.9 Archivos remotos 3.10 Conexión a Bases de datos

9.2.5 Recomendación para incluir una materia de humanidades como “Comunicación efectiva ISC/usuario”.

En el capítulo 8 de este trabajo de Tesis se reconoce la falta de una materia que proporcione a los alumnos el conocimiento de conceptos y métodos administrativos que se usan en las empresas (visto todo ello desde el punto de vista computacional), que a través de la misma el alumno aprenda el “lenguaje de la administración de negocios”, y con ello entienda el lenguaje y necesidades de los usuarios.

Aportación de la asignatura al perfil del egresado:

El alumno entenderá fielmente a sus usuarios al conocer y aplicar su mismo lenguaje, de esta forma podrá diseñar soluciones tal y como el usuario los necesita.

UNIVERSIDAD AUTÓNOMA DE NUEVO LEÓN

Objetivo general del curso :

El alumno conocerá los diferentes conceptos de los procesos administrativos y de producción, así como el flujo de los datos entre ellos, de forma que entienda la función de cada uno de ellos y los ubique en el plano de los negocios.

Tabla 16 Temario para la Materia Comunicación efectiva ISC/usuario

Unidad	Tema	Subtema
1	Clasificación y codificación de datos	1.1 Código secuencial 1.2 Código de bloques 1.3 Código fonético 1.4 Catalogo de cuentas 1.5 Catalogo de clientes
2	Abastecimientos	3.1 Proveedores 3.2 Registro de compromisos de pago 3.3 Ordenes de compra 3.4 Gastos menores 3.5 Requerimientos 3.6 Catalogo de proveedores 3.7 Condiciones de pago
3	Inventarios	4.1 Ordenes de trabajo 4.2 Recepción, almacen y embarques 4.3 Costo estandar y semiestandar 4.4 Costos históricos
4	Administración	5.1 Contabilidad 5.2 Facturación 5.3 Cuentas por cobrar 5.4 Administración del efectivo
5	Remuneración de los trabajadores	6.1 Aspectos legales (Artículo 123) 6.2 Disposiciones de la Ley Federal del Trabajo 6.3 Calculo de una Nomina

Bibliografía

Textos:

Curso de programación, de los autores Jorge Castro, Felipe Cucker, Xavier Messenger, Albert Rubio y Luis Solano. de la editorial Mc Graw Hill, primera edición.

Su compañía de manufactura, manual de la administración de la pequeña y mediana industria, del autor Robert Crinkley A., de la editorial Mc Graw Hill, primera edición.

Internet gráfico, del autor Gonzalo Ferreira Cortés, de la editorial Computec, primera edición.

Intelligent Software Systems Development an IS Manager's Guide, de los autores Paul Harmon & Curtis Hall, de la editorial Jhon Wiley & Sons, primera edición.

Programación orientada a objetos, Modelado, diseño y codificación en C++, del autor Luis Joyanes Aguilar, de la editorial Mc Graw Hill, primera edición.

Procesamientos de bases de datos, Fundamentos, diseño e implantación, del autor David M. Kroenke, de la editorial Prentice Hall, primera edición.

Elementos de Computación, del autor Guillermo Levine, editorial Mc Graw Hill, primera edición.

Fascículos 1,2 y 18 de la serie Curso IBM de Programación, de la editorial Planeta, de Ediciones Multimedia.

Artículos A58359773

Las Bases de datos en el mercado, de la editorial Sayrols S.A. de C.V. de Febrero de 1999, disponible en Infotrac de Bivitec (Sistema Nacional de Institutos Tecnológicos).

Páginas Web

<http://members.es.tripod.de/adm/popup/roadmap-search.html>

<http://www.recurso-as400.com>

<http://audisoft.com>

<http://programacion.net/cursos/vbcliserv>

<http://www.concytec.gob.pe/institucional/servidor.htm>

<http://www.angel.perez.net/MODINET/desarr.htm>

<http://www.terabyte.cl/cobolwow.htm>

<http://www.cercosoft.com/wowprog.htm>

<http://acucorp.com/company/Español/acugtmain-span.htm>

<http://escobol.com/cow-acu.htm>

<http://www.merant.com/products/microfocus>

<http://ar.clarin.com/suplementos/informatica/2001-06-13/f-278575.htm>

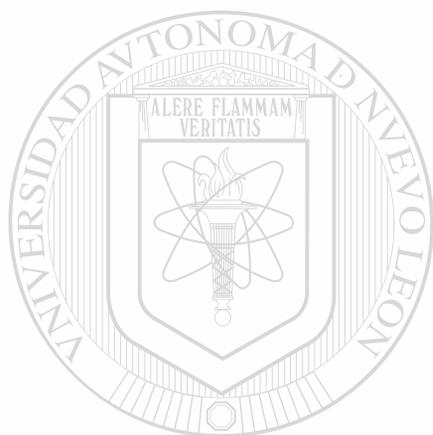
LISTADO DE TABLAS

	Página
Tabla 1 Tipos de empleos de los alumnos de primer ingreso que trabajan	65
Tabla 2 Algunas características entre versiones de Visual Basic 6.0	84
Tabla 3 Algunas características entre versiones de Delphi 5.0	87
Tabla 4 Algunas características entre versiones de C++ Builder 5.0	90
Tabla 5 Requerimientos de Recursos para el Lenguaje Visual Basic	98
Tabla 6 Requerimientos de Recursos para el Lenguaje C++ Builder	98
Tabla 7 Requerimientos de Recursos para el Lenguaje Delphi	99
Tabla 8 Requerimientos de Recursos para el Lenguaje Visual C++	99
Tabla 9 Cuadro comparativo de cumplimiento a los criterios de selección	104
Tabla 10 Empresas y sus productos DBMS	129
Tabla 11 Motores de Bases de datos	138
Tabla 12 Temario para la Materia de Introducción a la Programación	170
Tabla 13 Temario para la Materia Programación I	172
Tabla 14 Temario para la Materia Programación avanzada	176
Tabla 15 Temario para la Materia de Lenguajes de programación emergentes	178
Tabla 16 Temario para la Materia Comunicación efectiva ISC/usuario	180

LISTADO DE GRAFICAS

	Página
Fig. 1 Cronograma de las generaciones de las computadoras	14
Fig. 2 Esquema del Proceso de la Programación en Lenguaje de máquina	32
Fig. 3 Esquema del Proceso de la programación en lenguaje ensamblador	34
Fig. 4 Esquema del Proceso de compilación y ejecución de un programa	37
Fig. 5 Perspectiva histórica de algunos lenguajes de programación	44
Fig. 6 Perspectiva histórica de los lenguajes Orientados a Objetos	46
Fig. 7 La evolución del Software	47
Fig. 8 Secuencia de la enseñanza de la programación a alumnos de I.S.C. en los 2 Primeros semestres	51
Fig. 9 Estadísticas del nivel de reprobación en Diseño Estructurado de Algoritmos	54
Fig. 10 Índice de reprobación en Materia de Diseño Estructurado de Algoritmos	55
Fig. 11 Nivel de aprovechamiento en la materia de Programación I carrera I.S.C.	56
Fig. 12 Comparativo de nivel de aprovechamiento en Programación I.S.C. y I.E.	57
Fig. 13 Comparativo de aprovechamiento entre hombres y mujeres	58
Fig. 14 Estadística Aprobados y Reprobados alumnos de primer ingreso por semestre	59
Fig. 15 Porcentajes en rangos de edades alumnos de primer ingreso	60
Fig. 16 Porcentajes de hombres y mujeres en alumnos de primer ingreso	61
Fig. 17 Conocimientos previos en Diseño de Algoritmos en alumnos de primer ingreso	62
Fig. 18 Conocimientos previos en programación alumnos de primer ingreso	63
Fig. 19 Porcentaje de alumnos de primer ingreso que estudian y trabajan	64
Fig. 20 Factores que inciden en el índice de reprobación en materia de Diseño Estructurado de Algoritmos	68
Fig. 21 Estructura del lote de un Proceso de Compilación	71
Fig. 22 Segmento de un archivo Job Control	73
Fig. 23 Datos distribuidos en un Sistema de Procesamiento de archivos	111

	Página
Fig. 24 Datos integrados en un Sistema de Base de datos	112
Fig. 25 Arquitectura de un DBMS	115
Fig. 26 Tablas del Modelo Relacional	117
Fig. 27 Esquema del Modelo de Red	119
Fig. 28 Esquema del Modelo Jerárquico	121



UANL

UNIVERSIDAD AUTÓNOMA DE NUEVO LEÓN



DIRECCIÓN GENERAL DE BIBLIOTECAS



APÉNDICE 1

Encuesta sobre identificación de rasgos característicos de los alumnos de primer ingreso en la carrera de I.S.C.

UNIVERSIDAD AUTÓNOMA DE NUEVO LEÓN



DIRECCIÓN GENERAL DE BIBLIOTECAS

01. Escuela preparatoria de Procedencia? _____

2. Sexo? Masculino Femenino

03. Edad? 17 – 19 años 20 – 25 años 25 – 30 años

04. Turno? Matutino Vespertino

05. Además de la escuela atiendes a un trabajo?

No

Sí En que horario: _____

Qué funciones tienes en tu trabajo? _____

UNIVERSIDAD AUTÓNOMA DE NUEVO LEÓN



DIRECCIÓN GENERAL DE BIBLIOTECAS

06. Porqué elegiste estudiar la carrera de Ingeniería en Sistemas Computacionales?

07. Al ingresar al I.T.N.L. ya conocías algo del área de Sistemas?

- Programación Redes
 Mantenimiento de PC's Paquetería (Word,Excel,PowerPoint,Lotus)
 Nada Otros(Especifique) _____

08. Al ingresar al I.T.N.L. conoces lo relativo al diseño de Algoritmos?

- Mucho Poco Nada

09. Antes de ingresar al I.T.N.L. ya habías tomado algún curso de programación?

- No
 Si

Qué Lenguaje? _____

En qué Escuela? _____

10. Sabes qué es un código de pieza y para qué sirve? Si No

11. Sabes qué es una Factura y para qué sirve? Si No

12. Sabes qué es una póliza de Diario? Si No

13. Sabes qué es un Salario neto? Si No

14. Sabes qué es un costo indirecto? Si No

15. Dominas el idioma inglés? Si No

Estudias el idioma inglés? Si No



APÉNDICE 2

Encuesta para la identificación de Plataformas de operación computacional y lenguajes de desarrollo mas usados en las empresas del área metropolitana de Monterrey Nuevo León.

UNIVERSIDAD AUTÓNOMA DE NUEVO LEÓN

®

DIRECCIÓN GENERAL DE BIBLIOTECAS

01 Nombre de la Empresa ó Institución:

02 Indique con una X el(los) tipo(s) de Plataforma(s) computacionales usadas en su empresa.

PC's Workstations Servidores Minis Mainframes

03 Indique con una X el(los) lenguajes que usan en su empresa para el desarrollo de aplicaciones.

Cobol C/C++ Visual Basic Delphi

RPG Java C++ Builder Power Builder

Fox Windows Visual Fox Visual C++

Otros(especifique): _____

04 Indique con una X el(los) sistema(s) operativo(s) que usan en su empresa

Windows Unix. Linux

Otros(especifique): _____

05 Indique con una X el(los) manejador(es) de Bases de datos usados en su empresa

SQL Oracle Progress Access

Otros(especifique): _____

06 Que procesos administrativos cree que debe conocer un Ingeniero en Sistemas recién egresado de la escuela?

Ventas Producción Contable Costos

Almacén Recursos Humanos Finanzas

Otros(especifique): _____

07 Indique con una X la(s) herramientas para desarrollo en Internet

HTML XML Java

Otros(especifique): _____

GLOSARIO

<u>Término</u>	<u>Descripción</u>
Batch	Proceso computacional de ejecución diferida, su ejecución se realizará en tiempo posterior a su envío.
Bifurcación	Tomar un rumbo ó dirección de entre varios posibles
Byte	Conjunto de 8 bits
Chip	Pequeño trozo de cristal semiconductor que contiene un conjunto de circuitos transistores integrados.
Comando	Instrucción ejecutable
e-business	Metodología electrónica para manejo de negocios
e-commerce	Metodología electrónica para operaciones comerciales
Extranet	Redes computacionales de proveedores, respecto a una Intranet.
Flip-flop	Tecnología electrónica de operación binaria (0,1).
Hipervínculo	Enlace electrónico entre páginas web.
Indexado	Tipo de archivo de datos con manejo de índices
Interface	Equivalente de interfaz, lo que permite la integración ó comunicación entre dos ó mas partes.
Macintosh	Modelo de micro-computadora fabricada por la empresa Apple
Mainframe	Computador de gran tamaño y gran capacidad, propio para las grandes empresas.
Monousuario	Operable por un solo usuario a la vez.
Nodo	Equipo computacional que forma parte de una red de computadoras.
Página web	Publicación electrónica a través de Internet.
Paradigma	Forma ó manera de pensar y actuar.
Píxel	Elemento mas pequeño de una imagen digital.
Procedural	Que opera bajo procedimientos definidos
Propedéutico	Curso de inducción de una área académica
Retícula	Plan de estudios, ó conjunto de programas de estudios de una especialidad académica.
Servidor web	Computadora cuya función es dar servicio a usuarios para las publicaciones electrónicas (ó paginas web).
Site	Lugar ó ubicación donde un computador está instalado.
Sitio web	Conjunto de paginas web que definen una publicación electrónica.
World Wide Web	Tecnología que opera en Internet propia para publicaciones y navegación electrónicas.
Wizard	Asistente ó ayudante electrónico contenido en un producto de software.

Resumen Autobiográfico

Datos generales:

Nombre: Fidel García Ochoa

Profesión: Ingeniero Administrador de Sistemas

Lugar y Fecha de Nacim. : San Pedro de las Colonias Coahuila, 6 / Octubre / 1951

Padre : Sr. Fidel García Juárez (q.e.p.d.)

Madre: Sra. Cristina Ochoa Agüero (q.e.p.d.)

Tema de Tesis : La enseñanza de la programación de computadoras en el Instituto Tecnológico de Nuevo León al inicio del segundo milenio.

Para la obtención de grado de Maestro en Ciencias de la Administración con especialidad en Sistemas.

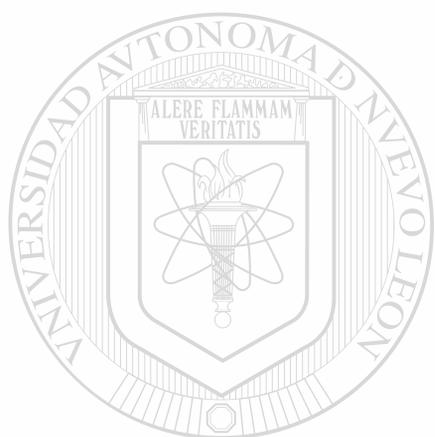
Estudios:

Profesionales: Facultad de Ingeniería Mecánica y Eléctrica
Universidad Autónoma de Nuevo León

Título obtenido : Ingeniero Administrador de Sistemas

Postgrado: Facultad de Ingeniería Mecánica y Eléctrica, División de Postgrado
Universidad Autónoma de Nuevo León

Título a obtener: Maestro en Ciencias de la Administración con especialidad en Sistemas



UANL

UNIVERSIDAD AUTÓNOMA DE NUEVO LEÓN

®

DIRECCIÓN GENERAL DE BIBLIOTECAS



